Российская академия наук (РАН)
Институт проблем управления им. В.А. Трапезникова
Российской академии наук (ИПУ РАН)
Российский университет дружбы народов (РУДН)
Институт информационных и телекоммуникационных технологий
Болгарской академии наук (София, Болгария)
Национальный исследовательский Томский государственный университет (НИ ТГУ)
Научно-производственное объединение
«Информационные и сетевые технологии» («ИНСЕТ»)

РАСПРЕДЕЛЕННЫЕ КОМПЬЮТЕРНЫЕ И ТЕЛЕКОММУНИКАЦИОННЫЕ СЕТИ: УПРАВЛЕНИЕ, ВЫЧИСЛЕНИЕ, СВЯЗЬ

МАТЕРИАЛЫ XXIV МЕЖДУНАРОДНОЙ НАУЧНОЙ КОНФЕРЕНЦИИ
(20–24 СЕНТЯБРЯ 2021 г., МОСКВА, РОССИЯ)

Под общей редакцией д.т.н. В.М. Вишневского, д.т.н. К.Е. Самуйлова

НАУЧНОЕ ЭЛЕКТРОННОЕ ИЗДАНИЕ

Москва
ИПУ РАН
2021

Russian Academy of Sciences (RAS)
V.A. Trapeznikov Institute of Control Sciences of RAS (ICS RAS)
Peoples' Friendship University of Russia (RUDN University)
Institute of Information and Communication Technologies of Bulgarian Academy of Sciences (Sofia, Bulgaria)
National Research Tomsk State University (NR TSU)
Research and development company
"Information and networking technologies"

DISTRIBUTED COMPUTER AND COMMUNICATION NETWORKS: CONTROL, COMPUTATION, COMMUNICATIONS

PROCEEDINGS OF THE XXIV INTERNATIONAL SCIENTIFIC CONFERENCE
(September 20–24, 2021, Moscow, Russia)

Under the general editorship of D.Sc. V.M. Vishnevskiy, D.Sc. K.E. Samouylov

MOSCOW
ISC RAS
2021

УДК 004.7:004.4].001:621.391:007
ББК 32.973.202:32.968
Р 24

Распределенные компьютерные и телекоммуникационные сети: управление, вычисление, связь (DCCN-2021) = Distributed computer and communication networks: control, computation, communications (DCCN-2021) **:** материалы XXIV Междунар. научн. конфер, 20–24 сент. 2021 г., Москва / под общ. ред. В.М. Вишневского, К.Е. Самуйлова; Ин-т проблем упр. им. В.А. Трапезникова Рос. акад. наук Минобрнауки РФ – Электрон. текстовые дан. (1 файл: 24,9 Мб). – М.: ИПУ РАН, 2021. – 1 электрон. опт. диск (CD-R). – Систем. требования: Pentium 4; 1,3 ГГц и выше; Acrobat Reader 4.0 или выше. – Загл. с экрана. – ISBN 978-5-91450-258-1. – № госрегистрации 0322103543. – Текст : электронный.

В научном электронном издании представлены материалы XXIV Международной научной конференции «Распределенные компьютерные и телекоммуникационные сети: управление, вычисление, связь» по следующим направлениям:

- Алгоритмы и протоколы телекоммуникационных сетей
- Управление в компьютерных и инфокоммуникационных системах
- Анализ производительности, оценка QoS / QoE и эффективность сетей
- Аналитическое и имитационное моделирование коммуникационных систем последующих поколений
- Эволюция беспроводных сетей в направлении 5G;
- Технологии сантиметрового и миллиметрового диапазона радиоволн;
- RFID-технологии и их приложения;
- Интернет вещей и туманные вычисления
- Системы облачного вычисления, распределенные и параллельные системы
- Анализ больших данных
- Вероятностные и статистические модели в информационных системах
- Теория массового обслуживания, теория надежности и их приложения
- Высотные беспилотные платформы и летательные аппараты: управление, передача данных, приложения

В материалах научной конференции DCCN-2021, подготовленных к выпуску к.ф.-м.н. Козыревым Д.В., обсуждены перспективы развития и сотрудничества в этой сфере.

Сборник материалов конференции предназначен для научных работников и специалистов в области управления крупномасштабными системами.

Текст научного электронного издания
воспроизводится в том виде, в котором представлен авторами

Утверждено к изданию Программным комитетом конференции

ISBN 978-5-91450-258-1

Содержание / Contents

UDC: 519.872

Waiting Time Asymptotic Analysis of a M/GI/1 Retrial Queue System

Anatoly Nazarov and Maria Samorodova (✉)

Institute of Applied Mathematics and Computer Science, National Research Tomsk State University, 36 Lenina Avenue, 634050, Tomsk, Russian Federation

samorodova21@gmail.com

Abstract

In this paper we deal with M/GI/1 retrial queueing system and conduct asymptotic analysis of the waiting time. The main result of this analysis is the asymptotic characteristic function of the waiting time distribution under heavy load condition . Also during the analysis the asymptotic distribution of the number of returns of the tagged request to the orbit and the asymptotic distributions of the number of requests in the orbit were obtained.

Keywords: Retrial queue, Asymptotic analysis, Waiting time, Number of returns, Number of Retrials

1. Introduction

In the retrial queue system theory it is hard to deal with the waiting time distribution because of the random order service of customers from the orbit. Different approaches to the investigation of the waiting time in $M/GI/1$ RQ - systems can be found in Artalejo, Gomez-Corral [1], Falin, Fricker [2], Nobel [3], Lee, Kim, Kim [4].

In this work we find the asymptotic characteristic function of the waiting time distribution under heavy load condition. As known the number of returns distribution is a counterpart of waiting time distribution and so we investigate both of them. The asymptotic distributions of the number of requests in the orbit under heavy load condition was also obtained during the analysis.

2. Mathematical model

In this paper we consider a $M/GI/1$ retrial queuing (RQ) system. Requests arrive in a Poisson process with intensity $\tilde{\lambda} = \rho\lambda$. If the server is idle at the moment of request arrival this request occupies the server and the service starts immediately. The service time of request follows a common probability law with arbitrary distribution function $B(x)$.

Served request leaves the system. If the server is busy at the moment of request arrival, the request joins to the orbit. Each request from the orbit after a random delay, that has exponential distribution with rate σ, retries to get accesses to the server. At the retrial moment server again can be idle or busy. In the first case this request occupies the server for a random service time; otherwise, it instantly returns to the orbit for a next random delay.

We define W - the waiting time of the tagged request in the orbit as the length of the interval from the moment the request arrives in the system till the start of the service. Also in our research, the following notations are used: $\tilde{\nu}$ - the number of transitions of the tagged request to the orbit; r - the probability that the server is busy at the moment the request arrives at the system. Obviously, $\tilde{\nu} = 0$ with the probability $(1-r)$, that the request finds the server idle at the moment of the arrival to the system; $\nu(t)$ - the number of returns of the tagged request to the orbit from the moment t until the start of the service. Following this logic, we can write for $\tilde{\nu}$:

$$\tilde{\nu} = \begin{cases} 0, & \text{with probability } (1-r), \\ 1+\nu(t), & \text{with probability } r. \end{cases}$$

Using above notations, the characteristic function for W can be written in the following form:

$$\begin{aligned} G(u) = E\left\{e^{juW}\right\} &= (1-r) + r\sum_{n=0}^{\infty} E\left\{e^{juW}/\tilde{\nu} = 1+n\right\} P\left\{\nu(t)=n\right\} = \\ &= (1-r) + r\sum_{n=0}^{\infty}\left(\frac{\sigma}{\sigma - ju}\right)^{1+n} P\left\{\nu(t)=n\right\}. \end{aligned} \tag{1}$$

The aim of our study is to find asymptotic characteristic function of W under heavy load condition. Obviously, for this purpose it is enough to find the probability r and the probability distribution $P\left\{\nu(t)=n\right\}$ under limiting condition.

3. Kolmogorov's equations

Let's denote by $i(t)$ the number of requests in the orbit at time t and by $k(t)$ - the state of the server at time t:

$$k(t) = \begin{cases} 0, & \text{if the server is idle,} \\ 1, & \text{if the server is busy.} \end{cases}$$

We introduce a process $y(t)$ - the elapsed service time at the moment t for a request standing on the server, and the conditional rate $\mu(x) = \frac{B'(x)}{1-B(x)}$ of service of a request standing on the server in case that the elapsed service time is equal to x.

We do not define process $y(t)$ when the server is free. Thus, we investigate a random process with a variable number of components $\{k(t), i(t), y(t)\}$, which forms a continuous time Markov process. Let's assume that the stationary probability distribution of the states of this process is exist.

Let's denote:

$$P_0(i,t) = P\{k(t) = 0, i(t) = i\},$$
$$P_1(i,y,t) = \frac{\partial P\{k(t) = 1, i(t) = i, y(t) < y\}}{\partial y}.$$

In stationary regime for $P_0(i,t)$ and $P_1(i,y,t)$ we get the following system of equations:

$$\begin{gathered} -(\tilde{\lambda} + i\sigma)P_0(i) + \int_0^\infty P_1(i,y)\mu(y)dy = 0, \\ \frac{\partial P_1(i,y)}{\partial y} = -(\tilde{\lambda} + \mu(y))P_1(i,y) + \tilde{\lambda}P_1(i-1,y), \\ P_1(i,0) = \tilde{\lambda}P_0(i) + (i+1)\sigma P_0(i+1). \end{gathered} \tag{2}$$

Let's introduce steady-state partial characteristic functions:

$$H_0(u) = \sum_{i=0}^{\infty} e^{jui}P_0(i), H_1(u,y) = \sum_{i=0}^{\infty} e^{jui}P_1(i,y), \tag{3}$$

After some actions on (2) using (3) the following system of equations has been composed for $H_0(u)$ and $H_1(u,y)$:

$$\begin{gathered} -\tilde{\lambda}H_0(u) + j\sigma\frac{\partial H_0(u)}{\partial u} + \int_0^\infty H_1(u,y)\mu(y)dy = 0, \\ \frac{\partial H_1(u,y)}{\partial y} = ((e^{ju}-1)\tilde{\lambda} - \mu(y))H_1(u,y), H_1(u,0) = \tilde{\lambda}H_0(u) - j\sigma e^{-ju}\frac{\partial H_0(u)}{\partial u}, \\ \tilde{\lambda}H_1(u) + e^{-ju}j\sigma {H'}_0(u) = 0. \end{gathered} \tag{4}$$

Characteristic function for $\nu(t)$ in stationary mode can be represented as follows:

$$G(u) = E\left\{e^{juv(t)}\right\} = \sum_{i=0}^{\infty}\left[G_0(i,u)P_0(i) + \int_0^\infty G_1(i,u,y)P_1(i,y)dy\right],$$

where $G_0(i,u)$ and $G_1(i,u,y)$ are conditional characteristic functions:

$$G_0(i,u,t) = E\left\{e^{juv(t)}/k(t)=0, i(t)=i\right\},$$
$$G_1(i,u,y,t) = E\left\{e^{juv(t)}/k(t)=1, i(t)=i, y(t)=y\right\}.$$

We compose a system of inverse Kolmogorov equations for the conditional characteristic functions $G_0(i,u,t)$and $G_1(i,u,y,t)$ and taking into account that the system is functioning in a stationary mode, for $G_0(i,u)$ and $G_k(i,u,y)$ we obtain the following system of equations:

$$-(\tilde{\lambda}+i\sigma)G_0(i,u)+\tilde{\lambda}G_1(i,u,0)+(i-1)\sigma G_1(i-1,u,0)+\sigma=0, \tag{5}$$

$$\begin{aligned}\frac{dG_1(i,u,y)}{dy}-(\tilde{\lambda}+\sigma(1-e^{ju})+\mu(y))G_1(i,u,y)+\\ +\tilde{\lambda}G_1(i+1,u,y)+\mu(y)G_0(i,u)=0.\end{aligned} \tag{6}$$

4. Asymptotic analysis

Theorem 1. Conditional asymptotic characteristic function under the condition that $y(t)=y$ for limiting value of the number of requests in the orbit in RQ system $M/GI/1$ in heavy load case ($\rho\to 1$) has the following form:

$$\lim_{\rho\to 1} M\left\{e^{jw(1-\rho)i(t)}\,|y(t)=y\right\}=\frac{1}{b_1}\left(1-\frac{b_2}{2{b_1}^2}jw\right)^{-\left(\frac{2b_1}{\sigma b_2}+1\right)}(1-B(y)) \tag{7}$$

where b_1 and b_2 are the first and the second order moments of service time respectively, $\rho=\lambda b_1$.

Theorem 1 will be used in proof of the next theorem about the asymptotic characteristic function of the number of returns of the tagged request to the orbit.

Integrating (7) over y, we find asymptotic characteristic function under heavy load condition:

$$F(w)=\left(1-\frac{b_2}{2{b_1}^2}jw\right)^{-\left(\frac{2b_1}{\sigma b_2}+1\right)}$$

$F(w)$ has the form of gamma distribution with density:

$$f_{\alpha,\beta}(x)=\frac{\alpha^\beta}{\Gamma(\beta)}x^{\beta-1}e^{-\alpha x}, x\geq 0, \tag{8}$$

where $\alpha=\frac{2{b_1}^2}{b_2}$ - scale parameter, $\beta=\frac{2b_1}{\sigma b_2}+1$ - shape parameter.

The expression for $F(w)$ match with the result obtained in [5], in which in order to find the asymptotic characteristic function under heavy load condition was used the method of residual service time. Accordingly, [5] does not contain the result (7) for the conditional asymptotic characteristic function $F_1(w, y)$, under the condition that the elapsed service time $y(t) = y$.

Theorem 2. The characteristic function $\tilde{G}(u)$ of the limit value of the number $\nu(t)$ of returns of the request to the orbit in RQ system $M/GI/1$ in heavy load case has the following form:

$$\tilde{G}(u) = \int_0^\infty \frac{(1-\rho)/\sigma x b_1}{(1-\rho)/\sigma x b_1 - ju} f_{\alpha,\beta}(x)dx.$$

The density of such distribution will have the following form:

$$\tilde{P}(z) = \int_0^\infty \frac{(1-\rho)}{\sigma x b_1} e^{-\frac{(1-\rho)}{\sigma x b_1} z} f_{\alpha,\beta}(x)dx. \tag{9}$$

5. Asymptotic probability distribution of the waiting time of the customer in the orbit

Using the found distribution density (9), we compose a discrete approximation:

$$P(n) = \tilde{P}(n) \cdot \left(\sum_{m=0}^{\infty} \tilde{P}(m) \right)^{-1},$$

where $P(n)$ - discrete approximation of asymptotic probability distribution of $\nu(t)$ the number of returns of the tagged request to the orbit: Let's substitute the resulting distribution into (1):

$$G(u) = E\left\{e^{juW}\right\} = (1-r) + r\sum_{n=0}^{\infty} E\left\{e^{juW}/\nu = 1+n\right\} P\left\{\nu(t) = n\right\} =$$
$$= (1-r) + r\sum_{n=0}^{\infty} \left(\frac{\sigma}{\sigma - ju}\right)^{1+n} P(n).$$

Thus, we have found the asymptotic characteristic function of the waiting time of the request in the RQ system $M/GI/1$ under heavy load condition. By performing the inverse Fourier transform, one can obtain the asymptotic distribution of the waiting time of the request in the orbit.

6. Conclusion

In this paper was presented an asymptotic analysis of the waiting time and the number of returns of a M/GI/1 retrial queueing system under heavy load condition. As a result the asymptotic characteristic function of the waiting time was found.

REFERENCES

1. Artalejo, J. R., Gómez-Corral, A.: Waiting time analysis of the M/G/1 queue with finite retrial group. Naval Research Logistics (NRL) 54(5), 524 – 529 (2007).
2. Falin, G., Fricker, C.: On the virtual waiting time in an M/G/1 retrial queue. Journal of Applied Probability 28(2), 446-460 (1991).
3. Nobel, R., Tijms, H.: Waiting-time probabilities in the M/G/1 retrial queue. Statistica Neerlandica 60(3), 73–78 (2006).
4. Lee, S. W., Kim, B., Kim, J.: Analysis of the waiting time distribution in M/G/1 retrial queues with two way communication. Annals of Operations Research (Accepted/In press), (2020), https://doi.org/10.1007/s10479-020-03717-2.
5. Moiseeva, E., Nazarov, A.: Asymptotic Analysis of RQ-Systems M/Gi/1 on Heavy Load Condition. In: Proceedings of the IV International Conference ≪Problems of Cybernetics and Informat-ics≫ (PCI 2012), pp. 164–166. IEEE (2012).

УДК: 519.23

Система $BMAP/PH/1$ с нагревом и охлаждением прибора

А.Н. Дудин[1,2], С.А. Дудин[1,2], О.С. Дудина[1,2]

[1]Факультет прикладной математики и информатики, Белорусский государственный университет, проспект Независимости, 4, Минск, Беларусь

[2]Институт прикладной математики и телекоммуникаций, Российский университет дружбы народов, ул. Миклухо-Маклая, 6, Москва, Россия

Аннотация

Исследуется однолинейная система массового обслуживания с групповым марковским потоком, фазовым процессом обслуживания, бесконечным буфером, нагревом и охлаждением прибора. В случае перегрева прибора его работа останавливается, а запрос, который был на обслуживании, теряется. Предложена стратегия включения и отключения прибора, чтобы минимизировать негативные последствия перегрева. Поведение системы описывается многомерной цепью Маркова с одной счетной компонентой. Выписан генератор цепи, найдено стационарное распределение вероятностей состояний системы и основные характеристики производительности.

Ключевые слова: маркированный марковский входной поток, передача данных, нагрев и охлаждение прибора

1. Введение

Работа многих реальных систем, например, серверов дата-центров, сопровождается нагревом сервера. Соответственно, используются определенные механизмы его охлаждения. Понятно, что для максимизации прибыли необходимо максимально использовать доступную мощность прибора. Однако это может привести к перегреву прибора, потере запроса, который был на обслуживании в момент перегрева, и временному вынужденному прекращению обслуживания для охлаждения сервера. Чтобы предотвратить перегрев сервера во время обслуживания, целесообразно остановить новые обслуживания, если температура сервера достигает некоторого уровня (порога). Определенно, этот порог должен

Публикация выполнена при поддержке Программы стратегического академического лидерства РУДН и проекта «Моделирование и оптимизация процессов передачи разнотипной информации в современных телекоммуникационных сетях» ГПНИ «Цифровые и космические технологии, безопасность общества и государства» на 2021-2025 годы

быть меньше критического уровня, достижение которого означает наступление перегрева, но более или менее близок к этому уровню. В противном случае определенная часть емкости прибора не будет использована, и это может привести к потере некоторой прибыли. Если обслуживание может быть отложено или прервано, необходимо указать, когда обслуживание будет возобновлено. Это может быть сделано путем введения еще одного порога. Сервис возобновляется, когда температура прибора падает до этого порога. Очевидно, что этот порог должен быть меньше первого порога. Разница между порогами должна быть не слишком большой. В противном случае, опять же, емкость прибора используется не полностью. Однако, если разница слишком мала, запреты и разрешения на начало новых обслуживаний могут устанавливаться и, затем, отменяться слишком часто, что может быть нежелательным по многим причинам. Поэтому проблема оптимального выбора двух порогов не является тривиальной.

В работе [1] исследована система обслуживания типа $MAP/PH/1$ с нагревом и охлаждением прибора. В данной статье мы существенно обобщаем результаты работы [1] путем рассмотрения группового марковского потока (BMAP) запросов. Известно, групповое поступление запросов является характерной особенностью многих реальных систем. Поэтому рассмотрение системы с BMAP-потоком существенно повышает адекватность модели.

2. Математическая модель

Мы рассматриваем однолинейную систему массового обслуживания с неограниченным буфером.

Структура системы представлена на рисунке 1.

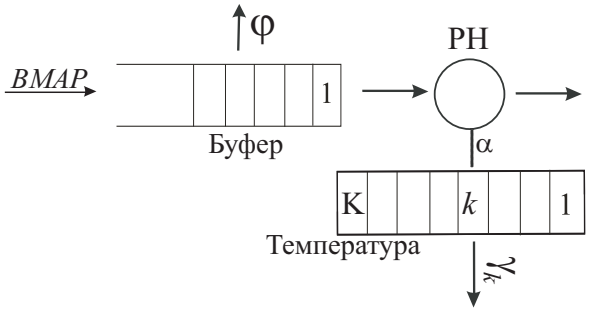


Рис. 1. Структура системы

В систему поступает BMAP-поток запросов. BMAP задается управляющим процессом ν_t, $t \geq 0$, который является неприводимой цепью Маркова с непрерывным временем и конечным пространством состояний $\{0, \ldots, W\}$, и матричной

производящей функцией $D(z) = \sum_{k=0}^{\infty} D_k z^k$, $|z| \leq 1$. Средняя интенсивность поступления запросов λ определяется как $\lambda = \boldsymbol{\theta} D'(1)\mathbf{e}$, где $\boldsymbol{\theta}$ – единственное решение системы $\boldsymbol{\theta} D(1) = \mathbf{0}$, $\boldsymbol{\theta}\mathbf{e} = 1$, и интенсивность λ_b поступления групп запросов определяется как $\lambda_b = \boldsymbol{\theta}(-D_0)\mathbf{e}$. Считаем,что число заявок в группе ограничено параметром L. Более подробное описание BMAP-потока и его свойств можно найти в [2].

Время обслуживания запроса на приборе имеет PH (Phase type) распределение с неприводимым представлением $(\boldsymbol{\beta}, S)$ и управляющим процессом m_t, $t \geq 0$, с пространством состояний $\{1, \ldots, M, M+1\}$, где состояние $M+1$ является поглощающим. Это означает следующее. Время обслуживания интерпретируется как время, за которое цепь Маркова m_t, $t \geq 0$, достигнет поглощающего состояния $M+1$. Переходы цепи m_t, $t \geq 0$, в пространстве состояний $\{1, \ldots, M\}$ задаются субгенератором S, а интенсивности переходов в поглощающее состояние задаются вектором $\boldsymbol{S}_0 = -S\mathbf{e}$. Когда обслуживание начинается, состояние процесса m_t, $t \geq 0$, выбирается в пространстве состояний $\{1, \ldots, M\}$ с вероятностями, являющимися компонентами стохастического вектора-строки $\boldsymbol{\beta}$. Полагаем, что матрица $S + \boldsymbol{S}_0\boldsymbol{\beta}$ неприводима. Интенсивность обслуживания задается как $\mu = -(\boldsymbol{\beta} S^{-1}\mathbf{e})^{-1}$, среднее время обслуживания $b_1 = \mu^{-1}$.

Во время обслуживания прибор нагревается. Считаем, что прибор может работать, когда его температура находится в интервале от K' до K''. Чтобы упростить обозначения, мы будем отслеживать не абсолютную температуру прибора, а превышение температуры над наиболее низким допустимым температурным уровнем. Это означает, что мы предполагаем, что (относительная) температура прибора должна находиться в диапазоне от 0 до K, где $K = K'' - K'$. Когда температура достигает верхнего уровня K, обслуживание запросов становится невозможным, т.е. прибор перегревается. Запрос, во время обслуживания которого происходит перегрев, считается потерянным. Мы предполагаем, что прибор не генерирует тепло, когда он не работает (простаивает или заблокирован). Когда прибор работает, скорость нагрева прибора предполагается равной α градусам в единицу времени, $\alpha > 0$. Параллельно с нагревом прибора он постоянно охлаждается. Предполагаем, что скорость охлаждения равна γ_k, $\gamma_k \geq 0$, когда текущая температура прибора равна k, $k = \overline{0, K}$. Когда прибор перегревается, он прекращает выделение тепла и только охлаждается. Мы предполагаем, что прибор остается заблокированным до тех пор, пока его температура не упадет до уровня (порога) K_1, $K_1 < K$. Мы предполагаем, что запросы, находящиеся в буфере, нетерпеливы. Каждый из этих запросов покидает систему (теряется) через произвольное время независимо от других ожидающих запросов. Это время имеет экспоненциальное распределение с параметром ϕ. Чтобы предот-

вратить возникновение перегрева и потери обслуживаемого запроса, целесообразно не начинать новое обслуживание, когда температура прибора становится высокой. Мы предполагаем, что зафиксирован некоторый порог K_2 такой, что $K_1 < K_2 \leq K$. Прибор не может начать новое обслуживание, если его температура равна или превышает K_2. Но текущее обслуживание при превышении порога K_2 не прерывается, пока прибор не перегреется, т.е. его температура станет равной K. Если это обслуживание успешно завершено, а прибор не перегрет, он остается заблокированным и не начинает новое обслуживание, пока его температура не упадет до K_1.

Очевидно, что показатели производительности системы зависят от выбора пары порогов (K_1, K_2), и наша первая цель – предоставить способ вычисления значений этих показателей для любой фиксированной пары порогов.

3. Процесс изменения состояний системы и его анализ

Пусть критическая температура K и пороги K_1 и K_2 зафиксированы, $0 \leq K_1 < K_2 \leq K$.

Легко видеть, что поведение рассматриваемой системы может быть описано следующей регулярной неприводимой цепью Маркова с непрерывным временем

$$\xi_t = \{n_t, r_t, k_t, \nu_t, m_t\},\ t \geq 0,$$

где в момент $t,\ t \geq 0$,

n_t – число запросов в буфере, $n_t \geq 0$;

r_t, $r_t = \overline{0,2}$, – состояние прибора: $r_t = 0$, если прибор свободен, $r_t = 1$, если прибор занят, и $r_t = 2$, если прибор заблокирован;

k_t – температура прибора, $k_t = \overline{0,K}$;

ν_t – состояние управляющего процесса BMAP, $\nu_t = \overline{0,W}$;

m_t – состояние управляющего процесса PH процесса обслуживания, $m_t = \overline{1,M}$.

Цепь Маркова ξ_t, $t \geq 0$, имеет следующее пространство состояний:

$$\Big(\{0,0,k,\nu\},\ k = \overline{0,K_2-1}\Big) \bigcup \Big(\{n,1,k,\nu,m\},\ n \geq 0,\ k = \overline{0,K-1},\ m = \overline{1,M}\Big) \bigcup$$

$$\Big(\{n,2,k,\nu\},\ n \geq 0,\ k = \overline{K_1+1,K}\Big),\ \nu = \overline{0,W}.$$

Обозначим через Q генератор цепи Маркова ξ_t. Из введенного выше порядка компонент цепи следует, что Q – это матрица, состоящая из блоков $Q_{n,n'},\ n, n' \geq 0$, которые определяют интенсивности переходов из состояний, имеющих значение n компоненты n_t, в состояния, имеющие значение n' этой компоненты.

Теорема 1. *Инфинитезимальный генератор Q цепи Маркова ξ_t, $t \geq 0$, имеет следующую блочно-верхне-Хессенберговую структуру:*

$$Q = \begin{pmatrix} Q_{0,0} & Q_{0,1} & Q_{0,2} & Q_{0,3} & Q_{0,4} & \cdots \\ Q_{1,0} & Q_{1,1} & Q_{1,2} & Q_{1,3} & Q_{1,4} & \cdots \\ O & Q_{2,1} & Q_{2,2} & Q_{2,3} & Q_{2,4} & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \end{pmatrix}.$$

Здесь у блоков $(Q_{0,0}^{r,r'})_{r,r'=\overline{0,2}}$ матрицы $Q_{0,0}$ диагональные элементы отрицательны и определяют с точностью до знака интенсивности выхода цепи Маркова ξ_t из соответствующих состояний, а недиагональные элементы определяют интенсивности переходов, которые не подразумевают появление запросов в пустом буфере. Эти блоки имеют следующий вид:

$$Q_{0,0}^{0,0} = I_{K_2} \otimes D_0 + (E_{K_2}^- - C_{K_2}) \otimes I_{\bar{W}},\ Q_{0,0}^{0,1} = I_{K_2,K} \otimes D_1 \otimes \boldsymbol{\beta},\ Q_{0,0}^{0,2} = O_{K_2\bar{W},(K-K_1)\bar{W}},$$

$$Q_{0,0}^{1,0} = I_{K,K_2} \otimes I_{\bar{W}} \otimes \mathbf{S}_0,\ Q_{0,0}^{1,1} = I_K \otimes D_0 \oplus S + (E_K^- - C_K + \alpha(E^+ - I_K)) \otimes I_{\bar{W}M},$$

$$Q_{0,0}^{1,2} = \bar{I}_{K,K-K_1} \otimes I_{\bar{W}} \otimes \mathbf{S}_0 + \alpha\hat{I} \otimes I_{\bar{W}} \otimes \mathbf{e}_M,\ Q_{0,0}^{2,0} = \gamma_{K_1+1}\tilde{I}_{K-K_1,K_2} \otimes I_{\bar{W}},$$

$$Q_{0,0}^{2,1} = O_{(K-K_1)\bar{W},K\bar{W}M},\ Q_{0,0}^{2,2} = I_{K-K_1} \otimes D_0 + (\tilde{E} - \tilde{C}) \otimes I_{\bar{W}}.$$

Матрица $Q_{0,l}$, $l = \overline{1,L}$, элементы которой определяют интенсивности переходов в случае, когда запросы поступает в момент, когда буфер пуст, имеет вид

$$Q_{0,l} = \begin{pmatrix} Q_{0,l}^{0,1} & O_{K_2\bar{W},(K-K_1)\bar{W}} \\ Q_{0,l}^{1,1} & O_{K\bar{W}M,(K-K_1)\bar{W}} \\ O_{(K-K_1)\bar{W},K\bar{W}M} & Q_{0,l}^{2,2} \end{pmatrix},$$

где

$$Q_{0,l}^{0,1} = I_{K_2,K} \otimes D_{l+1} \otimes \boldsymbol{\beta},\ l = \overline{1,L-1},\ Q_{0,L}^{0,1} = O_{K_2\bar{W},K\bar{W}M},\ Q_{0,l}^{1,1} = I_K \otimes D_l \otimes I_M,\ l = \overline{1,L},$$

$$Q_{0,l}^{2,2} = I_{K-K_1} \otimes D_l,\ l = \overline{1,L}.$$

Матрица $Q_{1,0}$, элементы которой определяют интенсивности переходов в случае, когда единственный запрос, находящийся в буфере, покидает буфер (из-за нетерпеливости или начала обслуживания), имеет вид

$$Q_{1,0} = \begin{pmatrix} O_{K\bar{W}M,K_2\bar{W}} & Q_{1,0}^{1,1} & O_{(K-K_1)\bar{W},K_2\bar{W}} \\ O_{(K-K_1)\bar{W},K_2\bar{W}} & Q_{1,0}^{2,1} & Q_{1,0}^{2,2} \end{pmatrix},$$

где

$$Q_{1,0}^{1,1} = \phi I_{K\bar{W}M} + B \otimes I_{\bar{W}} \otimes \mathbf{S}_0\boldsymbol{\beta},\ Q_{1,0}^{2,1} = \gamma_{K_1+1}\tilde{I}_{K-K_1,K} \otimes I_{\bar{W}} \otimes \boldsymbol{\beta},\ Q_{1,0}^{2,2} = \phi I_{(K-K_1)\bar{W}}.$$

Блоки $(Q_{n,n}^{r,r'})_{r,r'=\overline{1,2}}$ *матрицы* $Q_{n,n}$, $n \geq 1$, *диагональные элементы которой отрицательны и определяют, вплоть до знака, интенсивности выхода цепи Маркова* ξ_t *из соответствующих состояний, когда число запросов в буфере равно* n, $n \geq 1$, *а недиагональные элементы определяют интенсивность переходов, которые не влекут изменение числа запросов в буфере, имеют следующий вид:*

$$Q_{n,n}^{1,1} = I_K \otimes D_0 \oplus S + (E_K^- - C_K + \alpha(E^+ - I_K) - n\phi I_K) \otimes I_{\bar{W}M},$$

$$Q_{n,n}^{1,2} = \bar{I}_{K,K-K_1} \otimes I_{\bar{W}} \otimes \mathbf{S}_0 + \alpha\hat{I} \otimes I_{\bar{W}} \otimes \mathbf{e}_M,$$

$Q_{n,n}^{2,1} = O_{(K-K_1)\bar{W},K\bar{W}M}$, $Q_{n,n}^{2,2} = I_{K-K_1} \otimes D_0 + (\tilde{E} - \tilde{C}) \otimes I_{\bar{W}} - n\phi I_{(K-K_1)\bar{W}}$, $n \geq 1$.

Блоки $(Q_{n,n+l}^{r,r'})_{r,r'=\overline{1,2}}$ *матрицы* $Q_{n,n+l}$, $n \geq 1$, $l = \overline{1,L}$, *элементы которой определяют интенсивности увеличения числа запросов в буфере с* n *до* $n+l$, *имеют следующий вид:*

$$Q_{n,n+l}^{1,1} = I_K \otimes D_l \otimes I_M,\ Q_{n,n+l}^{1,2} = O_{K\bar{W}M,(K-K_1)\bar{W}},\ Q_{n,n+l}^{2,1} = O_{(K-K_1)\bar{W},K\bar{W}M},$$

$$Q_{n,n+l}^{2,2} = I_{K-K_1} \otimes D_l,\ n \geq 1,\ l = \overline{1,L}.$$

Ненулевые блоки $(Q_{n,n-1}^{r,r'})_{r,r'=\overline{1,2}}$ *матрицы* $Q_{n,n-1}$, $n \geq 2$, *элементы которой определяют интенсивности уменьшения числа запросов в буфере с* n *до* $n-1$, *имеют следующий вид:*

$$Q_{n,n-1}^{1,1} = n\phi I_{K\bar{W}M} + B \otimes I_{\bar{W}} \otimes \mathbf{S}_0\boldsymbol{\beta},\ Q_{n,n-1}^{2,1} = \gamma_{K_1+1}\tilde{I}_{K-K_1,K} \otimes I_{\bar{W}} \otimes \boldsymbol{\beta},$$

$$Q_{n,n-1}^{2,2} = n\phi I_{(K-K_1)\bar{W}},\ n \geq 2.$$

Здесь E_l^- *– квадратная матрица порядка* l *с нулевыми элементами, кроме элементов* $(E_l^-)_{k,k-1} = \gamma_k$, $k = \overline{1,l-1}$; C_l *– квадратная матрица порядка* l *с нулевыми элементами, кроме элементов* $(C_l)_{k,k} = \gamma_k$, $k = \overline{1,l-1}$; $I_{K_2,K}$ *– матрица порядка* $K_2 \times K$ *с нулевыми элементами, кроме элементов* $(I_{K_2,K})_{n,n} = 1$, $n = \overline{0,K_2-1}$; I_{K,K_2} *– матрица порядка* $K \times K_2$ *с нулевыми элементами, кроме элементов* $(I_{K,K_2})_{n,n} = 1$, $n = \overline{0,K_2-1}$; E^+ *– квадратная матрица порядка* K *с нулевыми элементами, кроме элементов* $(E^+)_{k,k+1} = 1$, $k = \overline{0,K-2}$; $\bar{I}_{K,K-K_1}$ *– матрица порядка* $K \times (K-K_1)$ *с нулевыми элементами, кроме элементов* $(\bar{I}_{K,K-K_1})_{n,n-K_1-1} = 1$, $n = \overline{K_2,K-1}$; $\hat{I}$ *– матрица порядка* $K \times (K -$

K_1) с нулевыми элементами, кроме элемента $(\hat{I})_{K-1,K-K_1-1} = 1$; $\tilde{I}_{K-K_1,K_2}$ – матрица порядка $(K-K_1) \times K_2$ с нулевыми элементами, кроме элемента $(\tilde{I}_{K-K_1,K_2})_{0,K_1} = 1$; $\tilde{E}$ – квадратная матрица порядка $K-K_1$ с нулевыми элементами, кроме элементов $(\tilde{E})_{k,k-1} = \gamma_{K_1+k+1}$, $k = \overline{1, K-K_1-1}$; $\tilde{C}$ – квадратная матрица порядка $K-K_1$ с нулевыми элементами, кроме элементов $(\tilde{C})_{k,k} = \gamma_{K_1+k+1}$, $k = \overline{0, K-K_1-1}$; B – квадратная матрица порядка K с нулевыми элементами, кроме элементов $(B)_{k,k} = 1$, $k = \overline{0, K_2-1}$; $\tilde{I}_{K-K_1,K}$ – матрица порядка $(K-K_1) \times K$ с нулевыми элементами, кроме элемента $(\tilde{I}_{K-K_1,K})_{0,K_1} = 1$; $\otimes$ и $\oplus$ обозначают операции Кронекерова произведения и суммы матриц.

Доказательство теоремы проводится посредством анализа различных сценариев поведения системы в моменты изменения состояний управляющих процессов поступления и обслуживания, изменения температуры прибора из-за нагрева и охлаждения, ухода запросов из-за нетерпеливости.

Теорема 2. *Стационарное распределение цепи Маркова ξ_t существует при любых значениях параметров системы.*

Утверждение теоремы вытекает из того факта, что запросы, находящиеся в буфере, предполагаются нетерпеливыми ($\phi > 0$). Строгое доказательство теоремы 2 можно провести, используя результаты из [4].

Обозначим через $\boldsymbol{\pi}(n, r, k)$ вектор-строку стационарных вероятностей состояний цепи, имеющих значение (n, r, k) первых трех компонент, упорядоченных в описанном выше порядке. Также обозначим:

$$\boldsymbol{\pi}(0,0) = (\boldsymbol{\pi}(0,0,0), \dots, \boldsymbol{\pi}(0,0,K_2-1)),\ \boldsymbol{\pi}(n,1) = (\boldsymbol{\pi}(n,1,0), \dots, \boldsymbol{\pi}(n,1,K-1)),$$

$$\boldsymbol{\pi}(n,2) = (\boldsymbol{\pi}(n,2,K_1+1), \dots, \boldsymbol{\pi}(n,2,K)),\ n \geq 0,$$

$$\boldsymbol{\pi}(0) = (\boldsymbol{\pi}(0,0), \boldsymbol{\pi}(0,1), \boldsymbol{\pi}(0,2)),\ \boldsymbol{\pi}(n) = (\boldsymbol{\pi}(n,1), \boldsymbol{\pi}(n,2)),\ n \geq 1.$$

Поскольку пространство состояний цепи Маркова ξ_t бесконечно и генератор Q этой цепи не имеет тёплицево-подобной структуры, проблема вычисления векторов $\boldsymbol{\pi}(n),\ n \geq 0$, является непростой. Вместе с тем, цепи с генератором такого типа были проанализированы в [3], и алгоритм, разработанный в этой статье, позволяет вычислять эти векторы.

4. Характеристики производительности

Среднее число запросов в буфере $N = \sum_{n=1}^{\infty} n\boldsymbol{\pi}(n)\mathbf{e}$.

Средняя температура прибора $T = \sum\limits_{k=1}^{K_2-1} k\boldsymbol{\pi}(0,0,k)\mathbf{e} + \sum\limits_{n=0}^{\infty}\sum\limits_{k=1}^{K-1} k\boldsymbol{\pi}(n,1,k)\mathbf{e} + \sum\limits_{n=0}^{\infty}\sum\limits_{k=K_1+1}^{K} k\boldsymbol{\pi}(n,2,k)\mathbf{e}$.

Вероятность того, что прибор простаивает в произвольный момент времени $P_{idle} = \boldsymbol{\pi}(0,0)\mathbf{e}$.

Вероятность того, что прибор занят в произвольный момент времени $P_{busy} = \sum\limits_{n=0}^{\infty} \boldsymbol{\pi}(n,1)\mathbf{e}$.

Вероятность того, что прибор заблокирован в произвольный момент времени $P_{block} = \sum\limits_{n=0}^{\infty} \boldsymbol{\pi}(n,2)\mathbf{e}$.

Вероятность того, что произвольный запрос будет потерян из-за нетерпеливости $P_{imp} = \frac{\phi N}{\lambda}$.

Интенсивность потока обслуженных запросов $\lambda_{out} = \sum\limits_{n=0}^{\infty} \boldsymbol{\pi}(n,1)(\mathbf{e}_{K\bar{W}} \otimes \mathbf{S}_0)$.

Вероятность того, что произвольный запрос будет потерян из-за перегрева прибора $P_{overheating} = \alpha\lambda^{-1} \sum\limits_{n=0}^{\infty} \boldsymbol{\pi}(n,1,K-1)\mathbf{e}$.

Вероятность того, что произвольный запрос будет потерян $P_{loss} = P_{imp} + P_{overheating} = 1 - \frac{\lambda_{out}}{\lambda}$.

ЛИТЕРАТУРА

1. Dudina, Olga, and Alexander Dudin. "Optimization of queueing model with server heating and cooling."Mathematics 7.9 (2019): 768.
2. Dudin A., Klimenok V. I., Vishnevsky V. M. The Theory of Queuing Systems with Correlated Flows. – Cham : Springer, 2019. .
3. Dudin, Sergei, et al. "Effective algorithm for computation of the stationary distribution of multi-dimensional level-dependent Markov chains with upper block-Hessenberg structure of the generator."Journal of Computational and Applied Mathematics 366 (2020): 112425.
4. Klimenok V.I., Dudin A.N. "Multi-dimensional asymptotically quasi-Toeplitz Markov chains and their application in queueing theory."Queueing Systems 54 (2006): 245-259.

УДК: 519.23

Система $MAP/PH/1$ с автономным ограниченным обслуживанием без прерывания

А.Н. Дудин[1,2], С.А. Дудин[1], О.С. Дудина[1]

[1]Факультет прикладной математики и информатики, Белорусский государственный университет, проспект Независимости, 4, Минск, Беларусь

[2]Институт прикладной математики и телекоммуникаций, Российский университет дружбы народов, ул. Миклухо-Маклая, 6, Москва, Россия

Аннотация

Исследуется однолинейная система массового обслуживания с марковским потоком, фазовым процессом обслуживания, отдыхами прибора, автономным ограниченным обслуживанием без прерывания и нетерпеливыми запросами. Поведение системы описывается многомерной цепью Маркова с одной счетной компонентой. Выписан генератор цепи, что позволяет найти стационарное распределение вероятностей состояний системы и основные характеристики ее производительности.

Ключевые слова: марковский входной поток, распределение фазового типа, отдыхи прибора, ограниченное обслуживание

1. Введение

В данной работе исследуется система массового обслуживания с отдыхами прибора. Такие системы интересны как сами по себе, так и с точки зрения применения их для анализа так называемых поллинговых систем массового обслуживания (иногда называемых системами с циклическим опросом очередей), которые являются адекватными математическими моделями процессов передачи информации в сетях связи со случайным множественным доступом, в частности, в беспроводных сотовых сетях, и поэтому их исследованию в настоящее время уделяется довольно много внимания, см., например, [1]. Близкая к рассматриваемой здесь модель системы с отдыхами прибора была недавно рассмотрена в работе [2]. Особенностью той модели является то, что дисциплина, при которой отдых берется при опустошении прибора или по истечении случайного времени непрерывной работы, имеющего заданное распределение вероятностей, дополняется условием, что обслуживание запроса, во время обслуживания которого

Публикация выполнена при поддержке Программы стратегического академического лидерства РУДН и проекта 1.6.01.2 ГПНИ «Конвергенция-2025» на 2021-2025 годы

время непрерывной работы закончилось, не может прерываться. В данной работе постановка задачи видоизменена таким образом, что пока время непрерывной работы прибора не истечет, следующий отдых не может начаться, даже если в системе не остается запросов. При опустошении очереди, прибор ждет поступление следующих запросов. Такая постановка описывает так называемые автономные системы, примером которых является работа светофора: при опустошении очереди автомобилей на регулируемом перекрестке переключение на другой свет не производится, пока не истечет заданное время, для ссылок см., например, работу [3].

2. Математическая модель

Мы рассматриваем однолинейную систему массового обслуживания с неограниченным буфером. Входной поток описывается как MAP (Markov Arrival Process). Прибытие запросов в MAP управляется неприводимой цепью Маркова ν_t, $t \geq 0$, с конечным пространством состояний $\{0, 1, ..., W\}$. Время пребывания цепи ν_t, $t \geq 0$, в состоянии ν имеет показательное распределение с параметром $\lambda_\nu, \nu = \overline{0, W}$. По истечении этого времени с вероятностью $p_k(\nu, \nu')$ процесс ν_t переходит в состояние ν', и k запросов, $k = 0, 1$, поступают в систему. Интенсивности перехода цепи Маркова из одного состояния в другое с генерацией k запросов объединяются в матрицы D_k, $k = 0, 1$, размера $\overline{W} \times \overline{W}$, где $\overline{W} = W + 1$. Матрица $D(1) = D_0 + D_1$ является инфинитезимальным генератором процесса ν_t, $t \geq 0$. Вектор стационарного распределения вероятностей $\boldsymbol{\theta}$ этого процесса вычисляется как единственное решение системы $\boldsymbol{\theta} D(1) = \mathbf{0}$, $\boldsymbol{\theta}\mathbf{e} = 1$. Здесь и далее $\mathbf{0}$ – нулевая вектор-строка, а $\mathbf{e}$ – единичный вектор-столбец соответствующего размера. Интенсивность потока λ определена как $\lambda = \boldsymbol{\theta} D_1 \mathbf{e}$. Более подробное описание MAP потока, его свойств и полезности для описания реальных потоков можно найти, например, в [4].

Обслуживающий прибор чередует рабочие периоды с периодами отдыха. Длительность периода отдыха имеет распределение фазового типа (PH) с непрерывным представлением $(\boldsymbol{\gamma}, \Gamma)$. Это означает следующее. Длительность периода отдыха определяется поведением процесса ξ_t, $t \geq 0$, который является цепью Маркова с непрерывным временем с пространством состояний $\{1, \ldots, R, R+1\}$. Начальное состояние процесса ξ_t, $t \geq 0$, в момента начала отдыха определяется в соответствии с вероятностями, заданными компонентами вектора-строки $\boldsymbol{\gamma} = (\gamma_1, \ldots, \gamma_R)$. Интенсивности переходов процесса ξ_t, $t \geq 0$, внутри множества $\{1, \ldots, R\}$, которые не приводят к завершению отдыха, определены как элементы неприводимой матрицы Γ размерности R. Интенсивности перехода в поглощающее состояние $R + 1$, которые приводят к завершению отдыха, определяются как элементы вектора-столбца $\boldsymbol{\Gamma}_0 = -\Gamma\mathbf{e}$. Функция распределения

длительности отдыха имеет вид $1 - \boldsymbol{\gamma} e^{\Gamma x}\mathbf{e}$. Преобразование Лапласа-Стилтьеса функции распределения имеет вид $\boldsymbol{\gamma}(sI - \Gamma)^{-1}\boldsymbol{\Gamma}_0$, $Re\ s > 0$. Средняя продолжительность отдыха определяется как $v_1 = \boldsymbol{\gamma}\,(-\Gamma)^{-1}\,\mathbf{e}$. Более подробное описание PH распределения и его свойств можно также найти в [4]. Во время отдыха прибор не производит обслуживания запросов. В применении систем с отдыхами для анализа поллинговых систем отдых прибора интерпретируется как время обслуживания прибором запросов из других очередей.

После завершения периода отдыха начинается период обслуживания. Длительность этого периода имеет распределение фазового типа (PH) с неприводимым представлением $(\boldsymbol{\tau}, T)$. Управляющий процесс χ_t, $t \geq 0$, периода обслуживания является цепью Маркова с непрерывным временем с пространством непоглощающих состояний $\{1, \ldots, K\}$. Средняя продолжительность τ_1 периода обслуживания определяется как $\tau_1 = \boldsymbol{\tau}(-T)^{-1}\mathbf{e}$. В отличие от модели, изученной в [2], прибор является автономным и не уходит на отдых, если во время периода обслуживания все запросы получили обслуживание и начинается простой прибора. Прибор ждет до тех пор, пока не закончится рабочий период (после чего он уходит на отдых) или не придет новый запрос. Такой запрос немедленно начинает обслуживание. Если во время обслуживания какого-либо запроса рабочий период не закончился, то, в отличие от стандартного обслуживания с ограничением периода обслуживания, текущее обслуживание не прерывается. Отдых начнется только после завершения этого обслуживания. Состояние прибора до завершения этого обслуживания будем называть пролонгированным периодом обслуживания.

Время обслуживания произвольного запроса имеет распределение фазового типа с неприводимым представлением $(\boldsymbol{\beta}, S)$. Управляющий процесс обслуживания η_t, $t \geq 0$, имеет пространство непоглощающих состояний $\{1, \ldots, M\}$. Среднее время обслуживания задается формулой $b_1 = \boldsymbol{\beta}(-S)^{-1}\mathbf{e}$. Отметим, что для упрощения обозначений мы предполагаем, что в момент окончания обслуживания запроса, если прибор не уходит на отдых, мы устанавливаем состояние управляющего процесса обслуживания для следующего запроса (в соответствии с вектором $\boldsymbol{\beta}$), даже если в системе нет запросов. Это состояние предполагается не изменяющимся до начала следующего обслуживания.

Предполагается, что запросы, ожидающие начала обслуживания, являются нетерпеливыми. Каждый такой запрос покидает систему, не обслужившись, независимо от других запросов, если его время ожидания превышает экспоненциально распределенное время с параметром α_r, где $r = 0$, если в данный момент прибор находится на отдыхе, $r = 1$, если идет период обслуживания, и $r = 2$, если идет пролонгированный период обслуживания.

Нашей целью является проанализировать стационарное поведение описанной модели массового обслуживания.

3. Процесс изменения состояний системы и его анализ

Пусть в произвольный момент времени $t,\ t \geq 0$,

- i_t – число запросов в системе, $i_t \geq 0$;
- r_t – текущее состояние прибора: $r_t = 0$, если прибор находится в состоянии отдыха, $r_t = 1$, если прибор находится в периоде обслуживания, и $r_t = 2$, если прибор находится в пролонгированном периоде обслуживания;
- ν_t – состояние управляющего процесса MAP потока, $\nu_t = \overline{0, W}$;
- m_t – текущая фаза управляющего процесса ξ_t времени отдыха, если $r_t = 0$; $m_t = (\chi_t, \eta_t)$ (пара текущих фаз управляющих процессов периода обслуживания и времени обслуживания), если $r_t = 1$; и $m_t = \eta_t$, если $r_t = 2$.

Нетрудно заметить, что пространство состояний многомерного процесса $\zeta_t = \{i_t,\ r_t,\ \nu_t,\ m_t\},\ \ t \geq 0$, имеет вид

$$\Omega = \{(i,\ 0, \nu, \xi),\ i \geq 0,\ \nu = \overline{0, W},\ \xi = \overline{1, R}\}$$

$$\bigcup \{(i,\ 1, \nu, \chi, \eta),\ i \geq 0,\ \nu = \overline{0, W},\ \chi = \overline{1, K},\ \eta = \overline{1, M}\}$$

$$\bigcup \{(i,\ 2, \nu, \eta),\ i \geq 1,\ \nu = \overline{0, W},\ \eta = \overline{1, M}\}$$

и этот процесс является неприводимой цепью Маркова с непрерывным временем с одной счетной компонентой i_t и несколькими конечными компонентами.

Для анализа цепи Маркова ζ_t мы должны получить ее инфинитезимальный генератор. Обозначим этот генератор $\mathbf{Q}$. Диагональные элементы генератора являются отрицательными. Модуль каждого диагонального элемента определяет интенсивность выхода цепи Маркова из соответствующего состояния. Недиагональные элементы являются неотрицательными и определяют интенсивности переходов цепи Маркова в ее пространстве состояний.

Чтобы упростить структуру генератора $\mathbf{Q}$, перенумеруем состояния цепи Маркова ζ_t в лексикографическом порядке и скомпонуем все состояния цепи имеющие значения (i, r) первых двух компонент в *под-уровень* (i, r). Под-уровень $(i, 0)$ содержит $\overline{W}R$ состояний, под-уровень $(i, 1)$ содержит $\overline{W}KM$ состояний и под-уровень$(i, 2)$ содержит $\overline{W}M$ состояний. Затем, мы скомпонуем все под-уровни $(i, r),\ r = 0, 1, 2$, в *уровень* i.

Лемма. Генератор $\mathbf{Q}$ цепи Маркова ζ_t имеет блочную трехдиагональную структуру:

$$\mathbf{Q} = \begin{pmatrix} \mathbf{Q}_{0,0} & \mathbf{Q}_{0,1} & O & O & \dots \\ \mathbf{Q}_{1,0} & \mathbf{Q}_{1,1} & \mathbf{Q}_{1,2} & O & \dots \\ O & \mathbf{Q}_{2,1} & \mathbf{Q}_{2,2} & \mathbf{Q}_{2,3} & \dots \\ \vdots & \vdots & \vdots & \vdots & \ddots \end{pmatrix},$$

где ненулевые блоки $\mathbf{Q}_{i,j}$, определяющие интенсивности переходов цепи из уровня i на уровень j, $j = \max\{0, i-1\}, i, i+1$, задаются следующим образом:

$$\mathbf{Q}_{0,0} = \begin{pmatrix} D_0 \oplus \Gamma & I_{\overline{W}} \otimes \mathbf{\Gamma}_0 \otimes \boldsymbol{\tau} \otimes \boldsymbol{\beta} \\ I_{\overline{W}} \otimes \mathbf{T}_0 \otimes \boldsymbol{\gamma} & D_0 \oplus T \otimes I_M \end{pmatrix},$$

$$\mathbf{Q}_{0,1} = \begin{pmatrix} D_1 \otimes I_R & O & O \\ O & D_1 \otimes I_K \otimes I_M & O \end{pmatrix},$$

$$\mathbf{Q}_{1,0} = \begin{pmatrix} \alpha_0 I_{\overline{W}} \otimes I_R & O \\ O & I_{\overline{W}} \otimes I_K \otimes \mathbf{S_0}\boldsymbol{\beta} \\ I_{\overline{W}} \otimes \mathbf{S_0} \otimes \boldsymbol{\gamma} & O \end{pmatrix},$$

$$\mathbf{Q}_{i,i-1} = \begin{pmatrix} O & O & O \\ O & I_{\overline{W}} \otimes I_K \otimes \mathbf{S}_0\boldsymbol{\beta} & O \\ I_{\overline{W}} \otimes \mathbf{S}_0 \otimes \boldsymbol{\gamma} & O & O \end{pmatrix}$$

$$+\mathrm{diag}\{i\alpha_0 I_{\overline{W}R}, (i-1)\alpha_1 I_{\overline{W}KM}, (i-1)\alpha_2 I_{\overline{W}M}\},\ i \geq 2,$$

$$\mathbf{Q}_{i,i} = \begin{pmatrix} D_0 \oplus \Gamma & I_{\overline{W}} \otimes \mathbf{\Gamma}_0 \otimes \boldsymbol{\tau} \otimes \boldsymbol{\beta} & O \\ O & D_0 \oplus T \oplus S & I_{\overline{W}} \otimes \mathbf{T}_0 \otimes I_M \\ O & O & D_0 \oplus S \end{pmatrix}$$

$$-\mathrm{diag}\{i\alpha_0 I_{\overline{W}R}, (i-1)\alpha_1 I_{\overline{W}KM}, (i-1)\alpha_2 I_{\overline{W}M}\},\ i \geq 1,$$

$$\mathbf{Q}_{i,i+1} = \begin{pmatrix} D_1 \otimes I_R & O & O \\ O & D_1 \otimes I_K \otimes I_M & O \\ O & O & D_1 \otimes I_M \end{pmatrix},\ i \geq 1.$$

Здесь, I – единичная матрица, и O – нулевая матрица соответствующего размера, $\mathrm{diag}\{\dots\}$ обозначает диагональную матрицу с диагональными блоками, указанными в скобках, $\otimes, \oplus$ – символы Кронекерова произведения и суммы матриц соответственно, см. [5], $\mathbf{S}_0 = -S\mathbf{e}$, $\mathbf{T}_0 = -T\mathbf{e}$.

Проверяя соответствие цепи Маркова ζ_t определению, данному в [6], можно убедится, что цепь Маркова ζ_t является асимптотически квазитеплицевой цепью Маркова. Используя этот факт и результаты из [6], можно доказать, что, если

хотя бы один из параметров $\alpha_0, \alpha_1, \alpha_2$ отличен от нуля, то цепь Маркова ζ_t является эргодичной при любых значениях параметров системы. Стационарное распределение вероятностей цепи Маркова ζ_t может быть вычислено с использованием численно устойчивых алгоритмов, разработанных в [6] и [7]. Это дает возможность вычислять различные характеристики производительности системы и решать различные задачи оптимизации. Эффективность применения этих алгоритмов для анализа подобных систем с отдыхами и их оптимизации была продемонстрирована в [2].

4. Заключение

В данной работе мы изучили модификацию системы массового обслуживания с отдыхами, изученной в [2], на практически важный случай, когда чередование периодов работы и отдыха строго контролируется. Досрочное начало отдыхов и прерывание текущего обслуживания не допускаются.

ЛИТЕРАТУРА

1. V. Vishnevsky, O. Semenova, O. Polling Systems and Their Application to Telecommunication Networks. Mathematics, 2021. 9(2). 117.
2. Kim C.S., Dudin A., Dudina O., Klimenok V. Analysis of queueing system with non-preemptive time limited service and impatient customers // Methodology and Computing in Applied Probability. 2020. V. 22. no. 2. P. 401-432.
3. de Haan R., Boucherie R. J., van Ommeren J. K. A polling model with an autonomous server //Queueing Systems. 2009. V. 62. No. 3. P. 279-308.
4. Dudin A.N., Klimenok V.I., Vishnevsky V.M. The theory of queuing systems with correlated flows. - Springer Nature. 2019. 431 P.
5. A. Graham, Kronecker products and matrix calculus with applications, Ellis Horwood, Cichester, 1981.
6. Klimenok V.I., Dudin A.N. "Multi-dimensional asymptotically quasi-Toeplitz Markov chains and their application in queueing theory."Queueing Systems 54 (2006): 245-259.
7. Dudin, Sergei, et al. "Effective algorithm for computation of the stationary distribution of multi-dimensional level-dependent Markov chains with upper block-Hessenberg structure of the generator."Journal of Computational and Applied Mathematics 366 (2020): 112425.

UDC: 004.94

Software Packages for Teaching Queueing Theory

János Sztrik[1], Zoltán Szilágyi [1], Csanád Kölcsei[1]

[1]University of Debrecen, Debrecen 4032, Hungary

sztrik.janos@inf.unideb.hu, zoltan.szilagyi.cse@gmail.com, kcsanad98@gmail.com

Abstract

The goal of the present paper is to give a short review of software packages for teaching Queueing Theory and to introduce an application called Queueing Systems Assistance (QSA). The software is integrated into a lecture note with the aim to calculate and visualize the main performance measures. In addition, it helps to minimize a quite general mean total cost per unit time with linear objective function. Several examples are given to illustrate the advantage of the graphical module included in the package.

Keywords: modeling, queueing, teaching, software, visualization

1. Introduction

The teaching of Queueing Theory (QT) needs innovation and new methods to attract the attention of the students. The field of applications has changed a lot in the past years and I am convinced that more and more students and practitioners need to use the methods and models of QT. The development of computational possibilities has greatly contributed to a better understanding of the theory.

In his lecture note Sztrik [1] discussed a number of basic queueing models that have proved to be useful in analysing a wide variety of stochastic service systems. The author feels that there is a need for such a treatment in view of the increased use of queueing models in modern technology. Actually, the application of queueing theory in the performance analysis of computer and communication systems has stimulated much practically oriented research on computational aspects of queueing models.

Furthermore, a software package called **QSA** (Queueing Systems Assistance) developed in 2021 is integrated into the lecture note of Sztrik [1] with the aim to calculate and visualize the main performance measures. In addition, it helps to minimize a quite general mean total cost per unit time with linear objective function.

The work/publication of J. Sztrik is supported by the EFOP-3.6.1-16-2016-00022 project. The project is co-financed by the European Union and the European Social Fund.

The greatest advantage of this application that these scripts can run in all modern devices including smart phones, too, thus the application is very convenient for students and improve the efficiency of a teacher.

To solve practical problems the first step is to identify the appropriate queueing system and then to calculate the performance measures. Of course the level of modeling heavily depends on the assumptions. It is recommended to start with a simple system and then if the results do not fit to the problem continue with a more complicated one. Various software packages help the interested readers in different level. The following links worth a visit

`http://web2.uwindsor.ca/math/hlynka/qsoft.html`

We have collected some basic books on QT in which software support is mentioned, for example, **Mathemica** in Allen [2], Harchol-Balter [3], **MatLab** in Bhat [4], Kobayashi and Mark [5], Kulkarni [6], Stidham [7].

A reasonable choice for calculations in teaching is the usage of spreadsheets. We highly recommend an Excel-based software package called QTSPlus to determine the main performance measures of basic models. It is associated to the book of Gross, Shortle, Thompson and Harris [8] and can be downloaded here

`ftp://ftp.wiley.com/public/sci_tech_med/queueing_theory/`

For application and problem solving oriented teaching courses we have also developed a software package called QSA (**Queueing Systems Assistance**) see, Szilágyi *et. al.* [9] to calculate and visualize the performance measures together with optimal decisions not only for elementary but more advanced queueing systems as well. It is available at

`https://qsa.inf.unideb.hu`

The **main advantages** of QSA over QTSPlus are the following

- It runs on desktops, laptops, mobile devices
- It calculates not only the mean but the variance of the corresponding random variables
- It gives the distribution function of the waiting/response times (if possible)
- It visualizes all the main performance measures
- It graphically supports the decision making

2. QSA in action, problem solving

QSA is a user interface, a web-based application written in TypeScript. Any browser (Firefox, Chrome, Edge, etc.) on every platform (Windows, Linux, Android, iOS) is supported, which means one can use mobile and desktop devices for performing any calculations which are executed on the server. There are no hardware limitations, the source code is available on GitHub, under the MIT license, so anyone interested can check out the code or help to develop the application. QSA is integrated into the lecture note of Sztrik [1].

In this section we show some examples how to use the application. After the opening one can select between the following modules

- **Table** - to calculate selected performance measures based on input values. The result is exportable into different file formats so that one can use it for further work
- **Chart** - to generate figures and compare the performance measures with each other. Also, it is useful for demonstration or learning purposes.
- **Compare Tables** - to compare two systems' performance measures with each other

One of the special features of the software is that the performance measures of $M/G/1/K/K$ systems with deterministic, Erlang, Hypo-exponential, Hyper-exponential, and gamma distributed service times are calculated. Distribution function of the waiting/response times of the $M/M/c/K$, $M/M/c/m/K$ systems and the performance measures of $M/M/c/K$, $M/M/c/m/K$ with balking and reneging are determined as well. It was our aim to determine, where it is possible, the distribution function of the waiting/response time to solve decision problems. In addition, not only the mean but the variances of the measures are derived. What is also unique is the calculation of the mean total cost per unit time in steady-state. For illustration let us see the following example.

Example Customers arrive to a 3 server system according to a Poisson process with rate 5. The service times are exponentially distributed with parameter 2. Find the minimum capacity of the system for which the probability of blocking is less than 0.01 and the probability that the waiting time exceeds 1.5 minutes is less than 0.05.
Solution: It is an $M/M/3/K$ system and the problem is that by increasing the capacity the blocking probability is decreasing but the waiting time is increasing thus the probability that it exceeds a certain level is increasing. First of all we have to switch to the distribution function of the waiting time and that is why its value should be at least 0.95 at 1.5.

It should be mentioned that for this system there is no closed-form analytical expression for the distribution function of the waiting time as in $M/M/c$ systems. However, it can be computed by the following formula, see Sztrik [1]

$$F_W(t) = 1 - \sum_{n=c}^{K-1} \Pi_n \sum_{i=0}^{n-c} \frac{(c\mu t)^i e^{-c\mu t}}{i!}, \quad \Pi_n = \frac{P_n}{1 - P_K}, \qquad (n \leq K-1).$$

Clearly we have to use the **Chart** module and to visualize the curves as the function of the capacity K. Of course the step is 1, after giving the required parameters λ, μ, c and time slot $t = 1.5$ we generate the chart showing only the measures in question. We can switch on/off the grid, too. Then we get the following Figure 1 showing that there is no solution under these conditions. However, if we change the blocking probability to 0.03 the solution is $K = 12$. Similar questions could be put for the service intensity, and the number of servers, too.

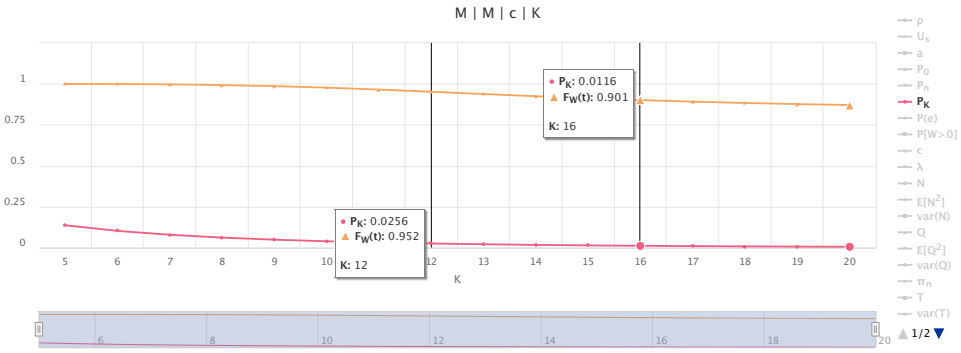


Fig. 1. Solution to the $M/M/5/K$ system

3. Conclusion

In this paper we introduced a new application to help teaching Queueing Theory. One of the main advantages of the software is that it runs on most platforms including smart phones and became very popular among the students. It is easy-to-use and in addition to the calculation of the main steady-state performance measures it visualizes the results and thus supports decision making and optimization of cost functions. The software is integrated into a lecture note where the theoretical part, formulas, and proofs can be found.

REFERENCES

1. J. Sztrik, Basic Queuing Theory, https://irh.inf.unideb.hu/~jsztrik/education/16/SOR_Main_Angol.pdf (2011).
2. A. O. Allen, Probability, statistics, and queueing theory with computer science applications, 2nd ed., Academic Press, Inc., Boston, MA, 1990.
3. M. Harchol-Balter, Performance modeling and design of computer systems: queueing theory in action, Cambridge University Press, 2013.
4. U. N. Bhat, An introduction to queueing theory: modeling and analysis in applications, Birkhäuser, 2015.
5. H. Kobayashi, B. Mark, System modeling and analysis: Foundations of system performance evaluation, Pearson Education Inc., Upper Sadle River, 2008.
6. V. Kulkarni, Modeling, analysis, design, and control of stochastic systems, Springer, New York, 1999.
7. S. Stidham, Optimal design of queueing systems, CRC Press/Taylor & Francis, 2009.
8. D. Gross, J. Shortle, J. Thompson, C. Harris, Fundamentals of queueing theory, 4th edition, John Wiley & Sons, New York, 2008, ftp://ftp.wiley.com/public/sci_tech_med/queueing_theory/.
9. Z. Szilagyi, S. Szaszi, C. Kolcsei, J. Sztrik, Queueing Systems Assistance (QSA), https://qsa.inf.unideb.hu (2021).

UDC: 004.85

On the applicability and limitations of formal verification of machine learning systems

Dmitry Namiot, Evgeniy Ilyushin, Ivan Chizov, Dennis Gamayunov[1]

[1]Faculty of Computational Mathematics and Cybernetics, Lomonosov Moscow State University, GSP-1, 1-52, Leninskiye Gory, Moscow, 119991, Russia

dnamiot@gmail.com

Abstract

The paper deals with the issues of formal verification of machine learning systems. With the growth of the introduction of systems based on machine learning in the so-called critical systems (systems with a very high cost of erroneous decisions and actions), the demand for confirmation of the stability of such systems is growing. How will the built machine learning system perform on data that is different from the set on which it was trained? Is it possible to somehow verify or even prove that the behavior of the system, which was demonstrated on the initial dataset, will always remain so? There are different ways to try to do this. This article deals with the formal verification of machine learning systems.

Keywords: machine learning; formal verification; robust models

1. Introduction

Neural networks and machine learning are among the most successful technologies today, which are usually referred to as the direction of artificial intelligence. At the same time, from the very beginning of the use of these technologies, the issue of justifying the solutions obtained has never been at the forefront. On the contrary, machine learning gained its popularity precisely for solving problems where it was impossible (or it was too difficult) to offer an analytical solution or a deterministic algorithm. The results were initially seen as some kind of magical black box. And only then, there were questions of explaining the solutions obtained [1]. In fact, these explanations are needed just in order to be able to evaluate the data obtained, to evaluate the proposed solutions, etc. Obviously, there are areas of application where the black box is not applicable. Irreversible decisions, for example, in medicine, require explanation.

Due to the fact that these technologies began to be used for critical applications, the question arose of proving the validity of the solutions generated (obtained) with their help. Explanations are also part of the rationale. The explanations themselves are closely related to the applied models. For example, decision trees are, to some extent, explanations [2], the same can be said about regression. Everyone messes with deep learning models, where hidden layers, as their name suggests, hide data processing and make explanations as difficult as possible. DNNs can contain millions of parameters, resulting in overly large search spaces for automated reasoning algorithms [3]. The literature notes that the task in automated verification of neural networks is the coordination of machine learning and automated reasoning [4].

There is another problem with the assessment (justification) of machine learning results, which is of a fundamental nature. Regardless of the models used, the methods of obtaining independent parameters (features), the choice of the analyzed variables, etc., any machine learning models always try to extend the results obtained to the entire general population based on the results of analyzing a certain subset of data. In the general case, generally speaking, there are no (or may not be) grounds for this. This is the main problem. Even explaining how the system works on a training dataset will not help if it turns out that the model does not work on real data. Accordingly, the problem of reliability consists in checking (verifying) that the constructed system operates on data that differ from those on which it was trained.

In this regard, they talk about the stability of the machine learning system. Stable (reliable) and safe machine learning systems are systems whose behavior during operation does not differ from that declared at the testing and training stage. In computer science (informatics), robustness is the ability of a computer system to cope with errors during execution [5, 6], as well as the ability to cope with erroneous input [6]. In the latter work, resilience is defined as: "The degree to which a system or component can function correctly in the presence of invalid inputs or stressful environmental conditions." Resilience can encompass many areas of computer science such as robust (robust) programming, robust machine learning, and robust safety net, etc. Formal methods such as fuzzing testing [7] are necessary to demonstrate robustness because this type of testing includes incorrect or unexpected inputs. Alternatively, artificial injection of faults (in the English-language literature -fault injection) can be used to test stability.

Robust (reliable) machine learning is usually understood as the robustness (reliability) of machine learning algorithms. For a machine learning algorithm to be considered reliable, either the testing error must be consistent with the training error, or the performance must be stable after adding some noise to the dataset.

Formally, for example, for a classification system, this can be defined as follows:

Some classifier C is σ-robust, at the point $\overrightarrow{x}$ only and if $||\overrightarrow{x} - \overrightarrow{x_0}||_\infty \leq \sigma \Rightarrow C(\overrightarrow{x}) = C(\overrightarrow{x_0})$ An intuitive definition that says that if the difference between the original data for all dimensions in the feature space does not exceed, then such objects should be classified in the same way.

It is important that we note exactly the problems (changes) in the data. Nothing is said about the nature of these changes. It may also be a training error - the selected dataset is very different from the known general population, it may be wrong conclusions (assumptions) in algorithms, wrong choice and work with properties (features), as well as deliberately introduced measurements into the initial data sets, which put purpose, for example, a required change in system operation.

Studying the stability of machine learning systems, as well as explaining the operation of such systems (explaining the results obtained), has many aspects. When data changes, they talk about adversarial systems [8] and attacks [9]. It should be noted that the term "attack" here should be understood in a broader sense - it is not necessarily some kind of special malicious data corruption. This should be interpreted, rather, as the presentation (search) of a refuting example. Such a dataset can exist without artificial modifications. A typical example from [11]: the paper describes a scenario in which an autonomous car seeks to change lanes. There is a human-driven car in the other lane, and as we know, people can be unpredictable. An autonomous vehicle has been trained to believe that a person will act in a way that makes overtaking safe. In fact, a person acts a little differently - and the result is an accident. This article is devoted to one of the possible aspects of confirming the results of ML systems - formal verification.

2. About formal verification

The general idea of a formal verification is that we are trying to determine the properties (characteristics) that the neural network should satisfy and use one way or another to verify these properties. There are some parallels with assertions in programming. A statement in programming is a statement in which a predicate (logical expression) must always have a true value in a given part of the code. Programs check assertions by actually evaluating the predicate at runtime, and if, in fact, the predicate is false, the program deliberately stops or throws an exception. Assertions can make the code easier to read, help the compiler compile the code, or help detect defects in the program [10].

Figure 1 [12] shows one of the possible classifications of formal verification systems.

- Theorem proving ML verification
- Linear programming based verification

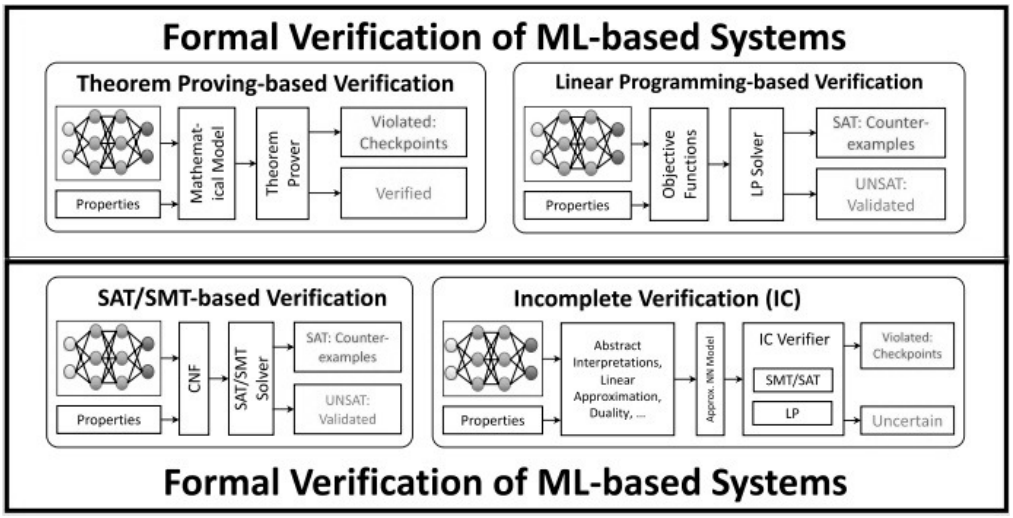


Fig. 1. Formal verification [12].

- SAT / SMT verification
- Incomplete verification

An example of the need for verification: One area where formal verification can be of great importance is in autonomous vehicles such as cars and airplanes. ACAS Xu (Fig. 2) is an unmanned aerial vehicle collision avoidance system. Until recently, the system used a large look-up table mapping sensor measurements to actions to be taken. Later, a neural network approach was used instead of a table as a possible replacement. Memory consumption has been reduced from 2 GB to 3 MB. The problem, however, was that it was difficult to prove that erroneous behavior did not exist in a neural network. Thus, the networks and the security of their use could not be certified [13].

While the above approaches differ in several aspects, they all solve the problem of validation by coding networks within the chosen system of restrictions [3].

SAT solution SAT is aimed at checking the satisfiability of the formula of propositional logic (logic of statements) ϕ, represented as Boolean combinations of atomic (Boolean) sentences. Accordingly, the condition of applicability is the ability to represent (compose) such logical expressions for a real network.

The satisfiability modulo theories (SMT) problem is a solvability problem for logical formulas, taking into account the underlying theories. Examples of such theories for SMT formulas are: theories of integers and real numbers, the theory of

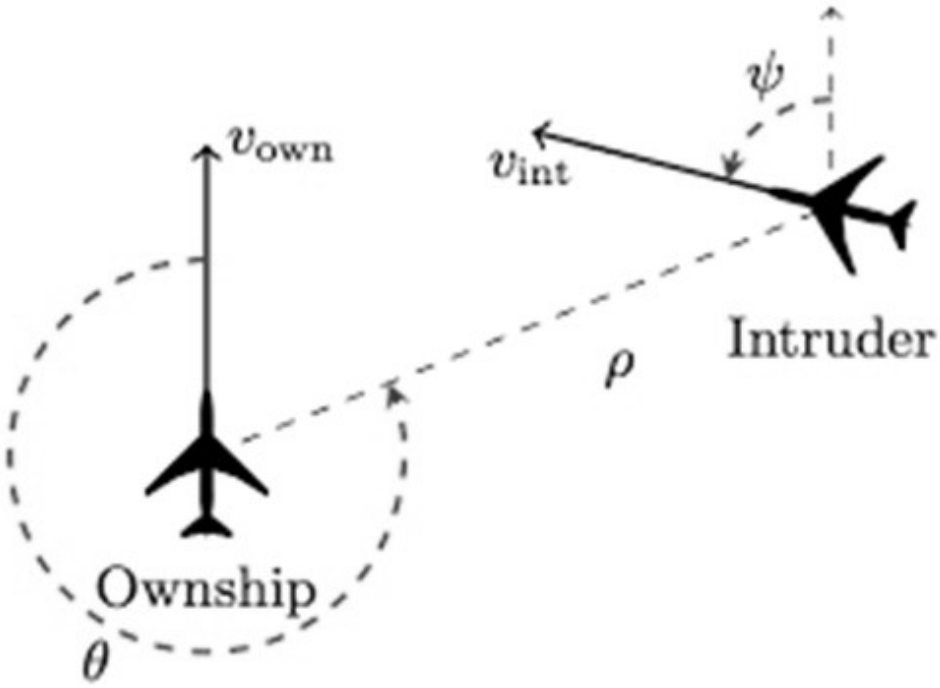


Fig. 2. ACAS Xu [14].

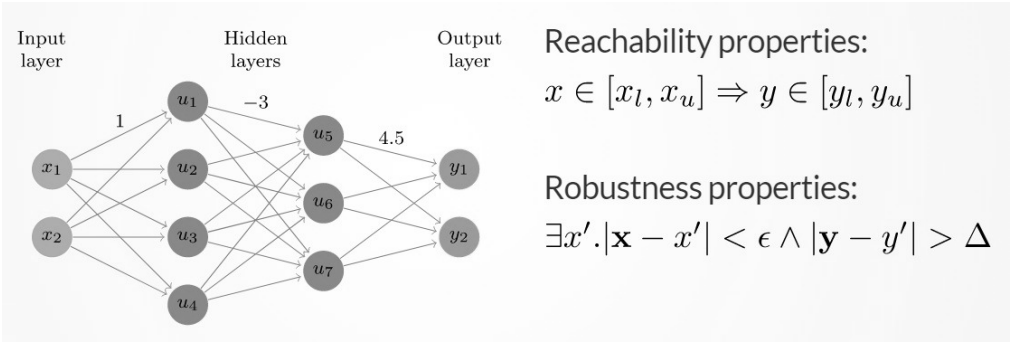


Fig. 3. Marabou [15].

lists, arrays, bit vectors, etc. Formally, an SMT formula is a formula in first order logic in which some functions and predicate symbols have additional interpretations. The challenge is to determine if a given formula is feasible. Unlike the problem of satisfiability of Boolean formulas, an SMT formula contains arbitrary variables instead of Boolean variables, and predicates are Boolean functions of these variables. Accordingly, the condition of applicability is the ability to represent (compose) such logical expressions for a real network.

The Marabou framework [15] can be cited as an example of a system for software verification of neural networks. As described, Marabou is an SMT-based tool that can respond to requests for network properties by converting those requests into constraint checks. It can handle networks with different activation functions and topologies. However, a little further down the text, it turns out that Marabou supports feed-forward networks and convolutional networks with piecewise linear activation functions in TensorFlow. Marabou can respond to several types of test requests (Figure 3):

Safety: if the input is in a given range, will it be guaranteed to be in a certain range? Stability: check if there are points (measurements) around a given entry point (input measurement) that change the network output.

3. Discussion

It turns out that in order to check (confirm) stability, the initial data must contain possible modifications. How else can the constraints be verified?

The changed data is described in terms of a certain boundary (threshold) σ. Naturally, the big question is, how big can this value σ be? For example, to modify

images in adversarial models, modifications are often spoken (considered) that are invisible (almost invisible) to the human eye. This implicitly assumes that there is a human judge who will immediately sweep away fake (modified) images if the modifications are visible to the "naked" eye. And if there is no such intermediary, and we are talking only about machine-to-machine interaction? A small amount of changes here does not seem to be so necessary.

Another issue may be that the feature space does not necessarily include only directly measurable (observable) characteristics, where this difference in measurements can have a physical interpretation. What if such features are artificially constructed based on real measurements and other constructed features? What is the physical interpretation (which is, by the way, part of explaining how the system works) for σ in this case? In this case, what does it mean to "check in the neighborhood" of some input point, if such a parameter is derived from the initial ones? A typical example is the features formed in speech recognition problems (wavelet transformation etc.) [16].

The classification introduced in [3] by constraint checks seems to be more realistic:

At the top level, a neural network can be represented as a function $\upsilon : I^n \rightarrow O^m$, which maps the input domain I of dimension n $(n > 0)$ to the output domain O^m of dimension m.

pre (x) and *post (y)* are first-order logical formulas. x and y are free variables of type S_I and S_O, respectively. S_I is the type of input and S_O is the output. The formulas *pre* define the preconditions at the input of the network, and *post* define the post-conditions at its output. The interpretation maps the variables x and y to values in the I^n and O^m domains, respectively. The expression $L(x \rightarrow e)$ denotes that the variable x is mapped to a value $e \in I^n$ in the interpretation of L, and ϕ^L denotes the value of the expression ϕ in the interpretation of L. The predicates "=", "$\neq$", and "$<$" with ordinary semantics are also considered.

All published studies on automatic verification of neural networks can be reduced to three types of checks [3]:

Invariance. For certain conditions before and after the assertion of the invariance of the property for the network υ is defined as $\forall x.\forall y.(pre(x) \bigwedge y = \upsilon(x)) \Rightarrow post(y)$. The purpose of an automated test is to prove this claim or find a counterexample, i.e. some value $e \in I^n$ such that $(pre(x) \bigwedge \neg post(y))^{L(x \rightarrow e, y \rightarrow \upsilon(e))}$ is true.

Reversibility. For certain conditions before and after the approval of the reversibility property for the network υ is defined as $\forall y.\exists x.(post(y) \bigwedge y = \upsilon(x)) \Rightarrow pre(x)$. Such a condition must either be proved or a specific implementation must be found,

i.e. for a pattern $p \in O^m$ find an input pattern $e \in I^n$ such that $(post(y) \bigwedge y = \upsilon(x) \bigwedge pre(x))^{L(x \to e, y \to p)}$ is true.

Equivalence. While invariance and reversibility refer to the same network, equivalence is a property involving the two networks ϕ and ϕ'. For example, this is an incomplete verification in Figure 1, where there is an approximating network. For certain conditions before and after the equivalence is defined as $\forall x. \forall y. \forall w(pre(x) \bigwedge y = \phi(x) \bigwedge post(y) \bigwedge w = \phi'(x) \bigwedge post(w)) \Rightarrow y = w$

Such a condition can either be proved as such, or a counterexample can be given, that is, some $e \in I^n$ such that $(pre(x) \bigwedge post(y) \bigwedge post(w) \bigwedge y \neq w)^{L(x \to e, y \to \phi(e), w \to \phi'(e))}$ is true. In contexts where strict equality may not be appropriate, we can replace the expression $y = w$ in the definition with $||y - w|| < \sigma$, assuming that $|| \cdot ||^L$ is the norm in O^m and $L \in O^m$ is the tolerance, that is, the threshold at which the considered response of the networks will be indistinguishable.

The possibility of putting forward (formulating) such a condition should be conditioned by the physical meaning of the problem being solved.

4. Conclusion

As with other approaches to testing the robustness of machine learning systems, we cannot note universal methods for formal verification. The possibility of using certain approaches depends on the formulations of the problems (problems to be solved), since it is the formulations of the problem that determine the possibilities of setting conditions. Potential data variances for adversarial examples should also be task-specific. Based on the classification of formal verification methods, for clustering (recognition) problems, the most suitable models are those oriented to checking equivalence.

REFERENCES

1. Roscher, Ribana, et al. "Explainable machine learning for scientific insights and discoveries." IEEE Access 8 (2020): 42200-42216.
2. Došilović, Filip Karlo, Mario Brčić, and Nikica Hlupić. "Explainable artificial intelligence: A survey." 2018 41st International convention on information and communication technology, electronics and microelectronics (MIPRO). IEEE, 2018.
3. Leofante, Francesco, et al. "Automated verification of neural networks: Advances, challenges and perspectives." arXiv preprint arXiv:1805.09938 (2018).

4. Bride, Hadrien, et al. "Towards dependable and explainable machine learning using automated reasoning." International Conference on Formal Engineering Methods. Springer, Cham, 2018.
5. "A Model-Based Approach for Robustness Testing" (PDF). Dl.ifip.org. Retrieved 2016-11-13.
6. 1990. IEEE Standard Glossary of Software Engineering Terminology, IEEE Std 610.12-1990
7. Chen, Chen, et al. "A systematic review of fuzzing techniques." Computers & Security 75 (2018): 118-137.
8. Huang, Ling, et al. "Adversarial machine learning." Proceedings of the 4th ACM workshop on Security and artificial intelligence. 2011.
9. Rouani, Bita Darvish, et al. "Safe machine learning and defeating adversarial attacks." IEEE Security & Privacy 17.2 (2019): 31-38.
10. Plösch, Reinhold. "Evaluation of assertion support for the java programming language." Journal of Object Technology 1.3 (2002): 5-17.
11. Dorsa Sadigh, S. Shankar Sastry, Sanjit A. Seshia, and U.C. Berkeley. "Verifying robustness of human-aware autonomous cars". In: IFAC-PapersOnLine 51.34 (2019), pp. 131–138.
12. Shafique, Muhammad, et al. "Robust machine learning systems: Challenges, current trends, perspectives, and the road ahead." IEEE Design & Test 37.2 (2020): 30-57.
13. Guy Katz, Clark Barrett, David Dill, Kyle Julian, and Mykel Kochenderfer. "Reluplex: An efficient SMT solver for verifying deep neural networks". In: International Conference on Computer Aided Verification . Springer. 2017, pp. 97–117.
14. Lin, Xuankang, et al. "ART: abstraction refinement-guided training for provably correct neural networks." 2020 Formal Methods in Computer Aided Design (FMCAD). IEEE, 2020.
15. Marabou https://github.com/NeuralNetworkVerification/Marabou
16. Wang, Kunxia, et al. "Wavelet packet analysis for speaker-independent emotion recognition." Neurocomputing 398 (2020): 257-264.

UDC: 519.872

Algorithmic Approach to Study the Model of Perishable Inventory System with Repeated Customers

Agassi Melikov[1], Mammad Shahmaliyev[2], János Sztrik[3]

[1]Institute of Control System of National Academy of Science, AZ1141, st. B. Vahabzadeh 9, Baku, Azerbaijan
[2]National Aviation Academy, Mardakan ave, 30, Baku, Azerbaijan
[3]Department of Informatics and Networks, Faculty of Informatics, University of Debrecen, 4032 Debrecen, Hungary

Abstract

The model of perishable inventory system with repeated customers is examined under (s, S) policy. The stability condition of the system is derived and the joint distribution of the number of customers in orbit and the inventory level is obtained by using matrix-geometric method. Formulas for calculation of the performance measures are developed.

Keywords: perishable inventory system, repeated customers, (s, S) policy, matrix-geometric method, calculation methods, performance measures

1. Introduction

One of the important class of operations management systems is a perishable inventory systems (PIS) in which an inventory life time is a finite random quantity, for example, blood banks, systems of processing an outdated information, food provision systems, etc. In PIS the inventory level decreases not only after its release to a customer but also due to the end of inventory lifetime. Note that works [1] - [4] contain references of numerous literature sources in this direction.

Here we consider PIS models without service facility [5] - [8]. This paper is similar to [8]. The main contributions of this paper are as follows: (i) We extend the model investigated in [8] by considering perishable inventory items. (ii) We assume that arrived primary customers (p-customers) in accordance to Bernoulli scheme either join the orbit or leave the system when the inventory level is zero. (iii) We assume that retrial customers (r-customers) might be impatient, i.e. if the inventory level is zero upon arrival, then the r-customers in accordance to Bernoulli scheme either leave the system or re-join the orbit. (iv) We consider (s, S) replenishment policy, i.e.

when the inventory level hits s or below, an order is placed to make the inventory full.

2. Description of the model

The system has a store house of limited volume S and p-customers forms Poisson input with rate λ. If at the moment of p-customer arrival the inventory level is positive, then it is instantly serviced and leaves the system; otherwise (i.e. when inventory level is zero) it with probability H_p either leaves for infinity orbit to repeat its inquiry, or with complementary probability $1 - H_p$ eventually leaves the system. Only r-customer on the head of the orbit repeat its request at random time which has exponential d.f. with parameter η, i.e. retrial rate is independent on the number of r-customers. If at the moment of a r-customer arrival inventory level is positive, then such customer is instantly serviced and leaves an orbit; otherwise the r-customer either leaves an orbit with probability H_r or with complementary probability $1 - H_r$ stays there to repeat its request. It is assumed that only one item can perish in a very little interval and random life time of each item has an exponential d.f. with mean $\gamma^{-1}, \gamma < \infty$. Here we consider (s, S) policy and assume that lead time is positive random variables that has exponential d.f. with the mean ν^{-1}.

3. Computation of the steady-state probabilities

Mathematical model of the system is 2D MC with states that defined by 2D vectors (n, m), where n is total number of customers in orbit, $n = 0, 1, ...,$ and n indicates the inventory level, $m = 0, 1, ..., S$. State space of the indicated 2D MC is given by

$$E = \bigcup_{n=0}^{\infty} L(n) \tag{1}$$

where $L(n) = \{(n, 0), (n, 1), ..., (n, S)\}$ called the n^{th} level, $n = 0, 1, 2, \ldots$.

The transition rate from the state $(n_1, m_1) \in E$ to the state $(n_2, m_2) \in E$ is denoted by $q((n_1, m_1), (n_2, m_2))$. According to the accepted service scheme and replenishment policy, we obtain the following relations for the determining of the

indicated transitions:

$$q((n_1,m_1),(n_2,m_2)) = \begin{cases} \lambda + m_1\gamma, & \text{if } m_1 > 0, (n_2,m_2) = (n_1, m_1 - 1) \\ \eta, & \text{if } n_1 m_1 > 0, (n_2,m_2) = (n_1 - 1, m_1 - 1) \\ \eta H_r, & \text{if } n_1 > 0, m_1 = 0, (n_2,m_2) = (n_2 - 1, m_1) \\ \lambda H_p, & \text{if } m_1 = 0, (n_2,m_2) = (n_1 + 1, m_1) \\ \nu, & \text{if } m_1 \le s, (n_2,m_2) = (n_1, S) \\ 0, & \text{in other cases} \end{cases} \tag{2}$$

Hereinafter, the equality of vectors means that their corresponding components are equal to each other. States from the space E is renumbered in lexicographical order as follows $(0,0),(0,1),...,(0,S),(1,0),(1,1),...,(1,S),....$ Then indicated 2D MC has the following generator:

$$\begin{pmatrix} B & A_0 & . & . & . \\ A_2 & A_1 & A_0 & . & . \\ . & A_2 & A_1 & A_0 & . \\ . & . & . & . & . \end{pmatrix} \tag{3}$$

All block matrices in (3) are square matrices of dimension $S+1$. From relations (2) we conclude that entities of the block matrices $B = ||b_{ij}||$ and $A_k = ||a_{ij}^{(k)}||, i,j = 0,1,...,S$ are determined as follows:

$$b_{ij} = \begin{cases} \nu, & \text{if } i \le s, j = S \\ \lambda + i\gamma, & \text{if } i > s, j = i - 1 \\ -(\nu + \lambda H_p), & \text{if } i = j = 0 \\ -(\nu + i\gamma + \lambda), & \text{if } 0 < i \le s, j = i \\ -(i\gamma + \lambda), & \text{if } s < i \le S, j = i \\ 0, & \text{in other cases} \end{cases} \tag{4}$$

$$a_{ij}^{(0)} = \begin{cases} \lambda H_p, & \text{if } i = j = 0 \\ 0, & \text{in other cases} \end{cases} \tag{5}$$

$$a_{ij}^{(1)} = \begin{cases} \nu, & \text{if } i \le s, j = S \\ \lambda + i\gamma, & \text{if } i > s, j = i - 1 \\ -(\nu + \lambda H_p + \eta H_r), & \text{if } i = j = 0 \\ -(\nu + i\gamma + \lambda + \eta), & \text{if } 0 < i \le s, j = i \\ -(i\gamma + \lambda + \eta), & \text{if } s < i, j = i \\ 0, & \text{in other cases} \end{cases} \tag{6}$$

$$a_{ij}^{(2)} = \begin{cases} \eta H_r, & \text{if } i = j = 0 \\ \eta, & \text{if } 1 < i, j = i - 1 \\ 0, & \text{in other cases} \end{cases} \tag{7}$$

Let $A = A_0 + A_1 + A_2$. Stationary distribution that correspond to the generator A is denoted by $\pi = (\pi(0), \pi(1), ..., \pi(S))$, i.e. we have

$$\pi A = 0, \pi e = 1 \tag{8}$$

where 0 is null row vector of dimension $S + 1$ and e is column vector of dimension $S + 1$ that contains only 1's.

From relations (5)-(7) we obtain that entities of generator $A = ||a_{ij}||, i, j = 0, 1, ..., S$, are determined as

$$a_{ij}^{(1)} = \begin{cases} -\nu, & \text{if } i = j = 0 \\ \nu, & \text{if } 0 \le i \le s, j = S \\ \lambda + i\gamma, & \text{if } i > s, j = i - 1 \\ -(\nu + \lambda H_p + \eta H_r), & \text{if } i = j = 0 \\ -(\nu + i\gamma + \lambda + \eta), & \text{if } 0 < i \le s, j = i \\ -(i\gamma + \lambda + \eta), & \text{if } s < i, j = i \\ 0, & \text{in other cases} \end{cases} \tag{9}$$

Proposition. The system is ergodic if and only if the following relation is fulfilled:

$$\lambda H_p \pi(0) < \eta(1 - (1 - H_r)\pi(0)) \tag{10}$$

Proof: From relations (9) we obtain that system of equations (8) has following form:

$$(\nu + (m\gamma + \lambda + \eta)(1 - \delta_{m,0}))\pi(m) = ((m + 1)\gamma + \lambda + \eta)(\pi(m + 1), 0 \le m \le s \tag{11}$$

$$\begin{aligned}(m\gamma + \lambda + \eta)\pi(m) = ((m + 1)\gamma + \lambda + \eta)(\pi(m + 1)\chi(s + 1 \le m \le S - 1) + \\ + \nu \sum_{m=0}^{s} \pi(m)\delta_{m,S}, s + 1 \le m \le S\end{aligned} \tag{12}$$

Here $\delta_{x,y}$ denotes Kronecker delta and $\chi(A)$ is indicator function of event A.

From (11) and (12) all values $\pi(m), m = 1, .., S$ are expressed by as follows:

$$\pi(m) = \begin{cases} a_m \pi(0), & \text{if } 1 \le m \le s + 1 \\ b_m \pi(0), & \text{if } s + 1 < m \le S \end{cases} \tag{13}$$

where $a_m = \prod_{i=1}^{m} \frac{\Lambda_{i-1}+\nu}{\Lambda_i}; b_m = \frac{\Lambda_{s+1}}{\Lambda_m} \prod_{i=1}^{s+1} \frac{\Lambda_{i-1}+\nu}{\Lambda_i}; \Lambda_i = \lambda + \eta + i\gamma, i = 1, 2, ..., S.$

The probability $\pi(0)$ is determined from normalizing condition, i.e.

$$\pi(0) = \left(1 + \sum_{m=1}^{s+1} a_m + \sum_{m=s+2}^{S} b_m\right)^{-1}$$

In accordance to [9] (chapter 3, pages 81-83) investigated 2D MC is ergodic if and only if the following condition is fulfilled:

$$\pi A_0 e < \pi A_2 e \tag{14}$$

By taking into account (5), (7) and (13) after some algebras from (14) we obtain that relation (10) is true.

Steady-state probabilities corresponding to the generator matrix G we denote by $p = (p_0, p_1, ...)$, where $p_n = (p(n, 0), p(n, 1), ..., p(n, S)), n = 0, 1, ...$ Under the ergodicity condition (10) steady-state probabilities are determined from the following equations:

$$p_n = p_0 R^n, n \geq 1 \tag{15}$$

where R is nonnegative minimal solution of the following quadratic matrix equation:

$$R^2 A_2 + R A_1 + A_0 = 0$$

Bound probabilities p_0 are determined from the normalizing condition:

$$\begin{aligned} p_0(B + RA_2) &= 0 \\ p_0(I - R)^{-1} e &= 1 \end{aligned} \tag{16}$$

where I is indicated identity matrix of dimension $S + 1$.

4. Performance measures

Performance measures are calculated via steady-state probabilities as follows.

Average inventory level: $S_{av} = \sum_{m=1}^{S} m \sum_{n=0}^{\infty} p(n, m)$

Average order size under (s, S) policy: $V_{av} = \sum_{m=S-s}^{S} m \sum_{n=0}^{\infty} p(n, S - m)$

Average number of customers in orbit: $L_o = \sum_{n=1}^{\infty} m \sum_{m=S}^{\infty} p(n, m)$

Average reorder rate: $RR = (\lambda + (s + 1)\gamma) \sum_{n=0}^{\infty} p(n, s + 1) + \eta \sum_{n=1}^{\infty} p(n, s + 1)$

Loss probability of p-customers: $P_p = (1 - H_p) \sum_{n=0}^{\infty} p(n, 0)$

Loss probability of r-customers: $P_r = H_r \sum_{n=1}^{\infty} p(n, 0)$

5. Conclusion

In this paper, the Markovian model of PIS without service facility and with repeated customers under (s, S) policy is proposed. It is assumed that arrived primary customer in accordance Bernoulli scheme either go to infinity orbit or leaves the system when inventory level is zero. By similar way, if upon arrival of a r-customer inventory level is zero then customer either leaves an orbit or stays to repeat its request. The stability condition of the system is derived and the joint distribution of the number of customers in orbit and the inventory level is obtained by using matrix-geometric method. Formulas to calculating the performance measures are developed. These formulas allow solve the design problems as well as optimization problems of the investigated PIS. Similar model of PIS under (s, Q) replenishment policy can be investigated by using developed here approach.

REFERENCES

1. Goyal S., Giri B. Recent trends in modeling of deteriorating inventory. European Journal of Operations Research. 2001. Vol. 134, Issue 1. P. 1-16.
2. Karaesmen I., Scheller-Wolf A., Deniz B. Managing perishable and aging inventories: Review and future research directions. Planning production and inventories in the extended enterprise. A state of the art handbook. (Eds. Kempf K., Keskinocak P, Uzsoy P.). Vol. 1. Springer, 2011. P. 393-438.
3. Bakker M., Riezebos J., Teunter R. H. Review of inventory systems with deterioration since 2001. European Journal of Operation Research. 2012. Vol. 221. P. 275-284.
4. Nahmias S. Perishable inventory theory. Heidelberg: Springer. 2011.
5. Liu L. An (s, S) continuous review models for inventory with random lifetimes. Operations Research Letters. 1990. Vol. 9. Issue 3. P. 161-167
6. Liu L., Yang T. An (s, S) random lifetimes inventory model with positive lead time. European Journal of Operations Research. 1999. Vol. 113. Issue 1. P. 52-63
7. Kalpakam S., Sapna K.P. Continuous review (s, S) inventory system with random lifetimes and positive lead times. Operations Research Letters. 1994. Vol. 16. Issue 2. P. 115-119
8. Anbazhagan N., Wang J., Gomathi D. Base stock policy with retrial demands. Applied Mathematical Modelling. 2013. Vol. 37. P. 4464–4473
9. Neuts M.F. Matrix-geometric solutions in stochastic models: An algorithmic approach. Baltimore: John Hopkins University Press. 1981. 332 p.

UDC: 519.7

Detection of cardiac arrhythmia based on the analysis of electrocardiogram using deep learning models

Eugene Yu. Shchetinin[1], Leonid A. Sevastianov[2,3], Anastasia V. Demidova[2], Yury A. Blinkov[2,4]

[1]Financial University, Government of the Russian Federation, Moscow, Russian Federation
[2]Peoples' Friendship University of Russia (RUDN University), Moscow, Russian Federation
[3]Joint Institute for Nuclear Research, Dubna, Russian Federation
[4]Saratov State University, Saratov, Russian Federation

riviera-molto@mail.ru, sevastianov-la@rudn.ru, demidova-av@rudn.ru, blinkov-yua@rudn.ru

Abstract

The use of computer algorithms for detecting cardiac rhythm disturbance in humans based on an electrocardiogram is studied. For this purpose, the MIT-BIH Physionet database was used, which contains five classes of different types of cardiac rhythm. We propose an electrocardiogram classifier model, which is an ensemble of convolutional (CNN) and recurrent deep neural networks with LSTM unit. The results of performed computer experiments show that the proposed model successfully classifies cardiac arrhythmia with an overall accuracy of 99.37%. The computer system developed can be efficient to detect cardiac arrhythmia at an early stage.

Keywords: arrhythmia, electrocardiogram, deep neural networks, CNN, LSTM, MIT-BIH

1. Introduction

Cardiovascular disease (CVD), according to the World Health Organization, is one of the most common causes of death in the world. More than 17 million people die from CVD annually, of which more than 7 million people die from coronary heart disease. Problems of the cardiac conduction system can lead to aberrations in electrical pulses that disrupt the normal heart rate and rhythm. This abnormality is widely known as arrhythmia, a dangerous disease that threatens human life, therefore, timely diagnosis of arrhythmia is of great importance in the prevention

This paper has been supported by the RUDN University Strategic Academic Leadership Program.

of cardiovascular diseases. The most effective clinical method for visualizing the electrical activity of the heart is electrocardiography (ECG). In addition to being non-invasive, it is also fast and easy to use, providing enough information to diagnose and treat heart disease [1, 2].

Manual analysis of the ECG signal is a complex task, which justifies the need to develop methods for the automated detection of cardiac arrhythmias. Automated ECG analysis has been a subject of great interest in the field of biomedical technology for many years, and it is still a challenging theoretical and practical task.

In this paper, a number of deep neural network models, including convolutional and recurrent networks and their ensembles, are investigated and implemented in software for the classification of ECG signals. A hybrid model of deep neural network is proposed based on a one-dimensional convolutional network 1D_CNN and a recurrent network with a long short-term memory (LSTM) unit. A comparative analysis of these deep models with popular machine learning algorithms is carried out and the effectiveness of using deep neural networks for the automated detection of cardiac arrhythmias is shown. The accuracy of the best deep model is achieved 99.37%.

2. Development of deep neural network models for ECG signal classification

2.1. Deep convolutional neural networks. Convolutional networks were originally designed for image recognition, however, due to their ability to extract the features of classes from the objects under study, they are also used for processing time sequences and digital signals [5, 6].

2.2. Recurrent neural networks. Another class of deep neural networks, often used in the analysis of one-dimensional digital signals and time sequences, are recurrent neural networks. They were originally created to solve the problem of text processing and natural language. In this work, for the classification of ECG signals, we used a fairly well-known model with a long short-term memory (LSTM) unit.

3. Description of the MIT-BIH database of ECG signal samples

To validate and test the proposed methods, we used the MIT-BIH database [10], a freely available dataset that is widely used to assess the effectiveness of ECG signal classification algorithms. In accordance with the Association for the Advancement of Medical Instrumentation (AAMI) standard EC57, each ECG signal can be divided into 5 types of heartbeats [11]:

- **N** – normal rhythm;

- **S** – Supraventricular ectopic beats (atrial premature): atrial (supraventricular) extrasystole, a violation of the heart rhythm, characterized by the occurrence of single or paired premature heart contractions (extrasystoles) caused by excitation of the myocardium. Frequent atrial extrasystoles can be harbingers of atrial fibrillation or atrial paroxysmal tachycardia, accompanying overload or changes in the atrial myocardium;
- **V** – Ventricular ectopic beats: ventricular extrasystole, premature ventricular contraction. Ventricular arrhythmia can be a manifestation of coronary heart disease;
- **F** – Fusion beats: fusion of ventricular and normal rhythm;
- **Q** – undefined rhythms.

4. Analysis of the effectiveness of the deep neural network application to the classification of cardiac arrhythmias

Let us enumerate the records by classes as 'N': 0, 'S': 1, 'V': 2, 'F': 3, 'Q': 4. Then the number of ECG records in each class is 0-72471, 1-2223, 2-5788,3-641, 4-6431, so, it could be seen from this, that the classses of ECG signals are extremely misbalanced.

The structure of a deep convolutional network proposed here has the following form: a vector of initial data of dimension (187, 1) is fed to the input of the neural network. The first layer of the network is a 1D convolution layer with 32 filters of size (6×1), a convolution step of 1 and a non-linear activation function of the ReLU layer. This is followed by the max pooling layer with a core size (3×1) and a step of 1. Its application halves the number of parameters, choosing only neurons with the maximum activation value within the region (3×1). This is followed by the second unit, consisting of a 1D convolution layer with 64 filters, a kernel (3×1) , a convolution step of 1 and an activation function of the ReLU layer, as well as a max pooling layer with a kernel size (3×1) and step 1.

The third block also consists of a 1D convolution layer with 128 filters, a kernel (6×1), a convolution step of 1, and a ReLU layer activation function. Next, a dropout layer is added with a coefficient of 0.25. Dropout is one of the most effective and common neural network regularization techniques.

Then comes the flatten layer, which converts the multidimensional feature vector to a 1D vector, preparing the output for a fully connected layer. The output of the flatten layer is then passed to a dense layer of 512 neurons and a ReLU activation function. This is followed by another dense layer with 32 neurons and the ReLU activation function. Finally, another dense layer is included in the network with the Softmax activation function, which is used to predict the class to which the input

belongs. The output size of this layer is 5 because there are 5 classes of ECG signal patterns.

The architecture of the proposed recurrent neural network based on long short-term memory is as follows. The input layer is proposed to convert ECG signals into a structure of dimension (187, 1) corresponding to the length of the ECG signal. Next comes the LSTM_1(none, 187) layer, which contains an LSTM cell with 50 neurons per layer. This is followed by a dense layer (none, 5) containing a Softmax activation function that is used to predict the class to which the input belongs. The neural network model was compiled with a sparse categorical cross-entropy loss function, an Adam optimizer, and an accuracy metric. To prevent overfitting in the neural network architecture, the dropout layer with the parameter equal 0.2 is also used.

To improve the classification accuracy of individual classes of the data under study, a hybrid stacked CNN-LSTM model was proposed, which combines the convolutional and recurrent models described above. In this model, the 1D CNN convolutional block first analyzes the ECG signals, selects the key features of the classes from them and transmits them to the subsequent part of the network. Further, the received features enter the LSTM recurrent network, where the classification is performed. In addition to the deep neural network models described above, the following machine learning algorithms were used: support vector machine (SVM), decision trees (DT), random forest (RF) and extreme gradient boosting classifier (XGB). It should be noted that in the process of training these algorithms, an important stage is the selection of the optimal values of their hyperparameters. For this purpose,we used the various methods such as Grid Search and Stratified Search from scikit-learn library of Python programming language [12].

The main results of the classification of ECG signals using machine learning algorithms and deep neural networks are shown in Table 1. Comparing various algorithms for the quality of classification for individual classes, it can be seen that machine learning algorithms provide a good classification for classes with a large amount of samples. For example, SVM and DT classify samples from class N and Q with an accuracy of 92% and 97%, respectively, and samples from classes S and F are classified significantly worse with an accuracy of only 63%, whereas already RF and XGB classify samples from classes S and F with an accuracy of 82%.

At the same time, analyzing and comparing the performance of various neural network models based on the obtained estimates of the classification accuracy, it can be argued that the hybrid CNN LSTM model allows not only to obtain a high classification accuracy of 99.37%, but also high values of other indicators of classification quality F1-metric, precision, recall. Their values are shown in Table 1.

As follows from the table, almost all of the studied algorithms have demonstrated high accuracy in the classification of ECG signal samples. On the other hand,

Model of classifier	Accuracy, %	Precision, %	Recall, %	F1-metric, %	AUC_macro
SVM	92.6	88.4	87.3	88.2	89.2
Random Forest	97	94.22	93.13	94.4	95.1
Decision Tree	95	95	95.4	95.4	95
XGB	97	97.44	97.46	97.34	97
CNN_1D_3block	98.6	97.66	97.54	97.68	97.4
LSTM	97	98	98	97.35	98
CNN_LSTM	99.37	99.2	99.4	99.1	99.8

Table 1. The results of classification of ECG classes

since that our main task is the detection of cardiac arrhythmia, which is mainly characterized by classes S, V and F, we are primarily interested in the accuracy of classification of these classes. In this situation, the confusion matrix (CM) is an effective characteristic of the performance of the machine classifier. After the performed computer calculations of CM for all classifiers, it can be argued that, despite the overall high classification accuracy, machine learning algorithms do not allow detecting heart rhythm disorders with high quality.

Visual analysis of the confusion matrix for the 1D CNN deep model showed that with a sufficiently high overall classification accuracy for all classes, the individual accuracy estimates for classes S and F turned out to be less accurate than for the rest classes N and Q. Analyzing the confusion matrix of the CNN LSTM model, it can be argued that it performs very well in the classification of ECG classes, and its overall accuracy made up 99.8%.

5. Conclusion

We studued the use of deep learning algorithms for detecting disturbances of the human heart rhythm based on the analysis of an electrocardiogram. As test data, we used the MIT-BIH Physionet database, which consists of 5 classes of various cardiac arrhythmias. The paper proposes a model of an ECG signal classifier, which combines a convolutional neural network (CNN) and a recurrent neural network (LSTM). The effectiveness of the constructed algorithms for the classification of ECG signals was confirmed by the results of their application to the Physionet MIT-BIH database. Computer experiments showed that the proposed model successfully classifies cardiac arrhythmias with an overall accuracy of 99.37%. The proposed classifiers can be applied in biomedical applications such as a medical robot that

processes and analyzes an electrocardiogram and helps doctors more accurately diagnose cardiac arrhythmias.

The most important practical application of the results of the work is the promotion of the developed algorithms for the study of other bases of electrocardiograms in order to create algorithms for the transfer of learning to recognize arrhythmia. The most important line of development of our research is the elaboration of a mobile application that allows solving the problem of remote detection of arrhythmias using an electrocardiogram sample uploaded by a doctor in the form of a photograph. For this, it is proposed to develop and implement a computer algorithm for processing a digital signal of an electrocardiogram into a two-dimensional image and study it using machine classification algorithms.

REFERENCES

1. Rangaraj M. Rangayyan. Biomedical Signal Analysis, 2nd Edition. Wiley–IEEE Press, 2015.
2. Dubrovin V.I., Tverdokhleb Yu.V., Kharchenko V.V. Automated system for analysis and interpretation of ECG // Radioelektronika, informatika, upravleniye. 2014. No. 1. P. 150–157 (in Russian).
3. Heart Disease and Stroke Statistics—2018 Update: A Report from the American Heart Association. E.J. Benjamin, S.S. Virani, C.W. Callaway et al. // Curculation. 2018. V. 137, No. 12. P. 67–492.
4. Coast D. A., Stern R. M., Cano G. G. and Briller S. A. An approach to cardiac arrhythmia analysis using hidden Markov models // IEEE Trans. Biomed. Eng. 1990. V. 37, No. 9, P. 826–836.
5. Isin A. and Ozdalili S, Cardiac arrhythmia detection using deep learning // Procedia Comput. Sci. 2017. V. 120. P. 268–275.
6. Zhai X. and Tin C. Automated ECG classification using dual heartbeat coupling based on convolutional neural network // IEEE Access. 2018. V. 6. P. 27465–27472,
7. Acharya U. R. et al. A deep convolutional neural network model to classify heartbeats // Comput. Biol. Med. 2017. V. 89. P. 389–396.
8. Kiranyaz S., Ince T. and Gabbouj M. Real-time patient-specific ECG classification by 1-D convolutional neural networks // IEEE Trans. Biomed. Eng. 2015. V. 63, No. 3. P. 664–675.
9. Jun J. Nguyen, H. M., Kang D., Kim D. and Kim Y.-H. ECG arrhythmia classification using a 2-D convolutional neural network // arXiv Prepr.: arXiv1804.06812. 2018.

10. Mark R. and Moody G. MIT-BIH arrhythmia database directory. Cambridge Massachusetts Inst. Technol., 1988.
11. Association for the Advancement of Medical Instrumentation, https://www.aami.org/.
12. Chollet F. Deep Learning with Python. Manning Publications, 2017. 384 p.

УДК: 519.21

Асимптотический анализ неоднородной СМО $M|GI|\infty$, функционирующей в марковской случайной среде, в условии эквивалентного роста времени обслуживания на приборах

Е.П. Полин[1], С.П. Моисеева[1], А.Н. Моисеев[1]

[1]НИ ТГУ, пр. Ленина 36, Томск, Россия

polin_evgeny@mail.ru, smoiseeva@mail.ru, moiseev.tsu@gmail.com

Аннотация

Рассматривается неоднородная система массового обслуживания (СМО) с неограниченным числом обслуживающих приборов, функционирующая в условиях изменяющейся внешней среды. На вход СМО поступает пуассоновский поток, время обслуживания заявок на приборах является положительной случайной величиной, имеющей произвольную функцию распределения вероятностей. Интенсивность входящего потока и параметры обслуживания поступающей заявки, не меняющие свои значения до окончания обслуживания, определяются состоянием внешней среды. Решается задача исследования многомерного случайного процесса – числа заявок, обслуживаемых с разной интенсивностью в системе методом асимптотического анализа. Доказано, что распределение вероятностей рассматриваемого процесса при условии эквивалентно растущего времени обслуживания является многомерным гауссовским.

Ключевые слова: бесконечнолинейная система массового обслуживания, случайная среда, метод асимптотического анализа.

1. Введение

В настоящее время существует множество различных технических систем передачи информации, данных, а также телекоммуникационных систем, функционирующих в условиях изменяющейся внешней среды. Эти изменения оказывают влияние на систему, которое может выражаться, например, в изменении параметров функционирования. В связи с этим возникают вопросы устойчивости таких систем к внешним воздействиям. Поэтому исследование систем, работающих в случайной среде [1], является актуальной и интересной задачей как в теоритическом, так и в прикладном плане.

Большинство работ по исследованию классических моделей систем массового обслуживания (СМО) не учитывают возможности изменения параметров системы во времени. С развитием технологий, производства, информационных сетей и сетей связи становится актуальным исследование СМО с изменяемыми или изменяющимися параметрами функционирования [2], так как такие системы более адекватно описывают некоторые случайные процессы, исследование которых требуется в задачах теории массового обслуживания.

Имеется много работ, посвященных исследованию бесконечнолинейных систем как в марковских [3, 4], так и полумарковских [5, 6] случайных средах. В разных работах рассмотрены различные варианты реакции заявок на переход среды в новое состояние. Например, в работе [7] представлен случай, при котором в момент смены состояния среды все заявки немедленно покидают систему. В работе [8] рассмотрен вариант, при котором в момент перехода среды в новое состояние заявки, имеющиеся в системе, переходят на соответствующий новый режим обслуживания. В данной же работе рассматривается случай, предполагающий, что режим обслуживания заявок не меняется до тех пор, пока они не покинут систему.

Исследование систем с непуассоновскими входящими потоками и произвольным временем обслуживания заявок на приборах требует применения специальных методов. Для более детального исследования применяются асимптотические методы [9] при различных условиях.

В настоящей работе рассматривается неоднородная система массового обслуживания $M|GI|\infty$. Интенсивность входящего потока и время обслуживания требований определяются состоянием случайной среды. Отличительной особенностью является зависимость параметров функционирования системы от состояния среды. Рассматривается случай, когда параметры обслуживания заявок не меняют свои значения до окончания обслуживания. Таким образом, приборы в рассматриваемой системе являются неоднородными (гетерогенными) [10], поэтому такую систему будем называть неоднородной СМО, функционирующей в случайной среде.

2. Постановка задачи

Рассматривается неоднородная система массового обслуживания $M|GI|\infty$ с неограниченным числом приборов, функционирующая в случайной среде, имеющей конечное множество состояний $k = 1, ..., K$. Процесс изменения состояний внешней среды является цепью Маркова $k(t)$, которая задается матрицей инфинитезимальных характеристик $\mathbf{Q} = [q_{ij}],\ i, j = 1, ..., K$. Дисциплина обслуживания определяется следующим образом: если вложенная цепь Маркова находится в состоянии $k(t) = n$, то заявка поступает с интенсивностью $\lambda_n,\ n = 1, ..., K$ и

обслуживается на приборе n-го типа в течение случайного времени, имеющего произвольную функцию распределения вероятностей $B_n(x) = Pr\{\tau_n < x\}$ и не меняющегося при изменении состояния среды. Таким образом, в системе одновременно обслуживаются заявки с разными параметрами обслуживания.

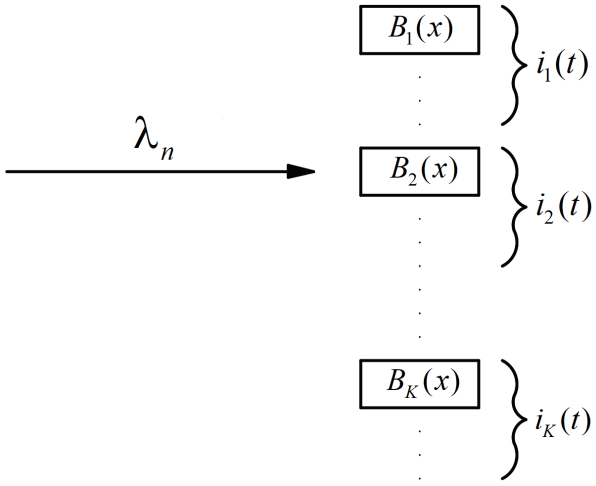


Рис. 1. Неоднородная система массового обслуживания $M|GI|\infty$ в марковской случайной среде

Обозначим $i_n(t)$ – число занятых приборов n-го типа в рассматриваемой СМО. Ставится задача исследования многомерного случайного процесса $\mathbf{i}(t) = [i_1(t), i_2(t), ..., i_K(t)]$ – числа занятых приборов разного типа в системе в момент времени t. Процесс $\mathbf{i}(t)$ не является марковским.

3. Модификация метода динамического просеивания

Для решения поставленной задачи предлагается модифицированный метод динамического просеивания потока. Метод просеянного потока позволяет заменить исследование сложных процессов изменения во времени числа обслуживающих приборов, занятых в системе, исследованием более простых процессов изменения числа событий, наступивших в нестационарных просеянных потоках до некоторого момента времени. Введем следующие обозначения: $n_1(T), n_2(T), ..., n_K(T)$ – число заявок, поступивших в систему в момент времени t и не закончивших обслуживание к некоторому фиксированному моменту времени T, $t < T$; $S_i(t) = Pr\{\tau_i > T - t\} = 1 - B_i(T - t)$ – вероятность того, что заявка i-го типа, пришедшая в момент времени t, к некоторому выделенному моменту времени T еще не закончила своего обслуживания ($i = 1, 2, ..., K$). Пусть в начальный

момент времени $t_0 < T$ система пуста, то есть $n_1(t_0) = n_2(t_0) = ... = n_K(t_0) = 0$. Тогда в момент времени T: $i_1(T) = n_1(T), i_2(T) = n_2(T), ..., i_K(T) = n_K(T)$. Таким образом, задача сводится к исследованию многомерного марковского случайного процесса $\{k(t), n_1(t), n_2(t), ..., n_K(t)\}$.

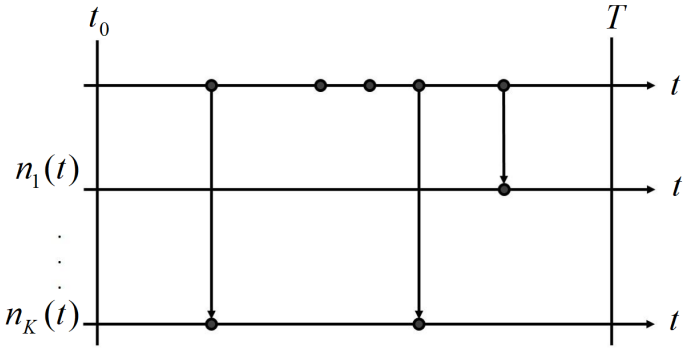


Рис. 2. Просеивание заявок входящего потока

Для распределения вероятностей $P(k, n_1, n_2, ..., n_K, t) = Pr\{k(t) = k, n_1(t) = n_1, n_2(t) = n_2, ..., n_K(t) = n_K\}$ запишем систему дифференциальных уравнений Колмогорова:

$$\frac{\partial P(k, n_1, ..., n_K, t)}{\partial t} = \lambda_k S_k(t) \left(P(k, n_1, ..., n_k - 1, ..., n_K, t) - P(k, n_1, ..., n_K, t)\right) +$$

$$+ \sum_{\nu} q_{\nu k} P(\nu, n_1, ..., n_K, t)$$

с начальными условиями

$$P(k, n_1, ..., n_K, t_0) = \begin{cases} r(k), & \text{если } n_1 = ... = n_K = 0,\ k = 1, ..., K \\ 0, & \text{если } n_1 > 0, ..., n_K > 0, \end{cases}$$

$r(k)$ – стационарные вероятности значений цепи Маркова $k(t)$.

Введем частичные характеристические функции вида

$$H(k, u_1, ...u_K, t) = \sum_{n_1=0}^{\infty} ... \sum_{n_K=0}^{\infty} e^{ju_1 n_1} ... e^{ju_K n_K} P(k, n_1, n_2, ..., n_K, t),$$

$$k = 1, 2, ..., K,\ j = \sqrt{-1}.$$

Запишем систему дифференциальных уравнений для частичных характеристических функций

$$\frac{\partial H(k,u_1,...u_K,t)}{\partial t} = \lambda_k S_k(t) H(k,u_1,...u_K,t)\left(e^{ju_k}-1\right) + \\ + \sum_{\nu} q_{\nu k} H(\nu,u_1,...,u_K,t),$$

$$H(k,u_1,...u_K,t_0) = r(k),\ k=1,2,...,K.$$

В векторно-матричной форме данная система примет вид

$$\frac{\partial \mathbf{H}(u_1,...u_K,t)}{\partial t} = \mathbf{H}(u_1,...u_K,t)\left(\mathbf{A}(u_1,...u_K,t)+\mathbf{Q}\right), \qquad (1)$$

где $\mathbf{A}(\mathbf{u},t) = \begin{bmatrix} \left(e^{ju_1}-1\right)\lambda_1 S_1(t) & 0 & ... & 0 \\ 0 & \left(e^{ju_2}-1\right)\lambda_2 S_2(t) & ... & 0 \\ ... & ... & ... & ... \\ 0 & 0 & ... & \left(e^{ju_K}-1\right)\lambda_K S_K(t) \end{bmatrix}$,

$\mathbf{H}(u_1,...u_K,t) = [H(1,u_1,...u_K,t),...,H(K,u_1,...u_K,t)]$.

Полученная система уравнений (1) является основной для дальнейших исследований. Предлагается провести анализ характеристик рассматриваемой СМО с помощью метода асимптотического анализа [10].

4. Метод асимптотического анализа в условии эквивалентно растущего времени обслуживания

Предлагаемый метод асимптотического анализа реализуется в построении последовательности асимптотик возрастающего порядка, в котором асимптотика первого порядка, аналогично закону больших чисел, определяет асимптотическое среднее значение числа занятых приборов. Асимптотика второго порядка, аналогично центральной предельной теореме, позволяет построить гауссовскую аппроксимацию распределения вероятностей числа занятых приборов в системе.

4.1. Асимптотика первого порядка. Обозначим $\frac{1}{b_i} = q_i\epsilon$, $i=1,...,K$ (ϵ – бесконечно малая величина). Решение системы (1) будем находить в асимптотическом условии эквивалентно растущего времени обслуживания, то есть при $\frac{1}{b_i} \to 0$, где $b_i = \int_0^\infty (1-B_i(x))\,dx$ – среднее время обслуживания на приборе i-го типа, $i=1,...,K$.

В уравнении (1) выполним следующие замены

$$t\epsilon = \tau,\ t_0\epsilon = \tau_0,\ u_i = \epsilon x_i,\ S_i(t) = \widetilde{S}_i(\tau),\ \mathbf{H}(u_1,...u_K,t) = \mathbf{F}(x_1,...x_K,\tau,\epsilon),$$

для $\mathbf{F}(x_1,...x_K,\tau,\epsilon)$ получим матричное уравнение

$$\epsilon\frac{\partial\mathbf{F}(x_1,...x_K,\tau,\epsilon)}{\partial\tau}=\mathbf{F}(x_1,...x_K,\tau,\epsilon)\left(\widetilde{\mathbf{A}}(x_1,...x_K,\tau,\epsilon)+\mathbf{Q}\right), \tag{2}$$

где $\widetilde{\mathbf{A}}=\begin{bmatrix}\left(e^{j\epsilon x_1}-1\right)\lambda_1\widetilde{S}_1(\tau) & 0 & ... & 0\\ 0 & \left(e^{j\epsilon x_2}-1\right)\lambda_2\widetilde{S}_2(\tau) & ... & 0\\ ... & ... & ... & ...\\ 0 & 0 & ... & \left(e^{j\epsilon x_K}-1\right)\lambda_K\widetilde{S}_K(\tau)\end{bmatrix}.$

Теорема 1. *Предельное при $\epsilon\to 0$ решение уравнения* (2) $\mathbf{F}(x_1,...x_K,\tau)$ *имеет вид*

$$\mathbf{F}(x_1,...x_K,\tau)=\mathbf{r}\exp\left\{j\sum_{i=1}^{K}r_ix_i\lambda_i\int_{\tau_0}^{\tau}\widetilde{S}_i(z)dz\right\}, \tag{3}$$

где $\mathbf{r}=[r_1,r_2,...,r_K]$ – вектор стационарного распределения вероятностей значений вложенной цепи Маркова.

Доказательство. В уравнении (2) выполним предельный переход при $\epsilon\to 0$

$$\mathbf{F}(x_1,...x_K,\tau)\mathbf{Q}=0.$$

Функцию $\mathbf{F}(x_1,...x_K,\tau,\epsilon)$ будем искать в виде разложения

$$\mathbf{F}(x_1,...x_K,\tau,\epsilon)=\Phi(x_1,...x_K,\tau)\mathbf{r}+. \tag{4}$$

Уравнение (2) поделим на ϵ, выполним предельный переход при $\epsilon\to 0$ и помножим на единичный вектор $\mathbf{e}$ размерности $K\times 1$, получим

$$\frac{\partial\mathbf{F}(x_1,...x_K,\tau)}{\partial\tau}\mathbf{e}=\mathbf{F}(x_1,...x_K,\tau)\widetilde{\mathbf{A}}_1(x_1,...x_K,\tau)\mathbf{e},$$

где $\widetilde{\mathbf{A}}_1(x_1,...x_K,\tau)=\begin{bmatrix}jx_1\lambda_1\widetilde{S}_1(\tau) & 0 & ... & 0\\ 0 & jx_2\lambda_2\widetilde{S}_2(\tau) & ... & 0\\ ... & ... & ... & ...\\ 0 & 0 & ... & jx_K\lambda_K\widetilde{S}_K(\tau)\end{bmatrix}.$

Подставляя в полученное выражение разложение (4), получаем уравнение для нахождения функции $\Phi(x_1,...x_K,\tau)$

$$\frac{\partial\Phi(x_1,...x_K,\tau)}{\partial\tau}\mathbf{re}=\Phi(x_1,...x_K,\tau)\mathbf{r}\widetilde{\mathbf{A}}_1(x_1,...x_K,\tau)\mathbf{e}.$$

Решение будет иметь вид

$$\Phi(x_1,...x_K,\tau)=\exp\left\{j\sum_{i=1}^{K}r_ix_i\lambda_i\int_{\tau_0}^{\tau}\widetilde{S}_i(z)dz\right\}.$$

Подставляя полученное решение в (4), получим (3). ■

В силу замены, а также равенства (3) можно записать приближённое (асимптотическое) равенство

$$\mathbf{H}(u_1,...u_K,t)=\mathbf{F}(x_1,...x_K,\tau,\epsilon)\approx\mathbf{F}(x_1,...x_K,\tau)=$$

$$=\mathbf{r}\exp\left\{j\sum_{i=1}^{K}r_ix_i\lambda_i\int_{\tau_0}^{\tau}\widetilde{S}_i(z)dz\right\}=\mathbf{r}\exp\left\{j\sum_{i=1}^{K}r_iu_i\lambda_i\int_{t_0}^{t}S_i(z)dz\right\}$$

Следовательно, для характеристической функции процесса $\{n_1(t),n_2(t),...,n_K(t)\}$ можно записать

$$M\exp\left\{j\sum_{i=1}^{K}u_in_i(t)\right\}=\mathbf{H}(u_1,...u_K,t)\mathbf{e}\approx\exp\left\{j\sum_{i=1}^{K}r_iu_i\lambda_i\int_{t_0}^{t}S_i(z)dz\right\}.$$

Тогда при $t=T=0,\ t_0=-\infty$ определим характеристическую функцию процесса $\{i_1(t),i_2(t),...,i_K(t)\}$

$$h_1(u_1,...u_K)=\exp\left\{j\sum_{i=1}^{K}r_iu_i\lambda_i\int_{-\infty}^{0}(1-B_i(-z))\,dz\right\}=$$

$$=\exp\left\{j\sum_{i=1}^{K}r_iu_i\lambda_i\int_{0}^{\infty}(1-B_i(z))\,dz\right\}=\exp\left\{j\sum_{i=1}^{K}r_iu_i\lambda_ib_i\right\},$$

которую будем называть асимптотикой первого порядка характеристических функций числа занятых приборов в системе.

4.2. Асимптотика второго порядка. Для построения многомерной гауссовской аппроскимации перейдем к построению второй асимптотики. Решение $\mathbf{H}(u_1,...u_K,t)$ уравнения (1) запишем в виде произведения

$$\mathbf{H}(u_1,...u_K,t)=\mathbf{H}_2(u_1,...u_K,t)\exp\left\{j\sum_{i=1}^{K}r_iu_i\lambda_i\int_{t_0}^{t}S_i(z)dz\right\},$$

подставляя которое в (1), получим уравнение для $\mathbf{H}_2(u_1,...u_K,t)$

$$\frac{\partial \mathbf{H}_2(u_1,...u_K,t)}{\partial t} = \mathbf{H}_2(u_1,...u_K,t)\left(\mathbf{B}\mathbf{\Lambda}\mathbf{S} + \mathbf{Q} - j\sum_{i=1}^{K} u_i r_i \lambda_i S_i(t)\mathbf{I}\right), \quad (5)$$

здесь $\mathbf{B}(\mathbf{u}) = \begin{bmatrix} e^{ju_1}-1 & 0 & ... & 0 \\ 0 & e^{ju_2}-1 & ... & 0 \\ ... & ... & ... & ... \\ 0 & 0 & ... & e^{ju_K}-1 \end{bmatrix}$, $\mathbf{\Lambda} = \begin{bmatrix} \lambda_1 & 0 & ... & 0 \\ 0 & \lambda_2 & ... & 0 \\ ... & ... & ... & ... \\ 0 & 0 & ... & \lambda_K \end{bmatrix}$,

$\mathbf{S}(t) = \begin{bmatrix} S_1(t) & 0 & ... & 0 \\ 0 & S_2(t) & ... & 0 \\ ... & ... & ... & ... \\ 0 & 0 & ... & S_K(t) \end{bmatrix}$, $\mathbf{I}$ – единичная матрица размера $K \times K$.

Обозначим $\dfrac{1}{b_i} = q_i\epsilon^2$ и в уравнении (5) выполним замены

$$t\epsilon^2 = \tau,\ t_0\epsilon^2 = \tau_0,\ u_i = \epsilon x_i,\ S_i(t) = \widetilde{S}_i(\tau),\ \mathbf{H}_2(u_1,...u_K,t) = \mathbf{F}_2(x_1,...x_K,\tau,\epsilon),$$

получим матричное уравнение для $\mathbf{F}_2(x_1,...x_K,\tau,\epsilon)$:

$$\epsilon^2\frac{\partial \mathbf{F}_2(x_1,...x_K,\tau,\epsilon)}{\partial \tau} = \mathbf{F}_2(x_1,...x_K,\tau,\epsilon)\left(\widetilde{\mathbf{B}}\mathbf{\Lambda}\widetilde{\mathbf{S}} + \mathbf{Q} - j\sum_{i=1}^{K} u_i r_i \lambda_i \widetilde{S}_i(\tau)\mathbf{I}\right). \quad (6)$$

Для функции $\mathbf{F}_2(x_1,...x_K,\tau,\epsilon)$ сформулирована и доказана следующая теорема:

Теорема 2. *Предельное при $\epsilon \to 0$ решение уравнения* (6) $\mathbf{F}_2(x_1,...x_K,\tau)$ *имеет вид*

$$\begin{aligned} &\mathbf{F}_2(x_1,...x_K,\tau) = \\ &= \mathbf{r}\exp\left\{\frac{j^2}{2}\left(\sum_{i=1}^{K} x_i^2\left(r_i\lambda_i\int_{\tau_0}^{\tau}\widetilde{S}_i(z)dz + 2f_{ii}\lambda_i^2 r_i\int_{\tau_0}^{\tau}\widetilde{S}_i^2(z)dz\right) + \right.\right. \\ &\left.\left. +2\sum_{i=1}^{K}\sum_{j\neq i}^{K} x_i x_j f_{ji}\lambda_i\lambda_j r_j\int_{\tau_0}^{\tau}\widetilde{S}_i(z)\widetilde{S}_j(z)dz\right)\right\}, \end{aligned} \quad (7)$$

где $\mathbf{r} = [r_1, r_2, ..., r_K]$ – вектор стационарного распределения вероятностей значений вложенной цепи Маркова, матрица $\mathbf{F} = [f_{ij}]$, $i,j = 1,...,K$ является решением матричного уравнения

$$\mathbf{F}\mathbf{Q} + \mathbf{I} - \mathbf{e}\mathbf{r} = \mathbf{0}$$

и удовлетворяет условию $\mathbf{F}\mathbf{e} = \mathbf{0}$.

В силу замены, а также равенства (7), для функции $\mathbf{H}_2(u_1,...u_K,t)$ можно записать приближённое (асимптотическое) равенство

$$\mathbf{H}_2(u_1,...u_K,t) \approx \mathbf{F}_2(x_1,...x_K,\tau) =$$

$$= \mathbf{r}\exp\left\{\frac{j^2}{2}\left(\sum_{i=1}^{K}u_i^2\left(r_i\lambda_i\int_{t_0}^{t}S_i(z)dz + 2f_{ii}\lambda_i^2 r_i\int_{t_0}^{t}S_i^2(z)dz\right) + \right.\right.$$

$$\left.\left. +2\sum_{i=1}^{K}\sum_{j\neq i}^{K}u_iu_jf_{ji}\lambda_i\lambda_jr_j\int_{t_0}^{t}S_i(z)S_j(z)dz\right)\right\}.$$

Таким образом, характеристическая функция числа занятых приборов в рассматриваемой системе является асимптотически гауссовской характеристической функцией и имеет вид

$$h_2(u_1,...u_K) =$$

$$= \exp\left\{j\sum_{i=1}^{K}r_iu_i\lambda_ib_i + \frac{j^2}{2}\left(\sum_{i=1}^{K}u_i^2\left(r_i\lambda_ib_i + 2f_{ii}\lambda_i^2r_i\beta_i\right) + \right.\right.$$

$$\left.\left. +2\sum_{i=1}^{K}\sum_{j\neq i}^{K}u_iu_jf_{ji}\lambda_i\lambda_jr_j\beta_{ij}\right)\right\},$$

где $b_i = \int_{t_0}^{t}S_i(z)dz = \int_0^{\infty}(1-B_i(z))\,dz$, $\beta_i = \int_{t_0}^{t}S_i^2(z)dz = \int_0^{\infty}(1-B_i(z))^2\,dz$, $\beta_{ij} = \int_{t_0}^{t}S_i(z)S_j(z)dz = \int_0^{\infty}(1-B_i(z))\,(1-B_j(z))\,dz$.

Матрица ковариации имеет вид

$$\mathbf{K} = \begin{bmatrix} r_1\lambda_1b_1 + 2f_{11}\lambda_1^2r_1\beta_1 & ... & 2f_{K1}\lambda_1\lambda_Kr_K\beta_{1K} \\ ... & ... & ... \\ 2f_{1K}\lambda_1\lambda_Kr_1\beta_{K1} & ... & r_1\lambda_Kb_K + 2f_{KK}\lambda_K^2r_K\beta_K \end{bmatrix}.$$

5. Заключение

В данной работе исследована математическая модель системы $M|GI|\infty$, функционирующей в условии изменяющейся внешней среды. С помощью метода асимптотического анализа при условии эквивалентно растущего времени обслуживания доказано, что распределение вероятностей числа занятых приборов разного типа в системе является многомерным гауссовским.

На основе сравнения результатов имитационного моделирования с асимптотическими результатами была определена область применимости гауссовской аппроксимации.

Литература

1. Дудин С.А., Дудина О.С. Многоканальная система обслуживания с марковским входным потоком нетерпеливых запросов, функционирующая в случайной среде // Информатика. 2015. № 1. С. 45-55.
2. Таташев А.Г. Система массового обслуживания с переменной интенсивностью входного потока // Автоматика и телемеханика. 1995. № 12. С. 78–84.
3. Baykal-Gursoy M., Xiao W. Stochastic Decomposition in $M|M|\infty$ Queues with Markov Modulated Service Rates. // Queueing Syst. 2004. V. 48. P. 75-88.
4. Blom J., Kella O., Mandjes M., Thorsdottir H. Markov-Modulated Infinite-Server Queues with General Service Times. // Queueing Syst. 2014. V. 76. P. 403-424.
5. D'Auria B. $M|M|\infty$ queues in semi-Markovian random environment. // Queueing Syst. 2008. V. 58. P. 221-237.
6. Fralix B.H., Adan I.J.B.F. An Infinite-Server Queue Influenced by a Semi-Markovian Environment. // Queueing Syst. 2009. V. 61. P. 65-84.
7. Linton D., Purdue P. An $M|G|\infty$ Queue with m Customer Types Subject to Periodic Clearing. // Opsearch, 1979. V. 16 P. 80-88.
8. Назаров А.А., Баймеева Г.В. Исследование системы $M|M|\infty$ в полумарковской случайной среде // Известия вузов. Физика. 2015. Т. 58, № 11/2. С. 198–203.
9. Назаров А.А., Моисеева С.П. Метод асимптотического анализа в теории массового обслуживания. Томск: Изд-во НТЛ. 2006. 112 с.
10. Убонова Е.Г., Панкратова Е.В. Гауссовская аппроксимация для системы массового обслуживания $MMPP|M|\infty$ с разнотипным обслуживанием // Известия вузов. Физика. 2015. Т. 58, № 11/2. С. 225–230.

UDC: 004.94

The Simulation of Finite-Source Retrial Queueing Systems With Two-Way Communications to the Orbit and Impatient Customers

János Sztrik [1], Ádám Tóth [1], Ákos Pintér [1], Zoltán Bács[1]

[1]University of Debrecen, Debrecen 4032, Hungary

{toth.adam,sztrik.janos}@inf.unideb.hu, bacs.zoltan@econ.unideb.hu,
apinter@science.unideb.hu

Abstract

The aim of the paper is to analyze a M/M/1//N finite-source, two-way communication retrial queueing system with an unreliable server and impatient customers. In this model, every request in the source is eligible to generate customers when the server does not function but they are forwarded immediately to the orbit. Customers may depart from the system during its waiting in the orbit after a random time and they get back to the source. All random variables involved in the model construction are supposed to be independent of each other. The novelty of the investigation is to carry out a sensitivity analysis comparing various distributions of failure time on the performance measures such as the mean number of customers in the orbit, the mean waiting time of an arbitrary customer, the probability of abandonment, etc. With the help of our self-developed simulation program, results are illustrated graphically.

Keywords: Simulation, blocking, sensitivity analysis, finite-source queueing system, unreliable server, retrial queue, impatient customers.

1. Introduction

Nowadays, network traffic increases in such a way that the design and optimisation of communication systems are required. This phenomenon can be followed both in the industrial sector like in the companies and in our homes due to the quick technological development and the great number of devices capable of IP communication. Therefore, researchers dedicate enough time to create new suitable models of telecommunication systems or adjust the current ones.

Retrial queues play quite an important role to depict real-life problems emerging from main telecommunication systems like telephone switching systems, call centers,

The research was supported by the Thematic Excellence Programme (TKP2020-IKA-04) of the Ministry for Innovation and Technology in Hungary.

computer networks, and computer systems. Investigating the available literature in the Internet many papers address topics related to retrial-queuing systems with repeated calls. In [1],[2],[3],[4] you can see some examples of it. In many areas of science analyzing these models can improve the efficiency of systems or bring about new advantageous features for example in the case of local-area networks with random access protocols and with multiple access protocols [5],[6].

Speaking of two-way communication, it possesses favorable impacts on most of the systems. Because similarities can be observed with the operation of certain real-life systems ergo it is no wonder that models based on a two-way communication scheme are introduced in many papers. This is particularly suitable in the case of call centers where the service unit (or agent) performs other actions pertaining to selling, promoting, and advertising products apart from satisfying incoming calls. In our model, the server may perform that action (calling customers residing in the orbit) after some random time when it is functional and no request is under service. Examining such scenarios has a great influence on the utilization of the service unit (or workload of agents) that is an important aspect and extensively examined by several papers like [7],[8].

Studying the related articles I found the assumption of having a service unit available all the time which is quite impractical regarding events in real-life applications of systems for example power outages, human error, or other failures. Although companies, providers want to ensure having fault-tolerant devices and services (the intention is to have high-availability scenarios), problems can occur at any time. Not to mention wireless communication where other factors could affect the transmission rate of the wireless channel and the forwarded information prone to undergo failure interruptions throughout transferring the packets. That is why random server breakdowns and repairs are centric topics so the inspection of these features alters undoubtedly the operation of systems, the system characteristics, and the performance measures. Finite-source retrial queues with server breakdowns have been studied in several papers like [9],[10],[3],[11],[12].

The main aim of this work is to investigate the operation of such a system containing a non-reliable service unit and customers which may leave the system without obtaining their service needs. The novelty of this investigation is to carry out a sensitivity analysis using different distributions of failure time on performance measures like the mean waiting time of an arbitrary customer, a customer leaving the system through the orbit, or the total utilization of the server. A simulation program is developed to accomplish our goal namely checking the effects of the distributions.

Our program is based on SimPack toolkit [13] which is a collection of C and C++ libraries. Several approaches and algorithms are supported providing a set of utilities to build a working simulation from a model description. Simpack contains very basic building blocks, during the coding of the model several functions, random number generator, and features were integrated. With the help of this program, results are illustrated graphically. This paper is the natural continuation of [14].

2. Model description and notations

The considered retrial queueing system of type $M/M/1//N$ contains a two-way communication feature and impatient customers. N customers are located in the finite-source where each of them can generate calls towards the server according to an exponential distribution with rate λ/N. In this model, every customer is characterized by an impatience feature that determines the maximum spent time of a customer in the orbit before leaving the system without completing its service requirement. This random variable also follows an exponential distribution with parameter τ. The model does not contain waiting queues therefore if the service unit is idle the service of an incoming customer starts immediately which is exponentially distributed with parameter μ. Upon its completion, request goes back to the source. Otherwise, the incoming customer is delivered to the orbit remaining in the system and after an exponentially distributed time with parameter σ/N they launch another attempt to reach the service facility. Our assumption is that every now and then the server breaks down according to gamma, hypo-exponential, hyper-exponential, Pareto, and lognormal distribution with different parameters but with the same mean value.

Throughout this period customers may proceed to produce their requests but they are transferred to the orbit immediately. The repair process is initiated instantaneously upon the failure of the server, which is also an exponentially distributed random variable with parameter γ_2. When the server breaks down during the service of a customer the execution will be cancelled and the customer returns to the orbit instantly. The feature of two-way communication is when the server becomes idle it may accomplish an outgoing call (secondary customers) after an exponentially distributed random time with rate ν that results in calling a customer in the orbit earlier. The service of these customers follows an exponential distribution with rate μ_2. Rates λ/N and σ/N are used because in [15],[16] very similar systems are evaluated by an asymptotic method where N tends to infinity, and was proved that the number of customers in the system follows a normal distribution. All the random variables in the model creation are assumed to be totally independent of each other.

3. Simulation

As mentioned earlier SimPack is the base of our simulation program which consists of a statistic package [17]. The method of batch means is applied and with the help of this tool, it is possible to perform a quantitative estimation of the mean and variance values of the desired variables. The fundamental operation of this method is that in every batch n observations take place and the useful run is divided into numerous batches. For having a valid and correct estimation the batches should be long enough and approximately independent of each other. This is one of the most popular mechanisms among the confidence interval techniques for a steady-state mean of a process. The following works [18],[19] comprise very precise description and algorithm about batch means. The simulations are performed with a confidence level of 99.9%. The relative half-width of the confidence interval required to stop the simulation run is 0.00001.

3.1. Simulation results. Four different distributions of failure time are used to investigate their effects on the main performance measures. To have a valid comparison we selected the parameters in such a way that the mean value and variance would be equal. Before that, a fitting process is necessary to be done to obtain the correct values of parameters. [20] describes in more detail the characteristics of the utilized distributions. In the first scenario, the squared coefficient of variation is greater than one therefore we utilized hyper-exponential, gamma, Pareto, and lognormal distributions and compared them with each other. Table 1 and Table 2 shows every values of the random variables including all the used input parameters of the various distributions of failure time as well.

Table 1. Used numerical values of model parameters

N	λ/N	γ_2	σ/N	μ	μ_2	ν
100	0.01	1	0.01	1	1.2	0.02

Table 2. Parameters of failure time

Distribution	**Gamma**	**Hyper-exponential**	**Pareto**	**Lognormal**
Parameters	$\alpha = 0.6$ $\beta = 0.5$	$p = 0.25$ $\lambda_1 = 0.41667$ $\lambda_2 = 1.25$	$\alpha = 2.2649$ $k = 0.67018$	$m = -0.3081$ $\sigma = 0.99037$
Mean	1.2			
Variance	2.4			
Squared coefficient of variation	1.6666666667			

Figure 1 shows the mean waiting time of an arbitrary customer in the function of arrival intensity. The disparity is quite obvious taking a closer look at the figure that represents the impact on the metrics using various distributions having the same first two moments. Customers spend by far more time in the orbit at Pareto

distribution and the least at gamma distribution. Also the interesting maximum property characteristic of a finite-source retrial queueing system occurs despite the increasing arrival intensity.

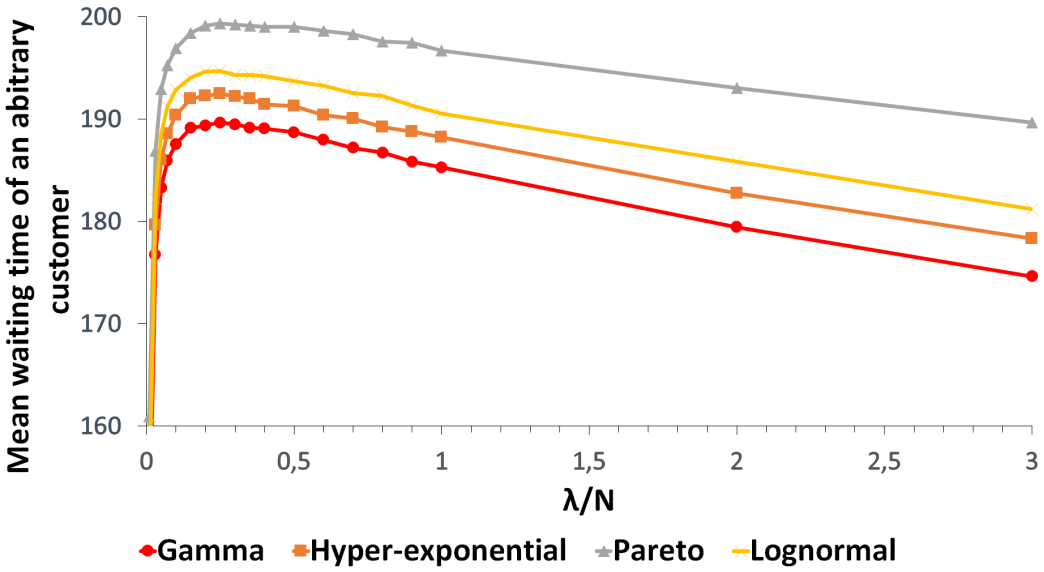


Fig. 1. Mean waiting time of an arbitrary customers

Figure 2 highlights the property of impatience under different parameter setting showing how the mean waiting of an arbitrary customer develops beside increasing arrival intensity. Actually the expected behaviour happens namely as the probability of leaving the system earlier increases fewer customers will be located in the system. This is logical and the results confirm our suspicion. However, impatience does not change the maximum property characteristic, it is clearly visible that every curve has a maximum value. Due to the page limitation other results and scenarios are intended to be published in the extended version of the paper.

4. Conclusion

We presented a finite-source retrial queueing system with an unreliable server that may call in requests residing in the orbit (two-way communication property) and impatient customers. The obtained results demonstrate the effect of impatience on the visualized performance measures indicating that customers with less patience depart much earlier without reaching the service facility. Results also display the influence of various distributions of failure time on the performance measures when

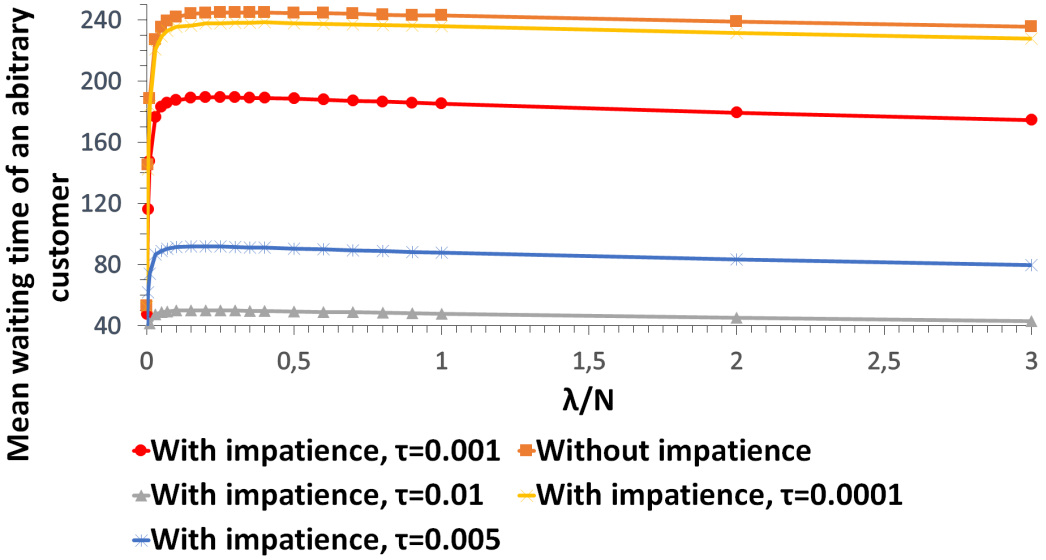


Fig. 2. The effect of impatience on the mean waiting time

the squared coefficient of variation was greater than one despite the fact that mean and variance are equal. In the case of less than one significant differences do not appear, curves almost totally overlap each other. In the future we plan to complete that system including other features like trying out other distributions or introducing disaster failure, or including more capacity of service.

REFERENCES

1. G. Falin, J. Artalejo, A finite source retrial queue, European Journal of Operational Research 108 (1998) 409–424.
2. D. Fiems, T. Phung-Duc, Light-traffic analysis of random access systems without collisions, Annals of Operations Research (2017) 1–17.
3. A. Krishnamoorthy, P. K. Pramod, S. R. Chakravarthy, Queues with interruptions: a survey, TOP 22 (1) (2014) 290–320.
4. B. K. Kumar, G. Vijayalakshmi, A. Krishnamoorthy, S. S. Basha, A single server feedback retrial queue with collisions, Computers & Operations Research 37 (7) (2010) 1247–1255.
5. J. Artalejo, A. G. Corral, Retrial Queueing Systems: A Computational Approach, Springer, 2008.

6. J. Kim, B. Kim, A survey of retrial queueing systems, Annals of Operations Research 247 (1) (2016) 3–36.
7. V. Dragieva, T. Phung-Duc, Two-way communication $M/M/1//N$ retrial queue, in: International Conference on Analytical and Stochastic Modeling Techniques and Applications, Springer, 2017, pp. 81–94.
8. A. Kuki, J. Sztrik, Á. Tóth, T. Bérczes, A Contribution to Modeling Two-Way Communication with Retrial Queueing Systems , in: Information Technologies and Mathematical Modelling. Queueing Theory and Applications, Springer, 2018, pp. 236–247.
9. V. I. Dragieva, Number of retrials in a finite source retrial queue with unreliable server., Asia-Pac. J. Oper. Res. 31 (2) (2014) 23. doi:10.1142/S0217595914400053.
10. N. Gharbi, B. Nemmouchi, L. Mokdad, J. Ben-Othman, The impact of breakdowns disciplines and repeated attempts on performances of small cell networks, Journal of Computational Science 5 (4) (2014) 633–644.
11. F. Zhang, J. Wang, Performance analysis of the retrial queues with finite number of sources and service interruptions, Journal of the Korean Statistical Society 42 (1) (2013) 117–131. doi:10.1016/j.jkss.2012.06.002.
12. Á. Tóth, T. Bérczes, J. Sztrik, A. Kvach, Simulation of finite-source retrial queueing systems with collisions and a non-reliable server, in: International Conference on Distributed Computer and Communication Networks, Springer, 2017, pp. 146–158.
13. P. A. Fishwick, Simpack: Getting started with simulation programming in c and c++, in: In 1992 Winter Simulation Conference, 1992, pp. 154–162.
14. J. Sztrik, Á. Tóth, Á. Pintér, Z. Bács, The simulation of finite-source retrial queueing systems with two-way communications to the orbit and blocking, in: V. M. Vishnevskiy, K. E. Samouylov, D. V. Kozyrev (Eds.), Distributed Computer and Communication Networks: Control, Computation, Communications, Springer International Publishing, Cham, 2020, pp. 171–182.
15. A. Nazarov, J. Sztrik, A. Kvach, A survey of recent results in finite-source retrial queues with collisions, in: Information Technologies and Mathematical Modelling. Queueing Theory and Applications, Springer, 2018, pp. 1–15.
16. A. Nazarov, J. Sztrik, A. Kvach, T. Bérczes, Asymptotic analysis of finite-source M/M/1 retrial queueing system with collisions and server subject to breakdowns and repairs, Annals of Operations Research 277 (2) (2019) 213–229.
17. A. Francini, F. Neri, A comparison of methodologies for the stationary analysis of data gathered in the simulation of telecommunication networks, in: Proceedings of MASCOTS '96 - 4th International Workshop on Modeling, Analysis and Simulation of Computer and Telecommunication Systems, 1996, pp. 116–122.

18. E. J. Chen, W. D. Kelton, A procedure for generating batch-means confidence intervals for simulation: Checking independence and normality, SIMULATION 83 (10) (2007) 683–694.
19. A. M. Law, W. D. Kelton, Simulation Modeling and Analysis, McGraw-Hill Education, 1991.
20. A. Toth, J. Sztrik, A. Kuki, T. Berczes, D. Efrosinin, Reliability analysis of finite-source retrial queues with outgoing calls using simulation, in: 2019 International Conference on Information and Digital Technologies (IDT), 2019, pp. 504–511.

UDC: 004.94

Simulation of Two-Way Communication Retrial Queuing Systems With Non-reliable Server, Impatient Customers to the Orbit and Blocking

Ádám Tóth [1], János Sztrik [1], Tamás Bérczes [1], Attila Kuki[1]

[1]University of Debrecen, Debrecen 4032, Hungary

{toth.adam,sztrik.janos,berczes.tamas,kuki.attila}@inf.unideb.hu

Abstract

The goal of this paper is to carry out a sensitivity analysis to examine the effect of different distributions of service time when blocking is applied with the help of retrial queueing systems having the property of two-way communication. This eventuates in outgoing calls (secondary customers) which are performed by the service unit after a random time in its idle state. Primary customers arrive from the finite-source according to an exponential distribution. This model does not contain queues so the service of an incoming request starts immediately if the server is functional and in an idle state. Impatience of the customers and server failures are characterized by this system which also follows an exponential distribution. The novelty of the investigation is to illustrate the effect of blocking with several figures obtained by simulation using various distributions of service time on the desired performance measures.

Keywords: Simulation, blocking, sensitivity analysis, finite-source queueing system, unreliable server, retrial queue, impatient customers.

1. Introduction

The explosive growth of network traffic in recent years evokes the necessity of investigating communication networks to understand the behaviour of different systems. More and more communication sessions evolve partly almost every device becomes "smart" leading to higher bandwidth requirements not just in multinational companies but in our homes as well. So many unknown quantities may modify the performance of networking systems making them very complex and difficult

The work of Ádám Tóth is supported by the ÚNKP-20-4 new national excellence program of the ministry for innovation and technology from the source of the national research,development and innovation fund. The research work of János Sztrik, Attila Kuki and Tamás Bérczes was supported by the construction EFOP-3.6.3-VEKOP-16-2017-00002. The project was supported by the Euro-pean Union, co-financed by the European Social Fund.

to realize every aspect of their operation. Consequently, researchers dedicate their time to develop mathematical models describing telecommunication systems. With the help of retrial queueing systems arising real-life problems can be modelled in main telecommunication systems like telephone switching systems, call centers, or computer systems. These systems possess a virtual waiting room the so-called orbit where customers get into when the service unit is unavailable. Some examples are listed where queueing models are utilized: [1],[2].

In this paper, the customer owns the impatience feature meaning that customers are able to decide to leave the system earlier without obtaining its service requisition. This is a natural occurrence of human behaviour and can be experienced in many fields of life like in healthcare applications, call centers, telecommunication networks so various works examine the effect of this phenomenon like in [3], [4], [5]. In these articles impatient request is portrayed: if the queue is sufficiently long balking customers choose to avoid entering the system, jockeying customers can alter queues if they encounter them may get served faster, and reneging customers leave the queue if they have waited a definite time for service.

Examining the available literature the considered models include service units that are assumed to be accessible all the time. This hypothesis does not reflect the reality as unexpected errors can take place like power outages, human negligence, or other sudden actions. Although devices are developing and become more reliable, unfortunate failures have a massive effect on the operation of the system modifying the performance measures significantly hence retrial queuing systems have been investigated in several papers recently for example in [6],[7],[8],[9].

Two-way communication scheme gains ground ultimately due to its usefulness in many application fields modelling arising actual problems. One prime example is call-center where service units in an idle state may perform other activities besides satisfying the needs of incoming calls including selling, advertising, and promoting products. In other words, whenever the server is idle it may call for customers outside of the system after a random time. Utilization of such systems is always a key issue in that way many scientists are trying to optimize the service of different requests see for example [10],[11],[12],[13].

The main focus of this paper is to carry out a sensitivity analysis inspecting the various distributions of service time of primary customers when blocking is applied on the main performance measures for instance the mean waiting time and variance of an arbitrary, successfully served and impatient customer, the total utilization of

the service unit, the probability of abandonment. Because giving exact formulas are difficult especially when one of the variables does not follow exponential distribution, the obtained results are gathered by stochastic simulation based on SimPack [14] which contains the basic building blocks of the code. One of the main motivation is to develop simulation models in this way because it gives us the freedom to calculate any performance measure which we desire using various values of input parameters. The achieved results indicate the relevance of the used distributions using various parameter settings and the effect of blocking illustrated by numerous figures concentrated on the interesting phenomena of these systems.

2. System model

The regarded system is a retrial queueing system of type $M/G/1//N$ with impatient customers and an unreliable server that is capable of producing outgoing calls. N denotes the number of sources where each individual generates requests according to an exponential distribution with rate λ/N so the distribution of inter-request time is exponential with parameter λ/N. There are no queues in our model in this way whenever an incoming customer finds the server in a busy state, it will be forwarded to the orbit. Otherwise, the service of an incoming customer starts instantly that follows gamma, hypo-exponential, hyper-exponential, Pareto, and lognormal distribution with different parameters but with the same mean value. During its residence in the orbit a customer may launch an attempt to reach the service unit after an exponentially distributed time with parameter σ/N. Call generation can not occur until the end of the successful service of the individual in the source. We suppose that the service unit breaks down after an exponentially distributed time interval with parameter γ_0 when it is busy and with parameter γ_1 when idle. The repair time is also an exponentially distributed random variable with parameter γ_2 which starts instantly after a failure takes place. During faulty period requests can not enter the system because of blocking. Customers have impatient characteristics therefore they may decide to leave the system after waiting a random exponential time in the orbit with rate τ. As mentioned earlier an idle server may perform an outgoing call towards the customers (secondary) from an infinite source after an exponentially distributed time with parameter γ. The service of secondary customers is a gamma distributed random variable with parameters α_2 and β_2. At the time the secondary request is arriving, if the server is busy or non-operational then it will be cancelled and returns without entering the system. In the case of breakdown:

- The service of a primary request is interrupted and it is forwarded immediately towards the orbit.
- The service of a secondary request is also interrupted but it departs the system.

3. Simulation

As mentioned earlier results are obtained by a self-developed simulation program and a statistical package [15] was integrated into our code to determine the performance measures. The method of batch means is used where the useful run is divided into N batches thus $n = M - K/N$ observations are carried out in every batch. K represents the warm-up period observations at the beginning of the simulation which is rejected. M represents the length of simulation. We just simply calculate the sample average of the whole run after the warm-up period. To have a valid estimation batches should be long enough and the sample averages of the batches should be approximately independent. In the following articles you can find more information about this process [16], [17]. The simulations are performed with a confidence level of 99.9%. The relative half-width of the confidence interval required to stop the simulation run is 0.00001. The size of a batch used to detect the initial transient duration is 1000.

Table 1 display the used values of input parameters in our scenarios.

$\mathbf{N}$	$\boldsymbol{\gamma_0}$	$\boldsymbol{\gamma_1}$	$\boldsymbol{\sigma/N}$	$\boldsymbol{\gamma}$	$\boldsymbol{\alpha_2}$	$\boldsymbol{\beta_2}$	$\boldsymbol{\tau}$
100	0.05	0.5	0.01	0.8	1	1	0.001

Table 1. Numerical values of model parameters

Table 2. Parameters of service time of primary customers

Distribution	**Gamma**	**Hyper-exponential**	**Pareto**	**Lognormal**
Parameters	$\alpha = 0.037$ $\beta = 0.015$	$p = 0.482$ $\lambda_1 = 0.385$ $\lambda_2 = 0.416$	$\alpha = 2.018$ $k = 1.261$	$m = -0.751$ $\sigma = 1.826$
Mean	2.5			
Variance	169			
Squared coefficient of variation	27.04			

3.1. Simulation results. We distinguished different scenarios where the values of service times of incoming customers are different to check how the various distribution modify the operation of the system. First, the squared coefficient of variation is greater than one, and to have a valid comparison we chose the parameters that the mean and variance would be the same in every case. For this, a fitting process was performed and [18] contains detailed info about these mechanisms.

Figure 1 demonstrates the mean waiting time of an arbitrary customer in the function of arrival intensity when the service time of the customer follows a gamma distribution. The results prove what we expected aforehand when blocking is applied, lower mean waiting time is obtained especially besides higher arrival intensity. That ratio is true for the other used distributions as well. Although having the same first

two moments maximum property characteristic of a finite-source retrial queueing system arises even with the appearance of blocking.

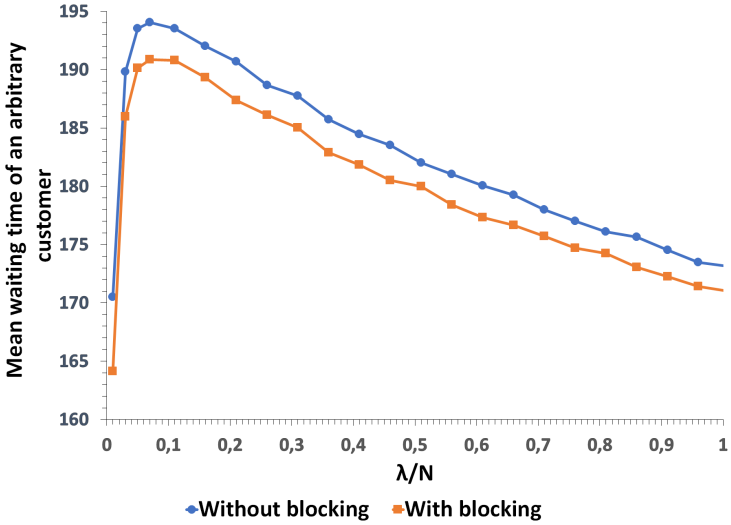


Fig. 1. The effect of blocking on the mean waiting of an arbitrary customer besides service time of gamma distribution

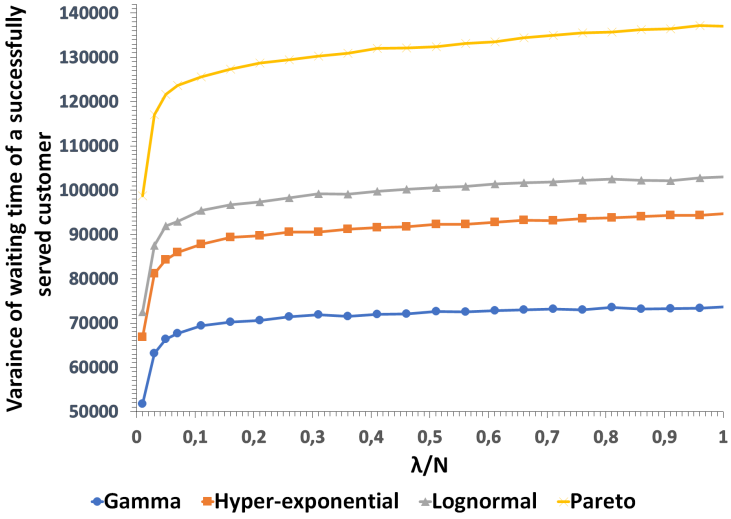


Fig. 2. The variance of waiting time of a successfully served customer

The variance of waiting time of a successfully served customer is depicted in Figure 2 versus arrival intensity. Interestingly the differences are significant among the used distributions in spite of the selected parameters having the same first two moments. This is especially remarkable if we compare the values at gamma distribution with the values at Pareto distribution. This performance measure starts to escalate rapidly and after λ/N reaches 0.1 variance stagnates around a certain value. Due to the page limitation results in connection with the squared coefficient of variation are less than one will be published in the extended version of the paper.

4. Conclusion

We introduced a retrial queueing system of type $M/G/1//N$ with impatient customers in the orbit and with an unreliable server having a two-way communication feature from an infinite source when blocking is implemented. Results are obtained by stochastic simulation and it is shown that the stationary probability distribution of the number of customers in the orbit tends to correspond to the Gaussian distribution despite the used distribution of service time of the primary customers. We investigated different scenarios for example when the squared coefficient of variation is greater than one the obtained values of mean waiting time of an arbitrary, successfully served customer significantly differ from each other even though the parameters are chosen that the mean and variance would be equal in case of every distribution. Results also revealed the effect of blocking which lowers the value of mean waiting time and the number of customers in the system. In our second scenario when the squared coefficient of variation is less than one interestingly the curves almost overlap each other minor disparity turns up examining all the desired performance measures. In the future the authors intend to continue their research work, analyzing other features of the system like collisions, outgoing calls toward the customers from the orbit, or carrying out sensitivity analysis on other random variables.

REFERENCES

1. J. Artalejo, A. G. Corral, Retrial Queueing Systems: A Computational Approach, Springer, 2008.
2. D. Fiems, T. Phung-Duc, Light-traffic analysis of random access systems without collisions, Annals of Operations Research (2017) 1–17.
3. N. Gupta, Article: A view of queue analysis with customer behaviour and priorities, IJCA Proceedings on National Workshop-Cum-Conference on Recent Trends in Mathematics and Computing 2011 RTMC (4) (May 2012).
4. R. Kumar, N. Jain, B. Som, Optimization of an $M/M/1/N$ feedback queue with retention of reneged customers, Operations Research and Decisions 24 (2014) 45–58. doi:https://doi.org/10.5277/ord140303.

5. G. Panda, V. Goswami, A. Datta Banik, D. Guha, Equilibrium balking strategies in renewal input queue with bernoulli-schedule controlled vacation and vacation interruption, Journal of Industrial and Management Optimization 12 (2015) 851–878. doi:https://doi.org/10.3934/jimo.2016.12.851.
6. V. I. Dragieva, Number of retrials in a finite source retrial queue with unreliable server., Asia-Pac. J. Oper. Res. 31 (2) (2014) 23. doi:10.1142/S0217595914400053.
7. N. Gharbi, C. Dutheillet, An algorithmic approach for analysis of finite-source retrial systems with unreliable servers, Computers & Mathematics with Applications 62 (6) (2011) 2535–2546.
8. N. Gharbi, M. Ioualalen, GSPN analysis of retrial systems with servers breakdowns and repairs, Applied Mathematics and Computation 174 (2) (2006) 1151–1168. doi:10.1016/j.amc.2005.06.005.
9. N. Gharbi, B. Nemmouchi, L. Mokdad, J. Ben-Othman, The impact of breakdowns disciplines and repeated attempts on performances of small cell networks, Journal of Computational Science 5 (4) (2014) 633–644.
10. V. Dragieva, T. Phung-Duc, Two-way communication $M/M/1//N$ retrial queue, in: International Conference on Analytical and Stochastic Modeling Techniques and Applications, Springer, 2017, pp. 81–94.
11. A. Kuki, J. Sztrik, Á. Tóth, T. Bérczes, A Contribution to Modeling Two-Way Communication with Retrial Queueing Systems , in: Information Technologies and Mathematical Modelling. Queueing Theory and Applications, Springer, 2018, pp. 236–247.
12. S. Pustova, Investigation of call centers as retrial queuing systems, Cybernetics and Systems Analysis 46 (3) (2010) 494–499.
13. T. Wolf, System and method for improving call center communications, uS Patent App. 15/604,068 (Nov. 30 2017).
14. P. A. Fishwick, Simpack: Getting started with simulation programming in c and c++, in: In 1992 Winter Simulation Conference, 1992, pp. 154–162.
15. A. Francini, F. Neri, A comparison of methodologies for the stationary analysis of data gathered in the simulation of telecommunication networks, in: Proceedings of MASCOTS '96 - 4th International Workshop on Modeling, Analysis and Simulation of Computer and Telecommunication Systems, 1996, pp. 116–122.
16. E. J. Chen, W. D. Kelton, A procedure for generating batch-means confidence intervals for simulation: Checking independence and normality, SIMULATION 83 (10) (2007) 683–694.
17. A. M. Law, W. D. Kelton, Simulation Modeling and Analysis, McGraw-Hill Education, 1991.
18. J. Sztrik, Á. Tóth, Á. Pintér, Z. Bács, Simulation of finite-source retrial queues with two-way communications to the orbit, in: A. Dudin, A. Nazarov, A. Moiseev

(Eds.), Information Technologies and Mathematical Modelling. Queueing Theory and Applications, Springer International Publishing, Cham, 2019, pp. 270–284.

UDC: 519.872.5

Asymptotic Analysis of a Closed Exponential Queueing Network with Unreliable Nodes

T.V. Rusilko

Yanka Kupala State University of Grodno, 22 Ozheshko St, Grodno, Belarus

tatiana.rusilko@gmail.com

Abstract

A closed exponential queueing network with unreliable nodes was studied. The process of changing the number of serviceable servers in network nodes was considered as the birth–death process. The process of changing the number of customers at the nodes was studied under the asymptotic assumption of a large number of customers. The last-mentioned process divided by the large number of customers is a continuous stochastic process with the Markov property. It was proved that its probability density function satisfies the Fokker–Planck–Kolmogorov equation. The system of differential equations for the first-order and second-order moments of this process was derived.

Keywords: queueing network, unreliable queueing node, birth–death process, asymptotic analysis, unreliable queueing system

1. Introduction

The study of queueing networks with multi-server nodes (systems) in case of server breakdowns and repairs is important for practical applications. It is a priori obvious that the queue lengths at the nodes depend on systematic server failures. The purpose of this paper is to asymptotically study a closed exponential network with unreliable queueing nodes. The unreliability of nodes lies in the fact that their servers can be broken down and be repaired according to a certain statistical law. Asymptotic analysis implies an approximation method of queueing network study under the assumption of a large number of customers (requests) [1].

Discrete (discontinuous-state) Markov processes are usually used to determine the state of queueing networks. In this paper, the passage to the limit from a Markov chain to a continuous-state Markov process was considered. In contrast to discontinuous processes, continuous processes in any small time interval $\Delta t \to 0$ have some small change in the state $\Delta x \to 0$. The mathematical approach used in this paper is based on a discrete model of a continuous Markov process described in many books on the theory of diffusion Markov processes [2, 3].

2. Formulation of the problem

A closed exponential queueing network with n nodes S_i, $i = \overline{1,n}$, is under study. The node S_i is a $\cdot/M/m_i$ type unreliable queueing system where all servers are identical and they have exponentially distributed service time, μ_i is the reciprocal of the mean service time, $i = \overline{1,n}$. Requests for service are selected accordingly to the discipline FIFO. The transition matrix is $P = \|p_{ij}\|_{n\times n}$, $\sum\limits_{j=1}^{n} p_{ij} = 1$, $i,j = \overline{1,n}$.

Servers in nodes $S_1, ..., S_n$ are subject to random failure. The continuous serviceable time of each server in S_i is exponentially distributed with the rate parameter α_i, $i = \overline{1,n}$. Server lifespan does not depend on whether the device is busy or not. The server immediately starts to be repaired after the failure. The server repair time in S_i also has an exponential distribution with the rate parameter β_i, $i = \overline{1,n}$. Suppose if the server fails during the customer service time, the interrupted customer will be completed after server repair. Let us assume that the server service time, server uptime, and server repair time are independent random variables.

The state of the network under study at time t is represented by a vector

$$(z(t); k(t)) = (z_1(t), z_2(t), ..., z_n(t); k_1(t), k_2(t), ..., k_n(t)), \tag{1}$$

where $z_i(t)$ is the number of serviceable servers, $k_i(t)$ is the number of customers in the queueing system S_i at the time t, $0 \le z_i(t) \le m_i$, $0 \le k_i(t) \le K$, $i = \overline{1,n}$, $t \in [0, +\infty)$. Since the network is closed, it is obvious that $\sum\limits_{i=1}^{n} k_i(t) = K$. The state vector (1) can be viewed as two simultaneous random processes.

I. The process of changing the number of serviceable servers in network nodes is $z(t) = (z_1(t), z_2(t), ..., z_n(t))$. Server failures and repairs in different queueing nodes are assumed to occur completely independently and regardless of the number of requests in these nodes. Therefore, the vector elements $z_i(t)$ are independent stochastic processes and the values of $z_j(t)$ are not determined by the values of $k_i(t)$, $i,j = \overline{1,n}$.

The process $z_i(t)$ can be considered as the birth–death process with birth rates β_i, death rates α_i and the finite integer state space $Z_i = \{0, 1, 2, ..., m_i\}$, $i = \overline{1,n}$. Denote $p_{z_i}^{(i)}(t) = P(z_i(t) = z_i)$ is the probability that system S_i has z_i serviceable servers at time t, $z_i \in \mathbb{Z}$, $0 \le z_i \le m_i$, $i = \overline{1,n}$. The differential equations for the probabilities $p_{z_i}^{(i)}(t)$ are well known

$$\begin{gathered} p_0^{(i)\prime}(t) = -\beta_i p_0^{(i)}(t) + \alpha_i p_1^{(i)}(t), \\ p_{z_i}^{(i)\prime}(t) = -(\alpha_i + \beta_i) p_{z_i}^{(i)}(t) + \alpha_i p_{z_i+1}^{(i)}(t) + \beta_i p_{z_i-1}^{(i)}(t), 1 \le z_i \le m_i - 1, \\ p_{m_i}^{(i)\prime}(t) = \beta_i p_{m_i-1}^{(i)}(t) - \alpha_i p_{m_i}^{(i)}(t), i = \overline{1,n}. \end{gathered} \tag{2}$$

The non-stationary probability distribution $p_{z_i}^{(i)}(t)$, $0 \le z_i \le m_i$, $i = \overline{1,n}$, can be found by solving system (2) with a certain initial condition. The expected value of serviceable servers at the node S_i is $E(z_i(t)) = \sum\limits_{z_i=0}^{m_i} z_i p_{z_i}^{(i)}(t)$, $i = \overline{1,n}$.

II. The process $k(t) = (k_1(t), k_2(t), ..., k_n(t))$ of changing the number of customers at the nodes related to the customer service process and to customer transitions between network nodes is a continuous-time Markov chain. The process $k(t)$ is determined by the service process at nodes and therefore depends on the number of servers, therefore $k(t)$ is determined by the random process $z(t)$. To sum up, it may be said $k(t)$ is a nested process with respect to $z(t)$.

Purpose of the study is to derive systems of differential equations for the first-order and second-order moments of the vector $k(t)$ in the asymptotic case of a large K. Probabilities $p_{z_i}^{(i)}(t)$, $i = \overline{1,n}$, are assumed to be predetermined from (2).

3. Asymptotic analysis

Theorem 1. In the asymptotic case of a large number of customers K the probability density function $p(x,t)$ of the random process $\xi(t) = \left(\frac{k(t)}{K}\right) = \left(\frac{k_1(t)}{K}, \frac{k_2(t)}{K}, ..., \frac{k_n(t)}{K}\right)$ provided that it is differentiable with respect to t and twice continuously differentiable with respect to x_i, $i = \overline{1,n}$, satisfies up to $O(\varepsilon^2)$, where $\varepsilon = \frac{1}{K}$, the multidimensional Fokker–Planck–Kolmogorov equation

$$\frac{\partial p(x,\ t)}{\partial t} = -\sum_{i=1}^{n} \frac{\partial}{\partial x_i} \left(A_i(x,t)p(x,t)\right) + \frac{\varepsilon}{2} \sum_{i,j=1}^{n} \frac{\partial^2}{\partial x_i \partial x_j} \left(B_{ij}(x,t)p(x,t)\right), \qquad (3)$$

with drifts

$$A_i(x,t) = \sum_{j=1}^{n} \sum_{z_j=0}^{m_j} \mu_j \min(\varepsilon z_j, x_j) p_{z_j}^{(j)}(t)(p_{ji} - \delta_{ij})$$

and diffusion coefficients (δ_{ij} is the Kronecker delta)

$$B_{ii}(x,t) = \sum_{j=1}^{n} \sum_{z_j=0}^{m_j} \mu_j \min(\varepsilon z_j, x_j) p_{z_j}^{(j)}(t)(p_{ji} + \delta_{ij}),$$

$$B_{ij}(x,t) = -\sum_{z_i=0}^{m_i} \mu_i \min(\varepsilon z_i, x_i) p_{z_i}^{(i)}(t) p_{ij}, i \ne j.$$

Proof. The process $k(t) = (k_1(t), k_2(t), ..., k_n(t))$ is a continuous-time Markov chain with a finite state space. Denote I_i as n-vector with zero components excluding i-th,

that is equals to 1. Having regard to all possible changes of $k(t)$ in the short time Δt, using the law of total probability, the following system of equations is valid for the probability $P(k,t) = P(k(t) = k)$:

$$P(k, t+\Delta t) = \sum_{i,j=1}^{n} \sum_{z_i=1}^{m_i} \mu_i \min(z_i, k_i(t)+1) p_{z_i}^{(i)}(t) p_{ij} P(k + I_i - I_j, t)\Delta t + \\ + \left(1 - \sum_{i=1}^{n} \sum_{z_i=1}^{m_i} \mu_i \min(z_i, k_i(t)) p_{z_i}^{(i)}(t)\Delta t\right) P(k,t) + o(\Delta t).$$

By simple transformations, we pass to the system of Kolmogorov difference-differential equations for these probabilities:

$$\frac{dP(k,t)}{dt} = \sum_{i,j=1}^{n} \sum_{z_i=0}^{m_i} \mu_i \min(z_i, k_i(t)) p_{z_i}^{(i)}(t) p_{ij} (P(k + I_i - I_j, t) - P(k,t)) + \\ + \sum_{i,j=1}^{n} \sum_{z_i=0}^{m_i} (\mu_i \min(z_i, k_i(t)+1) - \mu_i \min(z_i, k_i(t))) p_{z_i}^{(i)}(t) p_{ij} P(k + I_i - I_j, t).$$

Unfortunately, the last equation defies analytical solution for large n. In connection with this, consider the important asymptotic case of a large number of customers $K >> 1$ [1, 4–6]. Suppose that we are interested in the properties of the process $k(t)$ as K becomes very large. The vector of relative variables $\xi(t) = k(t)/K$ in a short time interval undergoes a state change by e_i, where $e_i = \varepsilon I_i$, $\varepsilon = 1/K$. In the case when $K \to \infty$ we have $\varepsilon \to 0$ and the vector $\xi(t) = \left(\frac{k_1(t)}{K}, \frac{k_2(t)}{K}, ..., \frac{k_n(t)}{K}\right)$ will be continuous-time continuous-state Markov processes with a probability density function $p(x,t)$, $X = \left\{x = (x_1, x_2, ..., x_n) : x_i \geq 0, i = \overline{1,n}, \sum_{i=1}^{n} x_i = 1\right\}$. The points $(x_1, x_2, ..., x_n)$ are located at the n-dimensional lattice vertex of the set X at a distance of ε from each other. At $K \to \infty$, the distance between the vertices decreases, $\varepsilon \to 0$, since we can also assume that the limiting distribution of $\xi(t)$ is continuous. The density $p(x,t)$ satisfies the asymptotic relation

$$K^n P(k,t) \xrightarrow[K\to\infty]{} p(x,t), x \in X. \tag{4}$$

Realizing the asymptotic transition (4), we obtain the following partial differential equation:

$$\frac{\partial p(x,t)}{\partial t} = K \sum_{i,j=1}^{n} \sum_{z_i=0}^{m_i} \mu_i \min(\varepsilon z_i, x_i) p_{z_i}^{(i)}(t) p_{ij} (p(x + e_i - e_j, t) - p(x,t)) + \\ + \sum_{i,j=1}^{n} \sum_{z_i=0}^{m_i} \mu_i \frac{\partial \min(\varepsilon z_i, x_i)}{\partial x_i} p_{z_i}^{(i)}(t) p_{ij} p(x + e_i - e_j, t). \tag{5}$$

If $p(x,t)$ is twice continuously differentiable function with respect to x, then

$$p(x+e_i-e_j,t)=p(x,t)+\varepsilon\left(\frac{\partial p(x,t)}{\partial x_i}-\frac{\partial p(x,t)}{\partial x_j}\right)+$$
$$+\frac{\varepsilon^2}{2}\left(\frac{\partial^2 p(x,t)}{\partial x_i^2}-2\frac{\partial^2 p(x,t)}{\partial x_i\partial x_j}+\frac{\partial^2 p(x,t)}{\partial x_j^2}\right)+o(\varepsilon^2).$$

Substituting this series into equation (5) and grouping terms, we obtain that $p(x,t)$ satisfies the multidimensional Fokker–Planck–Kolmogorov equation (5) with drifts $A_i(x,t)$ and diffusion coefficients $B_{ij}(x,t)$, $i,j=\overline{1,n}$. The error in this approximation is no more than $\varepsilon^2=(1/K)^2$. ■

Unfortunately, it is not possible to find an exact analytical solution of (3), so we have to make one more approximation.

4. System of differential equations for the first-order and second-order moments

The characteristic function of a stochastic process $\varphi(\lambda,t)=\int\limits_{\mathbb{R}^n} e^{I\lambda x^T}p(x,t)dx$, $I=\sqrt{-1}$, gives as much information about the process as the probability density. Multiplying both sides of (3) by $e^{I\lambda x^T}$, then integrating over x and taking into account certain initial and boundary conditions for equation (3) [3,4], we derive the equation for the characteristic function

$$\frac{\partial\varphi(\lambda,t)}{\partial t}=\int\limits_{\mathbb{R}^n}\left\{\sum_{i=1}^{n}I\lambda_i A_i(x,t)-\frac{\varepsilon}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}\lambda_i\lambda_j B_{ij}(x,t)\right\}p(x,t)e^{I\lambda x^T}dx.$$

According to one of the most important properties of the characteristic function, the first-order moment of $\xi_i(t)$ is defined as

$$\nu_i^{(1)}(t)=E\left(\xi_i(t)\right)=I^{-1}\left.\frac{\partial\varphi(\lambda,t)}{\partial\lambda_k}\right|_{\lambda=0},i=\overline{1,n}.$$

Mixed second-order moments are determined using the second-order derivative:

$$\nu_{ij}^{(1,1)}(t)=\nu_{ji}^{(1,1)}(t)=E\left(\xi_i(t)\xi_j(t)\right)=I^{-2}\left.\frac{\partial^2\varphi(\lambda,t)}{\partial\lambda_i\partial\lambda_j}\right|_{\lambda=0},i,j=\overline{1,n}.$$

It was found that the system of differential equations for the first-order and second-order moments of the state vector elements $\xi_i(t)$ is

$$\frac{d\nu_i^{(1)}(t)}{dt} = \frac{dE\left(\xi_i(t)\right)}{dt} = I^{-1}\left.\frac{\partial^2\varphi(\lambda,t)}{\partial t\partial\lambda_i}\right|_{\lambda=0} = E\left(A_i(\xi(t),t)\right), i=\overline{1,n};$$

$$\frac{d\nu_{ij}^{(1,1)}(t)}{dt} = \frac{dE\left(\xi_i(t)\xi_j(t)\right)}{dt} = I^{-2}\left.\frac{\partial^3\varphi(\lambda,t)}{\partial t\partial\lambda_i\partial\lambda_j}\right|_{\lambda=0} = E\left(\xi_i(t)A_j(\xi(t),t)\right) +$$
$$+E\left(\xi_j(t)A_i(\xi(t),t)\right) + \varepsilon E\left(B_{ij}(\xi(t),t)\right), i=\overline{1,n}, j=\overline{1,n}.$$

The technique described in [4–7] is used to calculate examples.

5. Conclusion

In this paper, an asymptotic method was presented for studying a closed exponential network with unreliable queueing systems under limiting condition of a large number of customers. This method makes it possible to predict the expected number of customers in network nodes in both transient and steady state. Second-order moments can be used to calculate the variability of the number of customers at the nodes and to study the correlation between the number of customers at different nodes with time. The greater the number of customers in the network, the greater the precision in calculation.

The areas of implementation of research results are the design of queueing networks, solving problems of their optimization and using them as models [6].

Pithy comments or suggestions would be greatly appreciated.

REFERENCES

1. Medvedev G. A. Closed Queueing Systems and Their Optimization // Proeedings of the USSR Academy of Sciences. Engineering Cybernetis. 1975. No. 6. P. 65–73.
2. Tikhonov V. I., Mironov M. A. Markov Processes. Soviet Radio, Moscow, 1977.
3. Gardiner K. V. Stochastic Methods in Natural Sciences. Mir, Moscow, 1986.
4. Rusilko T. V. The First Two Orders Moments Determination Method for the State Vector of the Queueing Network in the Asymptotic Case // Vesnik of Yanka Kupala State University of Grodno. Series 2. 2021. V. 11. No. 2. P. 152–161.
5. Matalytskiy M. A., Rusilko T. V. Approximate Methods for Analysis of Networks with a Central Queueing System and Their Applications. GrSU, Grodno, 2003.
6. Matalytskiy M. A., Rusilko T. V. Mathematical Analysis of Stochastic Models of Processing Claims of Various Types in Insurance Companies // Doklady of the National Academy of Sciences of Belarus. 2005. V. 49. No. 1. P. 18-23.
7. Koluzaeva E., Matalytski M. Analysis and Optimization of Queueing Networks. Lambert Academic Publishing, Saarbrucken, 2011.

UDC: 519.21, 004.94

Stability conditions for a multi-orbit retrial system with general retrials under classical retrial policy

R. S. Nekrasova[1,2]

[1]IAMR Karelian Research Centre RAS, Petrozavodsk, Russia
[2]Petrozavodsk State University, Petrozavodsk, Russia

Abstract

We consider a single server multi-class retrial system. The arrival customer, who meets the server busy, joins the corresponding orbit and then retries to capture the server. The model obeys to the classical retrial policy: the total rate of orbit customers depends on their number. Retrial times are assumed to be generally distributed, and that makes the analysis much more complicated. We use the previous results for the systems with exponential retrials and regenerative approach to establish the sufficient stability conditions to the model under consideration. The key element of the proof relies on Lorden's inequality, which is a the significant result from the renewal theory.

Keywords: retrial model, classical retrial policy, stability analysis, renewal theory, Lorden's inequality, regenerative approach

1. Introduction

The paper deals with a multi-class retrial queue under classical retrial policy. The blocked arrival joins the corresponding orbit and then after generally distributed retrial time attacks the server again.

Retrial queuing systems are successfully applied in simulation of multiple access systems like call centers [1], telephone networks [2], cellular mobile networks, etc. Most of stability results were obtained for more wide-spread retrial models with exponential retrials, see for instance [3, 4]. The analysis of more general case with an arbitrary distribution of retrial times is a challenging problem.

Our goal in this paper is to establish sufficient stability conditions for a single server model under consideration. Namely, we expand the previous analysis, obtained for particular cases of multi-class retrial models [5, 6]. The research is based on regenerative approach. We present just the main steps of the proof, which are focused

The publication has been prepared with the support of Russian Science Foundation according to the research project No.21-71-10135 https://rscf.ru/en/project/21-71-10135/.

on the application for general retrials and rely on the results from renewal theory, namely, Lorden's inequality.

The paper is organized as follows. Section 2 contains a detailed description of multi-class retrial system. Then, in a basic section 3, we obtain sufficient stability conditions for the presented model. Section 4 concludes the talk.

2. Description of the model

We define a multi-class retrial queue with a single server. Incoming customers arrive at instants $\{t_n,\, n \geq 1\}$ according to the renewal input with a generic interarrival time τ . The system admits $K \geq 1$ classes of customers. Class-i customer, where $i = 1, \ldots, K$ arrives with a rate $p_i\lambda$. Note, $\lambda = 1/\mathsf{E}\tau$ is a total input rate and the probability p_i is defined from a given distribution $p = (p_1, \ldots, p_K)$. Service times are independent and stochastically equivalent to $S^{(i)}$, where i defines the class number. Thus we obtain the following load coefficient

$$\rho = \lambda(p_1 S^{(1)} + \cdots + p_K S^{(K)}.$$

Class-i arrival, who meets the server busy, joins to the corresponding infinite capacity buffer so-called orbit and then after a random time, distributed as $\xi^{(i)}$, makes attempts to capture the server again. Next we make the basic assumptions: the model obeys to classical retrial policy, thus the total rate from all K orbits is proportional to the number of orbit customers; and retrial times $\xi^{(i)}$ are generally distributed, unlike more wide-spread case with exponential retrials.

3. Stability analysis

In this section we present the sufficient stability conditions for the system under consideration. Note, that the proof is based on the regenerative approach [7, 8], thus under the term "stabilit" we actually mean positive recurrence of the basic regenerative processes, related the system. Namely, for the stability, it is enough to show that the process X(t), associated with the total number of customers in the system at instant t, is positive recurrent.

Theorem 1. Consider a single-server K-class retrial queuing system with zero initial state and assume

$$\rho < 1, \tag{1}$$

$$\mathsf{P}(\tau > x) > 0, \quad \text{for all } x \geq 0, \tag{2}$$

$$\mathsf{E}\big(\xi^{(i)}\big)^2 < \infty, \quad i = 1, \ldots, K. \tag{3}$$

Then the system is stable.

Proof. Note that new results are focused on the application for general retrials case. Thus we give details of the proof, related to the general distribution of $\xi^{(i)}$, as the rest steps follow the analysis for the particular case of exponential retrials, presented in [5].

Denote by Δ_n the sum idle period in $[t_n, t_{n+1})$. The condition $\rho < 1$ implies $\mathsf{E}\Delta_n \not\to 0$, see [6].

Next define the summary orbit size just before t_n and the total number of departures in $[t_n, t_{n+1})$ by N_n and D_n, respectively. Then for some arbitrary constants d, $d_0 > 0$ we present mean idle period as follows

$$\begin{aligned} \mathsf{E}\Delta_n &= \mathsf{E}[\Delta_n,\, N_n \le d + d_0] + \mathsf{E}[\Delta_n,\, N_n > d + d_0,\, D_n > d_0] && (4)\\ &+ \mathsf{E}[\Delta_n,\, N_n > d + d_0,\, D_n \le d_0]. && (5) \end{aligned}$$

From [5, 6] and independently on distribution of $\xi^{(i)}$ we obtain the upper bounds for the first summands in (4) as follows

$$\begin{aligned} \mathsf{E}[\Delta_n,\, N_n \le d + d_0] &\le \mathsf{E}\tau\mathsf{P}(N_n < d + d_0), && (6)\\ \mathsf{E}[\Delta_n,\, N_n > d + d_0,\, D_n > d_0] &\le a\mathsf{P}(M(a) > d_0) + \mathsf{E}[\tau,\, \tau > a], && (7) \end{aligned}$$

where $a > 0$ is an arbitrary constant and $\{M(t)\}$ defines a zero-delayed renewal process, built on intervals, stochastically equivalent to $\min_{i=1,\dots,K} S^{(i)}$.

Our goal is to obtain the upper bound of (5). Namely, we explore a mean idle period Δ_n in case the number of departures D_n is not greater, than d_0 and the summary orbit just before the instant t_n is lower bounded by $d + d_0$. Thus up to the next arrival the summary orbit contains at least d customers: $N_{n+1} > d$. That means, the retrial attempts at least of d orbit customers are unsuccessful at τ_n. Define the set of numbers for such customers by $\mathcal{C}$. Then denote by $A_1 < A_2 < \cdots < A_d \le t_n$ the arrival instants of customers from $\mathcal{C}$.

Consider $c_0 = \max(1, \lfloor d/2 \rfloor)$ and $c_1 = d - c_0$ and divide $\mathcal{C}$ for two sets: $\mathcal{C}_0$ the numbers of customers, arrived at instants $A_1, \dots, A_{c_0}$, and $\mathcal{C}_1$ the numbers of customers, arrived at instants $A_{c_0+1}, \dots, A_d$.

Next, denote by t_n^* the first *departure* instant after t_n. (Note that if $t_n^* > t_{n+1}$, then $\Delta_n = 0$ with probability 1.) Thus assume $t_n^* < t_{n+1}$. Then define by $\mathcal{T}(t_n^*)$ an interval since t_n^* up to the next retrial. Namely, $\mathcal{T}(t_n^*)$ coincides with the first idle period in $[t_n, t_{n+1})$, note $t_n^* + \mathcal{T}(t_n^*) < t_{n+1}$.

Define by $\mathcal{T}_{c_0}(t_n^*)$ the remaining retrial time for the costumers from the set $\mathcal{C}_0$. Recall $N_{n+1} > d$ and the assumption that $\mathcal{C}_0$ contains only customers, that would not capture the server before t_{n+1}. Thus

$$\begin{aligned} t_n^* + \mathcal{T}_{c_0}(t_n^*) &\ge t_{n+1}, \quad \text{or}\\ t_n^* + \mathcal{T}_{c_0}(t_n^*) &< t_{n+1} \text{ and the server is busy at instant } t_n^* + \mathcal{T}_{c_0}(t_n^*). \end{aligned}$$

Hence

$$\mathcal{T}(t_n^*) \le \mathcal{T}_d(t_n^*). \tag{8}$$

The relation of remaining retrial times for successful and unsuccessful attempts for the case $t_n^* + \mathcal{T}_{c_0}(t_n^*) < t_{n+1}$ is illustrated on figure 1. Note that the server is busy at instant $t_n^* + \mathcal{T}_{c_0}(t_n^*)$, while there could be idle intervals in $\Big(t_n^* + \mathcal{T}(t_n^*),\ t_n^* + \mathcal{T}_{c_0}(t_n^*)\Big)$.

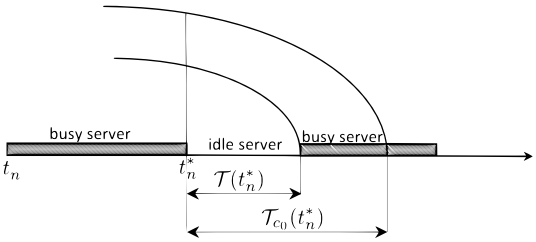


Fig. 1. Remaining retrial times

Then for $t \ge A_{c_0+1}$ we construct a set of *renewal processes* $\Lambda_j(t)$, $j = 1, \dots, c_0$, associated with the number of unsuccessful attempts for the j-th orbit customer from $\mathcal{C}_0$. Note that all the customers in $\mathcal{C}_0$ had already been in the system before the moment A_{c_0+1}. Consider the j-th orbit customer belongs to class $i_j \in \{1, \dots, K\}$, thus inter-renewal times of a process $\Lambda_j(t)$ are stochastically equivalent to $\xi^{(i_j)}$. Next construct $\mathbf{B}_j(t_n^*)$ – remaining time from t_n^* up to the next renewal in a process $\Lambda_j(t)$ (the next after the instant t_n^* attempt of the j-th orbit customer from the set $\mathcal{C}_0$). Namely, $\mathbf{B}_j(t_n^*)$ coincides with a remaining retrial time of the corresponding class $i_j \in \{1, \dots, K\}$. Then

$$\mathsf{P}\big(\mathcal{T}(t_n^*) > x\big) \le \mathsf{P}\big(\mathcal{T}_{c_0}(t_n^*) > x\big) = \Pi_{j=1}^{c_0} \mathsf{P}(\mathbf{B}_j(t_n^*) > x) \le \big(\max_j \mathsf{P}(\mathbf{B}_j(t_n^*) > x)\big)^{c_0}.$$

Consider for simplicity $\beta(t_n^*) := \mathbf{B}_j(t_n^*) :\ \mathsf{P}(\beta(t_n^*) > x) = \max_{j=1,\dots,c_0} \mathsf{P}(\mathbf{B}_j(t_n^*) > x)$ and define by ξ the generic renewal time of the j-th renewal process, which corresponds to the maximal value of $\mathsf{P}(\mathbf{B}_j(t_n^*) > x)$. Note that ξ is stochastically equivalent to the retrial time $\xi^{(i_j)}$ of corresponding class $i_j \in \{1, \dots, K\}$, and the number i_j depends on t_n^*. The instant t_n^* is random with distribution $F_{t_n^*}$ and depends on the service time, while $\mathcal{T}(t_n^*)$ depends on the number of orbit customers. Thus the mean for the first idle period is defined as follows

$$\mathsf{E}\mathcal{T}(t_n^*) = \int_{u \in \tau_n} \mathsf{E}\mathcal{T}(u) dF_{t_n^*}(u). \tag{9}$$

Next from (8) for all deterministic $u \in \tau_n$

$$\mathsf{E}\mathcal{T}(u) \ = \ \int_0^\infty \mathsf{P}\big(\mathcal{T}(u) > x\big) dx \le \int_0^\infty \big(\mathsf{P}(\beta(u) > x)\big)^{c_0} dx. \tag{10}$$

Because β defines remaining renewal time for a corresponding renewal process, then by **Lorden's inequality** (see [7]) :

$$\int_0^\infty \mathsf{P}(\beta(u) > x)dx \equiv \mathsf{E}\beta(u) \leq \frac{\mathsf{E}\xi^2}{\mathsf{E}\xi}.$$

By condition of the theorem $\mathsf{E}\left[\xi^{(i)}\right]^2 < \infty$ for all $i = 1, \ldots, K$, then $\mathsf{E}\xi^2/\mathsf{E}\xi < \infty$. Thus $\mathsf{P}(\beta(u) > x)$ is integrable with respect to x. Hence $\Big(\mathsf{P}(\beta(u) > x)\Big)^{c_0}$ is dominated by integrable function and we can apply dominance convergence (Lebesgue) as follows:

$$\lim_{d\to\infty} \mathsf{E}\mathcal{T}(u) \leq \lim_{d\to\infty} \int_0^\infty \Big(\mathsf{P}(\beta(u) > x)\Big)^{\lfloor d/2 \rfloor} dx = 0.$$

Taking into account (9), we obtain

$$\mathsf{E}\mathcal{T}(t_n^*) \to 0, \quad d \to \infty. \tag{11}$$

Note that on the event $\{N_n > d + d_0,\, D_n \leq d_0\}$ the system admits not more than $(d_0 + 1)$ idle periods in τ_n, while the orbit is not less than d. By the same arguments, as in (11), we can obtain that each mean idle period in τ_n goes to zero with a growth of d. Define by $\mathbf{T}_n(d)$ the longest mean idle period in τ_n. Thus

$$\mathsf{E}[\Delta_n,\, N_n > d + d_0,\, D_n \leq d_0] \leq (d_0 + 1)\mathbf{T}_n(d). \tag{12}$$

Next, taking into account bounds (6), (7) and (12), for all $\varepsilon > 0$ we chose appropriate values of the constants $a = a(\varepsilon)$, $d_0 = d_0(a)$, $d = d(d_0)$ such, that

$$\mathsf{E}[\tau,\, \tau > a] + a\mathsf{P}(M(a) > d_0) + (d_0 + 1)\mathbf{T}_n(d) < \varepsilon/2.$$

Now assume that the summary orbit infinitely grows: $N_n \Rightarrow \infty$, as $n \to \infty$. The assumption implies that there exist such a number n_1, that

$$\mathsf{E}\tau\mathsf{P}(N_n < d + d_0) \leq \varepsilon/2$$

for all $n \geq n_1$. Thus, we obtain that $\mathsf{E}\Delta_n < \varepsilon$, $n \geq n_1$, which leads to the contradiction. Hence, the orbit is tight: $N_n \not\Rightarrow \infty$. Next, using the condition $\mathsf{P}(\tau > x) > 0$ and regenerative approach, exactly as in [6], we are able to show that the system under consideration is stable (positive recurrent). Note that the demand of zero initial state is used in regenerative method. ■

4. Conclusion

We considered a single server multi-class retrial queue under classical retrial policy. Relying on previous analysis for particular distributions of retrial times $\xi^{(i)}$, we obtained that the condition $\rho < 1$ indeed guarantees the stability, at least if inter-arrival times are unbounded and additional moment properties of $\xi^{(i)}$ holds true.

REFERENCES

1. Aguir S. et al. The impact of retrials on call center performance// OR Spectrum. 2004. V. 26 P. 353–376.
2. Tran-Gia P. and Mandjes M. Modeling of customer retrial phenomenon in cellular mobile networks // IEEE Journal on Selected Areas in Communications. 1997. V. 15. P. 1406–1414.
3. Artalejo J.R. and Phung-Duc T. Single server retrial queues with two way communication // Applied Mathematical Modelling. 2013. V. 37. P. 1811–1822.
4. Sakurai H. and Phung-Duc T. Two-way communication retrial queues with types of outgoing calls// TOP. 2015. V. 23. P. 466–492.
5. Morozov E., Phung-Duc., T. Stability analysis of a multiclass retrial system with classical retrial policy// Performance Evaluation. 2017. V. 112. P. 15–26.
6. Morozov E., Nekrasova R. Stability Conditions of a Multiclass System with NBU Retrials// Lecture Notes in Computer Science. 2019. V. 11688. P. 51–63.
7. Asmussen S. Applied probability and Queues. 2nd edt. Springer-Verlag, New York, 2003.
8. Morozov E., Delgado R. Stability analysis of regenerative queues// Automation and Remote control. 2009. V. 70. P. 1977–1991.

UDC: 681.51

Autonomous Infrared Guided UAV Precision Landing System

M. Mondal[1], S.V Shidlovskiy[1], D.V. Shashev[1], M.V. Okunsky[1]

[1]National Research Tomsk State University, Tomsk, Russia 634050

mainakme2140@gmail.com, shidlovskiysv@mail.ru, dshashev@mail.ru, iamleftbrain@gmail.com

Abstract

This article highlights the requirements for precision landing systems in multi rotors and proposes a 3 point IR-guided Landings system for a standard portable landing pad. The proposed system uses a monocular camera with an IR Filter and Open CV. The geometrical Centroid formula is used to anchor itself over the landing pad.

Keywords: Multi-rotor, IR Landing, Guided Landing, Autonomous Navigation

1. Introduction

Today most off-the-shelf consumer multi-rotors are equipped with features like autonomous flight, GPS way-point-mission, optical-flow stabilization and a lot more. An important aspect of autonomous flight is GPS or Global Positioning System. It uses GNSS to triangulate a position approximate to 5 meters. Using this, UAVs have achieved great feats in the past decade by using this to navigate faraway territories without human intervention. These features enable multi-rotors to navigate through the skies with ease, and in recent years the industry of multi-rotors has grown due to the consumer interest in these devices. Manual control of these multi-rotors are as safe as the pilots but autonomous fights depend on the flight controller as well as the on-board computer on the UAV itself. Since it is already established that GPS is approximately accurate to 5 meters[1] and which is in the best of conditions, landing autonomously in tight or dangerously small spots is fairly risky. Camera Assisted landing makes autonomous flights in these UAVs more reliable. The main objective of this research is to use a monocular camera(with IR Lens) in a UAV to detect visual cues(IR Beacons), and use it as an anchor to align itself while landing at the spot, to avoid unnecessary hitches or movement due to wind or other susceptible causes like COG.

1.1. Design of a Quad-copter. A Multi rotor is made up of multiple thrust-generating engines. A quad-copter, as the name suggests has four rotors or 4 thrust-generating engines. These 4 rotors have to be places such that 2 rotors spin clockwise and 2 rotors spin anti-clockwise. There is flight controller in the quad-copter, usually near the center of gravity, which has an IMU or Inertial Measurement Unit, which measures linear acceleration and the angular velocity. This data can be processed by the controller to produce, the pitch and roll rates/angles and this data can be used by the controller to command the 4 rotors accordingly to stabilize the UAV [2]. The yaw rate is also measured by it, but its not very accurate, and thus a magnetometer is used and to physically yaw the reactive torque generated by the motors is used. A general 'x' quad-copter as seen in figure 1 usually has the following equations for distributing thrust to the motors.

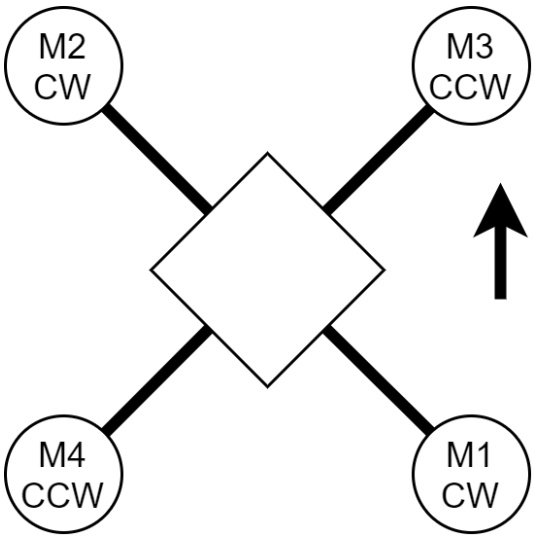


Fig. 1. X Quad-copter

$$M1 = Thrust - Roll + Pitch - Yaw \tag{1}$$

$$M2 = Thrust + Roll - Pitch - Yaw \tag{2}$$

$$M3 = Thrust - Roll - Pitch + Yaw \tag{3}$$

$$M4 = Thrust + Roll + Pitch + Yaw \tag{4}$$

1.2. Control of a Quad-copter. The control system of the Quad-copter can be seen in figure 2[3] where various sensors are used to estimate the local position of the drone in a local/global space. The equations (1),(2),(3) and (4) make up the MMA or Motor Mixer Algorithm. MMA or Motor Mixer Algorithm, which translates the inputs received (pitch, roll, yaw, thrust) to the values understandable by the Electronic Speed Controllers concerning the frame of reference of the quadcopter.

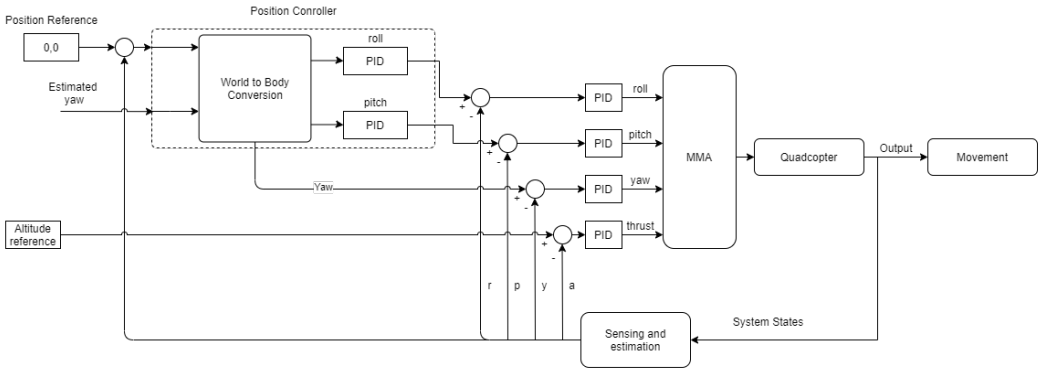


Fig. 2. Control System of a UAV

2. Concept

Assisted landing has existed for a while in open-source flight controllers like the pixhawk, using an infrared emitter beacon and receiver. The controller is connected to an IR-LOCK sensor which is a slightly modified camera to detect IR light. The sensor spits out the position of the detected IR light and the controller uses this data to align itself and land at approximately 1 m/s. This mode of landing can be difficult to use on sunny days as the IR Sensor might recognize the sunlight as a landing beacon. In the tests conducted, on a fairly sunny day, it was observed that 3 out of 4 times the sensor mistook the spot formed by crepuscular rays under a tree as a landing beacon.One can suggest the use of specialized colors and landing markers to use for this process but it might also be mistaken by the camera in many situations.

In usual cases, a brightly colored landing pad can be used by the UAV as the landing area. Recognizing a bright color like orange is simple using Open CV and it works well because there is not a lot of computation involved. This method however fails to work in a dimly lit day and completely fails during the night when it's dark. Keeping this scenario in mind, High Intensity Infrared Beacons were chosen because they are visible from far away distances and are visible in the daylight as well as in the dark. (Note : Visible to a camera with the IR Lens). This method improves on

the "Assisted landing" and uses 3 Infrared Beacons around the landing area and a special IR Lens/Filter to let only the IR Light pass.

It is theorized that using a monocular camera(with a IR filter) and three infrared beacons as a marker will provide ample guidance for a multirotor to land at a designated location.

3. Placement of the High Intensity IR Beacons

The 3 High Intensity IR beacons should be placed as marked in Figure 3. The IR Beacons should be placed such that the center of the triangle (formed by them) and the center of the landing pad fall over each other.

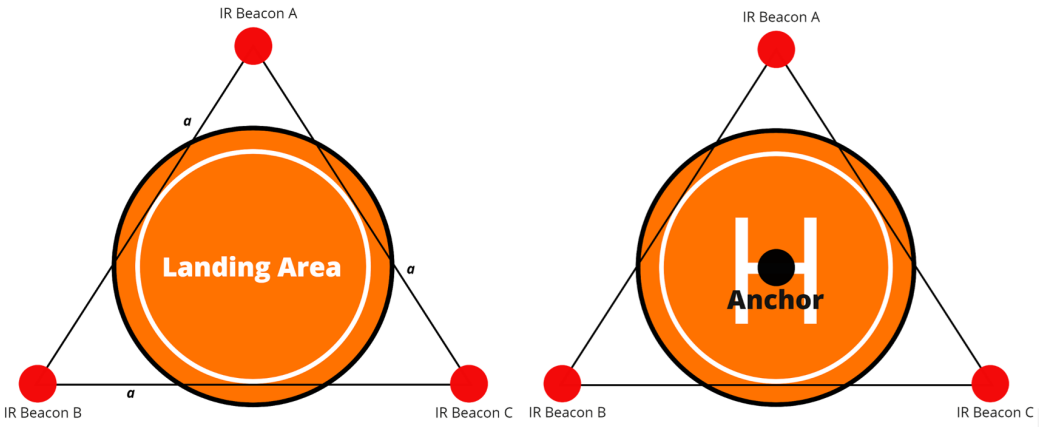


Fig. 3. Landing Pad with beacons

3.1. Calculation : To find the distance between the IR Beacons.. *Let's assume the Landing Pad to be 1.5 meters in diameter, and the landing area(white circle) has a diameter of 1 meter. The triangle formed around the landing area is an Equilateral Triangle, i.e. the length of sides (a) are equal to each other and the angle formed in each vertex is 60 degrees.*

$$r = (\sqrt{3}a)/6 \tag{5}$$

or,

$$a = 2\sqrt{3}r \tag{6}$$

In (6), r is the radius of the inscribed circle or 0.5 meters. Thus, the length of each side is **1.73 meters**. Thus, the IR Beacons should be at least 1.73 meters apart from each other.

4. Anchor Point Calculation

The Anchor Point is shown in figure 3. It is the center of the Landing Pad and it marks the in-center of the triangle formed by the High Intensity IR Beacons.

The anchor is the centroid of the triangle.Given the coordinates of the three vertices of the triangle ABC, the coordinates of the Anchor(O), are given by the equations - (7) and (8).

$$O_x = (A_x + B_x + C_x)/3 \tag{7}$$

$$O_y = (A_y + B_y + C_y)/3 \tag{8}$$

Where, Ax and Ay are the x and y coordinates of the point A(IR Beacon). Bx and By are the x and y coordinates of the point B(IR Beacon). Cx and Cy are the x and y coordinates of the point C(IR Beacon).

5. Algorithm

Figure 4 outlines the process of assisted autonomous landing. This landing sequence can be initiated in the air, which will wait for the UAV to navigate to the final waypoint in its mission parameters. Once the final waypoint is reached, it will look for the IR beacons in the ground using the monocular camera (with IR Lens) pointed downwards.

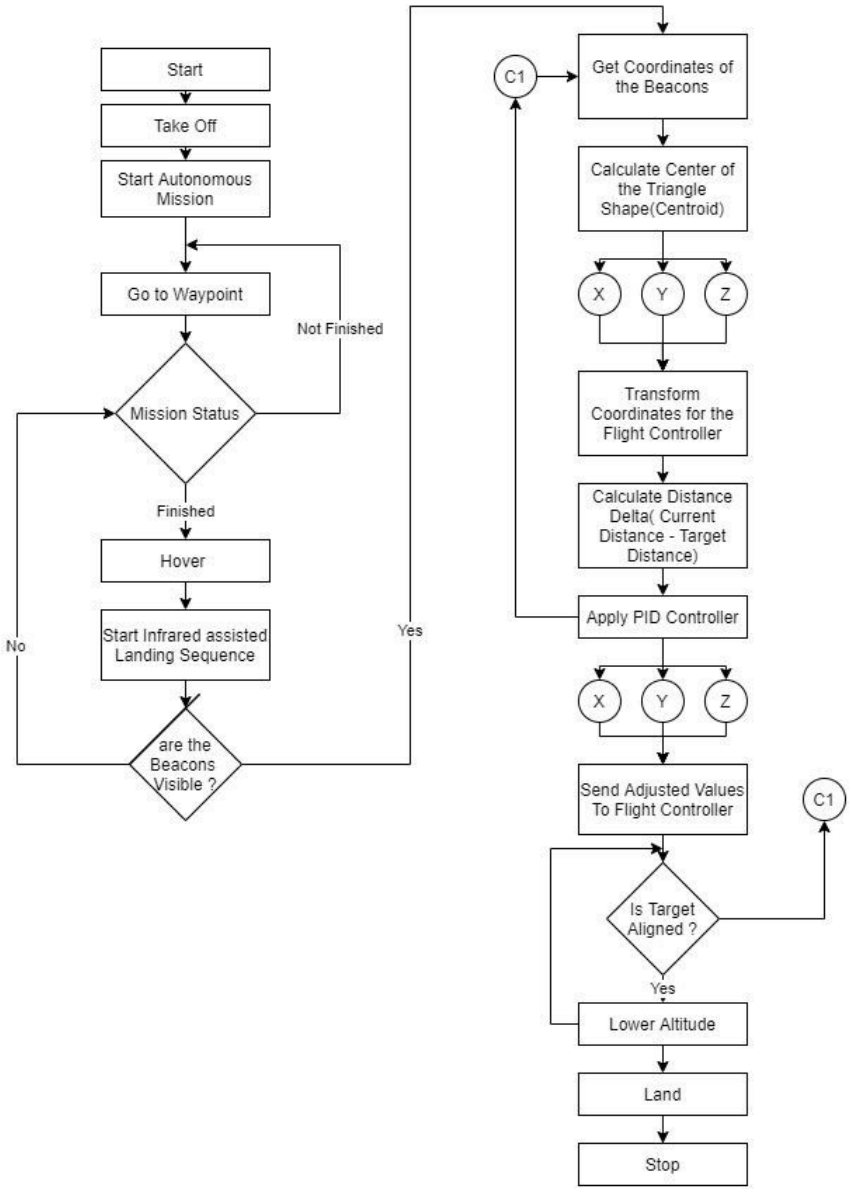


Fig. 4. Flowchart

6. Results

The output of the algorithm should look like figure 5 from about 10 meters altitude. Once the Beacons are located, the OpenCV will broadcast(publish) the detected location in the ROS network. This data is read by another subscriber which does the necessary Centroid Calculations and transformations required and publishes delta values to the flight controller.

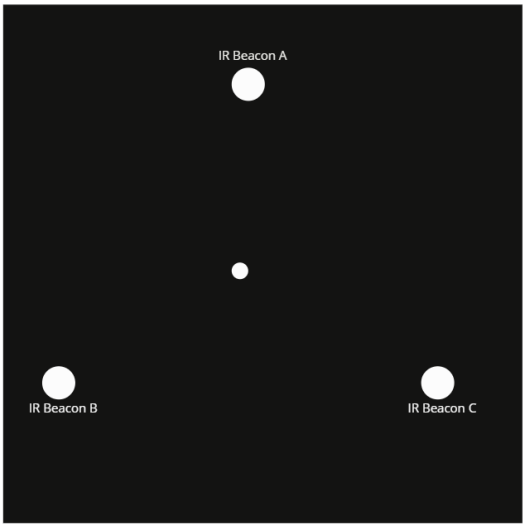


Fig. 5. Flowchart

A PID controller is also applied for accurate reaction to the changes in the position of the UAV and to account for external factors. The delta distance values will be sent to the flight controller, so the UAV aligns itself over the marker and lowers altitude, while locked on the marker. Eventually, the altitude reaches 0 and the UAV will land.

Similar research presented in [4] and [5] can be seen in table 1 with excellent results.

Method	Wind Speed	Gusts	Number of Trials	Average Accuracy
GPS Mission	4 m/s	8 m/s	15	3.9 meters
Precision Landing[4]	3 m/s	5 m/s	17	0.44 meters
Apriltag Landing[5]	3 m/s	4 m/s	15	0.15 meter
IR Landing	4 m/s	5 m/s	15	0.19 meter

Table 1. Comparison of the Landing Methods

Remark 1. The wind speed and wind gusts have been recorded from windy[6]

The IR Landing method presented in this article is very robust and reliable as seen in the table 1. It is worth noting that the precision landing presented by the authors of [4] is at a disadvantage as it was performed in an uneven surface and it is safe to assume that the method presented might also be inconsistent when the conditions are less than perfect.

In comparison to the Apriltag Landing, this method was less accurate by 0.04 meters but that may be due to the margin of error in experiments conducted.

7. Conclusion

Today, multirotors are not only expected to perform military tasks like payload delivery and remote recon but are also expected to fulfill general civilian needs like simple point to point delivery. Tech giants like DHL and Amazon have invested a huge chunk of money into unmanned delivery systems. Many cities have already allowed these companies to test their platform and adding a precision landing system like the one mentioned in this article, will significantly make these multirotors safer to use in a populated area. A precision landing system will also build confidence with the local aviation authorities as the Landing Areas(landing pad + ir sensors) can be marked as the only designated places where a multirotor could land/ take off.

REFERENCES

1. Van D., Frank E.: The World's first GPS MOOC and Worldwide Laboratory using Smartphones. In: Proceedings of the 28th International Technical Meeting of the Satel-lite Division of the Institute of Navigation, Tampa, Florida. pages. 361-369 (2015).
2. B. Siciliano, L. Sciavicco, L. Villani, G. Oriolo. Robotics. McGraw-Hill.
3. Mainak Mondal, Sergey Poslavskiy, Offline Navigation (Homing) of Aerial Vehicles (Quadcopters) in GPS Denied Environments , Unmanned Systems , 2020
4. Kevin P., Sebastian S.: Precision UAV Landing in Unstructured Environments. In: Pro-ceedings of International Symposium on Experimental Robotics, Argentina (2018)
5. Mainak Mondal, Stanislav Shidlovskiy and Dmitriy Shashev - Camera Assisted Autonomous UAV Landing. In: CEUR Workshop Proceedings Vol - 2744, ITMO University, Russia(2020).
6. Windy Homepage, `http://windy.com`

UDC: 519.21

On polynomial convergence rate for reliability system with warm standby

G.A. Zverkina[1]

[1]V. A. Trapeznikov Institute of Control Sciences of Russian Academy of Sciences, 65 Profsoyuznaya street, Moscow 117997, Russia

Abstract

We are considering a reliability system consisting of two restorable elements. The behaviour (intensity of work or repair) of the elements depends on each other. Switching between operating mode and repair mode and vice versa may not occur instantly. The time of such switching is random but limited.

Obviously, in reliability theory, all random times have absolutely continuous distribution. Here, they can be *mixed*.

The random time of work and repair of elements is determined using intensities (or hazard rate function). These intensities depend on the *full state of the system*, i.e. on the state (work, repair) of each element and on the time of the stay in this state.

In the general case, the random process describing the behaviour of such a system is not regenerative.

We have identified the conditions under which this process is ergodic. Also, the conditions under which the upper polynomial bound for the convergence rate of this process distribution to the limit distribution can be calculated are defined.

Keywords: convergence rate, generalized Lorden's inequality, strong upper bounds, dependent alternating processes, reliability theory

1. Introduction

The reliability system consisting of two dependent restorable elements is studied in this paper. Many studies of reliability systems assume that the time for the operation of each element of the system and that the repair time of the restorable elements has an exponential distribution. It is assumed that the time for switching between operating and repair modes and vice versa occurs instantly.

Using an exponential distribution for work time has some technical rationale. However, in real reliability systems, the operating time may have a non-exponential

distribution, for example, the Erlang distribution, etc. The repair time distribution can be continuous or discrete. For example, a repair can be a simple replacement of a failed item. For example, a repair can be a simple replacement of a failed item in a fixed amount of time.

In addition, switching between the states of an element from working to being repaired and vice versa can occur in some (random) limited time.

The behaviour of both elements will be described by *intensities* of failures and repairs.

1.1. Intensities. Recall, that the intensity of the distribution of a continuous positive random variable (r.v.) ξ with distribution function (d.f.) $F(s)$ is the function $\lambda(s) \stackrel{\text{def}}{=} \dfrac{F'(s)}{1-F(s)}$, and $\mathbf{P}\{\xi \in (s, s+\Delta) | \xi > s\} = \lambda(s)\Delta + o(\Delta)$.

The function $\lambda(s)$ defines d.f. $F(s)$:

$$F(s) = 1 - \exp\left(- \int\limits_0^s \lambda(v)\, \mathrm{d}\, v\, \mathrm{d}\, u \right). \tag{1}$$

For mixed distributions, where d.f. is a sum of continuous function and step function, we put (see [6], this is not a "classical" distribution density!!!):

$$f(s) = \begin{cases} F'(s), & \text{if } F'(s) \text{ exists} \\ 0, & \text{otherwise;} \end{cases}$$

$\lambda(s) \stackrel{\text{def}}{=} \dfrac{f(s)}{1-F(s)} - \sum\limits_i \delta(s-a_i) \ln(F(a_i+0) - F(a_i-0))$, where $\{a_i\}$ is the set of discontinuity points of $F(s)$, and $\delta(s)$ is a standard δ-function;
The formula (1) is true with these notations.

So, the distribution of continuous, discrete and mixed r.v.'s can be defined by intensities.

1.2. Studied reliability system. We consider two restorable elements numbered by 1 and 2, each of which can be in working mode (denoted by "0") and in repair mode (denoted by "1"). Work and repair periods alternate.

Work and repair periods are not i.i.d.

Random variable (r.v.) $\xi_i^{(n)}$ is n-th time of the stay of i-th element in the state "0" has the distribution function $F_i^{(n)}(s)$; r.v. $\eta_i^{(n)}$ is n-th time of the stay of i-th element in the state "1" has the distribution function $\mathcal{F}_i^{(n)}(s)$. Denote $\theta_i^{(n)}(s) \stackrel{\text{def}}{=} \xi_i^{(n)} + \eta_i^{(n)}$.

Denote $\lambda_i^{(n)}(s)$ – the intensity of the failure in the n-th work period of i-th element, and $\mu_i^{(n)}(s)$ – the intensity of the repair in the n-th repair period of i-th element.

We suppose that the intensities $\lambda_i^{(n)}$ and $\mu_i^{(n)}$ are are variable and depend on the *full state of the system* under study. This *full state of the system* is described by pairs: $X_t \stackrel{\text{def}}{=} ((i_t, x_t), (j_t, y_t))$, where $i_t = 0$ or $i_t = 1$ if the first (main) element is working or not at the time t correspondingly. The value x_t is equal to the time elapsed from the last change in the state of i_t of the first (main) element to the time t. The pair (j_t, y_t) describes the state of the reserve element at the time t in the same way.

So, at the time t, the intensity of failure of i-th element is equal $\lambda_i^{(n_t)}(t) = \lambda_i^{(n_t)}(X_t)$, if i-th element works at the time t, and this is n_t-th work period. Analogously, the intensity of repair i-th element is equal $\mu_i^{(n_t)}(t) = \mu_i^{(n_t)}(X_t)$, if i-th element works at the time t is faulty.

The process X_t is Markov on the state space $\mathcal{X} \stackrel{\text{def}}{=} \{0,1\} \times \mathbf{R}_+ \times \{\mathbf{0},\mathbf{1}\} \times \mathbf{R}_+$ with the standard Borel σ-algebra. This process is not-regenerative, and the ergodicity of this process is not obvious.

Recall, $\xi_i^{(n)}$ and $\eta_i^{(n)}$ are n-th work and repair times of i-th element.

The case $\{\exists\, c > 0,\ C > 0$ such that for all t, $c \le \lambda_i^{(n)}(X_t) \le C$, $c \le \mu_i(X_t)^{(n)} \le C\}$ is studied in [6], where the ergodicity of this process is proved, also the exponential convergence rate was estimated. All distributions of the work periods and repair periods are absolutely continuous in this paper. Also, all switching between work and repair periods are instantaneous.

But in many real reliability systems, the switching can be delayed. Also, the time of operation or repair can have a discrete component - for example, a repair can consist of replacing a failed element (i.e. this time can be not-random).

Therefore, the rates of failures and repairs are assumed to be generalized functions of the process X_t in this paper.

1.2.1. Conditions

The work times $\xi_i^{(n)}$ have d.f. $F_i^{(n)}(s) \stackrel{\text{def}}{=} 1 - \exp\left(-\int\limits_0^s \lambda_i^{(n)}(u)\,\mathrm{d}\,u\right)$ which corresponds to the intensity $\lambda_i^{(n)}(s)$.

Conditions I. The following conditions for the generalized intensities $\lambda_i(s)$ are assumed:

Ia. The generalized measurable non-negative functions $\varphi(s)$ and $Q(s)$ exist such that for all $s \ge 0$, $\varphi(s) \le \lambda_i^{(n)}(s) \le Q(s)$;

Ib. $\int\limits_0^\infty \varphi(s)\,\mathrm{d}\,s = \infty$, and $\int\limits_0^\infty x^{k-1}\exp\left(-\int\limits_0^x \varphi(s)\,\mathrm{d}\,s\right)\,\mathrm{d}\,x < \infty$ for some $k \geq 2$;

Ic. $Q(s)$ is bounded in some neighbourhood of zero and for all $M > 0$, $\int\limits_0^M Q(s)\,\mathrm{d}\,s < \infty$;

Id. There exists the constant $\mathfrak{T} \geq 0$ such that $\varphi(s) > 0$ a.s. for all $s > \mathfrak{T}$. ▷

Remark 1.

1. Condition Ia holds: $G(s) = \mathbf{P}\{\zeta \leq s\} \geq F_i^{(n)}(s) = \mathbf{P}\{\xi_i^{(n)} \leq s\} \geq \Phi(s) = \mathbf{P}\{\widetilde{\zeta} \leq s\}$, or $\zeta \prec \xi_i^{(n)} \prec \widetilde{\zeta}$ are ordered in distribution. Here

$$G(s) \stackrel{\text{def}}{=\!=} 1 - \exp\left(-\int\limits_0^s Q(u)\,\mathrm{d}\,u\right); \qquad \Phi(s) \stackrel{\text{def}}{=\!=} 1 - \exp\left(-\int\limits_0^s \varphi(u)\,\mathrm{d}\,u\right). \tag{2}$$

Moreover,

$$\varphi(s)\exp\left(-\int\limits_0^s Q(u)\,\mathrm{d}\,u\right) \leq \lambda_i^{(n)}(s) \leq Q(s)\exp\left(-\int\limits_0^s \varphi(u)\,\mathrm{d}\,u\right). \tag{3}$$

2. Condition Ib holds: there exists $\mathsf{E}\,\widetilde{\zeta}^k < \infty \Rightarrow \mathsf{E}\,\zeta^k < \infty$ and $\mathsf{E}\left(\xi_i^{(n)}\right)^k < \infty$, and the support of r.v. is infinite.

3. Condition Ic holds: $\mathsf{E}\,\zeta^2 > 0$.

4. Condition Id holds: $\Phi'(x) > 0$ a.s. for $x > \mathfrak{T}$, i.e. we may consider delayed switching and the switching time $\leq \mathfrak{T}$. ▷

The repair times $\eta_i^{(n)}$ have d.f. $\mathcal{F}_i^{(n)}(s) \stackrel{\text{def}}{=\!=} 1 - \exp\left(-\int\limits_0^s \mu_i^{(n)}(u)\,\mathrm{d}\,u\right)$ which corresponds to the intensity $\mu_i^{(n)}(s)$; the distributions $\mathcal{F}_i^{(n)}(s)$ can be discreet or not-random.

Condition II. The distributions of any repair times $\eta_i^{(n)}$ also satisfy the conditions Ia, Ib and Ic, but with other generalized functions $\widehat{\varphi}(s)$ and $\widehat{Q}(s)$. In the case when its are not not-random $\left(\eta_i^{(n)} \not\equiv \mathrm{H}\right)$ we suppose that the distributions of $\eta_i^{(n)}$ are non-lattice. ▷

Remark 2. Likewise Remark 1, $0 < \mathsf{E}(\varsigma)^\ell \leq \mathsf{E}\left(\eta_i^{(n)}\right)^\ell \leq \mathsf{E}\,(\widetilde{\varsigma})^\ell$ for $\ell \leq k$, where ς has the intensity $\widehat{Q}(s)$, and $\widetilde{\varsigma}$ has the intensity $\widehat{\varphi}$. In the case $\eta_i^{(n)} \equiv \mathrm{H}$, $\widehat{Q}(s) = \delta(s - \mathrm{H})$ – δ-function. ▷

Remark 3. Conditions Id and II ensure that the distributions of the period of work and repair $\left(\xi_i^{(n)} + \eta_i^{(n)} \text{ or } \eta_{i-1}^{(n)} + \xi_i^{(n)}\right)$ are non-lattice. ▷

2. Main results

Theorem 1. *In conditions I and II are satisfied, then the process X_t is ergodic.* ▷

The proof of this Theorem 1 based on the Lemma 1 and results of [7].

Lemma 1. *If the Conditions I and Condition II are satisfied, then the distribution of the sum of any two consecutive periods $\xi_i^{(n)}$ and $\eta_i^{(n)}$ satisfies Conditions I.* ▷

Here we skip the technical proof of the Lemma 1.

Proof of the Theorem 1. Denote $\theta_i^{(n)} \stackrel{\text{def}}{=} \xi_i^{(n)} + \eta_i^{(n)}$. This is i-th cycle of work and repair of n-th element. These periods can be defined by the intensity of distribution $F_i^{(n)} * \mathcal{F}_i^{(n)}(s)$ and its intensity is dependent on the full state of the process X_t.

Consider the paired sequence of numbers $\left(t_i^{(1)}, t_i^{(2)}\right)$, where $t_0^{(n)}$ is the time of the first finish of repair of n-th element, and $t_{i+1}^{(n)} \stackrel{\text{def}}{=} t_i^{(n)} + \theta_i^{(n)}$.

Put $R^{(n)}(t) \stackrel{\text{def}}{=} \sum\limits_{i=1}^{\infty} \mathbf{1}\left(t_i^{(n)} < t\right)$, where $\mathbf{1}$ is the indicator of the event. $R^{(n)}(t)$ are Generalized Renewal Processes (GRP)

The pair $\left(R^{(1)}(t), R^{(1)}(t)\right)$ is Generalized Markov Modulated Poisson Process (GMMPP), and it is ergodic (see [7]).

The state of the process X_t is determined by the state of the process process $\left(R^{(1)}(t), R^{(1)}(t)\right)$, therefore, the process X_t is ergodic. ■

Now, consider the GRP's

$$R_\xi^{(n)}(t) \stackrel{\text{def}}{=} \sum_{m=1}^{\infty} \mathbf{1}\left(\sum_{i=1}^{m} \xi_i^{(n)} < t\right) \text{ and } R_\eta^{(n)}(t) \stackrel{\text{def}}{=} \sum_{m=1}^{\infty} \mathbf{1}\left(\sum_{i=1}^{m} \eta_i^{(n)} < t\right).$$

The backward and forward renewal times of the process $R_\xi^{(n)}(t)$ are $B_\xi^{(n)}(t) \stackrel{\text{def}}{=} t - \sum\limits_{i=1}^{N_t} \xi_i^{(n)}$ and $W_\xi^{(n)}(t) \stackrel{\text{def}}{=} t - \sum\limits_{i=1}^{N_t+1} \xi_i^{(n)}$. Analogously we define $B_\eta^{(n)}(t)$ and $W_\eta^{(n)}(t)$.

By generalized Lorden's inequality (see [2]) we have:

$$\mathsf{E}\, B_\xi^{(n)}(t) \le \mathsf{E}\,\zeta + \frac{\mathsf{E}\,\zeta}{2\mathsf{E}\left(\widetilde{\zeta}\right)^2} \stackrel{\text{def}}{=} \Upsilon (> \mathfrak{T}), \tag{4}$$

$$\mathsf{E}\, W_\eta^{(n)}(t) \le \mathsf{E}\,\varsigma + \frac{\mathsf{E}\,\varsigma}{2\mathsf{E}\,(\widetilde{\varsigma})^2} \stackrel{\text{def}}{=} \widehat{\Upsilon} \tag{5}$$

– at any Markov moment t. (In the case when $\eta_i^{(n)} \equiv \gamma = \text{const}$, $\mathsf{E}\, W_\eta^{(n)}(t) \le \gamma$.) Let us fix some number $\widetilde{\Upsilon} > \max(\Upsilon, \widehat{\Upsilon})$.

Denote $\mathcal{P}_t$ – distribution of X_t: $\mathbf{P}\{X_t \in A\} = \mathcal{P}_t(A)$ for all $A \in \mathcal{B}(\mathcal{X})$. The ergodicity of X_t implies $\mathcal{P}_t \Longrightarrow \mathcal{P}$, where $\mathcal{P}$ is an invariant probability measure.

Theorem 2. *If the Conditions I and Condition II are satisfied, then for all $\ell \le k-1$, and for all initial states of the process X_t, there exists the countable constant $K = K(\ell, X_0, C, \varphi(\cdot), Q(\cdot))$ such that for all $t \ge 0$, $\|\mathcal{P}_t - \mathcal{P}\|_{TV} \le \dfrac{K}{t^\ell}$,* where $\|\mathcal{P}_t - \mathcal{P}\|_{TV}$ is a distance in total variation metrics: $|\mathcal{P}_t - \mathcal{P}\|_{TV} \stackrel{\text{def}}{=} \sup\limits_{A \in \mathcal{B}(\mathcal{X})} |\mathcal{P}_t(A) - \mathcal{P}(A)|$. ▷

Schema of the proof. The proof of Theorem 2 is similar to the proof of the main result in [6]; it is based on the coupling method (see, e.g., [4]), and Lorden's inequality by the schema given in [8].

This fact made it possible to use the "classical" Basic Coupling Lemma (BCL) for continuous distribution (see, e.g., [3]). But in our case, we need to use the modified BCL (see [7]).

For the processes X_t and $\widehat{X}_t$ which have different initial states we will create *successful coupling* (see [1]). These processes correspond to two different reliability systems S and $\widehat{S}$.

Firstly, we wait for the time, when both elements of both systems will be repaired at least once. Let it be time $T_0 = \max\left(t_0^{(1)}, t_0^{(2)}\right)$.

At the time $t_i^{(1)}$ (after T_0), the first element of the process X_t starts to work, and the second element of the process X_t and both elements of the process $\widehat{X}_t$ can be in working or repaired state.

If some of these three elements is in working state (in the period denoted $\xi_\wr^{(\dagger)}$), then the elapsed time of work $B_\xi^{(n)}(t)$ satisfies inequality (4), and by Markov inequality, $\mathbf{P}\left\{B_\xi^{(n)}(t) < \widetilde{\Upsilon}\right\} \ge 1 - \dfrac{\Upsilon}{\widetilde{\Upsilon}} \stackrel{\text{def}}{=} p_\xi\left(\widetilde{\Upsilon}\right)$. Denote $\pi\left(\widetilde{\Upsilon}\right) \stackrel{\text{def}}{=} \mathbf{P}\left\{\xi_\wr^{(\dagger)} > \widetilde{\Upsilon}\right\}$. The value π can be estimated by the formulae (2).

If some of these three elements is in repair state, then the residual time of repair $W_\eta^{(n)}(t)$ satisfies inequality (5), and by Markov inequality,

$$\mathbf{P}\left\{W_\eta^{(n)}(t) < \widetilde{\Upsilon}\right\} \ge 1 - \frac{\widehat{\Upsilon}}{\widetilde{\Upsilon}} \stackrel{\text{def}}{=} p_\eta\left(\widetilde{\Upsilon}\right).$$

Denote also $\varkappa(\alpha,\beta,\gamma,\delta) \stackrel{\text{def}}{=} \iiiint\limits_{S(\alpha,\beta,\gamma,\delta)} \min\{f(u),f(v),f(w),f(v)\}\,\mathrm{d}\,u\,\mathrm{d}\,v\,\mathrm{d}\,w\,\mathrm{d}\,v$, where $S(\alpha,\beta,\gamma,\delta) \stackrel{\text{def}}{=} \{u>\alpha, v>\beta, w>\gamma, v>\delta\}$ – see (3).

At the time t_i, there are three cases:

1. All three elements excluding the first element of X_t, are in the working regime. With probability greater then $\left(p_\xi\left(\widetilde{\Upsilon}\right)\right)^3$ they have the elapsed work time less then $\widetilde{\Upsilon}$. Thus, we can prolong the processes X_t, $\widehat{X}_t$ by such a way that with probability greater then $\varkappa\left(0,\widetilde{\Upsilon},\widetilde{\Upsilon},\widetilde{\Upsilon}\right)$ the processes X_t and $\widehat{X}_t$ coincide at the time $t_i+\xi_{i+1}^{(1)} \stackrel{\text{def}}{=} \tau$; τ is a coupling epoch.

2. All three elements excluding the first element of X_t, are in the repair regime. With provability greater then $\left(p_\eta\left(\widetilde{\Upsilon}\right)\right)^3$ they will finish the repair periods before time $t_i+\widetilde{\Upsilon}$, with probability greater then $\left(p_\xi\left(\widetilde{\Upsilon}\right)\right)^3$ their next working periods not finish, and at this time with probability $\pi\left(\widetilde{\Upsilon}\right)$ the working period is greater then $\widetilde{\Upsilon}$. Then we can prolong the processes X_t and $\widehat{X}_t$ by such a way that with probability greater then $\varkappa\left(0,0,0,\widetilde{\Upsilon}\right)$ all elements finish their working period at the time $t_i+\xi_{i+1}^{(1)} \stackrel{\text{def}}{=} \tau$; τ is a coupling epoch.

Analogously we can consider the situations *'wwrw', 'wwwr', 'wrww', 'wrrw', 'wrwr', 'wwrr',* where the letter w marks the working state, and letter r marks the repair state; two first letters designate the elements of the process X_t, and other letters – for the process $\widehat{X}_t$.

Finally, after the time t_i we can prolong the processes X_t and $\widehat{X}_t$ by such a way, that the time $t_i+\xi_{i+1}^{(1)}$ is a coupling epoch with probability greater then $p \stackrel{\text{def}}{=} \varkappa\left(0,0,0,\widetilde{\Upsilon}\right)\min\left\{\left(p_\xi\left(\widetilde{\Upsilon}\right)\right)^3,\ldots,\left(p_\eta\left(\widetilde{\Upsilon}\right)\right)^3\pi\left(\widetilde{\Upsilon}\right)\right\}$ the time $t_i+\xi_{i+1}^{(1)}$ is a coupling epoch. The value of p can be optimized by choosing a constant $\widetilde{\Upsilon}$ and analyzing a specific system under study.

So, the coupling epoch τ is a *conditional* sum of random variables. The technology of the construction of an estimate for such sum and its integration is given in [8, 7].

Here we skip the case of not-random repair time – it is easier than the considered situation. ■

3. Conclusion

Obviously, in reliability theory, all periods have absolutely continuous distribution, or for simplicity are exponential. For example, in the paper [6], all intensities

have been separated from zero and bounded – so, the distributions are absolutely continuous, and there are the exponential moments. We give the approach for analysis of the reliability systems with delay switching, discreet or not-random repair time, and mixed distribution of the working time.

Acknowledgments

The work is supported by RFBR, project No 20-01-00575 A.

The author thanks two anonymous referees for their valuable comments. The original text was terrible.

REFERENCES

1. D. Griffeath, *A maximal coupling for Markov chains* / Zeitschrift für Wahrscheinlichkeitstheorie und Verwandte Gebiete, 1975, Volume 31, Issue 2, pp. 95–106.
2. E. Kalimulina, G. Zverkina, *On some generalization of Lorden's inequality for renewal processes* / arXiv.org. Cornell: Cornell university library, 2019. 1910.03381v1. P. 1–5
3. K. Kato, *Coupling Lemma and Its Application to The Security Analysis of Quantum Key Distribution* // Tamagawa University Quantum ICT Research Institute Bulletin Vol.4 No.1 : 23-30 (2014) P.23–30.
4. T. Lindvall, *Lectures on the coupling method.* Wiley, New York, 1992.
5. G. Lorden, *On Excess Over the Boundary*/ The Annals of Mathematical Statistics Vol. 41, No. 2, pp. 520-527, 1970.
6. G. Zverkina, *A System with Warm Standby* / Computer Networks (Proceedings of the 26th International Conference (CN 2019, Kamień Šląski, Poland). Cham: Springer, 2019. P. 387-399. DOI: https://doi.org/10.1007/978-3-030-21952-9_28
7. G. Zverkina, *Ergodicity and Polynomial Convergence Rate of Generalized Markov Modulated Poisson Processes* / Proceedings of the 23rd International Conference on Distributed Computer and Communication Networks: Control, Computation, Communications (DCCN-2020, Moscow). Cham: Springer, 2021. Vol.1337. P. 367-381. DOI: https://doi.org/10.1007/978-3-030-66242-4_29
8. G. Zverkina, *On strong bounds of rate of convergence for regenerative processes* / Communications in Computer and Information Science. 2016. 678. P. 381–393.

UDC: 004.272

Developing of models of dynamically reconfigurable neural network accelerators based on homogeneous computing environments

V. Shatravin[1], D.V. Shashev[1], S.V. Shidlovskiy[1]

[1]Tomsk State University, 36 Lenina Ave., Tomsk, Russia

shatravin@stud.tsu.ru, dshashev@mail.ru, shidlovskiysv@mail.ru

Abstract

Nowadays, machine learning algorithms are widely used in many intelligent systems. As a result, the development of high-performance, power-efficient, flexible and reliable computing devices is becoming a major challenge. This is especially important for autonomous and mobile systems that use several different machine learning algorithms at the same time. Reconfigurable hardware accelerators are one possible solution to the problem. A key feature of these accelerators is the ability to be dynamically configured by external configuration signal to implement a required at the moment neural network model. In this paper reconfigurable accelerators based on the concept of reconfigurable homogeneous computing environments are proposed. Benefits of the reconfigurability and the usage of reconfigurable environments are discussed. Possible models of a computing environment and its elements, simulation results and their comparison with a classical non-reconfigurable solution are presented.

Keywords: neural networks hardware accelerators, reconfigurable computing environments, homogeneous structures, high performance computing.

1. Introduction

In recent decades, machine learning algorithms have become widespread due to their ability to solve fuzzy, poorly formalized tasks. Some examples of such tasks are natural language processing, image recognition, classification and many others. However, a large power consumption of the algorithms limits their application in some specific areas with stringent requirements to autonomy, power consumption, performance and weight. For example: mobile robots and UAV, smartphones and wearable devices, smart IoT sensors and nodes of Edge Computing systems [1]. A common approach to the application of machine learning algorithms in these systems is using of specialized neural processors and coprocessors, and hardware accelerators

Acknowledgments: The reported study was funded by RFBR, project number 20-37-90034.

based on FPGA or ASIC [2]. Such accelerators provide high performance with low power consumption, but in most cases they are rigidly tied to a particular neural network model. This fact restricts reuse of such devices for another tasks and leads to the need to involve several computing devices to meet all needs. Moreover, sometimes it is not known in advance which models the device will need. Reconfigurable hardware accelerators can be used to solve these problems. Their key feature is the ability to be dynamically adjust to implement required neural network model.

The reconfigurability of the accelerator's structure provides additional useful features. Some of them are: support for several different operational modes, for example – low-power standby mode and high accuracy active mode; remote modification or replacing obsolete models; late configuration of remote devices when there is no prior knowledge of an operational environment; the ability to implement a lot of models with different architectures (the only limitation is the amount of device memory); running deep networks using multi-cycle processing; remote recovery of the device by redistributing computations to undamaged sections of a semiconductor.

The idea of reconfigurable hardware accelerators is not new and different approaches have been proposed [3]. This paper discusses the implementation of reconfigurable accelerators based on the concept of homogeneous computing environments.

2. Reconfigurable computing environments

The reconfigurable computing environment (RCE, homogeneous structure) is a discrete mathematical model of a computing device, consisting of many simple computing elements (CE) of the same type. Each element connected with its neighbors in the same way. Thus, RCE is a regular lattice of small computing nodes and interconnections between them (Fig. 1).

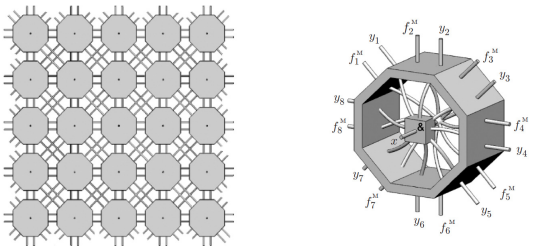


Fig. 1. Example of reconfigurable computing environment and its element

Each CE has a configuration input, by which it can be configured to perform a specific operation from some predefined set. Due to this fact, the RCE is capable to perform complex algorithms [4, 5].

The RCE concept provides useful benefits: high performance according to independent parallel operation of each cell; high flexibility and power efficiency due to low-level customization; reliability due to ability of recovery by redistribution of computations; structure homogeneity is convenient for scaling and manufacturing.

3. Development of a reconfigurable accelerator model

3.1. Determination of key features of the RCE. At the first stage key features of the designing RCE must be chosen. Generally, RCE can be of any shape and dimension. In this paper 2D RCE with square CEs are discussed. Each CE is connected with four neighboring elements.

Also, the RCE must support of deep network models by distributing its layers over several clock cycles. This can be done by dividing RCE into several segments and cyclic movement of a signal within the RCE, dynamically configuring the segments to the required layer. Since CEs are square, it will be convenient to split the RCE into four rectangular segments. Each segment implements its own layer of network and switches it to another on a clock cycle. Movement of the signal inside such RCE is shown in Fig. 2. In fact, there is usually no need to reconfigure the segment at every cycle, it is enough to use a specific one. Thus, we can implement pipelining in the RCE, when signals arriving at different clock cycles are processed in different segments at the same time. The RCE with four segments has one reconfiguration cycle (separate for each segment) and three operating cycles.

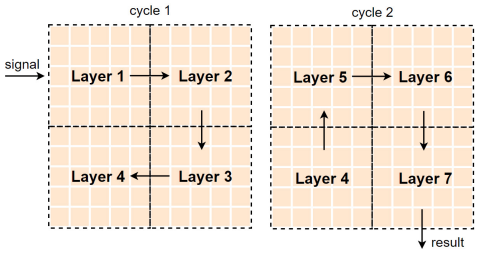


Fig. 2. Cyclic movement of a signal in a multi-cycle RCE

3.2. Designing the computing element model. The main part of developing the CE model is defining a set of supported operations. To do this, all computations in supported neural network architectures must be analyzed and decomposed into a list of elementary operations. Any operations, that are difficult to calculate on the device, must be replaced with more convenient alternatives. In this paper only a feed-forward neural network accelerator model is discussed.

Through analyzing of typical FFNN model, following list of simple operations was revealed: signal source, signal transfer, multiplication with accumulation (MAC),

ReLU activation, sigmoid activation, signal latching. To provide support of another architectures of neural networks in the RCE, it is just enough to add missing operations to the CE, without any changes in the structure of the RCE.

"Source" operation is used to set bias value of neurons. This operation can be omitted by passing the bias to the RCE via an external input. Due to complexity of computing the natural sigmoid function, the CE model uses its piecewise linear approximation proposed in [6]. The latching operation serves to hold intermediate results between segments of the RCE.

Some operations (like signal source and MAC) use an extra internal argument. For the "signal source" operation this is a bias value, for the MAC is a weight value. The argument can be set by a configuration signal or stored in the RCE's internal memory. In this paper, the argument is set by the configuration signal.

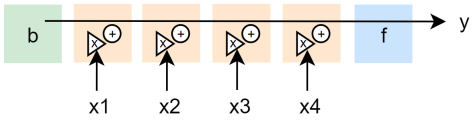


Fig. 3. Implementation of a neuron based on computational elements

To support the cycling movement of a signal within the RCE, each CE must be configured to receive a result from the correct neighbor. Thus, each CE must support four source directions. Note that the MAC operation rotates the signal 90 degrees clockwise. The configured operation only affects the main output of the CE. The other three outputs repeat signal from the opposite input. In other words, CE transfers a signal transparently in all directions, except the configured one, which applies a specified operation.

Thereby, a simple neuron with four inputs can be implemented in six CEs (Fig. 3). Due to a such decomposition, it is possible to implement neuron with any number of inputs and any activation. Example of implementation a small network on proposed RCE is shown in Fig. 4.

4. Analysis of the developed models

In this paper two key performance properties of the developed CE models are discussed: processing delay and propagation delay. Processing delay is the delay of applying an operation to an input and transferring the result in the main direction. Propagation delay is the delay of transferring a signal in any another direction (transparent transfer). All delays are in nanoseconds and reflect the worst case.

To evaluate these metrics, the proposed models were implemented with Verilog HDL in Quartus Prime software. Delays were estimated using the Timing Analyzer

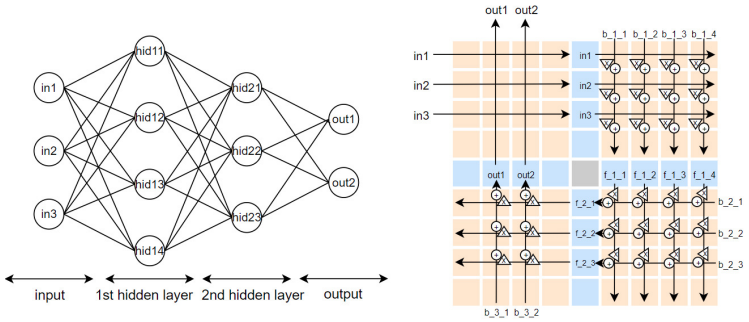


Fig. 4. Network model and its implementation on the proposed RCE

tool (part of the Quartus). The models operate with 32-bit numbers. During the simulations, the sigmoid operation was dropped because of unsupported floating-point numbers. Due to large number of inputs and outputs, all measurements were performed before the fitting step (on post-map). Timing analyzer was configured to "Fast" mode, interconnection delays were enabled. The simulation of single CE shows following results: 136 logic cells, 11.3 ns processing delay, 3.6 ns propagation delay.

Also, the proposed RCE models were compared with the classical implementation of a non-reconfigurable neuron. RCE has been tested in two modes - without a predefined configuration signal and with a constant configuration signal. The simulation results are presented in Table 1. As we can see, the RCE neurons have significantly longer delays and require much more logical cells. Such a difference is the cost of reconfigurability. Each CE contains all supported operations and reconfiguration logic. In addition, Quartus can more efficiently optimize the design of simple neurons as opposed to complex RCE elements. After pre-configuration, the parameters of RCE neuron are not that much different from the results of a simple neuron. But keep in mind, that Quartus uses powerful optimization techniques, and a pre-configured implementation may show overly optimistic results. Therefore we omit the number of LE for the pre-configured neurons.

Number of inputs	3		5		15		25	
Parameter	LE	delay, ns	LE	delay	LE	delay	LE	delay
Simple neuron	58	12.3	81	12.3	196	13	320	13.6
RCE neuron	583	31	857	43.4	2252	111	3602	175
Pre-conf. RCE neuron	–	13	–	14.2	–	20.7	–	27.1

Table 1. Experimental results

5. Conclusion and further work

Applying machine learning algorithms in mobile and autonomous systems requires high performance, flexible and reliable hardware. Dynamically reconfigurable hardware accelerators are intended for such challenges. The use of homogeneous reconfigurable environments provides useful features to the accelerators: support for many different models, late remote modification of algorithms, recovery, scaling.

Proposed in this paper models of RCE are highly flexible due to the low-level configuration of each computing element. The multi-cycle mode allows to implement deep networks and pipelining. The timing analysis of the models was performed. Comparison of these results with classical, non-reconfigurable alternatives showed that the RCE neuron has a longer delay and requires more logical cells. Such a difference is the cost of reconfigurability, because each CE contains all basic operations and handles its configuration signal. Using a dedicated reconfiguration clock cycle reduces the delays. The proposed accelerator architecture can be effectively implemented on a FPGA or ASIC.

As part of further work, we are going to improve the models to operate with floating point numbers of lower precision, optimize the design of the models, and provide support for convolutional networks.

REFERENCES

1. Carrio A., Sampedro C., Rodriguez–Ramos A., Campoy P. A Review of Deep Learning Methods and Applications for Unmanned Aerial Vehicles // Journal of Sensors. 2017. V. 2017. P. 1–13.
2. Yang Y., Wang C., Gong L., Zhou X. FPNet: Customized Convolutional Neural Network for FPGA Platforms // International Conference on Field-Programmable Technology (ICFPT). 2019. P. 399–402.
3. Kan Y., Wu M., Zhang R., Nakashima Y. A multi-grained reconfigurable accelerator for approximate computing // IEEE Computer Society Annual Symposium on VLSI (ISVLSI). 2020. P. 90–95.
4. Shatravin V., Shashev D. V. Designing high performance, power-efficient, reconfigurable compute structures for specialized applications // Journal of Physics. 2020. V. 1611.
5. Shidlovskiy S. Boolean differentiation equations applicable in reconfigurable computational medium // MATEC Web of Conferences. VII Scientific Conference “Information-Measuring Equipment and Technologies”. 2016. V. 79.
6. Faiedh H., Gafsi Z., Besbes K. Digital Hardware Implementation of Sigmoid Function and its Derivative for Artificial Neural Networks // ICM 2001 Proceedings. The 13th International Conference on Microelectronics. 2001. P 189–192.

UDC: 519.217, 519.872

Application of a queuing network with positive and negative arrivals for modeling a computer network with antivirus software

K.U. Kosarava[1] and D.Y. Kopats[1]

[1]Yanka Kupala State University of Grodno, 22 Orzheshko str., Grodno, Belarus

koluzaeva@gmail.com, dk80395@mail.ru

Abstract

In paper we investigate a queuing network with positive and negative requests, consisting of systems with control and quarantine queues. The application of such a network as a stochastic model of a computer network with antivirus is described. A system of difference-differential equations for possible states of described queueing network is derived. In case when the network is in saturation mode the expressions for the expected number of requests in the control and quarantine queues are obtained.

Keywords: queueing network, computer network, negative arrivals, control queue, quarantine

1. Introduction

Consider a computer network consisting of computers with antivirus software installed. In general, antivirus software perfom 3 basic function: detecting malicious codes in the system, removing them by destroying or isolating them, take preventive measure [1]. For each of the three types of software, its own part of the RAM(Random Access Memory) and CPU(Central Processing Unit) is allocated. Antivirus software monitors and checks the contents of the computer's RAM command blocks for viruses and if the check is successful, the file is passed to the queue for program execution. But there is a possibility that the antivirus software might not recognize the virus, for example, due to an untimely update of the antivirus signature database. We will assume that there can be 2 categories of unidentified viruses: 1) resident, which do not affect the file queue for processing and are attached to the PC, but during file processing they can infect an executable file, and 2) viruses that make it impossible to execute the file, pending processing. After "processing" the file can be transferred over the network to another computer for further processing, or transferred to the

computer's hard drive for storage and waiting for a subsequent call. Files downloaded to a computer from the network are automatically scanned by antivirus software and if file is declared infected it is quarantined on the computer [2]. Quarantine consists of 2 components: 1) storage of infected files that will not be executed by the processor and 2) software that "cures" malicious files. We will assume that 3 categories of files can be quarantined: 1) mistakenly recognized as malicious, the user has the ability to manually remove them from the quarantine; 2) files containing a virus code that, while in quarantine, can be "cured" of this code and continue execution on the user's computer; 3) viruses that cannot be neutralized and must be removed from the computer's memory. The extracted file is returned back to the location on the computer where it was extracted from and placed in RAM for loading and subsequent execution. To simulate the described computer network with an antivirus, in current work we propose to use a G-network with positive and negative requests, consisting of single-channel queueing systems (QS) with a control queue and quarantine. In this research we don't impose strict restrictions on behavior of a negative request introduced by Gelenbe [3]: after the destroying of one positive request, negative request can either leave the network or go to the quarantine queue of another system. Recently, G-networks have been widely used in a number of applications related to the simulation of attacks in computer networks (attacks on smart technologies, Denial of Service attacks) [4, 5], modelling of Intrusion Detection Systems [6], customers resets [7] and solving deep learning problems [8].

2. Model description

Let us consider a G-network with n QS. Each QS S_i has external arrivals of positive and negative arrivals which occur according to mutually independent Poisson processes, with rates λ_{0i}^{+}, λ_{0i}^{-} respectively, $i = \overline{1,n}$. For described network λ_{0i}^{+}, λ_{0i}^{-} mean the number of files that are not dangerous for the PC and files that pose a threat to it respectively, which entered the computer's RAM from its hard disk or from outside the network. The request initially received by the i-th QS enters the control queue, where it is checked for standardness, i.e. for the presence of a virus. The verification time of a request for standardness in the i-th QS has an exponential distribution with the parameter $\mu_i^{(v)}$, $i = \overline{1,n}$. After verification in the i-th QS a positive request is recognized as such with probability p_i^{+} and enters the QS for servicing and with a probability $(1-p_i^{+})$ it is recognized as negative and redirected to quarantine for treatment. A negative request after verification in the control queue of the i-th QS with probability p_i^{-} is recognized as such and redirected to the quarantine queue for treatment, and with probability $(1-p_i^{-})$ it can be mistakenly recognized as positive (for example, due to a failure to update the antivirus databases) and sent to a processing queue, where it immediately destroys the positive request. In

this paper, we investigate the model under the assumption that a negative request destroys one positive request if the QS is not empty, and leaves the system without having any impact on it, otherwise. In the physical model, this may correspond, for example, to the fact that a memory resident virus overwrites a copy of itself into a piece of computer memory, regardless of what was in that location. The files are no longer readable, and it is completely impossible to restore them using special programs. The resident copy of the virus remains active and infects newly created files. Let the service time of requets in the i-th QS has an exponential distributed function (d.f.) with the parameter μ_i, $i = \overline{1,n}$. A positive request after being served in the i-th QS can make several transitions: 1) with probability p_{ij}^{+} it goes to the j-th QS control queue as a positive request, 2) with probability p_{ij}^{-} it goes to S_j as a negative one, infected during the service with resident viruses and 3) with probability $p_{i0} = 1 - \sum_{j=1}^{n} \left(p_{ij}^{+} + p_{ij}^{-}\right)$ it leaves the network, $i, j = \overline{1,n}$.

Let us describe the behavior of a quarantine: requests recognized as non-standard are placed in the quarantine queue for treatment. Physically, the quarantine queue is a folder of files placed in quarantine by the antivirus. Suppose that the treatment time in the quarantine queue of the QS S_i has an exponential d.f. with a parameter $\mu_i^{(c)}$, $i = \overline{1,n}$. If the treatment was successful, then the request with probability $p_i^{(s)}$ is returned to the i-th QS for servicing, otherwise the request (infected file) with probability $(1 - p_i^{(s)})$ turns out to be a virus and is removed, i.e. leaves the network, $i = \overline{1,n}$. In this description of the quarantine, we assume that the virus cannot trick it during treatment.

3. Network state probabilities

The state vector of the described network has the form $\left(\overrightarrow{k}, \overrightarrow{l}, t\right) = \left(\overrightarrow{k_1}, \overrightarrow{k_2}, \ldots \overrightarrow{k_n}, \overrightarrow{l_1}, \overrightarrow{l_2}, \ldots \overrightarrow{l_n}, t\right)$, where $(\overrightarrow{k_i}, \overrightarrow{l_i}, t) = (k_i^{(p)}, k_i^{(s)}, l_i^{(n)}, l_i^{(c)}, t)$; $k_i^{(p)}$ and $l_i^{(n)}$ - the number of positive and negative requests in the control queue of QS S_i, respectively; $k_i^{(s)}$ is the number of positive requests for service in i-th queue; $l_i^{(c)}$ - the number of the requests in the quarantine, $i = \overline{1,n}$. Let requests are selected from the control queue randomly, then we estimate the probability that a positive request will be selected as $q_i^{+} = \frac{\mathbf{E}\left[k_i^{(p)}\right]}{\mathbf{E}\left[k_i^{(p)} + l_i^{(n)}\right]} = \frac{N_i^{(p)}}{N_i^{(p)} + L_i^{(n)}}$, where $N_i^{(p)}$ and $L_i^{(n)}$ are expected values of positive and negative requests in control queue of S_i respectively, $i = \overline{1,n}$.

Using the formula for the total probability of transitions between the states of the described network, it is possible to prove that the system of differential equations for a given network has the form:

$$\frac{dP\left(\overrightarrow{k},\overrightarrow{l},t\right)}{dt} = -\sum_{i=1}^{n}\left(\lambda_{0i}^{+}+\lambda_{0i}^{-}+\mu_i^{(v)}+\mu_i^{(c)}+\mu_i\right)P\left(\overrightarrow{k},\overrightarrow{l},t\right)+$$
$$+\sum_{i=1}^{n}\Big\{\lambda_{0i}^{+}u\left(k_i^{(p)}\right)P\left(\overrightarrow{k}-\widetilde{I}_{2i-1},\overrightarrow{l},t\right)+\lambda_{0i}^{-}u\left(l_i^{(n)}\right)P\left(\overrightarrow{k},\overrightarrow{l}-\widetilde{I}_{2i-1},t\right)+$$
$$+\mu_i^{(v)}q_i^{+}p_i^{+}u\left(k_i^{(s)}\right)P\left(\overrightarrow{k}+\widetilde{I}_{2i-1}-\widetilde{I}_{2i},\overrightarrow{l},t\right)+$$
$$+\mu_i^{(v)}q_i^{+}\left(1-p_i^{+}\right)u\left(l_i^{(c)}\right)P\left(\overrightarrow{k}+\widetilde{I}_{2i-1},\overrightarrow{l}-\widetilde{I}_{2i},t\right)+$$
$$+\mu_i^{(v)}\left(1-q_i^{+}\right)\left(1-p_i^{-}\right)P\left(\overrightarrow{k}+\widetilde{I}_{2i},\overrightarrow{l}+\widetilde{I}_{2i-1},t\right)+ \qquad (1)$$
$$+\mu_i^{(v)}\left(1-q_i^{+}\right)p_i^{-}u\left(l_i^{(c)}\right)P\left(\overrightarrow{k},\overrightarrow{l}+\widetilde{I}_{2i-1}-\widetilde{I}_{2i},t\right)+$$
$$+\mu_i^{(c)}p_i^{(s)}u\left(k_i^{(s)}\right)P\left(\overrightarrow{k}-\widetilde{I}_{2i},\overrightarrow{l}+\widetilde{I}_{2i},t\right)+$$
$$+\mu_i^{(c)}\left(1-p_i^{(s)}\right)P\left(\overrightarrow{k},\overrightarrow{l}+\widetilde{I}_{2i},t\right)+\mu_i p_{i0}P\left(\overrightarrow{k}+\widetilde{I}_{2i},\overrightarrow{l},t\right)+$$
$$+\sum_{j=1}^{n}\Big[\mu_i p_{ij}^{+}u\left(k_j^{(p)}\right)P\left(\overrightarrow{k}+\widetilde{I}_{2i}-\widetilde{I}_{2j-1},\overrightarrow{l},t\right)+$$
$$+\mu_i p_{ij}^{-}u\left(l_j^{(n)}\right)P\left(\overrightarrow{k}+\widetilde{I}_{2i},\overrightarrow{l}-\widetilde{I}_{2j-1},t\right)\Big]\Big\},$$

where $u(x)=1$ if $x>0$ and zero otherwise; $\widetilde{I}_r$ - $2n$-size zero vector except component with a number r which is equal to 1. The system (1) can be solved by the method of successive approximations, applying the general scheme of the method described in the paper [9]. The solution of the system (2) allows us to find any characteristics of the described network, but its solution is very laborious. Therefore, in this paper we will use the technique described in section 4.

4. Calculating expected values of positive and negative requests

Let us write the number of positive and negative requests in the control queue of the i-th QS in the form (2):

$$k_i^{(p)}(t+\Delta t)=k_i^{(p)}(t)+k_i^{(p)}(t,\Delta t),\quad l_i^{(n)}(t+\Delta t)=l_i^{(n)}(t)+l_i^{(n)}(t,\Delta t), \qquad (2)$$

where $k_i^{(p)}(t,\Delta t)$ and $l_i^{(n)}(t,\Delta t)$ - change the number of positive and negative files in the control queue of the system S_i on an interval $[t,t+\Delta t]$, $i=\overline{1,n}$. Then

$$\frac{d\mathbf{E}\left[k_i^{(p)}(t)\right]}{dt} = \lim_{\Delta t\to 0}\frac{\mathbf{E}\left[k_i^{(p)}(t,\Delta t)\right]}{\Delta t}, \quad \frac{d\mathbf{E}\left[l_i^{(n)}(t)\right]}{dt} = \lim_{\Delta t\to 0}\frac{\mathbf{E}\left[l_i^{(n)}(t,\Delta t)\right]}{\Delta t}, \tag{3}$$

where

$$\begin{aligned}\mathbf{E}\left[k_i^{(p)}(t,\Delta t)\right] &= \left(\lambda_{0i}^{+} + \sum_{j=1}^{n} p_{ji}^{+}\mu_j u\left(k_j^{(s)}\right)\right)\Delta t - \mu_i^{(v)}\frac{\mathbf{E}\left[k_i^{(p)}\right]}{\mathbf{E}\left[k_i^{(p)} + l_i^{(n)}\right]}\Delta t,\\ \mathbf{E}\left[l_i^{(n)}(t,\Delta t)\right] &= \left(\lambda_{0i}^{-} + \sum_{j=1}^{n} p_{ji}^{-}\mu_j u\left(k_j^{(s)}\right)\right)\Delta t - \mu_i^{(v)}\frac{\mathbf{E}\left[l_i^{(n)}\right]}{\mathbf{E}\left[k_i^{(p)} + l_i^{(n)}\right]}\Delta t.\end{aligned} \tag{4}$$

Suppose that the network described in the this paper operates in saturation mode, i.e. $k_i^{(p)}(t) > 0$ and $l_i^{(n)}(t) > 0$, $\forall t > 0$. Then equations (3), (4) take the form:

$$\begin{aligned}\frac{dN_i^{(p)}(t)}{dt} &= \lambda_{0i}^{+} + \sum_{j=1}^{n} p_{ji}^{+}\mu_j - \mu_i^{(v)}\frac{N_i^{(p)}(t)}{N_i^{(p)}(t) + L_i^{(n)}(t)},\\ \frac{dL_i^{(n)}(t)}{dt} &= \lambda_{0i}^{-} + \sum_{j=1}^{n} p_{ji}^{-}\mu_j - \mu_i^{(v)}\frac{L_i^{(n)}(t)}{N_i^{(p)}(t) + L_i^{(n)}(t)}, \quad i = \overline{1,n},\end{aligned} \tag{5}$$

The solution to the system of differential equations (5) is

$$\begin{aligned}N_i^{(p)}(t) &= N_i^{(p)}(0) + a_i^{+}t\frac{\alpha_i - \mu_i^{(v)}}{\alpha_i},\\ L_i^{(n)}(t) &= L_i^{(n)}(0) + a_i^{-}t\frac{\alpha_i - \mu_i^{(v)}}{\alpha_i},\end{aligned} \tag{6}$$

where $a_i^{+} = \lambda_{0i}^{+} + \sum_{j=1}^{n} p_{ji}^{+}\mu_j$, $a_i^{-} = \lambda_{0i}^{-} + \sum_{j=1}^{n} p_{ji}^{-}\mu_j$, $\alpha_i = \lambda_{0i}^{+} + \lambda_{0i}^{-} + \sum_{j=1}^{n}\mu_j(p_{ji}^{+} + p_{ji}^{-})$, $i = \overline{1,n}$.

Similarly as in (2) and taking into account (6) we can find the request's number in quarantine of i-th QS:

$$\begin{aligned}L_i^{(c)}(t) = L_i^{(c)}(0) + \frac{\mu_i^{(v)}\left(N_i^{(p)}(0)\left(1 - p_i^{+}\right) + L_i^{(c)}(0)p_i^{-} - w_i\Theta_i\right)}{\alpha_i - \mu_i^{(v)}}\times\\ \times\log\left(\frac{(\alpha_i - \mu_i^{(v)})t + \Theta_i}{\Theta_i}\right) - t\left(\mu_i^{(c)} - \mu_i^{(v)}w_i\right), i = \overline{1,n}.\end{aligned} \tag{7}$$

where $\Theta_i = N_i^{(p)}(0) + L_i^{(n)}(0)$, $w_i = \frac{a_i^+\left(1-p_i^+\right)+a_i^- p_i^-}{\alpha_i}$, $i = \overline{1,n}$.

5. Conclusion

In this paper we described a stochastic model of a computer network with antivirus software. A system of difference-differential equations for states probabilities of such a network is obtained. The expected number of positive and negative requests in the control queue of the network's systems, as well as the expected number of requests in the quarantine queue, have been calculated. The results obtained can be used to predict the reliability of servicing a computer network and the probabilities of failure of its systems, which will reveal the effectiveness of antivirus software. Further research in this area will be aimed at finding the expected income of the computer network with antivirus software and calculating the economic profitability of antivirus software.

REFERENCES

1. Bhaskar V. Patil1, Milind J. Joshi. Usages of Selected Antivirus Software in Different Categories of Users in selected Districts // JECET: Journal of Environmental Science, Computer Science and Engineering & Technology. 2014. V. 3. No. 2. P. 801–807.
2. The official website of Kaspersky Lab, `https://support.kaspersky.ru/KIS4Mac/16.0/en.lproj/pgs/59231.html`
3. Gelenbe E. Random neural networks with negative and positive signals and product form solution // Neural Comp. 1989. V. 1. P. 502–510.
4. Fourneau J. M. G-networks of unreliable nodes // Probability in the Engineering and Informational Sciences. 2016. V. 30 (3). P. 361–378.
5. Hanif Sohaib et al. Intrusion Detection In IoT Using Artificial Neural Networks On UNSW-15 Dataset // 2019 IEEE 16th International Conference on Smart Cities: Improving Quality of Life Using ICT & IoT and AI (HONET-ICT). 2019. P. 152–156.
6. Qureshi Ayyaz-Ul-Haq et al. A Novel Random Neural Network Based Approach for Intrusion Detection Systems // 2018 10th Computer Science and Electronic Engineering (CEEC). 2018. P. 50–55.
7. Gelenbe E. G-Networks with resets //Performance Evaluation. 2002. V. 49. P. 179–192.
8. Yonghua Yin. Deep Learning with the Random Neural Network and its Applications //arXiv. 2018. `https://arxiv.org/abs/1810.08653`
9. Matalytski M., Kopats D. Finding nonstationary probabilities of open Markov networks with multiple classes of customers and various features // Probability in the Engineering and Informational Sciences. 2021. V. 34 (1). P. 158–179.

УДК: 519.23

Ненадежная система массового обслуживания с повторными вызовами и резервным прибором

В.И. Клименок[1], А.Н. Дудин[1], О.В. Семенова[2]

[1]Факультет прикладной математики и информатики, Белорусский государственный университет, проспект Независимости, 4, Минск, Беларусь

[2]Институт проблем управления Российской академии наук„ ул. Профсоюзная, 65, Москва, Россия

klimenok@bsu.by, dudin@bsu.by, olgasmnv@gmail.com

Аннотация

Исследуется система массового обслуживания с повторными вызовами и ненадежным прибором. Во время восстановления прибора обслуживание запросов производится абсолютно надежным резервным прибором. Если в момент поступления запроса основной прибор занят, то запрос идет на орбиту, откуда делает попытки попасть на обслуживание через экспоненциально распределенные интервалы времени. Если в момент поступления первичного или повторного запроса основной прибор на ремонте, а резервный прибор занят, то запрос также идет на орбиту. Выписан инфинитезимальный генератор цепи Маркова, описывающей функционирование системы, и получено условие существования стационарного распределения, что позволяет вычислить стационарные вероятности и основные характеристики производительности системы.

Ключевые слова: ненадежная система массового обслуживания, повторные вызовы, резервный прибор, стационарное распределение

1. Введение

В последние годы активно ведутся разработки сверхскоростных и надежных средств связи – гибридных систем на базе лазерной (FSO - Free Space Optics) и радио технологий. К основным преимуществам атмосферных оптических линий связи относятся: высокая пропускная способность и качество цифровой связи, защищенность от несанкционированного доступа и скрытность, помехоустойчивость, скорость и простота развертывания FSO-сети. Наряду с основными преимуществами беспроводных оптических систем известны и их главные недостатки: зависимость доступности канала связи от погодных условий и необходимость обеспечения прямой видимости между излучателем и приемником.

Неблагоприятные погодные условия, такие как снег, туман, могут значительно снизить эффективный диапазон работы лазерных атмосферных линий связи. Поэтому для обеспечения операторских значений приходится прибегать к использованию гибридных решений. Взаимодополняющее поведение оптических и широкополосных радиоканалов позволило выдвинуть концепцию гибридных систем операторского класса, надежно функционирующих в любых погодных условиях. Из-за высокой практической потребности в гибридных системах связи в последнее время появилось значительное число исследований этого класса систем путем математического моделирования, см., например, монографию [1] и ссылки в ней.

В настоящей статье мы рассматриваем систему массового обслуживания, которая принципиально отличается от предыдущих работ по гибридным системам связи наличием повторных вызовов. Рассматриваемая система может быть применена для моделирования гибридной системы связи, где радиоволновой канал считается абсолютно надежным и заменяет FSO канал в тех случаях, когда последний прерывает функционирование вследствие неблагоприятных погодных условий.

2. Описание системы

Рассматривается система массового обслуживания с повторными вызовами, состоящая из двух обслуживающих приборов, один из которых (основной) является ненадежным, а другой (резервный) – абсолютно надежным. Последний находится в так называемом “холодном” резерве. Интерпретация: ненадежный прибор – это лазерный канал, а надежный – беспроводной канал IEEE 802.11n. Под влиянием погодных условий лазерный канал, т.е., основной прибор, может выходить из строя и сразу начинает восстанавливаться. Во время восстановления информация передается по резервному каналу, скорость передачи в котором ниже скорости передачи в основном канале. После восстановления основного канала резервный прибор отключается до следующей поломки основного прибора. Запросы поступают в систему в MAP-потоке, который описывается пространством состояний $\{0, 1, \dots, W\}$ неприводимой цепи Маркова $\nu_t, t \geq 0$, с непрерывным временем, называемой управляющим процессом MAP, и $(W+1) \times (W+1)$ матрицами D_0 и D_1. Запросы в MAP могут генерироваться только в моменты скачков управляющего процесса. Интенсивности переходов управляющего процесса ν_t, сопровождающиеся генерацией запроса, задаются матрицей D_1, а "холостые"переходы этого процесса - недиагональными элементами матрицы D_0. Матрица $D = D_0 + D_1$ является инфинитезимальным генератором цепи Маркова $\nu_t, t \geq 0$. Интенсивность λ поступления запросов определяется как $\lambda = \boldsymbol{\theta} D_1 \mathbf{e}$, где $\boldsymbol{\theta}$ – вектор стационарного распределения процесса $\nu_t, t \geq 0$, который опреде-

ляется как единственное решение системы линейных алгебраических уравнений $\boldsymbol{\theta} D(1) = \mathbf{0}$, $\boldsymbol{\theta}\mathbf{e} = 1$, где $\mathbf{e}$– вектор-столбец, состоящий из единиц, $\mathbf{0}$ - вектор-строка, состоящая из нулей.

Если в момент поступления запроса основной прибор занят, то запрос идет на орбиту – виртуальное место для нахождения таких запросов, откуда делает попытки попасть на основной прибор через случайные моменты времени, распределенные по экспоненциальному закону с параметром $\chi > 0$. Предполагается, что объем орбиты неограничен. Если в момент поступления первичного или повторного запроса основной прибор находится на ремонте, а резервный прибор свободен, то запрос занимает резервный прибор и начинает обслуживаться. Если во время обслуживания основной прибор восстанавливается, то запрос переходит на восстановившийся прибор и обслуживается заново. Если же в момент поступления первичного или повторного запроса основной прибор находится на ремонте, а резервный занят, то запрос идет на орбиту.

Поломки поступают на основной прибор в MAP-потоке, характеризующимся матрицами H_0 и H_1 порядка $(V+1)\times(V+1)$. Стационарный вектор MAP определяется как решение системы $\boldsymbol{\gamma}(H_0+H_1)=\mathbf{0}$, $\boldsymbol{\gamma}\mathbf{e}=1$. Интенсивность потока поломок равна $h=\boldsymbol{\gamma}H_1\mathbf{e}$. Если поломка поступает на занятый основной прибор, то обслуживающийся запрос переходит на резервный прибор и начинает обслуживаться заново. Если во время обслуживания основной прибор восстанавливается, то запрос переходит на восстановившийся прибор и обслуживается заново.

Время обслуживания запроса j-м прибором имеет PH распределение с неприводимым представлением $(\boldsymbol{\beta}^{(j)}, S^{(j)})$, $j=1,2$. Это означает, что процесс обслуживания на j-м приборе происходит под управлением цепи Маркова $m_t^{(j)}$, $t\geq 0$, с пространством состояний $\{1,\dots,M^{(j)},M^{(j)}+1\}$, где $M^{(j)}+1$ есть поглощающее состояние. Интенсивности переходов в поглощающее состояние задаются вектором $\boldsymbol{S}_0^{(j)} = -S^{(j)}\mathbf{e}$. Интенсивности обслуживания вычисляются как $\mu^{(j)} = -[\boldsymbol{\beta}^{(j)}(S^{(j)})^{-1}\mathbf{e}]^{-1}$, $j=1,2$.

Время ремонта имеет PH распределение с неприводимым представлением $(\boldsymbol{\tau}, T)$. Процесс ремонта происходит под управлением цепи Маркова ϑ_t, $t\geq 0$, с пространством состояний $\{1,\dots,R,R{+}1\}$, где $R{+}1$ есть поглощающее состояние. Интенсивности переходов в поглощающее состояние задаются вектором $\boldsymbol{T}_0 = -T\mathbf{e}$. Интенсивность ремонта вычисляется как $\tau = -(\boldsymbol{\tau}T^{-1}\mathbf{e})^{-1}$.

В данной работе процесс функционирования системы описывается цепью Маркова, выводится условие ее эргодичности, вычисляются стационарное распределение и вероятностные характеристики производительности системы.

3. Цепь Маркова, описывающая поведение системы

Пусть в момент времени t

• i_t – число запросов на орбите, $i_t \geq 0$,

• $n_t = 0$, если основной прибор исправен и свободен; $n_t = 1$, если основной прибор исправен и занят; $n_t = 2$, если основной прибор на ремонте, а резервный свободен; $n_t = 3$, если основной прибор на ремонте, а резервный занят;

• $m_t^{(j)}$ - состояние управляющего процесса обслуживания на j-м занятом приборе, $j = 1, 2, m_t^{(j)} = \overline{1, M^{(j)}}$;

• ϑ_t - состояние управляющего процесса ремонта, $\vartheta_t = \overline{1, R}$;

• ν_t и η_t - состояния управляющих процессов MAP потока запросов и MAP потока поломок соответственно, $\nu_t = \overline{0, W}$, $\eta_t = \overline{0, V}$.

Процесс функционировния системы описывается регулярной неприводимой цепью Маркова $\xi_t, t \geq 0$, с пространством состояний

$$X = \{(i, n, \nu, \eta),\ i \geq 0, n = 0, \nu = \overline{0, W}, \eta = \overline{0, V}\} \bigcup$$

$$\{(i, n, \nu, \eta, m^{(1)}),\ i \geq 0, n = 1, \nu = \overline{0, W}, \eta = \overline{0, V}, m^{(1)} = \overline{1, M^{(1)}}\} \bigcup$$

$$\{(i, n, \nu, \eta, \vartheta),\ i \geq 0, n = 2, \nu = \overline{0, W}, \eta = \overline{0, V}, \vartheta = \overline{1, R}\} \bigcup$$

$$\{(i, n, \nu, \eta, m^{(2)}, \vartheta),\ i \geq 0, n = 3,\ \nu = \overline{0, W}, \eta = \overline{0, V}, \vartheta = \overline{1, R}, m^{(2)} = \overline{1, M^{(2)}}\}.$$

Далее будем предполагать, что состояния цепи $\xi_t, t \geq 0$, упорядочены в лексикографическом порядке. Обозначим через $Q_{i,j}$ матрицу интенсивностей переходов цепи из состояний, соответствующих значению i первой (счетной) компоненты в состояния, соответствующие значению j этой компоненты, $i, j \geq 0$.

Лемма 1. *Инфинитезимальный генератор ЦМ ξ_t имеет следующий вид:*

$$Q = \begin{pmatrix} Q_{0,0} & Q_{0,1} & 0 & 0 & 0 & \dots \\ Q_{1,0} & Q_{1,1} & Q_{1,2} & 0 & 0 & \dots \\ 0 & Q_{2,1} & Q_{2,2} & Q_{2,3} & 0 & \dots \\ 0 & 0 & Q_{3,2} & Q_{3,3} & Q_{3,4} & \dots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \end{pmatrix},$$

где

$$Q_{i,i-1} = i\chi \begin{pmatrix} O_a & I_a \otimes \boldsymbol{\beta}^{(1)} & O & O \\ O & O_{aM^{(1)}} & O & O \\ O & O & O_{aR} & I_a \otimes \boldsymbol{\beta}^{(2)} \otimes I_R \\ O & O & O & O_{aM^{(2)}R} \end{pmatrix}, \ i \geq 1,$$

$$Q_{i,i} = \begin{pmatrix} D_0 \oplus H_0 - i\chi I & D_1 \otimes I_{\bar{V}} \otimes \boldsymbol{\beta}^{(1)} & I_{\bar{W}} \otimes H_1 \otimes \boldsymbol{\tau} & O \\ I_a \otimes \boldsymbol{S}_0^{(1)} & D_0 \oplus H_0 \oplus S^{(1)} & I_{\bar{W}} \otimes H_1 \otimes \mathbf{e}_{M^{(1)}} \otimes \boldsymbol{\tau} & O \\ I_a \otimes \boldsymbol{T}_0 & O & D_0 \oplus H \oplus T - i\chi I & D_1 \otimes I_{\bar{V}} \otimes \boldsymbol{\beta}^{(2)} \otimes I_R \\ O & I_a \otimes \boldsymbol{\beta}^{(1)} \otimes \mathbf{e}_{M^{(2)}} \otimes \boldsymbol{T}_0 & I_a \otimes \boldsymbol{S}_0^{(2)} \otimes I_R & D_0 \oplus H \oplus S^{(2)} \oplus T \end{pmatrix},$$

$$i \geq 0,$$

$$Q_{i,i+1} = \begin{pmatrix} O_a & O & O & O \\ O & D_1 \otimes I_{\bar{V}M^{(1)}} & O & O \\ O & O & O_{aR} & O \\ O & O & O & D_1 \otimes I_{\bar{V}M^{(2)}R} \end{pmatrix}, \; i \geq 1,$$

где $H = H_0 + H_1$, $\bar{W} = W + 1$, $\bar{V} = V + 1$, $a = \bar{W}\bar{V}$, $\otimes$, $\oplus$ *символы кронекерова произведения и суммы матриц соответственно.*

Следствие 1. *Цепь Маркова* ξ_t *принадлежит классу асимптотически квазителицевых цепей Маркова (АКТЦМ).*

Доказательство. Обозначим через $A^{(i)}$ матрицу, диагональные элементы которой совпадают с модулями диагональных элементов матрицы $Q_{i,i}$. Из [2] следует, что рассматриваемая цепь принадлежит классу АКТЦМ, если существуют пределы

$$Y_k = \lim_{i\to\infty} (A^{(i)})^{-1} Q_{i,i+k-1}, k = 0, 2, \; Y_1 = \lim_{i\to\infty} (A^{(i)})^{-1} Q_{i,i} + I \tag{1}$$

и матрица $Y_0 + Y_1 + Y_2$ является стохастической.

Каждую их матриц $A^{(i)}, i > 0$, представим в виде блочной диагональной матрицы $diag\{A_0^{(i)}, A_1, A_2^{(i)}, A_3\}$, где порядок блока с нижним индексом n равен порядку соответствующего диагонального блока матрицы $Q_{i,i}$. С учетом этих обозначений матрицы $Y_k, k = 0, 1, 2$, будут иметь вид

$$Y_0 = \begin{pmatrix} O_a & I_a \otimes \boldsymbol{\beta}^{(1)} & O & O \\ O & O_{aM^{(1)}} & O & O \\ O & O & O_{aR} & I_a \otimes \boldsymbol{\beta}^{(2)} \otimes I_R \\ O & O & O & O_{aM^{(2)}R} \end{pmatrix},$$

$$Y_1 =$$

$$= \begin{pmatrix} O & O & O & O \\ A_1^{-1}(I_a \otimes \boldsymbol{S}_0^{(1)}) & A_1^{-1}(D_0 \oplus H_0 \oplus S^{(1)}) + I & A_1^{-1}(I_{\bar{W}} \otimes H_1 \otimes \mathbf{e}_{M^{(1)}} \otimes \boldsymbol{\tau}) & O \\ O & O & O & O \\ O & A_3^{-1}(I_a \otimes \boldsymbol{\beta}^{(1)} \otimes \mathbf{e}_{M^{(2)}} \otimes \boldsymbol{T}_0) & A_3^{-1}(I_a \otimes \boldsymbol{S}_0^{(2)} \otimes I_R) & A_3^{-1}(D_0 \oplus H \oplus S^{(2)} \oplus T) + I \end{pmatrix},$$

$$Y_2 = \begin{pmatrix} O_a & O & O & O \\ O & A_1^{-1}(D_1 \otimes I_{\bar{V}M^{(1)}}) & O & O \\ O & O & O_{aR} & O \\ O & O & O & A_3^{-1}(D_1 \otimes I_{\bar{V}M^{(2)}R}) \end{pmatrix}.$$

Нетрудно проверить, что сумма матриц Y_k является стохастической матрицей. Таким образом, пределы (1) существуют и их сумма есть стохастическая матрица. Это значит, что цепь Маркова $\xi_t, t \geq 0$, принадлежит классу АКТЦМ.

Далее при установлении условия эргодичности и при вычислении стационарного распределения цепи Маркова $\xi_t, t \geq 0$, будем использовать результаты для АКЦМ, полученные в [2].

4. Условие эргодичности. Стационарное распределение

Теорема 1. *Цепь Маркова $\xi_t, t \geq 0$, является эргодической, если выполняется неравенство*

$$\mathbf{x}Y_0\mathbf{e} > \mathbf{x}Y_2\mathbf{e}, \tag{2}$$

где вектор $\mathbf{x}$ вычисляется как единственное решение системы линейных алгебраических уравнений

$$\mathbf{x}(Y_0 + Y_1 + Y_2) = \mathbf{0}, \mathbf{x}\mathbf{e} = 1.$$

Цепь Маркова ξ_t не является эргодической, если неравенство (2) имеет противоположный знак.

Стационарное распределение вероятностей цепи Маркова $\xi_t, t \geq 0$, вычисляется по алгоритму, разработанному в [2]. На основе этого распределения получены формулы для расчета ряда важных стационарных характеристик производительности системы.

5. Заключение

В данной работе исследовано стационарное поведение ненадежной системы массового обслуживания с повторными вызовами и резервным прибором, которая может служить как математическая модель гибридной системы связи, состоящей из ненадежного FSO канала и резервного надежного радиоканала.

ЛИТЕРАТУРА

1. Dudin A.N., Klimenok V.I., Vishnevsky V.M. The theory of queuing systems with correlated flows. Springer, 2020. ISBN 978-3-030-32072-0.
2. Klimenok V.I., Dudin A.N. Multi-dimensional asymptotically quasi-Toeplitz Markov chains and their application in queueing theory // Queueing Systems. 2006. V. 54. P. 245–259.

UDC: 004.773

Targeted massive incident notification system for a globally distributed computation network

V.V. Efimov[1]

[1]Peter the Great St. Petersburg Polytechnic University, 2vadim@inbox.ru

Abstract

A typical 'cloud' service, provided by means of a globally distributed computations network commonly has availability SLA below 100%, meaning a possibility of an incident with service outage or service degradation. With a multi-service approach and global segmentation, this incident in most cases does not happen for the whole system and all services at the same time. On the contrary, there are multiple incidents in different global regions and different services, each affecting only subset of customers. An incident notification to all customers creates a negative service provider image in terms of service availability. A targeted incident notification system is introduced in this paper, sending an incident notification to customers who are currently experiencing service outage or degradation only. An implementation of this system in RingCentral, a global 'cloud' telecommunications service provider is given.

Keywords: massive notifications, incident management, cloud computing, targeted notification, distributed system

1. Introduction

A typical globally distributed network providing a 'cloud' service is build to maximize its availability by adopting multi-service and data sharding techniques, but regardless of this effort Microsoft Office365, Teams, Exchange Online and other 'cloud' services of the IT giant have availability SLA of 99,9% [2]. RingCentral Inc., a global Unified Communications service provider, has 99,999% SLA [3]. 5 minutes of downtime per year is expected and needs to be dealt with. One of the means of reducing the customer dissatisfaction during an incident is Incident Communication process, which is part of ITIL Incident Management best practice. Incident Communication has a goal of timely, clear and accurate communication of an incident status to affected customers.

RingCentral Inc., a global 'cloud' telecommunications service provider [4] has more the 1.5 million service consumers all over the world and the customer base is

growing 20% year-to-year. In case of an incident, a broadcast message via a social network such as Twitter can be sent out to notify all customers, but it has a downside. An incident in such a globally distributed network is most commonly happening in its part and affects only a subset of service consumers due to the network sharding [1]. A broadcast incident notification message is delivered to those customers who are not experiencing any denial of service from RingCentral, building an image of an unreliable service provider.

To address this issue, a targeted incident massive notification system is introduced in this paper.

2. Targeted incident notification system

Incident Communication is part of Incident Management process, which is an IT industry best practice, described in ITIL [6], and adopted in RingCentral company [5].

Regardless of if the customer is experiencing service degradation during an incident, he may or may not be willing to receive the incident notification. All customers, willing to receive an incident notification are called *subscription base* and form the initial set S of customers to be considered for an incident notification.

After an incident is detected, one of the first steps in the Incident Management process is to identify the impact - which part of the network is having troubles. This is done based on monitoring events and network architecture in one hand and customer data allocation in the other. This forms the set I of *impacted customers*.

Intersection of the subscription base and impacted customers

$$S \cap I$$

is the list of customers to receive initial incident notification. But this type of notification is only delivered to a customer once, while incident may evolve with additional network segments impacted which brings the need of tracking the list of customers who already received an initial incident notification - this forms the set N of previously notified customers. Thus, the list of customers to receive an initial notification is following:

$$initial = (S \cap I) \setminus N$$

While service provider is working on service restoration, customers are periodically updated with the current incident status. This type of notification is called ongoing and the list of customers to receive ongoing notification is the intersection of three sets:

$$ongoing = S \cap I \cap N$$

As soon as a network segment is restored, which results in service restoration for some of previously impacted customers, they are informed accordingly with a *final* incident notification. The list of customers to receive the final notification is the intersection of subscription base and previously notified customers who are no longer impacted:

$$final = (S \cap N) \setminus I$$

Figure 1 shows different types of incident notifications on a Venn diagram with intersections of the three sets.

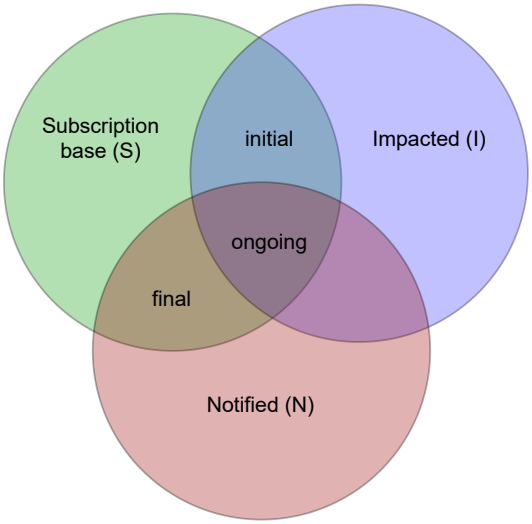


Fig. 1. Types of incident notifications

Summarizing what was described previously, every time a service provider needs to send out an incident notification, lists of customers to receive an initial, ongoing and final type of notification need to be identified.

From the Venn diagram on Figure 1 it is clear that

$$S \cap I = initial \cup ongoing$$

and

$$S \cap N = ongoing \cup final$$

A set of *ongoing* notification recipients is a subset of both S and I, which makes the algorithm more efficient in case we change the way *initial* and final are calculated:

$$initial = (S \cap I) \setminus ongoing$$

and

$$final = (S \cap N) \setminus ongoing$$

So the most optimal sequence is following:

1) Identify the list of *ongoing* notification recipients
2) Identify the list of *initial* notification recipients
3) Identify the list of *final* notification recipients

3. Targeted incident notification system architecture

Notification is executed by the Customer Notification System as shown on Figure 2. A customer opts in or out to receive an incident notification using a web user interface of this system. In order to authenticate a user, RingCentral Platform authorisation is used via oAuth 2.0 protocol. Thus, all customer subscriptions (customer base) are stored in the Customer Notification System database.

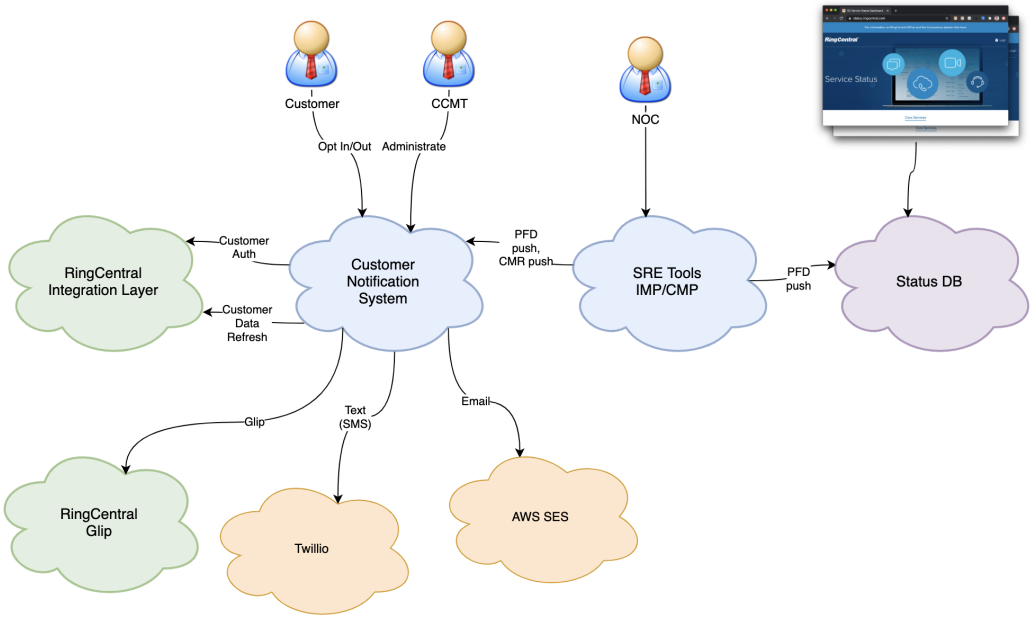


Fig. 2. Targeted incident notification system architecture

Incident record is stored in SRE Tools IMP (Incident Management Portal). It contains all required information that allows system to identify the list of impacted customers. This information includes the list of network segments that experience failure or performance degradation.

When an incident notification is triggered, IMP pushes impact information into the delivery system, which based on the customer base and previously sent notifications as described in section 2 of this paper, prepares the list of customers to receive initial, final and ongoing notifications. Then executes the delivery using vendors such as Twillio, AWS AES and RingCentral Glip.

4. Conclusion

Analysis of Incident Management process in a globally distributed telecommunications network of RingCentral company was done. Targeted Customer Incident Notification process was introduced to address the need of sending out a notification to only those customers who are currently experiencing service downtime or service degradation. Architecture diagram and notification execution flow implemented in RingCentral company was provided.

REFERENCES

1. Sikha Bagui, Loi Tang Nguyen *Database Sharding: To Provide Fault Tolerance and Scalability of Big Data on the Cloud*. International Journal of Cloud Applications and Computing (IJCAC) 5 (2015): 2, accessed (April 15, 2021), doi:10.4018/IJCAC.2015040103
2. Microsoft Online Service Level Agreement , `https://www.microsoftvolumelicensing.com/DocumentSearch.aspx?Mode=3&DocumentTypeId=37`
3. Ashu Varshney The importance of enterprise reliability and 99.999 SLAs in a work-from-home world, `https://www.ringcentral.com/us/en/blog/in-a-work-from-home-world-service-level-agreements-for-cloud-communicatio7`
4. RingCentral Cloud Services, `http://www.ringcentral.com`
5. Ardulov Y., Mescheryakov S., Shchemelinin D. *Monitoring and Remediation of Cloud Services Based on 4R Approach*. The 41st Annual International Conference by CMG, San Antonio, TX, USA, 2015
6. Axelos *ITIL Foundation, ITIL 4 edition*. UK: TSO (The Stationery Office), 2019, ISBN 978-0113316076

UDC: 004.925.5

Influence of Informational Content on Film Frame Perception

Ekaterina Borevich

Peter the Great St. Petersburg Polytechnic University, Polytechnicheskaya, 29,
195251, Saint-Petersburg, Russia

plasma5210@mail.ru

Abstract

In the paper we present the study of the influence of the film frame informational content on viewer perception. Experimental data was collected using hardware-software complex and an eye-tracker which records oculomotor activity. Methods for preparing stimulus material, a technique for conducting the experiment, and the algorithm for statistical analysis of the experimental data are described. The statistically significant influence of informational content on parameters of a viewing pattern was revealed. In addition, it was observed that gender and the demonstration factor had a statistically significant difference in the perception of stimulus material.

Keywords: Composition, visual appeal, informational content, film frame, eye-tracker, statistical data analysis, gestalt psychology, art photography

1. Introduction

Currently, cinematography is rapidly adopting advanced computer technologies, allowing for a vast control of film frame. Technologies for integrating computer graphics into video material allow to change the perception of a film frame. Computer graphics engineers replace cinema artists in the process of film frame creation. In classical painting, special attention is paid to the physiology of human perception of visual information. The rules of frame construction in cinematography are similar to rules of painting and strive for such a reflection of reality that would contribute to the deepest immersion of the viewer into the action taking place on the screen. A well-designed film frame, which is the minimum structural unit of a film, plays a huge role in the immersion process. However, there is a lack of research about a film frame design. There are several alternative approaches on the pattern of information perception [1, 2]. One of them considers this process like integrates by spatially matching information from each glimpse and summing information from successive

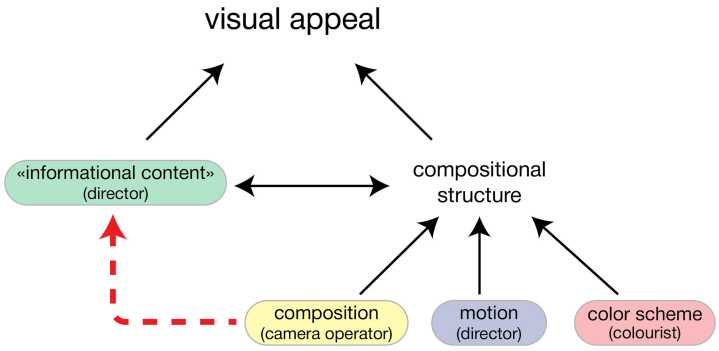


Fig. 1. Schematic of the film frame elements which contribute to the visual appeal

fixations [1]. Alternative approach that information from successive commits is not aggregated by combining snapshots of the commit but by integrating more complex visual attributes at medium to high levels of analysis [2].

The visual appeal of a film frame is determined by its elements (Fig. 1). Several experiments have been carried out to investigate the influence of one of the elements of the film frame. [3, 4, 5]. We studied correlation between the color scheme, on the perception of the visual information by an observer. For minimized the influence of other elements of the frame we excluded human's faces from a film frame. A face always carries an emotional load and is a center of interest in a film frame.

Studies show that when there is a person's face in a frame, the viewer's attention is invariably concentrated on the face, especially in close-ups. The most variable features on our faces are in the triangle of eyes, mouth, and nose [6]. Moreover, the viewer looks at person's eyes before he looks at other parts of the face. It has been experimentally proven that the examination of a human face begins precisely from the area of the eyes [7]. The visual fixation is not always limited to any specific facial feature, for example mouth attracts only a small fraction of visual attention [8]. Initial saccades on faces are driven by general stimulus properties, followed by eye movements to specific facial features which are of interest to an observer [9].

However, the main difference between artistic portrait photography and just photo is the variety of compositional decisions and the presence of individual accents and details. They create the general atmosphere of the photo frame and affect the pattern of its examination. Artistic portrait photography and cinematography both require

the mandatory presence of the main subject and the background. The presence of minor objects in the frame can affect the pattern of its examination. Dyko writes about a well-developed compositional photograph, where the subject develops in space. The photographer convincingly reveales and conveyed the spatial characteristics of the object as a whole. The main depth zones are clearly distinguished: the zone of the main object of the image, the background [10].

An important role in the portrait compositional construction portrait is played by the color, homogeneity or heterogeneity of the background. The plot also has the great role. It includes the presence of additional objects in the frame that determine the line of interaction, the theme andthe ideological concept of the photograph. However, the minor object may be present implicitly or not at all. In this case, the photo may not have a strong storyline.

2. Theoretical model

Scheme of the elements influencing the visual attractiveness was constructed on the basis of the study of the visual attractiveness of a film frame (see Fig. 1). In this article, we explore the concept of informational content of a film frame. The informative value is described, but not defined and measured [11]. Lotman speaks of the informational content of the frame, defining it as a certain conditional value (he does not define the dimension), which characterizes the degree of interest of the viewer when examining the frame for the purpose of its analysis [12]. If the frame is trivial (an obvious statement), then it is clear, obvious and easy for the viewer to interpret, that is, there is no additional (implicit) information in it that can arouse the viewer's interest for a more detailed study. The diagram of the dependence of the viewer's interest on the presence of the conditional novelty of information in the frame is shown in Fig. 2 [11].

The novelty of information is convention because is determined by the readiness of the viewer, his erudition, education, interests, intellectual development. We can talk about the informational saturation of the frame, that is, about its meaningfulness. The more semantic load the frame carries, the more conditional novelty it contains. That is, the viewer needs more time to read and interpret this information. The informational content of the frame affects its visual appeal - the capacity of its elements to attract and retain the viewer's attention (see Fig. 1) [5]. Informational content is an important element that determines the expressiveness of any frame, including a film frame [11]. However, according to Zheleznyakov, informational content is difficult to control due to the lack of regulatory and controlling tools.

According to the principles of art photography [10], deep zones affect the expressiveness of a portrait. The studies about influence of faces presence on the viewing pattern are based on stimulus material without a background, that does not

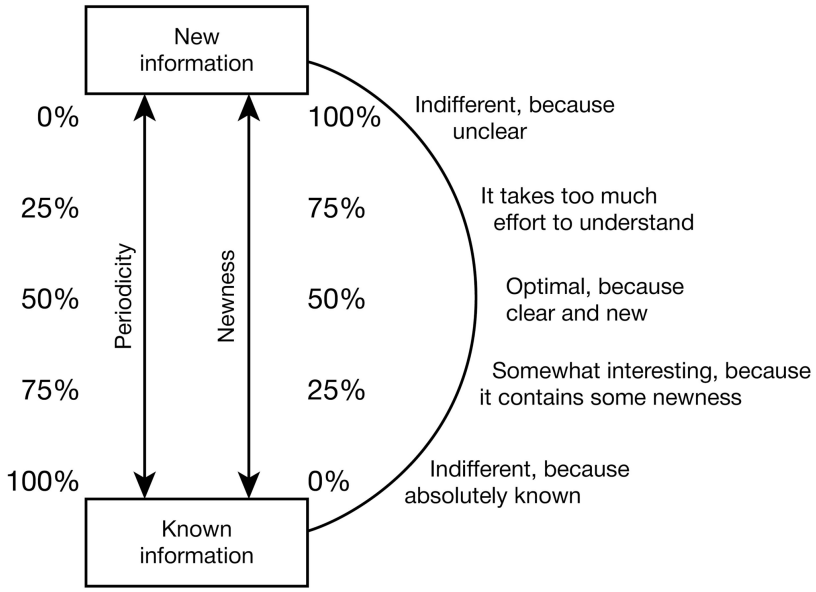


Fig. 2. The viewer's interest degree depending information content of a film frame

allow us to assess the effect of the level of detail of the background on the viewing pattern [6, 7, 8, 9].

We formulated a hypothesis that the informational content of the frame depends on the compositional construction (see Fig. 1) and the additional points of interest in the frame, which determined by the plot. In this case, the plot means a certain relationship between the object (model) and the subjects (additional objects and background). The goal of this work is to obtain experimental data of the pattern of viewing a film frame and to identify statistical patterns for confirming or refuting the formulated hypothesis.

The goal of the research is to conduct an experimental study of the influence of informational content has on the parameters of the viewing pattern of film frame. A viewing pattern is defined as a set of quantitative parameters of oculomotor activity obtained using the eye-tracker software and hardware complex when the observer examines the stimulus material [13].

3. Preparing the stimulus material

To compose a visual series of stimulus material (see Fig. 3), mainly waist and chest portrait shots of horizontal orientation in 16:9 format were selected. They met a number of requirements: the obscurity of the personalities of the models (to

Fig. 3. Examples of stimulus material with different informational content

exclude the moment of "recognition" of a famous person seen earlier), limited age range (people from 30 to 50 years old), lack of racial differences in models. Portrait photographs of people with closed eyes were excluded during the selection of the stimulus material, as well as photographs in which the model's gaze is directed towards the camera lens (in order to avoid the influence of the psychological factor - eye-to-eye gaze when viewing).

When developing the stimulus material, photographs were selected in four groups with different complexity of the composition, determined by the plot. Group 0 - a photographic portrait against a simple uniform background. Group 1 - a photographic portrait against an active background (open-air portrait). Group 2 - a social portrait in the studio. Group 3 - social portrait in the environment. Groups 2 and 3 imply the presence of a secondary object. The relationship of the main and secondary object is determined by the plot of the portrait. Half of the photos were converted to black and white color scheme and used as a control group. When developing incentives, an assumption was made: the more complex the subject of the photograph, the higher the informational content.

4. Experimental setup

There are researches on eye tracking technologies. The authors of [14] introduced the latest gaze tracking and gaze assessment technologies. Also, there are practical methods for measuring human-computer interaction [15]. For the current experiment, we used a hardware complex that records oculomotor activity - an eye-tracker. A chair and a head support were adjusted for each observer (see Fig. 4). The eye-tracker was also calibrated individually.

For the first stage of the experiment, 16 frames were chosen: 8 male and 8 female portraits in four color-plot combinations. Half of the stimulus images were converted to black and white, half to a complementary color scheme (the object is contrasting

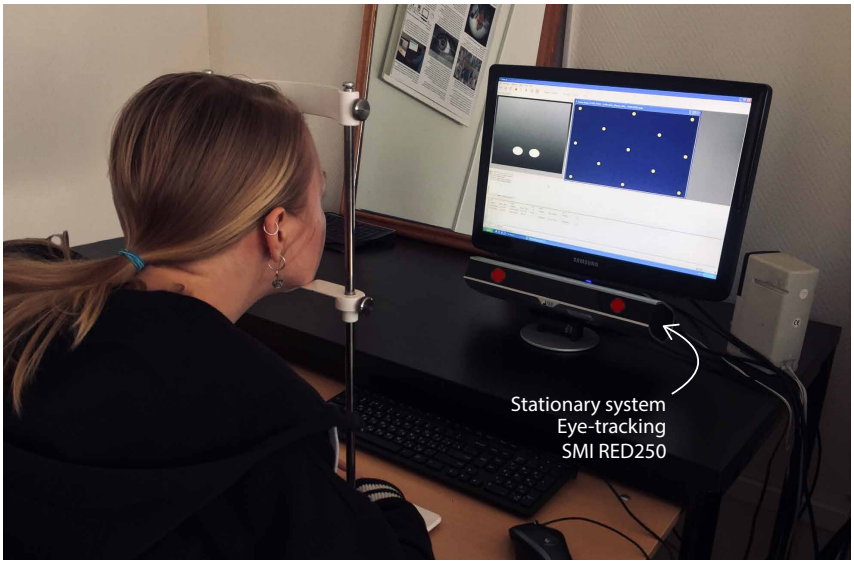


Fig. 4. An experimental setup with an eye-tracker: a computer monitor, an eye-tracker, and a head support

with respect to the background in both cases). For the second stage, the primary row was supplemented with 16 more portraits, selected in the same way.

At the first stage, 16 photographic portraits were shown for the observers. At the second stage, 16 more similar photographic portraits were added to them. At the first stage of the experiment, the observer was asked to memorize portraits, which are shown on the monitor screen sequentially in random order. The time for memorization was not limited. At the end of the first stage of the experiment, a time interval of 30 minutes is set for each observer before proceeding to the second part of the experiment. At the second stage, the task for the observers is recognizing the stimulus. Each participant of the experiment is asked to choose from 32 stimuli with portrait frames that he/she has seen before. The order of demonstration of the material in the second stage is also sequential and random. The observer had to answer "yes" or "no", that is, to answer: whether he/she has seen this stimulus at the first stage of the experiment. The duration of the examination of the stimulus by the observer at the recognition stage was set independently by the observer.

To teach the observers to interact with the interface and train the solution to the recognition problem, at the beginning of the second stage, the two test stimuli were shown for observers.

Table 1. Average values of the parameters of the pattern viewing the stimulus material at the first stage of the experiment

	Time(ms)	Num Fix	Fix Dur (ms)	NumSac	SacDur (ms)
Male	**5430**	**15.1**	**4030**	**18.3**	**904**
Female	**6480**	**15.1**	**4180**	**26.6**	**1370**

Table 2. Significance criterion p-value

	Time	NumSac	SacDurTotal
p-value	**0,034183**	**0,000093**	**0,000068**

5. Experimental results

The experiment involved students of Peter the Great St. Petersburg Polytechnic University aged 20 to 28 years. 20 males and 20 females participated. There are 6768 (stage 1) + 3961 (stage 2) fixations and 10038 (stage 1) + 4093 (stage 2) saccades were collected. As stated earlier, the experiment consisted of two stages. At the first stage, the observers' task was to remember the demonstrated stimulus material. The average values of the parameters of the viewing pattern are presented in Table 1. Here follows the explanation of the column labels:

- Time - average time of stimulus consideration;
- Num Fix - average number of fixations when considering a stimulus;
- Fix Dur Total - average fixation time when considering one stimulus;
- NumSac - average number of saccades when considering one stimulus;
- SacDurTotal - average time of saccades when considering one stimulus.

It should be noted at the first stage of experiment (memorizing stage) the informative factor does not affect the parameters of the viewing pattern. However, a statistically significant difference was found in the parameters of the viewing pattern for males and females. The graphs of the distribution density of the parameters of the viewing pattern at the first stage are presented in figures 5, 6. The informational content are represented by:

- 0 - studio portrait against a uniform background;
- 1 - portrait in the environment (open air);
- 2 - a social portrait in the studio;
- 3 - social portrait in the environment.

Statistical processing of experimental data was carried out using analysis of variance ANOVA [16]. The values of the p-value criterion are presented in Table 2.

Analyzing the experimental data, we can conclude that male observers solve the problem of memorizing the stimulus faster than female. At the same time, the

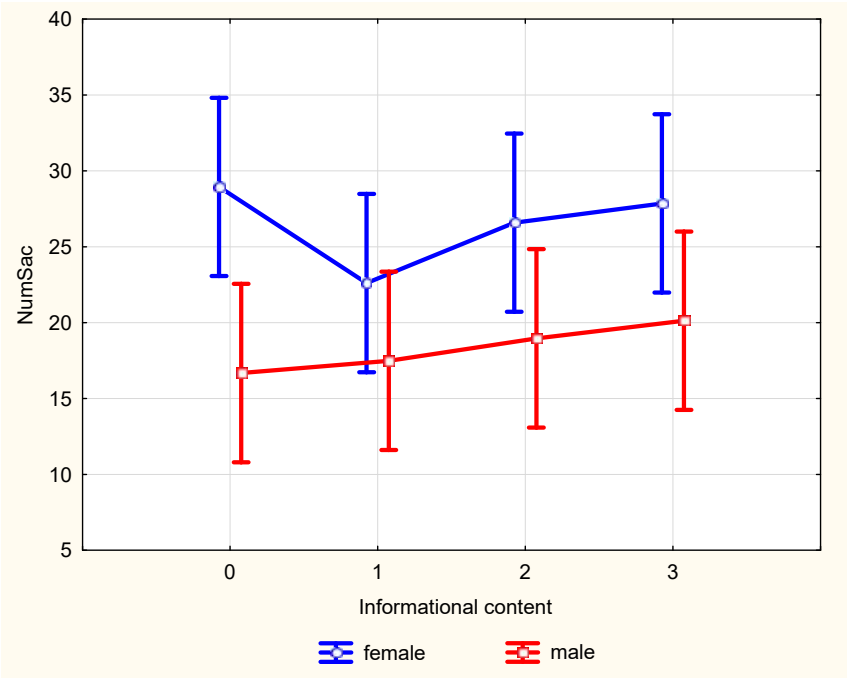


Fig. 5. The distribution density of the average number of saccades depending on the gender (male, female)

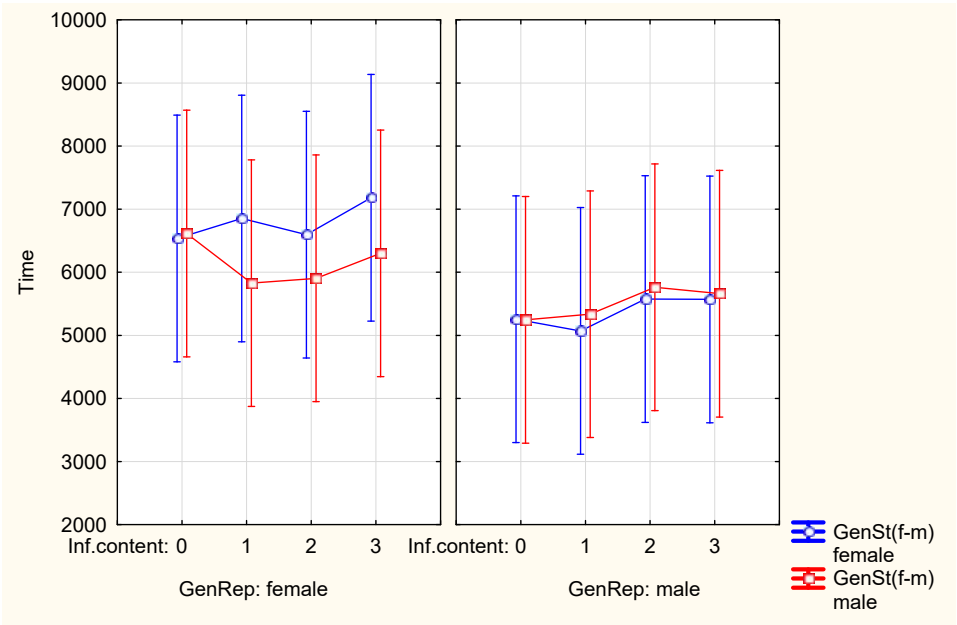


Fig. 6. The distribution density of the average time spent viewing a stimulus depending on the stimulus gender (a male or a female is displayed in the frame) for both sexes (female, male)

Table 3. Average values of the parameters of the viewing pattern for considering the stimulus material at the second stage of the experiment

	Inf	Time(ms)	Num Fix	Fix Dur (ms)	NumSac	SacDur (ms)
Male	**0**	**1090**	**4.45**	**814**	**4.63**	**214**
Male	**1**	**1040**	**4.04**	**796**	**4.07**	**190**
Male	**2**	**916**	**3.69**	**661**	**3.63**	**179**
Male	**3**	**917**	**3.63**	**697**	**3.65**	**171**
Female	**0**	**1420**	**5.01**	**977**	**5.97**	**289**
Female	**1**	**1320**	**4.47**	**901**	**5.63**	**262**
Female	**2**	**1090**	**3.88**	**782**	**4.49**	**221**
Female	**3**	**1160**	**3.86**	**804**	**4.46**	**220**

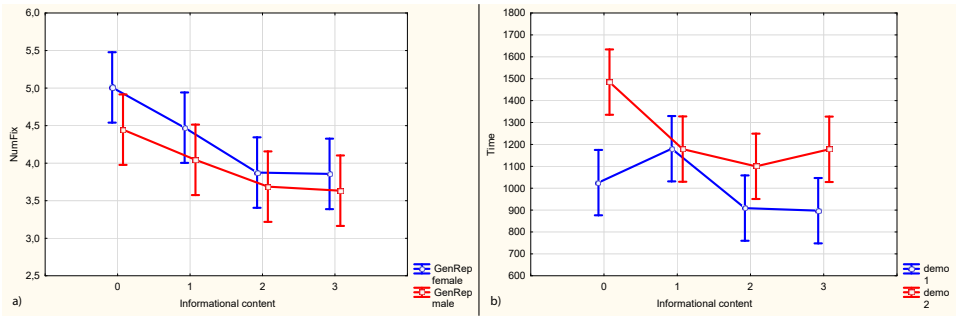


Fig. 7. The distribution density of a) the average number of fixations when considering a stimulus, depending on the informational content factor for two values of the gender (male, female) b) the average time of stimulus viewing depending on the informational content factor for two values of the demonstration factor (1 - seen at the first stage, 2 - not seen at the first stage)

average number of fixations in male and female observers practically does not differ (Table 1), but the average number and duration of saccades is differ. The average values of the parameters of the viewing pattern at the second stage of the experiment are presented in Table 3.

Figure 7 shows the distribution density of the parameters of the viewing pattern at the second stage of the experiment. The results of the execution are presented in Table 4. For deciding on statistical significance, the p-value = 0.05 [17].

Table 4. Average values of the parameters of the viewing pattern for considering the stimulus material at the second stage of the experiment

	factor	NumFix	Time
Inf	**0**	**0,000059**	**0,002884**
GenRep	**1**	**0,040390**	**0,000003**
Demo	**2**	**0,000023**	**0,000021**

6. Conclusion

Based on the results of statistical processing of the data obtained during this experiment, the following conclusions can be made:

- The informational content of the frame has a statistically significant effect on the parameters of the frame viewing pattern when performing the recognition task;
- The demonstration factor has a statistically significant effect on the parameters of the frame viewing pattern;
- The color rendering factor does not have a statistically significant effect on the parameters of the frame viewing pattern;
- Statistically significant differences were found in the patterns viewing the stimulus material between males and females (Fig. 7).

At the first stage of the experiment, when performing the task of remembering a frame, the influence of the informational content factor has no statistical significance on the parameters of the viewing pattern. Male observers perform the task of memorizing a frame faster due to the smaller number of saccades and their shorter duration (Table 1) At the second stage, male observers also perform the assigned task of frame recognition faster (Fig. 6, Table 3). The findings suggest that male observers are more focused on task completion and less distracted. An important result of the performed experiment is the identification of a statistically significant effect of the informational content of the frame on the parameters of the viewing pattern. It should be noted that the social portraits in the studio and the social portraits in the real setting do not have a statistically significant difference in viewing patterns.

7. Acknowledgment

The work was carried out at Peter the Great St. Petersburg Polytechnic University in Higher School of Design and Architecture under the guidance of professor of Higher School of Design and Architecture, Doctor of Technical Sciences Meshcheryakov S. V. and associate professor of Higher School of Design and Architecture, Ph. D. Yanchus V. E.

REFERENCES

1. Burr D., Morrone M. C. Eye movements when perceiving information // Eye movements: building a stable world from glance to glance. Curr Biol. 2005. V. 15(20). P. 839–40.
2. Jonides J., Irwin D. E., Yantis S. A picture is folded from the fixations. // Integrating visual information from successive fixations Science. 1982. V. 215. P. 192–194.
3. Mescheryakov S. V., Yanchus V. E., Borevich E. V. Experimental Research of Digital Color Correction Models and Their Impact on Visual Fixation of Video Frames // Humanities and Science University Journal. 2017. V. 27. P. 15–24.
4. Yanchus V. E., Borevich E. V. Investigation of the value of the color solution in the process of harmonization of the film frame // Scientific and technical bulletin of SPbSPU. 2016. V. 4. P. 53–68.
5. Borevich E. V., Mescheryakov S. V., Yanchus V. E. Statistical Model of Computing Experiment on Digital Color Correction // DCCN. 2019. P.140–150.
6. Janik S. W., Wellens A. R., Goldberg M. L., Dell'osso L. F. Eyes as the center of focus in the visual examination of human faces // Perceptual and Motor Skills. 1978. V. 47. P. 857–858.
7. Sheehan M. J., Nachman M. W. Morphological and population genomic evidence that human faces have evolved to signal individual identity // Nat. Commun. 2014. V. 5.
8. Hickman L., Firestone A., Beck F., Speer S. Eye fixations when viewing faces // Journal of the American Dental Association. 2010. V. 141(6). P. 40–6.
9. Bindemann M., Scheepers C., Burton A. M. Viewpoint and center of gravity affect eye movements to human faces // JVis. 2009. V. 9(2). P. 1–16.
10. Dyko L. P. Fundamentals of composition in photography. M.: Higher school, 1988.
11. Zheleznyakov V. N. Color and contrast. Technology and creative choice. Tutorial. M.: VGIK, 2001.
12. Lotman Y. M. Semiotics, cinema and problems of cinema aesthetics. About art, 1998.
13. Orlov P. A., Laptev V. V., Ivanov V. M. On the question of the use of eye-tracking systems // Scientific and technical statements of SPbSPU. 2014. V. 5(205). P. 84–94.
14. Norman D. A., Draper S. W. User Centered System Design: New Perspectives on Human-Computer Interaction. Lawrence Erlbaum Associates, 1986.
15. Majaranta P., Bulling A. Eye tracking eye-based human-computer interaction // Advances in Physiological Computing. 2014. P. 39–65.

16. Kabakov R. I. R in action. Analysis and visualization of data in the program R / from English Volkova P. A. M.: DMK Press, 2014.
17. Motrenko A., Strijov V., Weber G.-W. Sample size determination for logistic regression // Journal of Computational and Applied Mathematics. 2014. V. 255. P. 743–752.

UDC: 004.82; 621.391

Ontology-based model for sensor network fault management

A.Y. Grebeshkov[1]

[1]Povolzhskiy State University of Telecommunications and Informatics, 443010, L.Tolstoy str., 23, Samara, Russia

grebeshkov-ay@psuti.ru

Abstract

Ontology-based approach suggests an effective method of sensor network failure sources identification for a quality-of-service analysis. Designed subject-oriented ontology can be used for an identification and classification of failure message sources at a heterogeneous environment. It helps to create unified semantic model for a sensor network failures description in the context of interconnections and interactions between all network elements and services. The ontology-based model for sensor network fault management can be used for a logical verification of primary failure source identification and with supervised learning model as a technique for automatically design of failure source classifier.

Keywords: Fault management, ontology-based model, sensor network, supervised learning, OWL

1. Introduction

At a framework of Industry 4.0 sensor networks integrate heterogeneous data and telemetry data sources with a wide range of modern telecommunication technologies and data processing methods such as cloud computing, fog and edge computing [1]. Wireless telecommunications are the important part of this framework as a subsystem for a data transfer between input sources, processing units and knowledge-based decision-making application. As a sensor infrastructure as a telecommunication subsystem requires the understanding of failure messages context for support the end-to-end quality of service. Thus, information and fault message data should be interpreted with a common semantic standard in the context of primary failure information sources finding. Next an ontology of the fault description and fault management for a sensor network domain proposed.

2. Ontology-based fault model design

For this day in telecommunications many fault management methods have been developed which was based on signal analysis with object-oriented model-based methods and analytical model. A great number of methods require an obvious and general model for representing of the input fault event and output system reaction relationships description. Using above approaches, it is very difficult to implement a general approach for the fault management and diagnosis of all type of sensor networks' faultier. It is needed to highlight that sometimes different types of sensor nodes have different components and design principles and can use different telecommunication technology for data transfer that resulting in implementation of various diagnostic and fault management methods. Even for the same type of sensor nodes, same type or version program or hardware components might be a quite incompatible due to repair or update. Thus, there is a problem for computer control system to recognize a primary fault source, sometimes is hard to estimate a degree of service degradation affected by a sensor's failures, and to reuse integrated fault diagnosis knowledge-based model of fault management for a heterogeneous sensor network infrastructure.

To overcome these difficulties the ontology-based approach can be proposed, basically due to a semantic and logical mechanism concerning definition of concepts and their semantic relationships. Use of ontology approach and semantic framework result in finding and modeling concepts, classes, attributes, relationships, and other knowledge of the domain semantically, which enables the control system to understand and to make classifications of the fault events for heterogeneous types of sensor networks. The next step is sharing and reusing this knowledge for a sensor service degradation estimation with ontology-based approach.

For is further studying a hybrid approach for ontology design will be used [2]. It might take in account results concerning Internet of thing process management with particular and incomplete fault message description [3; 4] and definition of cases and concepts that was formulated in high-level Semantic Sensor Networks (SSN) [5], but without conceptualization of failures as event. In DogOnt ontology there are appliances modeling with concepts design for control, state changing and state request message only. Finally, SAREF ontology includes only sensor node modelling characteristics without networking environment concepts description [7].

Proposed ontology-based fault model includes classes like as *Fault*, *Sensor network services*, *Collaborative information processing*, *RadioChannel*, *Gateway, Wireless Sensor Network*, *Backbone Network*, *Communication Node*, as shown in Figure 1. The *Sensor Node* is composed of three sub-classes like *Sensor*, *Component* and *Actuator*. All these classes and sub-classes can be described as ontology concepts with definition in specially defined vocabulary. An example of term in vocabulary can be next,

"Sensor node is a sensor networks element or network element, which includes at least one sensor and, possibly, actuators with the capabilities of wireless communication and/or data processing directly at this node".

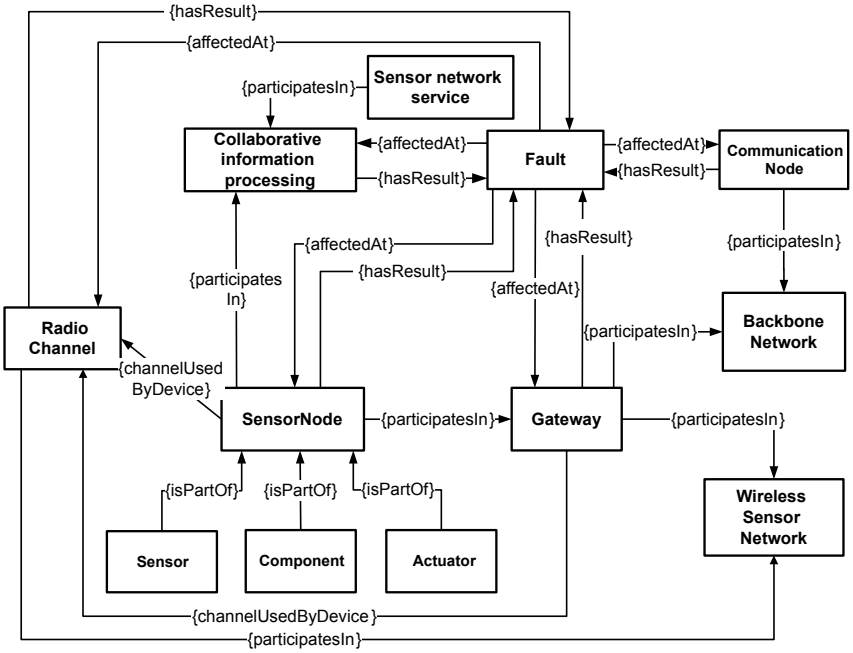


Fig. 1. Ontology-based sensor network fault model

In accordance with OWL base principles and notations [8; 9] an axiom on the properties of *Sensor Node* can be written by equation (1) - (3):

$$ObjectPropertyAssertion\begin{pmatrix} :\ isPartOf\ :Sensor \\ :\ SensorNode \end{pmatrix}; \tag{1}$$

$$ObjectPropertyAssertion\begin{pmatrix} :\ isPartOf\ :Component \\ :SensorNode \end{pmatrix}; \tag{2}$$

$$ObjectPropertyAssertion\begin{pmatrix} :\ isPartOf\ :Actuator \\ :SensorNode \end{pmatrix}. \tag{3}$$

With next axiom can be declared that relations *isPartOf* has a property of equivalence formulated as:

$$\begin{array}{l} Declaration\,(ObjectProperty\,(:hasPart))\,; \\ InverseObjectProperty\,(:isPartOf)\,. \end{array} \tag{4}$$

If some *Objects* are the part of any *Network Element*, than *FAULT* event at one object can affect another object and can be modeled with next equation:

$$\begin{array}{l}\forall Object_i, Object_j \quad useNetworkElement\,(Object_i, Object_j) \\ \Rightarrow FAULT(Object_i) = FAULT(Object_j).\end{array} \tag{5}$$

Finally, an example of logical rule to find a primary *FAULT* message type source as a *Gateway* object can be formulated in non-rigorous formal notation as:

$$\begin{array}{c} SensorNode\,(?SNode)\widehat{}Gateway\,(?SGateway)\widehat{} \\ \widehat{}BackboneNetwork\,(?BBNetwork)\widehat{} \\ CommunicationNode\,(?CNode)\widehat{} \\ isConnectedTo\,(?SNode, ?SGateway)\widehat{} \\ isConnectedTo\,(?SGateway, ?BBNetwork)\widehat{} \\ isConnectedTo\,(?BBNetwork, ?CNode)\widehat{}Fault\,(?Fault)\widehat{} \\ affectedUnit\,(?Fault, ?SNode)\widehat{} \\ affectedUnit\,(?Fault, ?SGateway)\widehat{} \\ affectedUnit\,(?Fault, ?BBNetwork)\widehat{} \\ affectedUnit\,(?Fault, ?CNode)\widehat{} \\ hasStatus\,(?SNode, StatusNoFault)\widehat{} \\ hasStatus\,(?BBNetwork, StatusNoFault)\widehat{} \\ hasStatus\,(?CNode, StatusNoFault) \Rightarrow \\ hasFaultSource\,(?Fault, ?SGateway)\,. \end{array} \tag{6}$$

In (6) an expression like *SensorNode* is a class identifier or unary predicate identifier, *(?SNode)* is a unary predicate's name of variable or individual object of model, *isConnectedTo* is a binary predicate identifier marking a relation between two objects, *affectedUnit* is a binary predicate identifier for description of *Fault* affection at any object, *hasStatus* and *hasFaultSource* are a binary predicate identifiers for description of failure or non-failure state. Proposed method can be used for automatically design of classifier for a sensor network fault management with supervised learning or as verification technique [10;11] for primary fault message identification correctness.

3. Conclusion

This paper includes result of ontology-based approach realization for description of fault event at a sensor network with different types of nodes, gateways in heterogeneous telecommunication networks environment. A structure of ontology-based sensor network model with general classes and relations between classes proposed, some axioms formulated and example of logical rules for a find of primary fault message sources proposed. For a further studying ontology-based model can be

realized with open-source ontology editor "Protege" or another product for practical implementation of research results. Another part of future research would be devoted for supervised machine learning with an automatically ontology-based classifier design and logical proof of primary fault message identification correctness.

REFERENCES

1. Kumar V. R. S., Khamis A., Fiorini S. et. al Ontology for Industry 4.0 //The Knowledge Engineering Review. 2019. V. 34. P. 1–14.
2. Wache H., Vögele T., Visser U. et. al Ontology-based information integration: A survey of existing approaches //Proceedings of the International Joint Conferences on Artificial Intelligence Organization (IJCAI-01). 2000. V. 47. P. 108–118.
3. Song Z., Cardenas A. A., Masuoka R. Semantic middleware for the Internet of Things //Proceedings of 2 International Internet of Things Conference. IEEE. 2010. P. 1–8.
4. Tao M., Ota K., Dong M. Ontology-based data semantic management and application in IoT-and cloud-enabled smart homes //Future Generation Computer Systems. 2016. Vol. 76. No. C. 20 pages.
5. Compton M., Barnaghi P., Bermudez L. et al. The SSN ontology of the W3C semantic sensor network incubator group // Journal of Web Semantics. 2012. Vol. 17. P. 25–32.
6. Bonino D., Corno F. Semantic middleware for the Internet of Things //Proceedings of 7 International Semantic Web Conference (ISWC 2008). 2008. P. 790–803.
7. Daniele L., den Hartog F., Roes J. Created in close interaction with the Industry: The Smart Appliances REFerence (SAREF) ontology //Proceedings of 7 International workshop Formal Ontology Meet Industry (FOMI 2015). 2015. LNBIP 225. P. 100–112.
8. Allemang D., Hendler J. Semantic Web for the Working Ontologist: Effective Modeling in RDFS and OWL. Morgan Kaufmann, 1988.
9. OWL 2 and SWRL Tutorial, `https://dior.ics.muni.cz/~makub/owl/#propax`
10. Noshad Z., Javaid N., Saba T. Fault detection in wireless network through the Random Forest Classfier //Sensors. 2019. Vol. 19. No. 1568. 21 pages.
11. Grebeshkov A.Y. Optical transport network management via machine learning and ontology-based technique // Optical Technologies for Telecommunications 2019 (OTT2019). Proceedings SPIE 11516. 2020. 1151601. 8 pages.

UDC: 519.872

Analysis of the Amount of Information in Semi-Markov Flow

A.A. Nazarov[1], A.N. Moiseev[1], I.L. Lapatin[1], S.V. Paul[1], O.D. Lizyura[1], P.V. Pristupa[1], Xi Peng[2], Li Chen[2], Bo Bai[2]

[1]National Research Tomsk State University, 36 Lenina ave., 634050, Tomsk, Russia
[2]Theory Lab, Central Research Center, 2012Labs, Huawei Tech. Investment Co., Ltd 8/F, Bio-informatics Center, No. 2 Science Park West Avenue, Hong Kong Science Park, Pak Shek Kok, Shatin, N.T., Hong Kong

Abstract

In this paper, we consider semi-Markov flow as a model of bit-level traffic. Each request of the flow brings some arbitrary distributed amount of information. The current paper aims to investigate the amount of information received in semi-Markov flow. We use the asymptotic analysis method under the limit condition of growing time of observation to derive the limiting probability distribution of the amount of information received in the flow and build its approximation.

Keywords: semi-Markov flow, asymptotic analysis, Gaussian approximation

1. Introduction

In telecommunication systems, the models of arrivals usually capture the structure of traffic from a packet-level point of view. Despite the interest in traffic models, few studies take into account packet length. Traffic modeling is focused on capturing such properties of telecommunication flows as burstiness, self-similarity and long-range dependence [1, 2, 3].

The idea of modeling arrivals together with the size of packets described in paper [4]. Authors use batch Markovian arrival process (BMAP) to model packet size as a size of the batch. In paper [5], authors build the model of traffic based on discrete-time BMAP model using two counting processes: the number of arriving packets and the number of bytes in those packets. Both processes in the model are affected by the state of the underlying Markov chain. More ideas of using packet size in traffic modeling are described in [6].

We propose semi-Markov flow as a model of bit-level traffic, which allows us to take into account the length of packets in telecommunication systems. In our model, packets arrivals are driven by the semi-Markov process and the lengths of packets

follow the arbitrary distribution. To research the model, we use the asymptotic analysis method under the limit condition of the growing time of the flow observation. We build a Gaussian approximation of the cumulative distribution function of the amount of information received in the flow.

We have organized the paper as follows. In section 2, we present a mathematical model of semi-Markov flow. Section 3 is devoted to the derivation of the balance equation for the probability distribution of the process describing the amount of information received in the flow. In section 4, we investigate the model using the asymptotic analysis method under the limit condition of growing time and build a Gaussian approximation. Section 5 is dedicated to the concluding remarks.

2. Mathematical Model of Semi-Markov Flow

Semi-Markov flow is determined by semi-Markov matrix $\mathbf{A}(x)$. Elements $A_{k\nu}(x)$ of the matrix has the following from:

$$A_{k\nu}(x) = P\{\xi(n+1) = \nu, \tau(n+1) < x | \xi(n) = k\}. \quad (1)$$

We also take into account that

$$\mathbf{P} = \mathbf{A}(\infty), \quad (2)$$

where $\mathbf{P}$ is the transition matrix of embedded Markov chain $\xi(n)$ at the moments of state changes of the semi-Markov process. Moments t_n of arrivals in semi-Markov flow we determine as follows:

$$t_{n+1} = t_n + \tau(n+1).$$

Further, we use semi-Markov process $k(t)$, which is defined by equality

$$k(t) = \xi(n+1), \text{if } t_n < t \leq t_{n+1} = t_n + \tau(n+1). \quad (3)$$

Each request of the flow brings some random amount of information with arbitrary distribution given by cumulative distribution function $B(x)$.

We denote $S(t)$ as the amount of information received in semi-Markov flow during time t. The problem is to derive the probability distribution of process $S(t)$.

We also denote $z(t)$ as the residual time of next arrival in the flow and consider three-dimensional process $\{k(t), S(t), z(t)\}$.

3. Balance Equation for the Probability Distribution of the Flow State

Three-dimensional process $\{k(t), S(t), z(t)\}$ is Markovian. Thus, we consider the function

$$P_k(s, z, t) = P\{k(t) = k, S(t) < s, z(t) < z\}$$

and derive balance equation

$$\frac{\partial P_k(s,z,t)}{\partial t} = \frac{\partial P_k(s,z,t)}{\partial z} - \frac{\partial P_k(s,0,t)}{\partial z} + \sum_{\nu=1}^{K} \int_0^s \frac{\partial P_k(s-x,0,t)}{\partial z} dB(x) A_{\nu k}(z), \quad (4)$$

where $\frac{\partial P_k(s,0,t)}{\partial z} = \frac{\partial P_k(s,z,t)}{\partial z}\Big|_{z=0}$.

We introduce partial characteristic functions

$$H_k(u,z,t) = \int_0^\infty e^{jus} d_s P_k(s,z,t)$$

and denote vector characteristic function

$$\mathbf{H}(u,z,t) = \{H_1(u,z,t), H_2(u,z,t), ..., H_K(u,z,t)\},$$

identity matrix $\mathbf{I}$ and vector of ones $\mathbf{e}$. After that, we rewrite equation (4) together with additional equation obtained taking the limit by $z \to \infty$

$$\frac{\partial \mathbf{H}(u,z,t)}{\partial t} = \frac{\partial \mathbf{H}(u,z,t)}{\partial z} - \frac{\partial \mathbf{H}(u,0,t)}{\partial z}\{\mathbf{I} - \mathbf{A}(z)B^*(u)\},$$

$$\frac{\partial \mathbf{H}(u,t)}{\partial t}\mathbf{e} = \frac{\partial \mathbf{H}(u,0,t)}{\partial z}\{B^*(u) - 1\}\mathbf{e}, \quad (5)$$

where $B^*(u) = \int_0^\infty e^{jux} dB(x)$ is the characteristic function of the amount of information in one request of semi-Markov flow and $\mathbf{H}(u,t) = \mathbf{H}(u,\infty,t)$.

4. Asymptotic Probability Distribution of the Amount of Information Received in Semi-Markov Flow under the Limit Condition of Growing Time

We introduce the equality $t = \tau T$, where $\tau \geq 0$ and T is an infinite parameter, as the limit condition of growing time. Solving system (5) in the limit by $T \to \infty$, we formulate the following theorem.

Theorem 1. For characteristic function $H(u,t) = \mathbb{E}e^{juS(t)} = \mathbf{H}(u,t)\mathbf{e}$ in the limit condition of growing time the following equality holds:

$$\lim_{t\to\infty} \left\{ H(u,t) - \exp\left(ju\kappa_1 t + \frac{(ju)^2}{2}\kappa_2 t \right) \right\} = 0, \quad (6)$$

where

$$\kappa_1 = \frac{b_1}{\mathbf{r}\mathbf{A}_1\mathbf{e}}, \tag{7}$$

$$\kappa_2 = \frac{b_2}{\mathbf{r}\mathbf{A}_1\mathbf{e}} + 2b_1\mathbf{g}'(0)\mathbf{e}. \tag{8}$$

Here b_1 and b_2 are the first and second raw moments of distribution function $B(x)$, matrices $\mathbf{A}_1$ and $\mathbf{A}_2$ are determined by formulas

$$\mathbf{A}_1 = \int_0^\infty (\mathbf{P} - \mathbf{A}(x))dx,$$

$$\mathbf{A}_2 = \int_0^\infty x^2 d\mathbf{A}(x).$$

Vector $\mathbf{g}'(0)$ is the solution of inhomogeneous system of equations

$$\mathbf{g}'(0)(\mathbf{I} - \mathbf{P}) = \kappa_1(\mathbf{r} - \mathbf{R}),$$

$$\mathbf{g}'(0)\mathbf{A}_1\mathbf{e} = \frac{b_1}{2}\frac{\mathbf{r}\mathbf{A}_1\mathbf{e}}{(\mathbf{r}\mathbf{A}_2\mathbf{e})^2} - b_1.$$

Vector $\mathbf{r}$ is the steady state probability distribution of embedded Markov chain $\xi(n)$, which is the solution of the system

$$\mathbf{r} = \mathbf{r}\mathbf{P}, \quad \mathbf{r}\mathbf{e} = 1.$$

Vector $\mathbf{R}$ is the steady state probability distribution of semi-Markov process $k(t)$, which is given by formula

$$\mathbf{R} = \frac{\mathbf{r}\mathbf{A}_1}{\mathbf{r}\mathbf{A}_1\mathbf{e}}.$$

As we can see, the distribution of the amount of information received in semi-Markov flow is asymptotically Gaussian with mean $\kappa_1 t$ and variance $\kappa_2 t$.

We note that by setting $b_1 = 1$ and $b_2 = 1$, we obtain the case when the amount of information in a packet is deterministic and equal to one. Thus, the obtained result is valid for the number of packet arrivals in the flow.

5. Conclusion

We have considered the bit-level traffic model in form of semi-Markov flow. For the amount of information received in the flow, we have obtained the limiting probability distribution under the limit condition of growing time of observation. We have derived the explicit formula for the mean and variance of Gaussian distribution. Since the distribution of the packet length in the model is arbitrary, the results are applicable for the number of packets arrivals when we set the size of each packet is equal to one.

REFERENCES

1. X. Yang, A. P. Petropulu, The extended alternating fractal renewal process for modeling traffic in high-speed communication networks, IEEE transactions on signal processing 49 (7) (2001) 1349–1363.
2. T. Yang, R. Zhao, W. Zhang, Q. Yang, On the modeling and analysis of communication traffic in intelligent electric power substations, IEEE Transactions on Power Delivery 32 (3) (2016) 1329–1338.
3. W. Willinger, M. S. Taqqu, W. E. Leland, D. V. Wilson, et al., Self-similarity in high-speed packet traffic: analysis and modeling of ethernet traffic measurements, Statistical science 10 (1) (1995) 67–85.
4. A. Klemm, C. Lindemann, M. Lohmann, Modeling ip traffic using the batch markovian arrival process, Performance Evaluation 54 (2) (2003) 149–173.
5. P. Salvador, A. Pacheco, R. Valadas, Modeling ip traffic: joint characterization of packet arrivals and packet sizes using bmaps, Computer Networks 44 (3) (2004) 335–352.
6. J. Gao, I. Rubin, Multifractal analysis and modeling of long-range-dependent traffic, in: 1999 IEEE International Conference on Communications (Cat. No. 99CH36311), Vol. 1, IEEE, 1999, pp. 382–386.

UDC: 004.94

Modeling of non-reliable retrial queueing systems with collisions and catastrophic breakdowns

A. Kuki [1], T. Bérczes[1], Á.Tóth[1], J. Sztrik[1]

[1]University of Debrecen, Faculty of Informatics, Kassai u. 26., Debrecen, Hungary

{kuki.attila, berczes.tamas, toth.adam, sztrik.janos}@inf.unideb.hu

Abstract

The aim of the investigation is a closed retrial queueing system with a finite source. The server is non-reliable, and collisions of customers are considered. The server can be reached from the source or the orbit. If an incoming job finds the server busy, the service of the job at the server is interrupted and both of them are transferred to the orbit (collision). The non-reliable server is subject to catastrophic breakdown. It means, that all of the customers at the server and in the orbit are sent back to the source. The novelty of this paper is to investigate the phenomenon of the catastrophic breakdown in a collision environment. Our goal is to calculate the steady-state probabilities and the performance characteristics (utilization, response time, etc.) of the system with the help of a software package. Figures illustrate the effect of the system parameters on the performance measures.

Keywords: retrial queues, collision of customers, catastrophic breakdown

1. Introduction

There are several tools for modeling and studying working systems from different areas of the real world. One of the most effective tools is the retrial queueing system (RQ-system). In RQ-systems the customers are not lost in case of a busy system. When an incoming job from the outside world (in the models from the sources or the queue of the system) finds the server busy, joins a virtual waiting room called orbit and after a random, usually exponentially distributed waiting time it retries to reach the server again. The most frequent application fields of an RQ-systems are the call centers, computer networks, telecommunication systems, etc. Infinite source models have been investigated and applied by many authors, very large number of results were published in the literature. But there exist cases, where the finite source models (finite number of customers in the source) are more adequate to describe the behavior of the considered system. The most characteristic examples are the mobile networks, sensor networks, some IoT systems, and cognitive radio systems.

The random and multiple access protocols for these types of systems have been investigated, for example, in [1], [2].

In the real-life situations, unfortunately, the systems are subject to breakdowns that is why this situation has to be investigated. In the modeling process of the system, random server failures and the corresponding repairs are included. The system characteristics and performance measures are highly dependent on the non-reliable operation of the systems. Finite-source RQ-systems with server breakdowns and repairs have been investigated in several recent papers, for example in [3], [4], [5].

A non-reliable $M/M/1//N$ retrial queueing system with collisions of customers is considered in the present paper. Collisions of requests (or conflict of customers) can be occurred frequently in unsynchronized communication systems with a limited number of resources, for example, communication channels. In this case, the transmission is lost and the interrupted requests need to be retransmitted, consequently, the performance of the system is sub-optimal. Developing methods and protocols which can prevent the system from the phenomenon of conflicts or at least try to minimize the damage has great importance. In this direction some recent results can be found in [6], [7], [8].

The focus of this paper is the catastrophic breakdown. Retrial queueing models in which customers are removed from the system due to catastrophic or disaster events have been studied extensively in the literature. Modeling special systems, e.g. automatic teller machines needs different types of breakdowns. A catastrophic event can be, for example, mechanical failures or power outages. Disaster events are known also as a negative arrival or a negative customer. When a negative customer arrives at the system, it immediately removes the positive customer in service if present. The case, when a negative customer removes all the positive customers from the system at once, is called a disaster. Disaster events not only break the service of the current customer but break down the server. The customers from the server and the orbit are sent back to the source. Detailed studies on negative customers can be found in [9], [10] [11], and reference therein.

In this paper, a software tool is used for calculating the steady-state probabilities of the system. Using these probabilities the most important performance measures can be computed. Several sample examples illustrate the effect of different parameters on the distribution of requests in the system.

2. Description of the system

A finite source closed retrial queuing system of type $M/M/1//N$ is considered. As the Kendall's notation says, this is a single server system with a number of sources N. Two scenarios of the system can be investigated and compared:

- The common break-down mode. The system is non-reliable, that is the server is subject to random breakdowns after an exponentially distributed time. In the case of an idle server, the breakdown parameter is γ_0. When the server is busy, the breakdown parameter is γ_1. Furthermore, it is assumed that the job under service is sent to the orbit. The repair starts immediately after the breakdown. The distribution of the repair time is also exponential with parameter γ_2.
- The catastrophic break-down mode. This is the situation when a disaster event removes all of the customers from the system (from the orbit and from the server after interrupting the service). The repair of the system starts immediately. The same breakdown parameters are used as in the common breakdown mode, i.e. γ_0 and γ_1 for an idle server breakdown and a busy server breakdown, respectively, and γ_2 for the repair.

In both scenarios, the sources are blocked during the repair period of the server. No new request can enter into the system.

The dynamic behavior of the system is the following. The sources generate jobs (requests, customers) towards the server. The inter-request times of the job generation are exponentially distributed with parameter λ/N. After generating a request the source waits for a successful service. Until the end of service of the job, the source can not generate a new request. The generated customer reaches the server, which can be busy or idle state. When the server is empty (idle), the service of the job begins immediately, and the service times are assumed to be exponentially distributed with parameter μ. When the server is in a busy state and a new customer is arriving, a collision of the customers occurs. In this situation, the customer under service and the newly arrived customer are transferred into the orbit. From the orbit, the customers retry reaching the server again after an exponentially distributed time with parameter σ/N. See the model on Figure 1.

Let's denote $i(t)$ the state of the system, that is the number of customers in the service facility that is either in the orbit or under service, and let $k(t)$ denote the status of the server:

$$k(t) = \begin{cases} 0, \text{if the server is up and idle,} \\ 1, \text{if the server is up and busy,} \\ 2, \text{if the server is down and under repair.} \end{cases}$$

.

Let $P(k(t) = k, i(t) = i) = P_k(i, t)$ the probability that at the time t there are i customers in the system and the server is in the state k. With the assumptions above the process $X(t) = \{k(t), i(t)\}$ is a 2-dimensional Markov-chain with a state space of $\{0, 1, 2\}x\{0, 1, ..., N\}$.

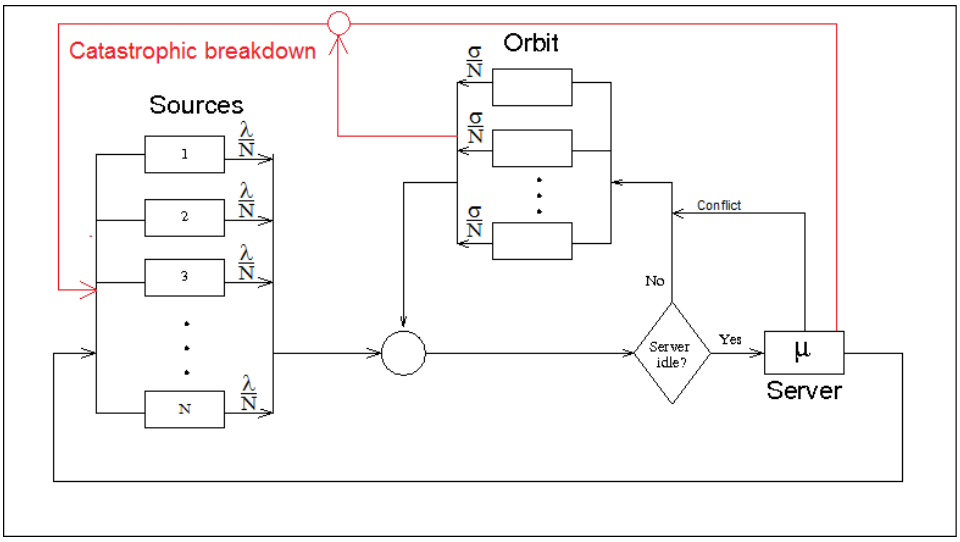


Fig. 1. System model

When the service of a request is successful, the request goes back to the source. All the random variables involved in the model construction are assumed to be totally independent from each other.

The $X(t) = \{k(t), i(t)\}$ process is a finite state Markov-chain, so the steady-state operation can be assumed: $P_k(i,t) = P_k(i)$.

The steady-state Kolmogorov balance equations for the normal breakdown case can be seen in [12]. The balance equations for the catastrophic breakdown case also can be formulated.

To demonstrate the effect of the input parameters on the operation of the system, different performance measures have to be calculated from the steady-state probabilities. These characteristics are the mean response time, mean waiting time, the utilization of the server, etc. For example, the mean number of customers in the system $\overline{Q}$ and in the orbit $\overline{O}$ can be obtained as

$$\overline{Q} = \sum_{i=0}^{N} iP(i), \qquad \overline{O} = \overline{Q} - P_1.$$

3. Numerical results

There are methods for solving the steady-state equations. Here an analytical software tool, namely the MOSEL-2 was chosen. With the assumption of exponentiality of the system parameters, this tool is effective and quick for a reasonably

large number of sources. The MOSEL tool builds up the system equations. The steady-state probabilities of the system are calculated.

Figure	Model	λ	μ	σ	N	γ_0	γ_1	γ_2
2	Catastrophic	1	1	5	100	Legend	Legend	1

Table 1. Numerical values of model parameters

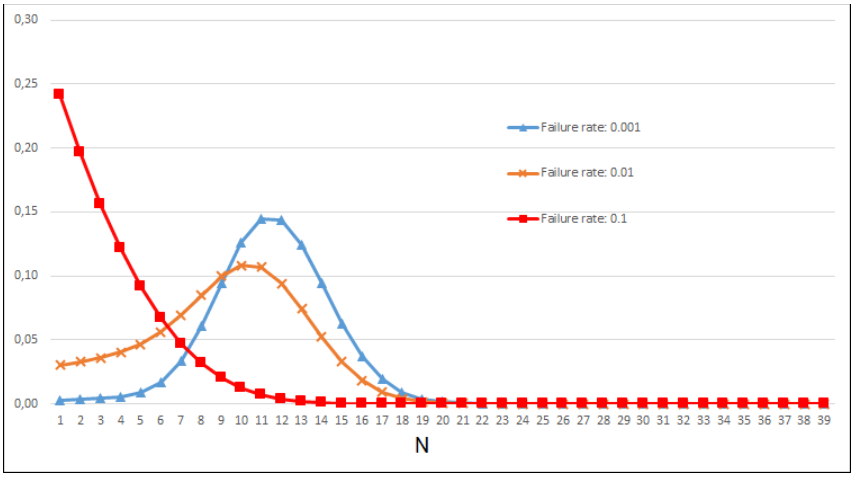


Fig. 2. The system probabilities with different failure rates

On Figure 2 the steady-state probabilities of the system can be seen. The parameters can be found in Table 1. Different failure rates were applied. The failure rates here for busy and idle states are the same. For a low failure rate, the usual normality of the probabilities can be observed [13]. For larger failure rates the property of the normal distribution of the system probabilities does not hold anymore. The reason for this is the frequent catastrophic breakdown, where the customers spend only a very short time in the system.

Based on the system probabilities further figures can be formed displaying the performance measures, e.g. Response times, waiting times, utilization, etc.

4. Conclusion

The paper compares the phenomena of common breakdown and the catastrophic breakdown in a non-reliable system. This comparison has great importance nowadays, because there are sensitive systems (e.g. automated teller machines), which have to face both types of failures. Equations, formulas, and further figures are not presented here due to the limitation of the available space.

Acknowledgment

The research work was supported by the construction EFOP - 3.6.3 - VEKOP - 16-2017-00002. The project was supported by the European Union, co-financed by the European Social Fund.

The research work was supported by the Austro-Hungarian Cooperation Grant No 106öu4, 2020.

REFERENCES

1. J.R. Artalejo and A. Gomez Corral. *Retrial Queueing Systems: A Computational Approach*. Springer, 2008.
2. Jeongsim Kim and Bara Kim. A survey of retrial queueing systems. *Annals of Operations Research*, 247(1):3–36, 2016.
3. B. Almási, J. Roszik, and J. Sztrik. Homogeneous finite-source retrial queues with server subject to breakdowns and repairs. *Math. Comput. Modelling*, 42(5-6):673–682, 2005.
4. Jinting Wang, Linfei Zhao, and Feng Zhang. Analysis of the finite source retrial queues with server breakdowns and repairs. *Journal of Industrial and Management Optimization*, 7(3):655–676, 2011.
5. Feng Zhang and Jinting Wang. Performance analysis of the retrial queues with finite number of sources and service interruptions. *Journal of the Korean Statistical Society*, 42(1):117–131, 2013.
6. Ahsan-Abbas Ali and Shuangqing Wei. Modeling of coupled collision and congestion in finite source wireless access systems. In *Wireless Communications and Networking Conference (WCNC), 2015 IEEE*, pages 1113–1118. IEEE, 2015.
7. Anatoly Nazarov, Anna Kvach, and Vladimir Yampolsky. *Asymptotic Analysis of Closed Markov Retrial Queuing System with Collision*, chapter 1, pages 334–341. Springer International Publishing, Cham, 2014.
8. T. V. Lyubina and A. A. Nazarov. Research of the non-markov dynamic retrial queue system with collision (in russian). *Herald of Kemerovo State University*, 1(49):38–44, 2012.
9. Sudha Subramanian et al. A stochastic model for automated teller machines subject to catastrophic failures and repairs. *Queueing Models and Service Management*, 1(1):75–94, 2018.
10. B Thilaka, B Poorani, and S Udayabaskaran. Performance analysis for queueing systems with close down periods subject to catastrophe. *International Journal of Pure and Applied Mathematics*, 119(7):39–57, 2018.

11. UC Gupta, Nitin Kumar, and FP Barbhuiya. A queueing system with batch renewal input and negative arrivals. In *Applied Probability and Stochastic Processes*, pages 143–157. Springer, 2020.
12. A. Kuki, J. Sztrik, T. Bérczes, Á. Tóth, and D. Efrosinin. Numerical analysis of non-reliable retrial queueing systems with collision and blocking of customers. *Journal of Mathematical Sciences*, 248:1–13, 2020.
13. A. Kuki, T. Bérczes, Á. Tóth, and J. Sztrik. Numerical analysis of finite source markov retrial system with non-reliable server, collision, and impatient customers. *Annales Mathematicae et Informaticae*, 51:53–63, 2020.

UDC: 519.248

Distributed Computing of Embarrassingly Parallel R Applications using RBOINC Package

S. Astafiev[1,2] and A. Rumyantsev[1,2]

[1]Institute of Applied Mathematical Research, KarRC RAS, 11 Pushkinskaya Str., Petrozavodsk, Russia

[2]Petrozavodsk State University, 33 Lenina Str., Petrozavodsk, Russia

seryymail@mail.ru, ar0@krc.karelia.ru

Abstract

R programming language is commonly used for statistical computing, data science and stochastic simulation. Existing packages for R allow to run parallel code on various parallel architectures, however, the support for distributed (volunteer) computing is rather weak. This article describes a new R package `RBOINC` that allows to run parallel code on grid systems via BOINC, a open source system for grid computing, which is a promising approach for parallel stochastic simulation.

Keywords: distributed computing, desktop grid, volunteer computing, BOINC, R software

1. Introduction

Parallel and distributed computing are widely used technologies of computationally consuming application speedup. Potential applications in scientific research include the areas of modeling and simulation, estimation and data analysis.

A typical parallel application is executed in several instances known as *threads*. In terms of architecture, the applications can be classified by the degree of interdependence of parallel threads, and the amount of synchronizations needed within the application. As such, the parallel architectures correspond to various classes of applications based on the demand of synchronization. These are: Uniform Memory Access (UMA), Non-Uniform Memory Access (NUMA), and Distributed Memory, ordered by the possible synchronization speed decreasingly.

The class of Distributed Memory systems includes various implementations of grid computing systems, including the Desktop Grids and Volunteer Computing. The applications most suitable for these systems belong to the class of so-called *embarrassingly parallel*. This means that a large computing task can be separated

into a huge number of small independent subtasks in such a way that allows to aggregate the computing results of the subtasks into the solution of the original task. In the field of stochastic modeling and simulation, a few examples from the class of embarrassingly parallel applications are the perfect simulation technique [1], time-parallel simulation [2, 3], discrete-event simulation [4] and heuristic optimization [5], to name a few.

One of the popular environments both for data analysis [6] and for stochastic modeling in queueing [7] is the R language environment. The language is extensible by over 18000 packages in various application fields available at CRAN repository. Existing packages for R support all the aforementioned architectures of parallel machines. There are two main types of parallel processing for R:

- creation a subprocess via fork system call (at unix-like machines, UMA architectures).
- running an additional R interpreter and establishing a connection with it at network level.

In particular, the following packages are widely used: `doMC` (UMA, unix-like systems only), `rslurm` (supercomputing backend), `snow` (simple network of workstations, connected by a conventional network), `parallel` (the one included into the basic distribution by default). At the same time, there are, to the best of our knowledge, almost no packages specific to the Volunteer Computing systems (which is a good choice for establishing a sustainable computing environment at low cost).

BOINC is a Volunteer Computing system [8] used for utilization of the idle CPU time and reducing the energy waste. All the resources are donated by the volunteers at no cost. Any BOINC project has a central server controlled by the project maintainers who create and upload the necessary applications, tasks, and summarize the results of computations. Usually the number of tasks in a project far exceeds the number of BOINC users. The tasks are distributed to the computing resources donated by the users with a certain level of redundancy. Upon the completion of computations, the results are uploaded to the server and summarized afterwards. As such, organizing a BOINC project may allow one to obtain significant computing resources at almost no cost.

In summary, using BOINC for certain tasks in simulation modeling and optimization is promising. Thus, it is important to develop an R software package to equip the researchers with a software solution for parallel simulation over the volunteer computing resources. In this paper we introduce such a software package named `RBOINC`, following the concept introduced in [9].

The structure of the paper is as follows. Firstly, we introduce the architecture of our framework. Secondly, we briefly outline the scheme of the working process. We finalize the scope with a conclusion.

2. Parallel Backend for R Using BOINC High-level Architecture

The presented software package `RBOINC` is available at R-forge and can be installed within R environment using the command

```
install.packages("RBOINC.cl", repos="http://R-Forge.R-project.org")
```

The package allows to organize a seamless connection from R environment to BOINC computing platform. To do so, a backend is used, which contains three parts:

- **Client part** - an R package that is installed by the package users (researchers) on their computers.
- **Virtual machine** - a specific virtual machine that runs jobs on the donated BOINC volunteer's computers.
- **Server part** - a set of scripts and configuration files that need to be run on BOINC server.

Next, we briefly describe the structure of these parts.

2.1. Client part. It is used by the researchers who plan to utilize BOINC as a computing resource. This package is intended for non-BOINC specialists who need to run some computations in parallel. Due to the long latency while transmitting, validating and assimilating the results, it is recommended to use this package for tasks with relatively large computational time.

This package provides following functions:

- Connection to BOINC server. This function requires the server's address, user login and password. There is an alternative method for ssh users to use keyfile for establishing the connection.
- Creation and transmission of tasks to the BOINC server.
- Requesting the server on the status information of jobs submitted earlier.
- Disconnection from BOINC server.
- The debug function. This function makes a batch of jobs and runs it locally on the user's computer. In the working process, the function allows one to gain information about the actions performed and the events occurred.

To use this package, users must install the R environment and establish an account with rights for job creation on the BOINC server. Connection to the BOINC server may use http, https or ssh protocols.

2.2. Virtual machines. This part must be specially prepared before running the compute tasks. We use the VirtualBox as a hypervisor.

- Firstly, a virtual machine must be bootable for IA32 and AMD64 processor architectures (one machine for each architecture). We recommend to install a Gentoo Linux on these machines due to space limits induced by necessity of

internet transmission, and because many R packages require a C++ headers and link libraries for system libraries when building. However, other Linux distributions will work as well.

- Secondly, R environment must be installed on these machines. We also recommend to install all the packages needed for computations beforehand, however, this is optional. If the required packages are not available on the virtual machine, the job will try to install them from the resources `rforge.net` or `cran.rstudio.org`.
- Thirdly, a regular user must be created. This user must automatically login into the system after boot.
- Fourthly, VirtualBox Guest Additions must be installed on this VMs. VirtualBox shared directory with the name `shared` must be mounted in the user home directory as `shared`. In the user home directory must exist a directory with the name `workdir` and full access. We recommend to move `workdir` to RAMFS.
- Finally, a shell script provided as a part of the package for virtual machines must be copied into the user home directories at VMs and added to the autoload.

When all these conditions are met, the virtual machine is ready to run the jobs.

2.3. Server part. It is a small set of scripts and programs that provide functions which are not available in BOINC. These functions are:

- File uploading to BOINC server via http and https protocols*.
- Unique name generation for uploaded files.
- Getting the state of job if the connection to the server uses `ssh` protocol.

Besides, the server part provides templates for BOINC applications and the so-called validator that may be used for any R jobs.

This part of package must be installed on BOINC server before it can run jobs from the client part of package. Installation requires copying the files to a BOINC project directory and editing the application templates. Virtual machines required for the application are not included in the server part and must be built manually.

The job creation starts on the package users (researchers) computers. R package collects all common files, generates the necessary code file and saves the tasks into files used by the BOINC server to create jobs. All obtained files are packed in a texttttar archive, compressed by the LZMA algorithm and uploaded to the BOINC server. Server part unpacks the archive, generates unique names for job files and registers these files in BOINC, returning these names back to the researcher's computer.

*In fact, BOINC server is able to download jobs files through these protocols, but in our case it was more convenient to pack all the necessary files into an archive, transfer it to the server, unpack it, resolve name conflicts, register the files in BOINC, and only then return the list of files back to the client.

Subsequently, requests are issued to the BOINC server to create a batch of jobs. The name for the batch is generated on the current date and time is always unique. The server returns the names of jobs in the batch upon a batch successful creation.

3. General Working Scheme

Volunteers install the standard BOINC client and configure it. BOINC client connects to the BOINC server and downloads the VirtualBox virtual machine with R and assigned job. The virtual machine is not downloaded every time a new job is received. Instead, it is downloaded once at the first connection to the server or if a new version of the application is available.

The virtual machine is booted and a job is started. Using the virtual machine it is allowed to provide running code isolation from the host that is good for security reasons. In addition, the job is independent on the software installed on the volunteer computer, such as the R interpreter, and only BOINC client, VirtualBox and VirtualBox Extension Pack are mandatory. The disadvantage, though, is a slight performance degradation.

Upon computation completion, the virtual machine is shut down. BOINC client uploads the result to the BOINC server where the result is processed and and validated it by the validator part provided by server part of package. Since every task generates only one job, assimilator just copies any valid job result to the results folder.

Users can update local status of jobs on their computers by the function provided by the package. Being called, this function sends messages to the BOINC server to get a list of complete jobs having results not yet downloaded. After that R downloads the results from the server and loads it into the users environment. As such, the package user observes the results as if the tasks were run on local machine.

4. Conclusion

We introduced a software package `RBOINC` for computing of embarrassingly parallel applications over BOINC environment using R language. We find this approach promising in the field of stochastic simulation and optimization, however, the package is rather general and allows various applications including bioinformatics, Monte-Carlo methods etc. As preliminary experiments show, the package allows seamless integration into the R environment. We plan to continue this research by using and enhancing the package for the sake of parallel stochastic simulation. In particular, it is planned to simplify the organization of iterations over the parameter space.

Acknowledgements

The authors thank anonymous referees for their comments that helped to improve the paper. We also thank Shane Conway who kindly allowed us to use the `RBOINC` name under R-forge workspace. This work is partially supported by the Russian Foundation for Basic Research, projects No. 19-57-45022, 19-07-00303.

REFERENCES

1. J.-M. Vincent, J. Vienne, L. Id-Imag, Perfect Simulation of Monotone Systems with Variance Reduction, in: Proceedings of the 6th International Workshop on Rare Event Simulation (RESIM 2006), Bamberg, Germany, 2006, pp. 275–285.
2. J.-M. Fourneau, F. Quessette, Monotone Queuing Networks and Time Parallel Simulation, in: Analytical and Stochastic Modeling Techniques and Applications. LNCS, Vol. 6751, Springer Berlin Heidelberg, Berlin, Heidelberg, 2011, pp. 204–218. doi:10.1007/978-3-642-21713-5_15.
3. J. M. Fourneau, F. Quessette, Tradeoff between Accuracy and Efficiency in the Time-Parallel Simulation of Monotone Systems, in: Computer Performance Engineering. LNCS, Vol. 7587, Springer Berlin Heidelberg, Berlin, Heidelberg, 2013, pp. 80–95. doi:10.1007/978-3-642-36781-6_6.
4. R. M. Fujimoto, Parallel discrete event simulation, Communications of the ACM 33 (10) (1990) 30–53. doi:10/frg5rh.
5. B. L. Fox, Integrating and accelerating tabu search, simulated annealing, and genetic algorithms, Annals of Operations Research 41 (2) (1993) 47–67. doi:10.1007/BF02022562.
6. H. Wickham, G. Grolemund, R for Data Science: Import, Tidy, Transform, Visualize, and Model Data, 1st Edition, O'Reilly Media, Inc., 2017.
7. A. Ebert, P. Wu, K. Mengersen, F. Ruggeri, Computationally efficient simulation of queues: The r package queuecomputer, Journal of Statistical Software, Articles 95 (5) (2020) 1–29. doi:10.18637/jss.v095.i05.
8. D. P. Anderson, BOINC: A System for Public-Resource Computing and Storage, in: Proceedings of the 5th IEEE/ACM International Workshop on Grid Computing, GRID '04, IEEE Computer Society, Washington, DC, USA, 2004, pp. 4–10. doi:10.1109/GRID.2004.14.
9. A. Rumyantsev, O. Sukhoroslov, A. Eparskaya, E. Blanzieri, V. Cavecchia, Parameter Sweep Experiments in Hybrid Computing Systems with R Language, International Journal of Innovative Technology and Exploring Engineering 8 (7S2) (2019) 590–596.

UDC: 004.932, 004.272

Binary gradient computation and implementation in reconfigurable computing environments

A.S. Bondarchuk[1], D.V. Shashev[1], S.V. Shidlovskiy[1]

[1]National Research Tomsk State University, 36 Lenin ave., 634050, Tomsk, Russia

bondarchuk.a.c@gmail.com

Abstract

The article outlines a new approach to constructing a feature vector for implementation on computers with a parallel pipeline architecture. The feature vector consists of the calculated characteristics of the gradient of a binary image (binary gradient) by analogy with the operation of the HOG algorithm. The proposed algorithm detects features of the contour pixels of objects in a binary image, which are further used for pattern recognition. After using the newly generated feature vectors for training a support vector machine (SVM) classifier, the speed of processing and classifying objects of interest on an image with a size of 1280 × 720 pixels increased by 3.5 times, in comparison with using the classical HOG descriptor.

Keywords: binary gradient, descriptor, reconfigurable computing environments

1. Introduction

In the field of creating automatic motion control systems for mobile platforms, the development of computer vision systems (CVS) has a great priority. To solve the problems of interaction with the environment, it is necessary to analyze the external situation in real time. CVS are used in modern underwater, surface, ground, aviation and space mobile robotic objects. For such platforms, the problem of reducing the time of image processing at high speeds of movement, as well as reducing power consumption, is urgent.

The solution to these problems is the implementation of image processing algorithms on parallel-pipelined computing architectures, such as Field-Programmable Gare Arrays (FPGA), Graphic Processing Unit (GPU), and Central Processing Unit (CPU) [1, 2]. Using FPGAs will allow you to achieve lower power consumption and higher performance through parallel computing.

Within the presented work, the use of a reconfigurable computing environments for processing and subsequent classification of binary images is considered with the implementation of the algorithm for calculating a binary gradient.

The reported study was funded by RFBR, project number 19-37-90110.

2. Finding the parameters of the image gradient and constructing the feature vector

A descriptor is an identifier of an image or image area, consisting of a set of features. Features are a descriptive element that characterizes an image. A feature vector is a numerical or binary vector of certain parameters. The type of parameters and the length of the vector depends on the algorithm used. Descriptors are used for pattern recognition and object detection in the image. One of these descriptors is the HOG (Histogram of Oriented Gradients) descriptor, commonly used for image processing and object detection in computer vision systems. The HOG descriptor is constructed by calculating the directions of the gradient in the local areas of the image. The main idea of the algorithm is the assumption that the appearance and shape of an object in the image can be described by the distribution of intensity gradients.

To extract the features of a binary image, it is proposed to use the algorithm for finding the gradient by analogy with the HOG algorithm. In this case, the binary gradient will be the value m and the direction ϕ of the change in the brightness of the neighboring image pixels from 0 to 1 or vice versa. The direction of the gradient can take three values: 180°, 225° and 270°. If there is no change in the brightness of pixels (between the current pixel and pixels of its neighbors), then $m = 0$, which means there is no gradient. In Fig. 1 shows 4 possible variants of the gradient values for the considered pixel I in relation to the neighboring pixels x and y.

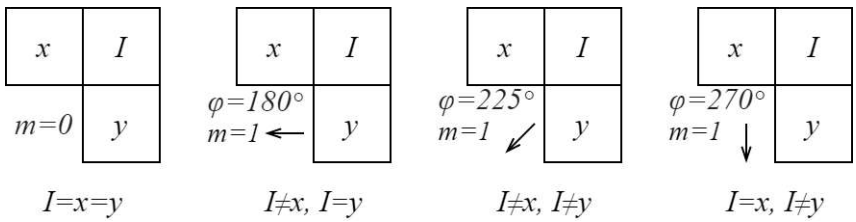


Fig. 1. Variants for the values of the binary gradient

Fig. 2 shows the result of determining the direction of the binary gradient, where Fig. 2a - initial binary image, Fig. 2b - visualization of a binary gradient in each pixel of the image.

The value of the direction of the gradient was encoded with a two-digit binary number $\phi = \phi_1\phi_2$ according to table 1. The feature vector, consisting of the values of m, ϕ_1, and ϕ_2 of the binary gradient of the image pixels, is further used to classify objects using the Support vector machine. Building a feature vector from the binary gradients of a binary image and multiplying these gradients by appropriate weights can be implemented using a reconfigurable computing environment.

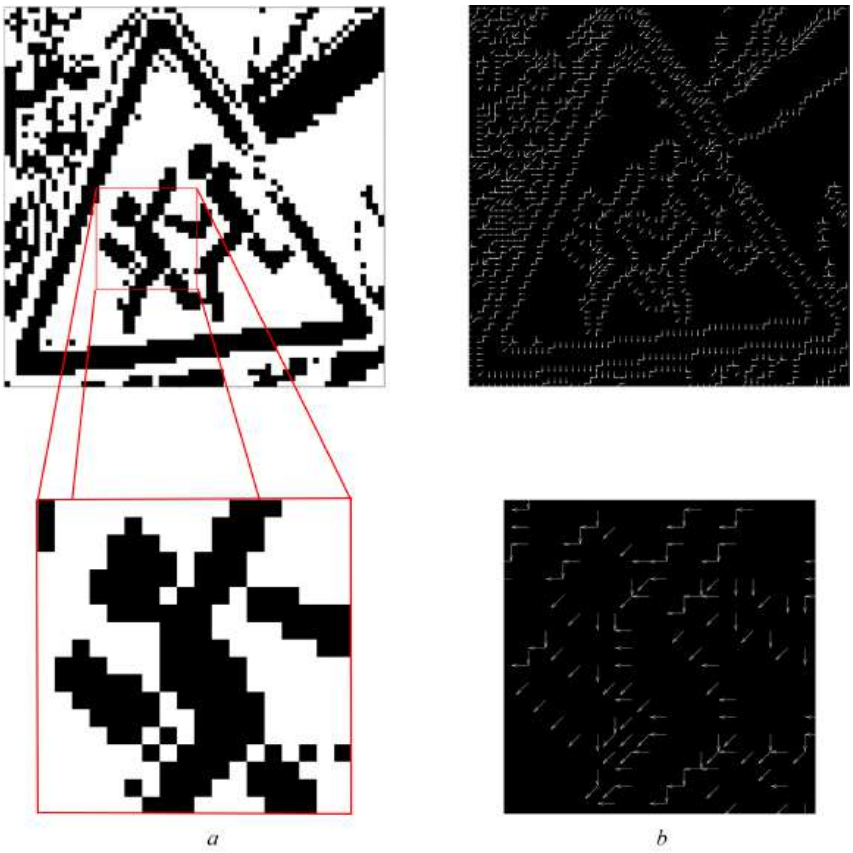


Fig. 2. Visualization of a binary gradient on the image

ϕ	≪no gradient≫	180°	225°	270°
ϕ_1	0	0	1	1
ϕ_2	0	1	0	1

Table 1. Encoding the directional values of the binary gradient

3. A reconfigurable computing environment (RCE)

RCE is a discrete mathematical model of a high-performance computing system, consisting of identical and equally connected to each other, the simplest universal elements (elementary calculators, ECs), programmatically tuned to perform any function from a complete set of logical functions, memory and any connection with its neighbours [3, 4, 5].

The fundamental principles of creating RCE are: parallelism, reconfigurability, homogeneity and pipelining of information processing. RCE has the form of a geometrically regular lattice, having at least two symmetry axes, with ECs located at the nodes, which contain a certain set of operations performed. A setting code is supplied to the input of each elementary calculator, with the help of which the reconfigurability of the RCE is carried out and it is determined which of the pledged operations will be performed. All elementary calculators are of the same type and are geometrically similarly connected with neighboring ones, and each of the ECs can be conventionally considered the center of symmetry with relation to its connections with the surrounding ECs. The cell of the elementary calculator has functional and connective completeness, i.e., can be configured to perform at a given moment any one function of at least one complete basis and function of the signal transmission channel in a given direction. The operations performed in the EC and the connections between them are intended to ensure the hardware execution of the algorithm being implemented.

The work of RCE can be considered from the point of view of the theory of automata. An automaton is called reconstructible if a set of automaton mappings implemented by it is given and an algorithm for tuning to implement each of these automaton mappings is defined [3]. An automatic mapping is an unambiguous mapping of the dependence of the output vector of the automaton on the vector of inputs, and tuning for the implementation of each of the automatic mappings is carried out by determining their tuning codes.

4. A model of ECs of the RCE for calculating binary gradients and their products by weight coefficients

The paper proposes an implementation of the previously described algorithm on the RCE architecture, where each EC is responsible for parallel processing of one of the pixels of a binary image. Thus, the dimension of the RCE coincides with the dimension of the processed image, while the elementary calculators are connected in the same way. Each automaton mapping was assigned a tuning code (z_4, z_3, z_2, z_1), at which the automaton is rebuilt to it. On the basis of the structural automaton method [3], we obtain the following system of equations (1), which will describe the work of the EC of the RCE:

$$\begin{cases} f_x{=}I(\bar{z}_1\bar{z}_2\bar{z}_3 + z_2z_4\bar{z}_1 + z_2z_3\bar{z}_1 + \bar{z}_2\bar{z}_3\bar{z}_4), \\ f_y{=}I(z_3z_4\bar{z}_1 + z_2\bar{z}_1\bar{z}_3 + z_2\bar{z}_1\bar{z}_4 + z_1\bar{z}_2\bar{z}_3\bar{z}_4), \\ W{=}(w_m(I \vee (I \cdot \bar{y} \vee \bar{I} \cdot y)) + w_{\phi 2}(I \cdot \overline{(I \cdot \bar{y} \vee \bar{I} \cdot y)} \vee \bar{I} \cdot (I \cdot \bar{y} \vee \bar{I} \cdot y)))(\bar{z}_1\bar{z}_2\bar{z}_3\bar{z}_4 + \\ +z_2z_3z_4\bar{z}_1) + w_{\phi 1}(I \cdot \bar{y} \vee \bar{I} \cdot y)(\bar{z}_1\bar{z}_2\bar{z}_3 + z_3\bar{z}_1\bar{z}_2 + z_3z_4\bar{z}_1 + \bar{z}_2\bar{z}_3\bar{z}_4) + (w_m((I \cdot \bar{x} \vee \\ \vee \bar{I} \cdot x) \vee (I \cdot \bar{y} \vee \bar{I} \cdot y)) + w_{\phi 2}((I \cdot \bar{x} \vee \bar{I} \cdot x) \cdot \overline{(I \cdot \bar{y} \vee \bar{I} \cdot y)} \vee \overline{(I \cdot \bar{x} \vee \bar{I} \cdot x)} \cdot (I \cdot \bar{y} \vee \\ \vee \bar{I} \cdot y)))(z_4\bar{z}_1\bar{z}_2 + z_3\bar{z}_1\bar{z}_2 + z_1\bar{z}_2\bar{z}_3\bar{z}_4) + (w_m((I \cdot \bar{x} \vee \bar{I} \cdot x) \vee I) + w_{\phi 2}((I \cdot \bar{x} \vee \\ \vee \bar{I} \cdot x) \cdot \bar{I} \vee \overline{(I \cdot \bar{x} \vee \bar{I} \cdot x)} \cdot I))z_2\bar{z}_1\bar{z}_3 + I(w_{\phi 1}(z_2\bar{z}_1\bar{z}_3 + z_2\bar{z}_1\bar{z}_4) + w_m z_2z_3\bar{z}_1\bar{z}_4). \end{cases} \quad (1)$$

where $I, x, y, w_m, w_{\phi 1}, w_{\phi 2}$ – information inputs; W – automaton output; f_x, f_y – outputs of inter-automaton connections. Thus, in the RCE model, simultaneous parallel pixel-by-pixel processing of a binary image is carried out and, for each pixel, the product of the binary gradient parameters by the corresponding weight coefficient is calculated. Subsequently, summing up the values of the array W, we obtain the result of the product of the feature vector by the vector of weights. Adding to the calculated number the parameter b, obtained during training using the SVM algorithm, we determine the class to which the considered image (or image region) belongs.

For training, 5459 images without an object of interest and 4000 images of vehicles with a size of 128 × 128 pixels obtained from the Berkeley Deep Drive 100K dataset were used. After that, the classifier was tested on 16 test images where the object of interest is present. On all these images, the location of the object was determined correctly, and the number of false positives was equal to 0. To compare the results, two more classifiers were trained. The first classifier uses HOG-descriptor feature vectors, and also correctly determines the object of interest on test images. The second classifier uses vectors consisting of sequential values of the magnitudes and directions of the HOG intensity gradients. As a result, the location of the object is determined correctly on 10 out of 16 test images. In 3 images, there are false positives along with the object, and in the remaining 3, the object was not detected.

The time of processing and recognition of vehicles was measured on an image with a size of 1280 × 720 pixels (Table 2).

Feature vectors	Binary	HOG descriptor	Magnitudes and directions of HOG intensity gradients
Processing time, sec	196.5	694	109

Table 2. Comparison of the processing speed of one image

The analysis of classification algorithms was carried out in MATLAB R2020a using a computer with the following characteristics:

- Intel(R) Core(TM) i9-9880H CPU @ 2.30GHz;
- 32 Gb RAM;
- NVIDIA Geforce RTX 2080.

5. Conclusion

The paper presents the results of constructing a RCE model for calculating and using the parameters of the gradient of a binary image in the problems of classifying objects in an image. The peculiarities of constructing the RCE architecture make it possible to implement an algorithm for finding a binary gradient in parallel processing of each pixel in 1 clock cycle of the EC operation. Using computer simulation methods in MATLAB R2020a, it was shown that an SVM classifier trained on binary feature vectors processes a 1280 $\times$ 720 pixel image 3.5 times faster than a classic HOG descriptor.

REFERENCES

1. Zhang L and Nevatia R 2008 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops (Anchorage, USA: IEEE) pp 1-7.
2. Hiromoto M and Miyamoto R 2009 IEEE 12th International Conference on Computer Vision Workshops (Kyoto, Japan: IEEE) pp 894-899.
3. Shidlovskiy S V 2016 MATEC Web of Conf. 79 01014.
4. Shashev D and Shidlovskiy S 2015 Opt. Instr. and Data Processing 51 19.
5. Khoroshevsky V G 2004 Int. Conf. on Parallel Computing in Electrical Engineering (Dresden, Germany: IEEE)

UDC: 519.87

Risks Ordering and Reliability of Some Applied Probability Systems

E.V. Bulinskaya[1]

[1]Lomonosov Moscow State University, Leninskie Gory 1, Moscow, Russia

ebulinsk@yandex.ru

Abstract

Risk management or decision making under uncertainty is an important research direction. For optimization of applied probability systems performance one needs an appropriate mathematical model. Discrete-time models became popular during the last decade not only because they can be used for approximation of the corresponding continuous-time ones. In many practical situations discrete-time models describe more precisely the real system functioning. We focus below on the so-called dual models taking the system reliability as criterium of its performance quality. In order to compare the models we study the influence of stochastic ordering of risks on systems reliability.

Keywords: Risk, Reliability, Stochastic orders, Applications

1. Introduction and model description

It is well-known that one has to choose an appropriate mathematical model in order to study such applied probability domains as queueing systems, inventory, insurance, finance, telecommunication, dams, population dynamics and others (see, e.g. [1]). Nowadays, discrete-time models are popular, since in many cases they describe more precisely the real situation. Moreover, they can be used for numerical investigation of continuous-time systems. In contrast to [2] we focus here on dual discrete-time models.

We suppose that capital (surplus, reserve) $\{R_n : n \geqslant 0\}$ has the form:

$$R_n = u - n + X_n, \quad n \in \mathbb{N}, \tag{1}$$

where $u \geqslant 0$ is the initial capital, n is the firm payment up to the n-th period, that means payment 1 per period, and $\{X_n : n \geqslant 0\}$ is the total gain during the n periods (with $X_0 \equiv 0$).

The publication has been prepared with the support of RFBR according to the research project No.20-01-00487.

Such a model can describe a life insurance company dealing with annuities (see, e.g., Cramér [3]) or pharmaceutical, oil, telecommunication, venture companies having a constant expenses flow and random gains X_n. Recently, a discrete-time dual model was treated by Palmowski et al. [4] who obtained an explicit expression of Parisian ruin probability. Continuous-time dual models were considered by many researchers, see, e.g., [5]. However, for the most part they were interested in dividends problems (see papers [6], [7], [8], [9] and references therein). Let us also mention the paper [10] investigating a dual model with tax payment.

Finite-time ruin probability was studied for the first time by Willmot [11] who obtained the explicit formulas using the generating functions. The compound binomial model was investigated by Li and Sendova [12], see also references therein.

Recall also that the non-ruin probability, that is, the probability of system non-destruction, measures the system reliability.

2. Notation

Assume that a company capital R_n at time n has the form (1) where the initial capital $u \in \mathbb{N}$. It is supposed that expenses each period are equal to 1. The total gain X_n up to time n is given by $X_n = \sum_{i=1}^{n} Y_i$, $n \geqslant 1$. Here Y_i, $i \geqslant 1$, are i.i.d. integer-valued random variables with finite mean $1 < E(Y_i) < \infty$, $F(x)$ is their distribution function and $p_k = P(Y_i = k)$, $k = 0, 1, 2, \ldots$.

Denote by T the ruin time for this discrete model, that is,

$$T = \inf\{n \geqslant 1 : R_n \leqslant 0\},$$

whereas $T = \infty$, if $R_n > 0$ for all $n \geqslant 1$. It is obvious from definition that $T \geqslant u$.

The finite-time ruin probability for the company with initial capital u is given by

$$\psi_F(u, t) = P(T < t | R_0 = u) = P_u(T < t).$$

In the same way we define the ultimate ruin probability as

$$\psi_F(u) = \lim_{t \to \infty} \psi_F(u, t).$$

3. Ruin probability and stochastic orders

3.1. Explicit form of ruin probability. Let $p(z)$ be the generating function of a random variable Y having the same distribution as all Y_i, $i \geqslant 1$, then

$$p(z) = Ez^Y = \sum_{i=0}^{\infty} p_i z^i.$$

Lemma 1. Equation

$$p(z) - z = 0 \tag{2}$$

has a unique solution z_F in the interval $[0, 1)$.

The proofs of almost all results are omitted due to the lack of space.

We are going to use the following statement proved in [4].

Theorem 1. The ruin probability as function of initial capital u is given by

$$\psi_F(u) = z_F^u. \tag{3}$$

3.2. Relationship between orders of random gains and ruin probability. Further on we suppose that the company has a choice of random gains. In other words, there exist two sequences Y_i and Z_i such that they are ordered

$$Y_i \prec Z_i \text{ for all } i \in \mathbb{N}.$$

Assume that Y_i, $i \geqslant 1$, have the distribution $p_k = P(Y = k)$, $k \geqslant 0$, and F is d.f. of Y. The ruin probability in this case is denoted by $\psi_F(u)$. In the same way Z_i, $i \geqslant 1$, have distribution $q_k = P(Z = k)$, and random variable Z has d.f. G, so the ruin probability is denoted by $\psi_G(u)$.

The generating function of Z is denoted by $q(z)$, analog of equation (2) is $q(z) - z = 0$.

Definitions of stochastic orders and their properties one can find in the books by Bulinskaya [13], Shaked and Shantikumar [14] and paper by Mulero et al. [15].

Definition 1. Y precedes Z in stochastic order ($Y <_{st} Z$), if and only if

$$F(x) \geqslant G(x) \text{ for any } x,$$

here $F(x)$ and $G(x)$ are the corresponding distribution functions.

Theorem 2. If $Y <_{st} Z$ then, for any $u > 0$,

$$z_F \geqslant z_G,$$
$$\psi_F(u) \geqslant \psi_G(u).$$

See Fig. 1 illustrating the situation.

Definition 2. Let random variables Y and Z have the same mean $EY = EZ$. Then Y precedes Z in the convex order ($Y <_{cx} Z$), if and only if

$$E[f(Y)] \leqslant E[f(Z)]$$

for all convex functions $f(x)$ for which the mentioned expectations exist.

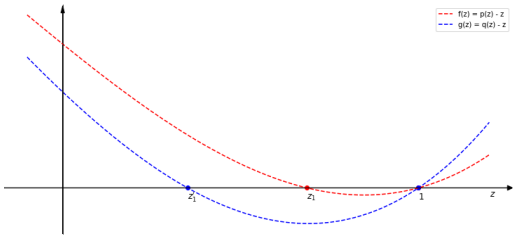


Fig. 1. Schematic layout.

Theorem 3. If $Y <_{cx} Z$ then, for any $u > 0$,

$$z_F \leqslant z_G,$$
$$\psi_F(u) \leqslant \psi_G(u).$$

Definition 3. Y precedes Z in the left-tail order ($Y <_{ltail} Z$), if and only if

$$E[YI(-\infty, a](Y)] \leqslant E[ZI(-\infty, a](Z)]$$

for any $a \in R$.

Theorem 4. Let $Y <_{ltail} Z$ and $p_0 \leqslant q_0$, moreover, the random variables take only finite number of values. Then

$$z_F \leqslant z_G,$$
$$\psi_F(u) \leqslant \psi_G(u)$$

for any $u > 0$.

Remark 1. If $p_0 > q_0$, then the statement of Theorem 4 is not correct. An example of such a case give random variables Y, Z with distributions provided by the Table 1 below.

Thus, $z_F = 0.06342453, z_G = 0.04297288$ and $z_F > z_G$. It follows immediately that $\psi_F(u) > \psi_G(u)$.

Definition 4. Y precedes Z in right-tail order ($Y <_{rtail} Z$), if and only if

$$E[YI[a, +\infty)(Y)] \leqslant E[ZI[a, +\infty)(Z)]$$

for any $a \in R$.

Theorem 5. Let $Y <_{rtail} Z$ and $EY + p_0 \geqslant EZ + q_0$, moreover, the random variables take only finite number of values. Then

Value	Probability Y	Probability Z
0	0.06	0.04
1	0.04	0.06
2	0.2	0.2
3	0.3	0.3
4	0.4	0.4

Table 1. Example

$$z_F \geqslant z_G,$$
$$\psi_F(u) \geqslant \psi_G(u)$$

for any $u > 0$.

3.3. Cooperation. Now suppose that the company participates in a project together with other companies carrying the quota α of expenses and acquiring the same quota of gains. Then the company capital is given by

$$\tilde{R}_n = u - \alpha n + \alpha X_n,\ n \in \mathbb{N}.$$

Its ruin probability has the form:

$$\tilde{\psi}_F(u) = \sum_{k=0}^{\infty} P_u(T = k),$$

where $T = \inf\{n \in \mathbb{N} : \tilde{R}_n \leqslant 0\}$.

Using the total probability formula, one gets

$$\tilde{\psi}_F(u + \alpha) = p_0 \tilde{\psi}_F(u) + \sum_{i=1}^{\infty} p_i \tilde{\psi}_F(u + \alpha i),$$

with restrictions $\tilde{\psi}_F(0) = 1$ and $\lim_{u\to\infty} \tilde{\psi}_F(u) = 0$. Thus, it is easy to prove

Theorem 6. The following relations hold

$$\tilde{z}_F = z_F^{\alpha},$$
$$\tilde{\psi}_F(u) = \psi_F^{\alpha}(u).$$

4. Conclusion

For evaluation of an applied system reliability we have to find its ruin probability. So it is interesting to be able to compare these probabilities on the base of stochastic

orders of systems characteristics such as their gains in the case of dual models. The results were established for 4 orders (stochastic dominance, convex order and left- and right-tail orders).

The next steps of investigation are consideration of Parisian ruin (instead of the usual one) thus increasing the system's solvability and study of dividends, investment, taxes and bank loans.

REFERENCES

1. Bulinskaya E. New research directions in modern actuarial sciences. // Modern problems of stochastic analysis and statistics – selected contributions in honor of Valentin Konakov (ed. V.Panov), Springer, 2017, P. 349–408.
2. Bulinskaya E. Asymptotic analysis and optimization of some insurance models. //Applied Stochastic Models in Business and Industry. 2018. V. 34. P. 762–773.
3. Cramer H. Collective risk theory: A survey of the theory from the point of view of the theory of stochastic process. Ab Nordiska Bokhandeln, Stockholm, 1955.
4. Palmowski Z., Ramsden L., Papaioannou A. D. Parisian ruin for the dual risk process in discrete-time. https://www.researchgate.net/publication/319256050
5. Yang C. On the dual risk model. https://ir.lib.uwo.ca/cgi/viewcontent.cgi?article=4659\&context=etd
6. Avanzi B., Gerber H. U., Shiu E. S. Optimal dividends in the dual model//Insurance: Mathematics and Economics. 2007. V. 41. P. 111–123.
7. Bergel A. I., Rodrigues-Martinez E. V., Egidio dos Reis A. D. On dividens in the phase-type dual risk model.//Scandinavian Actuarial Journal. 2016. P. 1–24.
8. Cheung E. C., Drekic S. Dividend moments in the dual risk model: exact and approximate approaches// Astin Bulletin. 2008. V. 38. P. 399–422.
9. Ng A. C. On a dual model with a dividend threshold// Insurance: Mathematics and Economics. 2009. V. 44. P. 315–324.
10. Albrecher H., Badescu A., Landriault D. On the dual risk model with tax payment// Insurance: Mathematics and Economics. 2008. V. 42. P. 1086–1094.
11. Willmot G. E. Ruin probabilities in the compound binomial model// Insurance: Mathematics and Economics. 1993. V. 12. P. 133–142.
12. Li S., Sendova K. P. The finite-time ruin probability under the compound binomial risk model. https://www.researchgate.net/publication/257803288
13. Bulinskaya E. V. Risk theory and reinsurance. Maylor, Moscow, 2009. (in Russian)
14. Shaked M., Shanthikumar L. G. (2007). Stochastic orders, 2007 https://www.springer.com/gp/book/9780387329154
15. Mulero J., Sordo M. A., de Souza M. C., Suares-Llorens A. Two stochastic dominance criteria based on tail comparisons// Applied Stochastic Models in Business and Industry. 2017. V. 33. P. 575–589.

UDC: 004.4:004.7

Survey of Load balancing mechanisms based on SDN in 5G/IMT-2020

Behrooz Daneshmand [1]

[1]ITMO University, Kronverkskiy Prospekt, 197101, saint peterburg, Russia

Daneshmandbehrooz@gmail.com

Abstract

The growing number of mobile devices and the demand for user data by 2030 are expected to put pressure on the current mobile network in an unprecedented way. Future mobile networks must have several requirements regarding data amount, latency, quality of service and experience, mobility, spectrum, and energy efficiency. Therefore, efforts have recently begun for more efficient mobile network solutions. To this end, load balancing has attracted much attention as a promising solution for greater resource utilization, improved system performance, and reduced operating costs. This is an effective way to balance traffic and reduce congestion in heterogeneous networks in future 5G/IMT-2020 networks. Load Balancing is one of the most critical tasks required to maximize network performance, scalability, and robustness. Nowadays, with the emergence of Software-Defined Networking (SDN), Load Balancing for SDN has become a significant issue in future network 5G/IMT-2020. SDN allows for programmable load balancers and provides the flexibility to design and implement load balancing strategies. In this survey, we highlight the methods of load balancing based on SDN networks and prospective load balancing requirements on 5G networks.

Keywords: Load Balancing , software defined networks, SDN , 5G/IMT-2020

1. Introduction

The fifth-generation network (5G / IMT-2020) was launched in 2018 in South Korea [1]. The goal of 5G is to provide high throughput, reduce latency, increase capacity. Demand for data traffic from mobile broadband users has been increasing over the past few years. 5G networks are designed to support mobile broadband connections and billions of M2M devices, and ultra-reliable low-latency communications devices in the future [2]. 5G base stations are optimized to support low latency for such devices. In addition, according to Cisco's annual Internet forecast, global Internet users will reach 5.3 billion in 2023. Moreover, Cisco predicts that there will

be 14.7 billion M2M connections by the year 2023. Therefore, operators must design efficient data processing in 5G, considering the potential for future growth in data usage and subscriber growth.

Existing networks' secure infrastructure and bone services lead to complex, inefficient resource allocation and low use of network resources, especially in wireless networks. Different load balancing techniques are based on the usefulness of a service provider and user satisfaction. SDN-based 5G network is another area of research for allocating resources and connecting to them on the 5G network. In this article, a survey on the load balancing of 5G integration with SDN is presented. Network Defined Software (SDN) is an emerging architecture that allows the physical segregation of the network control panel from the forwarding plane or infrastructure layer where the control panel controls multiple devices. Advantages offered by SDN include automated load balancing, demand provisioning, and the ability to scale network resources. In addition, SDN reduces hardware management and costs. Networks are provided without manual configuration. It provides companies with a platform to prepare for new technologies such as cloud-based applications, IoT devices, and big data applications.

Load balancing is a technique to divide the workload onto multiple resources to avoid overload on any resources [3]. The goals of load balancing are to maximize throughput, minimize response time, and optimize traffic. Essential and functional components in SDN are the OpenFlow protocol and controllers. The network controller is the brain of SDN architecture. It lies between network devices and applications. It is based on operating systems in computing. In [4], the controller is defined as a software abstraction that controls all functionalities of any networking system. It maintains control over the network through interfaces, first, southbound interface (e.g., OpenFlow), second, northbound interface (e.g., API). The southbound interface abstracts the functionalities of programmable switches and connects them to the controller. NOX is one of the first publicly available OpenFlow controller implementations in Windows, Linux, Mac OS, and other platforms. OpenFlow, the first leading authorized communications interface grafting the forwarding and controls layers of the SDN architecture, allows direct access and manipulation of forwarding planes on a network device such as virtual or physical switches and routers. In this paper, we compare and review the SDN-based load balancing algorithms performed by researchers and describe the advantages and disadvantages of each method. Load balancing is an essential component of fifth-generation network infrastructure due to the increasing number of Internet-connected devices where resources are distributed across a wide range of systems and require a subscription from an end-user base. The rest of the paper is organized as follows: After the section introduction, survey the architecture of SDN and overview of load balancing based on SDN and examine the

advantages and disadvantages of each method are provided in Section II, subsequently. In Section III, we describe several techniques for intelligent load balancing. Then we end the article with the conclusion in section IV

2. Architecture SDN and survey the Load Balancing based on SDN

2.1. SDN Architecture .

SDN is an entirely software configurable computer network in which the control levels of the network itself and data transmission are separated from each other by transferring control functions to a separate device - the network controller [3]. SDN is defined by the ability to dynamically control the behavior of a network using software through open interfaces. The main difference between SDN and conventional networks is the centralized intelligent network management and monitoring, which allows you to check, control and modify the transmitted data streams.

SDN technology aims to solve the following problems:

- Improving the efficiency of network bandwidth management mechanisms
- Reduction of capital costs and operating costs ;
- Simplification of network management and increasing its level of automation
- Acceleration and automation of the process of creating new services and their launch
- Increasing the security of the entire info-communication system ;
- Increasing the efficiency of routing

The SDN network infrastructure should be built in accordance with open protocols. OpenFlow, be unified and provide the ability to implement multi-vendor technical solutions. The architecture of SDN is divided into three primary layers, namely, an infrastructure layer (forwarding plane), a control layer (control plane), and an application layer, assembled over each other, as shown in Figure 1 [5].

2.2. Load Balancing based on SDN.

What is SDN load balancing? Software-defined networking SDN provides flexible control so enterprises can react to changing business requirements more quickly. Load balancing in SDN separates the physical network control plane from the data plane. An SDN-based load balancer allows for the control of multiple devices. This is how networks can become more agile. The network control can be programmed directly for more responsive and efficient application services. While computing and storage have seen innovations in virtualization and automation, networks have been lagging. Load balancing using SDN allows the network to function like the virtualized versions of computing and storage.

How does load balancing using SDN work? The SDN load balancing removes the protocols at the hardware level to allow for improved network management and

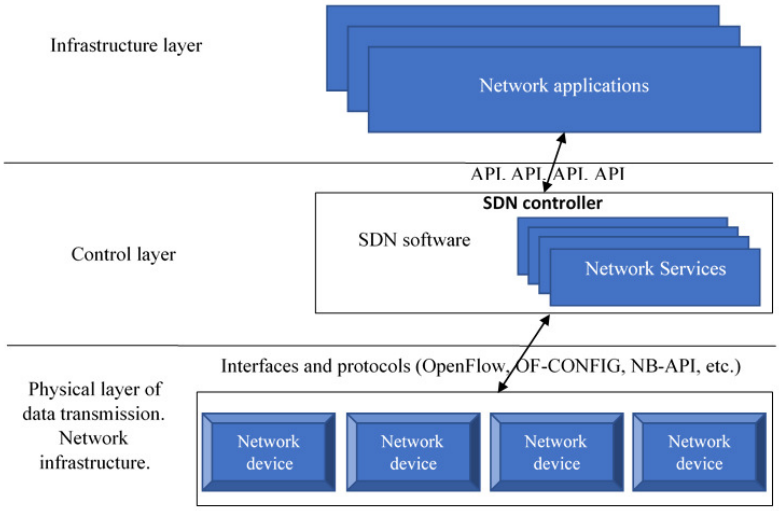


Fig. 1. SDN architecture

diagnosis. SDN controller load balancing makes data path control decisions without relying on algorithms defined by traditional network equipment. An SDN-based load balancer saves running time by controlling an entire network of applications and web servers.

Some numerous techniques and algorithms can be used to load balance client access requests across server pools intelligently. these methods can be static, dynamic (Distributed and Centralized), or a combination of both (hybrid), shown in figure 2 :

2.2.1. Static Load Balancing

In a static algorithm, traffic is evenly distributed between servers. The static algorithm is suitable for systems with low load changes. This algorithm must have prior information from system sources to ensure that the load switch decision is independent of the system's current state [6] However, static load balancing algorithms have the disadvantage of assigning tasks to the processor or machine only after creation, and tasks cannot be transferred to any other device for load balancing at runtime [7].

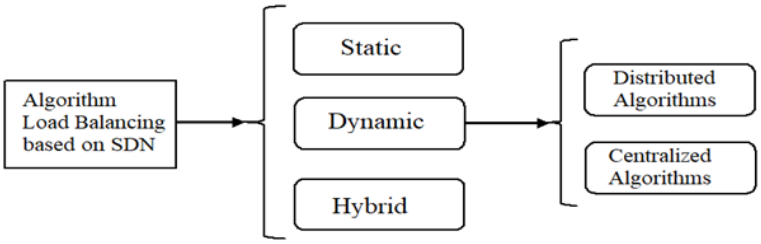


Fig. 2. Algorithm Load Balancing based on SDN

2.2.2. Dynamic Load Balancing

1. In the non-distributed method, one node (centralized) receives all requests and distributes them to the servers. Centralized controllers implement all of the control plane logic in one place. In such a controller, a single server takes over all actions at the control level. The main advantages of these controllers are simplicity and control, as they provide a single point of control. However, they suffer from scalability issues because each server has a limited capacity to handle data plane devices. Several distributed algorithms are referred to in the following sections:

- QoS-Aware Algorithm
- Heuristic Approaches
- Wardrop Load Balancing

2. In the distributed method, all nodes are shared with the distribution of the requests. The distributed controller has no scalability issues and has the advantage of high performance under heavy traffic loads. There are several centralized algorithms as mentioned in the following subsections:

- Routing Control Platform (RCP) ;
- Server-Based Load Balancing Algorithm ;
- DUTE Algorithm.

2.2.3. Hybrid Load Balancing

These methods are used to overcome the disadvantages of dynamic and static load balancing methods, and they are used to collect the advantages and disadvantages of static and dynamic algorithms to develop a new method [8]. This means that

combining the advantages of two or more existing algorithms, a dynamic or static algorithm, can create a new algorithm.

An essential and fundamental feature of static methods is having prior knowledge of the system. The rule is programmed directly into the load balancer in static methods because the user behavior is unpredictable. Static load balancing methods can be inefficient in a network. Dynamic methods are more efficient because the load is distributed dynamically according to some programmed patterns in the load balance [24]. Proper load balancing helps maximize scalability, minimize response time, maximize power consumption, minimize resource consumption, prevent overload of any single resource, and more. The essential qualitative parameters for load balance in SDN are presented in Table 1:

No	Load Balancing parameters	Description
1	An average number of synchronizations per minute	The average number of controller state synchronization per minute in SDN
2	Cumulative frequency	A performance index that supplies a measure of the accuracy of an algorithm
3	Degree of Load Balancing	A metric of uniformity of the load distribution among entities.
4	Energy Consumption	The amount of consumed energy in the network.
5	Execution Time	The length of time that a program is running.
6	Forwarding Entries	Routers use a forwarding table to decide to send the packet out.
7	Guaranteed Bit Rate (GBR)	One of the Quality-of-Service (QoS) parameters in networks for guaranteeing the bandwidth of the bearer
8	Latency	The time required to forward a packet across a network.
9	Migration Cost	It consists of two primary costs: the message exchanging cost and load cost. Some messages must be transmitted between the controllers for switching migration, such as migration requests, role requests, and asynchronous messages.
10	Overhead	Any composition of excessive or indirect computation time, memory, bandwidth, or other resources needed to carry out a particular task is overhead.
11	Packet Loss Rate	Packet loss occurs when one or more packets fail to reach their destination. This is usually due to network congestion. This is the percentage of packages lost compared to packages sent.
12	Peak Load Ratio	For route performance measurement.
13	Percentage of matched deadline flows	This parameter represents the percentage of flows satisfying the deadline.
14	Resource Utilization	The degree to which the network's resources are utilized, such as link, bandwidth, processor, and memory utilization.
15	Response Time	It is defined by the interval that starts from accepting a request or job to responding to a request or task for the server.
16	Root Mean Squared Error (RMSE)	A metric for assessing load balancing performance. Better performance has a smaller RMSE.
17	Throughput	The quantity of data correctly moved from one place to another during a specific period
18	Workload	The amount of work to be done by the controller. In order to balance the workloads among controllers, load balancing approaches have been introduced

Table 1. Load Balancing parameters

Load balancing can be implemented in software or physical equipment. The technique chosen will depend on the type of service or application being served and the status of the network and servers at the time of the request. These methods will

be used in combination to determine the best server to service new requests. The current level of requests to the load balancers often determines which method is used. When the load is low, then one of the simple load balancing methods will suffice. In times of high load, more complex methods ensure an even distribution of requests [10,11,12]. Load balancing in SDN leads to discovering the best pathway and server for the fastest delivery of requests.

SDN load balancing includes the following benefits:

- Lower cost
- Greater scalability
- Higher reliability
- Flexibility in configuration
- Reduced time to deploy
- Automation
- Ability to build a network without any vendor-specific software/hardware .

In order to optimize network flow and achieve intelligent load balance, various techniques and algorithms have been used .We mentioned the types of current SDN-based load balances methods which are using in fifth-generation networks are described below :

In order to optimize network flow and achieve intelligent load balance, various techniques and algorithms have been used. We mentioned the types of current SDN-based load balances methods used in fifth-generation networks, summarized and described below:

- SDN multi-Controllers Load Balancing

A single controller represents the bottleneck problem. Using a centralized controller may limit the scalability and reliability, while decentralized controllers in SDN networks perform better. Thus, employing multi controllers will serve as a vital solution to improve the control plane's scalability, reliability, and capability. In order to increase the performance of load balancing, it is essential to use multi SDN controllers instead of centralized controllers [13,14,15].

- Server Load Balancing (SLB)

The Server Load Balancing (SLB) deploys one load balancer in front of the multi servers. Thus it can reasonably allocate the load to several servers to make full use of the server resource. The authors propose in [12] a load balancing strategy for distributing clients' requests across multiple servers. Server Load Balancing provides network services and content delivery using a series of load balancing algorithms. IT teams are increasingly relying on server load balancers to, here we mentioned some advantages of Server Load Balancing (SLB):

- Increase Scalability
- Redundancy

• Maintenance and Performance
• Different links selection Load Balancing

The system actively monitors the quality in terms of latency, packet loss and jitter, and the capacity in terms of the throughput of the links and steers traffic to the most appropriate link based on the business intent defined as a policy. This technique increases the scalability and effective utilization of network resources. In [16], the authors proposed a Dynamic Load-balancer Path Optimization (DLPO) algorithm based on SDN, useful for data center network topologies. The multi-link algorithm can quickly balance link loads in a network to eliminate congested paths [16,17,18].

• Load-balancing based on Artificial Neural Networks

Artificial Neural Networks (ANN) predicts the demand and thus allocates resources according to that demand. Thus, it always maintains the active servers according to current demand, resulting in lower energy consumption than the conservative approach of over-provisioning. Furthermore, high utilization of servers results in more power consumption, server running at higher utilization can process more workload with similar power usage. The pickup of the Artificial Neural Network (ANN) method based on SDN in Load Balancing is because this method is capable of categorizing and select input data into specified groups or predetermined paths with the primary role in load balancing [19,20,21,22].

• Load Balancing in IEEE 802.11 standard (Wireless Links)

The IEEE 802.11 standard specifies that the client device decides which access point to connect to. In high-density environments, the client device's choice to connect to one AP can lead to an AP overload [23]. It might also lead to oscillations in the AP association as a client device has limited network performance data. Algorithms deployed on the SDN controller can dynamically balance access points (APs) load by choosing the less loaded ones from the sharing locality for the client's association.

3. Investigation of load balancing techniques

Several methods and algorithms can be applied using several queuing methods in load balancing techniques on the network. The technique chosen will depend on the type of service or application being served and the status of the network and servers at the time of the request. The methods outlined below will be used in combination to determine the best server to service new requests. The current level of requests to the load balancers often determines which method is used. When the load is low, then one of the simple load balancing methods will suffice. In times of high load, more complex methods are used to ensure an even distribution of requests. Software load balancing (SLB) is typically offered as the application delivery controller (ADC)

that runs on a standard server or a virtual machine. A hardware load balancing device (HLD) is a stand-alone piece of hardware that runs load balancing software. It is traditionally deployed as part of a pair in case one load balancing device fails. Software load balancing offers the same functionality as an HLD, but it does not require a dedicated load-balancing device. The load-balancing software can run on a regular server or even a virtual server. In Table 2, we mentioned the advantages and disadvantages of each method

SLB (SOFTWARE LOAD BALANCING)	
SOFTWARE PROS	SOFTWARE CONS
• Flexible Implementation • Greater Scalability for Software Load Balancing • Reduced Cost • Load Balancing Via Cloud	• When scaling beyond initial capacity, there can be some delay while configuring load balancer software. • Ongoing costs for upgrades.
HLB (HARDWARE LOAD BALANCING)	
HARDWARE PROS	HARDWARE CONS
• Fast throughput due to software running on specialized processors. • Increased security since only the organization can access the servers • physically. • Fixed cost once purchased.	• Requires more staff and expertise to configure and program the physical machines. • Inability to scale when the set limit on the number of connections has been made. Connections are refused, or service degraded until additional machines are purchased and installed. • Higher cost for purchase and maintenance of physical network load balancer. Owning a hardware load balancer may also require paying consultants to manage it.

Table 2. Characteristics of SLB and HLB

Load balance occupies a vital position to solve the problem of excessive traffic in the network. It has been one of the first attractive applications on SDN networks. Typically, the following methods are used as a combination to find the best server to respond to new requests. In this section, we refer to some of the commonly used load balancing approaches, summarized in table 3.

4. Conclusion

This paper provides an overview of the load balancing mechanism in SDN. The SDN-based solutions are used to provide load balancing in networks 5G/IMT-2020 networks. The SDN architecture and methods of load balancing that classify into three groups, dynamic, static, and hybrid, were investigated. The essential qualitative parameters of load balancing in SDN are mentioned in this article. We endeavored to review the types of SDN-based Load Balancing techniques which are using in Load Balancing. In addition, we compared two techniques SLB and HSL and investigated the advantages and disadvantages of both. Based on the results of our previous sections, SDN-based load balancing mechanisms provide an overview of the network,

No	Load Balancing Approaches	Description
1	Round Robin (RR)	Round robin load balancing is a simple way to distribute client requests across a group of servers. A client request is forwarded to each server in turn.
2	Weighted Round Robin (WRR)	Weighted Round Robin builds on the simple Round-robin load balancing algorithm to account for differing application server characteristics. The administrator assigns a weight to each application server based on criteria of their choosing to demonstrate the application servers traffic-handling capability.
3	Least Connection / Loaded (LL)	Least Connection load balancing is a dynamic load balancing algorithm where client requests are distributed to the application server with the least number of active connections at the time the client request is received. In cases where application servers have similar specifications, an application server may be overloaded due to longer lived connections; this algorithm takes the active connection load into consideration
4	Weighted Least Connection	Weighted Least Connection builds on the Least Connection load balancing algorithm to account for differing application server characteristics. The administrator assigns a weight to each application server based on criteria of their choosing to demonstrate the application servers traffic-handling capability.
5	Software Defined Networking (SDN) Adaptive	SDN Adaptive is a load balancing algorithm that combines knowledge from Layers 2, 3, 4 and 7 and input from an SDN Controller to make more optimized traffic distribution decisions. This allows information about the status of the servers, the status of the applications running on them, the health of the network infrastructure, and the level of congestion on the network to all play a part in the load balancing decision making.
6	Fixed Weighting	Fixed Weighting is a load balancing algorithm where the administrator assigns a weight to each application server based on criteria of their choosing to demonstrate the application servers traffic-handling capability. The application server with the highest weigh will receive all of the traffic. If the application server with the highest weight fails, all traffic will be directed to the next highest weight application server
7	Weighted Response Time	Weighted Response Time is a load balancing algorithm where the response times of the application servers determines which application server receives the next request. The application server response time to a health check is used to calculate the application server weights. The application server that is responding the fastest receives the next request .
8	Source IP Hash	Source IP hash load balancing algorithm that combines source and destination IP addresses of the client and server to generate a unique hash key. The key is used to allocate the client to a particular server. As the key can be regenerated if the session is broken, the client request is directed to the same server it was using previously. This is useful if it's important that a client should connect to a session that is still active after a disconnection.
9	Random	As its name implies, this algorithm matches clients and servers by random, i.e. using an underlying random number generator. In cases wherein the load balancer receives a large number of requests, a Random algorithm will be able to distribute the requests evenly to the nodes. So like Round Robin, the Random algorithm is sufficient for clusters consisting of nodes with similar configurations (CPU, RAM, etc).

Table 3. Load Balancing Technique

so these methods typically improve system performance compared to traditional load balancing approaches. In the following works, we will focus on the practical implementation and comparison of various balancing methods in the field of the 5G network so that we can achieve a better load balancing method in the next generation networks (5G) by comparing and combining different algorithms.

REFERENCES

1. Bega, D, Gramaglia, M, Bernardos Cano, CJ, Banchs, A, Costa-Perez, X. Toward the network of the future: From enabling technologies to 5G concepts. Trans Emerging Tel Tech. 2017; 28:e3205

2. N. Subburayalu, S. Natarajan and D. Das, "Dynamic Load Balancing across Multi-radio Access Bearers in 5G," 11th International Conference on Communication Systems Networks (COMSNETS), Bengaluru, India, 2019, pp. 306-311.
3. A. A. Neghabi, N. Jafari Navimipour, M. Hosseinzadeh and A. Rezaee, "Load Balancing Mechanisms in the Software Defined Networks: A Systematic and Comprehensive Review of the Literature," in IEEE Access, vol. 6, pp. 14159-14178, 2018.
4. Nam Tuan Le, Mohammad Arif Hossain, Amirul Islam, Do-yun Kim, Young-June Choi, Yeong Min Jang, "Survey of Promising Technologies for 5G Networks", Mobile Information Systems, vol. 2016, Article ID 2676589, 25 pages, 2016.
5. Open Networking Foundation (ONF),https://opennetworking.org.
6. Shah, Nadeem and M. Farik. "Static Load Balancing Algorithms In Cloud Computing: Challenges Solutions." International Journal of Scientific Technology Research 4 (2015): 365-367
7. Amandeep, V. Yadav and F. Mohammad, "Different Strategies for Load Balancing in Cloud Computing Environment: A Critical Study," vol. 3, issue. 1, pp. 85-90, 2014.
8. A. S. Milani and N. Jafari, "Load balancing mechanisms and techniques in the cloud environments : Systematic literature review and future trends," J. Netw. Comput. Appl., vol. 71, pp. 86–98, 2016.
9. S. Chen, Y. Chen, and S. Kuo, "CLB: A novel load balancing architecture and algorithm for cloud services," Comput. Electr. Eng., vol. 58, pp. 154–160, 2017.
10. A. A. Neghabi, N. Jafari Navimipour, M. Hosseinzadeh and A. Rezaee, "Load Balancing Mechanisms in the Software Defined Networks: A Systematic and Comprehensive Review of the Literature," in IEEE Access, vol. 6, pp. 14159-14178, 2018.
11. Mustafa Hasan Al.B, Nurul A.Z, Z.Zainal Abidin "Load balancing algorithms in software defined network," International Journal of Recent Technology and Engineering (IJRTE) , Volume-7 Issue-6S5, April 2019 , pp.686-692.
12. Semong, Thabo; Maupong, Thabiso; Anokye, Stephen; Kehulakae, Kefalotse; Dimakatso, Setso; Boipelo, Gabanthone; Sarefo, Seth. 2020. "Intelligent Load Balancing Techniques in Software Defined Networks: A Survey" Electronics 9, no. 7: 1091.
13. T. Hu, P. Yi, J. Zhang and J. Lan, "Reliable and load balance-aware multi-controller deployment in SDN," in China Communications, vol. 15, no. 11, pp. 184-198, Nov. 2018
14. K. Sridev , M. A. Saifulla , "Multi Controller Load Balancing in Software Defined Networks: A Survey," Advances in Decision Sciences, Image Processing, Security

and Computer Vision ,pp 417-425 . Learning and Analytics in Intelligent Systems, vol 3. Springer, Cham.
15. Ma, YW., Chen, JL., Tsai, YH. et al. Load-Balancing Multiple Controllers Mechanism for Software-Defined Networking. Wireless Pers Commun 94, 3549–3574 (2017).
16. Y. Lan, K. Wang and Y. Hsu, "Dynamic load-balanced path optimization in SDN-based data center networks," 2016 10th International Symposium on Communication Systems, Networks and Digital Signal Processing (CSNDSP), Prague, Czech Republic, 2016, pp. 1-6
17. C. Hopps. 2000. RFC2992: Analysis of an Equal-Cost Multi-Path Algorithm. RFC Editor, USA.
18. J. Li, X. Chang, Y. Ren, Z. Zhang and G. Wang, "An Effective Path Load Balancing Mechanism Based on SDN," 2014 IEEE 13th International Conference on Trust, Security and Privacy in Computing and Communications, Beijing, China, 2014, pp. 527-533.
19. Cui Chen-xiao, and Xu Ya-bin, " Research on Load Balance Method in SDN," International Journal of Grid and Distributed Computing Vol. 9, No. 1 (2016), pp.25-36
20. Andika Malraherawan Pradana, Tito Waluyo Purboyo and Roswan Latuconsina , "A simulation of load balancing in software defined network (SDN) based on Artificial Neural Networks method," ARPN Journal of Engineering and Applied Sciences , March 2020 ,Vol. 15 No. 6
21. Alex M. R. Ruelas and Christian Esteve Rothenberg. 2018. A Load Balancing Method based on Artificial Neural Networks for Knowledge-defined Data Center Networking. In Proceedings of the 10th Latin America Networking Conference (LANC '18). Association for Computing Machinery, New York, NY, USA, 106–109.
22. Z. Li, X. Zhou, J. Gao and Y. Qin, "SDN Controller Load Balancing Based on Reinforcement Learning," 2018 IEEE 9th International Conference on Software Engineering and Service Science (ICSESS), Beijing, China, 2018, pp. 1120-1126.
23. Yen, L.H.; Yeh, T.T.; Chi, K.H. Load balancing in IEEE 802.11 networks. IEEE Internet Comput. 2009, 13, 56–64
24. Ghosh, Anish, Mrs. T. Manoranjitham. "A study on load balancing techniques in SDN." International Journal of Engineering Technology [Online], 7.2.4 (2018): 174-177.

УДК: 519.23

Многолинейная система с разнотипными ненадежными приборами и повторными вызовами

А.Н. Дудин[1,2] and Лю Мэй[1]

[1]Факультет прикладной математики и информатики, Белорусский государственный университет, проспект Независимости, 4, Минск, Беларусь
[2]Институт прикладной математики и телекоммуникаций, Российский университет дружбы народов, ул. Миклухо-Маклая, 6, Москва, Россия

Аннотация

Исследуется многолинейная система массового обслуживания с разнотипными ненадежными приборами и марковским входным потоком. Приборы перенумерованы в порядке убывания скорости обслуживания и входящий запрос всегда занимает свободный исправный прибор с минимальным номером. Если свободных исправных приборов нет, запрос идет на орбиту, с которой совершает попытки попасть на обслуживание. Поведение системы описывается многомерной цепью Маркова с одной счетной компонентой. Выписан генератор цепи, что позволяет найти стационарное распределение вероятностей состояний системы и основные характеристики ее производительности.

Ключевые слова: марковский входной поток, разнотипные ненадежные приборы, повторные вызовы

1. Введение

В данной работе исследуется многолинейная система массового обслуживания (СМО) с разнотипными ненадежными приборами, повторными вызовами и марковским входным потоком запросов. Многолинейные СМО с повторными вызовами являются сложным объектом для исследования. При этом практически все известные работы по таким системам предполагают, что приборы являются идентичными и при прибытии запроса извне или при повторной попытке запрос с равной вероятностью занимает любой свободный прибор. Одним из редких исключений являются так называемые системы с адресными повторами, см. например, [1], [2], в которых приборы являются однородными, но при совершении попытки попасть на обслуживание запрос выбирает с определенной

Публикация выполнена при поддержке Программы стратегического академического лидерства РУДН и проекта 1.6.01.2 ГПНИ «Конвергенция-2025» на 2021-2025 годы

вероятностью некоторой конкретный прибор и при его занятости он идет (или возвращается на орбиту). Наиболее близкая к рассмотренной в данной работе модель была изучена в [3]. Это - многолинейная СМО с разнотипными приборами, повторными вызовами и марковским входным потоком запросов. В данной работе рассмотрено обобщение этой системы на случай ненадежных приборов.

2. Математическая модель

Мы рассматриваем N-линейную СМО Входной поток описывается как MAP (Markov Arrival Process). Прибытие запросов в MAP происходит под управлением неприводимой цепи Маркова ν_t, $t \geq 0$, с конечным пространством состояний $\{0, 1, ..., W\}$. Время пребывания цепи ν_t, $t \geq 0$, в состоянии ν имеет показательное распределение с параметром $\lambda_\nu, \nu = \overline{0, W}$. По истечении этого времени с вероятностью $p_k(\nu, \nu')$ процесс ν_t переходит в состояние ν', и k запросов, $k = 0, 1$, поступают в систему. Интенсивности перехода цепи Маркова из одного состояния в другое с генерацией k запросов объединяются в матрицы D_k, $k = 0, 1$, размера $\overline{W} \times \overline{W}$, где $\overline{W} = W + 1$. Матрица $D(1) = D_0 + D_1$ является инфинитезимальным генератором процесса ν_t, $t \geq 0$. Вектор стационарного распределения вероятностей $\boldsymbol{\theta}$ этого процесса вычисляется как единственное решение системы $\boldsymbol{\theta} D(1) = \mathbf{0}$, $\boldsymbol{\theta}\mathbf{e} = 1$. Здесь и далее $\mathbf{0}$ – нулевая вектор-строка, а $\mathbf{e}$ – единичный вектор-столбец соответствующего размера. Интенсивность потока λ определена как $\lambda = \boldsymbol{\theta} D_1 \mathbf{e}$. Более подробная информация о MAP потоке может быть найдена, например, в [4]. Важным свойством MAP потока является то, что он позволяет учитывать возможность зависимости длин интервалов между моментами поступления и величину дисперсии этих интервалов, что актуально при моделировании современных телекоммуникационных систем и сетей.

Обслуживающие приборы системы являются разнородными. Время обслуживания запроса n-м прибором имеет экспоненциальное распределение с параметром μ_n, $n = \overline{1, N}$. Для определенности, без ограничения общности, предположим, что приборы пронумерованы таким образом, что $\mu_1 \geq \mu_2 \geq \cdots \geq \mu_N$.

Поступивший в систему запрос выбирает для своего обслуживания свободный прибор с наименьшим номером. Если такой прибор существует, запрос немедленно начинает обслуживания. Если свободных приборов нет, то запрос уходит на так называемую орбиту, с которой пытается попасть на обслуживание (снова на свободный прибор с наименьшим номером) через случайные промежутки времени. Эти промежутки, при фиксированном числе i запросов на орбите, имеют экспоненциальное распределение в параметром α_i, $i > 0$. Мы не делаем никаких предположений о зависимости интенсивности повторов α_i от числа запросов на орбите $i > 0$, за исключением предположения, что α_i стремится к бесконечности при i, стремящемся к бесконечности. Как очень частный

случай можно рассмотреть так называемую классическую стратегию повторов, при которой $\alpha_i = i\alpha,\ i > 0$, где α трактуется как индивидуальная интенсивность повторов запроса с орбиты, см., например, [5].

Занятый n-й прибор может выходить из строя. Время до выхода его из строя имеет экспоненциальное распределение с параметром $\varphi_n,\ n = \overline{1, N}$. Для определенности считаем, что запрос, во время обслуживания которого прибором произошла поломка прибора, теряется. Другие варианты (например, запрос перемещается в другой свободный и исправный прибор, если таковой имеется, и продолжает обслуживание или запрос при прерывании обслуживания уходит на орбиту для совершения повторных попыток), могут быть проанализированы путем соответствующей модификации генератора цепи Маркова, описывющей динамику системы. Прибор с номером n, вышедший из строя, восстанавливается в течение времени, имеющего экспоненциальное распределение с параметром $\gamma_n,\ n = \overline{1, N}$.

Нашей целью является проанализировать стационарное поведение описанной модели СМО.

3. Процесс изменения состояний системы и его анализ

Пусть в произвольный момент времени $t,\ t \geq 0$,

- i_t – число запросов на орбите, $i_t \geq 0$;
- $r_t^{(n)}$ – текущее состояние n-го прибора: $r_t^{(n)} = 0$, если n-й прибор свободен, $r_t^{(n)} = 1$, если n-й прибор производит обслуживание запроса, $r_t^{(n)} = 2$, если n-й прибор находится на ремонте, $n = \overline{1, N}$;
- ν_t – состояние управляющего процесса MAP потока, $\nu_t = \overline{0, W}$.

Обозначим $\mathbf{r}_t = \{r_t^{(1)}, r_t^{(2)}, \ldots, r_t^{(N)}\},\ t \geq 0$, векторный процесс, задающий текущие состояния приборов системы. Компонентами этого процесса являются числа 0,1, 2. Пространство состояний этого процесса состоит из $\mathcal{N}$ элементов, где $\mathcal{N} = 3^N$. Будем считать, что состояния процесса $\mathbf{r}_t$ упорядочены в лексикографическом порядке. При такой нумерации состояние $\{r_1, r_2, \ldots, r_N\},\ r_n = 0, 1, 2,\ n = \overline{1, N}$, имеет порядковый номер

$$\sum_{n=1}^{N} r_n 3^{N-n} + 1.$$

Нетрудно заметить, что многомерный случайный процесс $\xi_t = \{i_t,\ \mathbf{r}_t,\ \nu_t\}$, $t \geq 0$, является неприводимой цепью Маркова с непрерывным временем с одной счетной компонентой i_t и несколькими конечными компонентами.

Для анализа цепи Маркова ξ_t необходимо выписать ее инфинитезимальный генератор. Обозначим этот генератор $\mathbf{Q}$. Диагональные элементы генератора являются отрицательными. Их модули задают интенсивности выхода цепи Маркова из соответствующего состояния. Недиагональные элементы являются неотрицательными и определяют интенсивности соответствующих переходов цепи Маркова в ее пространстве состояний.

Введем в рассмотрение следующие вспомогательные матрицы размера $\mathcal{N}$:

$\mathbf{J}$ является диагональной матрицей с диагональными элементами отличными от 0 и равными 1 в строках с номерами $\sum_{n=1}^{N} r_n 3^{N-n} + 1$, такими, что для всех значений l, $l = \overline{1, N}$, выполняются неравенства $r_l \geq 1$, т.е., среди значений компонент $r_l \geq 1$, нет нулевых. Это означает, что при соответствующем состоянии процесса $\mathbf{r}_t$ в системе нет свободных исправных приборов;

$\mathbf{E}^+$ есть матрица с нулевыми элементами $(\mathbf{E}^+)_{r,r'}$, $r, r' = \overline{1, \mathcal{N}}$, за исключением равных 1 элементов $(\mathbf{E}^+)_{r,r'}$, таких, что

$$r = \sum_{n=1}^{q-1} r_n 3^{N-n} + \sum_{n=q+1}^{N} r_n 3^{N-n} + 1,\ \ r' = r + 3^{N-q},$$

где $q = \arg\min_q \{r_q = 0\}$;

$\mathbf{E}^-$ есть матрица с нулевыми элементами $(\mathbf{E}^-)_{r,r'}$, $r, r' = \overline{1, \mathcal{N}}$, за исключением равных μ_l элементов $(\mathbf{E}^-)_{r,r'}$, таких, что

$$r = \sum_{n=1}^{N} r_n 3^{N-n} + 1,\ \ r' = r - 3^{N-l},$$

где $l = \arg\{r_l = 1\}$;

$\mathbf{E}^{\varphi}$ есть матрица с нулевыми элементами $(\mathbf{E}^{\varphi})_{r,r'}$, $r, r' = \overline{1, \mathcal{N}}$, за исключением равных φ_l элементов $(\mathbf{E}^{\varphi})_{r,r'}$, таких, что

$$r = \sum_{n=1}^{N} r_n 3^{N-n} + 1,\ \ r' = r + 3^{N-l},$$

где $l = \arg\{r_l = 1\}$;

$\mathbf{E}^{\gamma}$ есть матрица с нулевыми элементами $(\mathbf{E}^{\gamma})_{r,r'},\ r,r'=\overline{1,\mathcal{N}}$, за исключением равных γ_l элементов $(\mathbf{E}^{\gamma})_{r,r'}$, таких, что

$$r=\sum_{n=1}^{N}r_n3^{N-n}+1,\ \ r'=r-2\times 3^{N-l},$$

где $l=\arg\{r_l=2\}$;

$\mathbf{E}^0$ есть диагональная матрица с диагональными элементами $(\mathbf{E}^0)_{r,r},\ r=\overline{1,\mathcal{N}}$, заданными формулами

$$(\mathbf{E}^0)_{r,r}=-\sum_{n=1}^{N}\delta_n,\ r=\sum_{n=1}^{N}r_n3^{N-n}+1,$$

где $\delta_n=0$, если $r_n=0,\ \delta_n=\mu_n+\varphi_n$, если $r_n=1,\ \delta_n=\gamma_n$, если $r_n=2$.

Лемма. Генератор $\mathbf{Q}$ цепи Маркова ζ_t имеет блочную трехдиагональную структуру:

$$\mathbf{Q}=\begin{pmatrix}\mathbf{Q}_{0,0}&\mathbf{Q}_{0,1}&O&O&\dots\\ \mathbf{Q}_{1,0}&\mathbf{Q}_{1,1}&\mathbf{Q}_{1,2}&O&\dots\\ O&\mathbf{Q}_{2,1}&\mathbf{Q}_{2,2}&\mathbf{Q}_{2,3}&\dots\\ \vdots&\vdots&\vdots&\vdots&\ddots\end{pmatrix},$$

где ненулевые блоки $\mathbf{Q}_{i,j},\ j=\max\{0,i-1\},i,i+1$, задаются следующим образом:

$$\mathbf{Q}_{i,i+1}=\mathbf{J}\otimes D_1,\ i\geq 0,\ \ \mathbf{Q}_{i,i-1}=\alpha_i\mathbf{E}^{+}\otimes I_{\bar{W}},\ i\geq 1,$$

$$\mathbf{Q}_{i,i}=I_{\mathcal{N}}\otimes D_0+\mathbf{E}^{+}\otimes D_1-\alpha_i(I-\mathbf{J})\otimes I_{\bar{W}}+\mathbf{E}^{-}\otimes I_{\bar{W}}+\mathbf{E}^{\varphi}\otimes I_{\bar{W}}+$$
$$\mathbf{E}^{\gamma}\otimes I_{\bar{W}}+\mathbf{E}^{0}\otimes I_{\bar{W}},\ i\geq 0.$$

Здесь, I – единичная матрица, и O – нулевая матрица соответствующего размера, $\otimes$ – символ Кронекерова произведения матриц, см. [6].

Доказательство леммы достаточно очевидно. Увеличение числа запросов на орбите с i до $i+1$ происходит при приходе в систему нового запроса в момент, когда в системе нет свободных исправных приборов. Матрица $\mathbf{J}$ выделяет соответствующие такой ситуации состояния процесса $\mathbf{r}_t$. При этом состояние приборов не изменяется, что объясняет тот факт, что матрица $\mathbf{J}$ – диагональная. Уменьшение числа запросов на орбите с i до $i-1$ происходит в момент осуществления повторной попытки в момент, когда в системе имеется свободный исправный прибор. Матрица $\mathbf{E}^{+}$ отражает факт изменения значения 0 компоненты процесса $\mathbf{r}_t$ с минимальным номером на значение 1 в момент начала обслуживания запроса.

Диагональные элементы матрицы $\mathbf{Q}_{i,i}$ отрицательные. Их модули задают интенсивность выхода процесса ξ_t из соответствующего состояния. Недиагональные элементы матрицы $\mathbf{Q}_{i,i}$ задают интенсивности переходов этого процесса, не влекущих изменение числа запросов на орбите. Так, матрицы $\mathbf{E}^{-}$, $\mathbf{E}^{\varphi}$, $\mathbf{E}^{\gamma}$, задают интенсивности переходов процесса $\mathbf{r}_t$ в моменты окончания обслуживания, поломки и восстановления прибора, соответственно.

Можно проверить, что цепь Маркова ξ_t является асимптотически квазитеплицевой цепью Маркова, см. [7]. Используя результаты из [7], можно получить условие ее эргодичности. Стационарное распределение вероятностей цепи Маркова ξ_t может быть вычислено с использованием численно устойчивых алгоритмов, разработанных в [7] и [8]. Это дает возможность вычислять различные характеристики производительности системы и решать задачи оптимизации.

4. Заключение

В данной работе мы изучили ненадежную многолинейную СМО с повторными вызовами, разнородными приборами и стратегией первоочередного занятия более быстрых приборов.

ЛИТЕРАТУРА

1. Mushko V.V., Jakob M.J., Ramakrishnan K. O., Krishnamoorthy A., Dudin A.N. Multiserver queue with addressed retrials // Annals of Operations Research. 2006. V. 141. P. 283—-301.
2. Falin G.I. Stability of the multiserver queue with addressed retrials // Annals of Operations Research. 2012. V. 196(1). P. 241-246.
3. Liu Mei, Dudin A. Analysis of Retrial Queue with Heterogeneous Servers and Markovian Arrival Process // Applied Probability and Stochastic Processes. 2020. Springer. P. 29-49.
4. Dudin A.N., Klimenok V.I., Vishnevsky V.M. The theory of queuing systems with correlated flows. - Springer Nature. 2019. 431 P.
5. Falin G., Templeton J. G. C. Retrial queues. CRC Press. 1997. V. 75.
6. A. Graham, Kronecker products and matrix calculus with applications, Ellis Horwood, Cichester, 1981.
7. Klimenok V.I., Dudin A.N. Multi-dimensional asymptotically quasi-Toeplitz Markov chains and their application in queueing theory // Queueing Systems 54 (2006): 245-259.
8. Dudin, Sergei, et al. Effective algorithm for computation of the stationary distribution of multi-dimensional level-dependent Markov chains with upper block-Hessenberg structure of the generator // Journal of Computational and Applied Mathematics 366 (2020): 112425.

UDC: 519.872

MULTI-SERVER LOSS QUEUEING SYSTEM WITH THE $BMMAP$ ARRIVAL PROCESS

Chesoong Kim[1], A.N. Dudin[2], S.A. Dudin[2], O.S. Dudina[2]

[1]Department of Industrial Engineering, Sangji University, Republic of Korea
[2]Belarusian State University, 4, Nezavisimosti Ave., Minsk, Belarus

Abstract

In this paper, we consider a multi-server queueing system with heterogeneous customers that arrive according to a Batch Marked Markovian Arrival Process. The service time is exponentially distributed with the rate depending on the type of customer. The system does not have a buffer and the acceptance discipline is Partial Admission. The operation of the system is described by the multi-dimensional Markov chain. The stationary distribution and performance measures of the system including loss probabilities of customers of different types are computed.

Keywords: Multi-server loss queueing system, Batch Marked Markovian Arrival Process, loss probabilities

1. Introduction

The queueing model considered in this paper is the direct generalization of the famous Erlang loss queue of $M/M/N/0$ type which was widely applied for performance evaluation and capacity planning in telecommunication networks. The disadvantage of this queue from the point of view of the needs of modeling modern telecommunication networks is that the model of the stationary Poisson process does not fit real-world systems. As more suitable model, the Batch Markovian Arrival Process ($BMAP$) was offered, for references see, [1], [2], [3]. The extension of the Erlang loss queue to the case of the $BMAP$ arrival process was analyzed in the paper [4] where the $BMAP/PH/N/0$ model was considered. A restriction of that model is that all customers are assumed to be homogeneous. While in many real-world systems the customers are heterogeneous and requiring a different amount of the service time. The flow of heterogeneous customers can be described by the Marked Markovian Arrival Process ($MMAP$), see, e.g. [5]. A generalization of the $MMAP$ to the case

This work was supported by the Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (Grant No. NRF-2020R1A2C1006999).

of the batch arrivals (the Batch Marked Markovian Arrival Process – $BMMAP$) was described, e.g., in [6]. Here, we assume that the arrival process is defined by the $BMMAP$.

The difficulty of the analysis of the model stems from the fact that if one tries to use the simple description of the Markov chain defining the behavior of the system by monitoring the type of a customer in service in each busy server, then the state space of the Markov chain may be very large. If the number of types of the customers is equal to R, the number of servers is N, the state space of the underlying process of the $BMMAP$ is W, then the state space of the Markov chain has $W\frac{R^{N+1}-1}{R-1}$ elements. E.g., if $W = 2$, $R = 3$, $N = 11$, the number of states in the state space is 531 440. But in real systems the number of servers, N can be much larger than 11 and computations become practically impossible. In this paper, following an idea from [7], the Markov chain describing the behavior of the system is constructed in such a way that the stationary distribution of the chain can be successfully computed for essentially larger values on N.

2. Model description

We consider a N-server queuing system with R types of customers whose structure is presented in Figure 1.

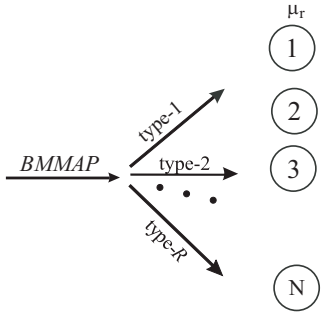


Fig. 1. Structure of the system.

The input flow of customers in the system is described by the $BMMAP$. Customers arrival in the $BMMAP$ is defined by the irreducible continuous-time Markov chain ν_t, $t \geq 0$, having a finite state space $\{1, \ldots, W\}$. The sojourn time of the chain ν_t, $t \geq 0$, in the state ν is exponentially distributed with the positive parameter λ_ν. After this time expires, with probability $p_0(\nu, \nu')$ this chain jumps into the state ν', $\nu' \in \{1, \ldots, W\}$, $\nu' \neq \nu$, without generation of customers and with probability $p_r^{(k)}(\nu, \nu')$ the chain jumps into the state ν', $\nu' \in \{1, \ldots, W\}$, and a batch consisting of k customers of type r is generated. Here, we assume that the maximal batch size

of type r customers is limited by the parameter $K_r, K_r \geq 1$. Let us denote as K the maximal batch size among all types of customers, i.e., $K = \max\{K_r,\ r = \overline{1,R}\}$.

The parameters defining the $BMMAP$ can be stored in the square matrices D_0, $D_r^{(k)}$, $r = \overline{1,R}$, $k = \overline{1,K_r}$, of size W defined by their entries:

$$(D_0)_{\nu,\nu} = -\lambda_\nu,\ (D_0)_{\nu,\nu'} = \lambda_\nu p_0(\nu,\nu'),\ (D_r^{(k)})_{\nu,\nu'} = \lambda_\nu p_r^{(k)}(\nu,\nu'),\ \nu,\nu' = \overline{1,W}.$$

The matrix $D(1) = D_0 + \sum\limits_{r=1}^{R}\sum\limits_{k=1}^{K_r} D_r^{(k)}$ is a generator of the Markov chain ν_t, $t \geq 0$.

Let us denote as $\boldsymbol{\theta}$ the stationary probability vector of the states of the Markov chain ν_t, $t \geq 0$. This vector can be found as the unique solution to the system $\boldsymbol{\theta}D(1) = \mathbf{0}, \boldsymbol{\theta}\mathbf{e} = 1$. Hereinafter, $\mathbf{0}$ is a zero row vector and $\mathbf{e}$ is a column vector consisting of ones.

The average intensity λ_r of type-r customers arrival and the total average intensity λ of customers arrival are defined as $\lambda_r = \boldsymbol{\theta}\sum\limits_{k=1}^{K_r} kD_r^{(k)}\mathbf{e},\ r = \overline{1,R},\ \lambda = \sum\limits_{r=1}^{R}\lambda_r$.

For more information on the $BMMAP$, see, e.g., [6].

If a batch of customers of any type arrives when the required number of servers is idle, the customers immediately start processing by the servers (service). If at a batch arrival moment the required number of servers is not available, the part of customers occupies the idle servers, and the rest of customers, for which there are not idle servers, leave the system forever (is lost). This means that we assume the partial admission discipline.

The service time of r-type customer is exponentially distributed with the parameter μ_r, $r = \overline{1,R}$.

3. Process of the system states

The behavior of the system under study can be described by the regular irreducible continuous-time Markov chain

$$\xi_t = \{n_t, \nu_t, \eta_t^{(1)}, \eta_t^{(2)}, \ldots, \eta_t^{(R)}\},\ t \geq 0,$$

where, during the epoch t,

- n_t is the number of busy servers in the system, $n_t = \overline{0,N}$;
- ν_t is the state of the underlying process of the $BMMAP$, $\nu_t = \overline{1,W}$;
- $\eta_t^{(r)}$ is the number of r-type customers on service, $\eta_t^{(r)} = \overline{0,n_t}$, $r = \overline{1,R}$, $\sum\limits_{r=1}^{R}\eta_t^{(r)} = n_t$.

To simplify the analysis of the chain ξ_t, we propose to use for description of the service time of an arbitrary customer the generalized PH distribution, see [8], with an irreducible representation $(\boldsymbol{\beta}^1, \boldsymbol{\beta}^2, \ldots, \boldsymbol{\beta}^R, S)$ where the vector $\boldsymbol{\beta}^r$ is of size R and has all zero entries except the r-th entry that is equal to 1 and the diagonal matrix S has the diagonal entries $-\mu_1, \ldots, -\mu_R$.

Let $P_n(\boldsymbol{\beta}^r)$ be the matrix the entries of which define the transition probabilities of the process $\eta_t^{(r)}$ at the moment when a new type-r customer is accepted to the system at the epoch when the number of busy servers is n, $0 \le n < N$.

Let $B_n(\boldsymbol{\mu})$ be the matrix the entries of which define the intensities of transitions of the process $\eta_t^{(r)}$ when some customer finishes its service in the system, where $\boldsymbol{\mu} = (\mu_1, \mu_2, \ldots, \mu_R)$.

We have elaborated the recursive procedures for computation of the matrices $P_n(\boldsymbol{\beta}^r)$ and $B_n(\boldsymbol{\mu})$ that are not presented here due to the paper size limitation.

Let I be the identity matrix, O be a zero matrix of an appropriate dimension; $\otimes$ and $\oplus$ be the symbols of the Kronecker product and sum of matrices, respectively; $\hat{I}_n = -\mathrm{diag}\{B_n(\boldsymbol{\mu})\mathbf{e}\}$, $n = \overline{1, N}$, where $\mathrm{diag}\{\ldots\}$ denotes the diagonal matrix with the diagonal entries defined by the vector in the brackets; $T_n = \binom{n+R-1}{R-1}$, $n = \overline{1, N}$.

By analyzing all possible transitions of the Markov chain $\xi_t, t \ge 0$, during an interval of the infinitesimal length and rewriting the intensities of these transitions in the block matrix form, we obtain the following result.

Theorem 1. The infinitesimal generator Q of the Markov chain ξ_t, $t \ge 0$, has the following block structure

$$Q = \begin{pmatrix} Q_{0,0} & Q_{0,1} & Q_{0,2} & Q_{0,3} & \ldots & Q_{0,N} \\ Q_{1,0} & Q_{1,1} & Q_{1,2} & Q_{1,3} & \ldots & Q_{1,N} \\ O & Q_{2,1} & Q_{2,2} & Q_{2,3} & \ldots & Q_{2,N} \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ O & O & O & O & \ldots & Q_{N,N} \end{pmatrix}.$$

The non-zero blocks are defined as follows:

$$Q_{0,0} = D_0,\ Q_{n,n} = D_0 \oplus \hat{I}_n,\ n = \overline{1, N-1},\ Q_{N,N} = D(1) \oplus \hat{I}_N,$$

$$Q_{n,n-1} = I_W \otimes B_n(\boldsymbol{\mu}),\ n = \overline{1, N},$$

$$Q_{0,n} = \sum_{r=1}^{R} X_n^r,\ n = \overline{1, \min\{N-1, K\}},$$

$$Q_{0,N} = \sum_{r=1}^{R} \sum_{k=N}^{K_r} D_r^{(k)} \otimes [P_0(\boldsymbol{\beta}^r) P_1(\boldsymbol{\beta}^r) \times \cdots \times P_{N-1}(\boldsymbol{\beta}^r)],$$

$$Q_{n,n+k} = \sum_{r=1}^{R} Y_n^{r,k},\ n = \overline{1,N},\ k = \overline{1, \min\{N-n-1, K\}},$$

$$Q_{n,N} = \sum_{r=1}^{R} \sum_{k=N-n}^{K_r} D_r^{(k)} \otimes [P_n(\boldsymbol{\beta}^r) P_{n+1}(\boldsymbol{\beta}^r) \times \cdots \times P_{N-1}(\boldsymbol{\beta}^r)],\ n = \overline{1,N},$$

where

$$X_n^r = \begin{cases} D_r^{(n)} \otimes [P_0(\boldsymbol{\beta}^r) P_1(\boldsymbol{\beta}^r) \times \cdots \times P_{n-1}(\boldsymbol{\beta}^r)], & \text{if } n \le K_r, \\ O, & \text{otherwise,} \end{cases}$$

$$Y_n^{r,k} = \begin{cases} D_r^{(k)} \otimes [P_n(\boldsymbol{\beta}^r) P_{n+1}(\boldsymbol{\beta}^r) \times \cdots \times P_{n+k-1}(\boldsymbol{\beta}^r)], & \text{if } k \le K_r, \\ O, & \text{otherwise.} \end{cases}$$

Let us form the row vectors $\boldsymbol{\pi}_n$, $n = \overline{0,N}$, of the stationary probabilities of the states of the Markov chain ξ_t which are enumerated in the reverse lexicographic order of the components $\eta_t^{(1)}, \ldots, \eta_t^{(R)}$ and the direct lexicographic order of the component ν_t. It is well known that the vectors $\boldsymbol{\pi}_n$, $n = \overline{0,N}$, satisfy the following system of linear algebraic equations:

$$(\boldsymbol{\pi}_0, \boldsymbol{\pi}_1, \ldots, \boldsymbol{\pi}_N) Q = \mathbf{0},\ (\boldsymbol{\pi}_0, \boldsymbol{\pi}_1, \ldots, \boldsymbol{\pi}_N)\mathbf{e} = 1 \tag{1}$$

where Q is the infinitesimal generator of the Markov chain $\xi_t,\ t \ge 0$.

System (1) can be solved by any of the well-known methods for solving the finite system of linear algebraic equations. It is advisable to use a special stable algorithm proposed in [4] and based on the idea of censored Markov chains.

4. Performance measures

Having the vectors $\boldsymbol{\pi}_n$ be calculated, we can calculate several stationary performance measures of the queue under consideration.

The average number of customers in the system is $N_{customers} = \sum_{n=1}^{N} n\boldsymbol{\pi}_n\mathbf{e}$.

The intensity of the output flow of the successfully serviced customers is

$$\lambda_{out} = \sum_{n=1}^{N} \boldsymbol{\pi}_n (I_W \otimes B_n(\boldsymbol{\mu}))\mathbf{e}.$$

The intensity of the output flow of the successfully serviced type-r customers is

$$\lambda_{out}^{(r)} = \sum_{n=1}^{N} \boldsymbol{\pi}_n (I_W \otimes B_n(\boldsymbol{\mu}_r))\mathbf{e} \quad \text{where} \quad \boldsymbol{\mu}_r = \underbrace{(0, 0, \ldots, 0, \mu_r, 0, \ldots, 0)}_{R},\ r = \overline{1,R}.$$

The average number of type-r customers in the system is $N^{(r)}_{customers} = \frac{\lambda^{(r)}_{out}}{\mu_r}$.
The loss probability of an arbitrary customer is

$$P_{loss} = 1 - \frac{\lambda_{out}}{\lambda} = \frac{1}{\lambda}\sum_{n=0}^{N}\boldsymbol{\pi}_n[(\sum_{r=1}^{R}\sum_{k=N-n+1}^{K_r}(k-N+n)D_r^{(k)})\otimes I_{T_n}]\mathbf{e}.$$

The loss probability of an arbitrary type-r customer is

$$P^{(r)}_{loss} = 1 - \frac{\lambda^{(r)}_{out}}{\lambda_r} = \frac{1}{\lambda_r}\sum_{n=0}^{N}\boldsymbol{\pi}_n[(\sum_{k=N-n+1}^{K_r}(k-N+n)D_r^{(k)})\otimes I_{T_n}]\mathbf{e},\ r=\overline{1,R}.$$

5. Conclusion

In this paper, we analyzed a multi-server queueing system with the correlated flow of the batches of heterogeneous customers and without buffer. We calculated the stationary distribution of the system states and the main performance measures including the probability of losses of different types of customers.

REFERENCES

1. Chakravarthy, S.R.: The batch Markovian arrival process: a review and future work // Advances in Probability Theory and Stochastic Processes, Notable Publications Inc., New Jersey, 2001, pp. 21-29.
2. Lucantoni D. New results on the single server queue with a batch Markovian arrival process // Communication in Statistics-Stochastic Models. 1991. V.7. P. 1–46.
3. Dudin A.N., Klimenok V.I., Vishnevsky V.M. The theory of queuing systems with correlated flows. - Springer Nature. 2019. P. 431.
4. Klimenok V.I., Kim C.S., Orlovsky D.S., Dudin A.N. Lack of invariant property of Erlang $BMAP/PH/N/0$ model // Queueing Systems. 2005. V. 49. P. 187–213.
5. He Q. M. Queues with marked customers // Advances in Applied Probability. 1996. V. 28. P. 567-587.
6. Al-Begain K., Dudin A.N., Mushko V.V. Novel queueing model for multimedia over downlink in 3.5 g wireless network // Journal of Communications Software and Systems. 2006. V. 2. No. 2. P. 68-80.
7. Ramaswami V., Lucantoni D. M. Algorithms for the multi-server queue with phase type service // Stochastic Models. 1984. V. 1. P. 393-417.
8. Dudin A. et al. Multi-server queueing system with a generalized phase-type service time distribution as a model of call center with a call-back option //Annals of Operations Research. 2016. V. 239. No. 2. P. 401-428.

UDC: 004.89

Intelligent system for forecasting the effectiveness of space services in solving economic problems

A.V. Yudin[1] and P.Yu. Grosheva[1]

[1]Peoples Friendship University of Russia (RUDN University), 6 Miklukho-Maklaya Street, Moscow, 117198, Russian Federation

yudin-av@rudn.ru, grosheva-pyu@rudn.ru

Аннотация

The development and application of space services development programs for more efficient solution of economic problems is a priority for many countries. However, the solution of such problems requires an increasing amount of information for analysis, which complicates the analysis itself and the construction of forecasting. In this regard, the article formulates approaches to forecasting the effectiveness of the use of space services for solving economic problems, on the basis of which a model of an intelligent system for managing the effectiveness of space services is proposed. It allows automatically predicting the effectiveness of space services, taking into account dynamically changing factors, which eventually ensures timely decision-making to increase the competitiveness of the created space services at the early stages of their life cycle.

Keywords: *intelligent systems, space services, automated evaluation procedures, efficiency forecasting*

1. Introduction

Currently, more than fifty countries are investing in space services development programs. National priorities in the provision of space services depend on the level of development of the country's space program and more general aspects of State regulation. In general, the main demand from national governments is for solutions in the areas of environmental monitoring and climate change. This demand contributes to the emergence of research units based on Earth remote sensing (ERS) in national and international research programs. Another driver of the space services market is the programs of developing countries that aim to achieve independence in providing the country with the necessary services.

Space services are used in various fields of activity, both in public administration and in the national economy. The scope of application of geographic information

This paper has been supported by the RUDN University Strategic Academic Leadership Program.

services based on satellite data is extremely wide. In terms of the structure of the world market of geoservices, in 2020-2028 the most demanded solutions will be in infrastructure monitoring, environmental monitoring, defense and natural resources. It is expected that in total they will account for more than 80% of the geoservice market. The remaining 20% of other areas of geoservice development will be distributed between energy, emergency monitoring, water management, the financial sector, and geolocation services.

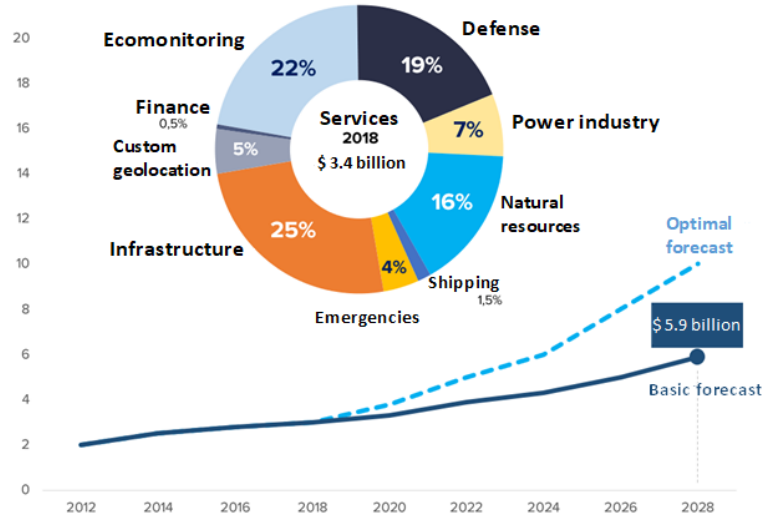


Fig. 1. Model of an intelligent system for forecasting the efficiency of space services for solving economic problems

Today, there is great interest in the possibilities of geoinformation solutions in the management of construction projects. These solutions are used not only by developers, general contractors, developers and designers, but also credit and financial institutions (banks, insurance companies) and investment companies. Earth remote sensing (ERS) data can be used in scoring models of banks, insurance and investment companies to determine lending rates, insurance rates and investment risks, as well as control the targeted spending of allocated loan or investment funds, simplify work with collateralized objects in the form of commercial and residential real estate. Due to to space imagery, users of geoservices quickly receive an objective assessment of the activities of economic entities for conducting due diligence procedures. Executive authorities get the opportunity to control the construction and reconstruction of strategically important objects for the country, the region, promptly receive notifications of threats to the deadline for work, and refer to retrospective data

when assessing potential performers of construction work. Individuals who actually finance the construction of housing facilities can refer to remote sensing data when choosing and assessing the reliability of a particular developer, giving preference to companies that do not have "long-term construction".

Today, there is a demand for comprehensive environmental monitoring. Geographic information solutions are becoming the basis for building integrated information and analytical systems for environmental monitoring. Remote space monitoring allows you to identify official landfills of waste storage and illegal landfills, conduct operational monitoring of emergency situations (oil spills and other man-made accidents), monitor the progress of liquidation of objects of accumulated environmental damage and reclamation of disturbed lands. The use of environmental monitoring services in Russia is of particular importance in connection with the implementation of the national project "Ecology". Satellite data are demanded by both state supervisory authorities and commercial enterprises whose economic activities affect the state of the environment, in particular, we are talking about enterpriseы for mining and processing of minerals. From space, the state of reservoirs and coastal areas is well monitored with dredging methods of gold mining; monitoring of unique natural objects, relict forests and water resources is carried out, the progress of construction of infrastructure facilities for waste management is monitored. In general, there is also a demand for objective independent environmental monitoring on the part of society: citizens want to have tools for objective control of the environmental situation in their region of residence and in the country as a whole.

The world market for Earth remote sensing satellites (ERS) from space today is characterized by several trends:

1) space constellations, including radar sensing satellites, instead of single satellites are becoming a new standard for the industry;
2) the number of launches of small spacecraft is increasing;
3) in the long term up to 2028, the number of remote sensing satellites in orbit is expected to increase from 227 to 1402 spacecraft;
4) in the long term up to 2028, it is expected that the average cost of a satellite will decrease from 110 million rubles. USD up to 49 million USD;
5) in the long term up to 2028, it is expected that the percentage of satellites worth less than 50 million US dollars will increase from 49
6) projects of space constellations with low CAPEX (capital expenses) are aimed at creating a large number of vehicles in order to ensure the most frequent coverage of the Earth;
7) projects of space constellations with high CAPEX (capital expenses) are aimed at creating several vehicles with the highest possible sensor resolution;

8) there is a diversification of the payload of the spacecraft of the same constellation (adding hyperspectral, infrared sensors, SAR, etc.);
9) constellations of optoelectronic remote sensing satellites of the world's leading operators provide survey of the territory of interest with a resolution of at least 1 m with a frequency of up to 30 minutes; the new constellations of radar sensing satellites being created are striving for the same survey frequency.

In view of these trends, the issue of increasing the competitive advantages of services is relevant, the assessment of which can be carried out through determining the indicator of competitiveness (discussed in detail in [1]) and the effectiveness of its application in solving economic problems.

The solution to this problem will be considered in the creation of automated logical inference systems that allow to evaluate the effectiveness.

2. Literature review

Currently, there is a widespread introduction of systems for evaluating and predicting the characteristics of various objects. For example, systems for assessing the creditworthiness of individuals or legal entities in banks, assessing the quality of products at enterprises, assessing various economic categories (risks, product competitiveness, etc.), the intensity of traffic flows in traffic regulation, etc. are widely used. Systems for evaluating the quality and feasibility of various projects are widely implemented. In this regard, the processing of satellite information by intelligent methods is justified and corresponds to the practice of many global companies and government agencies. The development of intelligent methods for solving problems in a specific subject area is carried out by scientific laboratories or specialized companies that have an R&D department as a staffing structure.

Existing approaches to the use of remote sensing data for solving applied economic problems to some extent cover the problem of assessing and increasing the efficiency of their solution.

For example, in the work of M. M. Zheleznov et al., an intelligent space system for project management (a system for monitoring potentially dangerous sections of railway track) is proposed, containing a set of remote sensing spacecraft connected with an expert system that provides the construction of a project implementation model and monitoring the current state of project execution associated with the project monitoring and management center [2]. In the work of A. A. Kaganovich, the problem of the use of geoinformation technologies in the management of rural development is analyzed, the economic importance of informatization in rural development is emphasized, the need for the introduction of geoinformation technologies in agricultural production is justified. Recommendations on the integration of a geoinformation

system based on an information and consulting support system into the mechanism of effective state management of the agro-industrial complex are proposed [3].

In addition, there are already theoretical developments in the creation of an intelligent space project management system that uses remote sensing data in preparing of informed management decisions [4].

A distinctive feature of the approaches proposed in this study is the development of algorithms for an automated information system for predicting the effectiveness of the use of space services, the main principle of which will be the use of a complex of intelligent methods for processing remote sensing data in solving economic problems related to the development, monitoring, providing the necessary resources for the functioning of the actor in the economy without human participation, which is particularly relevant in the digital economy.

3. Algorithms for predicting the effectiveness of space services for solving economic problems

To solve the problem of forecasting the effectiveness of the use of space services for solving economic problems and ensuring its functioning, it is necessary to consistently solve the following tasks:

1) Analysis of existing information on the use of space services and traditional ground-based methods for solving economic problems. We can consider the following requirements for geoinformation services and providing consumers with satellite data:
   - high speed of data acquisition: to effectively solve many economic problems, data is needed that arrives within a few hours after placing the order;
   - high detail: a large demand for high-and ultra-high spatial resolution satellite imagery, both optoelectronic and radar;
   - high frequency: multiple regular coverage of the same territories to create monitoring products;
   - ease of use: placing an order and receiving analytics in a few clicks on a single geoportal;
   - extensive functionality: the ability for the user to participate in the planning of shooting and order management in the personal account;
   - ease of use: access to geoservices 24/7, user support, the ability to pay online for services using a bank card.
2) Building a statistical knowledge base on the effectiveness of the use of space services and the most accurate ground-based methods in solving economic problems.

3) Search in the statistical database of knowledge about the effectiveness of the use of space services and available ground-based methods in solving economic problems that are closest to the problem under consideration.
4) Building a forecast of the effectiveness of the use of space services to solve an economic problem.

4. Model of the system for forecasting the efficiency of space services for solving economic problems

Let us consider the formal definition of the forecast model [5] эof the efficiency of using space services in solving economic problems. Consider the set of possible space services A. We also introduce the set B of potential economic problems in which space services are used or planned to be used from the set A. Then the evaluation of the effectiveness of space services for solving these problems will be expressed as a numerical function $E : A \times B \to R$.

This function is defined on pairs $< a, b >$, where $a \in A$, $b \in B$.

Thus, for each space service and economic problem, we define a numerical function that has the meaning of the effectiveness of the use of this service in the economic problem under consideration.

We will assume that the function E satisfies the conditions $E(a, b) \geq 0$, and $E(a, b) \leq 1, \forall a \in A, b \in B$.

In this case, we will say that service a is completely ineffective for solving problem b if $E(a, b) = 0$. On the other hand, we will say that service a is completely effective for solving problem b if $E(a, b) = 1$.

To construct the function $E(a, b)$ it is necessary to use methods based on automated evaluation procedures, since the exact information for calculating the function E is usually unknown.

The layout of the intelligent system for predicting the effectiveness of the use of space services for solving economic problems has the following form, shown in figure 2.

At the input such a system receives heterogeneous information about the characteristics of the space services being developed and a description of the tasks that these services are aimed at, which is further formalized in an intelligent system.

At the output, the system gives out the results of forecasting the effectiveness of the use of space services for solving economic problems at a time interval specified by the user, taking into account dynamically changing factors. As is typical of all forecasts, the shorter the forecast period, the more accurate its results.

The proposed intelligent system for forecasting the effectiveness of the use of space services for solving economic problems allows to take into account dynamically

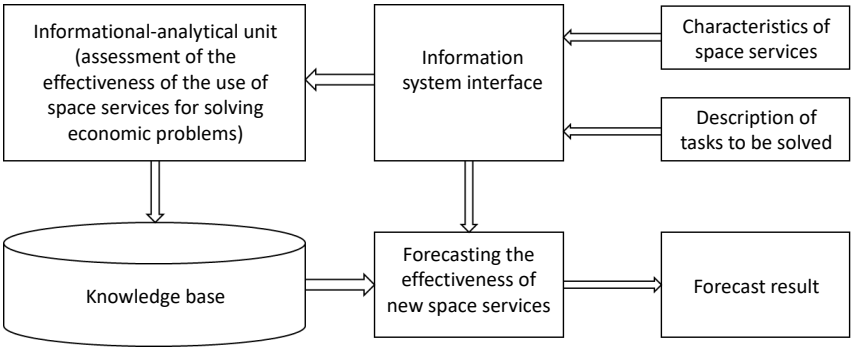


Fig. 2. Model of an intelligent system for forecasting the efficiency of space services for solving economic problems

changing environmental factors when making forecasts, in order to make timely managerial decisions to increase a high level of competitiveness.

We will describe each stage of the system of forecasting the effectiveness of the use of space services for solving economic problems.

Stage 1. Analysis of existing information on the use of space services and traditional ground-based methods for solving economic problems

Forecasting the economic efficiency of space services should be based on the experience of using space technologies and services to solve economic problems, as well as the experience of using traditional ground-based methods for this. To obtain and generalize this experience, it is proposed to use automated systems that will allow obtaining and formalizing the relevant knowledge.

The complexity of assessing the efficiency of using space services to solve economic problems is a circumstance that determines the subjectivity of this information, since standard economic indicators cannot fully describe the effectiveness of space services in solving each specific problem.

Consider the set B' which is a subset of the set of potential problems B. The set B' consists of economic tasks, the solution of which is supposed to be carried out through the use of satellite information. Let this set consist of K elements: $B' = \{b_1, b_2, \ldots, b_K\}$.

For each element b_k, we consider the set A_k, which is a subset of the set A and consists of elements that are either space services or existing ground-based methods used to solve the economic problem b_k. We will consider the set $A^k = \bigcup_{k=1}^{K} A_k$.

Thus, for the operation of the information system at the first stage, we will use two sets – A_k and B'.

We introduce the set C as follows: $C = A^k \times B'$. This set consists of all possible pairs $\{\langle a, b\rangle\}$, $a \in A^k$, $b \in B'$.

For each pair $\langle a, b\rangle$ the efficiency $Q(\langle a, b\rangle)$ of applying method a (satellite or traditional) for solving a specific economic problem b is estimated.

As an assessment of efficiency, one can consider the indicator of the competitiveness of a space service (satellite service) used to solve a specific problem, the construction procedure of which is described in detail in [8]. The procedure involves determining the competitiveness of IQ_1 space services in relation to other services (satellite services) and determining the competitiveness of IQ_2 in relation to ground-based methods for solving similar problems.

Based on these values, the indicator $Q(\langle a, b\rangle)$ of the efficiency of using the space service a for solving a specific economic problem b can be obtained as the geometric mean of the integral indicator of competitiveness in relation to other services (IQ_1) and the integral indicator of competitiveness in relation to ground-based methods problem solving (IQ_2):

$$Q(\langle a, b\rangle) = \sqrt{IQ_1 \cdot IQ_2} \tag{1}$$

Stage 2. Being in the statistical database of knowledge on the effectiveness of the use of space services and the most accurate ground-based methods in solving economic problems is the closest problem for the considered

After completing stage 1, we have a statistical knowledge base of the implemented economic tasks using both space services and ground-based methods. To assess the effectiveness of solving new economic problems based on space services, it is necessary to complete the present stage, at which for a new economic problem there is an existing economic problem that is closest to the one under consideration.

Let us consider a new economic task of the national economic complex, for which it is planned to use space services. We denote it by β. In order to find the closest economic problem from those already solved, it is necessary to consider a set of characteristics of these problems: x_1 - the first characteristic of the problem; x_2 is the second characteristic of the problem, etc. These characteristics may not apply to all economic problems from set B. We will consider each characteristic as a numerical characteristic. If any characteristic is not related to a specific problem, then we will assume that this characteristic is equal to zero.

Since the characteristics are numbers, we will consider their normalized values. In this case, normalization will be carried out for each characteristic for all economic problems from the set B. As a result of this normalization, we should get that the

minimum value of the characteristic is 0, and the maximum value is 1. Thus, we have $x_n(b) \in [0,1]$, $\forall b \in B$, where $x_n(b)$ is the value of the characteristic x_n for problem b.

The closest problem to the economic problem β we will call the economic problem b^*, which satisfies the following condition:

$$\sum_{n=1}^{N} |x_n(b^*) - x_n(\beta)| = \min_{b \in B} \sum_{n=1}^{N} |x_n(b^*) x_n(\beta)| \tag{2}$$

Thus, the problem b^* is an economic problem that most corresponds to the economic problem β in terms of the selected characteristics.

Stage 3. Obtaining a predictive assessment of the effectiveness of the use of space services to solve an economic problem

At the previous stage, we identified the economic problem b^*, which is closest to the economic problem β, for which it is necessary to calculate the predicted value of the efficiency of using space services.

Consider a subset A_* of the set A, which consists of space services corresponding to the economic problem b^*. We will assume that the same space services are also used to solve the economic problem β.

To predict the effectiveness of the use of space services in solving economic problems, an average perfomance index is used:

$$IE(\beta) = \frac{1}{|A^*|} \sum_{a \in A_*} Q(\langle a, \beta \rangle) \tag{3}$$

This index characterizes the predicted overall level of efficiency in the use of space services in solving economic problems.

Using the calculated index, it is possible to track the dynamics of changes in the efficiency of space services over time.

The use of the method of calculating this index in conjunction with the approach developed and described in the scientific literature for determining the dynamics of competitiveness of products and services based on weak signals [2] allows for forecasts to take into account dynamically changing environmental factors in order to make timely management decisions to increase the level of competitiveness (see Figure 3).

The figure simulates a situation when, based on the results of the analysis of weak signals, it is predicted at time $t = 1$ that a competitor with higher characteristics will appear on the market, as a result of which the efficiency index of the assessed service decreases and a decision is made to implement measures to increase its competitiveness. At the moment of time $t = 3$, a positive result of the implementation of measures is achieved, at which the index of the assessed service significantly exceeds the performance index of the competitor's service.

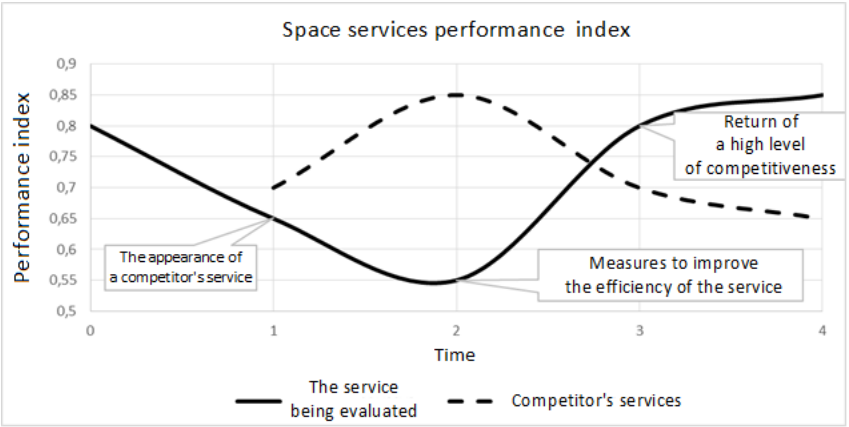


Fig. 3. Perspective structure of the geoservices market (according to the Euroconsult agency, 2019)

5. Conclusion

The approaches proposed in the article to forecasting the effectiveness of the use of space services for solving economic problems can form the basis for an intelligent system for managing the effectiveness of space services, the synergistic effect of which will be due to the use of a set of economic tools and methods that allow such management to be carried out in the face of dynamically changing factors and risks.

REFERENCES

1. Tyulin A.E., Chursin A.A., Elerdova M.A., Yudin A.V. Creation of radically new products and their commercialization // Creative Economy. 2020. Volume 14.No. 7.P. 1257-1278. doi: 10.18334 / ce.14.7.110697
2. Zheleznov M.M., Ponomarev V.M., Pevzner V.O. Prevention of emergency situations by detecting volumetric deformations on potentially dangerous sections of the railway track using aerospace survey // Science and technology of transport. 2017. No. 4. S. 95-104.
3. Kaganovich A. A. Application of GIS technologies in the management of rural development // Economics and Entrepreneurship. 2018. No. 3 (92). pp. 429-433.
4. Intelligent space system for project management // RF Patent No. RU 2679541, 11.02.2019 / Tyulin A.E., Chursin A.A., Shamin R.V., Yudin A.V.
5. Petrusevich D.A. Analysis of mathematical models used for econometrical time series forecasting. Russian Technological Journal. 2019;7(2):61-73. (In Russ.) https://doi.org/10.32362/2500-316X-2019-7-2-61-73

UDC: 004.896

Basis for the formation of a digital ecosystem of an industrial holding

A.E. Tyulin[1], A.A. Chursin[1], A.V. Yudin[1], P.Yu. Grosheva[1]

[1]Peoples Friendship University of Russia (RUDN University), 6 Miklukho-Maklaya Street, Moscow, 117198, Russian Federation

tyulin-ane@rudn.ru, chursin-aa@rudn.ru, yudin-av@rudn.ru, grosheva-pyu@rudn.ru

Abstract

The article proposes and discusses new approaches to the formation of digital ecosystems of an industrial holding as a mechanism for organizing an innovation process based on a system of innovations inherent in a specific, relatively stable network that unites representatives of the State, business, science and education around a common vision of scientific and technological development and common approaches to the development of new innovative technologies, creating value, facilitating exchanges between two or more interdependent groups of participants based on big data analytics.

Keywords: *unique products, intelligent systems, digital economy, systemic approach, digital ecosystem*

1. Introduction

Currently, the economies of most countries are undergoing digital transformation. This process is related to the emergence of new economic categories and paradigms, as well as the formation of new approaches to the management of economic processes at equal levels with the establishment of new subjects of management. In this regard, in Russia it was proposed to establish a new subject of regulation at the state level — a digital platform or ecosystem and to establish qualifying features for them at the legislative level, as well as to define a regulatory body.

The understanding of the digital ecosystem corresponds to the general concept of the Innovation Triple Helix, which represents the relations of such key participants and stakeholders of the innovation ecosystem as the State, business and scientific organizations [1]. This concept is currently being actively developed [2, 3]. The global economic trends lead to the development of radical innovations, new growing markets, industries and activities [4, 5].

This paper has been supported by the RUDN University Strategic Academic Leadership Program.

In the modern conditions of the new economy (the economy of creativity, knowledge and innovation), innovations and radically new products and services appear not so much as a result of successive linear processes that take place in the development units of industrial holdings from the moment of shaping the product idea to bringing it to the market, but as a result of cross-functional interactions between different areas of knowledge within the ecosystem, where the processes of managing innovation and creating radically new products require competencies that go far beyond one subject area.

2. Theoretical basis for the formation of a digital ecosystem of an industrial holding

The formation of a digital ecosystem focused on the manufacture of diversified promising products using the same production capacities, basic technologies and competencies (with the addition of new ones if necessary for the manufacture of certain products) for different market segments and different consumer groups provides the manufacturer with economic stability through the successful sale of products that are competitive in quality and price on the market. In this case, when building digital ecosystems, it is necessary to determine the optimal structure of the organization of production, in which the introduction of digital production will be most effective, taking into account its characteristics. To this end, it is necessary to determine the optimal structure of its own production, the list of parts and components that will use the created flexible automated production at full capacity, and the work that will be transferred to related organizations that will produce certain parts, components, units of high quality in a shorter time based on their competencies and technological capabilities, while ensuring the lowest cost of production, which is carried out for a given competitiveness.

The development of digital ecosystems of the industrial holding, as one of the key factors and sources of economic development, is carried out at several levels, corresponding to the stages of the life cycle of radically new products. The development of the ecosystem is not a one-time, but a continuous process, the course of which is determined by the flexibility of the company and the influence of external factors-trends in global technological development.

The main factor in creating new products aimed at long-term satisfaction of needs is the continuous growth of the innovative capacity of the participants of the digital ecosystem, related to a large volume of basic research, extracting valuable information from large volumes, mastering and developing a set of competencies, technologies and equipment based on this valuable information and this research, improving management methods.

The development of such processes in economic systems is related to the effect of the economic law of advanced development, which states that the developed products must have consumer utility (value) that increases the needs of society, leading to the emergence of new markets and creating a stable economic development of the producer. In the context of globalization, outdated economic mechanisms and technological solutions are stagnating, and only the most innovatively developed organizational and economic structures find new economic and technological niches due to the previously created competitive advantages and are able to create conditions for the establishment and development of markets for innovative goods.

Production management processes are rapidly changing under the influence of advanced digital technologies integrated into the entire business process of the organization. Digital technologies stimulate new business models aimed at creating and producing goods of the future that can put the company on a path of advanced development. The emergence of new players on the market with fundamentally new products entails the need to organize such production management, which will create highly competitive products and services with the best technical characteristics and quality on the market at the lowest cost.

Here is a formal description of the mechanisms underlying the procedures for the coordinated development of the digital ecosystem of a holding and the companies operating within its perimeter.

3. Economic and mathematical model for the development of a digital ecosystem of a holding

To describe the development model of the digital ecosystem of the holding, we will consider the problem of optimal management, taking into account the active evolution of its companies, striving to achieve their own goals and operating in the scientific, technological and innovative framework of the main platform. We will use and develop the models suggested in [**?**].

Let P_0 (the digital ecosystem of the holding) strives to achieve the highest value of efficiency $f_0(xi, u)$, which can be understood, for example, as the profit from the sale of high-tech products of companies, where u is the innovative capacity of the digital ecosystem, and x_i is the competitiveness of companies, $x_i \in X$, $x = (x_1, ..., x_n)$. The companies of the holding, in turn, strive to increase their own efficiency $f_i(x_i, u_i)$, $i = 1, 2, ..., n$, which can be understood as its profitability. Let's consider several possible mechanisms for the coordinated development of the digital ecosystem and its companies.

Mechanisms of the 1st type (direct). The digital ecosystem does not control the competitiveness of companies, but freely provides them with the competencies and knowledge of its own core. The best values of the control variables are determined

from the solution of the problem:

$$G_1 = \sup_{u \in U} \min_{x_i \in B_i^1} f_0(x, u),$$

where B_1 is the set of optimal controls for the competitiveness of companies:

$$B_i^1 = \{x_i \in X_i | f_i(x_i, u_i) = \max_{y_i \in X_i} f_i(y_i, u_i)\}$$

.

Mechanisms of the 2nd type (closed-loop). The digital ecosystem relies on the use of the innovative capacity of companies in the formation of its own optimal management strategy, and formulates it as functions $u_i = u_i(x_i)$. Then

$$B_i^2 = \{x_i \in X_i | f_i(x_i, u_i) = \sup_{y_i \in X_i} f_i(y_i, u_i') - \delta_i(u_i')\}, \delta_i(u_i') \geq 0,$$

and the greatest guaranteed result (efficiency of functioning) of the digital ecosystem is

$$G_2 = \sup_{u' \in U} \inf_{x_i \in B_i^2(u_i')} f_0(x, u').$$

In this case, the effectiveness of the holding companies is determined from the condition

$$sup_{(x_i, u_i) \in D_i}, D_i = \{x_i \in X_i, u_i \in U_i | f_i(x_i, u_i) > \max_{x_i \in X_i} \min_{u_i \in U_i} f_i(x_i, u_i)\},$$

where the efficiency $L = \max \min f_i(x_i, u_i)$ of company i is guaranteed.

As a result of such mechanisms, the increase in the efficiency of the holding companies determines the increase in the efficiency of the entire ecosystem as a whole.

4. Conclusion

The solution to the problem of forming a digital ecosystem of an industrial holding is associated with building its scientific and technological capacity and methods of its transformation into a new product, taking into account the sufficiency of resource provision.

In real economic conditions, new products are created in companies that already are participants of the commodity markets and produce goods created within the digital ecosystem. The creation of radically new products is linked to the continuous development of the ecosystem itself due to the constant increase in scientific and technical potential and its transformation into highly competitive products that ensure the stability of the industrial holding in the market.

Thanks to a comprehensive analysis and processing of information about the current level of competitiveness of the entire range of products manufactured by the holding, the existing scientific and technological capacity and the level of unique competencies of both the holding and its companies, the level of development of an adaptive production system, the level of interaction with customers in terms of shaping the image of new products and considering market consumer preferences, it is possible to build the trajectory of economic development of a high-tech industrial holding and determine the conditions under which the holding is in a state of advanced development and global technological superiority.

REFERENCES

1. Etzkowitz, H. and Leydesdorff, L. (1995) The Triple Helix: University - Industry - Government relations a laboratory for knowledge based economic development, EASST Review, 14 (1): 14-19.
2. Ranga, M. and Etzkowitz, H. (2013) Triple Helix systems: analytical framework for innovation policy and practice in the Knowledge Society, Industry and Higher Education, 27 (4): 237-262
3. Carayannis, E. and Campbell, D. (2009) Mode 3" and "Quadruple Helix": toward a 21st century fractal innovation ecosystem, International journal of technology management, 46(3-4): 201-234
4. Chursin, A. and Tyulin, A. (2018) Competence management and competitive product development: Concept and implications for practice. Springer International Publishing, Cham
5. Tyulin, A. and Chursin, A. (2020)The new economy of the product life cycle: Innovation and design in the digital era. Springer Nature Switzerland, Cham
6. Chursin, A.A., Dubina I.N., Carayannis E.G, Tyulin A.E, Yudin A.V. (2021) Technological Platforms as a Tool for Creating Radical Innovations, Journal of the Knowledge Economy

УДК: 519.718

Определение показателей долговечности распределённой коммуникационной сети метеостанций минимальной конфигурации

Е.Э. Головинов[1], Д.А. Аминев[1,2], Д.В. Козырев[2,3], В.Н. Кулыгин[4]

[1]ФГБНУ «ВНИИГиМ им. А.Н. Костякова», ул. Большая Академическая, 44 корпус 2, Москва, Россия
[2]Институт проблем управления им. В.А. Трапезникова РАН, ул. Профсоюзная, 65, Москва, Россия
[3]Российский университет дружбы народов, ул. Миклухо-Маклая, д. 6, Москва, 117198, Россия
[4]Национальный исследовательский университет «Высшая школа экономики», ул. Мясницкая, д. 20, 101000 Москва, Россия

evgeny@golovinov.info, aminev.d.a@ya.ru, kozyrev-dv@rudn.ru, trancercom@gmail.com

Аннотация

Рассматривается топология распределённой коммуникационной сети метеостанций (РКСМ) минимальной конфигурации и структура её аппаратуры. Проведена экспертная оценка показателей долговечности РКСМ и определены режимы её функционирования. Разработаны модель комплексного коэффициента нагрузки и ресурса РКСМ, типовая модель эксплуатации элементов РКСМ. Проведен расчет срока службы РКСМ с учётом режимов в условиях Южного региона России.

Ключевые слова: надежность, долговечность, агрометеопараметры, метеостанция, резервирование, интенсивность отказов, работоспособность, срок службы.

1. Введение

В современных реалиях развития цифровой экономики, включая сельское хозяйство, одной из задач является оснащение пространства средствами автоматического мониторинга агрометеопараметров, в том числе полей сельскохозяйственного назначения. Метеостанции являются техническими средствами для регистрации таких агрометеопараметров как температура, влажность приземного слоя атмосферы, осадков, температуры и влажности почвы на различных

Работа выполнена при финансовой поддержке РФФИ в рамках научного проекта № 19-29-06043.

глубинах. По этим данным определяется поливная норма, качество, и однородность сельскохозяйственных культур на поле, урожайность и на эффективность хозяйства в целом. Вышеуказанные факторы определяют жесткие требования по надежности агрометеостанций [1].

Модель надежности РКСМ минимальной и расширенной конфигурации исследована в работах [2, 3]. Также проведена оценка комплектов запасных частей, инструментов и принадлежностей (ЗИП) для РКСМ минимальной конфигурации [4]. Однако вопросы определения показателей долговечности, включая оценку комплексного коэффициента нагрузки и ресурса РКСМ, не проработаны. Поэтому для полноты определения надёжности как показателя качества, необходимо создать модель комплексного коэффициента нагрузки и ресурса РКСМ минимальной конфигурации, чтобы оценить её долговечность.

2. Распределенная коммуникационная сеть метеостанций

Расположенная на землях сельскохозяйственного назначения РКСМ, имеющая топологию сети типа многоуровневая звезда, состоит из аппаратуры необслуживаемых метеостанций (МС), каналов связи (КС) точек доступа к глобальной сети Интернет посредством мобильной связи или WiFi, которые могут быть удалены от МС на расстояния до нескольких десятков километров [5, 6, 7]. Минимальная конфигурация РКСМ состоит из точки доступа мобильной связи, и трех МС, причем $МС_1$ и $МС_2$ соединены с точкой доступа напрямую, а $МС_3$ через станцию $МС_2$ (рис. 1а).

Согласно структурной схеме аппаратуры МС, представленной на рис. 1б, микроконтроллер (МК) принимает и обрабатывает данные от датчиков метеопараметров и GPS приемника, управляет передачей данных по GSM модему и модулю WiFi. В минимальной конфигурации используются датчики температуры приземного слоя атмосферы, влажности почвы и воздуха. Телеметрия и данные местонахождения передаются в сеть Интернет на мониторинговый сервер для дальнейшей обработки, анализа и представления оператору. Наблюдать за местонахождением МС и считывать метеопараметры можно в любой момент времени. Источник питания (ИП) представляет собой аккумуляторную батарею.

3. Модель комплексного коэффициента нагрузки и ресурса РКСМ

Для оценки ресурса элементов в зависимости от режима применения используют базовую модель:

$$T_{p\gamma\,\text{раб}} = \frac{T_{p\gamma}}{K_{\text{н}}}, \tag{1}$$

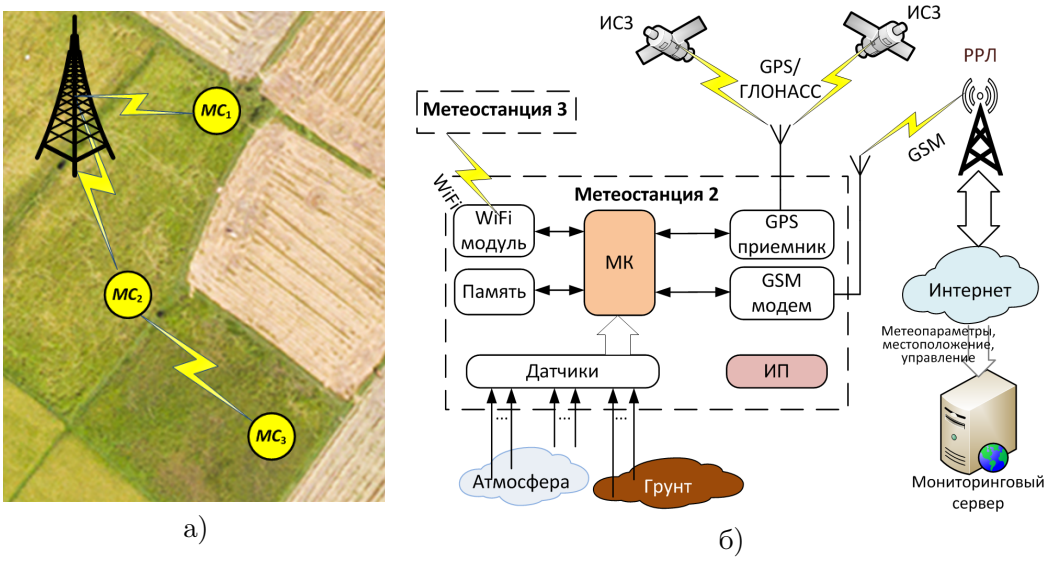


Рис. 1. РКСМ минимальной конфигурации на местности (а), структура аппаратуры $МС_2$ (б)

где: $T_{p\gamma\,\text{раб}}$ — гамма-процентный ресурс элемента в рабочем режиме; $T_{p\gamma}$ — гамма-процентный ресурс элемента в максимальном по техническим условиям режиме; $K_{\text{н}}$ — коэффициент нагрузки элемента.

На рисунке 2 приведен общий случай модели эксплуатации РКСМ.

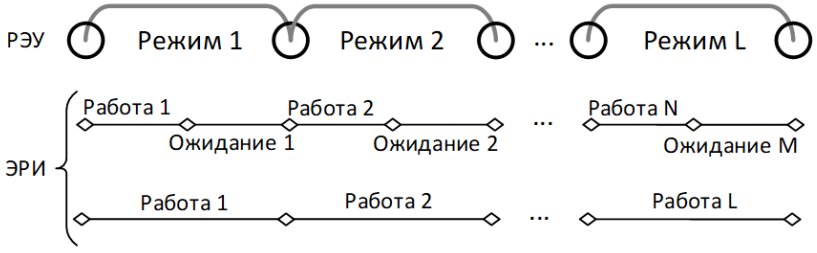


Рис. 2. Типовая модель эксплуатации элементов РКСМ

Модель эксплуатации МС может содержать L режимов работы. При этом ее элементы могут содержать либо L режимов работы, либо N режимов работы и M режимов ожидания. Математическая модель гамма-процентного ресурса элемента с учетом модели эксплуатации для режимов работы $T_{p\gamma\,\text{цикл}}$ будет иметь

вид:

$$T_{p\gamma\,\text{цикл}} = \sum_{n=1}^{N} (T_{p\gamma\,\text{раб}\,n} \cdot K_{\text{и.э.}\,n}) \tag{2}$$

где: $T_{p\gamma\,\text{раб}\,n}$ – гамма-процентный ресурс элемента в n-ом режиме работы; $K_{\text{и.э.}\,n}$ – коэффициент интенсивности эксплуатации элемента n-ого режима работы.

Коэффициент интенсивности эксплуатации для n-ого режима работы будет определяться выражением:

$$K_{\text{и.э. раб}\,n} = \frac{t_{\text{и.э. раб}\,n}}{t_{\text{общее}}}, \tag{3}$$

где: $t_{\text{и.э. раб}\,n}$ – время работы в n-ом режиме; $t_{\text{общее}}$ – общее время работы МС.

4. Расчёт показателей долговечности РКСМ минимальной конфигурации

Экспертная оценка показателей долговечности РКСМ, раскрытая в таблице 1, включает оценку минимальной наработки — количество часов работающего изделия до первого отказа, т.е. определяет наработку (минимальную) приходящуюся на один отказ. Здесь ресурс радиоканала не оценивается.

Компонент	Наработка на отказ	Срок хранения
МК	50000 часов непрерывной работы	25 лет
Память	50000 часов непрерывной работы (не превышая 100000 циклов перезаписи)	25 лет
Датчики	20000 часов (оценка сделана на основе датчиков температуры)	15 лет
GPS приемник	50000 часов	25 лет
GSM модем	80000 часов	25 лет
Антенна	10000 часов	25 лет
WiFi модуль	45000 часов	25 лет
ИП (с учетом аккумулятора)	5 лет + 900 циклов (при условии разрядки аккумулятора до 20% емкости)	5 лет

Таблица 1. Экспертная оценка показателей долговечности РКСМ

Ресурс РКСМ минимальной конфигурации оценивается по формуле:

$$\min(\text{Компонент}_1, ..., \text{Компонент}_n) \tag{4}$$

Для условий южного региона России предлагается следующая модель эксплуатации РКСМ. Для Краснодарского края эксплуатация апрель-октябрь, по данным [8], градиент температур приведен на графике на рисунке 3.

Из графика видно, что:

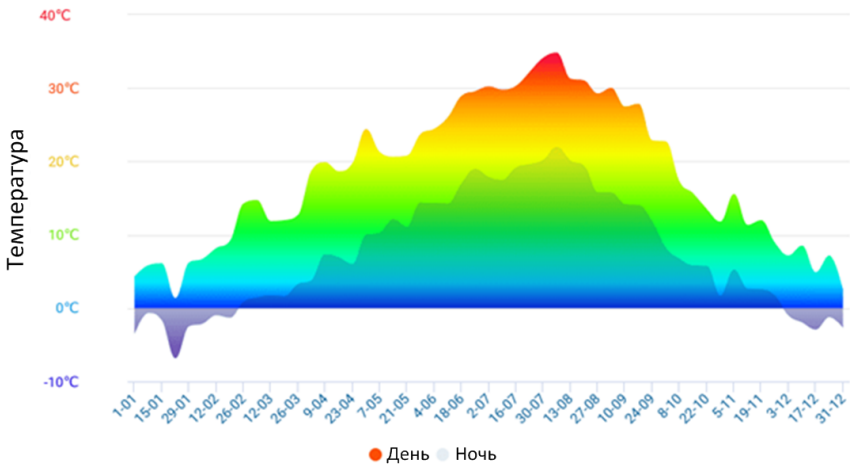


Рис. 3. Градиент температур Краснодарского края России

- в летние месяцы средняя дневная температура: +33° С, средняя ночная температура +18° С;
- в весенние и осенние месяцы средняя дневная температура: +15° С, средняя ночная температура +5° С;
- в зимние месяцы средняя дневная температура: +5° С, средняя ночная температура -5° С.

Из приведенных выше данных следует что, метеостанция находится в режиме работы 7 месяцев и 5 месяцев в режиме хранения. При этом температура окружающей среды:

- в течение 3 месяца∗1/2 суток (лето день) ≈ 33-35° С;
- 3 месяца*1/2 суток (лето ночь) + 4 месяца*1/2 суток (осень день) ≈ 15-18° С;
- 4 месяца*1/2 суток (осень весна день) ≈ 5° С.

Для оценки срока службы метеостанции на основе формул (1)-(3) можно вывести следующую математическую модель:

$$T_{\text{службы}} = \sum_{n=1}^{n} K_{\text{и.э.}\,n} \cdot \frac{T_{p\,\gamma}}{K_{\text{н}\,n}} \tag{5}$$

Исключая датчики, для которых в справочнике Надежность ЭРИ 2006 [9] не приводятся зависимости от температур окружающей среды, критичными элементами метеостанции в условиях повышенной окружающей среды являются

интегральные микросхемы. Как правило температура превышения для интегральных микросхем (ОЗУ и процессоров) работающих в щадящих режимах составляет 20-30° относительно температуры окружающей среды.

Коэффициент режима $K_{\text{р ИМС}}$ для ИМС при заданных температурах определяется в соответствии со справочником Надежность ЭРИ 2006 [9]. По нему можно рассчитать коэффициент нагрузки $K_{\text{н ИМС}}$ для ИМС (таблица 2).

Период эксплуатации	Температура ИМС*	$K_{\text{р ИМС}}$	$K_{\text{н ИМС}}$	Длительность периода	$K_{\text{и.э.}}$
Лето день	80°C	3	1	0,5*(3/12)	0,125
Лето ночь Весна/осень день	45°C	1,5	0,5	0,5*(3/12) + + 0,5*(4/12)	0,292
Весна/осень ночь	25°C	1	0,33	0,5*(4/12)	0,167
Зима	-	1,4 ($K_{\text{усл ИМС}}$†)	1,4	5/12	0,417

Таблица 2. $K_{\text{н ИМС}}$ и $K_{\text{и.э}}$ аппаратуры в зависимости от периода эксплуатации

Сроки хранения в соответствии с такими условиями снижаются в 1,4 раза.

Подставляя значения из таблицы 2 можно получить оценку срока службы метеостанции, на основе оценки срока службы ИМС:

$$T_{\text{службы}} = \sum_{n=1}^{N} K_{\text{и.э.}\,n} \cdot \frac{T_{p\,\gamma}}{K_{\text{н}\,n}}. \tag{6}$$

$T_{\text{службы}} = 3/12\text{месяца} * 1/2 * 50000\text{часов}/1 + (3/12 + 4/12)\text{месяца} * 1/2 * 50000/0,5 + 4/12\text{месяца} * 0,5/0,33 + 5/12 * 25\text{лет} * 365\text{дня} * 24\text{часа}/1,4 = 100500\text{часов} \approx 11$ лет.

При этом, в период пониженной температуры (совпадает с периодом хранения) температура изменяется в диапазоне от -5° С до +15° С, что можно отнести к условиям хранения по классификации справочника «Надежность ЭРИ 2006» - «В полевых условиях». Наиболее критичным элементом системы в таких условиях является аккумуляторная батарея. Сроки хранения в соответствии с

*Рабочая температура ИМС при соответствующих температурах окружающей среды.

†Коэффициент условий хранения по классификации справочника «Надежность ЭРИ 2006» [9].

такими условиями снижается в 2,86 раза относительно хранения в отапливаемом хранилище. Что в общей сумме снизит срок службы аккумуляторов (в соответствии с (1)) на 16 месяцев.

Подставляя значения из таблицы 2, можно получить оценку срока службы метеостанции, на основе оценки срока службы АКБ:

$T_{\text{службы}}$ = 5 лет*(7/12 месяцев) + (5 лет/2,86)*(5/12 месяцев) = 3,65 лет.

Поэтому целесообразно дать рекомендацию извлекать из метеостанции аккумуляторные батареи с последующей транспортировкой их в отапливаемое помещение на период простоя, что обеспечивает срок службы метеостанции в течение 10 лет при условии однократной замены аккумуляторных батарей по истечении 5 лет эксплуатации, замены датчиков каждые 2 года.

5. Заключение

Предложенная модель позволяет оценить комплексный коэффициент нагрузки и ресурса РКСМ минимальной конфигурации на основе эксплуатационной интенсивности отказов элементов и детальных сведений о режимах эксплуатации.

Для оценки ресурса РКСМ, имеющей сложную модель эксплуатации, при расчете необходимо учитывать каждый отрезок времени в отдельности, в соответствии с режимами применения, соответствующими данному отрезку времени. За время жизненного цикла РКСМ может эксплуатироваться в разных режимах работы, при этом, во время работы МС отдельные её элементы могут находиться как в режиме работы, так и в режиме ожидания.

По результатам расчётов показателей долговечности РКСМ минимальной конфигурации, её срок службы составляет порядка 10 лет, что является вполне допустимым для такого класса систем.

Литература

1. Бородычев В.В., Лытов М.Н, Головинов Е.Э. Мониторинг и управление орошением в режиме реального времени: монография. – М.: МЭСХ, 2017. – 154с. ISBN: 978-5-9909008-9-9
2. Головинов Е.Э., Аминев Д.А., Козырев Д.В., Ларионов А.А. Модель надёжности распределённой коммуникационной сети метеостанций минимальной конфигурации / Распределенные компьютерные и телекоммуникационные сети: управление, вычисление, связь (DCCN-2019) материалы XXII Международной научной конференции. Российский университет дружбы народов. 2019. С. 484–491. eLIBRARY ID: 41384277
3. Aminev D., Golovinov E., Kozyrev D., Larionov A., Sokolov A. Reliability evaluation of a distributed communication network of weather stations / Lecture

Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 11965 LNCS, pp. 591-606. Springer, Cham. 2019. DOI 10.1007/978-3-030-36614-8_45

4. Головинов Е.Э., Аминев Д.А., Татунов С. Ю., Полеский С.А., Козырев Д.В. Оценка комплектов ЗИП для распределённой коммуникационной сети метеостанций минимальной конфигурации// Труды 23-й Международной конференции "Распределенные компьютерные и телекоммуникационные сети: управление, вычисление, связь"(DCCN 2020, Москва). М.: ИПУ РАН, 2020. С. 733-742. eLIBRARY ID: 46409089
5. Аминев Д.А., Головинов Е.Э., Инновационный подход к проведению полевых экспериментов // Качество. Инновации. Образование. – М.: –2015 № 1. – С. 26-30.
6. Е.Э. Головинов, Д.А. Аминев, В.А. Кулаков, Ш.М. Бакиров, П.В. Григорьев Анализ системных решений портативных метеостанций // Международный форум «Микроэлектроника-2017» 3-я МНК «Электронная компонентная база и электронные модули». Республика Крым, г. Алушта, 02-07 октября 2017 г. С.155-159.
7. Головинов Е.Э., Аминев Д.А., Бакиров Ш.М. Анализ элементной базы для реализации мобильного измерительного агрометеокомплекса//Проектирование и технология электронных средств – Владимир: №3, 2017. С. 33–40.
8. `http://hikersbay.com/climate/russia/krasnodar`
9. Надежность ЭРИ: справочник. – М.: МО РФ, 2006. – 256 с.

UDC: 519.234.6

Information Spreading in Non-Homogeneous Evolving Networks *

Natalia M. Markovich[1] and Maksim S. Ryzhov[1]

[1]V.A. Trapeznikov Institute of Control Sciences, Russian Academy of Sciences , Profsoyuznaya Str. 65, 117997 Moscow , Russia

Abstract

The paper is devoted to finding of leader nodes in evolving directed random networks with regard to the information spreading. We consider non-homogeneous networks consisting of several weakly connected subgraphs having different distributions of node in- and out-degrees. This is a plausible situation for real complex networks. The evolution of the network in time starting from a seed set of nodes is provided by linear preferential attachment schemes. We compare the spreading rate of nodes which share their messages with other nodes when they belong to different subgraphs of the non-homogeneous network. It is found that the nodes of the subgraph with the most heavy tailed out-degree distribution may spread their messages faster. We compared the spreading capacity of the linear preferential attachment used also for the graph evolution with a well-known SPREAD algorithm and found that the latter can disseminate the information faster.

Keywords: non-homogeneous evolving network, information spreading, linear preferential attachment, SPREAD algorithm, leading nodes

1. Introduction

Evolving scale-free network model has been studied in different areas: citation networks [1], the web-page popularity by PageRank during evolutionary changes [2], the evolution of the network [3]. Graphs reflecting a structure of the networks can be directed or undirected.

Information spreading, as a message delivery model in the whole network (the full spreading) [4] or in some community (the partial spreading) [5], [6] has an application for the parallel grid calculations in the computation network.

One of our objectives is to study a linear preferential attachment (PA) schemes as a tool to spread the information. The PA has been proposed in [3] for the network

*The authors were partly supported by Russian Foundation for Basic Research (grant 19-01-00090).

evolution. We use the PA also to spread a message among nodes. We aim to demonstrate that the PA may spread faster than the SPREAD algorithm in [4] for some parameter values of the PA.

Another objective is to find leading nodes which can spread the information fast among nodes of a non-homogeneous graph. On each evolution step, a newly appended directed edge (i, j) may cause the message exchange if a node i with a message communicates to a node j without it. We compare the linear PA schemes with the SPREAD algorithm for directed graphs.

Here, two novelties are implemented. Firstly, the information spreading is studied in a directed evolving graph where the probabilities to choose nodes and create edges for communications are determined by the linear PA schemes α, β and γ. These probabilities depend on the in-/out-degrees and parameters of the PA schemes. Since the in- and out-degrees of nodes may change over time during the evolution, the latter probabilities are evolving, too. Secondly, non-homogeneous graphs are considered. The latter consist from subgraphs whose in- and out-degrees have different power law distributions. We aim to study, does the spreading rate of nodes depend on the heaviness of tail of their node degrees.

The paper is organized as follows. In Section 2, related works regarding the spreading information (Section 2.1), the PA (Section 2.2) are provided. In Section 3, our main results concerning for the spreading are presented. The exposition is finalized by Conclusions.

2. Related works

2.1. Information spreading. Let us describe the idea of the spreading algorithm SPREAD proposed in [4] for a undirected graph $G = (V, E)$. Here, V and E are sets of graph vertices and edges, respectively. The most realistic situation is that clocks of all nodes are not quite synchronized. Considering an asynchronous time model, a node may initiate a communication by ticks of a global clock which are modelled as a Poisson process of rate $n = |V|$, [4, 5]. Let $k \geq 0$ denote the index of a tick, on which at most one node can receive messages by communicating with another node. To this end, on a clock tick one of n nodes (let say a node i) of the graph is chosen uniformly. Then this node i chooses a node j uniformly among its neighbors with probability $P_{ij} = 1/d_{\max}$, where $d_{\max} = \max_{i \in V} d_i$, d_i is the node degree of node i. In [5] it is proposed to use $P_{ij} = 1/d_i$ to avoid the knowledge of the maximal node degree in the network. As in [4] $S_i(k)$ defines the set of nodes that have the message m_i from node i at the end of the clock tick k. After clock tick $k + 1$, we have either $|S_i(k + 1)| = |S_i(k)|$ or $|S_i(k + 1)| = |S_i(k)| + 1$.

Here, we use the SPREAD for directed graphs considering a partial information spreading, i.e. a node i spreads its message m_i to a part of the rest nodes, only.

Then the next node j is proposed to select uniformly in the set of nodes $V\backslash S_i(k)$ without m_i at the clock tick k.

Due to directed graphs we assume that node $i \in S_i(k)$ may share its message with node j, if there is a directed edge $(i \to j)$ from i to j. We consider probabilities $P_{ij} = 1/d_i$, $P_{ij} = 1/I_i$ and $P_{ij} = 1/O_i$, where I_i and O_i are the in- and out-degree of the node i, respectively. Let us explain the difference between the cases. When I_i and O_i are used the node i initiates the communication to one of its nearest neighbors with in-coming and out-going edges of the node i, respectively. The node i may share its message with the node j only in the case of probability $1/O_i$. The probability of $1/I_i$ is excluded since the message cannot be spread.

2.2. Preferential attachment. The linear PA schemes [3, 7] start with an initial directed graph $G(k_0)$ with at least one node and k_0 edges. For the non-negative parameters α, β, γ such as $\alpha + \beta + \gamma = 1$, and $\Delta_{in}, \Delta_{out}$, the model constructs a growing sequence of directed random graphs $G(k) = (V(k), E(k))$. A graph $G(k)$ is produced from $G(k-1)$ by adding a directed edge. Denote the number of nodes at step k as $N(k)$, and in- and out-degree of node w in the graph $G(k)$ with k edges as $I_k(w)$ and $O_k(w)$. Three scenarios of the edge creation are proposed in [3, 7], which are activated by flipping a 3-sided coin with probabilities α, β and γ. The i.i.d. trinomial r.v.s with values 1, 2 and 3 and the corresponding probabilities α, β and γ are generated to select schemes.

- According to α-scheme add a new node w_{new} and an edge $(w_{new} \to w)$ with probability α. Choose the existing node $w \in V(k-1)$ with probability $P(choose\ w \in V(k-1)) = \frac{I_{k-1}(w)+\Delta_{in}}{k-1+\Delta_{in}N(k-1)}$.
- According to β-scheme add a new edge $(w_{new} \to w)$ with probability β, where both existing nodes w_{new} and w are chosen independently and with probability $P(choose\ (w_{new} \to w)) = \frac{O_{k-1}(w_{new})+\Delta_{out}}{k-1+\Delta_{out}N(k-1)} \cdot \frac{I_{k-1}(w)+\Delta_{in}}{k-1+\Delta_{in}N(k-1)}$.
- According to γ-scheme add a new node w_{new} and an edge $(w \to w_{new})$ with probability γ. Choose $w \in V(k-1)$ with probability $P(choose\ w \in V(k-1)) = \frac{O_{k-1}(w)+\Delta_{out}}{k-1+\Delta_{out}N(k-1)}$.

This means that $N(k) = N(k-1)$ for β-schema and $N(k) = N(k-1)+1$ for the others. These scenarios realize a 'rich-get-richer' mechanism, when a node with large number of in-/out- edges can likely increase them with a high probability. As mentioned in [3], such model can create multiple edges between two nodes and self loops.

3. Main results

3.1. Comparison of the linear PA and SPREAD algorithm. Despite the linear PA is used for the evolution of directed graphs, we will also use it for the

information spreading. Let a message exchange between two nodes starts with the initial directed graph G_0 with N_0 nodes and k_0 edges. We assume that the message which is in disposal of one of the nodes is spreading among a fixed number n of nodes of the network. The nodes are assumed to have asynchronous clocks. We do a step of the linear PA (see, Section 2.2) with predefined values of the parameters $\alpha, \beta, \gamma, \Delta_{in}, \Delta_{out}$ by global poissonian clock ticks.

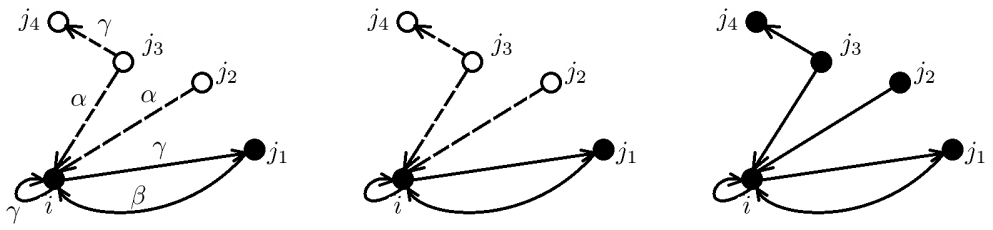


Figure 1. Example of spreading of the message m_i from the node i by the PA (left), the SPREAD with $P_{ij} = 1/O_i$ (middle) and the SPREAD with $P_{ij} = 1/d_i$ (right). Black filled points mark vertices with the message m_i at step $k = 6$. Doted lines show edges that cannot spread m_i. For the PA, the edges are marked with the names of the schemes which produce them.

The message m_i of node i can be delivered to a node j without the message only if the directed edge $(i \to j)$ is created. Such edge can be appended to the network by means of $\gamma-$ or $\beta-$ schemes, only. If the node i has no message, then the edge $(i \to j)$ does not spread the message further to the node j. The α-scheme increases the number of appending nodes without the message.

The evolution of the network by the PA schemes in [3] may lead to the appearance of multiple edges and self-loops due to using of the $\beta-$schema. This leads to graphs with cycles. The loops and "bottle-neck" edges may cause a stuck of messages and increase a spreading time.

Example 1. Let us demonstrate the spreading by methods considered above on the small graph. Fig. 1 shows that the SPREAD with $P_{ij} = 1/d_i$ can deliver the message m_i from node i to all nodes of the graph since the edge direction is disregarded for the information spreading. The PA and the SPREAD with $P_{ij} = 1/O_i$ spread along directed edges $(i \to j)$, only. The β-scheme leads to multiple edges and self-loops. Thus, the message can circulate between existing nodes having the message beforehand. This leads to the delay. For the SPREAD with $P_{ij} = 1/O_i$ new nodes cannot be selected among nodes with the message, but m_i cannot be spread from nodes without message, e.g., from j_3 to j_4. The PA and the SPREAD with

$P_{ij} = 1/O_i$ differ by the number of required rounds and by the order to obtain the message.

At clock tick k we obtain the graph $G(k) = (V(k), E(k))$ with the number of edges $|E(k)| = k + k_0$ and the number of nodes $|V(k)| = N(k) + N_0$ nodes.

We compare a spreading ability of the PA schemes and the SPREAD algorithm. To this aim, we first generate a graph by the PA schemes up to step k. Then we apply the SPREAD in the prepared directed graph starting with a node having a message. 100 graphs simulated by the PA are provided for each set of parameters α, β, γ all taken from the interval $[0.04, 0.96]$ with step 0.04 such that $\alpha + \beta + \gamma = 1$, and $\Delta_{in} = \Delta_{out} = 1$.

The graphs evolved start with a triangle of connected nodes and one of these nodes has a message to spread. In this part of the simulation, we do not study an impact of initial nodes and a non-homogeneity of the graph on the spreading time. The number of steps k is assumed to be limited as $k \leq K'$. We define the number of clock ticks required to disseminate the message from an initial node to n nodes with probability not less than $1 - \delta$ as

$$K^*(n, \delta) = inf\{0 < k \leq K' : Pr(|S(k)| = n) > 1 - \delta\}, \quad \delta \in (0, 1).$$

Let us take K' equal to 3000. If $K^* \leq K'$ holds, then $S(K^*) = n$ is likely held for a sufficiently small δ. If $S(K^*) < n$ holds, then K' steps of the evolution are likely not enough to disseminate the message to n nodes. Results of the comparison of the PA and the SPREAD algorithms for $n = 100$ nodes are presented in Fig. 2.

Let $q(k) = \frac{|S(k)|}{k}$ be an average number of nodes received the message per tick. The dependence between the averaging clock ticks $\langle K^* \rangle$ and the averaging proportions $\langle q(K^*) \rangle$ over 100 simulations for different γ values as well as the dependence between $\langle K^* \rangle$ and β are shown in Fig. 2 in left and right columns, respectively. For the PA schemes the small or the large β values imply that the information is spreading to newly appended nodes mostly by the $\gamma-$schema or the $\beta-$schema, respectively. The SPREAD algorithm operates in the graphs created by the PA schemes starting with an initial node having a message and a uniform selection of newly appearing nodes among its nearest neighbors.

The PA schemes spread the information faster for the larger γ, see Fig. 2 (top left and right). The number of ticks is similar for boundary values of β irrespective of γ as shown in Fig. 2 (top left). The events $\{S(K^*) < n\}$ are likely rare except the cases when γ is small, i.e. $\gamma \leq 0.04$, see Fig. 2 (middle top). This implies that $K' = 3000$ ticks are enough to spread the message to $n = 100$ nodes.

In contrast to the PA, the SPREAD can share the message to larger number of nodes irrespective to γ value, namely, the SPREAD strategy with the probability

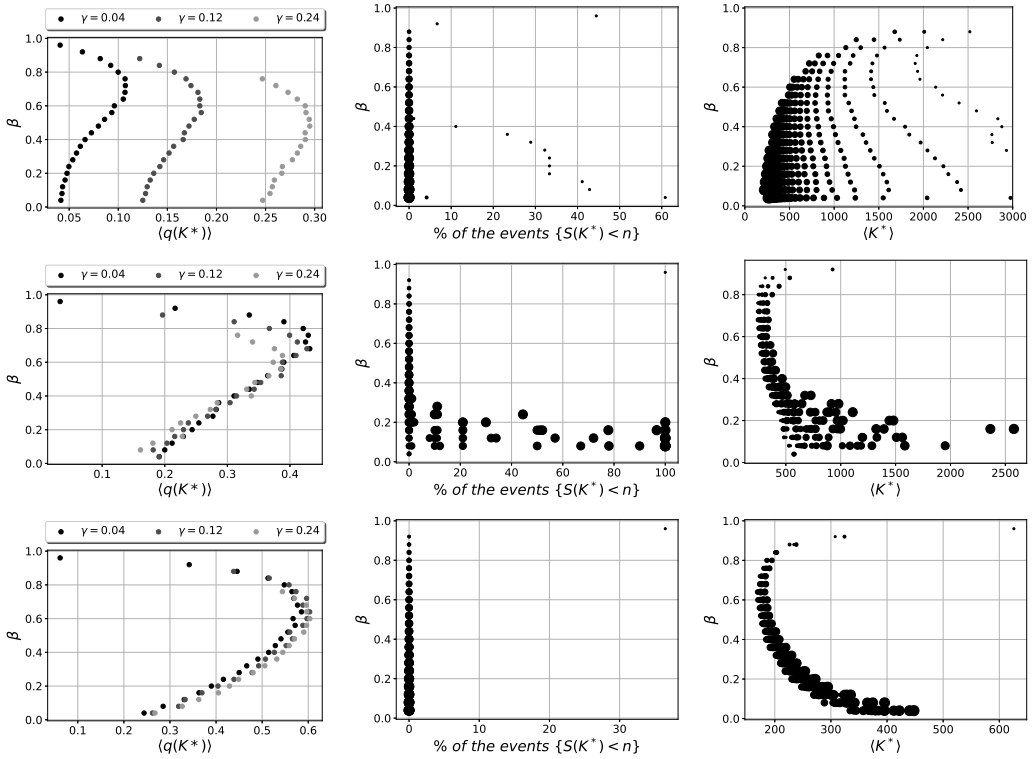


Figure 2. The β against the $\langle q(K^*)\rangle$ (left column); the parameter β against the proportion of the events $\{S(K^*) < n\}$ (middle column); the β against $\langle K^*\rangle$ (right column) for the PA schemes (top line) and the SPREAD algorithm corresponding to $P_{i,j} = 1/O_i$ (middle line) and $P_{i,j} = 1/d_i$ (bottom line). On the left, the point sizes are increasing with increasing of β values within the interval $[0.04, 0.96]$; in the middle and right figures - with increasing of γ values in the interval $[0.04, 0.96]$.

$P_{ij} = 1/d_i$ spreads the message among about 60% of nodes as far as the one with $P_{ij} = 1/O_i$ to 40%.

Comparing both SPREAD strategies, one can see that the choice of $P_{ij} = 1/d_i$ provides the message delivery to all n nodes for any β apart of the single point corresponding to γ not larger than 0.04, see Fig. 2 (middle bottom). For $P_{ij} = 1/O_i$, the number of events $\{S(K^*) < n\}$ grows up for approximately constant $\beta < 0.2$ and $\alpha + \beta > 0.8$, Fig. 2 (middle, the second line). For the same β the number of ticks K^* required to spread the information among n nodes is larger if one uses $1/O_i$ rather than $1/d_i$, see Fig. 2 (right middle and bottom). The minimum $\langle K^*\rangle$ is shown in Tab. 1. The PA demonstrates the best results for given parameters (α, β, γ).

	(α, β, γ)	minimum $\langle K^*(n,\delta)\rangle$
SPREAD, $1/O_i$	$(0.08, 0.84, 0.08)$	231
SPREAD, $1/d_i$	$(0.2, 0.6, 0.2)$	125
PA	$(0.04, 0.92, 0.04)$	115

Table 1. The minimum $\langle K^*(n,\delta)\rangle$ values with the set of parameters (α, β, γ), on which the minimums are received, for the PA and the SPREAD algorithms.

3.2. Spreading in non-homogeneous graphs. Note that K^* may strongly depend on the choice of the initial node that begins to spread its message. To investigate this problem, we consider a non-homogeneous graph whose nodes belong to communities with different distributions of in- and out-degrees.

To this aim, we simulate three Thorny Branching Tree (TBT) graphs by methodology proposed in [8] with predefined tail indexes (TIs) $(\alpha_{in}, \alpha_{out})$ of in- and out-degree power law distributions with tail distribution function $F(x, \alpha) = P\{X > x\} \sim cx^{-\alpha}$, $c > 0$. The symbol $\sim$ means an asymptotically equal. Simulated graphs are connected by few links to be not completely isolated. In practice, one can partition a graph into communities and test the distributions of their in- and out-degrees regarding the stationarity.

The TBT is a specific graph with possible cycles such that the sums of in- and out-degrees are the same. Each TBT contains 500 nodes in our experiment. The TBTs are simulated with the following pairs $(\alpha_{in}, \alpha_{out})$: the TBT_1 has $(3.8, 2)$, the TBT_2 - $(2.5, 2.5)$ and the TBT_3 - $(3, 4.5)$. After adding 60 inter-links between the TBTs in such a way to change values of the TIs not much, we estimate the TIs by the Hill's estimator. We obtain that the TBT_1 has $(3.79, 2.06)$, the TBT_2 - $(2.41, 2.61)$ and the TBT_3 - $(3.75, 4.48)$. Then the TBT_1 and TBT_2 have the smallest TIs of the out- and in-degree distributions, respectively. The smallest TI implies the most heavy-tailed distribution. This follows from the properties of regularly varying distributions [9]. Obviously, the distribution of the out-degree is more significant for the spreading than the distribution of the in-degree.

The TI of the TBT node degree, i.e. sum of its mutual independent in- and out-degrees coincides with the TI of the most heavy-tailed term according to properties of regularly varying distributions, [9]. We obtain the following TI $\alpha_{deg} = (2, 2.5, 3)$ and their estimates $\widehat{\alpha}_{deg} = (2.86, 1.99, 3.98)$ for TBT_1, TBT_2 and TBT_3, respectively.

We simulate 50 graphs evolved by the linear PA schemes with the same parameters. Afterwards, the PA and the SPREAD algorithm are used to spread one message from each node belonging to one of the TBT graphs to $n = 100$ other nodes. We examine K^* when the spreading starts from every node of each TBT graph. The following parameters $(\alpha, \beta, \gamma) = (0.4, 0.2, 0.4)$ and $(\alpha, \beta, \gamma) = (0.2, 0.1, 0.7), \Delta_{in} =$

$\Delta_{out} = 1, K' = 3000$ are taken. The resulted triples ($\min\{K^*\}$, $\langle K^* \rangle$, $\max\{K^*\}$) are presented in Tab. 2.

TBT $(\alpha_{in}, \alpha_{out}, \alpha_{deg})$	SPREAD, $1/O_i$	SPREAD, $1/d_i$	PA
$(\alpha, \beta, \gamma) = (0.4, 0.2, 0.4)$			
$TBT_1(3.8, 2.0, 2.0)$	(1500, 2000, 2500)	(171, 241.1, 330)	(1800, 2325, 2850)
$TBT_2(2.5, 2.5, 2.5)$	(2600, 2700, 2850)	(187, 267.1, 401)	(2150, 2225, 2300)
$TBT_3(3.0, 4.5, 3.0)$	(700, 2012.5, 2850)	(218, 349.7, 669)	(1150, 1883.3, 2350)
$(\alpha, \beta, \gamma) = (0.2, 0.1, 0.7)$			
$TBT_1(3.8, 2.0, 2.0)$	(100, 1566.6, 2800)	(169, 215.1, 289)	(100, 1356.5, 2800)
$TBT_2(2.5, 2.5, 2.5)$	(100, 1498.1, 2950)	(162, 226.3, 333)	(100, 1691.5, 2900)
$TBT_3(3.0, 4.5, 3.0)$	(100, 1531.3, 2900)	(183, 263.4, 421)	(100, 1717.3, 2900)

Table 2. The triple ($\min\{K^*\}$, $\langle K^* \rangle$, $\max\{K^*\}$) shows the minimum, the average and the maximum of K^* values over 50 simulations applying to the SPREAD with $P_{ij} = 1/O_i$ and $P_{ij} = 1/d_i$, and the PA schemes. The row i, $i \in \{1, 2, 3\}$, corresponds to the spreading of one message initiated by each node from TBT_i to other $n = 100$ nodes. Each value of the triple is obtained by the values over all nodes and each simulation.

Comparing the methods, the SPREAD with $P_{ij} = 1/d_i$ demonstrates the triples with the smallest tick numbers which have a small variations for all TBTs. The SPREAD with $P_{ij} = 1/O_i$ and the PA work similar, but the PA is faster on the TBT_2 in case $(\alpha, \beta, \gamma) = (0.4, 0.2, 0.4)$. Both SPREAD strategies and the PA show that the best spreading nodes are in the TBT_1 subgraph with the smallest TI value of the out-degree. The PA and the SPREAD with $P_{ij} = 1/O_i$ are more sensitive to the parameter γ than the SPREAD with $P_{ij} = 1/d_i$. As larger γ as smaller minimum and average of their K^*.

4. Conclusions

We study the linear PA schemes to share one message from one node to a fixed number n of nodes of the network. The information spreading is investigated both for homogeneous (Section 3.1) and non-homogeneous (Section 3.2) graphs. We compare the PA and the well-known SPREAD algorithm on directed graphs with possible cycles and multiple edges generated by the PA with different sets of parameters.

Considering the homogeneous graphs studied in Section 3.1 one may conclude that the SPREAD algorithm that ignores directions of edges spreads the message faster than both the PA and the SPREAD algorithm that takes the edge directions into account. However, the PA may be the best spreader for some sets of its parameters (α, β, γ). Other parameters $(\Delta_{in}, \Delta_{out})$ were taken constant in our simulation study.

Regarding the non-homogeneous graphs (see Section 3.2) consisting of the TBT subgraphs with different tail indexes of the in- and out-degrees, we found that the

node from the TBT with the smallest tail index of the out-degrees, i.e. having the heaviest tail distribution of the out-degrees, spreads its message faster than nodes from the TBTs with the larger tail index of the out-degrees, i.e. with the lighter tail distribution of the out-degrees. The increasing of the parameter γ leads to the decreasing of the number of clock ticks required to share the message to arbitrary n nodes and hence, to the increasing of the spreading rate.

REFERENCES

1. Newman M. E. J., Networks: An Introduction // Oxford University Press, Second edition, 2018.
2. Avrachenkov K., and Lebedev D., PageRank of scale-free growing networks // Internet Mathematics. 2006. V. 3(2), P. 207-231.
3. Wan P., Wang T., Davis R. A., Resnick S.I., Are extreme value estimation methods useful for network data?// Extremes. 2020. V. 23 P. 171-195.
4. Mosk-Aoyama D., Shah D., Computing separable functions via gossip // In Proceedings of the 25th ACM symposium on Principles of distributed computing (PODC '06), ACM, New York, USA. 2006. P. 113-122.
5. Censor-Hillel K., Shachnai H., Partial Information Spreading with Application to Distributed Maximum Coverage // In Proceedings of the 29th ACM symposium on Principles of distributed computing (PODC '10), ACM, New York, USA. 2010. P. 161-170.
6. Censor-Hillel K., Shachnai H., Fast information spreading in graphs with large weak conductance, SODA '11, 2011.
7. Bollobás B., Borgs C., Chayes J., Riordan O., Directed scale-free graphs // In Proceedings of the fourteenth annual ACM-SIAM symposium on Discrete algorithms (SODA '03), Society for Industrial and Applied Mathematics, USA. 2003. P. 132–139.
8. Chen, N., Litvak, N., Olvera-Cravioto, M., PageRank in scale-free random graphs, , Memorandum of the Department of Applied Mathematics, Department of Applied Mathematics, University of Twente, V. 2046. 2015.
9. de Haan, L., Ferreira, A., Extreme Value Theory: An Introduction. Springer, 2006.

UDC: 519.872, 519.217

Single-server queuing systems with exponential service times and threshold-based renovation

Viana C. C. Hilquias[1], I. S. Zaryadov[1,2], T. A. Milovanova[1]

[1]Department of Applied Probability and Informatics, Peoples' Friendship University of Russia (RUDN University), 6 Miklukho-Maklaya St, Moscow, 117198, Russian Federation

[2]Institute of Informatics Problems, Federal Research Center "Computer Science and Control" of the Russian Academy of Sciences, 44-2 Vavilov Str., Moscow 119333, Russian Federation

hilvianamat1@gmail.com, zaryadov-is@rudn.ru, milovanova-ta@rudn.ru

Abstract

In this paper we study two types of $GI/M/1$ infinite capacity queues with the implemented threshold-based renovation mechanism. As usual renovation implies probabilistic dropping of customers from the queue upon service completions. In the systems of the first type there the threshold value (indication the queue length) controls the activation of the renovation mechanism. In the systems of the second type the threshold value not only triggers the renovation, but also specifies the area in the queue wherefrom the customers cannot be dropped. For both types of systems the main stationary characteristics are obtained. Numerical results are also provided, which illustrate the performance of the queues.

1. Introduction

According to [1], the development of modern mechanisms for active queue management (AQM) keeps attracting attention from the operation research community. An AQM is usually based on a rule (algorithms such as random early detection (RED) [2], Explicit Congestion Notification (ECN), controlled delay (CoDel) and others) for intelligent dropping of packets from the buffer (queue) as its saturation level increases. Numerous AQM schemes have already been proposed [3]. In the vast majority of cases, their performance analysis is performed by simulation (for

This paper has been supported by the RUDN University Strategic Academic Leadership Program (Viana C. C. Hilquias, T.A. Milovanova and I.S. Zaryadov, mathematical model formulation and simulation model development). Also the publication has been funded by Russian Foundation for Basic Research (RFBR) according to the research project No. 19-07-00739 (T.A. Milovanova and I.S. Zaryadov, mathematical model development and numerical analysis).

example, [4]) and the bridges between the available use-case results and analytic results, as well as between the available analytic results are very few (see, for example, [5, 6, 7]). In this paper we present some new results of the ongoing research of queuing systems with renovation as the AQM. Unlike RED type AQMs renovations is based on completely different idea: the decision about a possible dropping is synchronized with the service completions (see [8, 9, 10, 11]). Here we elaborate further on the mechanism of renovation and describe two new settings. In the first setting we consider the single-server queue with the threshold, which determines the boundary in the queue, starting from which the dropping of customers begins. The second setting covers the case when the threshold value also specifies the area in the queue, wherefrom the customers cannot be dropped.

The structure of the paper is as follows. Section 2 presents the main results for the queuing system under the first setting, section 3 is devoted to the system under the second setting. Simulation results are presented in the section 4. The last section concludes the paper with the short discussion.

2. First setting

Consider the $GI/M/1/\infty$ queuing system, shown in the Fig. 1, with the implemented renovation [8, 9, 11] mechanism and a threshold value Q_1, which determines the boundary in the queue, starting from which the dropping of customers begins.

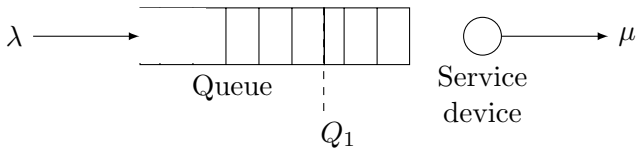


Fig. 1. Queuing system type 1

2.1. System of equilibrium equations. Let π_i be the steady-state probability (of the embedded Markov chain) that at the time of a new request arrival in the system will be exactly i packets. Then the system of equilibrium equations for steady-state probabilities will have the form (1):

$$\pi_0 = \sum_{i=0}^{\infty} \pi_i A^*_{i+1}, \quad \pi_i = \sum_{j=i-1}^{\infty} \pi_j A_{j-i+1}, \tag{1}$$

where

$$A_{i+1-j} = \int_0^{\infty} p^{i-Q_1} \frac{(\mu x)^{i+1-j}}{(i+1-j)!} e^{-\mu x} dA(x), 0 < j \leq Q_1, \tag{2}$$

for the case, when the current queue length has not exceeded the threshold value, and

$$A_{i+1-j} = \int_0^\infty \frac{(p\mu x)^{i+1-j}}{(i+1-j)!} e^{-\mu x} dA(x), \tag{3}$$

if the threshold value Q_1 is exceeded. The probabilities A_j^* are determined by the formula:

$$A_{i+1}^* = 1 - \sum_{j=0}^{i+1} A_j, \quad i \geq 0. \tag{4}$$

It is not difficult to prove that $\pi_i = \pi_{Q_1+1} \cdot g^{i-Q_1-1}$, where $g = \alpha(\mu(1-pg)), g \in (0,1)$, here $\alpha(s)$ is the Laplace-Stieltjes transform for the incoming flow distribution function.

2.2. The service probability and the loss probability for a received packet. Let $p^{(serv)}$ be the probability that incoming packet (request) will be served and let $p^{(loss)}$ be the probability that the received in the system packet will be dropped by renovation mechanism.

The probability $p^{(serv)}$ is determined by the formula (5)

$$p^{(serv)} = \pi_0 + \sum_{i=0}^{\infty} \pi_{i+1} \int_0^\infty p_{i,0}^{(serv)}(x) dx, \tag{5}$$

where for $i + j \leq Q_1$

$$p_{i,j}^{(serv)}(x) = \frac{(\mu x)^{i+1}}{(i+1)!} e^{-\mu y} \overline{A}(x) + \int_0^y \sum_{k=0}^{i} \frac{(\mu y)^k}{k!} e^{-\mu y} dA(y) p_{i-k,j+1}^{(serv)}(x,y), \tag{6}$$

and for $i + j > Q_1$

$$p_{i,j}^{(serv)}(x) = \overline{A}(x) \cdot \frac{(\mu x)^{i+1}}{(i+1)!} e^{-\mu x} \cdot p^{min(i+1, i+j+1-Q_1)} + \\ + \int_0^x \sum_{k=0}^{i} \frac{(\mu y)^k}{k!} e^{-\mu y} \cdot p_{k,j}^{(sevr)} dA(y) p_{i-k,j+1}^{(serv)}(x-y). \tag{7}$$

Similarly, we may determine the probability $p^{(loss)}$ by the formula (8)

$$p^{(loss)} = \sum_{i=1}^{\infty} \pi_i \int_0^\infty p_{i-1,0}^{(loss)}(x) dx, \tag{8}$$

where $p_{i,j}^{(loss)}(x)$ are defined similarly to (6) and (7).

2.3. Time characteristics for a served packet and a dropped packet. Let $W^{(serv)}(x)$ and $W^{(loss)}(x)$ be the distribution functions of the time spent in the queue by the served and dropped packets. Then

$$W^{(serv)}(x) = \frac{1}{p^{(serv)}} \sum_{i=0}^{\infty} W_{i,0}^{(serv)}(x)\pi_i, \tag{9}$$

where $W_{i,j}^{(serv)}(x)$ — distribution function of the time spent in the queue by the served packet, if there are i other packets in the queue before the considered request and there are j other packets after it. For densities $w_{i,j}^{(serv)}(x) = \left(W_{i,j}^{(serv)}(x)\right)'$, we obtain:

$$w_{i,j}^{(serv)}(x) = \overline{A}(x)\cdot\frac{\mu^{i+1}x^i}{i!}e^{-\mu x} + \int_0^x \sum_{k=0}^{i} \frac{(\mu y)^k}{k!}e^{-\mu y}dA(y)\cdot w_{i-k,j+1}^{(serv)}(x-y), \quad i+j \le Q_1, \tag{10}$$

$$w_{i,j}^{(serv)}(x) = \overline{A}(x)\cdot\frac{\mu^{i+1}x^i}{i!}e^{-\mu x}p_{i+1,j}^{(serv)} + \\ + \int_0^x \sum_{k=0}^{i} \frac{(\mu y)^k}{k!}e^{-\mu y}p_{k,j}^{(serv)}dA(y)\cdot w_{i-k,j+1}^{(serv)}(x-y), \quad i+j > Q_1, \tag{11}$$

$$p_{i+1,j}^{(serv)} = \begin{cases} p^{i+1}, & j \ge Q_1, \\ p^{i+j+1-Q_1}, & j < Q_1. \end{cases} \tag{12}$$

Similarly to $W^{(serv)}(x)$ the distribution function $W^{(loss)}(x)$ of the time spent in the queue by the dropped packet may be obtained.

3. Second setting

Consider now the $GI/M/1/\infty$ system, shown in the figure 2, where the threshold value Q_1 defines not only the boundary in queue, upon exceeding which by the current queue length the renovation mechanism is activated, but also specifies the area in the queue, wherefrom the customers cannot be dropped

3.1. System of equilibrium equations. Just as in 2.1, let π_i be the steady-state probability (of the embedded Markov chain) that in the system there is exactly

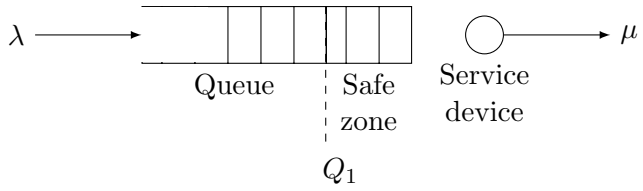


Fig. 2. Queuing system type 2

i packets at the time of a new request arrival, and system of equilibrium equations has the form:

$$\pi_0 = \sum_{i=0}^{\infty} \pi_i \tilde{A}_{i+1}, \quad \pi_i = \sum_{j=i-1}^{\infty} \pi_j A_{j-i+1}, \tag{13}$$

where

$$A_{i+1-j} = \frac{-(1)^{i+1-j}\mu^{i+1-j}}{(i+1-j)!}\alpha^{(i+1-j)}(\mu), \quad 0 \le i \le Q_1 + 1, \tag{14}$$

$$A_{i+1-j} = \frac{(-p\mu)^{i+1-j}}{(i+1-j)!}\alpha^{i+1-j}(\mu), \quad i > Q_1 + 1, j \ge Q_1 + 1, \tag{15}$$

$$A_{i+1-Q_1} = \frac{(-\mu p)^{i+1-Q_1}}{(i+1-Q_1)!}\alpha^{(i+1-Q_1)}(\mu) + \\ + \sum_{k=1}^{i+1-Q_1} \frac{(-\mu)^k}{k!} \cdot p^{k-1} q \alpha^k(\mu), \quad i > Q_1 + 1, j = Q_1, \tag{16}$$

$$A_{i+1-j} = \int_0^{\infty} \int_0^{x} A_{i+1,Q_1}(y) dy A^*_{Q_1,j}(x-y) dA(x), \quad i > Q_1 + 1, 0 < j < Q_1, \tag{17}$$

where

$$A_{i,Q-1}(y) = \frac{(\mu y)^{i+1-Q_i}}{(i+1-Q_i)!} e^{-\mu y} p^{i+1-Q_i} + \sum_{k=1}^{i+1-Q_i} \frac{(\mu y)^k}{k!} e^{-\mu y} p^{k-1} q, \tag{18}$$

$$A^*_{Q_1,j}(x-y) = \frac{(\mu(x-y))^{Q_1-j}}{(Q_1-j)!} e^{-\mu(x-y)}. \tag{19}$$

The probabilities $\tilde{A}_i$ are defined by the formula

$$\tilde{A}_{i+1} = 1 - \sum_{j=0}^{i+1} A_j, \quad i \ge 0. \tag{20}$$

It is also not difficult to prove that $\pi_i = \pi_{Q_1+1} \cdot g^{i-Q_1-1}$ where $g = \alpha(\mu(1-pg)), g \in (0,1)$.

3.2. Service and loss probability for the incoming packet. Let $p^{(serv)}$ be the probability that the packet in the system will be served and $p^{(loss)}$ — the probability that the packet in the system will be dropped.

Then

$$p^{(serv)} = 1 - \pi_{Q_1+1} \cdot \frac{q}{(1-g)(1-pg)}, \quad p^{(loss)} = \pi_{Q_1+1} \frac{q}{(1-g)(1-gp)}, \tag{21}$$

where q — drop probability.

3.3. Time Characteristics of Queuing System. In the terminology of the section 2.3 and in terms of the Laplace-Stieltjes transformation, we get that

$$\omega^{serv}(s) = \frac{1}{p^{(serv)}} \left(\sum_{i=0}^{Q_1} \left(\frac{\mu}{\mu+s} \right)^i \pi_i + p \left(\frac{\mu}{\mu+s} \right)^{Q_1+1} \pi_{Q_1+1} \frac{\mu+s}{\mu+s-p\mu g} \right). \tag{22}$$

The average waiting time for a served packet is:

$$w^{(serv)} = \frac{1}{p^{(serv)}} \left(\sum_{i=0}^{Q_1} \frac{i}{\mu} \pi_i + p\pi_{Q_1+1} \left(\frac{Q_1+1}{\mu - p\mu + g} + \frac{p^2}{\mu(1-pg)^2} \right) \right). \tag{23}$$

For a dropped packet:

$$\omega^{(loss)}(s) = \frac{1}{p^{(loss)}} \cdot \frac{q\pi_{Q_1+1}}{1-g} \cdot \frac{\mu}{\mu+s-\mu pg}. \tag{24}$$

4. Simulation results

Below (see table 1) is presented a table with simulation results (GPSS simulations for both systems (sys.1 and sys.2) were performed with the following initial parameters: threshold value $Q_1 = 30$, arrival rate — 1 task per 2 unit of time, service rate — 1 task per 6 unit of time, and the simulation time is 100000 unit of time) for different drop probabilities.

Thus, at a very high system load ($\rho = 3$) and at low values of the renovation probability q, the number of dropped and the number of serviced claims is approximately the same for both models, but the the average queue length and the maximum queue length are 7-12% greater for the second as well as the average waiting time of a task in the queue.

Drop probability		**0.0025**	**0.005**	**0.01**	**0.025**	**0.05**
Generated tasks	**sys.1**	50113	50502	50289	49880	50014
	sys.2	50240	49848	50045	49830	50022
Serviced tasks	**sys.1**	16931	16413	16720	16554	16566
	sys.2	16766	16771	16615	16570	16698
Serviced tasks without calling the renv. mech.	**sys.1**	559	1176	2196	4718	6994
	sys.2	30	46	78	106	244
Dropped tasks	**sys.1**	32023	33507	33277	33222	33413
	sys.2	32105	32789	33256	33172	33286
Probability of servicing tasks	**sys.1**	0.3379	0.3250	0.3325	0.3319	0.3312
	sys.2	0.3337	0.3364	0.3320	0.3325	0.3338
Probability of dropping tasks	**sys.1**	0.6390	0.6635	0.6617	0.6650	0.6681
	sys.2	0.6390	0.6578	0.6645	0.6667	0.6654
Average queue length	**sys.1**	840	431	196	86	46
	sys.2	735	463	212	111	70
Maximum queue length	**sys.1**	3942	2136	1388	471	281
	sys.2	2767	2364	1564	616	335
Average waiting time of a task in the queue	**sys.1**	1463	855	392	172	94
	sys.2	1676	929	423	223	140

Table 1. Simulation results for different drop probabilities

5. Conclusion

In this paper, we considered two types of a single-server queuing system with an infinite capacity storage, with renovation mechanism and a threshold value. For each system, analytical expressions for the distribution of the number of applications in the system were found, expressions for calculating the time and probability characteristics of the systems were obtained, and the obtained simulation results in the GPSS system were compared.

REFERENCES

1. Baker F., Fairhurst G. IETF Recommendations Regarding Active Queue Management. RFC 7567. Internet Engineering Task Force. `https://tools.ietf.org/html/rfc7567`. Last accessed 29 April 2021.
2. Floyd S., Jacobson V. Random Early Detection Gateways for Congestion Avoidance // IEEE/ACM Transactions on Networking. 1993. V. 4 (1). P. 397–413.

3. Adams R. Active Queue Management: A Survey. // IEEE Communications Surveys & Tutorials. 2013. V. 15 (3). P. 1425–1476.
4. Menth M., Veith S. Active Queue Management Based on Congestion Policing (CP-AQM) // In: Measurement, Modelling and Evaluation of Computing Systems. MMB 2018. Lecture Notes in Computer Science. 2018. V. 10740. P. 173–187.
5. Chydzinski A., Chrost L. Analysis of AQM queues with queue size based packet dropping // International Journal of Applied Mathematics and Computer Science. 2011. V. 21 (3). P. 567–577.
6. Chydzinski A., Mrozowski P. Queues with dropping functions and general arrival processes // PLoS ONE. 2016. V. 11 (3). https://doi.org/10.1371/journal.pone.0150702
7. Konovalov M. G., Razumchik R. V. Numerical Analysis of Improved Access Restriction Algorithms in a $GI|G|1|N$ System // Journal of Communications Technology and Electronics. 2018. V. 63 (6). P. 616–625.
8. Kreinin A. Queueing Systems with Renovation // Journal of Applied Math. Stochast. Analysis. 1997. V. 10 (4). P. 431–443.
9. Bocharov P. P., Zaryadov I. S. Probability Distribution in Queueing Systems with Renovation // Bulletin of Peoples' Friendship University of Russia. Series ''Mathematics. Information Sciences. Physics". 2007. No. 1-2. P. 15–25.
10. Konovalov M., Razumchik R. Comparison of two active queue management schemes through the M/D/1/N queue // Informatika i ee Primeneniya (Informatics and Applications). 2018. V. 12 (4). P. 9–15.
11. C. C. Hilquias Viana, I. S. Zaryadov, T. A. Milovanova, V. V. Tsurlukov, A. V. Korolkova, D. S. Kulyabov, The General Renovation as the Active Queue Management Mechanism. Some Aspects and Results, in: Communications in Computer and Information Science, vol. 1141, 488-502, 2019. doi:10.1007/978-3-030-36625-4_39.

UDC: 519.872

Joint stationary distribution in the two-channel queueing system with ordered entry, governed by one queue skipping policy

R.V. Razumchik[1]

[1]Federal Research Center "Computer Science and Control" of the Russian Academy of Sciences, Moscow 119333, Russian Federation

rrazumchik@ipiran.ru

Abstract

Consideration is given to the queueing system with the ordered entry, which is governed by one special queue skipping policy. The system consists of two single-server queues, say Q1 and Q2, each with infinite capacity. New customers arrive only in batches and only to Q1. Upon arrival of a batch its size is compared with the current total number of customers in Q1. If the size of the batch is larger than that number, all customers residing in the system (including the one in server) are pushed-out to Q2 and the arrived batch enters the system; otherwise the new batch goes to Q2. Whenever a batch arrives to Q2 the same comparison is performed. The batch pushed-out from Q2 is considered as lost. Under the assumption that the service times are exponential and the batch inter-arrival times are i.i.d. we sketch the procedure for the computation of the joint stationary distribution of the queues' content. Obtaining stability criteria as well as closed-form expressions remain the open issue.

Keywords: batch arrivals, queue skipping policy, ordered entry

1. Introduction

In the recent paper [1] the authors have considered the $M/G/1$ batch-arrival queue with the special admission control called the queue skipping policy. The motivation behind this policy is the maximization of the system utilization. This is in sharp contrast with the majority of the studies in the operation research community, since the latter are usually devoted to the minimization of waiting time (or queue size) related quantities. Roughly speaking (see the detailed description in the next section) the considered queue skipping policy implies that system works only on the

The reported study was funded by RFBR, project number 20-07-00804. The research was conducted in accordance with the program of Moscow Center for Fundamental and Applied Mathematics.

biggest batches. Thus is if a bigger bath (than the currently served) arrives, it forces the drop of the smallest batch and occupies the system itself. Since the customers are served from the queue one-by-one, the size of the batch in the queue is changed upon each service completion. This is the main reason why the analytic analysis of queues with such a policy under general assumptions about the arrival and service times is a challenge. For the latest results under some special assumptions the reader can refer, for example, to [1, 2].

So far in the analysis of such queues it was assumed that they operate in isolation. As the consequence, dropped batches were considered as lost. In this paper we try to make a step further and add to the system one more single-server queue with a queue skipping policy. In what follows we sketch the derivation of the joint steady-state distribution of the queues' content and reveal some of the challenges in its analysis. As it will be seen, the main challenge — obtaining closed-form expressions — remains an open issue.

It is worth noticing here, that under the queue skipping policy new arrivals may push out customers in the queue. But nevertheless these systems cannot be reduced to queues with negative arrivals/signals [3, 4]. Moreover, even though the order in which the customers join the queues is specified, the analytic results available for ordered entry queues (see, for example, [5, 6, 7]) are of no use here.

2. System description and problem statement

Consider the system consisting of two single-server queues (say, Q1 and Q2), running in parallel. The capacity of each queue is infinite. New customers arrive only to Q1 and arrive in batches of random size B. The distribution $b_k = \mathsf{P}\{B = k\}$, $k \geq 1$, is considered to be known. The batch inter-arrival times are i.i.d. with the known cumulative distribution function $A(x)$ and finite mean $a = \int_0^\infty x dA(x)$.

When a batch arrives to Q1 its size, say X, is compared to the current total number of customers in Q1, say Y. Clearly, $Y \geq 0$. If $X > Y$ then all the customers in Q1 are (instantly) removed from it and newly arrived batch X is all placed in Q1, with the first customer in the job entering the server. Those Y customers, which were removed from Q1, arrive (instantly) in a single batch to Q2 and see $Z \geq 0$ other customers in it. If $Y > Z$ then all Z customers are (instantly) removed from Q2 (and are considered to be lost) and Y tasks are all placed in Q2 with the first customer in the job entering the server. If $Y \leq Z$ then Y customers leave the network having no further effect on Q2 and, of course, Q1. Finally, if $X \leq Y$ then all X tasks leave Q1, without entering it, and arrive at Q2. If $X > Z$ then then Z customers are (instantly) removed from Q2 (and are considered to be lost) and X tasks are all placed in Q2 with the first customer in the batch entering the server. If $X \leq Z$ then X customers leave the system having no effect on it. Each server serves only

one customer at a time according to the FIFO discipline. The service time of a task in Q1 is exponentially distributed size with the parameter μ_1 and in Q2 — with the parameter μ_2.

Let $\xi(t)$ denote the total number of customers in Q1 at time t and $\eta(t)$ denote the total number of customers in Q2 at time t. Under the assumption that the stationary regime of the described system exists, the problem is to find $\pi(i,j) = \lim_{t\to\infty}\pi(t;i,j)$, where

$$\pi(t;i,j) = \mathsf{P}\{\xi(t) = i, \eta(t) = j\},\ i \geq 0,\ j \geq 0.$$

3. Joint stationary distribution

Since instants of batch arrivals to Q1 are renewal instants, one can use the embedded Markov chain technique to compute $\pi(i,j)$. Let τ_n be the time instant of the arrival of the n^{th} batch to Q1. Let ξ_n and η_n be the total number of customers in Q1 and in Q2 respectively immediately after τ_n (i.e. after the possible customers' removals from Q1 and/or Q2). The sequence $\{(\xi_n, \eta_n),\ n \geq 0\}$ constitutes the embedded Markov chain with the state space $\mathcal{X} = \{(i,j), i \geq 1, j \geq 0\}$. State (i,j) means that (at the instant right after some τ_n) there are i customers in Q1, and there are j customers in Q2. States $(0,0), (0,1), (0,2), \ldots$ can never happen in this Markov chain. Assuming that the stationary distribution of $\{(\xi_n, \eta_n),\ n \geq 0\}$ exists we denote it by $p_{i,j}$ i.e. $p_{i,j} = \lim_{n\to\infty}\mathsf{P}\{\xi_n = i, \eta_n = j\}$. One way to compute $p_{i,j}$ is to use Chapman–Kolmogorov equations. Indeed, denote by $\mathbb{P}$ such a transition probability matrix, which entry $[\mathbb{P}]_{(n,m),(i,j)}$ is the probability that between two consecutive arrivals the total number of customers in Q1 and in Q2 will drop down from (n,m) to (i,j). Then, for example, for the probabilities $p_{i,0}$ we have the equations:

$$p_{i,0} = b_i \sum_{n=1}^{\infty}\sum_{m=0}^{\infty} p_{n,m}[\mathbb{P}]_{(n,m),(0,0)},\ i \geq 1.$$

By analogy one can obtain the system for $p_{i,j}$, $j \geq 1$, and then solve it altogether numerically. In order to make sure that the matrix $\mathbb{P}$ can be computed let us sketch the procedure for the entry $[\mathbb{P}]_{(1,0),(0,0)}$. Consider the two-state (with states states $(1,0)$ and $(0,0)$, state $(0,0)$ being absorbing) continuous time Markov chain with the rate matrix

$$\mathbb{Q}_{1,0} = \begin{pmatrix} -\mu_1 & \mu_1 \\ 0 & 0 \end{pmatrix} = \begin{pmatrix} \tilde{\mathbb{Q}}_{1,0} & \vec{q}_{1,0} \\ 0 & 0 \end{pmatrix}$$

and the initial distribution $(1,0)$. Denote by $T_{1,0}$ the time to absorption in this Markov chain. Then

$$\mathsf{P}\{T_{1,0} < x\} = 1 - (1,0)e^{\tilde{\mathbb{Q}}_{1,0}x}\vec{1}. \quad (1)$$

where $\vec{1}$ denotes the vector of ones. Having (1), one can compute

$$[\mathbb{P}]_{(1,0),(0,0)} = \int_0^\infty \mathsf{P}\{T_{1,0} < x\} dA(x). \tag{2}$$

Theoretically it is the way to determine all $[\mathbb{P}]_{(n,m),(i,j)}$. Another unified way to determine all $[\mathbb{P}]_{(n,m),(i,j)}$ is to use the structured Markov chains. The idea is the following. Assume that at time $t = 0$ the system is in the state (n, m). When the next arrival occurs, the system may be only in the state (i, j), where $0 \le i \le n$ and $0 \le j \le m$. Thus we have the quasi-death process, say $\chi(t)$, with the finite state space $\{(i, j),\ 0 \le i \le n,\ 0 \le j \le m\}$. We can take i as the level of the process $\{\chi(t), t \ge 0\}$ and j as its phase. If we do so, its rate matrix has the form

$$\mathbb{Q} = \begin{pmatrix} \mathbb{M}_2 & 0 & 0 & \dots & 0 & 0 \\ \mathbb{M}_1 & \mathbb{M} & 0 & \dots & 0 & 0 \\ 0 & \mathbb{M}_1 & \mathbb{M} & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & 0 \\ 0 & 0 & 0 & \dots & \mathbb{M} & 0 \\ 0 & 0 & 0 & \dots & \mathbb{M}_1 & \mathbb{M} \end{pmatrix},$$

$$\mathbb{M}_2 = \begin{pmatrix} 0 & 0 & 0 & \dots & 0 & 0 \\ \mu_2 & -\mu_2 & 0 & \dots & 0 & 0 \\ 0 & \mu_2 & -\mu_2 & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & 0 \\ 0 & 0 & 0 & \dots & -\mu_2 & 0 \\ 0 & 0 & 0 & \dots & \mu_2 & -\mu_2 \end{pmatrix}, \quad \mathbb{M}_1 = \begin{pmatrix} \mu_1 & 0 & 0 & \dots & 0 & 0 \\ 0 & \mu_1 & 0 & \dots & 0 & 0 \\ 0 & 0 & \mu_1 & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & 0 \\ 0 & 0 & 0 & \dots & \mu_1 & 0 \\ 0 & 0 & 0 & \dots & 0 & \mu_1 \end{pmatrix},$$

$$\mathbb{M} = \begin{pmatrix} -\mu_1 & 0 & 0 & \dots & 0 & 0 \\ \mu_2 & -(\mu_1+\mu_2) & 0 & \dots & 0 & 0 \\ 0 & \mu_2 & -(\mu_1+\mu_2) & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & 0 \\ 0 & 0 & 0 & \dots & -(\mu_1+\mu_2) & 0 \\ 0 & 0 & 0 & \dots & \mu_2 & -(\mu_1+\mu_2) \end{pmatrix}.$$

The matrices $\mathbb{M}_2$, $\mathbb{M}_1$ and $\mathbb{M}$ are square of size $(m + 1)$. Denote

$$q_{i,j}(t) = \mathsf{P}\{\chi(t) = (i, j) | \chi(0) = (n, m)\},$$

$$\vec{q}(t) = (q_{00}(t), \dots, q_{0,m}(t), q_{10}(t), \dots, q_{1,m}(t), \dots, q_{n0}(t), \dots, q_{n,m}(t)).$$

Then the time-dependent (conditional) probability distribution $\vec{q}(t)$ can be computed by

$$\vec{q}(t) = \underbrace{(0,0,\ldots,0,1)}_{1\times(n+1)(m+1)}\mathbb{Q}e^{\mathbb{Q}t} = (0,0,\ldots,0,1)e^{\mathbb{Q}t}\mathbb{Q}, \quad (3)$$

and the probability $[\mathbb{P}]_{(n,m),(i,j)}$ is the $(i,j)^{th}$ entry of the vector

$$\int_0^\infty \vec{q}(t)dA(t) = (0,0,\ldots,0,1)\mathbb{Q}\int_0^\infty e^{\mathbb{Q}t}dA(t). \quad (4)$$

Thus in order to determine all $[\mathbb{P}]_{(n,m),(i,j)}$ one has to compute for each pair (n,m) the vector (4). This is a costly operation but some simplifications, which make it feasible, can be suggested [8, 9].

Once the $[\mathbb{P}]_{(n,m),(i,j)}$ are available, $\pi(i,j)$ are computed using the known results for the Markov regenerative processes, which say that

$$\pi(i,j) = \frac{1}{a}\sum_{n=1}^{\infty}\sum_{m=0}^{\infty} p_{n,m}f_{(n,m),(i,j)},$$

where $f_{(n,m),(i,j)}$ is the mean time (between two consecutive arrivals), which the system spends in the state (i,j), provided that after the last arrival to Q1 the system was in the state (n,m). Computation of $f_{(n,m),(i,j)}$ is challenging as can be already seen from the expression for $f_{(1,0),(0,0)}$:

$$\begin{aligned} f_{(1,0),(0,0)} &= \int_0^\infty dA(x)\int_0^x (x-u)\mathsf{P}\{u \le T_{1,0} < u+du\} = \\ &= -(1,0)\int_0^\infty dA(x)\int_0^x (x-u)\tilde{\mathbb{Q}}_{1,0}e^{\tilde{\mathbb{Q}}_{1,0}u}du\vec{1}. \end{aligned}$$

REFERENCES

1. Marin A., Rossi S. A queueing model that works only on biggest jobs // European Workshop on Performance Engineering. Ed. M. Gribaudo, M. Iacono, T. Phung-Duc, R. Razumchik. Lecture Notes in Computer Science ser. Springer. 2020. V. 12039. P. 118–132.
2. Matyushenko S. I., Razumchik R. V. Stationary characteristics of discrete-time $Geo/G/1$ queue with batch arrivals and one queue skipping policy // Inform. Primen., 2020. V. 14. No. 4. P. 25–32. doi:10.14357/19922264200404
3. Dudin A.N., Klimenok V.I., Vishnevsky V.M. The theory of queuing systems with correlated flows. Heidelberg, Germany: Springer, 2019. 447 p.

4. Gelenbe E. R'eseaux stochastiques ouverts avec clients negatifs and positifs, et reseaux neuronaux // Comptes-Rendus de l'Academie des Sciences. 1989. V. 309. Serie II. P. 972–982.
5. Disney R. L. Some multichannel queueing problems and ordered entry // J. Ind. Eng. 1962. V. 13. P. 46–48.
6. Nanwijn W. M. A note on many-server queueing system with ordered entry, with an application to conveyor theory // J. Appl. Prob. 1983. V. 20. No. 1. P. 144–152.
7. Zaryadov I. S., Meykhanadzhyan L. A., Milovanova T. A., Razumchik R. V. On the method of calculating the stationary distribution in the finite two-channel system with ordered input // Systems and Means of informatics. 2015. V. 25. No. 3. P. 44–59.
8. Moler C., Van Loan C. Nineteen dubious ways to compute the exponential of a matrix: Twenty-five years later // SIAM Review. 2003. V. 45. No. 1. P. 3–49.
9. Powell P.D. Calculating determinants of block matrices // arXiv:1112.4379 [math.RA] 2011. `https://arxiv.org/pdf/1112.4379.pdf`.

UDC: 621.391

Enhancing the Resource Sharing Capabilities of a Network by Deploying Network Slicing Procedure

M.S. Stepanov[1], S.N. Stepanov[2], Umer Andrabi[3], D.S. Petrov[4], Juvent Ndayikunda[5]

[1,2,5]Moscow Technical University of Communications and Informatics, Department of communication networks and commutation systems, 8A, Aviamotornaya str., Moscow, 111024, Russia

[3,4]Moscow Institute of Physics and Technology (State University), 9 Institutskiy per., Dolgoprudny, Moscow Region, 141701, Russia

mihstep@yandex.ru, stpnvsrg@gmail.com, umer.andrabi@rediffmail.com, petrov.ds@phystech.edu, juvndayi@mail.ru

Abstract

An analytical framework to model the resource allocation procedures for transmission of multiservice traffic has been constructed and analyzed. The model consists of arbitrary number of traffic streams originating from different types of real time applications. All random variables used in the model have exponential distribution with corresponding mean values. Two different scenario of resource sharing has been taken into consideration for incoming traffic streams: Network Slicing, when resources are strictly divided among incoming traffic streams, and Filtering, when the access to resource is restricted depending on the amount of resource occupied by all traffic streams. It has been shown how to use both scenarios for creating conditions for differentiated servicing of heterogeneous traffic. Given numerical assessment proves that scenario based on filtering is more efficient to solve the formulated task in contrast to the analogous scenario based on slicing.

Keywords: network slicing, resource allocation and sharing, restricted access, recursive algorithm

1. Introduction

Earlier developed standards for serving multiservice traffic streams followed the "one size fits all" paradigm. Because of high level of diversity in traffic parameters and quality of service indicators such approach is not suitable for upcoming 5G era. This follows from the fact that implementation of such scenario leads to uncontrolled allocation of transmission resources in favor of traffic flows with relatively small data rate requirements. To overcome such challenges and create conditions for

differentiated servicing of heterogeneous traffic, the concept of network slicing has been introduced. This technique gives the rules to realize distribution of transmission resources in the form of separate logical network group (slice) of traffic with close resource requirements [1–3]. Grouping of the traffic streams can also be based on the achievement of certain indicators of economic efficiency and in some cases the "slice per service" procedure may be applied, e.g. when a service requires special quality of servicing or uses unique scheme of charging by the CSP (Communications Service Provider) [4]. Another realization of the concept "slice per service" takes place when a couple of similar service instances are associated with the same service model created by the CSP during the service design phase [5, 6]. In this case a slice can be reserved per service but will serve many similar service instances.

In this paper two general scenarios for distribution of resources between slices are discussed: the static one and the dynamic one. Static scenario implies constant allocation of such resources as bandwidth for each slice with no reference to the changes of the incoming traffic flows. Dynamic (or partly dynamic) scenario implies alternating allocation of resources when the distribution of resources between the slices changes in response to the changes in the incoming traffic flows.

Positive features of static scenario are easy to notice. Firstly, a complete separation of network slices simplifies the procedure of slice configuration for the flow of requests of a specific type. Secondly, instances such as failures, overloads and network attacks that affects one slice will not affect the functionality of others. Thirdly, the static allocation of resources is a subject of relatively simple mathematical analysis. Negative features of static scenario are also easily predicted. Main among them is inefficient use of the resources reserved for each slice, which cannot be reallocated to other slices because of the principles of static allocation scenario [1–3, 7, 8].

In contrast to static distribution of resources is a dynamic resource distribution scenario. In dynamic allocation, available transmission resources are given to incoming traffic flows without taking into account traffic requirements and performance indicators [8, 9, 11]. This approach increases the efficiency of resource usage but at the expense of bad losses of requests for "heavy" traffic transmission. To combine positive features of static and dynamic scenarios various forms of dynamic allocation procedures with restricted access have already been suggested [8, 11]. These procedures are based on the idea to allocate some part of the available capacity for common use, and distribute the remaining part of the resources between chosen slices. This form of resource distribution requires more complex mathematical modeling, but such scenarios have more efficient form of utilization of network resources. Another positive feature of partly dynamic allocation procedures lies in the fact that they make the network more robust to handle the fluctuations of the incoming traffic [8,11].

In this paper we have constructed and analyzed an analytical framework to model the resource allocation procedures for transmission of multiservice traffic. Two scenarios of resource sharing has been considered for incoming traffic streams, these are: Network Slicing when resources are strictly divided among incoming traffic streams, and Filtering, when the access to resource is restricted depending on the amount of resource occupied by all traffic streams. The proposed model generalizes the results of [8, 11] by considering more efficient procedure of creating conditions for differentiated servicing of heterogeneous traffic.

2. Differentiated Servicing Based on the Network Slicing

The mathematical model of access node can be represented as a pool of v resource unit (r.u.) that are used for servicing n incoming Poisson flows of requests with intensities a_k, $k = 1, \ldots, n$. Here a_k is intensity of offered load for class k calls expressed in Erlangs. A request from kth flow uses b_k r.u. for the whole duration of connection. Without loss of generality we shall assume that the total holding time is exponentially distributed with the same mean value chosen to one.

Let $i_k(t)$ denote the number of calls from the kth flow served at time t. The model is described by n-dimensional markovian process of the type $r(t) = (i_1(t), \ldots, i_n(t))$ with state space S consisting of vectors $(i_1, \ldots, i_n)$, satisfying condition $i_1 b_1 + \ldots + i_n b_n \leq v$, where i_k is the number of calls from the kth flow being served under stationary conditions. Let $p(i_1, \ldots, i_n)$ denote the values of stationary probabilities of $r(t)$ and $p(i) = \sum_{i_1 b_1 + \cdots + i_n b_n = i} p(i_1, \ldots, i_n)$.

The process of transmission of kth flow, $k = 1, \ldots, n$, is described by π_k the ratio of lost requests and by m_k the mean number of occupied r.u.

$$\pi_k = \sum_{i=v-b_k+1}^{v} p(i), \quad m_k = a_k b_k (1 - \pi_k). \tag{1}$$

The most efficient calculation scheme for the introduced model is the recurrence algorithm [9]. The recurrence follows from the reversibility of $r(t)$. It gives the relations of detailed balance for state $(i_1, \ldots, i_n)$ in the form

$$p(i_1, \ldots, i_k, \ldots, i_n) i_k = p(i_1, \ldots, i_k - 1, \ldots, i_n) a_k. \tag{2}$$

After summing (2) for $(i_1, \ldots, i_n) \in S$ such as $i_1\, b_1 + \ldots + i_n\, b_n = i$, subsequent multiplication on b_k and summation over $k = 1, 2, \ldots, n$ we obtain

$$p(i)\, i = \sum_{k=1}^{n} a_k\, b_k\, p(i - b_k) I(i - b_k \geq 0), \quad i = 1, \ldots, v, \tag{3}$$

where function $I(\cdot)$ equals one, if the formulated condition is fulfilled else equals to zero. Consistent implementation of (3) gives values of $p(i)$, $i = 0, 1, \ldots, v$ and performance measures π_k, m_k of kth traffic stream, $k = 1, \ldots, n$.

The model and algorithms developed on such basis will be used to analysis conditions for differentiated servicing of heterogeneous traffic on a common pool of resource units. To simplify the investigation we limit our analysis to the case of conjoint servicing of $n = 2$ traffic streams. Traffic flows have fixed parameters of offered load a_k and required number of r.u. b_k, $k = 1, 2$. It is supposed that $b_1 \ll b_2$. It means that the first flow forms traffic stream with "light" requests, the second traffic stream with "heavy" requests. Conjoint servicing of "light" and "heavy" requests leads to the uncontrolled allocation of r.u. in favor of "light" requests. Let us implement the Network Slicing concept to create the conditions for differentiated servicing. We are considering two problem settings.

1) For given pool of v r.u. find the division of the resource into slices by equalizing the rate of losses of both flows.
2) Find the minimum value of v and the division of the common resource v into slices by providing the rate of losses for the first flow at a given advance level π_1^* and for the second flow at a given advance level π_2^*.

The solution of formulated problems are presented in Fig. 1 (the first task) and in Fig. 2 (the second task). The model input parameters are as follows: $v = 200$ r.u., $n = 2$, $b_1 = 1$ r.u., $b_2 = 10$ r.u., $a_1 = 100$ Erl; $a_2 = 10$ Erl. Let us denote by v_1 the number of r.u. in the first slice. The performance measures are calculated with the help of recursion (3). Numerical results show the possibility of using the Network Slicing concept for solving the formulated problems but later we will show that more efficient solution for differentiated servicing can be obtained by filtering the input flows. This results will be presented in Section 3.

3. Differentiated Servicing Based on the Filtering of the Input Flows

Let us consider the model of multiservice access node discussed in Section 2 and limit the access to servicing of requests from kth flow by filtering function $f_k(i)$. This function denotes the probability of acceptance of the request from kth flow when such request arrives into the state with i occupied r.u. We call this procedure as Filtering of requests arriving from kth flow. Let us denote requests from kth flow by λ_k the intensity of incoming calls, by $1/\mu_k$ the mean time of call servicing and by b_k the number of r.u. required for servicing of one call. Let us suppose that all random variables used for model description have exponential distributions with corresponding mean values.

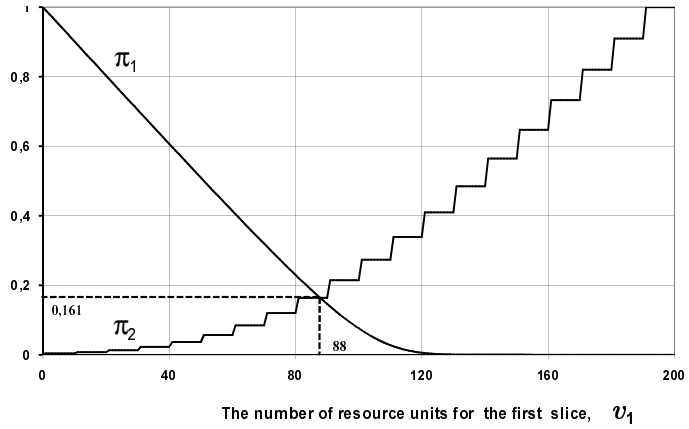


Fig. 1. The results of division of the common resource into slices equalizing the rate of losses of both flows for given value of v.

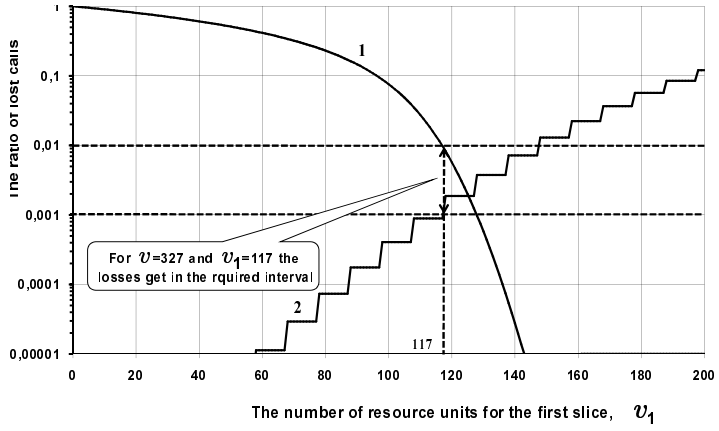


Fig. 2. The results of finding the minimum value of v and the division of the common resource into slices providing the rate of losses for the first flow at level $\pi_1^* = 0{,}01$ and for the second flow at level $\pi_2^* = 0{,}001$.

The model functioning is described by n-dimensional markovian process $r(t) = (i_1(t), \ldots, i_n(t))$ where $i_k(t)$ is the number of requests from the kth flow served at

time t. The state space S depends on the choice of filtering functions but for any choice the model states $(i_1, \ldots, i_n)$ should satisfy the inequality $\sum_{k=1}^{n} i_k b_k \le v$. Let us by $P(i_1, \ldots, i_n)$ denote the unnormalized values of stationary probabilities of $r(t)$. After normalization the values $p(i_1, \ldots, i_n)$ can be used for estimation of the portion of lost calls π_k and the mean number of occupied r.u. m_k. Assume that for state $(i_1, \ldots, i_n)$ the value i denotes the total number of occupied r.u. $i = i_1 b_1 + \cdots + i_n b_n$

$$\pi_k = \sum_{(i_1,\ldots,i_n)\in S} p(i_1, \ldots, i_n)\,(1 - f_k(i)), \quad m_k = \lambda_k b_k (1 - \pi_k)/\mu_k. \tag{4}$$

Because of filtering the markovian process $r(t)$ does not have a reversibility property that simplify the estimation of performance measures of the model introduced in Section 2. The values of $p(i_1, \ldots, i_n)$ can be found after normalization from solution the system of state equations by Gauss-Zeidel iteration algoritm

$$P(i_1, \ldots, i_n) \sum_{k=1}^{n} (\lambda_k\, f_k(i) + i_k\, \mu_k) = \tag{5}$$

$$= \sum_{k=1}^{n} P(i_1, \ldots, i_k - 1, \ldots, i_n)\, \lambda_k\, f_k(i - b_k)\, I(i_k > 0) +$$

$$+ \sum_{k=1}^{n} P(i_1, \ldots, i_k + 1, \ldots, i_n)\, (i_k + 1)\, \mu_k\, I(i + b_k \le v).$$

Another way to estimate π_k and m_k is to use the approximate procedure. Such algorithm can be constructed if we suppose that reversibility property is valid for $r(t)$. We get the following relation (mark $\hat{}$ means the approximate character of result)

$$\hat{p}(i_1, \ldots, i_k, \ldots, i_n)\, i_k\, \mu_k = \hat{p}(i_1, \ldots, i_k - 1, \ldots, i_n)\, \lambda_k\, f_k(i - b_k). \tag{6}$$

These relations are similar to the detailed balance relations (2) obtained in Section 2. By using the similar concept that produced the recursive formula (3) we can get the same type of recursion for this particular model based on filtering

$$\hat{P}(i) = \frac{1}{i} \times \sum_{k=1}^{n} \lambda_k\, b_k \hat{P}(i - b_k)\, f_k(i - b_k)/\mu_k, \tag{7}$$

here $\hat{p}(i) = \sum_{i_1 b_1 + \ldots + i_n b_n = i} \hat{p}(i_1, \ldots, i_n)$. After realizing the recursions (7) the values $\hat{p}(i)$, $i = 0, 1, \ldots, v$ can be used for estimation $\hat{\pi}_k = \sum_{i=0}^{v} \hat{p}(i)\,(1 - f_k(i))$ and $\hat{m}_k = \lambda_k b_k (1 - \hat{\pi}_k)/\mu_k$.

The problems of resource planning to create the conditions for differentiated servicing formulated in Section 2 were solved by Network Slicing procedure. The obtained results are presented in Fig.1–2. Now we solve these problems by applying the procedure of Filtering to the input flows. The results are presented in Fig. 3 (the first task) and in Fig. 4 (the second task). The model input parameters that are used here are same as that were used for representation of data in Fig. 1–2. By applying the filtering procedure in the concerned model we can regulate the priority of servicing the requests of second ("heavy") stream by varying the value of r_1 the number of reserved r.u. in favor of requests of second flow. This can be done by choosing $f_1(i)$ as follows: $f_1(i) = 0$, $i \leq v - b_1 - r_1$; $f_1(i) = 1$, $i > v - b_1 - r_1$. The value of r_1 varies from 0 (no reservation) to some value that gives the right answer to the studied task. The performance measures are calculated by solving (5) using Gauss-Zeidel algorithm.

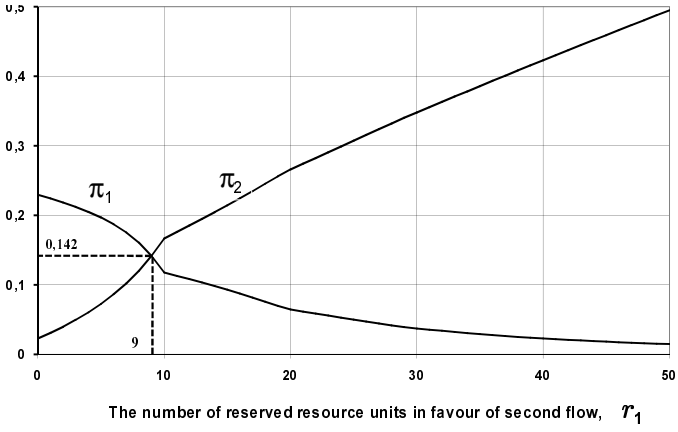


Fig. 3. The results of usage the Filtering for equalizing the rate of losses of both flows for given value of v.

Numerical results show that by using Filtering we can solve the formulated problems in a more efficient way than by using Network Slicing. This conclusion is drawn by comparing the data presented in Fig. 1 and Fig. 3, where we see that the Filtering equalize the rate of losses on a given volume of r.u. at lesser level than Network Slicing (0,161 for Slicing and 0,142 for Filtering). The same conclusion follows by comparing data presented in Fig. 2 and Fig. 4, where we see that by using Filtering we provide conditions for differentiated servicing at 10 % lesser volume of

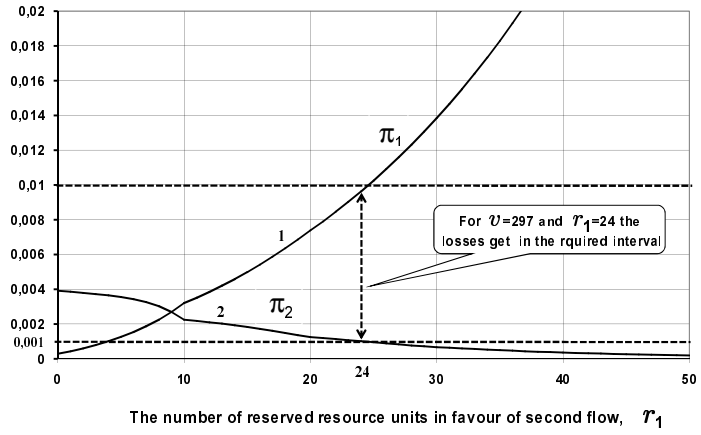


Fig. 4. The results of usage the Filtering for finding the minimum value of v that provide the rate of losses for the first flow at level $\pi_1^* = 0{,}01$ and for the second flow at level $\pi_2^* = 0{,}001$.

resource than by using Network Slicing (327 r.u. for Slicing and 297 r.u. for Filtering correspondingly).

The results presented in Fig. 2 and Fig. 4 are obtained for two conjoint traffic streams. The same ideas can be used in the case when $n > 2$. In this instance we can divide incoming flows into two groups with similar QoS indicators or used more complicated optimizing procedures.

4. Conclusion

An analytical framework for modeling resource allocation procedures for transmission of multiservice traffic was constructed and analyzed. Two scenarios of resource sharing for incoming traffic streams were considered: Network Slicing where resources are strictly divided among incoming traffic streams, and Filtering, where the access to resource is restricted depending on the amount of resource occupied by overall traffic streams. It was shown how model can be used to create conditions for differentiated servicing of heterogeneous traffic. The numerical assessment showed that simplest form of Network Slicing scenario has number of drawbacks. But the main drawback is additional requirement for required number of resource units (r.u) to serve incoming request flows with the required quality in contrast to the scenario where shared resources are not divided in separate slices. Also this method of resource sharing is highly sensitive to the change in the values of the offered load. The paper concludes

by highlighting the fact that these drawbacks can be reduced by incorporating the Filtering procedure in the resource allocation strategies designed for heterogeneous data scenarios.

REFERENCES

1. Study on scenarios and requirements for next generation access technologies. 3GPP Technical Report (TR) 138.913 version 15.0.0 Release 15. (2018)
2. System architecture for the 5G System. 3GPP Technical Specification (TS) 123.501 version 15.9.0. Release 15. (2020)
3. Network Slice Selection Services. 3GPP Technical Specification (TS) 129.531 version 15.5.0. Release 15. (2019)
4. Ericsson. Network slicing: A go-to-market guide to capture the high revenue potential. (2016) https://www.ericsson.com/assets/local/digital-services/network-slicing/network-slicing-value-potential.pdf
5. Tovinger, T., Cornily, J. M., Gardella, M., Shan, C., Ai, C., Andrianov, A., et al.: Management, Orchestration and Charging in the New Era. Journal of ICT Standardization, **6**(1), 159–178 (2018)
6. Alliance NGMN. Description of network slicing concept. NGMN 5G P 1.1 (2016)
7. Marquez, C. et al.: Resource sharing efficiency in network slicing. IEEE Transactions on Network and Service Management. **16**(3), 909–923 (2019)
8. Begishev, V., Petrov, V., Samuylov, A., Moltchanov, D., Andreev, S., Koucheryavy, Y., Samouylov, K.: Resource Allocation and Sharing for Heterogeneous Data Collection over Conventional 3GPP LTE and Emerging NB-IoT Technologies. Comput. Communicat. **120**(2). 93–101 (2018).
9. Broadband network traffic. Performance evaluation and design of broadband multiservice networks. Final report of action COST 242 / James Roberts ... (ed). Lecture notes in computer sciences (LNCS). **1155** Springer, (1996).
10. Stepanov, S.N., Stepanov, M.S.: Efficient Algorithm for Evaluating the Required Volume of Resource in Wireless Communication Systems under Joint Servicing of Heterogeneous Traffic for the Internet of Things. Automation and Remote Control. **80**(8). 1970-1985 (2019)
11. Stepanov, S.N., Stepanov, M.S., Andrabi, U.M., Ndayikunda, J.: The Analysis of Resource Sharing for Heterogenous Traffic Streams over 3GPP LTE with NB-IoT Functionality. Lecture Notes in Computer Science (LNCS). **12563**. Springer, Cham. 422-435 (2020)
12. Stepanov, S.N., Stepanov, M.S.: Methods for Estimating the Required Volume of Resource for Multiservice Access Nodes. Automation and Remote Control. **81**(12). 2244-2261 (2020)

УДК: 519.872

Метод марковского суммирования для исследования потока повторных обращений в двухфазной системе MAP|GI|∞

М.А. Шкленник[1], А.Н. Моисеев[1], Л.А. Задиранова[1]

[1]Национальный исследовательский Томский государственный университет, пр. Ленина, 36, Томск, Россия

shklennikm@yandex.ru, moiseev.tsu@gmail.com, zhidkovala@mail.ru

Аннотация

Рассматривается двухфазная бесконечнолинейная система массового обслуживания с возможностью повторного обслуживания на второй фазе. На вход системы поступает MAP-поток заявок. Время обслуживания заявки на каждой фазе системы является произвольной случайной величиной, заданной соответствующей функцией распределения. Ставится задача исследования потока повторных обращений в систему методом марковского суммирования. Получено выражение для характеристической функции числа повторных заявок в системе при асимптотическом условии высокой интенсивности входящего потока в нестационарном режиме работы.

Ключевые слова: *Метод марковского суммирования, поток повторных обращений, система массового обслуживания, метод асимптотического анализа, неограниченное число приборов, мгновенная обратная связь, нестационарный режим.*

1. Введение

Системы массового обслуживания (СМО) используются как для описания и исследования процессов в производственных, экономических и социальных системах [1, 2], так и широко применяются при исследовании характеристик коммуникационных сетей [3, 4, 5]. Зачастую на практике имеется потребность циркуляции заявки в системе, то есть ее возвращении для повторного обслуживания, вследствие чего возникает необходимость построения математических моделей, учитывающих этот факт. В частности, класс таких систем составляют системы с обратной связью. Различают СМО с мгновенной и отложенной (отсроченной) обратной связью [6, 7].

В данной работе рассматривается двухфазная система массового обслуживания с неограниченным числом обслуживающих устройств и возможностью повторного обслуживания на второй фазе (с мгновенной обратной связью). На вход системы поступает MAP-поток, время обслуживания на каждой фазе является произвольной случайной величиной, заданной соответствующей функцией распределения. Методом марковского суммирования [8, 9] составлена система дифференциальных уравнений Колмогорова для распределения вероятностей числа повторных обращений заявок к системе за фиксированный интервал времени при условии нестационарного режима работы системы. При асимптотическом условии высокой интенсивности входящего потока [10] получено выражение для характеристической функции числа повторных обращений в системе.

2. Постановка задачи

Рассмотрим двухфазную систему массового обслуживания с неограниченным числом обслуживающих устройств и возможностью повторного обращения на второй фазе.

На вход системы поступает MAP-поток заявок, который определяется следующим образом: пусть имеется цепь Маркова $k(t)$ с конечным числом состояний, $k(t) = 1, 2, \ldots, K$, определяемая матрицей $\mathbf{Q}$ инфинитезимальных характеристик $q_{k\nu}$, $k, \nu \in \{1, \ldots, K\}$; пусть заданы неотрицательные величины $\lambda_1, \lambda_2, \ldots, \lambda_K$, определяющие условные интенсивности наступления событий в MAP-потоке для каждого состояния цепи $k(t)$, а также совокупность условных вероятностей $d_{k\nu}$ того, что в потоке наступает событие в момент, когда цепь $k(t)$ меняет свое состояние с k на ν (предполагается, что $k \neq v$ и $d_{kk} = 0$). Цепь $k(t)$ называется управляющей цепью Маркова для рассматриваемого MAP-потока.

Если $k(t) = k$, то за время dt может произойти следующее:

1) с вероятностью $\lambda_k dt$ в потоке наступает событие, при этом цепь Маркова не меняет свое состояние;

2) с вероятностью $q_{k\nu}dt$ процесс $k(t)$ переходит из состояния k в состояние ν ($\nu \neq k$) и с вероятностью $d_{k\nu}$ в MAP-потоке наступает событие;

3) либо с вероятностью $(1 - \lambda_k d + q_{kk}dt)$ ничего не произойдет.

Время обслуживания заявки на первой фазе системы является произвольной случайной величиной, заданной функцией распределения $B_1(x)$. Каждая заявка, завершив обслуживание на первой фазе системы, с вероятностью r_1 может перейти на вторую фазу системы (это обращение будем тоже считать повторным) или с вероятностью $(1-r_1)$ может покинуть систему. Время обслуживания заявки на второй фазе системы также является произвольной случайной величиной, имеющей функцию распределения $B_2(x)$. Завершив обслуживание на второй фазе системы, заявка может с вероятностью r_2 вернуться на вторую фазу системы

для следующего повторного обслуживания или с вероятностью $(1 - r_2)$ может покинуть систему. Исследуем поток повторных обращений заявок к системе (всех обращений на вторую фазу), который будем называть r-потоком.

3. Метод марковского суммирования

Будем полагать, что в момент времени $t_0 = 0$ система свободна и в ней нет обслуживаемых заявок.

Каждая заявка входящего потока, поступив в систему в произвольный момент времени t, будет формировать события r-потока, которые наступят после момента времени t.

Зафиксируем некоторый момент времени $T > t$ в будущем. Введем следующие обозначения:

- $\xi(t)$ – число событий в r-потоке, сформированных за интервал времени $[t, T]$ одной заявкой, пришедшей в момент времени t;
- $g(i, t) = P\{\xi(t) = i\}$ – вероятность того, что заявка, поступившая в систему в момент времени t, к моменту времени T сформирует в r-потоке i событий;
- $m(t)$ – число событий r-потока, сформированных всеми заявками входящего потока, поступившими в систему на интервале $[0, t]$.

Случайный процесс $\xi(t)$ будем называть локальным r-потоком.

Пусть

$$P_k(m, t) = P\{m(t) = m, k(t) = k\},$$

тогда для распределения вероятностей $P_k(m, t)$ можно записать систему дифференциальных уравнений Колмогорова

$$\frac{\partial P_k(m,t)}{\partial t} = \lambda_k P_k(m,t) + \lambda_k \sum_{i=0}^{m} P_k(m-i,t)g(i,t) +$$

$$+ \sum_{\nu \neq k} \left(\sum_{i=0}^{m} P_\nu(m-i,t)g(i,t) \right) d_{\nu k} q_{\nu k} + \sum_{\nu} P_\nu(m,t)(1 - d_{\nu k})q_{\nu k}$$

с начальным условием

$$P_k(m, 0) = \begin{cases} 1, m = 0, \\ 0, m \neq 0 \end{cases}$$

для всех $k = 1, \ldots, K$.

Переходя к характеристическим функциям вида

$$H_k(u, t) = \sum_{m=0}^{\infty} e^{jum} P_k(m, t),$$

$$G(u,t) = \sum_{i=0}^{\infty} e^{jui} g(i,t),$$

где $j = \sqrt{-1}$, получим систему дифференциальных уравнений

$$\frac{\partial H_k(u,t)}{\partial t} = \lambda_k H_k(u,t)(G(u,t)-1) + \sum_{\nu} H_\nu(u,t) q_{\nu k} \left(d_{\nu k}(G(u,t)-1)+1\right). \quad (1)$$

Характеристическая функция $G(u,t)$ процесса $\xi(t)$ была получена в работе [9] и имеет вид

$$G(u,t) = 1 + r_1(e^{ju}-1)B_1(T-t) + \frac{r_1 r_2}{2\pi}(e^{ju}-1)e^{ju}\varphi(u,t),$$

где

$$\varphi(u,t) = \frac{1}{2\pi}\int_{-\infty}^{+\infty} \frac{b_1^*(\alpha) b_2^*(\alpha)\left(1-e^{-j\alpha(T-t)}\right)}{\left(1-r_2 e^{ju} b_2^*(\alpha)\right) i\alpha} d\alpha,$$

$$b_1^*(\alpha) = \int_0^{\infty} e^{j\alpha\tau} dB_1(\tau), \qquad b_2^*(\alpha) = \int_0^{\infty} e^{j\alpha\tau} dB_2(\tau).$$

Введем следующие матричные обозначения:
$\mathbf{h}(u,t) = \{H_1(u,t), H_2(u,t), ..., H_K(u,t)\}$;
$\mathbf{Q}$ – матрица инфинитезимальных характеристик $q_{\nu k}$;
$\mathbf{\Lambda}$ – диагональная матрица с элементами λ_k по главной диагонали;
$\mathbf{D}$ – матрица из элементов $d_{\nu k}$.
Тогда систему уравнений (1) можно записать в матричном виде

$$\frac{\partial \mathbf{h}(u,t)}{\partial t} = \mathbf{h}(u,t)\left[\mathbf{Q} + r_1(e^{ju}-1)\left(B_1(T-t) + r_2 e^{ju}\varphi(u,t)\right)(\mathbf{\Lambda} + \mathbf{DQ})\right] \quad (2)$$

с начальным условием

$$\mathbf{h}(u,0) = \mathbf{r}, \quad (3)$$

где $\mathbf{r}$ – вектор-строка стационарного распределения вероятностей значений марковского процесса $k(t)$, удовлетворяющая системе уравнений

$$\begin{cases} \mathbf{rQ} = \mathbf{0}, \\ \mathbf{re} = 1. \end{cases} \quad (4)$$

Здесь $\mathbf{e}$ – единичный вектор-столбец, $\mathbf{0}$ – вектор-строка из нулей.

Будем рассматривать уравнение (2) в асимптотическом условии высокой интенсивности входящего потока. Интенсивность входящего потока представим в виде $N\lambda$, где λ – фиксированная величина, определяемая выражением

$$\lambda = \mathbf{r}(\mathbf{\Lambda} + \mathbf{DQ})\mathbf{e},$$

а параметр N имеет большие значения (асимптотическое условие высокой интенсивности имеет вид $N \to \infty$). Для этого высокоинтенсивный MAP-поток будем задавать матрицами вида $N\mathbf{Q}$, $N\mathbf{\Lambda}$ и $\mathbf{D}$ [10]. В этом случае уравнение (2) можно переписать в виде:

$$\frac{1}{N}\frac{\partial \mathbf{h}(u,t)}{\partial t} = \mathbf{h}(u,t)\left[\mathbf{Q} + r_1(e^{ju}-1)\left(B_1(T-t) + r_2 e^{ju}\varphi(u,t)\right)(\mathbf{\Lambda}+\mathbf{DQ})\right]. \quad (5)$$

Результат асимптотического анализа представим в виде теоремы.

Теорема 1. *Пусть* $\mathbf{r} = \{R(1), R(2), \ldots, R(K)\}$*–вектор-строка стационарного распределения вероятностей состояний цепи Маркова* $k(t)$*, определяемый решением системы уравнений* (4).

Тогда вектор-строка частичных характеристических функций $\mathbf{h}(u,t) = \{H_1(u,t), H_2(u,t), ..., H_K(u,t)\}$ *распределения числа событий в потоке повторных обращений за интервал времени* $[0;T]$ *при условии высокой интенсивности входящего потока, имеет вид*

$$\mathbf{h}(u,T) = \mathbf{r}\exp\left\{juN\lambda r_1\left[\int_0^T B_1(y)dy + \right.\right.$$

$$\frac{r_2}{2\pi}\int_{-\infty}^{+\infty}\frac{b_1^*(\alpha)b_2^*(\alpha)}{j\alpha(1-r_2b_2^*(\alpha))}\left(T-\frac{1-e^{-j\alpha T}}{j\alpha}\right)d\alpha\Big] +$$

$$\frac{(ju)^2}{2}r_1N\left[\lambda\int_0^T B_1(y)dy +\right.$$

$$\lambda\frac{r_2}{2\pi}\int_{-\infty}^{+\infty}\frac{b_1^*(\alpha)b_2^*(\alpha)\left(3-r_2b_2^*(\alpha)\right)}{j\alpha(1-r_2b_2^*(\alpha))^2}\left(T-\frac{1-e^{-j\alpha T}}{j\alpha}\right)d\alpha +$$

$$\left.\left.\kappa\int_0^T\left(B_1(y)+\frac{r_2}{2\pi}\int_{-\infty}^{+\infty}\frac{b_1^*(\alpha)b_2^*(\alpha)}{j\alpha(1-r_2b_2^*(\alpha))}\left(\frac{1-e^{-j\alpha T}}{j\alpha}\right)d\alpha\right)^2 dy\right]\right\},$$

где $\kappa = 2\mathbf{g}(\mathbf{\Lambda}+\mathbf{DQ}-\lambda\mathbf{I})\mathbf{e}$*, а вектор-строка* $\mathbf{g}$ *определяется как решение уравнения*

$$\mathbf{gQ} = r_1[\lambda\mathbf{r} - \mathbf{r}(\mathbf{\Lambda}+\mathbf{DQ})].$$

Следствие 1. *Асимптотическая характеристическая функция распределения вероятностей числа событий в* r*-потоке совпадает с характеристической функцией нормального распределения с математическим ожиданием*

$$\mathrm{M}\{m(T)\} = N\lambda r_1\int_0^T\psi(x)dx,$$

и дисперсией

$$\mathrm{D}\left\{m(T)\right\} = r_1\lambda\int_0^T \psi(x)dx + 2r_1r_2\lambda\int_0^T \left(a_0(x)+a_1(x)\right)dx + r_1\kappa\int_0^T \psi^2(x)dx,$$

где

$$a_0(x) = \frac{1}{2\pi}\int_{-\infty}^{+\infty} \frac{b_1^*(\alpha)b_2^*(\alpha)}{1-r_2b_2^*(\alpha)}\left(\frac{1-e^{-j\alpha(T-x)}}{j\alpha}\right)d\alpha,$$

$$a_1(x) = r_2\cdot\frac{1}{2\pi}\int_{-\infty}^{+\infty} b_1^*(\alpha)\left(\frac{b_2^*(\alpha)}{1-r_2b_2^*(\alpha)}\right)^2\left(\frac{1-e^{-j\alpha(T-x)}}{j\alpha}\right)d\alpha,$$

$$\psi(x) = B_1(T-x) + \frac{r_2}{2\pi}\int_{-\infty}^{+\infty} \frac{b_1^*(\alpha)b_2^*(\alpha)}{1-r_2b_2^*(\alpha)}\left(\frac{1-e^{-j\alpha(T-x)}}{j\alpha}\right)d\alpha.$$

4. Заключение

В работе рассмотрена двухфазная система массового обслуживания $MAP|GI|\infty$ с мгновенной обратной связью на второй фазе. Для нахождения распределения вероятностей числа событий в потоке повторных обращений за фиксированный интервал времени используется метод марковского суммирования. Найдено выражение для характеристической функции искомого распределения вероятностей при нестационарном режиме работы в асимптотическом условии высокой интенсивности входящего потока заявок.

Литература

1. Жидкова Л. А., Моисеева С. П. Математическая модель потоков покупателей двухпродуктовой торговой компании в виде системы массового обслуживания с повторными обращениями к блокам// Известия Томского политехнического университета. 2013. Т. 322, № 6. С. 5–9.
2. Shklennik M., Moiseeva S., Moiseev A. Optimization of Two-Level Discount Values Using Queueing Tandem Model with Feedback// Communications in Computer and Information Science. 2018. Vol. 912. P. 321–332.
3. Шкленник М. А., Моисеев А. Н. Математическая модель системы обработки результатов физических экспериментов с необходимостью повторной обработки данных//Известия вузов. Физика. 2019. Т. 62., № 3. С. 148–153.
4. Van Doorn E. A., Jagers A. A. A Note on the GI/GI/infinity system with identical service and interarrival-time distributions// Queueing Systems. 2004. Vol. 47. P. 45–52.

5. Whitt, W. Fluid models for multiserver queues with abandonments// Operations Research. 2006. Vol. 54. P. 37–54.
6. Королюк В. С., Меликов А. З., Пономаренко Л. А., Рустамов А. М. Методы анализа многоканальной системы обслуживания с мгновенной и отсроченной обратными связями// Кибернетика и системный анализ. – 2016. Т.52. №1. С. 64-77.
7. Melikov A. Z., Aliyeva S. H., Sztrik J. Analysis of queuing system MMPP/M/K/K with delayed feedback// Mathematics. 2019. V. 7. 1128. 14 p.
8. Назаров А. А., Даммер Д. Д. Исследование дополнительно формируемого потока в системе с неограниченным числом приборов и рекуррентным обслуживанием методом марковского суммирования// Автоматика и телемеханика. 2019. №12 С. 133–145.
9. Шкленник М. А., Моисеев А. Н. Метод марковского суммирования для исследования потока повторных обращений в двухфазных системах $M|GI|\infty \to GI|\infty$// Известия Саратовского университета. Новая серия. Серия: Математика. Механика. Информатика. 2021. Т. 21, вып. 1. С. 111–123.
10. Моисеев, А. Н., Назаров А. А. Бесконечнолинейные системы и сети массового обслуживания// Томск: Изд-во НТЛ, 2015. 240 с.

UDC: 004.451.87

Linux network device drivers: NAPI polling in kernel threads

A. Borisovskaya

RUT (MIIT), Obraztsova 9, Moscow, Russia

borisovsk0101@gmail.com

Abstract

This paper discusses the basic concepts of Linux network device drivers. A new function added in a series of patches for kernel 5.12 is considered, in which the NAPI mechanism is implemented.

Keywords: Networking, device drivers, softirq, polling, kernel thread

1. Introduction

The Linux networking subsystem (figure 1) is modeled on the BSD stack. Data reception and transmission at the transport and network layers occurs with the socket interface [1]. Unlike Unix sockets for interprocess communication, TCP/IP sockets use the network protocol for operation. When created (`sys_socket`), it takes parameters for domain, type, local and remote IP and port. The socket buffer (`sk_buff`) is actually a packet. A linked list of instances of such structures is the network interface queue (`tx_queue`, `rx_queue`).

The function of network drivers is to implement the link layer (resolution MAC addresses) and providing an interface between kernel system calls and a network card. Processing of incoming and outgoing packets is performed using the `xmit` and `rx` functions. They are protected from concurrent access by spin locks. The operations of updating statistics and changing transmission parameters are also protected by spin locks. The interface itself is defined by the `net_device` structure, and the `alloc_netdev` and `register_netdev` functions are called to create and register a network device.

2. Packet receiving. Interrupt handling

A network device driver is similar to a block device driver: it transmits and receives data on demand. But block drivers only respond to kernel requests, and network drivers receive packets from the outside asynchronously. For a long time

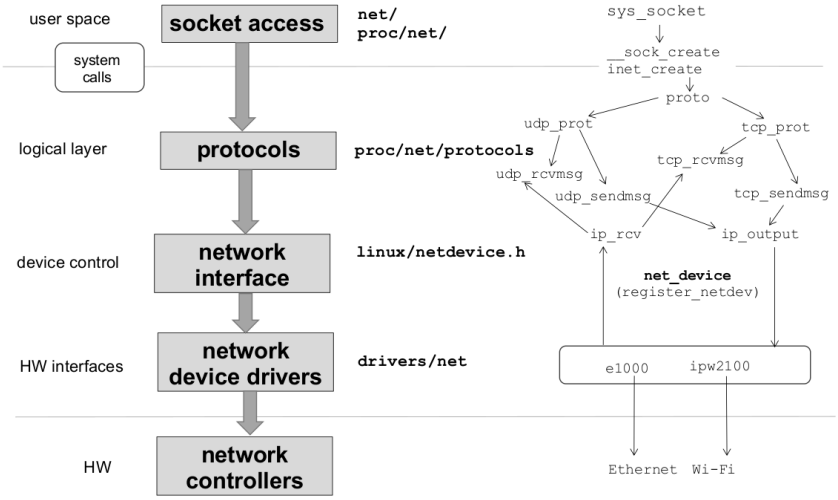


Fig. 1. Linux networking subsystem

Linux had a hardware interrupt handling mechanism when a network device "asked" to put incoming packets into the kernel.

Prior to the 2.3 kernels, after calling the top half interrupt handler, the bottom half and the task queue were used to perform main tasks. In version 2.3, the BH interface has been replaced with a software interrupt handler (softirq), tasklets, and work queues [2]. The advantage of softirqs is that they can run concurrently on different processors. They are explicitly used in the networking subsystem.

3. New API. Polling

As long as the network traffic was moderate, the interrupt mechanism for receiving packets successfully coped with its work. While the traffic was increasing, the constant interrupt handling led to a shortage of processor time for user programs and packet loss.

A solution to the problem was suggested in 2001 and appeared as a New API in the 2.4 kernels. In the original paper, the method was tested on an SMP (Symmetric Multiprocessing) system using a traffic generator like pktgen [3],[4]. The main purpose of NAPI is to reduce the number of interrupts generated when receiving packets. NAPI combines the interrupt mechanism with the polling mechanism. Most often in development, the use of polling is avoided, as resources can be wasted when the device is idle. High-load interfaces do not have this problem.

NAPI compliant drivers disable interrupts when a packet arrives on an interface. The interrupt handler then calls `rx_schedule` to ensure that packet processing is

done later. When incoming packets fill the buffer (the limit is budget), the `dev->poll` method is called for processing. The `poll` method will be called concurrently on no more than one processor, making synchronization easier. If the load decreases, the interrupts are enabled again. This allows performance to be dynamically adjusted based on the load on the interface. The polling method can also be used for packet transmission.

When implementing a NAPI compliant driver, several requirements must be met:

- Ability to store incoming packets in a DMA ring or a buffer in the NIC itself
- Ability to disable interrupts
- In the poll method, the ability to pick up several packets at a time must be implemented
- Since the poll method works in the context of softirq and is controlled by the ksoftirqd daemon, on systems with high load, you need to change the polling priority to balance the resources between the interrupt handler and user programs.

Disadvantages of NAPI:

- In some cases, there may be delays in the system if the entire interrupt handler is placed in `dev->poll`
- IRQ masking can be slow
- An IRQ race condition is possible if a packet arrives while checking the bit for new packets and enabling interrupts.

4. Kthread based NAPI polling. Softirq

In a series of patches in kernel 5.12, the poll method from the softirq context has been moved to the kernel thread [5],[6],[7].

Wei Wang in a commentary to the patch says that the reason for this decision is the inability to track software interrupts in the system. The scheduler cannot measure the time it takes to process the softirq. The kernel thread, on the other hand, is visible to the CPU task scheduler. This will avoid overloading the processor on which it works, and make the scheduling of userspace processes more deterministic. It is easier for the system administrator to control it. Kthread can be associated with a specific CPU group to explicitly decouple user threads from CPUs polling network interfaces.

The changes mainly affected `net/core/dev.c`.
Updated `__napi_poll` method called from `napi_poll` context. There is a new `sysfs` attribute in `net_device` to enable/disable threaded polling for all NAPI instances of a given network device without having to call `up/down`. A “thread” field has been added to `napi_struct` to implement polling within a thread. To enable thread support after

creating a `kthread`, you need to call `napi_set_threaded` (`NAPI_STATE_THREADED` flag). Due to the addition of threading, there is a new method `napi_thread_wait`.

5. Conclusion

Summing up, we can say that the polling method effectively copes with the problem of reducing the number of interrupts from network devices. NAPI provides this function in context of software interrupt handler. But, since softirq has a number of disadvantages, in particular, it is difficult to manage. In the new version of the kernel (5.12), the poll method has been moved to the context of the kernel thread. One kernel thread runs once on each CPU, that makes the synchronization easier.

REFERENCES

1. Linux Device Drivers, 3 rd ed. J. Corbet, A. Rubini, G. Kroah-Hartman, O'Reilly. 2005. P. 478-520.
2. Linux Kernel Development. R. Love, Addison Wesley, 2019.
3. J. H. Salim, R. Olsson, A. Kuznetsov Beyond Softnet // Proceedings of the 5 th Annual Linux Showcase Conference Oakland, California, USA. 2001.
4. J. H. Salim When NAPI comes to town // Linux Conference and Tutorials, University of Wales, Swansea. 2005.
5. J. Corbet NAPI polling in kernel threads, 2020, `https://lwn.net/Articles/833840/`
6. J. Corbet Threadable NAPI polling, softirqs, and proper fixes, 2016, `https://lwn.net/Articles/687617/`
7. W. Wang, J. Kicinski, E. Dumazet, P. Abeni, H. F. Sowa, F. Fietkau, [PATCH net-next v9 0/3] implement kthread based napi poll, 2021, `https://lore.kernel.org/netdev/20210129181812.256216-1-weiwan@google.com/T/`

УДК: 519.872.5:656.073

К вопросу о применении теории массового обслуживания при моделировании работы железнодорожных станций

М.Л. Жарков, А.Л. Казаков, А.Л. Лемперт

Институт динамики систем и теории управления имени В.М. Матросова СО РАН, Лермонтова 134, Иркутск, Российская Федерация

zharkm@mail.ru, kazakov@icc.ru, lempert@icc.ru

Аннотация

В статье представлено развитие разработанного авторами подхода к математическому моделированию работы транспортных систем, который основан на теории массового обслуживания. За счет учета различных дисциплин принятия групп заявок стало возможно отобразить в модели особенности принятия поездов на железнодорожных станциях и тем самым повысить точность их моделирования. Построена и идентифицирована математическая модель работы одной из крупнейших в мире железнодорожных станций в виде сети массового обслуживания с BMAP-потоком (Batch Markovian Arrival Process). В ней учтены сложный входящий вагонопоток и особенность его принятия, а также сортировочные и грузовые функции этой станции.

Ключевые слова: *теория массового обслуживания, сети массового обслуживания, BMAP, железнодорожная станция, транспортный поток.*

1. Введение

В настоящее время теория массового обслуживания (ТМО) имеет широкое практическое применение в различных сферах деятельности [1]. Наиболее часто данный математический аппарат используется при исследовании информационно-телекоммуникационных систем [2]. Однако он применяется и в сфере транспорта, так как оптимизационные модели, которые являются наиболее распространенными в этой отрасли [3], не всегда позволяют получить содержательные результаты для прикладных задач при наличии существенных случайных воздействий.

В области железнодорожного транспорта системы массового обслуживания (СМО) начали применять еще в 70-е годы [4, 5], к настоящему времени они

Исследование выполнено при финансовой поддержке РФФИ в рамках проекта №20-010-00724, РФФИ и Правительства Иркутской области в рамках проекта №20-47-383002

нередко используются при изучении железнодорожных станций и участков сети [6]. Как известно, СМО хорошо подходят для исследования объектов, в которых регулярно повторяются однотипные действия [1, 2]. К ним как раз относятся грузовые и сортировочные железнодорожные станции (ГСЖС), производительность которых определяет грузовую пропускная способность сети в целом [4, 7].

На основе этого математического аппарата авторы разработали подход к моделированию работы ГСЖС [8, 9]. В нем для описания входящего вагонопотока применяется модель $BMAP$ (Batch Markovian Arrival Process) [10]. Входящий поток принимается в систему согласно дисциплине полного отказа – если не хватаем места хотя бы для одной заявки из группы, то вся она получает отказ. Для описания процесса обслуживания заявок (вагонов) в системе используются сети массового обслуживания (СеМО) [1, 2]. Все это позволило описать сложный входящий поток заявок, учесть наличие случайных воздействий на его поступление и обслуживание, а также детально отобразить маршрут движения заявки в системе с нелинейной иерархической структурой.

В работе предлагается развитие данного подхода за счет учета различных дисциплин принятия групп заявок и его применение для моделирования одной из крупнейших станций в мире, которая имеет сложную нелинейную структуру и выполняет как сортировочной, так и грузовые функции. Далее в статье представлено обобщенное описание ГСЖС, подход к моделированию их работы и математическая модель выбранной станции в виде СеМО с $BMAP$-потоком.

2. Объект исследования

Рассмотрим грузовые и сортировочные железнодорожные станции. Первые предназначены для массовой погрузки и выгрузки сырья и товаров, вторые – для массового расформирования грузовых поездов на отдельные группы вагонов, накопление и формирование из них новых поездов. Данные объекты обладают общими свойствами, которые позволяют использовать единый подход для их моделирования. Так, ГСЖС выполняют стандартные операции по расформированию и формированию поездов, в частности, типовые грузовые станции дублируют функции сортировочных, но в гораздо меньшем объеме. На крупных сортировочных станциях могут проводиться погрузочные и разгрузочные работы. ГСЖС имеют нелинейную иерархическую структуру, которая может включать и кольцевые маршруты [3, 5]. Выделим подсистемы, которые будем учитывать при моделировании: парк прибытия; сортировочная горка; сортировочный парк; грузовой двор; парк отправления. На типовые ГСЖС поезда поступают минимум с двух направлений. Отметим особенность, которая ранее не учитывалась: допускается временная остановка поезда на маневровых путях, расположенных перед парком прибытия. Это позволяет принять поезд на станцию при отсутствии

свободных путей в самом парке. Таким образом, при моделировании ГСЖС необходимо учитывать наличие сложного группового входящего вагонопотока, особенности его принятия в систему и ее сетевой структуры, элементы которой имеют различные характеристики работы. Пассажирские поезда исключаются из рассмотрения, так как они не обслуживаются ГСЖС.

3. Подход к моделированию

Математическое описание работы ГСЖС строиться в два этапа. На первом описывается входящий на станцию вагонопоток, включающий не менее двух подпотоков, каждый из которых характеризуется направлением, интенсивностью поступления, распределением числа вагонов в составе. Для его математического описания применяются $BMAP$-потоки, что позволяет учесть наличие нескольких групповых подпотоков с различными параметрами в рамках одной модели [10]. Под группой заявок на обслуживание понимается прибывающий в систему поезд.

На втором этапе происходит определение способа принятия входящего вагонопотока в ГСЖС и математическое описание процесса его обслуживания. Группы заявок принимаются в систему согласно дисциплинам полного отказа или полного принятия. Последняя означает, что если в очереди есть место хотя бы для одной заявки из группы, то к обслуживанию допускается вся группа. Эта дисциплина позволяет учитывать особенность принятия поездов при достаточно длинных маневровых путях. Для описания сетевой структуры системы, состоящей из разнотипных подсистем с различными параметрами работы, мы применяем открытые сети массового обслуживания. СеМО представляют собой структуру, состоящую из конечного числа S СМО (далее – узлов). В ней заявки следуют из узла в узел, в соответствии с маршрутной матрицей P [2]. В случае открытой системы, заявки прибывают из внешнего источника, который, как правило, принимают за дополнительный узел с индексом 0. Тогда маршрут заявки определяется стохастической матрицей $P = ||P_{ij}||$ размера $(S+1) \times (S+1)$, элементы которой P_{ij} – вероятности перехода заявки из узла i в узел j, $P_{00} = 0$, $\sum_{j=0}^{S} P_{ij} = 1$.

Модель обслуживания вагонопотока в виде СеМО строится следующим образом. Узел СеМО соответствует отдельной подсистеме выбранного объекта. Каналы в узле описывают работу обслуживающих устройств, соответственно их параметры работы определяют параметры каналов: распределения времени обслуживания и размеров обслуживаемых групп заявок. Очередь в узле характеризует суммарную вместимость всех путей в подсистеме. Заявки на обслуживание выбираются согласно $FIFO$ (первым пришел – первым ушел) [3, 4]. Группы заявок поступают из внешнего источника в определенные узлы и далее движутся по системе согласно маршруту, который описывается в матрице P.

Каналы текущего узла могут временно блокироваться, если в следующем узле нет достаточного количества мест для принятия группы заявок [1, 2].

Таким образом, для структурной идентификации модели необходимо определить количество узлов, задать маршрутную матрицу P и способ принятия входящего потока заявок в систему. Для параметрической идентификации требуется установить параметры входного $BMAP$-потока, а затем для каждого из узлов СеМО – количество каналов, распределения времени обслуживания и размеров обслуживаемых групп заявок, число мест в очереди. Искомыми характеристиками в данном случае являются: вероятность отказа, среднее время пребывания заявки в системе (T_{sist}), средняя длина очереди, среднее число заянтых каналов, вероятность блокировки каналов.

4. Моделирование станции Бэйли Ярд

Рассмотрим одну из самых крупных зарубежных станций – двухсистемную сортировочную станцию Бэйли Ярд, которая расположена в Норт-Платт, штат Небраска, США. Ежесуточно в Бэйли Ярд поступает в среднем 10 тыс. вагонов [11, 12]. Из них примерно по 4 тыс. вагонов перерабатывается на двух сортировочных системах – Восточной и Западной. Данная станция выбрана для апробации предложенного подхода, поскольку, во-первых, с точки зрения операций по расформированию поездов она является типовой двухсистемной станцией с последовательным расположением парков. Во-вторых, в ее структуре имеются Грузовой двор для разгрузки угольных поездов, Угольный парк для формирования собственных угольных маршрутов, и Западный транзитный парк для приема и отправления транзитного, т.е. без расформирования, неугольного вагонопотока. Это позволяет Бэйли Ярд выполнять функции грузовой станции и делает ее структуру нелинейной. Схема станции представлена на рис. 1. Ранее нами моделировалась только Западная система [9], что не позволило отобразить в модели грузовые функции станции. В данной статье работа Бэйли Ярд моделируется с учетом сортировочных и грузовых функций.

Модель работы Бэйли Ярд в терминах ТМО запишется следующим образом. В систему поступает $BMAP$-поток, включающий 131 матрицу $D_k, k = \overline{0,130}$ размера 2×2. СеМО имеет 12 узлов: Узел 0 – источник заявок; Узел 1 (Восточный парк прибытия) – $BMAP/G^B/2/1170$; Узел 2 (Восточная сортировочная горка) – $*/G^B/1/130$; Узел 3 (Восточный сортировочный парк) – $*/G^B/3/3840$; Узел 4 (Восточный парк отправления) – $*/G^B/2/780$; Узел 5 (Грузовой двор) – $*/G^B/8/130$; Узел 6 (Западный парк прибытия) – $BMAP/G^B/2/837$; Узел 7 (Западная сортировочная горка) – $*/G^B/1/0$; Узел 8 (Западный сортировочный парк) – $*/G^B/3/3000$; Узел 9 (Западный парк отправления) – $*/G^B/2/736$; Узел 10 (Западный транзитный парк) – $BMAP/G^B/2/920$; Узел 11 (Угольный парк)

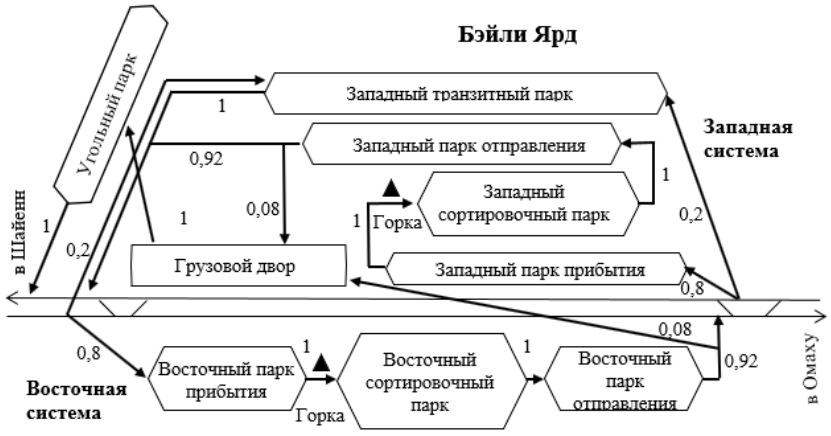


Рис. 1. Схема станции Бэйли Ярд

– $*/G^B/2/736$. Здесь G – обозначает произвольный закон распределения времени обслуживания заявок, B – групповое обслуживание в канале.

Группы заявок принимаются в Восточном и Западном парках прибытия и Западном транзитном парке согласно дисциплине полного принятия. Далее они движутся по системе согласно матрице P. Опустим ее представление из-за большого размера. Вероятности переходов заявок между узлами представлены виде весов на рис. 1. Прямыми стрелками на нем обозначены направления движения заявок. Для идентификации модели использованы данные, полученные из открытых источников и путем изучения космических снимков станции.

5. Численное исследование модели

Исследование полученной СеМО было выполнено численно с помощью разработанной авторами имитационной модели, которая реализована в виде программного комплекса [9]. С его помощью были вычислены показатели эффективности работы системы при различной интенсивности $BMAP$-потока. Подробное описание проведенного вычислительного эксперимента представить здесь не возможно из-за ограниченности объема статьи (они будут представлены в докладе). Отметим лишь, что при увеличении интенсивности до $\lambda = 4,95$ групп заявок в час или в среднем до 13,9 тыс. вагонов в сутки Узлы 3 и 8 оказываются перегруженными: среднее число работающих каналов равно 2,95, что близко к максимальному, вероятность отказа становится ненулевой (0,002). Таким образом, Бэйли Ярд имеет запас пропускной способности до 13,9 тыс. вагонов в сутки. Однако в этом случае сортировочные парки оказываются «узким местом».

6. Заключение

В статье представлено развитие подхода к математическому моделированию работы транспортных систем, основанного на теории массового обслуживания. За счет учета различных дисциплин принятия групп заявок стало возможно отобразить особенности принятия транспортных средств в изучаемые системы. Полученные модели позволяют определить пропускную способность выбранных объектов на этапе проектирования или реконструкции при учете случайных воздействий на процесс их работы, а также оценить уровень загрузки при увеличении входящего транспортного потока в будущем. На основе предлагаемого подхода разработана математическая модель работы одной из крупнейших сортировочных железнодорожных станций в мире в виде СеМО с входящим $BMAP$-потоком и дисциплиной полного принятия.

Литература

1. Гнеденко Б. В., Коваленко И. Н. Введение в теорию массового обслуживания. ЛКИ, Москва, 2007.
2. Вишневский В. М. Теоретические основы проектирования компьютерных сетей. Техносфера, Москва, 2003.
3. Миротин Л. Б. [и др.]. Управление грузовыми потоками в транспортно-логистических системах. Горячая линия-Телеком, Москва, 2010.
4. Поттгофф Г. Учение о транспортных потоках. Транспорт, Москва, 1975.
5. Акулиничев В.М. Математические методы в эксплуатации железных дорог. Транспорт, Москва, 1981.
6. Wilson N., Fourie C. J., Delmistro R. Mathematical and simulation techniques for modelling urban train networks // South Afr. J. Ind. Eng. 2016. V. 27. P. 109–119.
7. Pyrgidis C. Railway Transportation Systems. Design, Construction and Operation. CRC Press, Boca Raton, 2016.
8. Казаков А. Л., Павидис М. М., Жарков М. Л. Применение многофазных систем массового обслуживания для моделирования сортировочной станции // Вестник УрГУПС. 2018. № 2 (38). С. 4–14.
9. Bychkov I., Kazakov A., Lempert A., Zharkov M. Modeling of Railway Stations Based on Queuing Networks // Applied Sciences. 2021. V. 11(5). P. 2425.
10. Дудин А. Н., Клименок В. И. Системы массового обслуживания с коррелированными потоками. БГУ, Минск, 2000.
11. Goossens J. W., Hoesel S. P. On solving multi-type railway line planning problems of USA // Eur. J. Oper. Res. 2006. V. 168. P. 403-–424.
12. Golden Spike Tower. Bailey Yard. `https://goldenspiketower.com/bailey-yard/`

UDC: 519.217

Resource Queueing System $M/M/\infty$ in Random Environment

Nikita Krishtalev[1], Ekaterina Lisovskaya[1], Alexander Moiseev[1]

[1]Tomsk State University, 36 Lenin Ave., Tomsk, Russian Federation, 634050

krishtalevnik@gmail.com, ekaterina_lisovs@mail.ru, moiseev.tsu@gmail.com

Abstract

In this paper the resource queueing system operating in a random environment is considered. When the environment changes its state, the arrival rate, the service rate, and resource requirements are changed. An approximation for the characteristic function of the probability distribution of the total amount of occupied resource is derived under the asymptotic condition of growing arrival rate and extremely frequent changes in the states of the random environment.

Keywords: Poisson arrivals, random environment, resource queue.

1. Introduction

Both random environment models and resource queues can be widely applied for modeling real-life systems. For example, we can consider the data processing cluster with some servers with shared RAM process requests. Over time, the regime of the system can be changed, and with it, the intensity of arrivals of requests, time of their processing, and the volume of resources they occupy are changing. For designing such a system, it is important to know the optimal the number of servers and the optimal reserve for resources volume to ensure normal operation. Such a system can be modeled as a $M/M/\infty$ queue operating in a random environment. Changes in its operating regime can be taken into account by describing this factor in the form of a continuous-time Markov chain with a finite number of states which is called a random environment, and its state determines the values of the system parameters.

There are many papers devoted to studies of queueing systems operating in a random environment. For example, a $M/M/1$ queueing model with disasters in a random environment was considered in [1]. The model contains a repair state, a checking state. The steady-state behavior of the underlying queueing model along with the average queue size is analyzed. In [2], a $M/M/1$ retrial queue with an unreliable server was studied whose arrival, service, failure, repair, and retrial rates

are all modulated by an exogenous random environment. Similarly, the steady-state behavior is considered, and the condition of the steady-state regime is obtained. Moreover, the optimization problem of the initial parameters is solved subject to cost and revenue constraints. There are some similar papers, where authors are considered queueing systems operating in a random environment [3, 4, 5].

Resource queueing systems [6, 7, 8, 9] allow to model systems, where each arrival takes a certain amount of resource in addition to the occupation of a server. The amount of resources may be deterministic or random, discrete or continuous. While a customer is being serviced, it occupies both the server and the resources, at the end of the service, the customer leaves the system and frees up the occupied server and resources.

Unfortunately, we did not find any publication that combines both features – resources and a random environment. In the paper, we consider the model of infinite-server resource queue which is operating in a random environment. Each arrived request occupies one server and a random amount of resources for its service. Arrival rate, service rate and amount of occupied resource depend on the state of the random environment. When the environment changes its state all parameters are changed. The study is made on the basis of the asymptotic results under the condition of growing arrival rate and extremely frequent changes in the states of the random environment.

2. Mathematical model

Consider a queueing system with an unlimited number of servers and an unlimited capacity of some resource that operates in a random environment, such the functioning of the system depends on the environment state. The random environment is specified by a continuous-time Markov chain with a finite number of states $s \in \{1, \dots, S\}$ and generator $\mathbf{Q} = \{q_{sk}\}$, $s, k = 1, \dots, S$. When the process is in state s the rate of the Poisson arrival process is equal to λ_s and the parameter of exponential distribution of the service time is equal to μ_s. We denote:

- the arrivals rates by matrix $\mathbf{\Lambda} = \text{diag}\{\lambda_s\}$, $s = 1, \dots, S$,
- the service rates by matrix $\mathbf{M} = \text{diag}\{\mu_s\}$, $s = 1, \dots, S$.

In addition, each arrival occupies a resource of a random size $v_s > 0$ with the probability distribution function $G_s(y) = \text{P}\{v_s < y\}$ which depends on the environment state.

When the state of the environment changes, the following parameters are changed: the arrival rate, the service rate both for new requests and for requests already under service (they restart their service with a new rate value), and also the distribution function of the required resource (resources will be re-occupied according to the new distribution function).

When a request completes servicing, it leaves the system and releases the resource that it occupied during the capture. *Capture* is understood as the moment when the request arrives or the environment state is changed, at which the resource is allocated.

A stochastic process $\{s(t), i(t), v(t)\}$ describes the system's state at time t as follows:

- the environment state at time t by $s(t)$, $s(t) \in \{1, \dots, S\}$,
- the number of requests in the system at time t by $i(t)$, $i(t) \in \{0, 1, 2, \dots\}$,
- the total amount of occupied resource at time t by $v(t)$, $v(t) \geq 0$.

Our goal is to find the steady-state probability distribution of the total amount of the occupied resources.

3. Main result

We will consider the solution under the asymptotic condition of the growing intensity of arrivals and extremely frequent changes in the states of the random environment. We introduce the approximation parameter $N \to \infty$, then the intensities matrix of the arrivals will take the form $N \cdot \mathbf{\Lambda}$, and the Markov chain generator for the random environment will be $N \cdot \mathbf{Q}$.

In the paper [10], the approximation of the characteristic function of the steady-state probability distribution of the number of requests in the system operating in a random environment was obtained under the asymptotic conditions of growing arrivals rate and extremely frequent changes in the environment state. It is expressed as follows:

$$h_s(u) = \sum_{i=0}^{\infty} e^{jui} p_s(i) = r_s \exp\left\{ juN\kappa_1 + \frac{(ju)^2}{2} N\kappa_2 \right\},$$

where

$$j = \sqrt{-1}, \quad \kappa_1 = \frac{\lambda}{\upsilon}, \quad \kappa_2 = -\frac{\lambda}{\upsilon^2}\mathbf{kMe} + \frac{\lambda}{\upsilon} + \frac{\mathbf{k\Lambda e}}{\upsilon},$$

$$\lambda = \mathbf{r\Lambda e}, \quad \upsilon = \mathbf{rMe}, \quad \mathbf{e} = [1, \dots, 1]^T,$$

vector $\mathbf{r}$ satisfies the system of equations

$$\mathbf{rQ} = \mathbf{0},$$
$$\mathbf{re} = 1,$$

vector $\mathbf{k}$ satisfies the system of equations

$$\mathbf{kQ} = \mathbf{r}\left(\kappa_1 \mathbf{M} - \mathbf{\Lambda}\right),$$
$$\mathbf{ke} = 1.$$

Knowing the form of the approximation of the partial characteristic function of the number of requests in the system, we can find the characteristic function for the total volume of the occupied resource in the system. We can derive the characteristic function for the total amount of occupied resource as follows:

$$h(u) = \sum_{s=1}^{S} \mathrm{E}\left\{ e^{ju \sum\limits_{k=1}^{i(t)} \xi_k^{(s)}} \right\} P\{s(t)=s\} =$$

$$= \sum_{s=1}^{S}\sum_{i=0}^{\infty} \mathrm{E}\left\{ e^{ju \sum\limits_{k=1}^{i} \xi_k^{(s)}} \right\} P\{i(t)=i\} P\{s(t)=s\} =$$

$$= \sum_{s=1}^{S}\sum_{i=0}^{\infty} \mathrm{E}\left\{ e^{jui\xi^{(s)}} \right\} P\{i(t)=i\} P\{s(t)=s\} =$$

$$= \sum_{s=1}^{S}\sum_{i=0}^{\infty} (\phi_s(u))^i P\{i(t)=i\} P\{s(t)=s\} = \sum_{s=1}^{S} F(\phi_s(u)) P\{s(t)=s\} =$$

$$= \sum_{s=1}^{S} r_s (\phi_s(u))^{N\kappa_1} e^{\frac{N\kappa_2}{2} ln^2 \phi_s(u)}$$

here $\phi_s(u)$ is a characteristic function of the amount of resource occupied by one request when the environment is in the state s.

Then, applying the inverse Fourier transform to function $h(u)$, we obtain the approximation of the steady-state probability distribution function (pdf) $P(v)$ of the total amount of occupied resource in the system as follows:

$$P(v) = \frac{1}{2\pi} \int\limits_{-\infty}^{\infty} e^{-juv} h(u) du =$$

$$= \frac{1}{2\pi} \sum_{s=1}^{S} r_s \int\limits_{-\infty}^{\infty} e^{-juv} (\phi_s(u))^{N\kappa_1} e^{\frac{N\kappa_2}{2} ln^2 \phi_s(u)} du. \quad (1)$$

The asymptotic parameter N appeared in the final expression while we were finding a solution under the asymptotic condition. This means that we can consider the resulting expression only as an approximation of pdf for sufficiently large N.

4. Numerical example

Let us consider a numerical example to determine the applicability area of approximation. To do this, we compare cumulative distribution function of the

total amount of occupied resource $F(v)$ obtained from the approximation (1) and cumulative distribution function $\hat{F}(v)$ built on the base of simulation results. The comparison will be made by using the Kolmogorov distance

$$\Delta = \sup_{v \geq 0} \left| F(v) - \hat{F}(v) \right|.$$

Let the system parameters are given as

$$\mathbf{\Lambda} = N \cdot \mathrm{diag}\{0.1,\, 1,\, 10\}, \quad \mathbf{M} = \mathrm{diag}\{0.95,\, 2,\, 7.3\},$$
$$\mathbf{Q} = N \cdot \begin{bmatrix} -3 & 1 & 2 \\ 1 & -2 & 1 \\ 2 & 2 & -4 \end{bmatrix},$$

and resource requirements have gamma distributions with parameters $\alpha = [0.5,\, 1.5,\, 2]$ and $\beta = [0.5,\, 1.5,\, 2]$.

We have performed series of experiments and obtained the Kolmogorov distances shown in Table 1 for various values of asymptotic parameter N. We can see, that this metric decreases while N grows. Therefore, obtained approximation (1) becomes more accurate for growing values of asymptotic parameter N. If we suppose that error $\Delta \leq 0.05$ is acceptable, then we can conclude that the approximation is applicable for values $N > 4$ (for the considered example).

Table 1. Kolmogorov distances

N	3	4	5	7	10
Δ	0.083	0.052	0.032	0.016	0.008

5. Conclusion

In the paper, we have considered a resource queueing system $M/M/\infty$ operating in a random environment. When the environment changes its state, the arrival rate, the service rate, and resource requirements are changed. An approximation of the characteristic function of the probability distribution of the total amount of occupied resource in the system is obtained under the condition of growing arrival rate and extremely frequent changes of the states of the random environment. The result is obtained on the base of the corresponding asymptotic results for the distribution of the number of requests in the system in the steady-state regime. Numerical experiments prove the obtained result and allow to obtain an error of the approximation or establish its applicability area.

REFERENCES

1. Sophia, S. Praba, B. Steady-state behavior of an $M/M/1$ queue in random environment subject to system failures and repairs // International Journal of Pure and Applied Mathematics. 2015. Vol. 101. P. 267–279.
2. Cordeiro, J., Kharoufeh, J. The Unreliable $M/M/1$ Retrial Queue in a Random Environment // Stochastic Models. 2012. Vol. 28. P. 29–48.
3. D'Auria, B. Stochastic decomposition of the $M/G/\infty$ queue in a random environment // Operations Research Letters. 2007. Vol. 35. P. 805–812.
4. D'Auria, B. $M/M/\infty$ queues in semi-Markovian random environment // Queueing Systems. 2008. Vol. 58. P. 221–237.
5. Liu, Z., Yu, S. The $M/M/C$ queueing system in a random environment // Journal of Mathematical Analysis and Applications. 2016. Vol. 436, Iss. 1. P. 556–567.
6. Tikhonenko, O. Queuing system with processor sharing and limited resources // Automation and Remote Control. 2010. Vol. 71. P. 803–815.
7. Naoumov, V., Samuilov, K., Samuilov, A. On the total amount of resources occupied by serviced customers // Automation and Remote Control. 2016. Vol. 77. P. 1419–1427.
8. Sopin, E., Vikhrova, O., Samouylov, K. LTE network model with signals and random resource requirements // 2017 9th International Congress on Ultra Modern Telecommunications and Control Systems and Workshops (ICUMT), Munich, 2017. P. 101–106.
9. Lisovskaya, E., Moiseev, A., Moiseeva, S., Pagano, M. Modeling of Mathematical Processing of Physics Experimental Data in the Form of a Non-Markovian Multi-Resource Queuing System // Russian Physics Journal. 2019. Vol. 61. P. 2188–2196.
10. Nazarov, A. A., Baymeeva, G. V. The study of $M|M|\infty$ in random environment // Materials of the second All-Russian youth conference "Mathematical and software support of information, technological and economic systems". Tomsk University Publishing House. 2014. P. 75–80.

UDC: 004.7

OpenFlow-based software-defined networking queue model

V.G. Kartashevskiy, M.A. Buranova

kartashevskiy-vg@psuti.ru, buranova-ma@psuti.ru

Abstract

SDN imposes requirements on manufacturers of infocommunication equipment. It concerns new scenarios support relating to network applications. That would require analysis of SDN networks features and assessment of their performance both at the design and operation stage.

In this work, analytical expressions are obtained to estimate the average values of packet delay times in $M/G/1$ and $G/G/1$ systems for SDN. To approximate arbitrary densities in $G/G/1$ system, an approach based on the use of hyperexponential distributions was used. When analyzing $G/G/1$ system, it is assumed that there are no correlations within the sequences of time intervals between packets and packet processing times. The paper presents the result of comparing the estimates of the average values of packet delay time in $M/G/1$ and $G/G/1$ systems, which simulated the SDN operation.

Keywords: queue model, SDN, quality of service, hyperexponential distribution, average package service time in the system

1. Introduction

Most of the works devoted to SDN performance are aimed at developing simulation models and setting up experiments on real equipment, or developing analytical SDN models based on OpenFlow [1-3], described by $M/M/1$, $M/G/1$ queues. However, it is known [4] that the flows generated by modern applications in the network are not Poisson. That requires taking into account the real properties of traffic in the applied models. In [5.6] mathematical models are presented for systems that process non-Poissonian flows. It should be noted that only functioning parameters of individual SDN sections are considered, not the network as a whole.

The SDN analytic model based on the OpenFlow protocol is discussed below, since it is the most common. As processed streams, we consider packet traffic generated in the form of bursts, which most closely corresponds to the nature of modern streams formation [7].

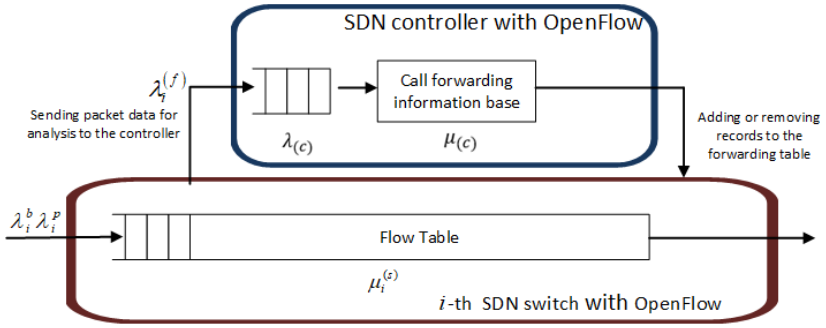


Fig. 1. Processing of incoming messages from the SDN controller

The process of packet arrival and the procedure for forwarding packets to the switch and the OpenFlow controller are analyzed separately. Then we studied the system for forwarding packets to SDN as a whole. The $M/M/1$ system is considered first, followed by $G/G/1$ system.

2. SDN network queuing model with OpenFlow

The procedure for forwarding packets to the switch is illustrated in fig. 1. Since requests to the forwarding tables for all packets are independent of each other, the packet processing time can be represented as a random variable with an exponential distribution, then for all incoming queues in the switch, the packet forwarding queue model in the OpenFlow switch can be represented by the $M/M/1$ queue model [3].

To describe the queue in the OpenFlow switch, we introduce the following designations: $\lambda_{\mathrm{i}}^{(\mathrm{b})}$ is the intensity of the Poisson packet flow of packets arriving at the i-th OpenFlow switch; $\lambda_{\mathrm{i}}^{(\mathrm{p})}$ is intensity characterizing the Poisson distribution of packets number in a packet; $\mu_{\mathrm{i}}^{(\mathrm{s})}$ is the intensity of processing the i-th packet by the switch (corresponds to the exponential distribution). For the SDN controller responsible for k OpenFlow switches, all packets of incoming messages from k switches correspond to the Poisson distribution with the parameter $\lambda_{(c)} = \sum_{i=1}^{k} \lambda_i^{(f)}$.

When an SDN controller processes an incoming request packet from a switch, the processing time of an incoming packet message is mainly determined by looking up the forwarding information base (FIB). In [3], it was assumed that the search time in the FIB can be considered as distributed normally with the mean value $1/\mu_{(\mathrm{c})}$ and variance $\sigma_{(\mathrm{c})}$. The parameter $\mu_{(\mathrm{c})}$ is the average processing rate of incoming packet messages in the controller, and the parameter $\sigma_{(\mathrm{c})}$ is the standard deviation. The queue in the controller is organized as a FIFO (First In - First Out).

In accordance with the assumptions, it is possible to characterize the processing of message packets by the SDN controller using the $M/N/1$, queuing model, where the symbol N corresponds to a normal distribution.

Using the above analysis of the service process for OpenFlow switches and SDN controllers, we can represent the packet processing model in OpenFlow networks as a queuing system according to fig. 1, where the i-th OpenFlow switch with intensity $\mu_{(s)}$ processes bursts of packets arriving at a rate of $\lambda^{(b)} \cdot \lambda^{(p)}$.

If W_i is the time of packet forwarding through the i-th OpenFlow switch, it could be calculated by taking into account two possible processing cases: direct forwarding and forwarding with the participation of the controller. In the latter case, the packet forwarding time consists of two parts [3]: the packet sojourn time in the switch $W_i^{(s)}$ and the sojourn time of the corresponding packet transmission message in the $W^{(c)}$ controller. In this way

$$W_i = \begin{cases} W_i^{(s)} & \textit{with probability } 1 - q_i, \\ W_i^{(s)} + W^{(c)} & \textit{with probability } \ q_i. \end{cases} \tag{1}$$

The average time to process a packet in SDN can be determined by the average time to forward packets through the OpenFlow switch $\overline{W_i}$, according to the diagram shown in fig. 1.

3. Results for $M/G/1$ system

It can be shown [3] that the average time for forwarding packets through the OpenFlow switch $\overline{W_i}$ can be obtained as

$$\overline{W_i} = E\left[W_i\right] = (1 - q_i)\, E\left[W_i^{(s)}\right] + q_i \left(E\left[W_i^{(s)}\right] + E\left[W^{(c)}\right]\right). \tag{2}$$

Let us introduce the notation $\delta_i = 1/\overline{W_i^{(s)}}$ for the i-th switch and $\delta_{(c)} = 1/\overline{W^{(c)}}$ for the controller, where δ_i and $\delta_{(c)}$ are the parameters of density distribution of processing time in the queue in the i-th switch and controller, respectively. The method for determining these parameters is described [8].

The expressions for the probability density of the packet processing time S in the controller $f_{(S)i}(\cdot)$ and the packet processing time C in the switch $f_{(C)}(\cdot)$ are represented in the form

$$f_{(S)i}(u) = \delta_i e^{-\delta_i u}, \tag{3}$$

$$f_{(C)}(u) = \frac{1}{\sqrt{2\pi}\sigma_{(c)}} \mathrm{e}^{-\frac{\left(u - 1/\delta_{(c)}\right)^2}{2\sigma_{(c)}^2}}. \tag{4}$$

For $M/G/1$ system, where the normal distribution is used as an example of an arbitrary distribution ($M/N/1$ system) the expression for the probability density of the packet processing time in the system can be written as:

$$w_{(S,C)i}(u) = (1-q_i)\,\delta_i e^{-\delta_i u} + \\ +\mathrm{q}_i \frac{\delta_i}{2R} e^{-\delta_i u \mathrm{e}\left(\left(\frac{1}{\delta_{(c)}}\right)^2 / 2\sigma^2_{(c)}\right) + \sigma^2_{(c)}\delta_i^2 - 2\delta_{\mathrm{i}}\frac{1}{\delta_{(c)}}} \left[1 - \Phi\left(\frac{\sigma_{(c)}}{\sqrt{2}}\left(\delta_i - \frac{1}{\delta_{(c)}\sigma^2_{(c)}}\right)\right)\right], \quad (5)$$

where $\Phi(x) = \frac{2}{\sqrt{\pi}}\int\limits_0^x e^{-t^2}dt$ [12], $R = \frac{1}{2}\left[1 + \Phi\left(\frac{1}{\sqrt{2}\delta_{(c)}\sigma_{(c)}}\right)\right]$ is normalizing constant.

The final expression for estimating the average packet processing time in the $M/N/1$ system for SDN will be

$$\bar{W}_{(S,C)i}(u) = (1-q_i)\frac{1}{\delta_i} + q_i M \frac{1}{{\delta_i}^2}, \quad (6)$$

where $M = \frac{\delta_i}{2R} e^{\left(\left(\frac{1}{\delta_{(c)}}\right)^2 / 2\sigma^2_{(c)}\right) + \sigma^2_{(c)}\delta_i^2 - 2\delta_{\mathrm{i}}\frac{1}{\delta_{(c)}}} \left[1 - \Phi\left(\frac{\sigma_{(c)}}{\sqrt{2}}\left(\delta_i - \frac{1}{\delta_{(c)}\sigma^2_{(c)}}\right)\right)\right]$.

This approach allows us to evaluate the performance of the SDN network and the main parameters of traffic service quality in the SDN network in the case of processing Poisson streams. Considering that modern applications generate non-Poissonian traffic, queuing systems are better described by $G/G/1$ models.

4. Results for $G/G/1$ system

For $G/G/1$ system, as an example of an arbitrary probability distribution of the request processing time in the controller, similarly to the approach shown for $M/N/1$ system, one can use the normal distribution (for the random variable C) (4), and for the random variable S it is possible to use the probability density in the form of an approximation by a hyperexponential distribution:

$$f_{(S)i}(u) = p\delta_{1i}e^{-\delta_{1i}u} + (1-p)\,\delta_{2i}e^{-\delta_{2i}u}, \quad (7)$$

where p, δ_{1i}, δ_{2i} are distribution parameters.

Then the $G/G/1$ model will be approximated by the $H_2/N/1$ model, and the value of the average packet processing time in the system when using the hyperexponential density approximation of the packet processing time distribution in the switch can be written as:

$$\bar{W}_{(S,C)i}(u) = (1-q_i)\left(\frac{p}{\delta_{1i}} + \frac{(1-p)}{\delta_{2i}}\right) + \left[q_i\left[M_1\frac{p}{{\delta_{1i}}^2} + M_2\frac{(1-p)}{{\delta_{2i}}^2}\right]\right], \quad (8)$$

where $M_1 = \frac{\delta_{1i}}{2R} e^{\left(\left(\frac{1}{\delta_{(c)}}\right)^2 / 2\sigma^2_{(c)}\right) + \sigma^2_{(c)}\delta^2_{1i} - 2\delta_{1i}\frac{1}{\delta_{(c)}}} \left[1 - \Phi\left(\frac{\sigma_{(c)}}{\sqrt{2}}\left(\delta_{1i} - \frac{1}{\delta_{(c)}\sigma^2_{(c)}}\right)\right)\right]$,

$M_2 = \frac{\delta_{2i}}{2R} e^{\left(\left(\frac{1}{\delta_{(c)}}\right)^2 / 2\sigma^2_{(c)}\right) + \sigma^2_{(c)}\delta^2_{2i} - 2\delta_{2i}\frac{1}{\delta_{(c)}}} \left[1 - \Phi\left(\frac{\sigma_{(c)}}{\sqrt{2}}\left(\delta_{2i} - \frac{1}{\delta_{(c)}\sigma^2_{(c)}}\right)\right)\right]$.

If we abandon the assumption that the distribution of the packet processing time in the controller is normal, then for the general case an approximation $H_2/H_2/1$ can be proposed, therefore it is necessary to use expression (7) for S, and for C

$$f_{(C)}(u) = g\delta_{(c)1} e^{-\delta_{(c)1}u} + (1-g)\,\delta_{(c)2} e^{\delta_{(c)2}}, \tag{9}$$

where g, $\delta_{(c)1}$, $\delta_{(c)2}$ are packet processing time density parameters in the controller. After such a replacement, the system $G/G/1$ will be approximated by the system $H_2/H_2/1$.

The analytical expression for the average packet processing time in the $H_2/H_2/1$ system for the SDN network can be written as:

$$\bar{W}_{(S,C)i}(u) = (1-q_i)\left[\frac{p}{\delta_{1i}} + \frac{(1-p)}{\delta_{2i}}\right] + q_i\left[\frac{(A+B)}{\delta_{(c)1}} + \frac{(L+D)}{\delta_{(c)2}}\right], \tag{10}$$

where $A = \frac{pg\delta_{1i}}{\delta_{1i}-\delta_{(c)1}}$ $B = \frac{(1-p)g\delta_{2i}}{\delta_{2i}-\delta_{(c)1}}$ $L = \frac{p(1-g)\delta_{1i}}{\delta_{1i}-\delta_{(c)2}}$ $D = \frac{(1-p)(1-g)\delta_{2i}}{\delta_{2i}-\delta_{(c)2}}$.

As a result, formulas were derived for the distribution density of packet service times in the SDN network and for the average packet processing time in the system if the queue system is represented by the $G/G/1$ model for two cases: the $H_2/N/1$ system and the $H_2/H_2/1$ system.

Based on the obtained dependencies, it is possible to define the parameters that determine the quality of traffic service in the network, for example, the average queue length, delay variation.

5. Numerical results

To determine the accuracy of the received estimations, it is necessary to compare the values obtained for various models, including the results derived for $M/N/1$ and $H_2/N/1$.

Let us compare the numerical estimates of the average processing times in the $M/N/1$ system for these two approaches. We will accept the following conditions for the functioning of the network: $\lambda_i^{(b)} = 1/2$, $\lambda_i^{(p)}$=1, $\mu_i = 1/2$, $\mu_{(c)} = 1$, $\sigma_{(c)} = 1$, the number of switches is 1.

The values of the functioning parameters: the average packet processing time for the SDN has the following values: $M/N/1$ – 11,0 conventional units, $H_2/N/1$ – 31,9 system conventional units, system $H_2/H_2/1$- 34,1 conventional units.

6. Conclusion

In this paper, a model of SDN network functioning is built on the basis of mathematical apparatus of the queuing theory. As a result, analytical expressions are obtained for evaluating the main parameters quality of traffic service in SDN for the $M/G/1$ system and for the $G/G/1$ system, provided that the flows entering the system are mutually independent and there are no correlations within the sequences of time intervals between packets and processing times packages.

Future research should focus on constructing SDN model for the $G/G/1$ system when processing correlated streams. This will make it possible to quickly analyze the efficiency of the SDN network in real operating conditions.

REFERENCES

1. Jarschel M., Oechsner S., Schlosser D., Pries R., Goll S., Phuoc T.G. Modeling and performance evaluation of an openflow architecture // Proceedings of the Twenty-third International Teletraffic Congress (ITC). 2011. P. 1-7.
2. Mahmood K., Chilwan A., Osterbo O., Jarschet M. Modelling of OpenFlow-based software-defined networks: the multiple node case // IET Networks. 2015. V.4(5). P. 278-284.
3. Xiong, B., Yang, K., Zhao, J., Li, W., Li, K.: Performance evaluation of OpenFlow-based software-defined networks based on queueing model // Computer Networks. 2016. V. 102. P. 174–183.
4. Sheluhin O.I., Osin A.V., Smolskij S.M. Samopodobie i fraktaly // Telekommunikacionnye prilozheniya. M.: Fizmatlit, 2008.
5. Samujlov K.E., Shalimov I.A., Buzhin I.G., Mironov Yu.B. Model funkcionirovaniya telekommunikacionnogo oborudovaniya programmno-konfiguriruemyh setej. Sovremennye informacionnye tekhnologii i IT-obrazovanie. 2018. V. 14(1). P. 13-26.
6. Muhizi S., Shamshin G. Muthanna A., Kirichek R., Vladyko A., Koucheryavy A. Analysis and Performance Evaluation of SDN Queue Model // Lecture Notes in Computer Science. 2017. V. 10372. P. 26–37. https://doi.org/10.1007/978-3-319-61382-63.
7. Okamura H., Dohi T., Trivedi K.S. Markovian arrival process parameter estimation with group data, IEEE // ACM Transactions on Networking. 2009. V. 17(4). P.1326-1339.
8. Tarasov V.N., Kartashevskij I.V. Opredelenie srednego vremeni ozhidaniya trebovanij v upravlyaemoj sisteme massovogo obsluzhivaniya $H_2/H_2/1$ // Sistemy upravleniya i informacionnye tekhnologii. 2014. V.3(57).P. 92-96.

UDC: 004.75

The Automata-based Approach to Large Systems Control in the Global Computer Environment

Yu.S. Zatuliveter [1] and E.A. Fishchenko [1]

[1]Institute of Control Sciences of Russian Academy of Sciences, 65 Profsoyuznaya street, Moscow 117997, Russia

zvt@ipu.rssi.ru, elena.fish@mail.ru

Abstract

With the increase in the size of distributed systems implemented in a global computer environment, mathematical methods for formalizing their development and functioning become one of the important problems. The article proposes the automata-based approach to distributed systems representation and control of their operation using balance-based equations. A feature of the automata-based model is that the control complexity in arbitrarily large distributed systems ceases to depend on the size of the distributed network systems and the problems they solve.

Keywords: global computer environment, heterogeneity, large distributed system, automata-based model, balance-based equation

1. Introduction

The Global computer environment (GCE) forms a qualitatively new information infrastructure (in its digital universality) with many network computing nodes. The number of network nodes totals tens of billions and continues rapid growth, expanding the GCE influence's scope.

In three decades of spontaneous, systemically unbalanced growth of the computer environment, there was a de facto global (in computer-network execution) violation of the Ashby principle [1] – one of the main principles of cybernetics: "For the implementation of processes control, the diversity of the control subsystem must be no less than the diversity of the controlled subsystem."

Currently, the GCE does not provide such opportunities. The reasons are that the existing GCE, consisting of a large number of locally universal computer devices connected by networks, at the global level of its total resources does not have the system-wide quality of functional completeness (universal programmability). Absent

this quality makes it impossible to time and continuously update the control part of systems according to the exponential growth rate of information coming from the controlled part of the systems.

One of the main reasons for the inconsistency of modern GCE with the requirements of the Ashby principle is the initial heterogeneity of hardware, software, and information network resources. GCE as a whole, due to the lack of a system-wide property of functional completeness, cannot be considered as a universally programmable carrier of globally distributed control processes.

Heterogeneity is the fundamental system-technical reason limiting the further growth of size distributed systems in the GCE. In [2] the reasons for continuous reproduction of heterogeneity of GCE and ways of their elimination are shown. Another fundamental limitation of the size of distributed systems in the GCE is the insufficient development of mathematical methods for formalizing such systems and control processes in them.

We present the principles of constructing an automata-based model, which, using the example of Data Flow models, shows the possibilities of control by distributed computing in the mode of their dynamic parallelization. A feature of the proposed model is that there is no direct dependence of the complexity of the parallelism control for distributed computing on the size of distributed systems and tasks, which coincides with the total number of computational operations.

2. Information graphs for the mathematical presentation of tasks

We assume that the initial computational tasks view a bipartite information graph, in which directed arcs connect nodes of two different types – operators and objects. Operator nodes represent computational actions, while object nodes represent variables that are arguments or values of operators. Model Data Flow uses this form of task representation, which is non-procedural.

The information graph $\mathbf{G} = \langle \mathbf{A} \cup \mathbf{U}, \mathbf{S} \rangle$, where $\mathbf{A} = \{a_1, a_2, \ldots\}$ is the set of operator nodes, $\mathbf{U} = \{u_1, u_2, \ldots\}$ is the set of object nodes, $\mathbf{S}=\mathbf{S(A,U)} \cup \mathbf{S(U,A)}$ is the set of oriented arcs $S(a,u)$ and $S(u,a)$, respectively.

In this case, $\mathbf{S(A,U)} = \{S(a,u) \neq \varnothing : a \in \mathbf{A}, u \in \mathbf{U}\}$ are the all arcs from the operators to the objects, $\mathbf{S(A,U)} = \{S(a,u) \neq \varnothing : a \in \mathbf{A}, u \in \mathbf{U}\}$ – from the objects to the operators.

All operator nodes $a \in \mathbf{A}$ and object $u \in \mathbf{U}$ have the following nodes types: $b(a) \in \mathbf{B}$ and $d(u) \in \mathbf{D}$, respectively, where $b(a) \in \mathbf{B}$ and $d(u) \in \mathbf{D}$ is a basic set of computational operations/functions $\mathbf{B} = \{b_1, b_2, \ldots, b_m\}$, which we will call operator types, and $\mathbf{D} = \{d_1, d_2, \ldots, d_n\}$ is a basic set of data types. Arcs $\mathbf{S(U,A)}$ for each operator determine the occurrence of objects (variables) as arguments (input data),

arcs **S(U,A)** – the transfer of calculated values (output data) to objects (variables) that take calculated values.

Information graphs one-to-one relate with a system of formula expressions in explicit form, in which object nodes have unique variable names, data types, and operator nodes have unique operator identifiers and their types of operations.

3. Control automata

The order of actions in the information graph is determined by the rule of automata-based asynchronous parallelization, which is the basis of Data Flow. Namely: each operator can be executed if the calculated values of all its arguments (input variables) are received and entered, and there are free memory cells in which the calculated values of objects store during computing.

The Data Flow model, as is known, provides asynchronous dynamic identification of the maximum possible parallelism of computational operators in information graphs.

Let us build the model of such control in the form of an automata network. We introduce two types of automata for constructing such control networks.

Every $a \in \mathbf{A}$ has two subsets of objects (variables):

$\mathbf{U}^-(a) = \{u \in \mathbf{U} : S(u,a) \neq \oslash\}$ - is the subset of input arguments, and

$\mathbf{U}^+(a) = \{u \in \mathbf{U} : S(a,u) \neq \oslash\}$ - is the output is the subset of values calculated by the operators of $a \in \mathbf{A}$.

Every $u \in \mathbf{U}$ has two subsets of operators:

$\mathbf{A}^-(u) = \{a \in \mathbf{A} : S(a,u) \neq \oslash\}$ – is the input subset of preceding operators, and

$\mathbf{A}^+(u) = \{a \in \mathbf{A} : S(u,a) \neq \oslash\}$ – is the output subset of operators that use the values of $u \in \mathbf{U}$ objects.

For each operator and object, we assign an A-automaton (Figure 1a) and a U-automaton (Figure 1b), respectively. During the execution of distributed computing processes, control automata mathematically determine the behavior of operator and object of information graphs and the interactions between neighboring automata.

The automata are connected in a control network. Each arc of the information graph S(a, u) and S(u, a) corresponds to two oppositely directed connections of forward and reverse synchronization, through which interactions between automata of different types are carried out.

Both types of automaton, in the considered simplest case, have three states.

A-automatons implement control by data flows: an operator can be executed if the values of its arguments (input variables) are defined. The alphabet of states of the **A**-automaton is $\{p, r, w\}$, where the p denotes "operator is passive" (not all input values are calculated still), the r - "operator is executing," and the w - "operator is waiting" for the result to be unloaded.

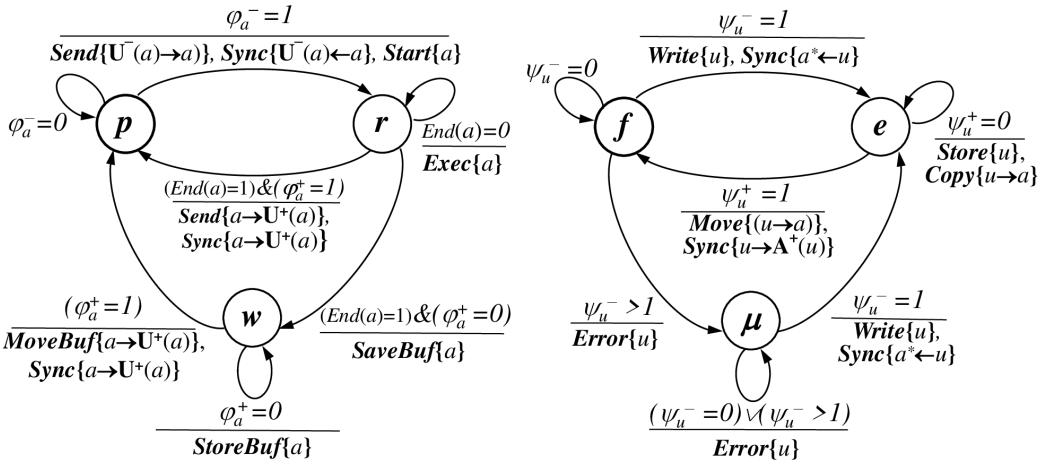


Fig. 1. Control automata

U-automatons implement the principle of object value protection: new data cannot be written to the object value storage memory until the previous values are fully used. The alphabet of states of the **U**-automaton is $\{f, e, \mu\}$, where the state f corresponds to the "free memory of the object," state e - to the "occupied memory" (storing the value), μ - to the state of uncertainty due to a conflict on the record from two or more operators (the diagnostic result of detecting incorrectness of information graphs).

For direct and reverse synchronization of neighboring automatons, every **A**-automaton attached to node $a \in \mathbf{A}$ and **U**-automaton attached to node $u \in \mathbf{U}$ have, respectively, four numeric variables, which during calculations take the values: $k^+(u), k^-(a), k^+(u), k^-(u) \in N_0 = \{0, 1, 2, \dots\}$.

Before the beginning of the calculations, these variables must have the following initial values:

$k^+(a) = \left|\mathbf{U}^+(a)\right|$ – size of subset $\mathbf{U}^+(a)$, $a \in \mathbf{A}$; $k^-(a) = \left|\mathbf{U}^-(a)\right|$ – size of subset $\mathbf{U}^-(a)$, $a \in \mathbf{A}$; $k^+(u) = \left|\mathbf{A}^+(u)\right|$ – size of subset $\mathbf{A}^+(u)$, $u \in \mathbf{U}$; $k^-(u) = 0$ for all $u \in \mathbf{U}$.

At the moments of transitions (Figure 1), the automata connected to the network perform internal (inter-automatons) synchronization interactions, as well as "external"

synchronization interactions between the automata and their nodes of the information graph.

Inter-automata interactions are carried out by calculating the threshold indicator functions $\varphi_a^-, \varphi_a^+, \psi_u^-, \psi_u^+$ associated with the graph's nodes.

Interactions with the graph nodes are performed using threshold indicator functions that determine the initial and final moments of changing the modes of operation of the graph nodes and other functions of performing information actions related to processing information during the computational process of the information graph.

Threshold indicator functions calculate the transition condition between the states of automata (Figure 1). Three of the 4 types of these functions are defined as the result of the comparison with 0: $\varphi_a^- = (k^+(a) = 0), \varphi_a^+ = (k^-(a) = 0), \psi_u^+ = (k^+(u) = 0)$. The fourth type of such interaction is defined as the result of a comparison with 1: $\psi_u^- = (k^-(u) = 1)$.

Each **A**-automaton during the automata interactions controls the switching of the operating mode of the operator utilizing the following signal functions (Figure 1a):

Start $\{a\}$ - beginning execution the operator,

$End(a)$ - asynchronous signal from this node ($End(a) = 1$) about ending execution.

4. Balance-based equation of control processes in large systems

The function of control a distributed process of computing on information graphs implement by the network of $\mathbf{A}, \mathbf{U}$ - automata. Each node of the graph is associated with an automaton of the corresponding type. Each automaton of such a network interacts with the automata of neighboring nodes connected by oriented arcs. Therefore, the size of the automata network is equal to the number of nodes in the graph. In this case, the complexity of the control network significantly depends on the size of the solved problem represented by the graph.

We build a mathematical model using **A,U**-automata that embodies the asynchronous parallelization of calculations that underlie Data Flow. The complexity of control distributed processes in the proposed model does not depend on the size of the information graphs.

This model is the balance-based equation (below referred to as the "balance equation") of control processes, which show the example of information graphs. The model gives an analytical expression of control processes in large distributed systems.

The sets of the **A** and **U** nodes of the graph at each step j, taking into account the current state of the corresponding automata, represent as partitions into disjoint subsets of nodes whose automata are in the identical states:

$$\begin{cases} U \equiv U^f(j) \cup U^e(j) \cup U^\mu(j), \\ A \equiv A^p(j) \cup A^r(j) \cup A^w(j), \\ j = 0, 1, 2, \cdots \end{cases} \quad (1)$$

It should be noted that the set at $\mathbf{A}^r(j)$ each step j defines all operators that can be compute in parallel.

Interaction of the information graph nodes leads to a change in the composition of subsets of nodes (1) in identical states. The following system of recurrence equations defines the rules for updating subsets:

$$\begin{cases} \mathbf{A}^p(j+1) = (\mathbf{A}^p(j) \cup \Delta\mathbf{A}^{wp}(j) \cup \Delta\mathbf{A}^{rp}(j)) \setminus \Delta\mathbf{A}^{pr}(j); \\ \mathbf{A}^r(j+1) = (\mathbf{A}^r(j) \cup \Delta\mathbf{A}^{wp}(j)) \setminus (\Delta\mathbf{A}^{rp}(j)) \cup \Delta\mathbf{A}^{rw}(j)); \\ \mathbf{A}^w(j+1) = (\mathbf{A}^w(j) \cup \Delta\mathbf{A}^{rw}(j)) \cup \Delta\mathbf{A}^{wp}(j)); \\ \mathbf{U}^f(j+1) = (\mathbf{U}^f(j) \cup \Delta\mathbf{U}^{ef}(j)) \setminus (\Delta\mathbf{U}^{fe}(j)) \cup \Delta\mathbf{U}^{f\mu}(j)); \\ \mathbf{U}^e(j+1) = (\mathbf{U}^e(j) \cup \Delta\mathbf{U}^{fe}(j) \cup \Delta\mathbf{U}^{\mu e}(j)) \setminus (\Delta\mathbf{U}^{ef}(j)); \\ \mathbf{U}^\mu(j+1) = (\mathbf{U}^\mu(j) \cup \Delta\mathbf{U}^{f\mu}(j)) \setminus \Delta\mathbf{U}^{\mu e}(j) \\ j = 0, 1, 2, \cdots \end{cases} \quad (2)$$

The composition of the sets in the left part (2) defines the balance of the transition flows of the sets of operators and objects in the right part, which we have designated $\Delta\mathbf{A}^{qs}$ and $\Delta\mathbf{U}^{qs}$, respectively. Their automata transitions from state q to state s on clock cycle j, where either $q, s\epsilon\{p, r, w\}$, or $q, s\epsilon\{f, e, \mu\}$.

5. Conclusion

The model of balance equations radically reduces the dimension of the state space of distributed processes compared to the multiplicative number of states of the entire network based on automata. The complexity of control distributed processes ceases to depend significantly on the size of distributed network systems and the tasks they solve. The proposed method of formalizing of distributed systems is intended for solving solve problems of modeling and controlling distributed processes in arbitrarily large computing environments.

REFERENCES

1. Ashby W.R. Introduction to Cybernetics. 1956. Chapman & Hall. `http://pcp.vub.ac.be/books/IntroCyb.pdf`.
2. Zatuliveter Yu. S., Fishchenko E. A. About the Universal Algorithmic Space of Distributed and Parallel Computing./ Proceedings of the 11th International Conference "Management of Large-Scale System Development" (MLSD). Moscow: IEEE. 2018. P.1-5. DOI:10.1109/MLSD.2018.8551799.

UDC: 004.046

Statistical Method for Support of Responsible Decision

A.A. Grusho[1], N.A. Grusho[1], M.I. Zabezhailo[1], E.E. Timonina[1]

[1] Federal Research Center "Computer Science and Control" of the Russian Academy of Sciences, Vavilova 44-2, 119333, Moscow, Russia

Abstract

The paper is devoted to the problem of evaluation of trust to the results of complex computer analysis of data. The approach of constructing empirical dependencies based on similarity of precedents in the training sample, which has already become classical, is used. The trust approximation is based on simulating training data by random sampling from an unknown distribution. This approach implements approximate causal analysis and have advantages and disadvantages.

Keywords: Information security; artificial intelligence; trust to distributed computing

1. Introduction

System administrators and security officers meet with the necessity to make a responsible decision based on empirical and sometimes incomplete data in large distributed information systems (DIS). The question arises whether the result of the execution of information technology (IT) in DIS is reliable. That is, whether the extraordinary results obtained as the result of monitoring can be the base for the presence of an anomaly or they are the result of correct calculations of extraordinary initial data.

Classical approaches to the study of trust in the decision making with using computer modeling may be inadequate in the situation of Big Data, limited time, IT openness in terms of new data. Serious problems arise due to incomplete information about the object of the study in information security (IS) applications, and due to complexity of calculations.

The effective alternative approach [1, 2] in this class of problems is the usage of the so-called interpolation-extrapolation mathematical models. In this approach, the initial data are presented in the form of descriptions of precedents, i.e. examples and counterexamples of the target phenomenon being studied, combined into a so-called

training sample. Further, this sample is interpolated by empirical dependencies (ED) of a particular form. Conclusion about presence or, conversely, absence of target effect in newly analyzed precedent is formed by means of check of extrapolation on it of ED. To date, a number of actively used implementations of this approach are known, based on both statistical and deterministic mathematical techniques of data analysis and decision making.

In a number of applications using computer-based data analysis, responsibility for the aftereffect of decisions is critical. In particular, such tasks include the providing of the information security of the critical information infrastructure [3]. As the result, we are talking about meaningful informal explainability of the conclusions and recommendations formed. That is, you need to get an answer not only to the question "how," but also to the question "why." The required properties of understandability, interpretability and stability of the results should be naturally provided [4]. It can be done in those versions of computer data analysis that are based on the causal relationships.

The creation of an explanation involves at least two components:

1) the generation of a variant of the argued, i.e. substantiated, non-contested, evidence-based causal "scheme," which provides an appropriate answer to the question "Why should you trust the result?";
2) the generation of a meaningful language, i.e. informal, not duplicating the method used to produce the result, interpretation for the answer to the question "Why...?" generated within the framework of the causal "scheme."

2. Evaluation of Trust Using Probabilistic Statistical Methods

The approximate method of estimating the causal bases of the obtained IT result based on probabilistic-statistical analysis of precedents is considered. The main idea of the result is as follows.

Suppose that the system administrator observes and accumulates the results of executing of IT. Each j-th copy of IT has its own source data x^j that belongs to the definition range D. The results of the IT^j execution are denoted y^j and belong to the range of values B. Each range B has a division into extraordinary range of IT outcome data B^+ and range of ordinary IT outcome data B^-. Similarly, the initial data is divided into D^+ and D^-. If y^j belongs to B^+, then this can be caused either by extraordinary source data, or by an implicit anomaly of the computational process. In this case, additional information is needed to make a decision. In the paper [5], the association of the cluster structure of empirical data with the measure of proximity associated with the probabilistic distribution, according to which the data are selected, is constructed. Suppose that the initial IT data is derived from

probabilistic distributions on D^+ and D^-. Let these probabilistic distributions have single-humped density of probabilities in space with a measure of proximity ρ. Then (see [5]) the closer data corresponds to belonging to the high probability area and therefore includes into a single cluster. This is due to the fact that significantly more data includes into the high probability area than into the low probability area. The data entering the low probability area is located at a greater distance from each other.

Let for ordinary data there is its own distribution that generates its own cluster. Therefore, including data into different clusters corresponds to different distributions (different statistical hypotheses). Causal bases for estimation of trust to computer results can be substantiated by comparing the densities of single-humped distributions. Consider the example of such substantiation.

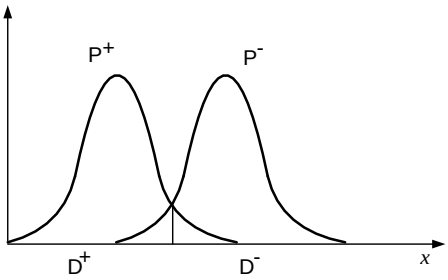


Fig. 1. The example of a contradiction in the case of single-humped distributions.

In Fig. 1 P^+ denotes the density of the probability distribution of the occurrence of extraordinary data, respectively, P^- - the density of the probability distribution of the occurrence of ordinary data. If the result of the computer calculation $y \in B^+$, and x is located near the maximum of density P^+, then it follows from the difference in densities P^+ and P^- that the difference ordinate $P^+(x) - P^-(x) > 0$. This condition does not contradict to $y \in B^+$, and therefore increases the trust to the result of computer result. If $y \in B^+$ and x is located near the maximum of density P^-, i.e. far from $\max P^+$, then the difference ordinate $P^+(x) - P^-(x) < 0$. Hence with a high probability $x \in D^-$, which contradicts the condition $y \in B^+$.

Since we cannot talk about exact values of functions in the tasks under consideration, comparative analysis of causal bases is convenient to express in terms of the concept of "contradiction," as shown above.

In the case of multidimensional data, cluster methods based on this proximity ideology are easier computed and required less initial data then in estimation of distribution density. The main problem of using this method is the choice of an appropriate measure of proximity, consistent with the "humps" of distributions.

To evaluate the causal bases, we construct the method that at least depends on the conjugation of "humps" of distribution density with measures of proximity. Let $x_1^+, ..., x_n^+$, $n > 1$, be the data of the training sample from D^+. Suppose that space D^+ is arranged so that for D^+ there is a division into local areas $D_{(1)}, ..., D_{(s)}$, $D_{(1)} \cup ... \cup D_{(s)} = D^+$, $D_{(i)} \cap D_{(j)} = \emptyset$, $i \neq j$, which allow to efficiently calculate the data belonging to each of these areas. These arias have approximately the same size according to the proximity ρ. Then, using the data $x_1^+, ..., x_n^+$, $n > 1$, it is possible to construct frequencies of these data appearance in the areas. Let these frequencies can be ordered in descending order, i.e. the set of order statistics is constructed:

$$\nu_{[1]} \geq \cdots \geq \nu_{[s]}. \tag{1}$$

Let's order the local areas $D_{(1)}, ..., D_{(s)}$ in according to the set of order statistics (1):

$$D_{[1]} \succ \cdots \succ D_{[s]}. \tag{2}$$

where $D_{[i]} \succ D_{[j]}$ in only case when $\nu_{[i]} \geq \nu_{[j]}$.

Then, according to the relationship between the "hump" of distribution density and the concentration of data around this "hump", we obtain the empirical order of local areas in accordance with the decrease in their probabilities. From here we get the approximate order of areas due to data $x_1^+, ..., x_n^+$, $n > 1$, and the method of determining the data belonging to the "hump" of probability density for the case of extraordinary data.

Let x be new data that generates the extraordinary result. If the data x belongs to the local area having a low number in order (2), this is the causal bases that the IT result is derived from the original data and not from anomalies. If the obtained data x belongs to the area with a high number in order (2), then they can refer to both D^+ and D^-. In this approach it is required that the local areas with small numbers in (2) are strictly ordered.

Example 1. Let data space D^+ be a family of balls in a space of a finite dimension, and the radii of all balls are the same. The IT result takes two values y_1, y_2, where y_1 is the extraordinary value. If the radii of the balls are not large, then there exist effective algorithms for calculating the belonging of data x to a ball.

The one-humped distribution in this scheme can be constructed, for example, using probabilities $p_{i_1} > ... > p_{i_m}$ that describe the independent sampling of balls $D_1, ..., D_m$ in the case of D^+. If all radii are equal to each other, the distributions inside the balls $D_1, ..., D_m$ are uniform, then obviously we get a one-humped distribution on D^+.

In this example, it is assumed that the distributions on D^+ and D^- are not known, but are one-humped. Then the initial training data $x_1^+, ..., x_n^+$, $n > 1$, effectively

determine the frequencies of occurrence of the balls and the order (2). Therefore, the appearance of new data x can be considered as the causal bases of an extraordinary result if the ball to which x belongs has small number in the order (2).

Consider some positive and negative aspects of the probability-statistical approach to the analysis of causal bases. This approach can be applied when it is possible to construct (at least indirectly) the sequence (2). In turn, this is possible when the source data is considered as a part of large areas. However, increasing the size of the area leads to a complication of the algorithm for calculating the belonging of the data to the corresponding areas.

3. Conclusion

The work is devoted to the usage of evaluation of causal bases for increasing trust to the results of computing. Trust in computing results is not an exactly computable characteristic. Therefore, the results of the paper are not focused on accurate estimates of causal bases.

It should be emphasized that approximate studies of causal bases have no purpose to accurately describe the causes of the studied property. Approximate estimates aim to independently interpret and increase of trust to computer results. Accompanying complex calculations by constructing estimates of causal bases allows us to interpret the result of computer calculations in terms of inconsistency of input and output data. The results of the paper were supported by practical system administrators.

REFERENCES

1. Rudakov K. V. Completeness and universal limitations in the problem of correction of heuristic classification algorithms // J. Cybernetics. 1987. V. 3. P. 106–108.
2. Zhuravlev Yu. I. Correct algebras over sets of incorrect (heuristic) algorithms. I-III // J. Cybernetics. I: 1977. V. 4. P. 5–17; II: 1977. V. 6. P. 21–27; III: 1978. V. 2. P. 35–43.
3. DARPA Sets Up Fast Track for Third Wave AI. – Jul 26, 2018, `https://defence.pk/pdf/threads/darpa-sets-up-fast-track-for-third-wave-ai.569563/`.
4. Gunning D., Aha D. DARPA's Explainable Artificial Intelligence (XAI) Program // J. AI Magazine. 2019. V. 40. No. 2. P. 44–58.
5. Grusho A. A. Statistical significance criteria for cluster structures based on pairwise proximity measures // OPPM Surveys on Applied and Industrial Mathematics. 1996. V. 3. No. 1. P. 43–46.

UDC: 004.057.4; 004.057.7

Evaluation of reactive routing protocols performance under malicious attacks in VANET

A.A. Sabbagh [1] and M.V. Shcherbakov[1]

[1]Volgograd State Technical University, Volgograd, Russian Federation

Abstract

Changing over digital age is becoming the driving force for modern approaches to provide a safe road system. VANETs play an important role in Intelligent Transportation System (ITS) due to their increasing importance for the building of it. The Wireless networks are more vulnerable to malicious attack than traditional wired networks. Thus, security is much more difficult in VANET networks than in other networks. This paper evaluates the performance of two reactive routing protocols under black hole attack, and analyzes which of protocols are most affected by the attack. The performance was experimentally measured using some of metrics like packet delivery ratio, throughput, average delay, overhead and packet loss ratio. The simulation is carried out in NS-3 simulator to create VANET network, and Bonnmotion to generate realistic mobility scenarios. *abstract* environment.

Keywords: reactive routing protocol, black hole attack, VANET, network performance, NS-3

1. Introduction

The routing protocols in VANET are an extension of the traditional routing protocols in MANET, taking into consideration that the best protocol in MANET under certain environmental conditions is not necessarily the best for VANET networks, although there are many common characteristics between the two networks. However, VANET is characterized by special characteristics such as high node traffic, strict delay restrictions and frequent interruption of communication, which makes the task of the routing protocol very difficult in choosing the stable route to keep the connection as long as possible [1]. The routing protocol is responsible for choosing the path between the nodes to route the data from the source to the destination. The more secure and stable path, the higher the network throughput and the delivered packets, thus improving the network performance and reaching the desired goal of this network. Routing protocols in ad-hoc networks are divided into three types -

proactive, reactive and hybrid [2], [3]. Reactive routing protocols are known as on-demand routing protocols as each node does not have information about the entire network topology, but stores only the effective route that is currently used to reach the destination. This type of protocol depends on performing the process of discovering the path to the destination only, when necessary, thus reducing the load of process of periodic updating of routing tables in network, but increasing the time required to discover the path between the two nodes. The most common reactive routing protocols are AODV (Ad Hoc On-Demand Distance Vector), DSR (Dynamic Source Routing) [4], [5]. In this paper we chose AODV and DSR routing protocol to study performance and analyze the results of the impact of malicious attacks on performance. Below are some more details on how each of these two protocols work.

1.1. Ad hoc On-demand Distance Vector (AODV) routing protocol. AODV is one of the most popular ad-hoc routing protocols. It depends on the mechanism of the functionality of interactive routing protocols in terms of road discovery and maintenance. It is type of hop-by-hop routing protocol, it is not necessary for the source node to have complete information about the path t destination node, but rather it only needs to know the next hop in the path and so on until the destination is reached. Therefore, each node is responsible for making independent and local routing decisions based on packets destination addresses and their route computation. AODV provides unicast, multicast and broadcast communication. It performs two types of processes: route discovery and route maintenance [6].

1.2. Dynamic Source Routing (DSR). DSR is one of the most popular ad-hoc routing protocols. It is reactive routing protocol which depends on the concept of source routing. It uses its route caches to store new routes. When a new route is detected or when there is a direct route between source and destination node, DSR updates its cache. Therefore, the source node has a complete information about the path to destination node It performs two types of processes: route discovery and route maintenance. From the above, we can notice that although DSR and AODV belong to the same type of Ad-Hoc routing protocols, they have some differences. in Route Discovery process, every node in DSR transmits the whole path unlike node in AODV that sends only the next- hop. Also, the discovered route is being maintained for period of time and stored in route cache of node while in AODV it is kept in the routing table [6].

2. Security Challenges in VANET

Ad-hoc networks have some characteristics such as flexibility, decentralization, topology changes, and the wireless medium to establish communications between nodes [7]. These characteristics make it suitable for linking between vehicles, and themself exploited by attackers to gain control of the network. Also, the expansion

of the VANET networks to cover the largest possible geographical area and the participation of the largest number of cars in one network helps attackers to choose any location where radio coverage of one of the nodes is located in order to access the network and work to control it by intercepting messages that are transmitted over the network and redirecting or blocking them. These malicious acts carried out by the attacker cause weak links between the nodes on the network by isolating the nodes from each other and then transmitting misleading messages on the network, which makes the network lose its effectiveness due to the lack of information needed by the cars in a desired time [8].

2.1. VANET routing attacks. Implementation of a security system for vehicular ad-hoc network is difficult and challenging due to, inherent characteristics such as high mobility with frequent disconnection and using wireless channels for exchanging important and safety information, leads to several security hazards and attacks in VANET. These challenges expose VANETs to various kinds of dangerous attacks. In particular, in order to provide secure communication in VANETs, a significant knowledge of threats and attacks is necessary to overcome all security challenges. Attacks against routing protocols are as follows: grey hole, black hole attack, sink hole attack, worm hole attack, and sybil attacks [9]. In this paper we focused on black hole attack which considered as serious attack. Therefore, it is necessary to identify its mechanism of operation and its method of controlling the network and study its effect on the performance of the protocol in the network in order to overcome it and try to isolate it in the future work.

2.2. Black hole attack. It is a known attack in manet networks which fools other nodes into sending packets through it. The attacker can exploit the vulnerability in the path discovery process for on-demand routing protocols where the attacker's node misuses the protocols by announcing itself that there is a better path with a low number of hops to the destination node, then the node picks up the packets and drops them in the second step. This may disable the network and prevent an important message from appearing to the recipients due to the malicious node [10]. The consequences of attacking in VANETs are more serious because loss of packets in safety-related applications may cause life-threatening accidents [11].

3. Simulation Methodology

Simulation was performed with a bonnmotion generator and network simulator NS3. First, we used the bonnmotion tool to get a Manhattan model, which uses the Manhattan Grid model that represents streets within a city to simulate movement in an urban area. After generating a realistic vehicle traffic scenario to simulate the traffic pattern of mobile nodes on the streets, the generated file (scenario.ns movements) is used as input for the network simulation NS3 where the VANET

network and routing protocol modeling are performed to study and analyze network performance under multiple black hole attacks. The performance of the protocols is evaluated in measuring the five well-known measures of packet-to-delivery ratio (PDR), throughput, overhead, and packet loss and delay ratio and the analysis is presented below. This research focuses on measuring the performance of AODV and DSR for the identified scenario in the VANETs. Scenarios with different attacks (2,4,6,8,10) involving 50 vehicular nodes to measure the scalability impact under black hole attacks. The simulation parameters we considered are stated in Table 1.

Parameter	Value
$Simulator$	NS-3.23
$Topology$	Manhattan grid road network, size 5x5 with 2000m
$Numberofnodes$	50
$Numberofattack$	2,4,6,8,10
$PropagationModel$	Log Distance Propagation Loss Model
$Packetsize$	500 bytes
$Traffictype$	CBR, UDP
$VANETtopologygenerationtool$	Bonn Motion
$RoutingAlgorithm$	AODV, DSR
$Macprotocol$	802.11b standard
$SimulationTime$	100Sec

Table 1. simulation parameters

4. Results and performance analysis

Comparison of two routing protocols AODV and DSR is carried out in VANET with black hole attack to analyze which protocol is more vulnerable to black hole attack. As shown in Fig. 2 , the packet delivery ratio (PDR) results for AODV and DSR routing protocol. It can be observed that AODV achieved 2 % more PDR value than DSR with increased a number of attacks until 20% in the network. That is because in AODV protocol all the intermediate nodes share the routing load, i.e., every node along the path uses the latest routing information to forward the packets. High PDR expresses the efficiency of ad hoc routing protocols. The results of the throughput and overhead are illustrated in Fig. 3,4 show that both parameters are increasing in the both protocols. DSR gives a higher throughput than AODV but in the same time gives overhead 13% more than AODV. we can conclude that DSR generates more overhead packets and it is not suitable for large network. As for the average delay in Fig. 5, it increased by almost 700% in both protocols, which calls

for the need to develop the protocols to become more secure with a suitable delay rate for VANET.

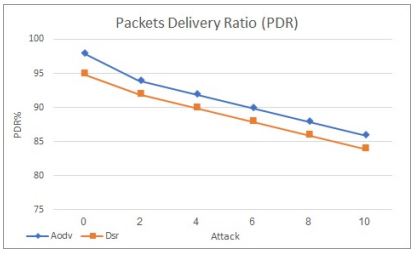


Fig. 1. Packets Delivery Ratio

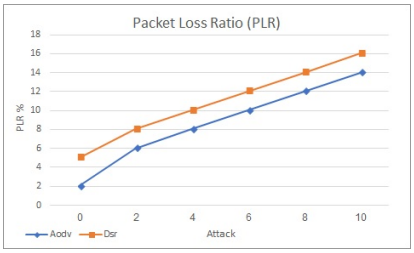


Fig. 2. Packets Loss Ratio

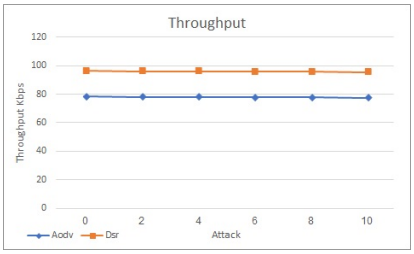


Fig. 3. Throughput

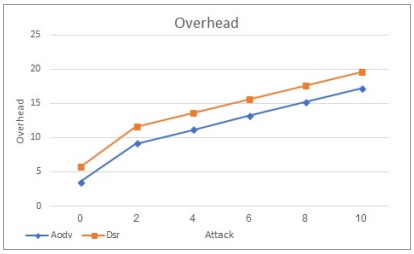


Fig. 4. Overhead

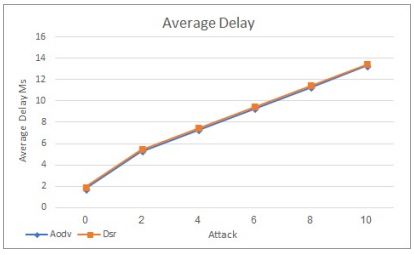


Fig. 5. Average Delay

5. Conclusion

This paper presented the results of evaluating of reactive routing protocols performance (AODV, DSR) in VANET under black hole attack to specify the impact of malicious attacks on the network and which routing protocol is more secure when increasing the number of black hole. From the above result analysis we can say that

the AODV protocol outperforms than DSR protocol when we consider the attack in terms of PDR, overhead and PLR.

REFERENCES

1. I. Wahid, A. A. Ikram, M. Ahmad, S. Ali, A. Ali, State of the art routing protocols in vanets: A review, Procedia computer science 130 (2018) 689–694.
2. M. Jain, R. Saxena, Overview of vanet: Requirements and its routing protocols, in: 2017 International Conference on Communication and Signal Processing (ICCSP), IEEE, 2017, pp. 1957–1961.
3. V. N. Talooki, K. Ziarati, Performance comparison of routing protocols for mobile ad hoc networks, in: 2006 Asia-Pacific Conference on Communications, IEEE, 2006, pp. 1–5.
4. D. T. Le, R. Kirichek, A. Shestakov, et al., Research on using the aodv protocol for a lora mesh network, in: International Conference on Distributed Computer and Communication Networks, Springer, 2020, pp. 149–160.
5. Z. S. Houssaini, I. Zaimi, M. Oumsis, S. E. A. Ouatik, Comparative study of routing protocols performance for vehicular ad-hoc networks, International Journal of Applied Engineering Research 12 (13) (2017) 3867–3878.
6. A. A. Al-khatib, R. Hassan, Performance evaluation of aodv, dsdv, and dsr routing protocols in manet using ns-2 simulator, in: International Conference of Reliable Information and Communication Technology, Springer, 2017, pp. 276–284.
7. R. Kaur, H. Kaur, Performance evaluation of routing protocols in vanet, International Journal of Future Generation Communication and Networking 8 (6) (2015) 239–246.
8. P. Tyagi, D. Dembla, A taxonomy of security attacks and issues in vehicular ad-hoc networks (vanets), International Journal of Computer Applications 91 (7) (2014).
9. F. Ishmanov, Y. Bin Zikria, Trust mechanisms to secure routing in wireless sensor networks: Current state of the research and open research issues, Journal of Sensors 2017 (2017).
10. F. Sakiz, S. Sen, A survey of attacks and detection mechanisms on intelligent transportation systems: Vanets and iov, Ad Hoc Networks 61 (2017) 33–50.
11. M. S. Al-Kahtani, Survey on security attacks in vehicular ad hoc networks (vanets), in: 2012 6th international conference on signal processing and communication systems, IEEE, 2012, pp. 1–9.

UDC: 519.872

Asymptotic Diffusion Analysis of an Retrial Queueing System M/M/1 with Impatient Calls

E.Yu. Danilyuk[1], S.P. Moiseeva[1], A.A. Nazarov[1]

[1]National Research Tomsk State University, 36 Lenina ave, Tomsk, Russian Federation

daniluc.elena.yu@gmail.com, smoiseeva@mail.ru, nazarov.tsu@gmail.com

Abstract

In the paper, the retrial queueing system of M/M/1 type with input Poison flow of events and impatient calls is considered. The service time, delay time of calls in the orbit and the impatience time of calls in the orbit have exponential distribution. Asymptotic diffusion analysis method is proposed for the solving problem of finding distribution of the number of calls in the orbit under a long delay of calls in orbit and long time patience of calls in the orbit condition.

Keywords: Retrial queueing system, Impatient calls, Asymptotic diffusion analysis

1. Introduction

At the present time retrial queueing systems (RQ-systems) research is in the demand as evidenced by numerous papers in this area and grants support. The systems as mathematical models are very suitable for modern telecommunication systems, networks, mobile networks describing. Along with the construction of mathematical models of RQ-systems, new methods of their study are being developed. A fairly new method is asymptotic diffusion analysis method, as a modification of the asymptotic analysis method. Both of them are suggested by the Tomsk research school, and there are interesting works [1, 2, 3, 4], in which the asymptotic diffusion analysis method is used.

The present paper is devoted to study of the retrial queueing system of M/M/1 with impatient calls by the asymptotic diffusion analysis method.

The repoted study was funded by the RFBR and Tomsk region according to the research project No.19-41-703002.

2. Mathematical Model

We consider an retrial queueing system with one server and Poisson arrival process with intensity λ. An arriving call (or customer) that has found the service device free takes it for the service for a random time distributed exponentially with parameter μ. If the device is busy, calls that arrive go into the orbit. On the orbit, each call, independently of others, waits for a random time whose duration has an exponential distribution with parameter σ, and then again accesses the device with a second attempt to obtain servicing. If the device is free, the call from orbit occupies it for random servicing time. If the device is busy, call immediately goes into the orbit and wait once more random time. Moreover, a call from the orbit leaves the system after exponential distributed time with parameter α, demonstrating the "impatience" property.

The problem is to find the stationary distribution of the number of calls in the orbit. This problem has been solved in [5] by the asymptotic analysis method under a long time patience of calls in the orbit condition. In the present paper we use asymptotic diffusion analysis under a long delay of calls in orbit and long time patience of calls in the orbit condition to study the stationary distribution $P(i)$ of the number of calls in the orbit.

3. Process of the System States: System of Kolmogorov Differential Equations in terms of Partial Characteristic Functions

Let us consider Markovian process $\{k(t), i(t)\}$ determined states of the considered RQ-system where $i(t)$ is the number of calls in the orbit at the moment t, $i(t) = 0, 1, 2, 3, \ldots$, $k(t)$ defines device state at the moment t and takes one of the following values: $k(t) = 0$, if server is free at the moment t, and $k(t) = 1$, if server is busy at the moment t.

Denote as $P_0(i,t) = P\{k(t) = 0, i(t) = i\}$ and $P_1(i,t) = P\{k(t) = 1, i(t) = i\}$ the probability that, at the moment t, there are i calls in the orbit, $i = 0, 1, 2, \ldots$, and the service device is free or the server is busy respectively.

Introduce the partial characteristic functions

$$H_k(u) = \sum_{i=0}^{\infty} e^{jui} P_k(i,t), \quad k = 0, 1, \quad j = \sqrt{-1}. \tag{1}$$

To obtain the probability distribution $P_0(i,t), P_1(i,t)$ for the states of the retrial queue M/M/1 with impatient calls in the orbit, we construct a system of Kolmogorov differential equations [5] and write it in terms of partial characteristic functions (1)

$$\begin{cases} \dfrac{\partial H_0(u,t)}{\partial t} = -\lambda H_0(u,t) + \mu H_1(u,t) + j\left(\sigma + \alpha\left(1 - e^{-ju}\right)\right)\dfrac{\partial H_0(u,t)}{\partial u}, \\ \dfrac{\partial H_1(u,t)}{\partial t} = \lambda H_0(u,t) - \mu H_1(u,t) - \lambda\left(1 - e^{ju}\right) H_1(u,t) - j\sigma e^{-ju}\dfrac{\partial H_0(u,t)}{\partial u} \\ +j\alpha\left(1 - e^{-ju}\right)\dfrac{\partial H_1(u,t)}{\partial u}. \end{cases} \quad (2)$$

In adding the first equation by the second equation of (2) we get (3)

$$\frac{\partial H(u,t)}{\partial t} = \left(1 - e^{-ju}\right)\left(\lambda e^{ju} H_1(u,t) + j\left(\sigma + \alpha\right)\frac{\partial H_0(u,t)}{\partial u} + j\alpha\frac{\partial H_1(u,t)}{\partial u}\right), \quad (3)$$

where $H(u,t) = H_0(u,t) + H_1(u,t)$.

We use the system (2) and equation (3) for diffusion approximation in three stages: 1) obtaining the drift (transfer) coefficient; 2) centering the process and obtaining the diffusion coefficient; 3)diffusion approximation.

4. Obtaining the Drift (Transfer) Coefficient

In the system (2) and equation (3), we make the substitutions $\sigma = \varepsilon$, $\alpha = q\varepsilon$, $u = \varepsilon w$, $\tau = \varepsilon t$, $H_k(u,t) = F_k(w,\varepsilon,\tau)$, $k = 0,1$, where ε is infinitesimal value, so

$$\begin{cases} \varepsilon\dfrac{\partial F_0(w,\varepsilon,\tau)}{\partial \tau} = -\lambda F_0(w,\varepsilon,\tau) + \mu F_1(w,\varepsilon,\tau) + j\left(1 + q - qe^{-jw\varepsilon}\right)\dfrac{\partial F_0(w,\varepsilon,\tau)}{\partial w}, \\ \varepsilon\dfrac{\partial F_1(w,\varepsilon,\tau)}{\partial \tau} = \lambda F_0(w,\varepsilon,\tau) - \mu F_1(w,\varepsilon,\tau) - \lambda\left(1 - e^{jw\varepsilon}\right) F_1(w,\varepsilon,\tau) \\ -je^{-jw\varepsilon}\dfrac{\partial F_0(w,\varepsilon,\tau)}{\partial w} + jq\left(1 - e^{-jw\varepsilon}\right)\dfrac{\partial F_1(w,\varepsilon,\tau)}{\partial w}, \\ \varepsilon\dfrac{\partial F(w,\varepsilon,\tau)}{\partial \tau} = \left(e^{jw\varepsilon} - 1\right)\left(\lambda F_1(w,\varepsilon,\tau) + j\left(1 + q\right)e^{-jw\varepsilon}\dfrac{\partial F_0(w,\varepsilon,\tau)}{\partial w}\right. \\ \left. +jqe^{-jw\varepsilon}\dfrac{\partial F_1(w,\varepsilon,\tau)}{\partial w}\right). \end{cases} \quad (4)$$

Transform the equations of (4) under $\varepsilon \to 0$ with $F_k(w,\tau) = \lim\limits_{\varepsilon\to 0} F_k(w,\varepsilon,\tau)$, $k = 0,1$, and find their solution $F_k(w,\tau)$, $k = 0,1$, in the form

$$F_k(w,\tau) = R_k \exp\left\{jwx(\tau)\right\}, \quad k = 0,1, \quad (5)$$

where $R_k = H_k(0)$, $k = 0,1$, $x(\tau)$ - unknown function of time τ.

Substituting (5) in (4) we get the following

$$\begin{cases} R_0 = R_0(x(\tau)) = \dfrac{\mu}{\lambda + \mu + x(\tau)}, \\ R_1 = R_1(x(\tau)) = \dfrac{\lambda + x(\tau)}{\lambda + \mu + x(\tau)}, \\ x'(\tau) = a(x(\tau)) = \lambda - qx(\tau) - \left(\lambda + x(\tau)\right) R_0(x(\tau)). \end{cases} \quad (6)$$

5. Centering the Process and Obtaining the Diffusion Coefficient

In (2) and (3) we let

$$H_k(u,t) = \exp\left\{\frac{ju}{\sigma}x(\sigma t)\right\} H_k^{(2)}(u,t), \quad k=0,1, \tag{7}$$

and then we make the substitutions $\sigma=\varepsilon^2$, $\alpha=q\varepsilon^2$, $u=\varepsilon w$, $\tau=\varepsilon^2 t$, $H_k^{(2)}(u,t) = F_k^{(2)}(w,\varepsilon,\tau)$, $k=0,1$, to obtain the system below after some transformations

$$\left\{\begin{array}{l} jw\varepsilon a(x(\tau))F_0^{(2)}(w,\varepsilon,\tau)=\mu F_1^{(2)}(w,\varepsilon,\tau)-[\lambda+x(\tau)+qjw\varepsilon x(\tau)]F_0^{(2)}(w,\varepsilon,\tau) \\ +j\varepsilon\dfrac{\partial F_0^{(2)}(w,\varepsilon,\tau)}{\partial w}+O(\varepsilon^2), \\ jw\varepsilon a(x(\tau))F_1^{(2)}(w,\varepsilon,\tau) = \left[\lambda+(1-jw\varepsilon)\,x(\tau)\right]F_0^{(2)}(w,\varepsilon,\tau) \\ +\left[\lambda jw\varepsilon-\mu-qjw\varepsilon x(\tau)\right]F_1^{(2)}(w,\varepsilon,\tau)-j\varepsilon\dfrac{\partial F_0^{(2)}(w,\varepsilon,\tau)}{\partial w}+O(\varepsilon^2), \\ \varepsilon^2\dfrac{\partial F^{(2)}(w,\varepsilon,\tau)}{\partial\tau}+jw\varepsilon a(x(\tau))F^{(2)}(w,\varepsilon,\tau) = \left(jw\varepsilon+\dfrac{(jw\varepsilon)^2}{2}\right) \\ \Big\{-(1+q)\,(1-jw\varepsilon)\,x(\tau)F_0^{(2)}(w,\varepsilon,\tau)+\left[\lambda-q\,(1-jw\varepsilon)\,x(\tau)\right]F_1^{(2)}(w,\varepsilon,\tau) \\ +j\varepsilon\,(1+q)\,\dfrac{\partial F_0^{(2)}(w,\varepsilon,\tau)}{\partial w}+jq\varepsilon\dfrac{\partial F_1^{(2)}(w,\varepsilon,\tau)}{\partial w}\Big\}+O(\varepsilon^3). \end{array}\right. \tag{8}$$

The solution of equations system (8) has the following form

$$\left\{\begin{array}{l} F_k^{(2)}(w,\varepsilon,\tau)=\Phi(w,\tau)\,(R_k+jw\varepsilon f_k)+O(\varepsilon^2), \quad k=0,1, \\ R_0+R_1=1, \end{array}\right. \tag{9}$$

where $R_k=R_k(x(\tau))$, $k=0,1$, are defined above, f_0, f_1, $(f_0+f_1=f)$, are constants, and $\Phi(w,\tau)$ is determined function.

Using (6) and (9) in (8) after transformations we can get

$$\left\{\begin{array}{l} -\left[\lambda+x(\tau)\right]f_0+\mu f_1=\left[a(x(\tau))+qx(\tau)\right]R_0-R_0\dfrac{\partial\Phi(w,\tau)/\partial w}{w\Phi(w,\tau)}, \\ \left[\lambda+x(\tau)\right]f_0-\mu f_1=\left[a(x(\tau))-\lambda+qx(\tau)\right]R_1+x(\tau)R_0+R_0\dfrac{\partial\Phi(w,\tau)/\partial w}{w\Phi(w,\tau)}, \\ \dfrac{\partial\Phi(w,\tau)}{\partial\tau}=(jw)^2\Phi(w,\tau)\left\{-a(x(\tau))f+qx(\tau)R_1+\left[\lambda-qx(\tau)\right]f_1\right. \\ \left.-(1+q)\,x(\tau)f_0+(1+q)\,x(\tau)R_0\right\}-w(1+q)R_0\dfrac{\partial\Phi(w,\tau)}{\partial w} \\ -wqR_1\dfrac{\partial\Phi(w,\tau)}{\partial w}+\dfrac{(jw)^2}{2}a(x(\tau))\Phi(w,\tau). \end{array}\right. \tag{10}$$

The solution of system (10) has the form

$$f_k = CR_k + g_k - \varphi_k \frac{\partial \Phi(w,\tau)/\partial w}{w\Phi(w,\tau)}, \quad k = 0, 1, \tag{11}$$

and after substitution (11) in the first and the second equations of the (10) we obtain the equations systems (12), (13) for the φ_k and g_k, $k = 0, 1$, respectively

$$\begin{cases} -\left[\lambda + x(\tau)\right] g_0 + \mu g_1 = \left[a(x(\tau)) + qx(\tau)\right] R_0, \\ \left[\lambda + x(\tau)\right] g_0 - \mu g_1 = \left[a(x(\tau)) - \lambda + qx(\tau)\right] R_1 + x(\tau) R_0, \end{cases} \tag{12}$$

$$\begin{cases} \left[\lambda + x(\tau)\right] \varphi_0 - \mu \varphi_1 = -R_0, \\ -\left[\lambda + x(\tau)\right] \varphi_0 + \mu \varphi_1 = R_0. \end{cases} \tag{13}$$

Equations (6) and additional condition $g_0 + g_1 = 0$ for the (12) lead us to (14)

$$\begin{cases} \varphi_k = \varphi_k(x(\tau)) = \dfrac{\partial R_k(x(\tau))}{\partial x(\tau)}, \quad \varphi_0 + \varphi_1 = 0, \quad k = 0, 1, \\ g_0 = g_0(x(\tau)) = -\dfrac{a(x(\tau)) + qx(\tau)}{\lambda + \mu + x(\tau)} R_0(x(\tau)), \quad g_1 = -g_0. \end{cases} \tag{14}$$

The third equation of the (10) with (6), (11), (14) can be rewritten as

$$\frac{\partial \Phi(w,\tau)}{\partial \tau} = a'(x(\tau)) w \frac{\partial \Phi(w,\tau)}{\partial w} + b(x(\tau)) \frac{(jw)^2}{2} \Phi(w,\tau), \tag{15}$$

where

$$b(x(\tau)) = a(x(\tau)) + 2\Big(qx(\tau) R_1(x(\tau)) + (1+q) x(\tau) R_0(x(\tau)) + \left[\lambda + x(\tau)\right] g_1\Big). \tag{16}$$

6. Diffusion Approximation

Using (15) and (1), (7), (8) we can get the Fokker-Plank equation for the probability density of an diffusion process $y(\tau)$ with drift (transfer) coefficient $a'(x(\tau))y(\tau)$ and diffusion coefficient $b(x(\tau))$

$$\frac{\partial P(y(\tau),\tau)}{\partial \tau} = -a'(x(\tau)) \frac{\partial \left\{y(\tau) P(y(\tau),\tau)\right\}}{\partial y(\tau)} + \frac{b(x(\tau))}{2} \frac{\partial^2 P(y(\tau),\tau)}{\partial y^2(\tau)}, \tag{17}$$

and the process $y(\tau)$ is the solution of the stochastic differential equation (18)

$$dy(\tau) = a'(x(\tau)) y(\tau) d\tau + \sqrt{b(x(\tau))} d\omega(\tau), \tag{18}$$

where $\omega(\tau)$ is the Wiener process.

Introduce diffusion process $z(\tau) = x(\tau)+\varepsilon y(\tau)$ and write the stochastic differential equation (19) for $z(\tau)$

$$dz(\tau) = a(z(\tau))d\tau + \varepsilon\sqrt{b(z(\tau))}d\omega(\tau). \tag{19}$$

Denote the probability density of the $z(\tau)$ as $\Pi(z(\tau),\tau) = \dfrac{\partial P\{z(\tau) < z\}}{\partial z}$ and he Fokker-Plank equation for it can be written as follows

$$\frac{\partial\Pi(z(\tau),\tau)}{\partial\tau} = -\frac{\partial\{a(z(\tau))\Pi(z(\tau),\tau)\}}{\partial z(\tau)} + \frac{\varepsilon^2}{2}\frac{\partial^2\{b(z(\tau))\Pi(z(\tau),\tau)\}}{\partial z^2(\tau)}. \tag{20}$$

The solution of the equation (20) for stationary probability distribution of the process $z(\tau)$ has the form

$$\Pi(z) = \frac{C}{b(z)}\exp\left\{\frac{2}{\sigma}\int_0^z\frac{a(x)}{b(x)}dx\right\}, \quad C - constant. \tag{21}$$

Finally, based on the (21) in (22) we get diffusion approximation $\widetilde{P}(i)$ for the stationary distribution $P(i)$ of the number of calls in the orbit

$$\widetilde{P}(i) = \frac{\Pi(i\sigma)}{\sum\limits_{k=0}^{\infty}\Pi(k\sigma)}. \tag{22}$$

7. Numerical Results

Preliminary calculations suggest that theoretical results are consistent with simulation ones. To compare the pre-limit probability distribution of the number of calls in the orbit of considered queueing system $P(i)$ calculated via matrix method and its approximation $PD(i)$ constructed by using the asymptotic diffusion analysis method for different values of the system parameters we use Kolmogorov distance Δ between respective distribution functions: $\Delta = \max\limits_{n\geq 0}\left|\sum\limits_{i=0}^{n}[P(i) - PD(i)]\right|$.

The comparison of the distributions is shown in Figures 1, 2.

8. Conclusion

In the present paper, retrial queueing system of M/M/1 type with impatient customers in the orbit is considered. In the course of the study, the asymptotic diffusion analysis method was used and the diffusion approximation of the stationary probability distribution of the calls number in the orbit was obtained. As a asymptotic

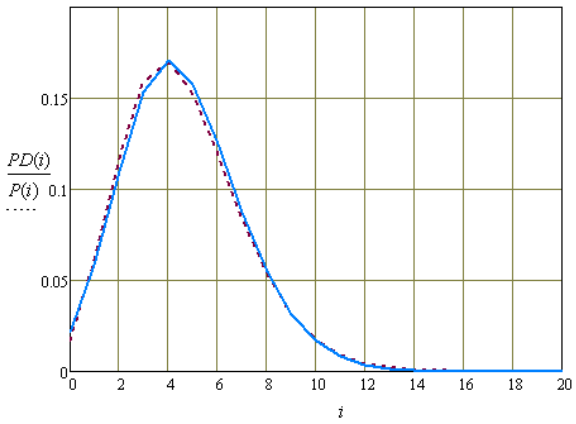


Fig. 1. Comparison of the asymptotic (solid line) and the pre-limit (dashed line) distributions for $\sigma = 0.01$, $\lambda = 0.4$, $\mu = 1$, $q = 2$, $H = 1$, $\Delta = 0.012$.

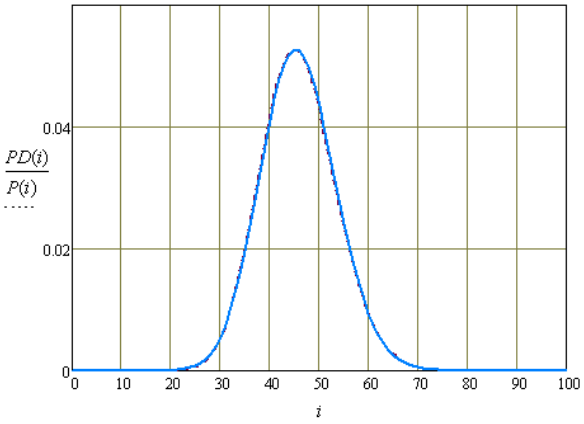


Fig. 2. Comparison of the asymptotic (solid line) and the pre-limit (dashed line) distributions for $\sigma = 0.001$, $\lambda = 0.4$, $\mu = 1$, $q = 2$, $H = 1$, $\Delta = 0.0027$.

condition it was taken condition of a long delay of calls in orbit and a long time patience of calls in the orbit. In further studies it is planned to obtain numerical results about diffusion approximation and compare them with the asymptotic analysis method results from [5]. Based on the works of the other authors from references, for example, it can be assumed that the asymptotic diffusion analysis method is more accurate than asymptotic analysis method under the same asymptotic condition.

REFERENCES

1. Nazarov A. A., Paul S. V., Lizyura O. D. Asymptotic Diffusion Analysis of Retrial Queue M/M/1/1 with Outgoing Calls // In: Vishnevskiy V. M. and Samouylov K. E. Distributed Computer and Communication Networks (DCCN-2019). 2019. P. 148–155. (in Russian)
2. Nazarov A., Phung-Duc T., Paul S., Lizyura O. Asymptotic-Diffusion Analysis of Retrial Queue with Two-Way Communication and Renewal Input // Proceedings of The 5th International Conference on Stochastic Methods (ICSM-5). 2020. P. 339–345.
3. Nazarov A. A., Phung-Duc T., Izmailova Ya. Ye. Asymptotic-Diffusion Analysis of Multiserver Retrial Queueing System with Priority Customers // Proceedings of the XIX International Conference named after A. F. Terpugov. Information Technologies and Mathematical Modelling (ITMM-2020). 2021. P. 88–98.
4. Moiseev A., Nazarov A., Paul S. Asymptotic diffusion analysis of multi-server retrial queue with hyperexponential service // Mathematics. 2020. V. 8, no. 4. P. 531.
5. Nazarov A. A., Fedorova E. A. Asymptotic Analysis of Retrial Queue M/M/1 with Impatient Calls under the Long Patience Time Condition // In: Vishnevskiy V. M. and Samouylov K. E. Distributed Computer and Communication Networks (DCCN-2016). 2016. P. 342–348. (in Russian)

UDC: 531.3

Construction of differential equations of a nonholonomic mechanical system and perspectives of motion control using artificial intelligence methods

A.V. Borisov[1], R.G. Mukharlyamov[2], I.E. Kaspirovich[2]

[1]The Branch of National Research University "Moscow Power Engineering Institute", Energetichesky proezd, 1, Smolensk, Russia

[2]Peoples' Friendship University of Russia, Miklukho-Maklaya, 6, Moscow, Russia

borisowandrej@yandex.ru, robgar@mail.ru

Abstract

First, a mechanical model of a mechanism with two links of variable length in space with holonomic constraints is considered. Its mathematical model is obtained, in the form of Lagrange equations of the second kind. Then we consider a model similar in structure - the number and design of links with a nonholonomic constraint in the form of a skier-snowboarder. The mathematical model of a system of rigid bodies with a nonholonomic constraint is based on the Routh differential equations for a nonholonomic system in generalized coordinates with Lagrange multipliers. A method is developed for constructing differential equations for a system containing nonholonomic constraints using Lagrange equations of the second kind for a model of a similar structure with holonomic constraints. As an example, the model is applied to the description of a skier-snowboarder with two variable-length movable links on one ski. Usually, the control of such systems is implemented on the basis of the method of stabilizing links, however, this article declares methods for controlling mechanical systems based on artificial intelligence systems and outlines approaches to their use in models with nonholonomic links.

Keywords: dynamics, system, nonholonomic constraint, Lagrange equations, Routh equations, snowboarder, variable length link, artificial intelligence systems

1. Introduction

Methods for setting equations of motion for holonomic mechanical systems of anthropomorphic type were considered in [1,2]. In this paper, we propose a method for constructing the equations of motion with the nonholonomic constraints described

This work was supported by the RFBR, project 19-08-00261 A

by the Routhian equations [3]. In [4,5], the methods of dynamics modelling were applied to describe the motions of nonholonomic systems. In this paper, on the basis of a comparative analysis of the equations of motion of mechanical systems of the relative structure, with holonomic and nonholonomic constraints, regularities for the transition from Lagrange equations of the second kind to the Routhian equations are obtained. To control the motion, the problems of modelling the movement of a skier-snowboarder with a nonholonomic constraint at the point of contact of the snowboard with the snow were considered using the method of constraint stabilization [6]. In this article, we will state the application of artificial intelligence methods in controlling the movement of mechanical systems. The use of artificial intelligence systems in relation to the motion control of various robots is considered in [7-10]. In [7-8], the application of fuzzy inference systems to robot motion control is investigated. Papers [9-12] describe the successful application of neural networks in mechanistic systems.

2. Description the mechanical model by Lagrange differential equations of the second kind

A model of a system with two links of variable length in space is shown in (Fig. 1). The position of the system will be determined relative to the rectangular Cartesian coordinate system $Oxyz$.

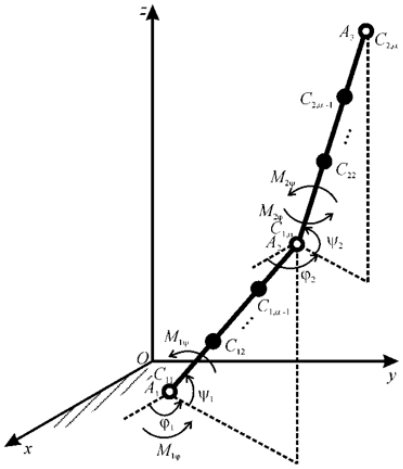


Fig. 1. Model of two moving links of variable length in space, with the pole A_1 moving in the xOy plane

The coordinates of the A_1 pole, the angles, and the lengths of the links are functions of time: $x_{A_1} = x_{A_1}(t) = x$, $y_{A_1} = y_{A_1}(t) = y$, $\varphi_i = \varphi_i(t)$, $\psi_i = \psi_i(t)$, $l_i = l_i(t)$, $i = 1, 2$. Thus, the model of the system has eight parameters that uniquely determine its position. The rotation of the links relative to each other occurs in a joint located at point A_2, in which there is a combination of two cylindrical joints, the axes of rotation of which are perpendicular to each other. In this joint, the following moments are applied: $M_{2\psi}$ ensures the maintenance of the link in the vertical plane, $M_{2\varphi}$ corresponds to the rotations of the link. The longitudinal forces acting along the rods and providing a change in their lengths are denoted as F_i $(i = 1, 2)$. On the rods there are point masses m_{i1}, m_{i2} at points C_{i1}, C_{i2}. Their position on the rod is determined by the constant multipliers $n_{i1}, n_{i2}, \ldots, n_{i\alpha_1}$, respectively $(0 \leq n_{i\beta} \leq 1, \beta = 1, 2, \ldots, \alpha_i, i = 1, 2)$. The first index of the multiplier indicates the number of the link, the second – the number of the point mass on the link. Pole A_1 with coordinates x_{A_1}, y_{A_1} is moving in the xOy plane. The position of the link $A_1A_2 = l_1(t)$ is determined by two angles: φ_1 – the angle between the axis Ox and the projection of the link A_1A_2 on the plane xOy, counted from the axis Ox counterclockwise; ψ_1 – the angle between the link A_1A_2 and its projection on the plane xOy, counted from the projection of the link A_1A_2 on the plane xOy counterclockwise and the change in its length l_1, the coordinates of the pole $A_1(x_{A_1}, y_{A_1})$. The position of the link $A_2A_3 = l_2(t)$ is determined similarly to the first link. For this model, a system of Lagrange differential equations of motion of the second kind is written. The first of the equations of this system is presented below (1).

$$\left(\sum_{\beta=1}^{\alpha_1} m_{1\beta} + \sum_{\beta=1}^{\alpha_2} m_{2\beta}\right)\ddot{x} - l_1\left(\sum_{\beta=1}^{\alpha_1} m_{1\beta}n_{1\beta} + \sum_{\beta=1}^{\alpha_2} m_{2\beta}\right)[\sin\varphi_1\cos\psi_1\ddot{\varphi}_1 +$$
$$+ \cos\varphi_1\sin\psi_1\ddot{\psi}_1 + \cos\varphi_1\cos\psi_1\dot{\varphi}_1^2 + \cos\varphi_1\cos\psi_1\dot{\psi}_1^2 - 2\sin\varphi_1\sin\psi_1\dot{\varphi}_1\dot{\psi}_1] +$$
$$+ \left(\sum_{\beta=1}^{\alpha_1} m_{1\beta}n_{1\beta} + \sum_{\beta=1}^{\alpha_2} m_{2\beta}\right)[2\cos\varphi_1\sin\psi_1\dot{l}_1\dot{\psi}_1 + 2\sin\varphi_1\cos\psi_1\dot{l}_1\dot{\varphi}_1 +$$
$$+ \cos\varphi_1\cos\psi_1\ddot{l}_1] - l_2\left(\sum_{\beta=1}^{\alpha_2} m_{2\beta}n_{2\beta}\right)[\sin\varphi_2\cos\psi_2\ddot{\varphi}_2 +$$
$$+ \cos\varphi_2\sin\psi_2\ddot{\psi}_2 + \cos\varphi_2\cos\psi_2\dot{\varphi}_2^2 + \cos\varphi_2\cos\psi_2\dot{\psi}_2^2 - 2\sin\varphi_2\sin\psi_2\dot{\varphi}_2\dot{\psi}_2] +$$
$$+ \sum_{\beta=1}^{\alpha_2} m_{2\beta}n_{2\beta}[2\cos\varphi_2\sin\psi_2\dot{l}_2\dot{\psi}_2 + 2\sin\varphi_2\cos\psi_2\dot{l}_2\dot{\varphi}_2 +$$
$$+ \cos\varphi_2\cos\psi_2\ddot{l}_2] = Q_x, \quad (1)$$

where: $\beta = 1, 2, \ldots, \alpha_i$ is the number of the mass concentrated on the i-th link, Q_x is the generalized force vector. The remaining differential equations are similar in structure to equation (1).

3. Description of the model of a snowboarder with a nonholonomic constraints by the Routhian equations

A model of a skier-snowboarder with a nonholonomic constraints in the contact zone of the ski with snow is presented (Fig. 2). A snowboarder, modelled by two mobile links of variable length, is pivotally fixed on an absolutely solid inertial ski (Fig. 2). The coordinates of the hinge attachment of the link A_1A_2 to the ski x_{A_1}, y_{A_1}. The angle $\varphi_0 = \varphi_0(t)$ sets the position of the ski relative to the Ox axis. All other coordinates are similar to the model of the mechanism moving along the horizontal plane (Fig. 1). In the plane of movement of the snowboarder, the component of the acceleration of free fall $g_1 = g\sin\alpha$.

The rotation of the links in the model of the snowboarder is presented at the points $A_i (i = 1, 2)$, where the joints are located, the design of which is similar to those described in the model shown in Figure 1. The moments and longitudinal forces are also similar. In the case of point A_1, an additional link is the ski.

The constraints on the contact surface of the ski and snow is nonholonomic [7]. The constraint equation can be written as follows:

$$\dot{x}\tan\varphi_0 - \dot{y} = 0. \quad (2)$$

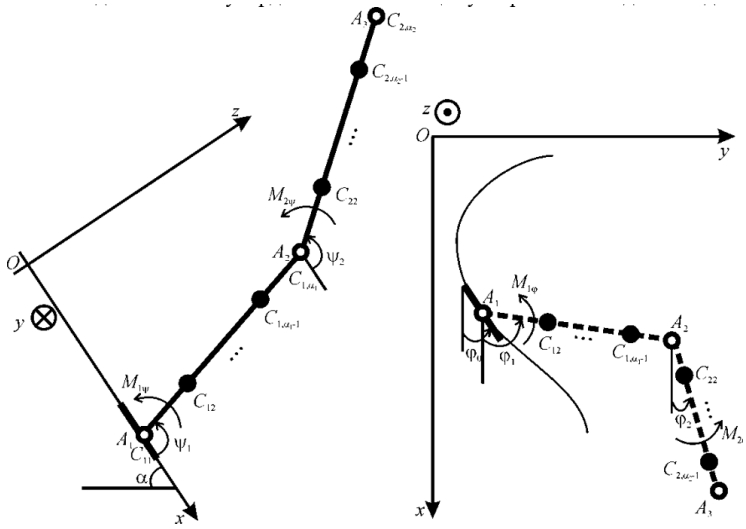


Fig. 2. Model of a snowboarder consisting of two mobile links on skis, sliding on an inclined plane in projections on the coordinate planes xOz and xOy

where $\varphi_0(t)$ is the variable angle that the ski forms with the Ox axis.

The trajectory of the snowboarder (Fig. 2) is a program motion and represents a given curved line, in our case a sinusoid.

$$y = A\sin(kx),\ \dot{y} = Ak\dot{x}\cos(kx), \tag{3}$$

Equation (2) restricts the coordinates variations:

$$\tan\varphi_0\delta x - \delta y = 0. \tag{4}$$

To describe the motion of a nonholonomic system, we use the Routhian equations with Lagrange multipliers. In this case, a single holonomic constraints (2) is imposed on the system. Following the work [3], we obtain the Routhian differential equations for the mechanical system under consideration. We will give only the first differential equation, since the rest are similar in structure.

$$\left(m_0+\sum_{\beta=1}^{\alpha_1}m_{1\beta}+\sum_{\beta=1}^{\alpha_2}m_{2\beta}\right)\sec^2\varphi_0[\ddot{x}+\tan\varphi_0\dot{x}\dot{\varphi}_0]+$$
$$+l_1\left(m_0+\sum_{\beta=1}^{\alpha_1}m_{1\beta}n_{1\beta}+\sum_{\beta=1}^{\alpha_2}m_{2\beta}\right)[(\cos\varphi_1\tan\varphi_0-\sin\varphi_1)\cos\psi_1\ddot{\varphi}_1+$$
$$+(\sin\varphi_1\tan\varphi_0+\cos\varphi_1)\sin\psi_1\ddot{\psi}_1-(\sin\varphi_1\tan\varphi_0+\cos\varphi_1)\sin\psi_1\ddot{\psi}_1-$$
$$-(\sin\varphi_1\tan\varphi_0+\cos\varphi_1)\cos\psi_1\dot{\varphi_1}^2-(\sin\varphi_1\tan\varphi_0+\cos\varphi_1)\cos\psi_1\dot{\psi_1}^2-$$
$$-2(\cos\varphi_1\tan\varphi_0-\sin\varphi_1)\sin\psi_1\dot{\psi}_1\dot{\varphi}_1-2(\sin\varphi_1\tan\varphi_0+\cos\varphi_1)\sin\psi_1\dot{\psi}_1\dot{l_1}-$$
$$-2(\cos\varphi_1\tan\varphi_0+\sin\varphi_1)\cos\psi_1\dot{l_1}\dot{\varphi}_1]+\left(\sum_{\beta=1}^{\alpha_1}m_{1\beta}n_{1\beta}+\sum_{\beta=1}^{\alpha_2}m_{2\beta}\right)\cdot(\sin\varphi_1\tan\varphi_0+$$
$$+\cos\varphi_1)\cos\psi_1\ddot{l_1}+l_2\left(\sum_{\beta=1}^{\alpha_2}m_{2\beta}n_{2\beta}\right)[(\cos\varphi_2\tan\varphi_0-\sin\varphi_2)\cos\psi_2\ddot{\varphi}_2+$$
$$+(\sin\varphi_2\tan\varphi_0+\cos\varphi_2)\sin\psi_2\ddot{\psi}_2-(\sin\varphi_2\tan\varphi_0+\cos\varphi_2)\sin\psi_2\ddot{\psi}_2-$$
$$-(\sin\varphi_2\tan\varphi_0+\cos\varphi_2)\cos\psi_2\dot{\varphi_2}^2-(\sin\varphi_2\tan\varphi_0+\cos\varphi_2)\cos\psi_2\dot{\psi_2}^2-$$
$$-2(\cos\varphi_2\tan\varphi_0-\sin\varphi_2)\sin\psi_2\dot{\psi}_2\dot{\varphi}_2-2(\sin\varphi_2\tan\varphi_0+\cos\varphi_2)\sin\psi_2\dot{\psi}_2\dot{l_2}-$$
$$-2(\cos\varphi_2\tan\varphi_0+\sin\varphi_2)\cos\psi_2\dot{l_2}\dot{\varphi}_2]+\sum_{\beta=1}^{\alpha_2}m_{2\beta}n_{2\beta}\cdot(\sin\varphi_2\tan\varphi_0+$$
$$+\cos\varphi_2)\cos\psi_2\ddot{l_2}=\left(m_0+\sum_{\beta=1}^{\alpha_1}m_{1\beta}+\sum_{\beta=1}^{\alpha_2}m_{2\beta}\right)g_1+R_x+R_y\tan\varphi_0. \quad (5)$$

Let us consider the differential equation (5). It consists some extra terms like m_0, $\sec\varphi_0$, $\tan\varphi_0$ that are connected with the constraint equation (2). This establishes the differences between the Lagrange differential equations for the holonomic system (1) and the Routhian equations for the nonholonomic system (5), which are the basis for creating a method of transition from the Lagrange equations of the second kind to the Routhian differential equations.

4. Transition from Lagrange equations of the second kind to Routhian equations

According to equation (5), it can be seen in comparison with equation (1) that each term of the equation contains a term associated with the product $\sin\varphi_2 \tan\varphi_0$ or $\cos\varphi_2 \tan\varphi_0$, $i = 1, 2$ and if in the original equation the considered term was a sine, then a term containing a cosine is added and vice versa. This term in the equation is related to the fact that the skier-snowboarder slides on the skis and takes into account the turn of the ski. In addition, the mass of the ski m_0 and the multiplier $\sec^2\varphi_0$, which takes into account the turn of the ski, are added to the coefficient to the generalized acceleration. In addition, the equation contains a term that takes into account the motion of the ski: $\left(m_0 + \sum\limits_{\beta=1}^{\alpha_1} m_{1\beta} + \sum\limits_{\beta=1}^{\alpha_2} m_{2\beta}\right) \sec^2\varphi_0 \tan\varphi_0 \dot{x}\dot{\varphi}_0$. Otherwise, the equation retains its structure.

Next, we describe the differences in those differential equations that were not presented in the text of the article.

The second equation from the system of Lagrange equations of the second kind, due to the nonholonomic constraint, is excluded from the system of Routhian differential equations. This reduces the dimension of the system of differential equations.

In the Routhan differential equations from the third to the eighth, the changes are insignificant, in comparison with the Lagrange equations of the second kind for the holonomic system. First, in the Lagrange equations, in the term containing, in the Routhian equations, it is replaced by $\sec^2\varphi_0 \tan\varphi_0 \dot{x}\dot{\varphi}_0$. Secondly, in the term with generalized acceleration in the Routhian equations, the term $\left(\sum\limits_{\beta=1}^{\alpha_1} m_{1\beta} n_{1\beta} + \sum\limits_{\beta=1}^{\alpha_2} m_{2\beta}\right) \tan\varphi_0 \cos\varphi_1 \cos\psi_1 \ddot{x}$ is added. All other elements of the Lagrange differential equations of the second kind and the Routhian equations coincide.

As a result, the method of constructing the Routhian equations in relation to the dynamics of a skier-snowboarder can be described as follows: in the first equation, each factor in each term $\cos\varphi_i$ is replaced by $(\sin\varphi_i \tan\varphi_0 + \cos\varphi_i$, and in each term with a sine: $\sin\varphi_i$ is replaced by $(\cos\varphi_i \tan\varphi_0 - \cos\varphi_i$. And the term $\left(\sum\limits_{\beta=1}^{\alpha_1} m_{1\beta} + \sum\limits_{\beta=1}^{\alpha_2} m_{2\beta}\right) \ddot{x}$ is replaced by $\left(m_0 + \sum\limits_{\beta=1}^{\alpha_1} m_{1\beta} + \sum\limits_{\beta=1}^{\alpha_2} m_{2\beta}\right) \sec^2\varphi_0 [\ddot{x} + \tan\varphi_0 \dot{x}\dot{\varphi}_0]$. The second equation is omitted, and in the other equations. Thus, formulas for the transition from the Lagrange differential equations of the second kind to the Routhian equations are obtained.

5. Possibilities of using artificial intelligence in motion control of mechanical systems with nonholonomic constraints

Since the motion of biological objects is modelled, the motion control of such objects can be based on artificial intelligence systems. In relation to the problem considered in this article, it can be determined that artificial intelligence is a field of science that deals with the automation of intelligent behaviour of mechanistic systems. It allows you to create a new generation of mechanistic systems that have fundamentally new qualities and high operational parameters, which ensures the relevance of considering the possibility of using them to control complex mechanical systems that have non-holonomic constraints. Currently, there is a tendency to turn software engineering into intelligent engineering, which considers more general problems of knowledge representation and information processing, which is important in solving management problems. Applying artificial intelligence techniques to motion control tasks can help you solve them more efficiently, with less energy and time. Creating a database of real movements can determine the kinematic characteristics of movement based on the results of video recording and create an expert system for using it in the training process of athletes. The use of neural networks will allow you to determine the control moments and forces to control the movement of the robot snowboarder.

6. Conclusion

As a result, a method for constructing differential equations of a nonholonomic mechanical system based on Lagrange equations of the second kind for a similar system is proposed. It is effective, since the complexity of obtaining differential equations of motion in the form of Lagrange equations of the second kind is less than the equations of motion of a nonholonomic system, and in addition, high-speed methods for writing equations have been developed for them [1-3]. Therefore, the proposed method for constructing differential equations for a nonholonomic mechanical system can be applied to a model with any number of links. It significantly reduces the time to obtain the differential equations of a multi-link nonholonomic system. The prospects of using artificial intelligence systems to control the movement of anthropomorphic devices are analysed, its main directions and possible approaches are shown. However, the development of control systems based on artificial intelligence methods is a time-consuming process that requires significant costs.

REFERENCES

1. Borisov A.V., Rosenblat G. M. Matrix method for composing differential equations of exoskeleton motion and control / / Applied Mathematics and Mechanics. - 2017. - Vol. 81. - No. 5. - p. 511-522.

2. Borisov A.V., Rosenblat G. M. Modeling of the dynamics of an exoskeleton with controlled moments in the joints and variable link length using the recurrent method of composing differential equations of motion // Izvestiya RAS. Theory and control systems. 2018. No. 2. pp. 148-174.
3. Buchholz N. N. Basic course of theoretical mechanics. In 2 ch. Ch. 2. - Moscow: Nauka, 1966. - 332 p.
4. Kaspirovich I. E., Mukharlyamov R. G. On methods for constructing dynamic equations taking into account the constraint stabilization // Izv. RAS. MTT. 2019. No. 3. pp. 124-135.
5. Baumgarte J. Stabilization of constraints and integrals of motion in dynamical systems // Computer Methods in Applied Mechanics and Engineering, Vol 1, Issue 1, 1-16 pp.
6. Borisov A.V., Kaspirovich I. E., Mukharlyamov R. G. Control of the dynamics of a composite structure with variable-length links // Proceedings of the Russian Academy of Sciences. Solid State Mechanics 2021, No. 2, pp. 72-87.
7. Yaqi Zhao, Jingcheng Wang, Langwen Zhang, Tao Hu Fuzzy-PID Based Induction Motor Control and Its Application to the TBM Cutter Head Systems. // Intelligent Robotics and Applications 8th International Conference ICIRA 2015, Portsmouth, UK, August 24-27, 2015, Proceedings, Part III R. 511-522.
8. Wang, S. Y., et al.: An adaptive supervisory sliding fuzzy cerebellar model articulation controller for sensorless vector-controlled induction motor drive systems. Sensors (Switzerland) 15(4), 7323–7348 (2015).
9. Sharapaev L. A. Touching an object with a robot manipulator controlled by a neural network // Bulletin of the Russian State University named after I. Kant. 2007. Issue 10. Physical and mathematical sciences. pp. 56-60.
10. Chaoliang Zhong, Shirong Liu, Qiang Lu and Botao Zhang. Continuous learning route map for robot navigation using a growing-on-demand selforganizing neural network. - International Journal of Advanced Robotic Systems, November-December 2017. – Pp. 1–13.
11. Kozhevnikov V. V., Leontiev M. Yu., Prikhodko V. V., Sergeev V. A., Fomin A. N. Neural network technologies for building intelligent robot control systems. Mathematics and Information Technology. UlSU. Electron. zhurnal. 2019, No. 2, pp. 36-53.
12. Voynov I. V., Kazantsev A.M., Morozov B. A., Nosikov M. V. Control system for a robot manipulator using neural network algorithms for limiting the working area of the grip // Bulletin of SUSU. Series "Computer technologies, control, radio electronics". - 2017. - Vol. 17, No. 4. - p. 29-36.

UDC: 519.872

Scaling limits of a tandem retrial queue with common orbit and Poisson arrival process

A.A. Nazarov[1], S.V. Paul[1], T. Phung-Duc[2], M.A. Morozova[1]

[1]National Research Tomsk State University, 36 Lenina ave., 634050, Tomsk, Russia
[2]University of Tsukuba, 1-1-1 Tennodai, Tsukuba, Ibaraki 305-8573, Japan

nazarov.tsu@gmail.com, paulsv82@mail.ru, tuan@sk.tsukuba.ac.jp,
morozova_mariya_a@mail.ru

Abstract

In this paper, we consider a tandem queueing system with one orbit, Poisson arrival process of incoming calls and two sequentially connected servers by a method of asymptotic analysis under the steady-state regime. Under the condition that the average delay time of calls in the orbit is extremely large, we obtain the limiting probability distribution of the number of calls there. Then we evaluate the applicability of asymptotic results by the simulation.

Keywords: Tandem RQ-system with two sequentially connected servers, asymptotic analysis method, Gaussian approximation.

1. Introduction

In queuing theory exists a special class of systems, in which the following situation is characterized: if a call finds the server busy, it goes into the orbit, from there, after some random time, it tries to get onto the server again. Such models with orbits are called retrial queuing systems or RQ-systems [1, 2, 3].

On the other hand, tandem queuing systems represent a connection between one node queue and queuing networks: such systems can be considered as queuing networks with a linear topology [4, 5].

Also, tandem RQ-systems are used to simulate the processing process, in which incoming requests are serviced sequentially at several stages. The need for sequential service arises in processing requests in call-centers [6, 7], in transmitting multimedia information [8], in controlling the data flow between elements of a multi-agent robotic system [9], etc.

Tandem queuing networks are extensively studied. If the buffer is full, then the request is lost in such systems [10, 11]. In contrast to this, we study tandem systems

with an orbit of infinite capacity. Research in this area has already been carried out by several authors. In [12], a tandem system with two servicing machines and two types of claims is studied. In [13], a model with a correlated flow of arrivals and the operation of the second station is described by a Markov chain.

In this paper, we study a two-phase tandem RQ-system with one orbit by the method of asymptotic analysis [14] under the condition of a large delay of calls in the orbit. The accuracy of the analytical results obtained is evaluated by comparing them with simulation modelling.

2. Mathematical model and problem statement

We consider a retrial queueing system with Poisson arrival process of incoming calls with rate λ and two sequentially connected servers (see Fig. 1). Upon the arrival of a call, if the first server is free, the call occupies it. The call is served for a random time exponentially distributed with parameter μ_1 and then tries to go to the second server. If the second server is free, the call moves to it for a random time exponentially distributed with parameter μ_2. When a call arrives, if the first server is busy, the call instantly goes to the orbit, stays there during random time exponentially distributed with parameter σ and then tries to occupy the first server again. If after serving at the first server if the call finds that the second server is busy, it instantly goes to the same orbit, where, after random exponentially distributed delay with parameter σ, tries to move to the first server for service again.

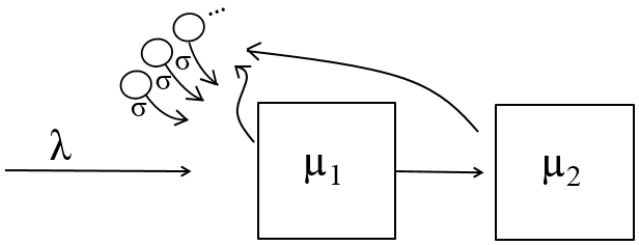


Fig. 1. Tandem RQ-system.

Let us denote:

Process $N_1(t)$ – the state of the first server at time t: 0, if the server is free; 1, if the server is busy;

Process $N_2(t)$ – the state of the second server at time t: 0, if the server is free; 1, if the server is busy;

Process $I(t)$ – the number of calls in the orbit at the time t.

The goal of the study is to obtain the stationary probability distribution of the number of calls in the orbit $I(t)$ and the probability distribution of servers' states in the considered system.

3. Derivation of differential Kolmogorov equations

We define probabilities

$$P_{n_1n_2}(i,t) = P\{N_1(t) = n_1, N_2(t) = n_2, I(t) = i\}; n_1 = 0,1; n_2 = 0,1. \quad (1)$$

The three-dimensional process $\{N_1(t), N_2(t), I(t)\}$ is a Markov chain. For probability distribution (1) we can write the system of differential Kolmogorov equations:

$$\begin{aligned}
\frac{\partial P_{00}(i,t)}{\partial t} &= -(\lambda + i\sigma)P_{00}(i,t) + \mu_2 P_{01}(i,t),\\
\frac{\partial P_{10}(i,t)}{\partial t} &= \lambda P_{00}(i,t) + (i+1)\sigma P_{00}(i+1,t) - (\lambda+\mu_1)P_{10}(i,t) +\\
&\quad +\lambda P_{10}(i-1,t) + \mu_2 P_{11}(i,t),\\
\frac{\partial P_{01}(i,t)}{\partial t} &= \mu_1 P_{10}(i,t) - (\lambda + i\sigma + \mu_2)P_{01}(i,t) + \mu_1 P_{11}(i-1,t),\\
\frac{\partial P_{11}(i,t)}{\partial t} &= \lambda P_{01}(i,t) + (i+1)\sigma P_{01}(i+1,t) - (\lambda+\mu_1+\mu_2)P_{11}(i,t) +\\
&\quad +\lambda P_{11}(i-1,t).
\end{aligned} \quad (2)$$

We introduce partial characteristic functions, denoting $j = \sqrt{-1}$

$$H_{n_1n_2}(u,t) = \sum_{i=0}^{\infty} e^{jui} P_{n_1n_2}(i,t). \quad (3)$$

Denote matrices

$$\begin{aligned}
\mathbf{A} &= \begin{bmatrix} -\lambda & \lambda & 0 & 0 \\ 0 & -(\lambda+\mu_1) & \mu_1 & 0 \\ \mu_2 & 0 & -(\lambda+\mu_2) & \lambda \\ 0 & \mu_2 & 0 & -(\lambda+\mu_1+\mu_2) \end{bmatrix},\\
\mathbf{B} &= \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & \lambda & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & \mu_2 & \lambda \end{bmatrix}, \mathbf{I}_0 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}, \mathbf{I}_1 = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix}.
\end{aligned} \quad (4)$$

Let's write the system (3) in the matrix form

$$\frac{\partial \mathbf{H}(u,t)}{\partial t} = \mathbf{H}(u,t)\{\mathbf{A} + e^{ju}\mathbf{B}\} + ju\frac{\partial \mathbf{H}(u,t)}{\partial u}\{\mathbf{I}_0 - e^{-ju}\mathbf{I}_1\}. \quad (5)$$

Multiplying equations of system (5) an identity column vector $\mathbf{e}$, we get scalar equation and add it to the system (5) in order to have

$$\begin{aligned}\frac{\partial \mathbf{H}(u,t)}{\partial t} &= \mathbf{H}(u,t)\{\mathbf{A}+e^{ju}\mathbf{B}\}+ju\frac{\partial \mathbf{H}(u,t)}{\partial u}\{\mathbf{I}_0-e^{-ju}\mathbf{I}_1\},\\ \frac{\partial \mathbf{H}(u,t)}{\partial t}\mathbf{e} &= (e^{ju}-1)\{\mathbf{H}(u,t)\mathbf{B}+j\sigma e^{-ju}\frac{\partial \mathbf{H}(u,t)}{\partial u}\mathbf{I}_1\}\mathbf{e}.\end{aligned} \tag{6}$$

This system of equations is the basis in further research. We will solve it by an asymptotic method under the asymptotic condition $\sigma \to 0$.

4. Research of the tandem RQ-system by the method of asymptotic analysis

We will solve the equation (6) by a method of asymptotic analysis under the asymptotic condition of unlimitedly increasing the average delay of calls in the orbit. Under the the steady-state regime, the system of equation (6) is written as follows.

$$\begin{aligned}\mathbf{H}(u)\{\mathbf{A}+e^{ju}\mathbf{B}\}+ju\mathbf{H}'(u)\{\mathbf{I}_0-e^{-ju}\mathbf{I}_1\}=0,\\ \{\mathbf{H}(u)\mathbf{B}+j\sigma e^{-ju}\mathbf{H}'(u)\mathbf{I}_1\}\mathbf{e}=0.\end{aligned} \tag{7}$$

4.1. The first order asymptotic. Denote $\sigma=\epsilon$ and perform the following substitution in (7)

$$u=\epsilon w, \mathbf{H}(u)=\mathbf{F}(w,\epsilon). \tag{8}$$

We obtain

$$\begin{aligned}\mathbf{F}(w,\epsilon)\{\mathbf{A}+e^{j\epsilon w}\mathbf{B}\}+j\frac{\partial \mathbf{F}(w,\epsilon)}{\partial w}\{\mathbf{I}_0-e^{-j\epsilon w}\mathbf{I}_1\}=0,\\ \{\mathbf{F}(w,\epsilon)\mathbf{B}+je^{-j\epsilon w}\frac{\partial \mathbf{F}(w,\epsilon)}{\partial w}\mathbf{I}_1\}\mathbf{e}=0.\end{aligned} \tag{9}$$

Theorem 1. Under the asymptotic condition $\sigma \to 0$, the following equality is true

$$\lim_{\sigma\to 0} Ee^{jw\sigma i(t)}=e^{jw\kappa_1}, \tag{10}$$

where κ_1 is a solution of the scalar equation

$$\mathbf{r}(\kappa_1)\{\mathbf{B}-\kappa_1\mathbf{I}_1\}\mathbf{e}=0, \tag{11}$$

and vector $\mathbf{r}(\kappa_1)$ satisfies the normality condition

$$\mathbf{r}(\kappa_1)\mathbf{e}=1, \tag{12}$$

and is a solution of matrix equation

$$\mathbf{r}(\kappa_1)\{(\mathbf{A}+\mathbf{B})-\kappa_1(\mathbf{I}_0-\mathbf{I}_1)\}=0. \tag{13}$$

The first order asymptotic only defines the mean asymptotic value κ_1/σ of the number of calls in the orbit in prelimit situation of nonzero values of σ. For more detailed research of the number $I(t)$ of calls in the orbit let's consider the second order asymptotic.

4.2. The second order asymptotic. Substituting the following equation in the system (7)

$$\mathbf{H}(u) = \exp\left(j\frac{u}{\sigma}\kappa_1\right)\mathbf{H}^{(2)}(u), \tag{14}$$

we obtain

$$\begin{gathered}\mathbf{H}^{(2)}(u)\{\mathbf{A} + e^{ju}\mathbf{B} - \kappa_1(\mathbf{I}_0 - e^{-ju}\mathbf{I}_1)\} + j\sigma\frac{d\mathbf{H}^{(2)}(u)}{du}\{\mathbf{I}_0 - e^{-ju}\mathbf{I}_1\} = 0,\\ \mathbf{H}^{(2)}(u)(\mathbf{B} - e^{-ju}\kappa_1\mathbf{I}_1)\mathbf{e} + j\sigma e^{-ju}\frac{d\mathbf{H}^{(2)}(u)}{du}\mathbf{I}_1\mathbf{e} = 0.\end{gathered} \tag{15}$$

Denote $\sigma = \epsilon^2$ and perform the following substitution in (15)

$$u = \epsilon w, \mathbf{H}^{(2)}(u) = \mathbf{F}^{(2)}(w,\epsilon), \tag{16}$$

and obtain the system

$$\begin{gathered}\mathbf{F}^{(2)}(w,\epsilon)\{\mathbf{A} + e^{j\epsilon w}\mathbf{B} - \kappa_1(\mathbf{I}_0 - e^{-j\epsilon w}\mathbf{I}_1)\} + j\epsilon\frac{\partial\mathbf{F}^{(2)}(w,\epsilon)}{\partial w}\{\mathbf{I}_0 - e^{-j\epsilon w}\mathbf{I}_1\} = 0,\\ \mathbf{F}^{(2)}(w,\epsilon)(\mathbf{B} - e^{-j\epsilon w}\kappa_1\mathbf{I}_1)\mathbf{e} + j\epsilon e^{-j\epsilon w}\frac{\partial\mathbf{F}^{(2)}(w,\epsilon)}{\partial w}\mathbf{I}_1\mathbf{e} = 0.\end{gathered} \tag{17}$$

Theorem 2. In the context of Theorem 1 the following equation is true

$$\lim_{\sigma\to\ 0} Ee^{jw\sqrt{\sigma}\left(i(t) - \frac{\kappa_1}{\sigma}\right)} = e^{\frac{(jw)^2}{2}\kappa_2}, \tag{18}$$

where κ_2 is a solution of the scalar equation

$$\mathbf{g}(\kappa_2)(\mathbf{B} - \kappa_1\mathbf{I}_1)\mathbf{e} = \mathbf{r}\mathbf{I}_1(\kappa_2 - \kappa_1)\mathbf{e}, \tag{19}$$

and vector $\mathbf{g}(\kappa_2)$ is a solution of the system

$$\begin{gathered}\mathbf{g}(\kappa_2)\{\mathbf{A} + \mathbf{B} - \kappa_1(\mathbf{I}_0 - \mathbf{I}_1)\} = \mathbf{r}(\kappa_2\mathbf{I}_0 - \kappa_2\mathbf{I}_1 - \mathbf{B} + \kappa_1\mathbf{I}_1),\\ \mathbf{g}(\kappa_2)\mathbf{e} = 0.\end{gathered} \tag{20}$$

The second order asymptotic shows that the asymptotic probability distribution of the number $I(t)$ of calls in the orbit is Gaussian with mean asymptotic κ_1/σ and dispersion as κ_2/σ.

5. Approximation accuracy and its application area

Now we could build a Gaussian approximation

$$P^{(2)}(i) = (L(i+0.5) - L(i-0.5))(1 - L(-0.5))^{-1}, \tag{21}$$

where $L(x)$ is the normal distribution function with parameters κ_1/σ and κ_2/σ.

Approximation accuracy $P^{(2)}(i)$ will be defined by using Kolmogorov range.

The table contains values for this range for various values of σ and ρ (system load):

$$\rho = \frac{\lambda(\mu_1 + \mu_2)}{\mu_1\mu_2}. \tag{22}$$

We consider $\mu_1 = 1$ and $\mu_2 = 2$ for all experiments.

σ	$\rho = 0.5$	$\rho = 0.6$	$\rho = 0.7$	$\rho = 0.8$	$\rho = 0.9$
0.5	0.142	0.125	0.112	0.146	0.198
0.1	0.071	0.049	0.055	0.071	0.097
0.05	0.034	0.039	0.04	0.036	0.074
0.02	0.022	0.024	0.026	0.031	0.049

Table 1. Kolmogorov range.

It can be seen from the table that the accuracy of the approximations increases with decreasing parameters ρ and σ. The Gaussian approximation is applicable for values of $\sigma < 0.02$, where the relative error, in the form of the Kolmogorov distance, does not exceed 0.05.

6. Conclusion

In this paper, we consider the tandem retrial queueing system with Poisson arrival process. Using the method of asymptotic analysis under the asymptotic condition of the long delay in the orbit, we obtain mean asymptotic κ_1/σ and dispersion as κ_2/σ and build the Gaussian approximation for the probability distribution of the number of calls in the orbit in the considered RQ-system. Comparing with the results of simulation, it is shown that the accuracy of the approximations increases with decreasing parameters σ and the system load.

REFERENCES

1. G. Falin, J. G. Templeton, Retrial queues, Vol. 75, CRC Press, 1997.
2. J. G. Templeton, Retrial queues, Top 7 (2) (1999) 351–353.

3. J. R. Artalejo, A. Gómez-Corral, Retrial queueing systems, Mathematical and Computer Modelling 30 (3-4) (1999) xiii–xv.
4. V. A. Zhozhikashvili, V. M. Vishnevsky, Queuing Networks: Theory and Application to Computer Networks, Radio and communication, 1988.
5. G. P. Basharin, Analysis of queues in computer networks: Theory and calculation methods, The science. Ch. ed. phys.-mat. lit., 1989.
6. C. Kim, A. Dudin, S. Dudin, O. Dudina, Tandem queueing system with impatient customers as a model of call center with interactive voice response, Performance Evaluation 70 (6) (2013) 440–453.
7. B. K. Kumar, R. Sankar, R. N. Krishnan, R. Rukmani, Performance analysis of multi-processor two-stage tandem call center retrial queues with non-reliable processors, Methodology and Computing in Applied Probability (2021) 1–48.
8. V. M. Vishnevsky, A. A. Larionov, O. V. Semyonova, Evaluating the performance of a high-speed wireless tandem network using centimeter and millimeter-wave radio channels in road safety management systems, Management problems (4) (2013).
9. N. Kuznetsov, D. Myasnikov, K. Semenikhin, Optimal control of data transmission in a mobile two-agent robotic system, Journal of Communications Technology and Electronics 61 (12) (2016) 1456–1465.
10. K. Avrachenkov, U. Yechiali, Retrial networks with finite buffers and their application to internet data traffic, Probability in the Engineering and Informational Sciences 22 (4) (2008) 519–536.
11. L. E. Meester, J. G. Shanthikumar, Concavity of the throughput of tandem queueing systems with finite buffer storage space, Advances in Applied Probability 22 (3) (1990) 764–767.
12. V. Klimenok, R. Savko, A retrial tandem queue with two types of customers and reservation of channels, in: Belarusian Workshop on Queueing Theory, Springer, 2013, pp. 105–114.
13. C. Kim, A. Dudin, V. Klimenok, Tandem retrial queueing system with correlated arrival flow and operation of the second station described by a markov chain, in: International Conference on Computer Networks, Springer, 2012, pp. 370–382.
14. A. A. Nazarov, S. P. Moiseeva, The method of asymptotic analysis in queuing theory (2006).

UDC: 519.218

A generalized loss priority system, with application to bandwidth sharing

S.S. Rogozin[1,2]

[1]Institute of Applied Mathematical Research Karelian Research Centre RAS, Petrozavodsk, Russia

[2]Petrozavodsk State University, Petrozavodsk, Russia

ppexa@mail.ru

Abstract

Some priority loss queueing systems and their generalisations are considered. In particular, we study a modification of Adan-Weiss loss system with patient non-priority customers, a system with blocking based on infinite server system (proposed by W.Whitt) and a bandwidth sharing model. We study relations between these models. Moreover, an application of Whitt blocking model for bandwidth sharing to obtain bounds for stationary number of customers for general service times is proposed. Finally, a numerical example is considered to compare the results in the case of exponential service times.

Keywords: loss system, bandwidth sharing, priority system, generalized Erlang model

1. Introduction

We consider some generalized multi-server models with losses and priorities. In particular, we recall well-known Erlang model and then consider Adan-Weiss reversible multi-class system [3] and also Whitt multi-station system with blocking [4], in which customers requires servers on several stations simultaneously. First, we show that Erlang model is a particular case of each two latter models. We consider the modification of Adan-Weiss model with second priority non-loss customers, and give stability condition of this system. Moreover, we give another interpretation of Whitt blocking model which can be potentially used to find bounds of the stationary number of users sharing a bandwidth in full utilization regime. The main contribution of this work is that, as we show, some results hold for general service times, while the known results are mainly established for exponential service times only, see for instance [6, 7]. Finally, a numerical example is presented to support theoretical results.

2. Models description

2.1. Erlang model. First, we recall well-known Erlang loss model containing J identical (parallel) servers. Customers follow Poisson process with rate λ, and have exponential service times with mean $1/\mu$. If all servers are busy then the customer is lost. The stationary distribution P_k that k servers are busy satisfies the following well-known Erlang formula:

$$P_k = \frac{\frac{\rho^k}{k!}}{\sum_{i=1}^{J} \frac{\rho^i}{i!}}, \ k = 0, \dots, J, \tag{1}$$

where $\rho = \frac{\lambda}{\mu}$ is the traffic intensity. This result has been extended from exponential service times to general service times (with the same mean) in paper [5].

2.2. Adan-Weiss model. This system is a modified multi-class Erlang system introduced and analyzed in the paper [3]. In this system each server can process only a limited set of customer classes. In general, different classes of customers, following independent Poisson input processes, have different arrival rates. Also each server in general has class-dependent service rate. For each customer class and for each set of the idle servers the *assignment probability* of this class customers to each available server must be found. The results obtained in [3] show that one can choose the assignment probabilities in such a way that the system is described by a reversible Markov process. In this case the stationary distribution $\{\mathsf{P}_i\}$ of the number of customers in the system has an explicit *product form*.

In the paper [2], we have combined the latter system with a priority system considered in the work [1]. In this system, the first priority (class-1) customers are divided into I subclasses which correspond to the classes of customers in the paper [3], while the second priority (class-2) customers form an *infinite capacity* queue if find all servers busy. A class-2 customer, being served, is interrupted by a new class-1 customer and resume service as soon as a server becomes available again. We note that the stationary distribution of class-1 customers in this system remain the same as in [3], because class-2 customers do not affect the assignment and service of the priority class-1 customers. For this system one can use stationary distribution $\{\mathsf{P}_i\}$ found in [3] to obtain stability condition of the modified system as follows:

$$\rho_2 + \sum_{i=1}^{J} i\mathsf{P}_i < J. \tag{2}$$

It is worth mentioning that condition (2) holds only if the assignment probabilities of class-1 customers satisfy some system equations, which is adapted from [3].

2.3. Whitt blocking model. Another multi-class loss system is described in the paper [4]. This system consists of n facilities (stations) and facility i has s_i servers. There are c classes of customers, and a class-j customer requires exactly one server at each facility of a subset of servers A_j. Class-j customers follow Poisson input process with rate λ_j and have service times with general distribution and finite mean $1/\mu_j$. An important result obtained in [4] for this system and which is a key one for our further analysis is as follows. Denote by N_j the stationary number of class-j customers in the system and N_j^∞ the stationary number of class-j customers in a similar system, provided that each facility has *infinite number of servers*. Then the stationary distribution of $(N_1, \dots, N_c)$ is given by the following expression:

$$\mathsf{P}(N_j = k_j, 1 \le j \le c) = \mathsf{P}(N_j^\infty = k_j, 1 \le j \le c \Big| \sum_{j \in C_i} N_j^\infty \le s_i, 1 \le i \le n) = \\ = \frac{\mathsf{P}(N_j^\infty = k_j, 1 \le j \le c)}{\mathsf{P}(\sum_{j \in C_i} N_j^\infty \le s_i, 1 \le i \le n)}, \tag{3}$$

where C_i is the subset of customer classes which require facility i. That is $C_i = \{j : i \in A_j\}$. At that, the stationary distribution of vector $(N_1^\infty, \dots, N_c^\infty)$ is known and given by

$$\mathsf{P}(N_j^\infty = k_j, 1 \le j \le c) = \prod_{j=1}^{c} \mathsf{P}(N_j^\infty = k_j) = \prod_{j=1}^{c} \frac{\rho_j^{k_j}}{k_j!} e^{-\rho_j}, \tag{4}$$

where $\rho_j = \lambda_j/\mu_j$ is the traffic intensity. We note that if the system has only *one facility* with J servers then the system becomes conventional Erlang loss system described in Section 2.1.

3. An application for bandwidth sharing

In this section we describe another interpretation of the Whitt model described in Section 2.3. First, suppose that we have an Internet broadband connection and a class-k user needs frequency width w_k of the given bandwidth or b_k units of the *basic digital channels*, $k = 1, \dots, K$. We assume that class-k users follow Poisson process with rate λ_k and service times are exponentially distributed with mean $1/\mu_k$. Such a network model is considered, for instance, in [6, 7], where in particular the stationary distribution of the number of class-k users in the system is given. Let $\mathbf{n} = (n_1, \dots, n_K)$ be the vector state of the corresponding Markov process describing the dynamics of this system, where n_i is the number of class-i users in the system, and we denote by S the state space of the process. Then the stationary distribution

satisfies [6, 7]

$$p(\mathbf{n}) := P(n_1, \ldots, n_K) = \frac{1}{G} \prod_{k=1}^{K} \frac{\rho_k^{n_k}}{n_k!}, \quad \mathbf{n} \in S, \tag{5}$$

with normalization constant

$$G = \sum_{\mathbf{n} \in S} \prod_{k=1}^{K} \frac{\rho_k^{n_k}}{n_k!},$$

and $\rho_k = \lambda_k / \mu_k$ being the traffic intensity (offered load).

3.1. Numerical examples. We now consider numerical examples related to the latter model, in which there are $n = 3$ servers $b_1 = 2$ and $b_2 = 1$. Using formula (5) we obtain probabilities for exponential case:

$$\begin{gathered} G = (1 + \rho_2 + \frac{\rho_2^2}{2} + \frac{\rho_2^3}{6}) + \rho_1(1 + \rho_2), \\ p(n_1 = 0) = \frac{1}{G}(1 + \rho_2 + \frac{\rho_2^2}{2} + \frac{\rho_2^3}{6}), \quad p(n_1 = 1) = \frac{1}{G}\rho_1(1 + \rho_2), \\ p(n_2 = 0) = \frac{1}{G}(1 + \rho_1), \quad p(n_2 = 1) = \frac{1}{G}\rho_2(1 + \rho_1), \\ p(n_2 = 2) = \frac{1}{G}\frac{\rho_2^2}{2}, \quad p(n_2 = 3) = \frac{1}{G}\frac{\rho_2^3}{6}. \end{gathered} \tag{6}$$

Also we can calculate the blocking probabilities:

$$\pi(1) = \frac{1}{G}(\rho_1(1 + \rho_2) + \frac{\rho_2^2}{2} + \frac{\rho_2^3}{6}), \quad \pi(2) = \frac{1}{G}(\rho_1\rho_2 + \frac{\rho_2^3}{6}),$$

where $\pi(k)$ is the stationary probability that user of class-k will be lost.

Now we calculate the stationary probabilities using Whitt formula (3) for similar model.

Let class-k contains $\binom{n}{b_k}$ subclasses of customers, and we denote by J_k the set of these subclasses. For each subclass $j \in J_k$ we denote by $A_j = \{j_1, j_2, \ldots, j_{b_K}\}$ the set of the required facilities (stations), that is, the subset A_j is one of $\binom{n}{b_k}$ combinations (of the stations numbers). Now we will treat these subclasses as the classes of customers in the Whitt blocking model from subsection 2.3. Further, we split the arrival Poisson process to a few Poisson processes with equal rates. Finally, we use Whitt formula (3) to obtain the stationary distribution, and then return to original classes by summing up the probabilities of the corresponding subclasses.

Now we consider the same example as presented above to compare these systems. Let there are $n = 3$ servers, $b_1 = 2$ and $b_2 = 1$. Then the subsets A_j are given by:

$$\begin{aligned} A_1 &= \{1,2\}, \quad A_4 = \{1\}, \\ A_2 &= \{1,3\}, \quad A_5 = \{2\}, \\ A_3 &= \{2,3\}, \quad A_6 = \{3\}, \end{aligned} \tag{7}$$

Then we can calculate the stationary probabilities using formula (3) and blocking probabilities, for example, we have:

$$\begin{aligned} L &= \rho_1(1+\frac{\rho_2}{3}) + (1+\frac{\rho_2}{3})^3, \\ b(1) &= \frac{1}{L}\Big((\rho_1+\frac{\rho_2}{3})(1+\frac{\rho_2}{3}) + \frac{\rho_1}{3}(1+\frac{\rho_2}{3})^2\Big), \\ b(2) &= \frac{1}{L}\Big(\frac{2\rho_1}{3}(1+\frac{\rho_2}{3}) + \frac{\rho_1\rho_2}{9} + \frac{\rho_1}{3}(1+\frac{\rho_2}{3})^2\Big), \end{aligned}$$

where $b(k)$ is the stationary probability that user of class-k will be lost.

The numerical results confirm a difference between the system by Whitt and the loss model described by the distribution (5). This difference is caused by the fact that, in the Whitt model, there are some additional limitations for arriving customer to occupy a server. Our conjecture is that indeed the model by Whitt can be potentially useful to construct the bounds of the blocking probability for the conventional system (5) but with *general service time distribution*. On the other hand, this system can be useful itself in the bandwidth sharing analysis in the case, when there are some limitations on the available servers depending on the class of the arriving customer.

4. Conclusion

We consider the priority loss queueing systems and some their generalisations, including modification of Adan-Weiss loss system with persist non-priority customers, a blocking system proposed by W.Whitt and a bandwidth sharing model. Relations between these models and the possibility of using in the bandwidth sharing problem are briefly discussed. Some numerical examples are given as well.

REFERENCES

1. Morozov, E., Rogozin, S., Nguyen, H.Q., Phung-Duc, T.: Modified Erlang loss system for cognitive wireless networks. Journal of Mathematical Sciences (2021) (submitted).

2. Rogozin, S., Simulation a modified Erlang loss system with multi-type servers and multi-type customers. Proceedings of The Second International Workshop on Stochastic Modeling and Applied Research of Technology (SMARTY-2020), volume 2792 of CEUR Workshop Proceedings, 2020.
3. Adan, I., Hurkens, C., Weiss, G.: A reversible erlang loss system with multitype customers and multitype servers. Probability in the Engineering and Informational Sciences 24(4), 535 - 548 (2010).
4. Whitt W., Blocking When Service is Required from Several Facilities Simultaneously. AT&T Technical Journal, 64(8), 1807 - 1856 (1985).
5. Sevastyanov, B.A., An ergodic theorem for Markov processes and its application to telephone systems with refusals. Theory of Probability and its Applications 2:104–112 (1957).
6. Basharin, G.P., Lectures on Mathematical Teletraffic Theory Moscow: PFUR, 2009 (in Russian).
7. Ross, K.W., Multiservice Loss Models for Broadband Telecommunication Networks, Telecommunication Networks and Computer Systems, 1995.

UDC: 004.75

An innovative solution for analyzing the dynamics of slowly developing processes of changing the geometry of engineering structures using the example of a system for strengthening a rocky slope

K.I. Mikhaylov[1] and A.G. Abramov[2]

[1, 2]Peter the Great St.Petersburg Polytechnic University, Polytechnicheskaya, 29, St.Petersburg, Russia

k.mikhaylov@gmail.com, abramov_ag@spbstu.ru

Abstract

The paper presents and discusses the developed original methods, technologies and the results of the implementation of a pilot project to ensure the possibility of automated control of a protective structure mounted on a potentially dangerous natural slope along a high-speed railway line. The information on the technologies used for "digitizing" the slope, the principles and methods of measuring, collecting, storing and delivering continuously collected data to a specialized software platform is given. The mathematical methods inherent in the algorithms for processing measurement data are considered, the developed capabilities of the platform for intellectual data analysis and visualization are presented.

Keywords: rocky slope, engineering protection, measurement sensors, LoRAWaN, automated monitoring and control system, software platform, SAYMON, time series, statistical analysis, data visualization, IoT

1. Introduction

As a result of various types of natural disasters around the world, every year, thousands of people are seriously injured and die, significant material damage is inflicted on the infrastructure of settlements and transport. One of the most common types of natural threats is rockfalls - the fall of fragments of rocks, boulders and large stone masses from mountains, stone slopes and walls. The main methods and scenarios for the combating slope collapse and rockfall have been thoroughly worked out in solving the problems of protecting highways and railways in mountainous areas: the slope draping system (covering with a metal mesh) [1, 2] and rockfall

barriers (net panels, struts, base plates, foundation and anchor elements, brakes, ropes) are traditionally used.

With regard to the life cycle of protective structures, it is quite obvious that the longest part of it is the operation of the installed structures. Manufacturers' warranties often last for decades, but despite this, it is required to carry out systematic control of the movement of rocks, deformation of protective meshes and other structures involved. Periodic visual assessment of changes is not always fully realizable, it may not give an objective picture of the processes, it is resource-intensive; the human factor that can affect the quality of monitoring cannot be also disregarded.

In recent years, thanks to significant progress in various engineering industries, the emergence of new telecommunication technologies, highly sensitive sensors, as well as advanced information and analytical cloud platforms (when used together, characterized as technologies of the Internet of things, IoT), the security engineering industry has gained access to new technologies and tools to ensure reliable protection of structures, life and health of people.

Real-time monitoring and control systems are used today in almost all sectors of the economy. The corresponding solutions are based on specialized software distributed both in commercial terms and free of charge (the latter is often open source). Solutions can be placed in dedicated own or leased computing resources and storage. An economically justified and increasingly popular approach is to leverage the resources of cloud service providers within SaaS (Software as a Service) / PaaS (Platform as a Service) models (see, for example, [3, 4]).

Russian software developers have created a number of systems focused on building cloud services for remote monitoring and digital asset management. In this work, the platform named "Central Pult" has been involved, which is included in the Unified Register of Domestic Software [5]. Such a choice has been due to the availability of the possibility of building a centralized, distributed solution, the openness and high level of documentation of the programming interaction protocols (API), the practical readiness to connect various data sources (including devices and instruments of the IoT class), as well as functionally rich, customizable and responsive user web interface.

The applied problem solved within the framework of the study is associated with a real natural object - a rocky slope located along the line of a high-speed railway. The slope is equipped with a rockfall mesh held by metal ropes, which are fixed to the rocks by means of an anchoring system.

The proposed solution for the engineering protection of the slope corresponds to the current state of the art in the industry [6, 7, 8, 9] and assumed the additional equipment of the net of ropes holding the rockfall system with inclination and tension sensors. Devices containing sensors and a base station of the IoT standard for

wireless data exchange LoRaWAN (Low Range Wide Area Network) [10, 11] have been manufactured by the company Vega-Absolute [12]. Data from the devices are transmitted when events and timers are triggered; settings of data transmission conditions are made remotely. The LoRaWAN wireless network is used to transfer data and settings. The continuous operation of a field device on a network of this standard with a single battery can be up to 5 years due to the economy of the used bandwidth and the lack of constant connectivity. The base station provides a radius of connection of devices, measured in kilometers; the specific range depends on the terrain and the relative placement of devices and the base station [10].

2. The general scheme of the solution and the principles of interaction of its individual components

The general scheme of the developed solution is shown in Fig. 1 indicating the links between its main components.

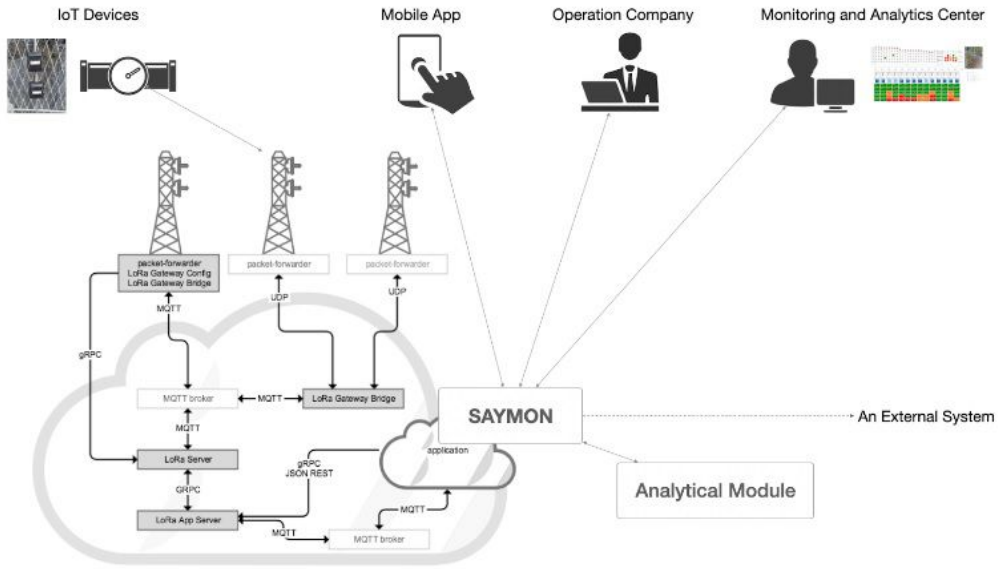


Fig. 1. General scheme of the automated monitoring system

The deployed LoRaWAN base stations are connected to the LoRa server, which performs tasks of controlling the access of devices to the network and (optionally) protecting the transmitted data, as well as routing information to the application server. The digital platform SAYMON acts as an application server. At the moment the data enters the platform, a special converter module is automatically launched,

which allows receiving and processing data from the LoRa server in agreeing formats, taking into account the specifics of an individual server.

The platform solves a whole pool of interrelated tasks of receiving, processing, operational classification, long-term storage, machine analytics and visualization of data [6]. The software has a software defined hierarchy of objects, each of which receives data after sending it from sensors installed on the rockfall protection mesh of the digitized natural slope.

The data transfer frequency depends on a number of factors and in the project under consideration is on the order of tens of seconds. A movement message or an incident a change in the angle of inclination of a sensor is registered in the system, in a specific object of a data model, and at the first stage is classified in terms of the static conditions specified for the object. A classified movement or accident on an object leads to a change in the color of the visualization and may be accompanied by the notification of the responsible personnel or an automatic action such as advanced calculations or services. The classification and actions of the system are flexibly configurable both for groups of objects and for a specific data object.

3. Analysis and visualization of measurement data

3.1. Mathematical methods for processing of measurement data. The analytical module implemented in the course of the project uses the algorithm for analyzing the historical data accumulated in the system and generates indicators of the dynamics of the change in the angle of inclination on a set of observation periods. This approach is quite justified and effective for fixing slowly developing processes. In particular, the algorithm for the accumulated change in the angle of inclination on the time periods of one hour, one day and one week is implemented. For each of the connected sensors, the delta of the change over the period is calculated as the absolute value of the difference between the maximum and the minimum values.

The calculation is performed for the history of recorded indicators for each sensor and allows to obtain the value of the dynamic deviation, regardless of the frequency of measurements and the quality of the time series. In addition, based on these values, one can track local peaks in periods and carry out subsequent visual or automated analysis at shorter time intervals. Obtaining aggregates from the built-in database of time-series (for example, finding the maximum for a period of 1 hour) is performed according to the formula:

$$max(1h) = F('1h - max', now() - 60 \cdot 60 \cdot 1000, now()) \quad (1)$$

Here now() is the time at the moment of starting the instance of the calculation procedure, rounded up to seconds and multiplied by 1000 (since the system stores

data with a resolution of one thousandth of a second). The averaging parameter '1h-max' informs the platform that it is necessary to return the maxima with hourly aggregation at a given time interval (second and third parameters). To obtain the minimum, a parameter with the value '1h-min' is used. A similar approach is applied for day and week intervals.

The values regularly computed in the described way for each sensor form additional calculated time series to the main time series of measurements, which can be visualized and analyzed. For the published data, the analytical module programmatically sets the classification criteria - the conditions for changing the state to alarming or normal with subsequent visualization and processing of associated actions. The conditions are represented as JSON (JavaScript Object Notation) descriptions.

The criteria for the operative classification of secondary (calculated) quantities implemented in this way allow building chains of actions: launching additional calculations, notifying the responsible personnel or visualizing the development of the situation for dispatching services.

Hierarchies (graphs) of information objects, time-series and mechanisms of state change propagation integrated into the software platform allow creating not only a digital copy of a slope at a specific point in time, but also considering the development of situations in a historical context. The implementation of such a "digital twin" of the protective mesh allows continuous analysis of developing processes and provides fundamental advantages over episodic visual observation (the role of which should not be completely ruled out, of course).

The frequency of calculations in the completed scope of the project was not a critical characteristic. The automated calculation has been performed every five minutes, since this period of time is sufficient for the selected observation periods. The work involved a simple classification of normal, alarm and emergency states when the modulus of the calculated value changes: less than 5 degrees - normal (green), more than 5 degrees - warning (orange) and more than 10 degrees - emergency (red).

3.2. Visualization of measurement data. The software platform selected for the implementation of the project allows designing visual web representations of data models through user and software interfaces. In particular, for users performing a substantive analysis of the dynamics of the behavior of an anti-rockfall construction, the view shown in Fig. 2.

The image at the top of the figure shows the location of the devices, taking data on the slope as well as a photograph of the slope and structure. For each device, a formed column of measured and calculated indicators is shown, visually reflecting the degree of normality of digital values (green, orange, red). The measured temperature, the battery charge of the device, the color designation of the presence of the registered

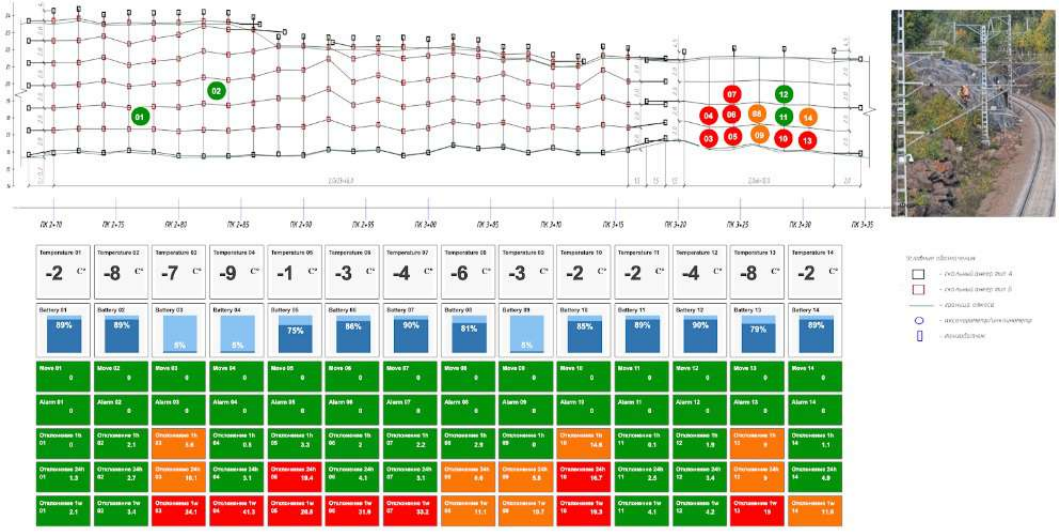


Fig. 2. Visual representation of the digital slope

movement and alarm, as well as the lines of visualization of the deviation state at the periods of one hour, day, week are shown.

The hierarchical data model of the platform allows storing calculated values in objects subordinate to the original ones, and transferring states from subordinate objects to higher ones in accordance with the priorities and weights of states. In the work, the values have been recorded in the objects subordinate to the devices, which makes it possible to visualize the sensor on the slope scheme as alarming in cases when the calculated indicators on the observation horizons go beyond the threshold values.

For each of the devices, a visual representation of changes is available on the dashboard in the form of graphs and values of hourly, daily and weekly maximums (Fig. 3). The level of detail of the aggregated data, if necessary, can be increased. The displays of the total values and graphs are collected in a visual picture programmatically - metadata and instructions for the web browser are formed for arranging images on the screen. Each line in the figure is a hyperlink by which one can move to a more detailed presentation of graphs and tables of values of quantities.

Additionally, the software platform provides the ability to add graphs in a form similar to that shown in the figure. Moreover, when forming such representations, mathematical functions are available that allow to perform summation and other mathematical operations with the extracted data before visualization.

Fig. 3. Visual annotation of data

4. Conclusion

A notably science-intensive problem and a high practical demand for analyzing and controlling the dynamics of slowly developing processes of changing the geometry of protective engineering structures encourages engineers and researchers to develop and implement up-to-date methods and technological solutions based on digital standards and technologies.

Based on the solution considered in the work, the pilot project has been implemented to ensure continuous automated control of the protective structure mounted on the potentially dangerous natural slope along a high-speed railway line. The obtained measurement and computed data are of a qualitative nature and are suitable for further use.

The assembled scheme provides opportunities for sharing locally measured data with predictive data obtained as a result of analytical computations, with data from open sources (meteo-, seismic, cosmic data), as well as the use of promising artificial intelligence systems opens up broad prospects for performing on the modern level of applied research and development, the use of innovative digital tools to ensure safety and increase the economic efficiency of the operation of dangerous geographically distributed objects.

REFERENCES

1. Cheer D., Giacchetti G. Rock and soil slope protection using a high stiffness geocomposite mesh system // In Proc. 2013 Int. Symp. on Slope Stability in

Open Pit Mining and Civil Engineering, pp. 1273-1284, Australian Centre for Geomechanics, Perth, 2013. doi:10.36487/ACG_rep/1308_90_Cheer
2. Badger T. C., Duffy J. D., Schellenberg K. Protection in rock-fall: characterization and control // In Transportation Research Board of the National Academies, Washington, USA, 2012, pp. 495–525.
3. Pourmajidi W., Steinbacher J., Erwin T., Miranskyy A. On challenges of cloud monitoring. 2018/06/15, https://www.researchgate.net/publication/325816940_On_Challenges_of_Cloud_Monitoring
4. Ward J. S., Barker A. Self managing monitoring for highly elastic large scale cloud deployments // In Proc. Sixth International Workshop on Data Intensive Distributed Computing, DIDC'14 (New York, NY, USA, 2014). https://doi.org/10.1145/2608020.2608022.
5. Software platform SAYMON (developed by the company "UNTU, Inc."), https://saymon.tech
6. Segalini A., Savi R., Cavalca E., Valletta A., Carri A. Innovative application of IoT technologies to improve geotechnical monitoring tools and early warning performances // Springer Series in Geomechanics and Geoengineering. 2021. P. 142–146. doi: 10.1007/978-3-030-61118-7_12
7. Park S., Lim H., Tamang B., Jin J., Lee S., Chang S., Kim Y. A Study on the slope failure monitoring of a model slope by the application of a displacement sensor // Journal of Sensors. 2019, Article ID 7570517, 9 p. doi:10.1155/2019/7570517
8. Barile G., Leoni A., Pantoli L., Stornelli V. Real-time autonomous system for structural and environmental monitoring of dynamic events // Electronics. 2018. Vol. 7(12). P. 420. doi:10.3390/electronics7120420
9. Lee H.-C., Ke K.-H., Fang Y.-M., Lee B.-J., Chan T.-C. Open-Source wireless sensor system for long-term monitoring of slope movement // IEEE Transactions On Instrumentation and Measurement. 2017. V. 66(4). 10 p.
10. LoRaWAN Specification v. 1.1. Open Standard Released for the IoT; LoRa Alliance: Fremont, USA, 2015, https://lora-alliance.org/resource_hub/lorawan-specification-v1-1
11. Haxhibeqiri J., Poorter E. D., Moerman I., Hoebeke J. A Survey of LoRaWAN for IoT: From Technology to Application // Sensors. 2018. V. 18(11). P. 3995.
12. Gusev O. Slope strengthening system: automated monitoring by LoRaWAN protocol // Informatization and Control Systems in Industry. 2020. No. 5(89). https://isup.ru/articles/3/16023/

УДК: 519.876.5

«Эффективность радиочастотной идентификации транспортных средств с использованием аналитической аппроксимацией и имитационного моделирования»

И.А. Федотов[1], А.А. Ларионов[1], Е.А. Михайлов[2]

[1]Институт проблем управления им. В.А.Трапезникова РАН, Профсоюзная 65, Москва, Россия

[2]Московский государственный университет им. М.В.Ломоносова, Ленинские горы 1, Москва, Россия

fedotov.ia15@physics.msu.ru, larioandr@gmail.com, ea.mikhajlov@physics.msu.ru

Аннотация

В данной работе описывается технология идентификации транспортных средств в потоке с помощью технологии радиочастотной идентификации (RFID). Целью работы было произвести оценку вероятности успешной идентификации при различных условиях и параметрах работы считывателя. Для этого была построена имитационная модель на основе стандарта EPC gen 1 class 2 и аналитическая модель.

Моделирование показывает, что при определенных условиях вероятность успешной идентификации превышает 0,9. Данный результат согласуется с реальными данными, полученные в ходе двух экспериментов в Казани в 2015 и 2020 годах.

Ключевые слова: RFID, безопасность дорожного движения, моделирование

1. Введение

Ежегодно на автодорогах происходит огромное количество дорожно-транспортных происшествий. Из-за них наносится ущерб транспортным средствам, дорожной инфраструктуре и людям - водителям и пассажирам. За 2020 год в России произошло 145 тысяч ДТП, в которых пострадало более 180 тысяч человек[1].

В большинстве стран мира для решения проблемы аварийности на дорогах используют различные методы фиксации нарушений, например, камеры. Но комплексы видео-фиксации имеют ряд недостатков. При плохих погодных условиях вероятность успешной идентификации автомобиля падает до 50%. Поэтому

предлагается рассмотреть и проанализировать технологию радиочастотной идентификации транспортных средств, которая позволяет фиксировать автомобили на расстоянии до 10 метров независимого от погодных условиях и прямого зрительного контакта.

В данной работе производится оценка вероятности успешной идентификации транспортных средств с помощью технологии RFID (Radio-Frequency IDentification) при различных настройках комплекса и различными методами.

2. Обзор технологии RFID

Радиочастотная идентификация применяется во многих сферах деятельности, в таких как логистика, учет товаров, системы доступа и многие другие. Особое место занимает технология идентификации в транспортной сфере. Уже в 1991 году появились первые системы сбора оплаты проезда по платным дорогам в США с помощью RFID [4].

Активно проводятся исследования применения RFID в системах "умный город". В работе [5] авторы предлагают использовать радиочастотную идентификацию для анализа трафика автомобилей в городе и составления моделей.

3. Принцип работы комплекса

Любая RFID-система принципиально состоит из двух компонент:

1) RFID-метка - пассивное устройство, основная задача которого передача своего идентификатора
2) RFID-считыватель - активное устройство, которое идентифицирует метки в своей зоне видимости.

На номер автомобиля помещается метка, которая содержит информацию о транспортном средстве. Над дорогой устанавливается считыватель, который инвентаризирует метки в области своей видимости. На рис 1 показана схема установки оборудования и моделируемый процесс. Автомобили двигаются с постоянной скоростью и одинаковыми интервалами между друг другом.

Так как метки пассивные и не знают о существование друг друга, то общение начинает и организует считыватель. Обмен сообщениями осуществляется на основе стандарта EPC class 1 gen 2 [6]. Протокол содержит средства предотвращения коллизий за счет разнесения ответов меток по времени.

Принцип передачи сигналов от метки к считывателю основан на методе обратного рассеяния [7]. Метка получает энергию из сигнала считывателя и отражает измененный сигнал, содержащий ответ. Если сообщения считывателя доставляются без ошибок, то ответы меток могут содержать битовые ошибки. Это происходит из-за более слабого сигнала у меток и помех, например, отраженного сигнала от асфальта.

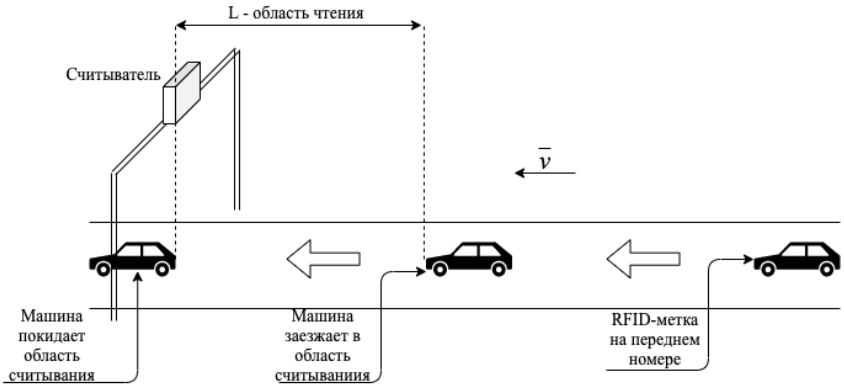


Рис. 1. Схема установки

4. Аналитическая аппроксимация вероятности

В работе [8] показана зависимость значения вероятности битовой ошибки от расстояния до считывателя. Если аппроксимировать данную зависимость (рис. 2) двумя параболами, то можно получить аналитическое выражение (6) для величины вероятности успешной идентификации транспортного средства.

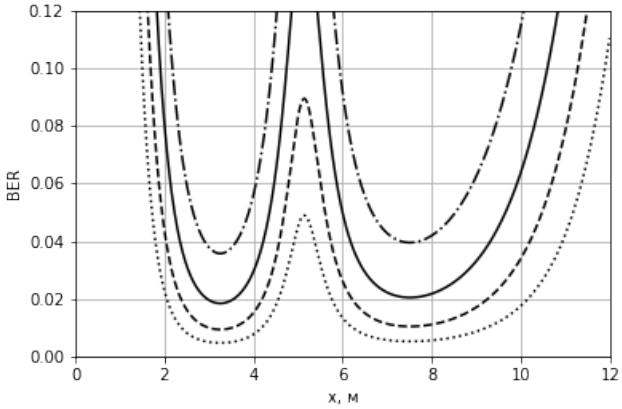


Рис. 2. Зависимость вероятности битовой ошибки от координаты автомобиля при t_{tari}=6,25 мс. Где штрих-пунктирная - FM0, сплошная - M2, пунктир - M4, штриховая - M8.

$$\beta(x) = \begin{cases} a_1 + b_1(x - x_1)^2, x < L_0 \\ a_2 + b_2(x - x_2)^2, x > L_0 \end{cases}, \quad (1)$$

где $a_1, a_2, b_1, b_2, x_1, x_2$ и L_0 - некоторые коэффициенты парабол.

Вероятность неуспешного раунда на всей области видимости считывателя:

$$P_F = e^{-P_M(x_1)\frac{\Delta x_1}{V\tau(x_1)}} e^{-P_M(x_2)\frac{\Delta x_2}{V\tau(x_2)}} \dots e^{-P_M(x_N)\frac{\Delta x_N}{V\tau(x_N)}} \tag{2}$$

которое можно перезаписать в виде интеграла:

$$P_F = e^{-\int_{VT}^{L} \frac{P_M(x)}{V\tau(x)} dx} \tag{3}$$

Следовательно вероятность успешной идентификации в области чтения считывателя будет:

$$\Pi = 1 - P_F = 1 - \exp\left(-\int_{VT}^{L} \frac{P_M(x)}{V\tau(x)} dx\right) \tag{4}$$

Подставив $P_M(x)$, получим следующую формулу:

$$\Pi = 1 - \exp\left(-\int_{VT}^{L} \exp\left(-\beta(x)Q\right) \frac{dx}{V\tau(x)}\right) \tag{5}$$

Итоговое выражение для оценки успешной вероятности идентификации будет выглядеть следующим образом:

$$\begin{aligned}\Pi = 1 - \exp\Bigg(-\frac{1}{V\tilde{N}\delta}\Bigg[&\frac{\sqrt{\pi}e^{-a_1(L-\tilde{L})}}{2\sqrt{b_1(L-\tilde{L})}}\Bigg[\Phi\left((L_0 - x_1)\sqrt{b_1(L-\tilde{L}}\right) + \\ &\Phi\left((x_1 - VT)\sqrt{b_1(L-\tilde{L}}\right) + \\ &+ \frac{\sqrt{\pi}e^{-a_2(L-\tilde{L})}}{2\sqrt{b_2(L-\tilde{L})}}\Bigg[\Phi\left((L - x_2)\sqrt{b_2(L-\tilde{L}}\right) + \Phi\left((x_2 - L_0)\sqrt{b_2(L-\tilde{L}}\right)\Bigg]\end{aligned} \tag{6}$$

5. Имитационная модель

Для оценки вероятности идентификации транспортного средства была реализована имитационная модель на языке Python. В ней моделировался процесс движения автомобилей, въезд и выезд в зону видимости, обмен сообщениями между метками и считывателем. В сообщениях от меток могли содержаться ошибки.

Вероятность того, что сообщение доставлено без ошибки:

$$P_i = (1 - P_e)^{Q_i}, \tag{7}$$

где Q_i - длина сообщениях в битах, P_e - вероятность битовой ошибки.

Всего 4 условных этапа обмена сообщениями между меткой и считывателем. Их длина равна 16, 128, 32 и 97 битов соответственно. Данные сообщения необходимы для налаживания связи между считывателем и меткой, и для передачи идентификаторов.

Так как результат модели содержит элементы случайных событий, то модель основывалась на методе Монте-Карло.

6. Результаты

На рис. 3 и рис. 4 показаны результаты имитационного и математического моделирования и их сравнение. Видно, что есть небольшое расхождение, но стабильное. Оно связано с тем, что аналитическая модель не учитывает коллизии ответов меток от транспортных средств, если в области чтения их больше одной.

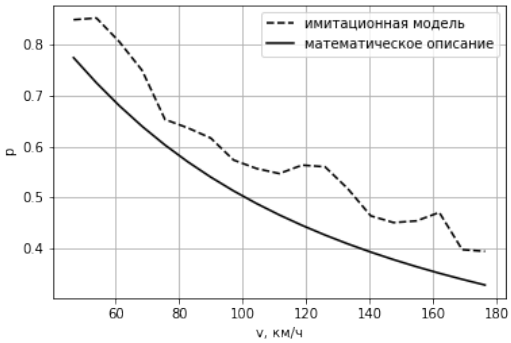


Рис. 3. Сравнение имитационной и математической модели.

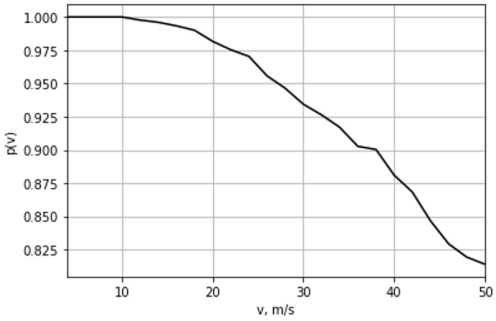


Рис. 4. Имитационная модель при реалистичных параметрах системы.

7. Заключение

В ходе работы были построены аналитическая и имитационная модель, которые дают достаточно адекватные результаты и хорошо согласуются между собой. Аналитическая модель основывалась на аппроксимации распределения битовой ошибки двумя параболами. Вторая модель имитировала процесс движения автомобилей и обмен сообщениями меток со считывателем. Между результатами моделей есть стабильное расхождение, которое объясняется тем, что в аналитической аппроксимации не учитываются коллизии между ответами меток. При определенных условиях настройки протокола вероятность идентификации автомобиля на больших скоростях превышает 90%.

Литература

1. Дорожно-транспортная аварийность в Российской Федерации за 2020 год. Информационно-аналитический обзор. – М.: ФКУ «НЦ БДД МВД России», 2021, 79 с.
2. Вишневский В.М., Минниханов Р.Н. Автоматизированная система безопасности на автодорогах с использованием RFID-технологий и новейших беспроводных средств. Проблемы информатики. 2012. № 1. С. 52-65.
3. Вишневский В.М., Ларионов А.А., Целикин Ю.В., Иванов Р.Е., Козырев Д.В. Опыт реализации системы безопасности на автодорогах с использованием радиочастотной идентификации UHF-диапазона / Proceedings of the 20th International Conference, Distributed Computer and Communication Networks (DCCN 2017, Moscow, Russia). M.: Техносфера, 2017. С. 152-163.
4. J. Landt, "The history of RFID,"in IEEE Potentials, vol. 24, no. 4, pp. 8-11, Oct.-Nov. 2005, doi: 10.1109/MP.2005.1549751.

5. Pawłowicz, B.; Trybus, B.; Salach, M.; Jankowski-Mihułowicz, P. Dynamic RFID Identification in Urban Traffic Management Systems. Sensors 2020, 20, 4225. https://doi.org/10.3390/s20154225
6. EPC Radio-Frequency Identify Protocols. Class-1 Generation-2 UHF RFID. Protocol for Communications at 860 MHz – 960 MHz. Version 2.0.1. EPCGlobal Inc., 2015.
7. Таненбаум Э. С., Дэвид У. Компьютерные сети. 5-е изд. – "Издательский дом Питер 2018.
8. A. A. Larionov, R. E. Ivanov and V. M. Vishnevsky, "UHF RFID in Automatic Vehicle Identification: Analysis and Simulation,"in IEEE Journal of Radio Frequency Identification, vol. 1, no. 1, pp. 3-12, March 2017, doi: 10.1109/JRFID.2017.2751592..

UDC: 025.4.03

The Importance of Conference Proceedings in Research Evaluation: a Methodology for Assessing Conference Impact

D.M. Kochetkov[1], A.A. Birukou[2,3], A.M. Ermolayeva[2]

[1]Ministry of Science and Higher Education of the Russian Federation, Moscow, Russia
[2]Peoples' Friendship University of Russia (RUDN University), Moscow, Russia
[3]Springer Nature, Heidelberg, Germany

kochetkovdm@hotmail.com, birukou@gmail.com, ermolaevaanna@bk.ru

Abstract

Conferences are an essential tool for scientific communication. In disciplines such as Computer Science the majority of original research results are published in conference proceedings. In this study, we have analyzed the role of conference proceedings in various disciplines and propose an alternative approach to research evaluation based on conference proceedings sources indexed in Scopus and Scimago Journal Rank (SJR). This allows one to categorize conference proceedings in quartiles Q1 - Q4 by analogy with SJR journal quartiles. Out of 171 conference proceedings sources analyzed, 38 conference proceedings in Engineering (45% of the list) and 23 in Computer Science (32% of the list) have an SJR level corresponding to the first quartile journals in these areas, which emphasizes the exceptional importance of conferences in these disciplines.The comparison of this bibliometric-driven ranking with the expert-driven CORE ranking in Computer Science showed a 62% overlap, as well as a significant average rank correlation of the category distribution.

Keywords: conference proceedings research evaluation research impact SJR CORE conference rankings bibliometrics

1. Introduction

In many countries (for instance, China [5], India [8], Russia, Turkey * , UK [6]) research evaluation is based on the indicators of the sources, i.e., journals, conference proceedings, book series, in which the results are published. This often leads to labeling publication sources with several predefined classes and judging the

The publication has been prepared with the support of the RUDN University Strategic Academic Leadership Program.The preprint was published on October 4, 2020; available at https://arxiv.org/ftp/arxiv/papers/2010/2010.01540.pdf.

*https://www.urapcenter.org/Methodology, last accessed 03.07.2020

importance of a publication based on the class of the source. As in many research areas original results are published in journals [11], research evaluation policies are often biased towards journals.

In this paper we 1) review the current practices of using conferences in the research evaluation; 2) identify scientific disciplines, where conference proceedings play a significant role in the communication of primary research results; 3) propose a new methodology for the assessment of conference proceedings based on Scopus and Scimago Journal Rank (SJR) data; 4) show that such bibliometric-driven methodology produces classification of conferences similar to the classification designed by domain experts, such as CORE. This article is structured as follows: Section 2 presents the role of conferences in the scientific community. Section 3 describes the methodology of the study, Section 4 presents the results of the study and its limitations, and Section 5 describes the conclusions and prospects for the development of this study.

2. Materials and Methods

At the first stage, in the Scopus sources list [†] we selected those which are as conference proceedings (Conference Proceedings post-1995). In terms of data collection time and completeness, Scopus is the optimal tool for conducting research [10]. The authors [3] who conducted a similar study, but much later than the first one, made the same conclusions. We then focused on those which are currently indexed (i.e., have the ongoing status), and for which an SJR [‡] score is available. This selection resulted in 171 sources with conference proceedings. Note that the way Scopus indexes conferences depends on the conference and the publication outlet (journal, book series, conference proceedings). So the 171 sources we selected contained a much bigger number of conferences, as sources like ACM International Conference Proceeding Series or CEUR Workshop Proceedings publish several hundreds conference proceedings per year.

The SCImago Journal Rank (SJR) is not just a citation indicator such as Impact Factor or CiteScore; it is based on a PageRank-like algorithm, which is an iterative process of prestige transfer among the publication sources. The calculation is an iterative process in which the prestige of each source depends on the prestige of the sources which cite it. The final SJR value is normalized over the number of documents published in the citation window [1].

Given that the SJR is computed based on Scopus data, we also used this database in our analysis. Out of the 171 conference proceedings sources, 153 were assigned

[†]Available at https://www.scopus.com/ (date accessed 04.03.2020).
[‡]Available at https://www.scimagojr.com/ (date accessed 04.03.2020).

one or more subject categories (third level ASJC [§]) in Scopus. For the 18 conference proceedings sources that were not assigned any subject category; we deduced the categories based on publications in Scopus.

Next, for each of the subject categories, we computed the threshold SJR values for the quartiles, in the same way SCImago calculates them for journals. This was necessary because SCImago does not assign quartiles to the conference proceedings sources, only to journals and book series. This allowed us to assign each source to the corresponding quartile (Q1, Q2, Q3, Q4) in each subject category. For example, the minimum SJR for journals and book series of the first quartile is 0.261, the second is 0.139, the third is 0.104, and the fourth is 0.1. The IOP Conference Series: Materials Science and Engineering has an SJR of 0.195, so we can classify the source as Q2. We emphasize that this is not a quartile itself; it is a conditional assignment of conference proceedings source to a quartile based on the SJR value. For journals covering several subjects, CMEPP research evaluation guidelines suggest using the maximum of the quartiles in those subjects. However, the importance of the same conference in different communities varies, as also mentioned in the CCF release notes. We therefore would like to stress the importance of using subject-specific quartiles for conferences, i.e., a conference can belong to several subject categories and can have different quartiles there.

3. Results

The distribution of conference proceedings sources across subject categories is shown in Fig. 1. Note that one conference can belong to several subject categories. Out of 171 sources, one was assigned to five subject categories, one to four, 13 to three, 66 to two, and 90 conferences had only one subject category. Figure 2 shows the distribution of conference proceedings across quartiles in the context of subject categories. If one considers all sources (journals, book series, conference proceedings), the share of each quartile is obviously 25%. However, as our selection is limited to the conference proceedings, the distribution between the categories Q1-Q4, is very different for each subject category. From the graph, it is evident that Engineering and Computer Science have not only the highest share of conference proceedings but also the largest number of high-impact conference proceedings. This once again confirms the thesis that conference proceedings must be considered when evaluating research in these areas. Our results are in line with the results of earlier studies [9]. The difference in the number and quality of conference proceedings between these subject categories and the rest is substantial. The source data is available in [2].

[§] All Science Journal Classification Codes. Available at `https://service.elsevier.com/app/answers/detail/a_id/15181/supporthub/scopus/` (date accessed 04.03.2020).

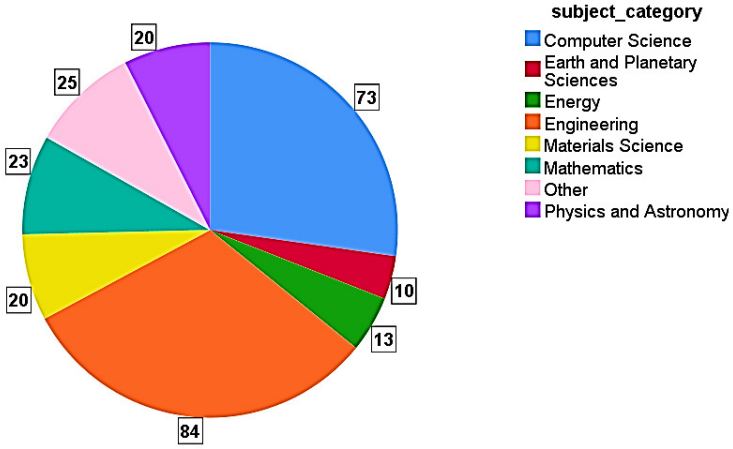


Fig. 1. The distribution of conference proceedings sources across subject categories. Source: authors' own calculations

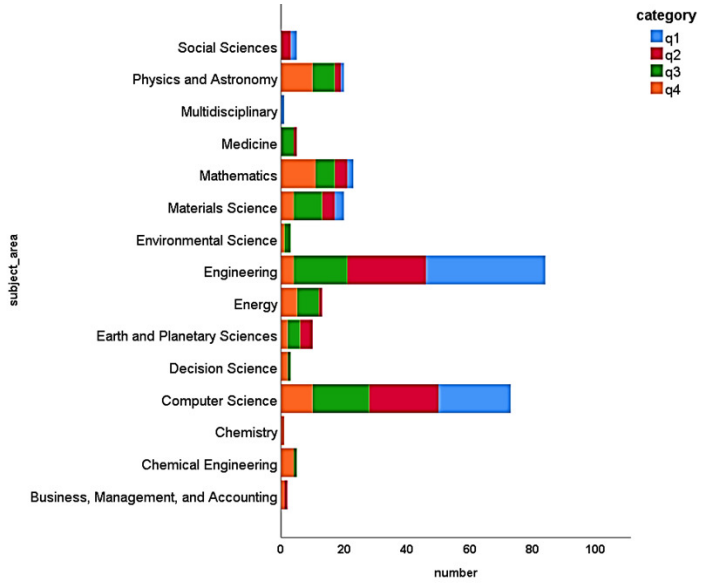


Fig. 2. Distribution of conference proceedings sources into categories. Source: authors' own calculations

Out of the 73 proceedings of Computer Science conferences, 45 conferences (62%) are in the CORE ranking; 10 sources are aggregators that publish the proceedings of

many conferences (e.g., Procedia Computer Science, ACM International Conference Proceeding Series, etc.); and 18 conference proceedings are not core CS conferences, but are from related fields (for example, IEEE MTT-S International Microwave Symposium Digest). The latter appear in our dataset because according to the ASJC classification conferences can fall simultaneously into several subject areas/categories. Such conferences, however, are out of scope for CORE, which focuses exclusively on Computer Science conferences. For the 45 conferences from our list, which are also present in CORE, we compared the distribution by category (Fig. 3, Q1 for SJR corresponds to A * for CORE, Q2 - A, Q3 - B, Q4 - C). Spearman's rank correlation coefficient was 0.452, which suggests an average correlation dependence. This is an interesting fact, given the fundamentally different approaches to the formation of lists, bibliometric and expert. The full table is also presented in the dataset available online (see footnote 12).

SJR 2018/CORE 2018	A*	A	B	C	NA
Q1	11	4	4	1	3
Q2	5	7	2	1	7
Q3	-	2	6	1	9
Q4	-	-	-	1	9
NA	51	407	402	793	

*Source: author's calculations.

Fig. 3. A Comparative Analysis of Distribution of Conferences into Categories*

The study has several limitations:

1. We have evaluated conference proceedings sources, not the conferences themselves. If one would like to evaluate conferences, they should take into account not only the bibliometric data, but also various other parameters: topical scope, program committee, authors, the peer review process, proceedings publication culture, etc. However, a quantitative assessment presented here may be a convenient auxiliary tool, even though it does not eliminate the need for expert evaluation.

2. The list reflects only non-journal and non-book sources. Conference proceedings published in journals and book series (e.g., Journal of Physics Conference Series, Lecture Notes in Computer Science) can use the SJR quartile of the corresponding journal or the book series.

3. The list only reflects conference proceedings with the serial ISSN; some conferences do not receive it due to the oversight of the organizing committee. Such

conferences are not included in the list of serials in Scopus and could not be included in the analysis.

4. The list includes not only the proceedings of individual conferences but also aggregators such as CEUR Workshop Proceedings, Leibniz International Proceedings in Informatics (LIPIcs). The level of conferences within such publications may vary significantly. Unfortunately, the data granularity in Scopus does not allow for the conference-level analysis within these sources.

Even though the CCF recommends not using conference rankings for academic evaluation, [7] shows how such rankings influence publishing behavior of scientists. Therefore, it is important to provide more transparency in how rankings are created, what is included, which metrics are used, etc. The methodology proposed in this paper represents a step in this direction, as it combines transparent bibliometric indicators and correlates with expert opinions.

The authors will continue research on the evaluation of conferences and conference papers. We would like to move towards paper-level metrics, as different papers in the same conference proceedings have different quality, citations, importance. In this regard we would like to mention several projects that aim at providing open identification of conferences, which is the first step before doing any bibliometrics. ConfIDent aims at developing a crodwsourcing platform for providing semantically structured metadata of scientific events [4]. The ConfRef.org project, which was created to provide information on scientific conferences and provide standard identifiers for conferences. The current prototype provides data on 40,000 conferences, mainly from computer science, provided by Springer Nature and DBLP. The primary purpose of ConfRef is to provide trusted information about the history, dates, venues, places of publication / past issues of a series of conferences (and related conferences) in various disciplines (Computer Science, Electrical Engineering, Mathematics), as well as information about upcoming conferences and invitations, dates and information about program committee. On top of this, ConfRef will deal with identifying predatory or fake conferences.

4. Conclusion

In this paper we made an attempt to review the role of conferences in the research evaluation and to identify scientific disciplines, where conference proceedings are an important outlet for publishing original research results. Next to the "usual suspects", i.e. Computer Science, conference proceedings often used for publishing results in Engineering, Mathematics, Energy, Decision Sciences. We also presented a new methodology for applying Scopus and Scimago Journal Rank (SJR) data for the assessment of conference proceedings and showed that it provides similar results to expert-designed ranking, such as CORE. The methodology shows that

some conference proceedings in Computer Science, Engineering, Material Science, Physics and Astronomy, and Mathematics are comparable with Q1-Q2 journals.

Future work includes development of tools which would implement the proposed methodology and work on removing the limitations such as the different granularity of conference proceedings sources

REFERENCES

1. Description of scimago journal rank indicator, 2020. https://www.scimagojr.com/SCImagoJournalRank.pdf.
2. Kochetkov, dmitry; birukou, aliaksandr; ermolayeva, anna (2020), "methodology for conference proceedings assessment: a conference proceedings dataset", mendeley data, v5, may 2020. doi:10.17632/hswn9y67rn.5.
3. Guerrero-Bote, V. P., Chinchilla-Rodríguez, Z., Mendoza, A., and de Moya-Anegón, F. Comparative analysis of the bibliographic data sources dimensions and scopus: An approach at the country and institutional levels. *Frontiers in Research Metrics and Analytics 5* (2021), 19.
4. Hagemann-Wilholt, S., Plank, M., and Hauschke, C. Confident–an open platform for fair conference metadata. In *GL Conference Series; 21* (2020), Amsterdam: TextRelease.
5. in Nature, E. China's research-evaluation revamp should not mean fewer international collaborations. *Nature 579* (2020), 8.
6. Koya, K., and Chowdhury, G. Metric-based vs peer-reviewed evaluation of a research output: Lesson learnt from uk's national research assessment exercise. *Plos one 12*, 7 (2017), e0179722.
7. Li, X., Rong, W., Shi, H., Tang, J., and Xiong, Z. The impact of conference ranking systems in computer science: a comparative regression analysis. *Scientometrics 116*, 2 (2018), 879–907.
8. Madhan, M., Gunasekaran, S., and Arunachalam, S. Evaluation of research in india–are we doing it right. *Indian J Med Ethics 3*, 3 (2018), 221–229.
9. Meho, L. I. Using scopus's citescore for assessing the quality of computer science conferences. *Journal of Informetrics 13*, 1 (2019), 419–433.
10. Meho, L. I., and Yang, K. Impact of data sources on citation counts and rankings of lis faculty: Web of science versus scopus and google scholar. *Journal of the american society for information science and technology 58*, 13 (2007), 2105–2125.
11. Vrettas, G., and Sanderson, M. Conferences versus journals in computer science. *Journal of the Association for Information Science and Technology 66*, 12 (2015), 2674–2684.

UDC: 004.93

Object classification using neural networks with binary input and binary feature extraction

S. Poslavskiy[1], D.V. Shashev[1], S.V. Shidlovskiy[1]

[1]National Research Tomsk State University, Russian Federation, Tomsk

Abstract

Although machine learning by its nature, has been resource-intensive, multiple resource efficient alternative approaches have been made in the field of embedded systems. Researchers over the years, have specially shown interest and thus it has fueled the research output, in the field of autonomous navigation, IoT and distributed embedded systems. These approaches are aimed at finding a trade-off between performance and resource consumption in terms of computational costs and energy efficiency. The development of appropriate algorithms is one of the main tasks of modern research in the field of machine learning and the key to ensuring smooth adaptation of machine learning technologies in an environment with limited or distributed computing resources. These algorithms can be divided into four non-mutually exclusive categories: quantization of neural networks, network pruning, structural efficient algorithms, binarized neural networks. Our approach is proposed for working with binary images and extracting binary features for training neural networks This approach provides the possibility of an implementation based on the architecture of a reconfigurable computing systems. Our approach was used to build a neural network architecture. The resulting architecture was trained and tested on the MNIST dataset. The results were compared with the results of the LeNet-5 architecture.

Keywords: Neural Networks, binary gradient, binary input, binary feature extraction, computational efficiency

1. Introduction

Modern approaches in the field of machine learning often turn out to be especially effective only when large amounts of data and vast computational resources are available to us. However, in real-time applications or applications for embedded systems, the computing infrastructure is usually severely limited already at the design stage, which actually excludes the use of most modern resource-intensive deep learning approaches [1]. To create effective deep learning algorithms in such a class of applications, it is necessary to consider several key problems:

The reported research has been funded by RFBR, project number 19-29-06078.

1) computational efficiency [2];
2) prediction quality;
3) representational efficiency.

Here, deep neural networks (DNN) are focused on as the most common machine learning models. The main areas of research related to improving the efficiency of resource use in DNN:

1) Quantized Neural Networks [3];
2) Network Pruning [4];
3) Binarized Neural Networks. The purpose of binarization is to represent floating point weights and / or activation functions using quantization of their values to 1-bit representation [5]. Binary neural networks, which significantly save memory and computation, are one of the promising options for deploying deep models on devices with limited resources. Despite the fact that binarization inevitably causes a serious loss of information due to rigid quantization of values in the range of 0 to 1, this opens up wide opportunities for creating specialized neural network architectures, greatly simplifies the implementation and application of such networks in embedded systems and systems with a reconfigurable structure. Quantization of weights and activation functions to 1 bit also makes it possible to build classical convolutional and fully connected DNN layers using a mathematical apparatus based on bit or logical operations, which allows achieving a significant increase in the speed of operations and significantly reduces the amount of memory occupied. An example of one of the most discussed architectures of this type is the XNOR-Net architecture [6].

Thus, it can be noted that all of the above areas focus on optimization methods that use various algorithms aimed directly at working with weights and / or activation functions. In this article, we propose a method for working with binary input data quantized by the values 0 and 1. This method allows you to extract binary features of an image, similar to the features of histogram of oriented gradients (HOG), but based on logical functions. The obtained features can be used for further training deep neural network models. This approach to image feature extraction allows us to abandon the use of convolutional layers and makes it possible to adapt computations for embedded systems with a reconfigurable computational structure.

2. Binary data feature extraction

HOG is a feature point descriptor that is based on calculating the directions of the gradient in local areas of the image. The main idea of the algorithm is the assumption that the appearance and shape of an object in the image can be described by the distribution of intensity gradients. Gradient values are calculated in the horizontal and / or vertical direction using a one-dimensional differential mask. This

method requires filtering the color or luma component using the following filter kernels: $[1, 0, 1]$ and $[-1, 0, 1]^T$.

In order to calculate the HOG, the direction and magnitude of the gradient for each pixel in the image is calculated using the following equations 1-4:

$$g_x = f(x+1, y) - f(x-1, y), \tag{1}$$

$$g_y = f(x, y+1) - f(x, y-1), \tag{2}$$

$$m(x, y) = \sqrt{g_x^2 + g_y^2}, \tag{3}$$

$$\theta(x, y) = \tan^{-1} \frac{g_y}{g_x}. \tag{4}$$

Here $f(x, y)$ is the brightness value of the pixel with coordinates x, y, $\theta(x, y)$ is the direction of the gradient, and $m(x, y)$ is the value of the pixel gradient x, y. Gradients represent places in the image where there is a sharp change in the brightness of pixels. The magnitude of the gradients is greater at the edges and corners of the object, which contain much more information about the object's shape than homogeneous areas. In this way, gradients emphasize the outlines of objects and discard unnecessary information such as a uniform background. From a practical point of view, the found characteristics of the gradient of the original image (magnitude and direction) in each pixel represent the very special points of the object in the image, which can be further analyzed and recognized. In the classical implementation of the HOG algorithm, the considered parameters are converted into a normalized histogram of the entire image, which ultimately forms a feature vector, which is subsequently transmitted for analysis and prediction to various classification algorithms (for example, Support Vector Machine).

2.1. Binary gradient. As you know, working with binary data fundamentally reduces the computational complexity of the algorithm; therefore, a mathematical model was developed for finding the characteristics of a binary gradient by analogy with the classical HOG. The emphasis in the development of the binary HOG algorithm was placed on the implementation of its work on the basis of the Boolean algebra, which also made it possible to reduce the computational complexity of the algorithm and greatly simplify its possible implementation on reconfigurable computing systems. By a binary gradient we mean a change in the brightness of neighboring image pixels from 0 to 1 or vice versa. The value of the gradient m and its direction ϕ for each pixel of the binary image will be determined in accordance with the following accepted rules:

1) $\phi \in \{0°, 45°, 90°\}$;

2) $m \in \{0, 1\}$ depending on whether there is a change in the brightness of the pixels between the current pixel and the pixels of the neighbors. If there is a change, then $m = 1$, if not, $m = 0$;
3) There are 4 possible options for determining the characteristics of the gradient for the considered pixel I in relation to the neighboring pixels x and y.

The result of determining the binary gradient is shown in Fig. 1, where Fig. 1a - original binary image, Fig. 1b - visualization of a binary gradient in each pixel of the image.

Fig. 1. Visualization of the binary gradient of the entire image

2.2. Boolean mathematical model and calculating the gradient of a binary image. Conceptual solutions and architectural features of reconfigurable computing system make it possible to implement at the hardware level various algorithms in the field of spatial image processing, due to which high performance indicators are achieved. This can be achieved by adapting the algorithm for hardware execution on the reconfigurable computing system architecture, the dimension of which in the system's element coincides with the dimension of the processed image in pixels, and each system's element processes the corresponding one pixel in parallel. In this case, the entire image processing algorithm is modified into a boolean mathematical model, which can be implemented as a combinational scheme on a reconfigurable computer (for example, Field-Programmable Gate Arrays, FPGA) almost instantly. The boolean mathematical model is described using the following system of equations:

$$\begin{cases} f_x = f_y = I \\ m = (I * \overline{x} \oplus \overline{I} * x) \oplus (I * \overline{y} \oplus \overline{I} * y) \\ \phi_0 = I * \overline{y} \oplus \overline{I} * y \\ \phi_1 = (I * \overline{x} \oplus \overline{I} * x) * \overline{(I * \overline{y} \oplus \overline{I} * y)} \oplus \overline{(I * \overline{x} \oplus \overline{I} * x)} * (I * \overline{y} \oplus \overline{I} * y) \end{cases} \tag{5}$$

Here, x, y are the input values of the brightness of the neighboring pixels, f_x, f_y are the output values of the brightness of the current pixel for transmitting information to the neighbors, I is the input value of the brightness of the current pixel, f is the output value of the brightness of the pixel, and m is the value of the gradient. The gradient direction value was encoded with a two-digit binary number $\phi = \phi_1 * \phi_0$.

3. Neural network architecture

In a given experiment, the implementation of all functions, building the architecture of the neural network model, the training process and the testing were performed using Tensorflow v2.4.1, an open source end-to-end platform for machine learning. In Fig. 2 shows the constructed architecture of the neural network.

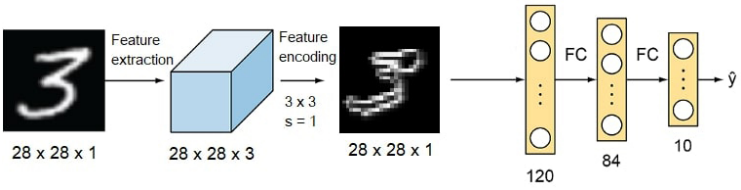


Fig. 2. Neural network architecture

It should be noted that we do not use Dropout or weight regularization in fully connected layers.

4. Experiment

Our main goal was to test the applicability of our binary image feature extraction method in image classification problems using standard training methods for deep neural networks. To test the strength of our method, we applied it to the MNIST classification problem. To do this, it was decided to compare our model with one of the first and well known convolutional neural network model - LeNet-5 [7], originally created to work with small 32x32 grayscale images.

MNIST Dataset [8] was downloaded from the Tensorflow Datasets collection and divided into three parts: train set, evaluation set and test set. LeNet-5 and our (Fig.2) models were trained with following parameters shown in Table 1.

4.1. Experiment evaluation. Fig. 4a and 4b show the graphs of the accuracy and loss of our proposed model after the completion of training.

In Fig. 4b, we can observe a relatively high error rate. However, this behavior of the model can be explained by the absence of any regularization of the weights or activation functions, as well as due to the significant compression of information due to the binarization of the data.

Parameter	Values
$LearningRate$	0.001
$Trainset/Evaluationset/Testset$	60000/9900/100
$Epoch$	10
$BatchSize$	32
$Optimizer$	Adam
$Lossfunction$	Categorical Crossentropy

Table 1. Training parameters

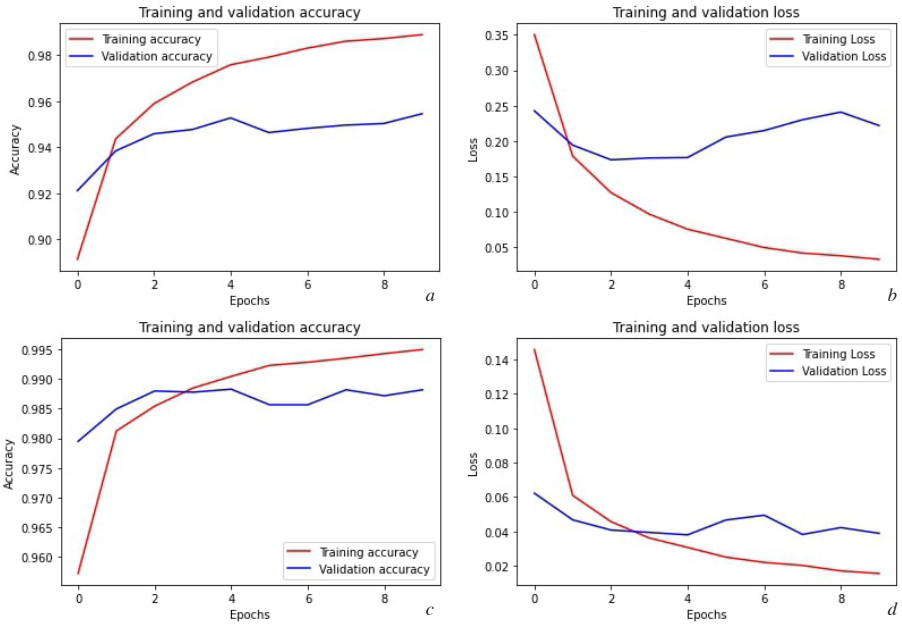


Fig. 3. Training and validation accuracy

Fig. 4c and 4d show the graphs of the accuracy and loss of the LeNet-5 model after the completion of training.

5. Conclusion

The proposed method for binary data on the extraction of binary features of the image showed similar results in comparison with the classical LeNet-5 architecture, the method described above helps us to perform parallel computations much more efficiently and transfer them to the reconfigurable system architecture. It is worth

noting that using this approach eliminates the use of convolutional layers. The model showed 95.13% accuracy on the evaluation set and 2% prediction errors on the test set without using data augmentation, weight regularization, or other algorithms to improve accuracy and optimization. Also, this approach does not negate the possibility of further application of existing optimization and quantization methods for fully connected layers. Thus, this article shows the practical applicability of our approach in classification problems using the classical method of training neural networks.

REFERENCES

1. A. Shrestha and A. Mahmood, "Review of Deep Learning Algorithms and Architectures," in IEEE Access, vol. 7, pp. 53040-53065, 2019, doi: 10.1109/ACCESS.2019.2912200.
2. Bacchus P., Stewart R., Komendantskaya E. (2020) Accuracy, Training Time and Hardware Efficiency Trade-Offs for Quantized Neural Networks on FPGAs. In: Rincón F., Barba J., So H., Diniz P., Caba J. (eds) Applied Reconfigurable Computing. Architectures, Tools, and Applications. ARC 2020. Lecture Notes in Computer Science, vol 12083. Springer, Cham. https://doi.org/10.1007/978-3-030-44534-8-10.
3. Guo, Yunhui. "A Survey on Methods and Theories of Quantized Neural Networks," December 2018, ArXiv:1808.04752v2.
4. Cheng, Y., Wang, D., Zhou, P., Zhang, T, "A Survey of Model Compression and Acceleration for Deep Neural Networks," 2017 arXiv:1710.09282v9.
5. Hubara, Itay et al. "Quantized Neural Networks: Training Neural Networks with Low Precision Weights and Activations, " 2017, arXiv:1609.07061.
6. Rastegari M., Ordonez V., Redmon J., Farhadi A. "XNOR-Net: ImageNet Classification Using Binary Convolutional Neural Networks," In: Leibe B., Matas J., Sebe N., Welling M. (eds) Computer Vision – ECCV 2016. ECCV 2016. Lecture Notes in Computer Science, vol 9908, pp. 525-542. Springer, Cham. https://doi.org/10.1007/978-3-319-46493-0_32.
7. Y. Lecun, L. Bottou, Y. Bengio and P. Haffner, "Gradient-based learning applied to document recognition," in Proceedings of the IEEE, vol. 86, no. 11, pp. 2278-2324, Nov. 1998, doi: 10.1109/5.726791.
8. LeCun, Y., Cortes, C. and Burgess, C.J.C., 2012. The MNIST Database of handwritten images.

УДК: 004.7:519.872

Использование машинного обучения для исследования систем поллинга с коррелированными входными потоками

В.М.Вишневский[1], О.В. Семёнова[1], З.Т. Тан [2]

[1]Институт проблем управления им. В.А. Трапезникова РАН
ул. Профсоюзная, д.65, Москва, Россия, 117997

[2]Московский физико-технический институт„ Институтский пер., д. 9, г. Долгопрудный, Московская обл., Россия, 141701

vishn@inbox.ru, olgasmnv@gmail.com, duytan@phystech.edu

Аннотация

В работе исследованы стохастические системы поллинга с использованием машинного обучения. Рассмотрены системы поллинга типа $M/M/1$ и $MAP/M/1$ с циклическим опросом, а также система типа $M/M/1$ с адаптивным циклическим опросом. Для обучения машинной модели системы поллинга типа $M/M/1$ использовались результаты аналитических расчетов, а для других рассмотренных систем, которые не поддаются точному анализу, использовались результаты имитационного моделирования. Приведены численные примеры, показано, что результаты машинного обучения с высокой точностью совпадают с результатами аналитических или имитационных расчетов.

Ключевые слова: машинное обучение, системы поллинга, коррелированный входной поток

1. Введение

Системы поллинга представляют собой системы массового обслуживания с несколькими очередями и общим обслуживающим прибором [1]. Сервер по определенному правилу посещает очереди и обслуживает находящиеся в них заявки. Системы поллинга широко ипользуются для оценки производительности, проектирования и оптимизации структуры телекоммуникационных систем и сетей, транспортных систем и систем управления дорожным движением, производственных систем и систем управления запасами [2]. Несмотря на значительное

Работа выполнена при финансовой поддержке Российского фонда фундаментальных исследований, проект №19-29-06043.

число работ в этой области, остается большое число нерешенных задач, в частности исследование систем с коррелированными входными потоками или системы с ограниченными дисциплинами обслуживания очередей. Для численного решения подобных задач может быть использован созданный авторами настоящей работы аналитико-имитационный программный комплекс [3], охватывающий широкий класс моделей поллинга, применяемых для исследования широкополосных беспроводных сетей.

В данной работе для расчета характеристик систем поллинга предлагается применять метод машинного обучения с использованием искусственных нейронных сетей. Данная область исследований является новой и, как показывают результаты расчетов, открывает новые возможности для исследований моделей массового обслуживания, не поддающихся или с трудом поддающихся анализу в рамках теории случайных процессов. Заметим, что в литературе известно лишь несколько работ в данном направлении [5, 6]. Представим далее результаты машинного обучения для систем поллинга типа $M/M/1$ и $MAP/M/1$ с циклическим опросом, а также системы типа $M/M/1$ с адаптивным циклическим опросом. Для обучения машинной модели системы поллинга типа $M/M/1$ использовались результаты аналитических расчетов, а для других рассмотренных систем, которые не поддаются точному анализу, использовались результаты имитационного моделирования. Приведены численные примеры, показано, что результаты машинного обучения с высокой точностью совпадают с результатами аналитических или имитационных расчетов.

Искусственная нейронная сеть (Artificial neural networks – ANN), или просто нейронная сеть, может быть определена как вычислительная модель, которая состоит из сетевой архитектуры, состоящей из искусственных нейронов. Структура нейронной сети представляет собой набор параметров (весов), которые необходимо определить или в дальнейшем настроить уже имеющийся набор весов для решения неободимых задач. Искусственные нейронные сети были впервые разработаны в начале 1940-х годов. ANN – это инструменты прогнозирования, используемые для построения математической модели сложной системы. Многослойное восприятие (MLP – Multilayer perceptron) ANN [4] является наиболее известным классом ANN. ANN MLP обычно имеют архитектуру прямой связи и обычно обучаются алгоритмам обратного распространения. Сети MLP состоят из одного входного уровня и одного выходного уровня, по крайней мере, с одним дополнительным скрытым уровнем. Для задачи подбора функции по исходным данным [7] один скрытый слой позволяет нейронной сети аппроксимировать любую функцию, представляющую собой непрерывное отображение из одного конечного пространства в другое. С двумя скрытыми слоями сеть может представлять произвольную границу решения с произвольной точностью [8]. В

работе [9] показано, что оптимальное число нейронов скрытого слоя обычно находится между числом входных данных и числом выходных данных. Поиск оптимального числа нейронов для скрытого слоя определяется эспериментальным путем с целью минимизации ошибки. В данной работе 10 нейронов для каждого скрытого слоя помогает получить достаточно хороший результат с минимальной ошибкой.

Некоторые свойства нейронных сетей. Нейронные сети широко применяются во многих отраслях, например, в розничной продаже, инжиниринге, на производстве, в банковской сфере, страховании, здравоохранении и др. Нейронные сети используются для обнаружения взаимосвязей, распознавания закономерностей и прогнозирования.

Большинство моделей нейронных сетей относятся к следующим типам:

- *Аппроксимация* (или функция регрессии). Аппроксимацию можно рассматривать как задачу подбора функции по исходным данным. При этом нейронная сеть обучается на информации, представленной набором данных, состоящих из выборки с входными и целевыми данными, а выходные данные являются результатом работы нейронной сети, построенной после ее обучения на выборке с целевыми данными.
- Классификация (или распознавание образов). Классификация может быть определена как процесс, в соответствии с которым принятый шаблон, характеризуемый набором признаков, присваивается одному из предписанного числа классов. Входные данные включают набор характеристик, которые характеризуют шаблон. Цели определяют класс, к которому принадлежит каждый шаблон. Основная цель задачи классификации состоит в моделировании апостериорных вероятностей принадлежности к классу, обусловленных входными данными.

Процесс обучения применяется к нейронной сети для получения минимально возможных ошибок. Это делается путем поиска набора параметров, которые соответствуют нейронной сети для набора данных. Общий процесс обучения состоит из двух разных концепций: минимизация ошибок и алгоритм оптимизации.

В качестве ошибки может выступать:

- Mean squared error (MSE) – средняя квадратическая ошибка;
- Normalized squared error (NSE) – нормализованная квадратическая ошибка;
- Weighted squared error (WSE) – взвешенная квадратическая ошибка;
- Cross entropy error – ошибка перекрестной энтропии;
- Minkowski error (ME) – ошибка Минковского.

Алгоритм оптимизации может быть выбран из следующего списка:

- Gradient descent (GD) – градиентный спуск. Это самый простой алгоритм оптимизации. С помощью этого метода параметры обновляются в каждый раз в направлении отрицательного градиента индекса ошибки;
- Conjugate gradient (CG) – сопряженный градиент. В алгоритме сопряженного градиента поиск выполняется вдоль сопряженных направлений, что обычно приводит к более быстрой сходимости, чем в направлениях градиентного спуска;
- Quasi-Newton method (QNM) – метод Ньютона. Этот метод использует Гессиан функции ошибки, которая является матрицей вторых производных, для вычисления направления обучения. Поскольку он использует информацию высокого порядка, направление обучения указывает на минимум функции ошибки с более высокой точностью;
- Levenberg-Marquardt algorithm (LM) – алгоритм Левенберга-Марквардта. Алгоритм предназначен для приближения к скорости обучения второго порядка без необходимости вычисления матрицы Гессиана.
- Stochastic gradient descent (SGD) – стохастический градиентный спуск. Алгоритм имеет иную природу, чем описанные выше алгоритмы. В каждый раз он многократно обновляет параметры на основе груповых данных.
- Adaptative linear momentum (ADAM) – Адаптивный линейный импульс: Этот алгоритм похож на градиентный спуск, но реализует более сложный метод для расчета направления обучения, который обычно обеспечивает более быструю сходимость.

Алгоритм оптимизации останавливается, когда выполняется заданны критерий остановки. Некоторые обычно используемые критерии остановки:

- Норма приращения параметров меньше минимального значения.
- Улучшение ошибки за одну попытку меньше установленного значения.
- Ошибки сведены к цели.
- Норма градиента индекса ошибки устанавливается ниже цели.
- Максимальное количество попыток достигнуто.
- Максимальное количество вычислительного времени было превышено.
- Ошибка в подмножестве выбора увеличивается в течение нескольких попыток.

Процесс обучения нейронной сети с заданным количеством скрытых уровней и нейронов или модели глубокого обучения происходит в шесть этапов:

- Шаг 1: Инициализация: всем нейронам присваиваются начальные веса;
- Шаг 2: Прямое распространение: входные данные из обучающего набора передаются через нейронную сеть и производится расчет выходных данных;

- Шаг 3: Функция ошибки: функция ошибки фиксирует погрешность между известными выходными данными и данными, которые необходимо получить в помощью построенной модели нейронной сети с учетом веса текущей модели (иными словами, «насколько далека» модель от известного результата);
- Шаг 4: Обратное распространение: цель обратного распространения состоит в том, чтобы изменить вес нейронов, чтобы уменьшить ошибку;
- Шаг 5: Обновление веса: веса изменяются на оптимальные значения в соответствии с результатами алгоритма обратного распространения;
- Шаг 6: Итерирование до сходимости: поскольку веса обновляются маленьким дельта-шагом за один раз, для обучения сети требуется несколько итераций. После каждой итерации сила градиентного спуска обновляет веса в сторону уменьшения значения функции глобальных потерь. Количество итераций, необходимых для сходимости, зависит от скорости обучения, мета-параметров сети и используемого метода оптимизации.

Суть этого метода заключается в том, что имеется набор входных и выходных данных. Программа построения нейронной сети по определеному алгоритму выбирает набор весов таким образом, чтобы, комбинируя эти веса с входными данными, получить выходные данные.

В данной работе для исследования систем поллинга использован алгоритм оптимизации Levenberg-Marquardt algorithm (LM) – алгоритм Левенберга-Марквардта, поскольку такой алгоритм позволяет проводить обучение нейронной сети наиболее точно и быстрее, чем другие алгоритмы. При этом в качестве ошибки принимается средняя квадратическая ошибка (MSE).

2. Машинное обучение для несимметричной системы поллинга с циклическим опросом и шлюзовой дисциплиной обслуживания

Рассматриваемая система поллинга имеет N ($N \geq 2$) очередей с неограниченным числом мест для ожидания и один обслуживающий прибор (сервер).

В i-ю очередь поступает простейший поток заявок с параметром λ_i. Времена обслуживания заявок в очереди i независимы и одинаково распределены с функцией распределения $B_i(t)$ со средним b_i и вторым моментом $b_i^{(2)}$. Время переключения сервера между очередями ($(i-1)$-й и i-й) имеет функцию распределения $S_i(t)$ со средним s_i и вторым моментом $s_i^{(2)}$, $i = \overline{1, N}$.

Сервер посещает очереди циклически (последовательно, от первой до N-й, затем вновь возвращаясь к первой очереди). После подключения сервера к очереди начинается обслуживание ее заявок согласно шлюзовой дисциплине, то есть обслуживаются лишь те заявки, которые находились в очереди в момент завершения переключения сервера к данной очереди (момент опроса). Заявки,

поступающие в течение данного периода обслуживания очереди должны ждать следующего ее опроса сервером. Время, которое сервер затрачивает на обслуживание всех очередей от первой до последней и на переключение между ними, называется циклом и в данном случае имеет следующее среднее:

$$C = \frac{\sum_{i=1}^{N} s_i}{1 - \sum_{i=1}^{N} \rho_i},$$

где $\rho_i = \lambda_i b_i$.

Для данной модели будем проводить обучение нейронной сети с целью нахождения средних времен пребывания в очередях системы. Формулы для расчета этих характеристик получены в [10] и имеют следующий вид:

$$V_i = \frac{f_i(i,i)(1+\rho_i)}{2\lambda_i^2 C} + b_i, i = \overline{1,N}, \tag{1}$$

где величины $f_i(j,k)$, $i,j,k = \overline{1,N}$ определяют вторые моменты числа заявок в очередях в моменты опроса сервером очередей и вычисляются как решение системы линейных алгебраических уравнений

$$\begin{aligned} f_{i+1}(j,k) &= \lambda_j\lambda_k s_{i+1}^{(2)} + s_{i+1}\lambda_k f_i(j) + s_{i+1}\lambda_j f_i(k) + f_i(i)\lambda_j\lambda_k[2b_i s_{i+1} + b_i^{(2)}] + \\ &+ f_i(i,i)\lambda_j\lambda_k b_i^2 + f_i(i,j)b_i\lambda_k + f_i(i,k)b_i\lambda_j + f_i(j,k), i \neq j, i \neq k, \\ f_{i+1}(i,j) &= \lambda_i\lambda_j s_{i+1}^{(2)} + s_{i+1}\lambda_i f_i(j) + f_i(i)\lambda_i\lambda_j[2b_i s_{i+1} + b_i^{(2)}] + f_i(i,j)b_i\lambda_i + \\ &+ \lambda_i\lambda_j b_i^2 f_i(i,i), \qquad i \neq j, \\ f_{i+1}(i,i) &= \lambda_i^2 s_{i+1}^{(2)} + f_i(i)\lambda_i^2[2b_i s_{i+1} + b_i^{(2)}] + \lambda_i^2 b_i^2 f_i(i,i), i,j,k = \overline{1,N}. \end{aligned} \tag{2}$$

В качестве примера рассмотрим систему типа $M/M/1$ с $N = 5$ очередями и экспоненциально распределенными временами обслуживания и переключения сервера, которая имеет $3N + 1$ входных параметров и N выходных. Модель нейронной сети для такой системы поллинга содержит два скрытых уровня, каждый из которых состоит из 10 нейронов (см. рис. 1). Для построения нейронной сети использована выборка из примерно 500 входных и целевых данных.

На рис. 2 представлен график зависимости среднего взвешенного времени пребывания заявок в системе $V = \sum_{i=1}^{N} \rho_i V_i$, вычисленного аналитически и полученного в результате машинного обучения. Заметим, что результаты указанных способов вычисления характеристик системы расходятся не более чем на 0,04%. Время обучения не превышает 3 минут, а расчет характеристик для одного набора данных занимает не более секунды.

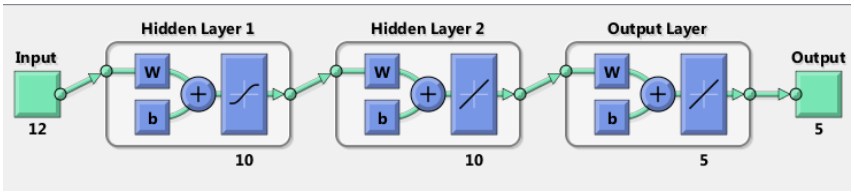


Рис. 1. Схема нейронной сети для системы поллинга типа $M/M/1$ с 5 очередями.

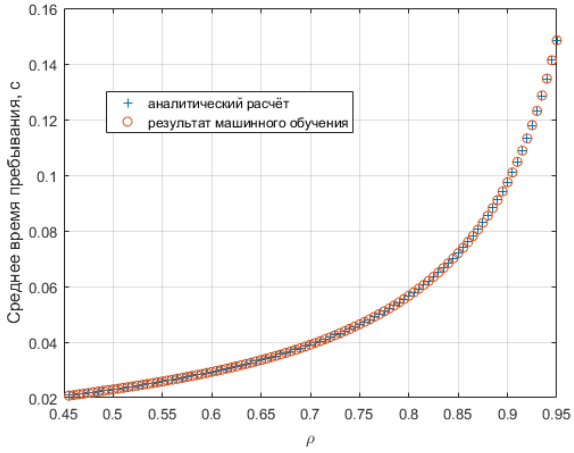


Рис. 2. Результаты машинного обучения для системы поллинга типа M/M/1

3. Машинное обучение для системы поллинга типа $MAP/M/1$ с циклическим опросом и шлюзовой дисциплиной обслуживания

В данном раделе рассмотрим систему поллинга с коррелированными входными потоками типа MAP. Входной поток в i-ю очередь описывается матрицами D_0 и D_1, задающими интенсивности переходов управляющей цепи Маркова, которые, соответственно, не сопровождаются и сопровождаются поступлением заявки в очередь [11]. К настоящему моменту анализ такой системы поллинга для произвольного числа очередей не представлен в литературе, поэтому в качестве данных для машинного обучения будем использовать результаты имитационного моделирования, полученные с помощью пакета прикладных программ оценки характеристик систем стохастического поллинга [3]. Такая модель с N очередями и W_i состояниями MAP-потока в i-ю очередь имеет $\sum_{i=1}^{N} W_i(2W_i - 1) + 2N + 1$ входных параметров.

Для примера рассмотрим систему с четырьмя очередями, для построения нейронной сети использована выборка из примерно 400 входных и целевых

данных. В i-ю очередь поступает MAP-поток заявок [11]. Обозначим этот поток через MAP_i.

Для обучения нейронной сети используем следующий набор входных данных:

MAP_1 характеризуется матрицами

$$D_0 = t_1 \times \begin{bmatrix} -6 & 2 \\ 0 & -1 \end{bmatrix}, D_1 = t_1 \times \begin{bmatrix} 0 & 4 \\ 0 & 1 \end{bmatrix}$$

и коэффициентом корреляции $c_1 = 0$, где t_1 меняется от 0.5 до 50 с шагом 0.5.

MAP_2 – матрицами

$$D_0 = t_2 \times \begin{bmatrix} -3 & 0 \\ 0 & -0.6 \end{bmatrix}, D_1 = t_2 \times \begin{bmatrix} 1 & 1 \\ 0.2 & 0.4 \end{bmatrix}$$

и коэффициентом корреляции $c_2 = 0,07843$, где t_2 меняется от 0.5 до 50 с шагом 0.5.

MAP_3 – матрицами

$$D_0 = t_3 \times \begin{bmatrix} -1.875 & 0.0625 \\ 0.0625 & -0.25 \end{bmatrix}, D_1 = t_3 \times \begin{bmatrix} 1.8125 & 0 \\ 0 & 0.1875 \end{bmatrix}$$

и коэффициентом корреляции $c_3 = 0,2704$, где t_3 меняется от 0.5 до 50 с шагом 0.5.

MAP_4 – матрицами

$$D_0 = t_4 \times \begin{bmatrix} -6 & 2 \\ 0 & -1 \end{bmatrix}, D_1 = t_4 \times \begin{bmatrix} 0 & 4 \\ 0 & 1 \end{bmatrix}$$

и коэффициентом корреляции $c_1 = 0$, где t_4 меняется от 0.5 до 50 с шагом 0.5. Интенсивность обслуживания для каждой очереди – $0,001$, интенсивность переключения сервера между очередями – $0,001$.

Целевые данные представляют собой среднее время пребывания заявок в каждой очереди. Средние времена пребывания для обучения нейронной сети рассчитаны с помощью пакета прикладных программ оценки характеристик систем стохастического поллинга [3]. Структура сети представлена следующим образом:

- Входной слой, куда подаются 38 входных данных, имеет 38 нейронов;
- 1 скрытый слой, состоящий из 10 нейронов;
- Выходной слой, имеющий 4 нейрона, который определяет результат работы нейронной сети.

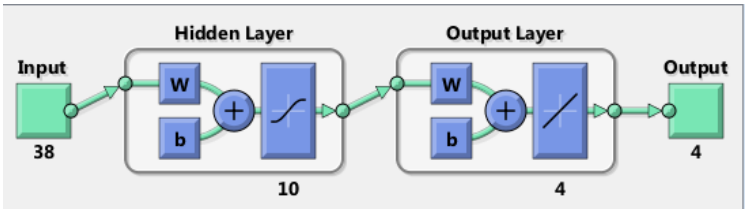


Рис. 3. Схема нейронной сети для обучения систем поллинга типа MAP/M/1

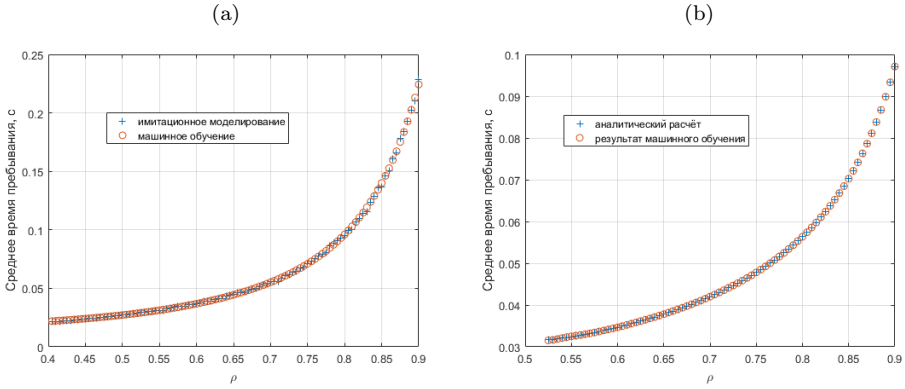


Рис. 4. Результаты машинного обучения систем поллинга типа $MAP/M/1$ (a) и $M/M/1$ (b) с адаптивным опросом и исчерпывающим обслуживанием

Здесь и далее число нейронов во входном и выходном слоях определяются числом входных и выходных параметров, сеть также имеет один скрытый слой с 10 нейронами.

На рис. 4 (a) представлены результаты расчетов среднего времени пребывания заявок, полученные с помощью имитационного моделирования и машинного обучения. В данном случае относительное расхождение результатов не превысило 1%.

4. Машинное обучение для системы поллинга типа $M/M/1$ с адаптивным опросом

Рассмотрим теперь систему поллинга с адаптивным циклическим порядком опроса, при котором сервер опрашивает очереди циклически, но пропускает (не опрашивает) те из них, которые были опрошены в предыдущем цикле и при этом оказались пусты в момент их опроса. Все очереди, которые пропускает сервер в данном цикле, будут опрошены в следующем цикле. Такие системы исследованы работах [12, 13] в случае шлюзовой и исчерпывающей дисциплин обслуживания

очередей методом производящих функций на основе построения вложенного марковского процесса. Используя обучающие данные, полученнные с помощью формул для вычисления средних времен пребывания в очередях системы [13], построим нейронную сеть и сравним результаты ее работы с аналитическими результатами, полученными на другой выборке входных данных. Полученные данные отображены на рис. 4(a) и 2, расхождение результатов составляет не более 0.04%. Время обучения нейронной сети – не более 3 секунд.

Рассмотрим далее случай ограниченного обслуживания очередей, не поддающегося точному анализу в силу специфики многомерного случайного процесса, описывающего поведение такой системы. При ограниченном обслуживании сервер может обслужить в очереди не более определенного (детерминированного или случайного) числа заявок при одном посещении очереди. Для примера рассмотрим систему с 4 очередями, 1-ограниченным и 2-ограниченным обслуживанием, то есть сервер обслуживает, соответственно, не более одной и двух заявок в очереди. Рис. 4(b) иллюстрирует полученные результаты. Средние времена пребывания для обучения нейронной сети рассчитаны с помощью модуля имитационного моделирования пакета прикладных программ [3]. Как следует из рис. 4(b), что расхождение результатов имитационного моделирования и модели нейронной сети составляет не более 2%. Время обучения не превышает 3 минуты, а для каждого набора входных данных расчет выходных данных требует не более 1 секунды.

(a) (b)

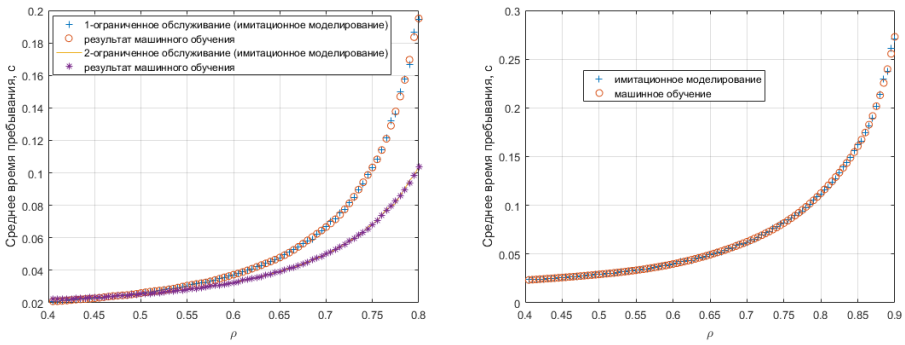


Рис. 5. Результаты машинного обучения систем поллинга типа $M/M/1$ (a) с адаптивным опросом и ограниченным обслуживанием и $MAP/M/1$ (b) с адаптивным опросом и исчерпывающим обслуживанием

5. Машинное обучение для системы поллинга типа $MAP/M/1$ с адаптивным опросом

Рассмотрим также пример системы поллинга с 5 очередями, адаптивным циклическим опросом и коррелированными входными потоками типа MAP. Обслуживание очередей – исчерпывающее. MAP-потоков в очереди системы (MAP_i для i-й очереди) описываются следующим образом.

MAP_1 характеризуется матрицами

$$D_0 = t_1 \times \begin{bmatrix} -6 & 2 \\ 0 & -1 \end{bmatrix}, D_1 = t_1 \times \begin{bmatrix} 0 & 4 \\ 0 & 1 \end{bmatrix}$$

и коэффициентом корреляции $c_1 = 0$, где t_1 меняется от $0,5$ до 50 с шагом $0,5$.

MAP_2 – матрицами

$$D_0 = t_2 \times \begin{bmatrix} -3 & 0 \\ 0 & -0.6 \end{bmatrix}, D_1 = t_2 \times \begin{bmatrix} 1 & 1 \\ 0.2 & 0.4 \end{bmatrix}$$

и коэффициентом корреляции $c_2 = 0,07843$, где t_2 меняется от $0,5$ до 50 с шагом $0,5$.

MAP_3 – матрицами

$$D_0 = t_3 \times \begin{bmatrix} -1.875 & 0.0625 \\ 0.0625 & -0.25 \end{bmatrix}, D_1 = t_3 \times \begin{bmatrix} 1.8125 & 0 \\ 0 & 0.1875 \end{bmatrix}$$

и коэффициентом корреляции $c_3 = 0,2704$, где t_3 меняется от $0,5$ до 50 с шагом $0,5$.

Для данной модели имеем 29 входных данных. Такая модель поллинга на данный момент не позволяет провести точный расчет ее характеристик, поэтому для в качестве данных для машинного обучения используем модуль имитационного моделирования пакета прикладных программ [3]. Сравнительный анализ представлен на рис. 5(b).

6. Заключение

В работе представлены результаты машинного обучения для систем поллинга с простейшими и коррелированными входными потоками. Для обучения использовались аналитические результаты расчетов характеристик систем, а в случае коррелированных входных потоков – результаты имитационного моделирования. По результатам обучения проведены контрольные сравнения характеристик, полученных аналитически (имитационно) и на основе машинного обучения. Показано, что полученные результаты совпадают с высокой точностью, при этом машинная модель позволяет значительно сократить время расчета характеристик систем поллинга по сравнению с имитационным моделированием.

Литература

1. Vishnevsky V., Semenova O. Polling systems and their application to telecommunication networks // Mathematics. January 2021. Vol. 9, No. 2. 117
2. Boon M.A.A., van der Mei R.D., Winands E.M.M. Applications of polling systems // Surveys in Operations Research and Management Science. 2011. Vol. 16, No. 2. P. 67-82.
3. Вишневский В.М., Семёнова О.В., Буй З.Т. Программный комплекс оценки характеристик систем стохастического поллинга: Свидетельство о государственной регистрации программы для ЭВМ № 2019614554 РФ; Зарег. 08.04.2019.
4. Cybenko J. Approximations by superpositions of a sigmoidal function // Mathematics of control, signals and systems. 1989. Vol. 2, No. 4. P. 303-314.
5. Sivakami Sundari M., Palaniammal S. Simulation of $M/M/1$ queuing system using ANN // Malaya Journal of Matematik. 2015. Vol. 1. P. 279-294.
6. Sivakami Sundari M., Palaniammal S. An ANN simulation of single server with infinite capacity queuing system // International Journal of Innovative Technology and Exploring Engineering. 2019. Vol. 8, No. 12.
7. Csáji B.C. Approximation with artificial neural networks // Faculty of Sciences, Eötvös Loránd University, Hungary. 2001.
8. Thomas A., Petridis M., Walters S., Malekshahi S., Morgan R. Two hidden layers are usually better than one // International Conference on Engineering Applications of Neural Networks. 2017. DOI 10.1007/978-3-319-65172-9_24.
9. Heaton J. Introduction to neural networks with Java. Heaton Research, Inc., 2008.
10. Yechiali U. Analysis and control of polling systems // SIGMETRICS 1993: Performance Evaluation of Computer and Communication Systems. P. 630-650.
11. Вишневский В.М., Дудин А.Н., Клименок В.И. Стохастические системы с коррелированными потоками. Теория и применение в телекоммуникационных сетях, М.: Техносфера, 2018, 564 с.
12. Vishnevsky V.M., Dudin A.N., et. al. Approximate method to study $M/G/1$-type polling system with adaptive polling mechanism // Quality Technology and Quantitative Management. 2012. Vol. 9, No. 2. P. 211-228.
13. Vishnevsky V.M., Semenova O.V., Bui D.T., Sokolov A. 2019. Adaptive cyclic polling systems: analysis and application to the broadband wireless networks // Proceedings of the 22nd International Conference on Distributed Computer and Communication Networks: Control, Computation, Communications (DCCN-2019, Moscow). Cham: Springer. 11965. P. 30-42.

UDC: 519.872

The two-dimensional Output Process of Retrial Queue with Two-Way Communication and MMPP input

A.L. Blaginin, I.L. Lapatin

alex-b.l@yandex.ru, ilapation@mail.ru

Abstract

In this paper, we review a retrial queue with MMPP input and two-way communication. Incoming calls, arriving at the server and finding it busy, join orbit and try to enter the server again after some exponentially distributed time. While idle, the server makes outgoing calls and serves them for exponentially distributed time with another intensity. MMPP (Markov Modulated Poisson Process) is an input process, in which control is driven by a continuous Markov chain. Changing its state entails a change of the intensity of the arrival process. For this model we present an asymptotic approximation of the two-dimensional characteristic function under the condition of a large delay of requests in the orbit. For this approximation we carried out a numerical experiment, where asymptotic results were compared to computations, which were obtained via simulation.

Keywords: output process, retrial queue, two-way communication, asymptotic analysis method, simulation, markov modulated poisson process

1. Introduction

The special property of RQ-systems [1, 2] with two-way communication [1] is presence of different types of requests, which gives rise to many new disciplines of service. For this reason RQ-systems with two-way communication are a powerful tool in the design and optimization of real-life systems with multiple random access to a resource. Despite the fact, that these systems are well studied, their output process is still a complex and insufficiently explored area to research.

In modern telecommunication networks there are also point processes with a varying rate of calls incoming. To simulate such processes within the framework of queuing theory, the Markov Modulation Poisson Process (MMPP) [3, 2] is used. It has a mechanism for taking into account the temporal inhomogeneity of the arrival rate of requests and also gives analytically processable queuing results [4]. For this reason, MMPP is widely used in Internet research, in particular, using MMPP in [5],

a traffic model, that accurately approximates the LRD (Long Range Dependence) characteristics of Internet traffic traces, was built. Using the concepts of sessions and streams, the proposed MMPP model simulates the real hierarchical behavior of the process of generating packets by Internet users. It allows to generate traffic with the desired characteristics with the ability to set several input parameters that have a clear physical meaning. The results prove that the queuing traffic behavior generated by the MMPP model is consistent with the model created by the real traces of packets collected at the edge router under various scenarios and load.

In this paper, we find the approximation of the characteristic function of the number of served requests in the considered system using the method of asymptotic analysis. Subsequently, we determine the applicability of the asymptotic results by comparing them to calculations, provided with simulation software, which was designed specially for this research.

2. Mathematical model

MMPP is defined by two matrices. Matrix of infinitesimal characteristics $\boldsymbol{Q}$ defines the state. Value q_{ij} determines the intensity of the transition of the process from the state i to the state j, and the value $-q_{ii}$ is the intensity of leaving the state i. The matrix $\boldsymbol{Q}$ has property $\sum_j q_{ij} = 0$. The diagonal matrix $\boldsymbol{\Lambda}$ specifies the rate of calls for each of the states of the process.

Let us consider the RQ-system with MMPP input. Incoming request, entering the system and finding the server free, takes him. The server, in turn, begins serving it for some random time, which is distributed exponentially with parameter μ_1. If upon entering the system the request finds the server busy, it instantly goes to the orbit, where carries out a random delay during an exponentially distributed time with parameter σ. In its free time from serving incoming requests, the server produces requests itself with intensity α and serves them for exponentially distributed time with parameter μ_2.

Let us denote following notations: *i(t)* — the number of requests in the orbit at the moment t, *k(t)* – state of the server: *0* – the server is free, *1* – the server is busy, *2* – the server is busy serving retrial request; $m_1(t)$ — the number of served requests from input process at the moment t, $m_2(t)$ — the number of served outgoing requests at the moment t, $n(t)$ - the state of input process at the moment t.

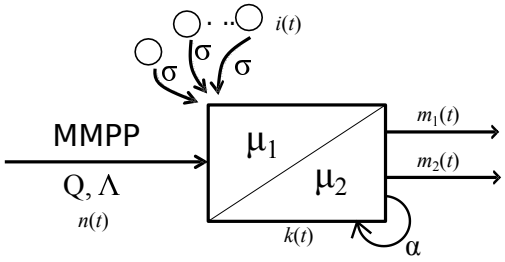


Fig. 1. System model

3. Kolmogorov equations

Let us consider five-dimensional Markov process

$$\{k(t), n(t), i(t), m_1(t), m_2(t)\}$$

Based on formulated markov process we introduce probabilities

$$P\{k(t) = k, n(t) = n, i(t) = i, m_1(t) = m_1, m_2(t) = m_2\}$$

and write down for them a system of Kolmogorov differential equations

$$\begin{aligned}
\frac{\partial P_0(n,i,m_1,m_2,t)}{\partial t} &= -(\lambda_n + i\sigma + \alpha)P_0(n,i,m_1,m_2,t) + \\
&+ P_1(n,i,m_1-1,m_2,t)\mu_1 + P_2(n,i,m_1,m_2-1,t)\mu_2 + \\
&+ \sum_{v=1}^{N} P_0(v,i,m_1,m_2,t)q_{vn}, \\
\frac{\partial P_1(n,i,m_1,m_2,t)}{\partial t} &= -(\lambda_n + \mu_1)P_1(n,i,m_1,m_2,t) + \\
&+ (i+1)\sigma P_0(n,i+1,m_1,m_2,t) + \lambda_n P_0(i,m_1,m_2,t) + \\
&+ \sum_{v=1}^{N} P_1(v,i,m_1,m_2,t)q_{vn}, \\
\frac{\partial P_2(n,i,m_1,m_2,t)}{\partial t} &= -(\lambda_n + \mu_2)P_2(n,i,m_1,m_2,t) + \\
&+ \lambda_n P_2(n,i-1,m_1,m_2,t) + \alpha P_0(n,i,m_1,m_2,t) + \\
&+ \sum_{v=1}^{N} P_2(v,i,m_1,m_2,t)q_{vn}.
\end{aligned} \tag{1}$$

It is not possible to solve provided equations analytically since it is a system of an infinite number of differential finite-difference equations with variable coefficients. In

order to pass to a finite number of equations, we introduce the partial characteristic functions, denoting $j^2 = -1$,

$$H_k(n, u, u_1, u_2, t) = \sum_{i=0}^{\infty} \sum_{m_1=0}^{\infty} \sum_{m_2=0}^{\infty} e^{jui} e^{ju_1 m_1} e^{ju_2 m_2} P_k(n, i, m_1, m_2, t).$$

For further analysis let us denote

$$\boldsymbol{H}_k(u, u_1, u_2, t) = \{H_k(1, u, u_1, u_2, t), H_k(2, u, u_1, u_2, t), \ldots, H_k(N, u, u_1, u_2, t)\},$$

diagonal unit matrix $\boldsymbol{I}$ with size N.

4. Method of asymptotic analysis

Resulting system of differential equations will be solved by the method of asymptotic analysis in the limit condition of a large delay of requests in the orbit ($\sigma \to 0$).

Denoting $\epsilon = \sigma, u = \epsilon w, \boldsymbol{F}_k(w, u_1, u_2, t, \epsilon) = \boldsymbol{H}_k(u, u_1, u_2, t)$, the system will be written as

$$\begin{aligned}
\frac{\partial \boldsymbol{F}_0(w, u_1, u_2, t, \epsilon)}{\partial t} &= (\boldsymbol{Q} - \boldsymbol{\Lambda} - \alpha \boldsymbol{I}) \boldsymbol{F}_0(w, u_1, u_2, t, \epsilon) + \\
&+ \mu_1 e^{ju_1} \boldsymbol{F}_1(w, u_1, u_2, t, \epsilon) + \mu_2 e^{ju_2} \boldsymbol{F}_2(w, u_1, u_2, t, \epsilon) + \\
&+ j \frac{\partial \boldsymbol{F}_0(w, u_1, u_2, t, \epsilon)}{\partial w}, \\
\frac{\partial \boldsymbol{F}_1(w, u_1, u_2, t, \epsilon)}{\partial t} &= \boldsymbol{\Lambda} \boldsymbol{F}_0(w, u_1, u_2, t, \epsilon) + \\
&+ (\boldsymbol{Q} + (e^{j\epsilon w} - 1)\boldsymbol{\Lambda} - \boldsymbol{I}\mu_1) \boldsymbol{F}_1(w, u_1, u_2, t, \epsilon) - \\
&- j e^{-j\epsilon w} \frac{\partial \boldsymbol{F}_0(w, u_1, u_2, t, \epsilon)}{\partial w}, \\
\frac{\partial \boldsymbol{F}_2(w, u_1, u_2, t, \epsilon)}{\partial t} &= \alpha \boldsymbol{F}_0(w, u_1, u_2, t, \epsilon) + \\
&+ (\boldsymbol{Q} + (e^{j\epsilon w} - 1)\boldsymbol{\Lambda} - \boldsymbol{I}\mu_2) \boldsymbol{F}_2(w, u_1, u_2, t, \epsilon).
\end{aligned} \tag{2}$$

The solution for system (2) is formulated in theorems 1 and 2.

Theorem 1. Let $i(t)$ is the number of requests in the orbit at the moment t, then in the stationary regime we obtain

$$\lim_{\epsilon \to 0} \{\sum_{k=0}^{2} \boldsymbol{F}_k(w, 0, 0, t, \epsilon)\} = \lim_{\sigma \to 0} M e^{jw\sigma i(t)} = e^{jw\kappa},$$

where κ is the positive root of equation

$$\kappa \boldsymbol{R}_0(\kappa)\boldsymbol{e} = [\boldsymbol{R}_1(\kappa) + \boldsymbol{R}_2(\kappa)]\boldsymbol{\Lambda}\boldsymbol{e}.$$

Vectors $\boldsymbol{R}_k$ are defined as

$$\begin{cases} \boldsymbol{R}_0(\kappa) = \boldsymbol{r}\{\boldsymbol{I} + [\boldsymbol{\Lambda} + \kappa\boldsymbol{I}](\mu_1\boldsymbol{I} - \boldsymbol{Q})^{-1} + \alpha(\mu_2\boldsymbol{I} - \boldsymbol{Q})^{-1}\}^{-1}, \\ \boldsymbol{R}_1(\kappa) = \boldsymbol{R}_0(\kappa)[\boldsymbol{\Lambda} + \kappa\boldsymbol{I}](\mu_1\boldsymbol{I} - \boldsymbol{Q})^{-1}, \\ \boldsymbol{R}_2(\kappa) = \alpha\boldsymbol{R}_0(\kappa)(\mu_2\boldsymbol{I} - \boldsymbol{Q})^{-1}. \end{cases}$$

The row-vector $\boldsymbol{r}$ is the stationary probability distribution of background process $n(t)$, which is obtained as the unique solution for the system $\boldsymbol{rQ} = 0, \boldsymbol{re} = 1$.

Theorem 2. The asymptotic approximation of the two-dimensional characteristic function of the number of served requests of the MMPP input process and the number of served outgoing requests for some time t has the form

$$\lim_{\sigma \to 0} M\{\exp(ju_1 m_1(t)) \exp(ju_2 m_2(t))\} = \lim_{\epsilon \to 0}\{\sum_{k=0}^{2} \boldsymbol{F}_k(0, u_1, u_2, t, \epsilon)\}\boldsymbol{e} = \\ = \boldsymbol{R} \cdot \exp\{\boldsymbol{G}(u_1, u_2)t\}\boldsymbol{ee},$$

where matrix $\boldsymbol{G}(u_1, u_2)$ can be written as

$$\boldsymbol{G}(u_1, u_2) = \begin{bmatrix} \boldsymbol{Q} - \boldsymbol{\Lambda} - (\alpha + \kappa)\boldsymbol{I} & \mu_1 e^{ju_1}\boldsymbol{I} & \mu_2 e^{ju_2}\boldsymbol{I} \\ \boldsymbol{\Lambda} + \kappa\boldsymbol{I} & \boldsymbol{Q} - \mu_1\boldsymbol{I} & 0 \\ \alpha\boldsymbol{I} & 0 & \boldsymbol{Q} - \mu_2\boldsymbol{I} \end{bmatrix}^T,$$

row-vector $\boldsymbol{R} = \{\boldsymbol{R}_0, \boldsymbol{R}_1, \boldsymbol{R}_2\}$ is two-dimensional stationary probability distribution of the process $\{k(t), n(t)\}$, where $\boldsymbol{R}_k$ has dimension N, κ is normalized average number of requests in the orbit, and e and ee are unit vector columns of dimensions N and $N \cdot K$, where K is the number of the server's states.

5. Numerical examples

Let us compare the results of simulation with the calculations, based on the obtained asymptotic approximation of the characteristic function. The value of σ affects the accuracy during comparison, since the solution of the system was obtained under the asymptotic condition of a large delay of requests in the orbit.

We measure accuracy of the results with Kolmogorov-Smirnov distance, which is calculated as

$$\Delta = \max_{0 \le i \le \infty} \left| \sum_{v=0}^{i} (P_0(v) - P_1(v)) \right|,$$

where $P_0(v)$ is and $P_1(v)$ are probability distributions, which are being compared.

Let us set following parameters:

$$\alpha = 0.6, \mu_1 = 2, \mu_2 = 1.5, t = 15,$$

$$\boldsymbol{Q} = \begin{bmatrix} -0.5 & 0.2 & 0.3 \\ 0.15 & -0.2 & 0.05 \\ 0.3 & 0.4 & -0.7 \end{bmatrix}, \boldsymbol{\Lambda} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0.6 & 0 \\ 0 & 0 & 0.7 \end{bmatrix}.$$

The intensity of the input process can be written in form $\boldsymbol{r} \cdot \boldsymbol{\Lambda} \cdot \boldsymbol{e}$, after calculation of which, we get the value 0.72. For the parameters set we obtained following results.

Let us denote: Δ_S — KS distance values for the summary distribution, which implies, that served incoming and outgoing requests are homogeneous, and Δ_{TD} — KS distance values for the two-dimensional distribution of served requests, which are, in two-dimensional case, of different types.

Table 1. KS distance values for various σ

σ	10	1	0.6	0.4	0.2	0.1	0.05	0.01
Δ_S	0.053	0.045	0.04	0.036	0.028	0.023	0.018	0.016
Δ_{TD}	0.059	0.049	0.042	0.035	0.024	0.015	0.01	0.003

In the table 1, we can see, that upon lowering the value of σ approximation gets more accurate. Also, for smaller σ the approximation of the two-dimensional distribution is more accurate than for the summary distribution.

6. Conclusion

Thus, we have obtained a formula for finding the asymptotic approximation of the two-dimensional characteristic function of the number of incoming and outgoing requests that have finished serving in retrial queue with two-way communication under the condition of a large delay in the orbit.

Based on the performed experiments, it can be concluded that the tendency to an increase in the accuracy of asymptotic results is always observed with a decrease in the value of σ, but even with the value of σ set higher than input process intensity, Kolmogorov-Smirnov distance does not go above the value 0.06. Based on this fact, we can conclude, that obtained approximation gives results of high accuracy.

REFERENCES

1. T. Phung-Duc, Retrial queueing models: A survey on theory and applications (2019).

2. I. Lapatin, A. Nazarov, Asymptotic analysis of the output process in retrial queue with markov-modulated poisson input under low rate of retrials condition, in: International Conference on Distributed Computer and Communication Networks, Springer, 2019, pp. 315–324.
3. A. Baiocchi, N. Blefari-Melazzi, Steady-state analysis of the mmpp/g/1/k queue, IEEE transactions on communications 41 (4) (1993) 531–534.
4. K. S. Meier-Hellstern, A fitting algorithm for markov-modulated poisson processes having two arrival rates, European Journal of Operational Research 29 (3) (1987) 370–377.
5. L. Muscariello, M. Meillia, M. Meo, M. A. Marsan, R. L. Cigno, An mmpp-based hierarchical model of internet traffic, in: 2004 IEEE International Conference on Communications (IEEE Cat. No. 04CH37577), Vol. 4, IEEE, 2004, pp. 2143–2147.

UDC: 004.057.4; 004.057.7

An Efficient Cluster Routing Protocol for Vehicular Ad-Hoc Network Using Bio-metaheuristic Algorithm

A.A. Sabbagh [1] and M.V. Shcherbakov[1]

[1]Volgograd State Technical University, Volgograd, Russian Federation

amanisabbagh86@gmail.com, maxim.shcherbakov@gmail.com

Abstract

In this paper, we propose a novel hybrid K-MCSA protocol that combines the features of k-means clustering algorithm and cuckoo search algorithm to establish an efficient, reliable and stable route between the source and destination vehicular nodes in VANET network. In our proposed protocol, k-means clustering and cuckoo search algorithm are used to identify an optimal route among known routes. Further, three weight parameters are used along with the modified cuckoo search algorithm as a fitness function to ensure a stable and reliable route. The simulation is carried out in simulator NS-3 and it demonstrates that the proposed hybrid K-MCSA protocol significantly improves the packet delivery ratio, average delay, packet loss ratio, overhead and throughput while compared to popular routing protocol AODV.

Keywords: K-means clustering, cuckoo search, weight parameters, hybrid approach, and vehicular nodes.

1. Introduction

The substantial growth in the automobile industry led to the exponential growth in wireless ad-hoc networks, specifically, Vehicular Ad-hoc Networks (VANETs). VANETs are constructed by pertaining to the rules of Mobile Ad-hoc Networks (MANETs) [1], [2]. VANETs are generally used in urbanized environments since they promote passenger safety in roads and prevent many accidents. An efficient routing technique of data packets could ensure delivering the warning messages during collisions on-time and they also could avoid a larger number of accidents [3]. Hence, an efficient routing technique that is capable of swiftly delivering the data packets along with lesser packet loss is a demanding need to satisfy the objective. This ensures vehicle security and promotes satisfaction to every user [4], [5]. These networks faced various routing issues such as vehicle mobility continuously leading

to topological modifications [6] , in addition, expansion of networks extensively results in higher routing overheads [7], [8]. There are several techniques available for routing optimization, clustering is an important and effective optimization technique. Clustering can be employed as a significant tool in enhancing the reliability and scalability of routing techniques in VANETs. Hence, in our proposed protocol, k-means clustering is used for cluster formation and a modified cuckoo search algorithm is used for selecting the cluster head. The utilization of a cuckoo search algorithm in this hybrid approach supports in identifying an optimal route among known routes. Further, three weight parameters are used along with the modified cuckoo search algorithm as a fitness function to ensure a stable and reliable route.

2. Proposed routing protocol

2.1. K-means clustering algorithm. Clustering techniques are widely used in a variety of applications namely ad-hoc networks such as MANETs and VANETs and they are also used in data mining approaches. K-means clustering algorithm has been extensively employed since they effectively manage routing problems in ad-hoc networks. Because of the swift convergence and easy implementation, they are generally preferred in VANETs. It is a static and location-based approach. K-means algorithm is an important type of flat clustering, aims to reduce the mean squared Euclidean distance from the centers of clusters [9]. The implementation of k-means algorithm can be established through the following steps:

1) Select C nodes to the space illustrated by the objects which are going to get clustered. Such nodes are picked as the primary centroid group.
2) Assign every object to the set which contains the nearest centroid.
3) Once every object is assigned, compute the C centroid location.
4) Repeat the steps 2 and 3 till the centroids stop their movement. This technique is called object segregation to clusters from which the standard to be reduced has to be done again.

2.2. Cuckoo search algorithm. Cuckoo search algorithm (CSA) is a widely used bio-inspired algorithm proposed by Yang et.al [10]. The algorithm functions on the basis of how the way of the life of cuckoo furnished ideas for various optimization techniques. To upraise the babies of cuckoo, it lays eggs onto the other bird's nest. They take out an egg from the host bird's nest and lay eggs by duplicating the eggs of the host bird. The eggs that have close correspondence with the host egg will have highly probable chances of hatching while the eggs identified by the host bird will be destroyed [11]. In this algorithm, each egg in the nest if the host bord refers to a solution to the capacity of what is calculated using few variables of adaptation function. If a new solution is found to be better than the past one, it automatically

Algorithm 1: K-Means Cluster Formation

Step1: Obtain c number of clusters as input
Step2: Obtain $M = (m_1, m_2, ...m_n) \in L_n$ (is the location of n nodes which are the set of data points)
Step3: for $b = 1: c$ do
Step4: randomly choose($\lambda_1, \lambda_2, ...\lambda_c$)
Step5: end for
Step6: for $b = 1: c$ do
Step7: for $a = 1: n$ do
Step8: determine $\lambda_b = \{\lambda_b || max \sum_{b=1}^{c} || m_a - \lambda_b ||^2\}$
Step9: end for
Step10: end for
Step11: Assign m_a to λ_b
Step12: Once, every data point allocation is complete, re-compute the cluster centroid position.
Step13: Repeat the steps 6 to 10 till all the centroids are found to be convergent.

Fig. 1. K-Means Cluster Formation Algorithm

replaces the old one [10]. According to Yang and Suash [10], the CSA algorithm follows three major rules:

1) Once in a time every cuckoo lays only one egg and places that egg in an arbitrarily selected host bird's nest.
2) The finer nests that carry superior egg quality will be passed to the next generation.
3) The quantity of the host bird's nests is predetermined. The probability of discovery that the host bird identifies cuckoo's eggs is (0,1).

2.3. K-MCSA Proposed Routing algorithm. The proposed algorithm is classified into two phases namely: (a) cluster formation and (b) cluster head (CH) selection. We utilized k-means clustering algorithm for cluster formation and modified cuckoo search algorithm for cluster head selection. The modified CS algorithm chooses the best reliable path through smart configuration of the weights depending on the parameters namely distance factor, angle factor and road factor . The modified CS algorithm discovers the best path in a reasonable time period. The primary advantage of the CS algorithm namely its faster convergence and improved speed of execution makes it further more suitable in route discovery. we consider the smallest weight parameter as the CH,where:

Algorithm 2: Cuckoo Search Algorithm

Cuckoo Search via Lévy Flights
Step1: Generate iteration time $t = 1$
Step2: Objective function f(x), x = (x1, ..., xd)
Step3: Generate initial population of n host nests xi (i = 1, 2, ..., n)
Step4: while (t <MaxGeneration) or (stop criterion)
Step5: Get a cuckoo randomly by Lévy flights
Step6: Evaluate its quality/fitness Fi
Step7: Choose a nest among n (say, j) randomly
Step8: if (Fi < Fj),
Step9: Replace j by the new solution;
Step10: end
Step11: A fraction (pa) of worse nests is abandoned and new ones are built;
Step12: Keep the best solutions (or nests with quality solutions);
Step13: Rank the solutions and find the current best
Step14: Update the generation number $t = t + 1$
Step15: End while
Step16: End

Fig. 2. Cuckoo Search Algorithm

Start procedure **Source node action begins** Set N b current node and N d destination node While N d destination Receive Packet or (stop criterion) Current node sends Hello packet to neighbor nodes; where hello packet contains node information (position, speed, road id, heading) Calculate the weight parameters, Execute the k-means algorithm to find cluster centers and to form cluster group Increment b End while Interrupt for receive RREQ (source address, destination address, hop, seq number, DF, AF, RF) **Destination node action begins** Initialize the size of the population to Get available population of S paths, a = 1, 2……, s	For $a = 1{:}S$ for the declared path of source to destination While a<s or (stop criterion) Calculate fitness value based on $W(a,b,c) = h_1 * DF(a,b,c) + h_2 * AF(b,c) + h_3 * RF(b,c)$ Execute Cuckoo Search algorithm Calculate the fitness Increment a End while Choose Optimized node as CH with best fitness value in that group CH node sends advertisement message to group nodes Nodes which receive the advertisement message will become the cluster member for that CH This process continues throughout in the network **End procedure**

Fig. 3. Proposed Routing Protocol

1) DF(a,b,c) : it is the distance factor between source node a, neighbour node b and destination node c, which is calculated as the mean value of three nodes.

$$distance(a,b) = sqrt[((xb - xa)^2) + ((yb - ya)^2)] \quad (1)$$

$$DF(a,b,c) = (distance(a,b) + distance(b,c) + distance(a,c))/3 \quad (2)$$

2) AF(b,c): it is the angle factor between two nodes which decides if a node b is moving near or moving from the targeted destination c where the angle between two nodes.

$$AF\,(b,c) = \cos(\theta) = \frac{\overrightarrow{V_b} - \overrightarrow{bc}}{||\overrightarrow{V_b}||.||\overrightarrow{bc}||} \quad (3)$$

3) RF(b,c): it is road factor which decides if two nodes on the same road or not.

$$RF\,(b,c) = \begin{cases} 0\ if\ the\ vehicles\ are\ \ on\ the\ same\ road \\ +1\ if\ the\ vehicles\ are\ \ in\ different\ roads \end{cases} \quad (4)$$

3. Performance Evaluation

3.1. Scenario description. The simulation for the proposed model is carried out using Network Simulator (NS-3) . We evaluate the performance and validate the effectiveness of proposed K-MCSA through this simulation. A comparative study on the metrics, with popular protocol namely AODV are also presented in the graphs below. The simulation parameters we considered are stated in Table 1.

Parameter	Value
Simulator	NS-3.23
Topology	Manhattan grid road network 5x5 with 2000m edge
Propagation Model	Log Distance Propagation Loss Model
Number of nodes	50 to 250
Packet size	512 bytes
Packet rate	2Kb/s
Speed range	0-50m/s
Transmission range	250m
Traffic type	CBR, UDP
topology generation tool	Bonn Motion
Routing algorithm	AODV, KMCSA (proposed protocol)
Mac protocol	802.11b standard
Simulation Time	200 s

Table 1. Simulation Parameters

3.2. Results and discussions. In this section, we evaluate the parameters such as packet delivery ratio (PDR), Average delay, Packet loss ratio, Overhead and throughput of the entire system as a function of number of vehicles considered. The test results for the proposed protocol are compared with the existing protocol. The following figures 4, 5, 6, 7 and 8 depicts that the proposed algorithm is highly scalable and dependable as the anticipated results were generated with the developing number of vehicles.

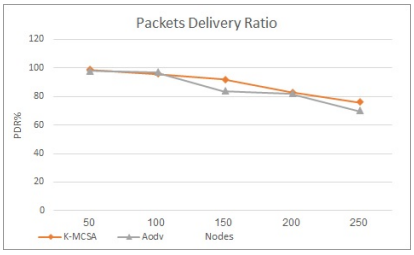


Fig. 4. Packets Delivery Ratio

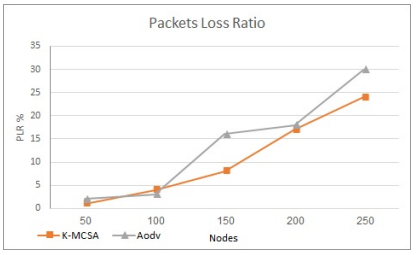


Fig. 5. Packets Loss Ratio

As shown in Fig. 4 and Fig. 5, the proposed protocol is superior to the popular protocol AODV in terms of PDR and PLR. This improvement is made possible by the hybrid K-MCSA protocol because of the modified CS algorithm and the weight parameters (fitness functions) namely distance, angle and road that are positively influenced to increase the packet delivery rate of the data packets by selecting highly stable routes with lesser number of link failures that influences packet loss.

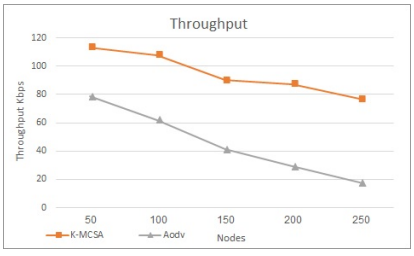


Fig. 6. Throughput

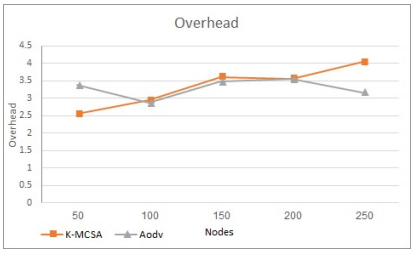


Fig. 7. Overhead

The results of the throughput and overhead are illustrated in Fig. 6,7. They show that the advantage of swift convergence rate increased the throughput of the proposed hybrid KMCSA protocol extensively with minimum control packets to select the best route and CH in shorter period of time as shown in Fig. 8.

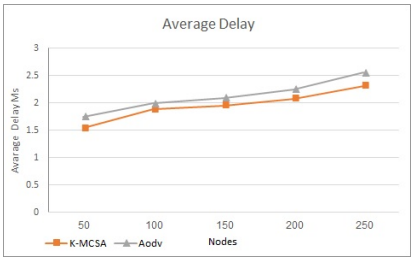


Fig. 8. Average Delay

4. Conclusion

this paper presented an efficient proposed routing protocol for VANETs that smartly utilizes k-means clustering and modified cuckoo search algorithm. In our proposed protocol, k-means clustering algorithm is responsible for cluster formation whereas modified cuckoo search algorithm is responsible for CH selection. The CS algorithm is one of the efficient metaheuristic algorithms particularly in higher searching space. The modified CSA identifies the optimal route from the known routes by computing the weights using three weight factors such as distance, angle, and road. Simulation results depict that the proposed hybrid K-MCSA protocol had produced superior performance compared to the AODV protocol in terms of packet delivery ratio, average delay, packet loss ratio and throughput. The weight functions used in the modified CSA helps to identify the optimal as well as stable and reliable route in shorter period of time. The proposed hybrid K-MCSA protocol outperformed the AODV protocol because of its distinct characteristics, specifically, the swift convergence rate because of the utilization of Levy distribution function.

REFERENCES

1. H. Fatemidokht, M. K. Rafsanjani, F-ant: an effective routing protocol for ant colony optimization based on fuzzy logic in vehicular ad hoc networks, Neural Computing and Applications 29 (11) (2018) 1127–1137.
2. S. Singh, S. Agrawal, Vanet routing protocols: Issues and challenges, in: 2014 Recent Advances in Engineering and Computational Sciences (RAECS), IEEE, 2014, pp. 1–5.
3. A. Kout, S. Labed, S. Chikhi, et al., Aodvcs, a new bio-inspired routing protocol based on cuckoo search algorithm for mobile ad hoc networks, Wireless Networks 24 (7) (2018) 2509–2519.
4. S. Al-Sultan, M. M. Al-Doori, A. H. Al-Bayatti, H. Zedan, A comprehensive survey on vehicular ad hoc network, Journal of network and computer applications 37 (2014) 380–392.

5. M. Jain, R. Saxena, Overview of vanet: Requirements and its routing protocols, in: 2017 International Conference on Communication and Signal Processing (ICCSP), IEEE, 2017, pp. 1957–1961.
6. C. Wu, S. Ohzahata, T. Kato, Routing in vanets: A fuzzy constraint q-learning approach, in: 2012 IEEE Global Communications Conference (GLOBECOM), IEEE, 2012, pp. 195–200.
7. B. Barekatain, D. Khezrimotlagh, M. A. Maarof, A. A. Quintana, A. T. Cabrera, Gazelle: An enhanced random network coding based framework for efficient p2p live video streaming over hybrid wmns, Wireless Personal Communications 95 (3) (2017) 2485–2505.
8. N. D. Kumari, B. Shylaja, Amgrp: Ahp-based multimetric geographical routing protocol for urban environment of vanets, Journal of King Saud University-Computer and Information Sciences 31 (1) (2019) 72–81.
9. Q. Zhang, M. Almulla, Y. Ren, A. Boukerche, An efficient certificate revocation validation scheme with k-means clustering for vehicular ad hoc networks, in: 2012 IEEE Symposium on Computers and Communications (ISCC), IEEE, 2012, pp. 000862–000867.
10. X.-S. Yang, S. Deb, Cuckoo search via lévy flights, in: 2009 World congress on nature & biologically inspired computing (NaBIC), Ieee, 2009, pp. 210–214.
11. A. H. Gandomi, X.-S. Yang, A. H. Alavi, Cuckoo search algorithm: a meta-heuristic approach to solve structural optimization problems, Engineering with computers 29 (1) (2013) 17–35.

UDC: 519.872

On the convergence of an iterative method for approximate analysis of a resource queuing system with signals

K.A. Ageev[1] and E.S. Sopin[1,2]

[1]Peoples Friendship University of Russia (RUDN University), 6 Miklukho-Maklaya Street, Moscow, 117198, Russian Federation

[2]Institute of Informatics Problems, FRC CSC RAS, 44-2 Vavilova street, Moscow, 119333, Russian Federation

ageev-ka@rudn.ru, sopin-es@rudn.ru

Abstract

Modern wireless networks are characterized by high user mobility. This factor can lead to changes in the quality of the channel during the lifetime of the interaction session. In order to take into account the fact of user movement in a queuing system with limited resources, signals are introduced. In this paper, an approximate model with signals is constructed. The simplification is to replace the signal with a new flow of requests. The mathematical proof of the method is given and the probabilistic-time characteristics are calculated.

Keywords: Queuing system, limited resources, random requirements, performance analysis, convergence of the method, simulation

1. Introduction

Resource queuing systems with signals can be applied for the analysis of the performance metrics of modern wireless networks [1, 2]. Upon arrival of a signal, a customer leaves the system and immediately comes again with a new resource requirements. Signal arrival indicates that a different amount of resources is required for the request.

Analytical calculations of probabilistic indicators of resource queuing system with signals with Poisson arrivals are presented in [3, 4, 5]. Application of such analytical formulas to calculate the stationary characteristics is rather complicated, since it implies calculation of multiple convolutions of resources requirements cumulative distribution function (CDF), and with an increase in the dimension of the system, the duration of calculations increases too.

Analytical calculations for such systems require significant computing resources, therefore, in [6] we have developed the simulation tool for queuing systems with limited resources with signals. Calculations with the help of simulation tools managed to reduce the load on computing resources in the calculations of the probability-time characteristics.

In [7] we proposed an approximate method for calculating stationary characteristics of the model. We provided comparison of calculations accuracy with the methods, proposed in [3, 4].

In this paper we propose the convergence of the iterative method for approximate model. The rest of the paper is organized as follows. Section II describes in brief an approximation method for the analysis of resource queuing system and proof of the convergence method. In section III we make short conclusion.

2. Approximate model for resource queuing system with signals

A multiserver queuing system with N servers, in which arriving customer occupies a server and a volume of limited resources R were considered in [7]. Resource requirements are independent identically distributed random variables with CDF $F(x)$. Customers arrive according to the Poisson process with intensity λ and the service times have exponential distribution with rate μ. Each customer in the system produces a flow of signals. Signals arrive according to the Poisson distribution with intensity γ. When a signal arrives, the customer releases the server and occupied resources and goes to the system again with new resource requirements. We got results with simulation modeling of the model, described above.

Assumption 1. Customers which arrive with signal contain a new type of customers with intensity $\widetilde{N}\gamma$, where $\widetilde{N}$ is an average number of customers in the system.

Denote types of customers as $l = \overline{1,2}$. We assume here that resource requirements are independent of arrival and serving processes.

Denote $p_{l,r}$ the probability that the customer of type l will require r resources. Then $p_{l,r}^k$ - the probability that k customers of type l will require r resources, where $p_{l,r}^k$ k-fold convolution of probabilities $p_{l,r}$. Denote $\rho_1 = \frac{\lambda}{\mu+\gamma}$ as offer load for customers of first type, and $\rho_2 = \frac{\widetilde{N}\gamma}{\mu+\gamma}$ for the second type.

According to [5] we can unite offer loads $p_r = \sum\limits_{l=1}^{L} \frac{\rho_l}{\rho} p_{l,r}$, where $\rho = \sum\limits_{l=1}^{L} \rho_l$, and calculate stationary probabilities as follows:

$$q_n(r) = q_0 \frac{\rho^k}{k!} p_r^{(k)}, \tag{1}$$

$$q_0 = \left(\sum_{k=0}^{N} \sum_{j=0}^{R} \frac{\rho^k}{k} p_j^{(k)} \right)^{-1}, \tag{2}$$

In [5], the recurrent algorithm for evaluation of normalization constant, which can be calculated by formula were developed:

$$G(n,r) = G(n-1,r) + \frac{\rho}{n} \sum_{i=0}^{r} p_i (G(n-1,r-i) - G(n-2,r-i)), \tag{3}$$

$$G^{-1}(N,R) = q_0. \tag{4}$$

In our model offered load for customers of second type depends on $\widetilde{N}$. We can calculate it by the formula bellow, using the normalization constant.

$$\widetilde{N} = \sum_{n=1}^{N} \sum_{r=0}^{R} n q_{n,r}, \tag{5}$$

$$\widetilde{N} = q_0 \rho \sum_{j=0}^{R} p_j G(N-1, R-j). \tag{6}$$

In [7] we developed approximation algorithm for calculating performance metrics. This algorithm based on recurrent computing of $\widetilde{N}$ until the difference between meanings of it become enough small.

To make a mathematical substantiation for this method introduce function

$$Z(x) = \frac{\lambda + \gamma x}{\gamma + \mu} \left(1 - \pi_B \left(\frac{\lambda + \gamma x}{\gamma + \mu} \right) \right), \tag{7}$$

where

$$\pi_B(\rho) = 1 - \left(1 + \sum_{n=1}^{N} \frac{\rho^n}{n!} F^{(n)}(R) \right)^{-1} \sum_{n=0}^{N-1} \frac{\rho^n}{n!} F^{(n+1)}(R), \tag{8}$$

Lemma 1. Equation $x = Z(x)$ has a single solution, and can be found this solution by fixed-point iteration.

Proof. Proof of Lemma 1. Consider the case in which the resource requirements of the redirected customer due to the arrival of the signal have the same distribution as the primary customer. Then in approximate model:

$$(\lambda + \gamma \widetilde{N})(1 - \pi_B(\rho)) = \widetilde{N}(\gamma + \mu), \tag{9}$$

where $\rho = \frac{\lambda+\gamma\widetilde{N}}{\gamma+\mu}$.

From here follows:

$$\widetilde{N} = \rho(1 - \pi_B(\rho)), \tag{10}$$

Thus, the average number of customers in the system is subject to the expression

$$\widetilde{N} = Z(\widetilde{N}). \tag{11}$$

According to the principle of contraction, if $Z(x)$ is a contraction when $x \geq 0$, then it has a single fixed point on the semi-straight line $x \geq 0$ which is the solution of the equation $Z(x) = x$ this means that the average number of applications in the system can be found using the simple iteration method.

Calculate the derivative of $Z(x)$:

$$Z'(x) = \frac{\gamma}{\mu+\gamma}\left(1 - \pi_B\left(\frac{\lambda+\gamma x}{\gamma+\mu}\right)\right) - \frac{\lambda+\gamma x}{\gamma+\mu}\frac{\gamma}{\mu+\gamma}\pi'_B\left(\frac{\lambda+\gamma x}{\gamma+\mu}\right). \tag{12}$$

Take into account that $\rho = \frac{\lambda+\gamma x}{\gamma+\mu}$

$$Z'(x) = \frac{\gamma}{\mu+\gamma}(1 - \pi_B(\rho) - \rho\pi'_B(\rho)). \tag{13}$$

Let us now estimate the derivative of $\pi'_B(\rho)$. From physical considerations, we can conclude that $\pi'_B(\rho) \geq 0$, because with increasing offered load blocking probability couldn't decrease. We introduce an additional notation:

$$G(\rho) = 1 + \sum_{n=1}^{N} \frac{\rho^n}{n!} F^{(n)}(R), \tag{14}$$

where $G(\rho)$ makes sense of a normalization constant. It is easy to see that the expression for the probability of blocking $\pi_B(\rho)$ looks like

$$\pi_B(\rho) = 1 - \frac{G'(\rho)}{G(\rho)}, \tag{15}$$

and its derivative

$$\pi'_B(\rho) = \left(\frac{G'(\rho)}{G(\rho)}\right)^2 - \frac{G''(\rho)}{G(\rho)}. \tag{16}$$

Then we estimate each of $\pi'_B(\rho)$.

$$\frac{G'(\rho)}{G(\rho)} = G^{-1}(\rho) \sum_{n=0}^{N-1} \frac{\rho^n}{n!} F^{(n+1)}(R) =$$
$$= \frac{1}{\rho} G^{-1}(\rho) \sum_{n=0}^{N-1} (n+1) \frac{\rho^{n+1}}{(n+1)!} F^{(n+1)}(R) = \quad (17)$$
$$= \frac{1}{\rho} G^{-1}(\rho) \sum_{n=1}^{N} n \frac{\rho^n}{n!} F^{(n)}(R).$$

It is easy to see that the expression on the right represents the mathematical expectation of the number of applications in the resource queuing system:

$$\frac{G'(\rho)}{G(\rho)} = \frac{\widetilde{N}}{\rho}. \quad (18)$$

Then turn to the second term of the expression for $\pi'_B(\rho)$

$$\frac{G''(\rho)}{G(\rho)} = G^{-1}(\rho) \sum_{n=2}^{N} \frac{\rho^{n-2}}{(n-2)!} F^{(n)}(R) =$$
$$= \frac{1}{\rho^2} G^{-1}(\rho) \sum_{n=2}^{N} n(n-1) \frac{\rho^n}{n!} F^{(n)}(R) = \quad (19)$$
$$= \frac{1}{\rho^2} (G^{-1}(\rho) \sum_{n=2}^{N} n^2 \frac{\rho^n}{n!} F^{(n)}(R) - G^{-1}(\rho) \sum_{n=2}^{N} n \frac{\rho^n}{n!} F^{(n)}(R)).$$

where $\widetilde{N}^{(2)}$ is the second point of the number of applications in the system. Taking into account the obtained relations, the expression for $\pi_B(\rho)$ looks like

$$\pi'_B(\rho) = \frac{\widetilde{N}^2 - \widetilde{N}^{(2)} + \widetilde{N}}{\rho^2} = \frac{\widetilde{N} - (\widetilde{N}^{(2)} - \widetilde{N}^2)}{\rho^2} \quad (20)$$

The expression in parentheses represents the positive variance of the number of applications in the system, which means

$$\pi'_B(\rho) < \frac{\widetilde{N}}{\rho^2} = \frac{(1 - \pi_B(\rho))}{\rho} < \frac{1}{\rho}. \quad (21)$$

Substituting the resulting estimate in the expression for $Z'(x)$, we get

$$|Z'(x)| < \frac{\gamma}{\mu + \gamma} < 1. \quad (22)$$

which is a sufficient condition for a contraction – which was required to be proved.

■

3. Conclusion

In this paper, the mathematical justification of the application of the approximate model is given. Based on the numerical experiment in comparison with the results of simulation modeling, it is concluded that the algorithm can be used to calculate the probability of blocking on receipt, the average number of applications in the system and the average amount of occupied resource, since the calculation error is from 5% to 10%.

4. Acknowledgements

The reported study has been funded by RFBR, projects no. 19-07-00933 and 20-07-01064.

REFERENCES

1. Lu X., Dohler M., Sopin E. et al. Integrated Use of Licensed- and Unlicensed-Band mmWave Radio Technology in 5G and Beyond // IEEE Access. 2019. Vol.7: 24 376-24 391.
2. Galinina O., Andreev S., Turlikov A,. Koucheryavy Y. Optimizing Energy Efficiency of a Multi-Radio Mobile Device in Heterogeneous Beyond-4G Networks. Performance Evaluation, vol.78, 2014, pp. 18-41.
3. Tikhonenko O., Generalized Erlang problem for service systems with finite total capacity. Problems of Information Transmission, 41 (3), 2005, pp. 243–253.
4. Sopin E., Vikhrova O., Samouylov K. LTE network model with signals and random resource requirements, Proc. of 9th International Congress on Ultra Modern Telecommunications and Control Systems and Workshops (ICUMT), 2017, pp. 101 – 106.
5. Sopin E.S., Ageev K.A., Markova E.V., Vikhrova O.G., Gaidamaka Yu.V., Performance Analysis of M2M Traffic in LTE Network Using Queuing Systems with Random Resource Requirements;, Automatic Control and Computer Sciences, 2018, Vol.52, No. 5, pp. 345–353.
6. Eduard Sopin, Kirill Ageev, Sergey Shorgin, Simulation Of The Limited Resources Queuing System For Performance Analysis Of Wireless Networks, Proceedings 32st European Conference on Modelling and Simulation, 2018, pp. 505 – 509.
7. Sopin E. S., Ageev K. A., Samouylov K. E. Approximate Analysis Of The Limited Resources Queuing System With Signals. In: 33rd International ECMS Conference on Modelling and Simulation, ECMS 2019 Caserta, Italy, June 11-14, 2019. Proceedings; 2019: pp. 462-465. ECMS proceedings.

UDC: 004.7

Availability factor analysis of a network in mesh structure

A.V. Dagaev[1], V.D. Pham[1], R.V. Kirichek[1], O.V. Afanaseva[2], E.A. Yakovleva[3]

[1]Bonch-Bruevich Saint-Petersburg State University of Telecommunications, St.Petersburg, Russian Federation
[2]Staint-Petersburg Mining University (SPMU), St.Petersburg, Russian Federation
[3]Ivangorod Humanitarian and Technical Institute (branch) of Federal State Autonomous Educational Institution of Higher Education "St. Petersburg State University of Aerospace Instrumentation", St.Petersburg, Russian Federation

adagaev@list.ru, fam.vd@spbgut.ru, kirichek@sut.ru, ovaf72@gmail.com, y_katerina@rambler.ru

Abstract

The development of network technologies leads to the introduction of data transmission systems in all areas of human activity. Problems cover large areas with reliable networks, intellectualizing devices for receiving and transmitting data, increasing the speed, quality and spectrum of transmitted information. Requirements for the given characteristics of the reliability of networks are among the most promising and essential tasks since the cost of equipment, service life, and network maintenance strategy depend on this. Analytical and simulation models play an essential role in determining the reliability characteristics; mathematical apparatus and programming become an inevitable factor in a successful project. This paper presents a methodology and an example of calculating the reliability of networks in a mesh topology.

Keywords: simulation, system, mesh network, mean availability factor, probability

1. Introduction

Mesh network topology is designed to solve a wide range of tasks, from monitoring the battlefield and analyzing weather conditions to redistribute smart sustainable city technologies' load. The mesh network structure can be represented in a graph or an $m \times n$ matrix. Determination of reliability characteristics depends on the initial structure of the network and, depending on the reliability of its elements, and their maintenance strategies can take different values. Today, many sources are known on the topic of determining the characteristics of the system reliability [1, 2].

Papers [3, 4, 5, 6] are devoted to studying the reliability characteristics of network structures and computer systems, such as calculations for the exponential distribution of reliability characteristics, packet coding in communication channels, and reliability calculation for Cisco equipment. In these articles, asymptotic estimates of reliability characteristics were used, however, evaluation of complex structures and non-asymptotic models were not given attention. Therefore, let us describe the functioning strategy more detail in this paper.

2. Description of the functioning model

The model assumes the presence of built-in control with the detection of failures in the system and its complete recovery. In the event of a failure, the system is repaired and is idle until it is restored.

At the initial moment $t_0 = 0$, the system starts to work, and the availability has a maximum value. After that, the system works to failure – ξ_i then recovery is performed, which lasts a period – $n_{i\,fr}$.

After recovery – τ_{ir} the system continues its work until the next moment of failure, then there is a recovery and transition to an operational state. This cycle is repeated until the selected time t. The presented strategy is shown in Fig. 1.

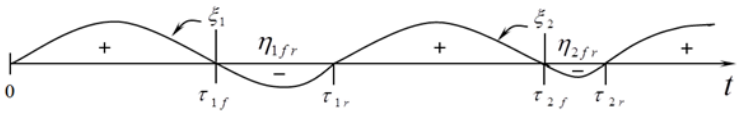


Fig. 1. The strategy takes into account the built-in control

The pros and cons of the above designations are the periods of operation and repair of the system; ξ_i – i-th time to failure; $n_{i\,fr}$ – the duration of the i-th disaster recovery; τ_{if} and τ_{ir} – time intervals from the start of work to the i-th failure and the i-th recovery. These values can be written in terms of several other random variables:

$$\tau_{0r} = 0; \begin{cases} \tau_{1f} = \xi_1 \\ \tau_{1r} = \xi_1 + \eta_{fr} \end{cases}; \begin{cases} \tau_{2f} = \tau_{1r} + \xi_2 \\ \tau_{2r} = \tau_{1r} + \xi_2 + \eta_{fr} \end{cases}; \begin{cases} \tau_{if} = \tau_{i-1,r} + \xi_i \\ \tau_{2r} = \tau_{i-1,r} + \xi_i + \eta_{fr} \end{cases}$$

The mean availability is the sum of the probabilities of the system being in a working state:

$$\begin{aligned} K(t) &= \sum_{i=1}^{\infty} P(\tau_{i-1,r} < t < \tau_{i,f}) = P_1(t < \xi_1) + \sum_{i=1}^{\infty} P(\tau_{i,r} < t < \tau_{i+1,f}) \\ &= (1 - F_\xi(t)) + \sum_{i=1}^{\infty} P(\tau_{i,r} < t < \tau_{i+1,f}) \end{aligned} \tag{1}$$

After performing some transformations, we obtain the convolution by doing the inverse Laplace transform. Then, we obtain the equation of the non-asymptotic system availability factor. Thus, the availability can be written as follows:

$$K(t) = [1 - F_\xi(t)] + \int_0^t f_{\eta_{fr}} \int_0^{t-\chi} f_\xi(y) K(t - x - y) dy dx \tag{2}$$

The asymptotic availability equation is found under the condition $t \to \infty$ and can be derived from this equation. The asymptotic availability is the ratio of the mathematical expectation of the failure time to the sum of the mathematical expectations of the failure and recovery times.

$$K_a = \frac{M(\xi)}{M(\xi) + M(\eta_{fr})} \tag{3}$$

2.1. Identical elements in the system. If under the condition of the same elements in the graph and a parallel data transfer condition, we can assume that the reliability of elements in each chain can be calculated as a sequential structure (Fig. 2).

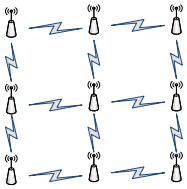


Fig. 2. The mesh network structure

Therefore, the reliability calculation (the probability of no-failure operation and the availability) can be calculated using a similar formula for non-recoverable and recoverable systems. Using the method of dimension reduction of the original problem, first, we calculate the reliability of independent paths of the system (chains) in the graph. Next, we determine the reliability of parallel chains. Finally, we use the method of minimal paths and minimal sections. A path is any set of elements for which the system is operable. Thus, the availability for an individual chain will have the following product form:

$$K_i == \prod_{i=1}^{m} K_{nod\,i} = {K_{nod\,i}}^m \tag{4}$$

where $K_{nod\,i}$ is the availability of the i-th element of the chain, in this case it should be taken into account that $K_{nod\,i}$ are the same for $\forall i \in (1 \dots m)$

As can be seen from the equation, chain availability is the availability of an individual element to the power m. Therefore, to receive the upper-reliability estimate based on the minimum path method, we first need to find all the minimum paths of the system. Then, we connect the elements of each minimum path in series, and all the resulting chains with a series connection of the elements are connected in parallel.

Then, we will find the system availability. For this, we write the availability (and the probability of no-failure operation) for a parallel connection. It is known that the element unavailability ratio is determined first for parallel connection of recoverable elements, or the element failure probability is determined for non-recoverable systems.

Thus, the system availability factor K_s is calculated as the deviation from the unit of the system unavailability factor $\prod_{i=1}^{n} K_{un.i}$, i.e.:

$$K_s = 1 - \prod_{i=1}^{n} K_{un.i} = 1 - (K_{un.i})^n = 1 - (1 - (K_{nod\,j})^m)^n \tag{5}$$

where $K_{un.i} = (1 - (K_{nod\,j})^m)$ is the chain unavailability factor; $K_{un.s} = (1 - (K_{nod\,j})^m)^n$ is the unavailability factor of all chains or system, where $j \in (1 \dots n)$.

Using the method of minimum sections, we find the upper availability value. To do this, we find all the minimum sections of the system, then we connect the elements of each minimum section in parallel, after which we connect all the minimum sections in series. Thus, if we find a homogeneous network section, it contains all elements vertically in the number n.

Then the availability equation can be written in the following form:

$$K_s = (1 - (1 - K_{nod\,j})^n)^m \tag{6}$$

where $(1 - K_{nod\,j})$ – the unavailability factor of the j-th element.

Next, we give an example of calculating the system under the conditions of the asymptotic formulation of the problem. In this case, the asymptotic availability value is calculated using the formula (3). The initial data are the mathematical expectations of the time of failure and recovery. Based on the real data of the equipment passports, it is known that the mathematical expectation of failure is 90000 hours, and the recovery time is two days. In this case, the value of the asymptotic availability factor of the element will be equal to $K_{nod\,j} = 0.999467$. It should be noted that this value is the worst estimate of the availability coefficient of a system element. As a mesh topology, we take a network with dimension $n = 8$ and

$m = 7$ cells. Substituting this value into the equation (5), we receive:

$$K_s = (1 - (1 - 0.999467^7))^8 = 1 - (3.71E^{-20}) \cong 1 \quad (7)$$

where the unavailability coefficient of a single chain is 0.003725. In this case, the system availability will tend to unity since the chain availability slightly decreases with an increase in its length (with an increase in the degree m of equation (5)).

We consider the lower value of the system availability coefficient in this case. We calculate the unavailability of an element, it will be as follows: $(1 - K_{nod\,j} = 5.33e^{-4}$. Then the lower value of the availability coefficient will be as follows:

$$K_s = (1 - (1 - 0.999467)^8)^7 \cong 1$$

Due to the high availability coefficient of one element, the upper and lower availability coefficient will be very high and tend to unity. Let us show using a test example that the availability can take values different from unity. Assume that we have a system with low reliability $Mean.failure = 1000$, $Mean.recovery = 300$. Then the asymptotic availability ratio will be 0.769. Based on this, substituting the value in (5) and (6), we get the availability value equal to 0.751 and 0.99994.

2.2. Failure of some elements. In this section, we present formulas that can be used to failure some elements, where the rest of the elements send data, and the system performs its function. For example, suppose that half of the elements of the system fail and evenly. After a failure, a different number of elements may remain in the system horizontally and vertically. Depending on how many elements - even or odd remain, the original formula can be converted to one of the followings presented below:

$$\textbf{even n, m} \rightarrow K_s = 1 - (1 - K_{nod\,j}^{m/2})^{n/2}$$
$$\textbf{odd n, m} \rightarrow K_s = 1 - (1 - K_{nod\,j}^{(m+1)/2})^{(n+1)/2}$$
$$\textbf{even n, odd m} \rightarrow K_s = 1 - (1 - K_{nod\,j}^{(m+1)/2})^{n/2}$$
$$\textbf{odd n, even m} \rightarrow K_s = 1 - (1 - K_{nod\,j}^{m/2})^{(n+1)/2}$$

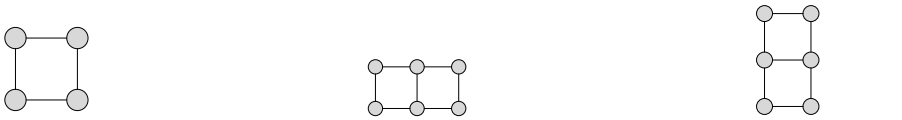
Where, n and m represent the initial number of nodes horizontally and vertically in the graph representing the mesh topology.

2.3. Different elements of the system. In this case, the availability coefficient cannot be written in a concise form. However, it is possible to write a general formula for the entire system in the following form:

$$K_s = 1 - \prod_{i=1}^{n} \left(1 - \prod_{j=1}^{m} K_{nod\,j_i} \right) \quad (8)$$

3. Example of availability analysis

We describe the results of calculating the availability factor of the mesh network using the Esari-Proshaan method. Figure 3 shows three simple mesh structures with a minimum number of elements. The formulas for calculating the upper (6) and lower (5) boundaries of the system availability factor were used in the calculations.



(a) square structure (b) horizontal structure (c) vertical structure

Fig. 3. Simple mesh structures

The following analytical formulas were obtained for calculating the boundaries of the availability factor of the presented structures. For the first structure 3a), simple equations for the lower and upper bounds of the availability factor are derived, respectively:

$$\begin{aligned} K_{s.up.b} &= (K_{nod})^2 \times (2 - (K_{nod})^2) \\ K_{s.low.b} &= (K_{nod})^2 \times (2 - K_{nod})^2 \end{aligned} \tag{9}$$

The figure below shows the behaviour of the boundaries of the availability factor for structure 3a) in identical system elements. The dotted line shows the average value of the availability factor, which can be considered close to the actual value. As can be seen from the graph, the upper limit of the availability factor is due to the initial parallel connection of single vertical elements, the resulting section of the circuit and their further serial connection. Moreover, the lower boundary is obtained by serial connection of horizontal elements and their further parallel connection.

For the second structure 3b, using formulas (5) and (6), the following equations for the boundaries of the availability factor were obtained:

$$\begin{aligned} K_{s.up.b} &= (K_{nod})^3 \times (2 - (K_{nod})^3) \\ K_{s.low.b} &= (K_{nod})^3 \times (2 - K_{nod})^3 \end{aligned} \tag{10}$$

Figure 5 below shows the boundaries of the availability factor with orange and dark blue lines and its average estimate with a dash-dotted line.

For the third structure 3c), the following equations for the bounds of the availability factor were obtained:

$$\begin{aligned} K_{s.up.b} &= (K_{nod})^2 \times (3 - 3 \times (K_{nod})^2 + (K_{nod})^4) \\ K_{s.low.b} &= (K_{nod})^2 \times (3 - 3 \times K_{nod} + (K_{nod})^2)^2 \end{aligned} \tag{11}$$

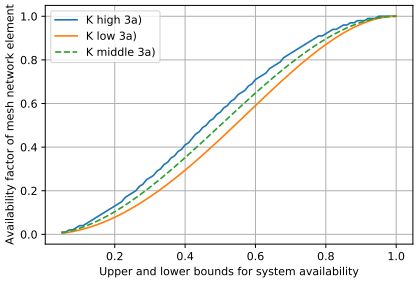


Fig. 4. Dependence of the boundaries and the average estimate of the system (3a) availability factor on the network element availability factor

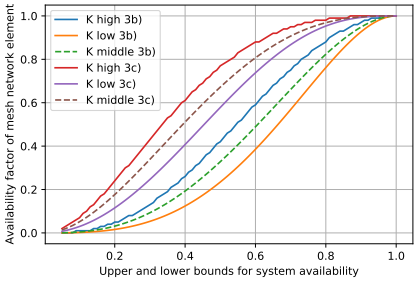


Fig. 5. Dependence of the boundaries and the average estimate of the (3b and 3c) systems availability on the element availability factor

As can be seen from the graph above, the reliability of the scheme for the third structure (3c) is on average higher than for the second. This is because, during the section in the third structure, there will be three parallel elements, while there are only two of them in the first cases. It should be noted that the average value of the availability factor should be used since it will give values close to reality since the failure of horizontal and vertical elements is equally likely. The construction of graphs similar to the graphs shown in Fig. 3,4 will allow installing elements in the system with an availability factor that satisfies the required level of the system resource, which is useful for systems with a high level of reliability and for planning preventive and remediation works.

4. Maintenance strategy description

4.1. Simulation model. In this work, we describe a simulation model for calculating reliability characteristics.

First, the model finds all paths in the graph (matrix) from the source node to the destination node. We can use different dynamic methods to find a path, such as

breadth-first and depth-first traversal. In the latter case, we take the initial node vertex of the graph and move in the direction of the output node to the right, while we can measure the path length and move along the vector of decreasing the distance to the output node. We mark the traversed vertices when we reach the output node and save all the paths in the matrix. Then we start moving in the opposite direction, accidentally adding a new and not yet traversed adjacent vertex to the path while checking the identity of the new path with the saved one. If there is no new adjacent vertex, we return to the vertex from which we got to the current one and make the next attempt. If all vertices are exhausted, then we have received a complete list of paths. Thus, the first path found will be the shortest. Finding all the paths manually for the dimension of network structures more than five seems to be impossible. Therefore the use of software and traversal methods is necessary.

The developed application defines all paths leading from the source node to the destination node. We use the Ezari-Proscan method to determine the reliability characteristics. In order to calculate reliability characteristics from above, it is required to know unique minimum paths. However, knowledge of all paths is necessary to determine both the least reliable nodes and paths and determine the most reliable ones.

After the paths have been determined, the minimum paths required to determine the upper-reliability estimate are determined. Then, the operation of the elements of the system is simulated, with a given functioning strategy. In the implemented version, the presence of built-in control is taken into account, as described above. The simulation of a random value of failure and recovery was carried out according to the normal distribution with the parameters: $M.f = 100000$, $Var.f = 25000$, $M.rec = 100$, $Var.rec = 25$. The size of the mesh structure was taken equal to 7×7. The total simulation time is taken equal to ten times the recovery period $(Mo.f + Mo.rec) * 10$.

We used the Box-Muller method for generating random variables. The method is convenient because it allows one to obtain two independently distributed random variables with zero mean and unit variance. Below are the formulas for generating random variables using this method.

$$X_1 = R_1\sqrt{\frac{-2lnD}{D}}; X_2 = R_2\sqrt{\frac{-2lnD}{D}} \tag{12}$$

where X_1, X_2 – the desired values, $D = R_1 + R_2$, where R_1, R_2 – uniformly distributed random variables on $(-1, 1)$. It should be noted that the presented random variables must satisfy the condition $D \in [0..1]$,

4.2. Methodology for reliability evaluation. 1. A structural diagram of the system is constructed in the form of a graph or matrix.

2. Find all the paths leading from the source node to the destination node.

3. The functioning strategies of the elements are set.

4. For modelling, periods of preventive maintenance, distribution characteristics of the time of failure, recovery, determination of the failure place, etc. are set.

5. Simulation of work with statistics is performed for each element at least ten thousand times. The average value of the simulated random variable is taken as the calculated estimate. Simulation is performed over the entire specified time interval. The random values of failure and recovery are stored in the element state matrix, which contains the times of the element state change. The number of elements in this matrix is defined as the ratio of the simulation time to the estimate of the mathematical expectation of the failure and recovery cycle, multiplied by two.

6. Using the element state matrix, a binary matrix (or vector for each) state is created for the entire simulation time. Binarization is performed to simplify calculations of reliability characteristics using functions of logic algebra and logical operations. The sampling step was taken equal to one minute, the size of the binary matrix is calculated as the size of the matrix of states multiplied by 60. When passing from the matrix of states of elements to the matrix of binarization, the state of the elements is checked. The working sections of the first matrix go to the unit elements of the second, non-working ones to zero.

4.3. Model calculation example. We present the results of simulation for the element and the cellular system. Below is a graph of the availability factor of one element with a value of a mathematical expectation of a failure time of 100000 hours, a recovery time of 50 hours, a standard deviation of a failure of 25000 hours and a recovery of 10 hours. The mean availability of the element according to the presented data was 0.9995.

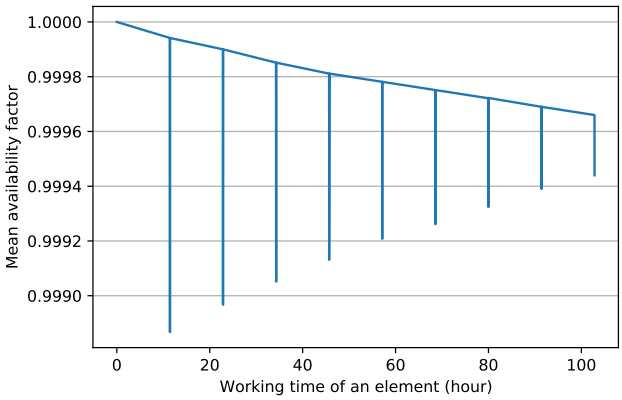


Fig. 6. Element availability

When modelling the mesh network, the parameters of the distribution laws presented above were taken, except the mathematical expectation of the recovery time, which was taken equal to 100 hours. As a result, the mean non-asymptotic availability was obtained approximately equal to one as:

$$K_c = 1 - (3.18e^{-25}) \cong 1 \tag{13}$$

As can be seen from equations (5)-(13), the calculated values of the asymptotic availability and the mean availability obtained by simulation converge to one. The rapid recovery of the system determines high values of the asymptotic and non-asymptotic availability coefficient of the element and the system as a whole in the event of the failure and the high value of the system operation time to failure.

Calculations of the probability of no-failure operation were performed in the case of non-recoverability of the elements in the mesh network with the parameters presented above. The graph of the probability of failure-free operation is shown in Fig. 7.

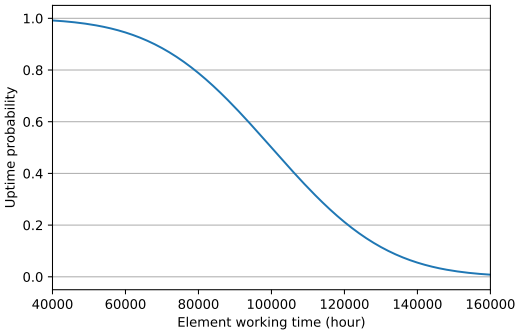


Fig. 7. Probability behavior of the uptime element

For non-recoverable elements, the probability of failure-free operation of the system can be calculated by analogous to formula (5) for the uptime probability. At the moment, one hundred thousand hours, the uptime probability will be 0.939, which is a sufficiently high value for 11 years of system operation.

5. Conclusion

In this research, the following results were obtained:

1. The analysis of the operation and functioning of the elements has been carried out in the network with mesh topology.

2. The technique of analytical determination of the availability is described for a homogeneous network. The calculation model of minimum paths and section was considered.

3. A method of availability determination has been developed using a simulation model.

4. The simulation model of the system has been developed, it is shown that it converges to an analytical model 5.

It should be noted that the simulation model can be used to analyze the reliability characteristics and other indicators of systems of any dimension and complexity. It can be used to model subsystems of hazardous and expensive objects, such as aerospace, telecommunications, energy, transport and other areas of human activity.

Acknowledgment

The publication has been prepared with the support of the grant from the President of the Russian Federation for state support of leading scientific schools of the Russian Federation according to the research project SS-2604.2020.9.

REFERENCES

1. N. Maalel, E. Natalizio, A. Bouabdallah, P. Roux, M. Kellil, Reliability for emergency applications in internet of things, in: 2013 IEEE International Conference on Distributed Computing in Sensor Systems, IEEE, 2013. doi:10.1109/dcoss.2013.40.
2. I. Beichl, E. Moseman, F. Sullivan, M. Bowie, Computing network reliability coefficients, in: Proceedings of the Forty-Second Southeastern International Conference on Combinatorics, Graph Theory and Computing, Vol. 207, 2011, pp. 111–127.
3. R. Sattiraju, H. D. Schotten, Reliability modeling, analysis and prediction of wireless mobile communications, in: 2014 IEEE 79th Vehicular Technology Conference (VTC Spring), IEEE, 2014. doi:10.1109/vtcspring.2014.7023170.
4. J. Xin, L. Guo, N. Huang, R. Li, Network service reliability analysis model, Chemical Engineering Transactions 33 (2013) 511–516. doi:10.3303/CET1333086.
5. X. Zhu, Y. Lu, J. Han, L. Shi, Transmission reliability evaluation for wireless sensor networks, International Journal of Distributed Sensor Networks 12 (2) (2016) 1346079. doi:10.1155/2016/1346079.
6. Cisco, Network availability: How much do you need? how do you get it? URL `https://www.cisco.com/web/IT/unified_channels/area_partner/cisco_powered_network/net_availability.pdf`

UDC: 004.7

A Method for Link Quality Estimation in LoRa Network based on Support Vector Machine

V.D. Pham[1], P.H. Do[2], D.T. Le[3], R.V. Kirichek[1]

[1]The Bonch-Bruevich Saint-Petersburg State University of Telecommunications, Saint Petersburg, Russia
[2]Danang Architecture University, Danang, Vietnam
[3]The University of Danang - University of Science and Technology, Danang, Vietnam

fam.vd@spbgut.ru, haodp@dau.edu.vn, letranduc@dut.udn.vn, kirichek@sut.ru

Abstract

In this paper, we propose a link quality estimation (LQE) method to classify the connection level between two nodes. The LQE method is developed based on the kernel support vector machine (kSVM), which is one of the machine learning techniques used in classification problems. Series of experiments were performed to collect a dataset consisting of received signal strength indicator (RSSI), signal-to-noise ratio (SNR) of received packets, and packet reception rate (PRR). The trained model shows a high prediction accuracy (mean = 95%) while using 10% of the dataset for training.

Keywords: Internet of Things, link quality estimation, wireless sensor network, LoRa, RSSI, SNR, support vector machine (SVM)

1. Introduction

Wireless sensor network (WSN) [1] has attracted much attention since its appearance by its applications based on many sensor nodes, which are capable of sensing and gathering information from the environment, then processing and transmitting those sensed data to the remote base station. The wireless sensor nodes can be deployed for different purposes [2] such as supervisory and security; environmental monitoring; smart living; smart agriculture; health care; military. Their main advantage is the ability to deploy in almost any kind of geography, including dangerous environments.

Currently, many wireless network technologies support WSN networks such as Wi-Fi, Bluetooth, Zigbee, LoRa, Sigfox, NB-IoT [3, 4]. The LoRa technology proved consistent to create a Low-Power Wide-Area Network (LPWAN). In some cases, Sigfox offers longer-range communication compared to LoRa, but it has service subscription costs. Meanwhile, NB-IoT is a cellular-based technology and consumes

much energy. Besides long-range capabilities (up to 15km), the LoRa technology also has advantages in battery life optimization, easy deployment, and robustness to interference. These features make LoRa a good choice for a vast number of WSN applications.

In the LoRa network in particular and the WSN network in general, the selection of optimal routing paths is always challenging due to the links' dynamic behavior [5, 6]. The link quality must be suitable for critical industrial applications as they require sensor nodes to accurately measure the environment and communicate these data to other nodes without error. Many factors affect the spread of radio signals, such as properties of the transmission environment, which leads to multi-path propagation effects, noise, and interference by concurrent transmissions of other communication technologies or electromagnetic sources; the power of wireless signals. Typically, there are several methods to assess link quality: link estimation based on data-link layer parameter [7], link estimation based on physical layer parameter [8], and comprehensive link quality estimation [9].

Link quality estimation (LQE) is considered as a process to estimate the reliability level of the connection between nodes. The high quality links ensures the network quality of service (QoS) in decreasing the packet loss. The LQE can assist higher network protocols to mitigate and overcome the less reliable link. For instance, link quality estimation is an assistant for routing protocols to maintain routing tasks. Moreover, link quality estimation also can assist to maintain the network topology. With the high quality links, the mechanism is considered to control efficiently the network topology to maintain robust network connectivity [10].

The metrics used to evaluate the quality of the link mainly are RSSI, LQI, SNR, and PRR [11]. RSSI – received signal strength indicator represent the power of a received radio signal. The received packet is correctly decoded if the RSSI value is more than receiver sensitivity. Similar to RSSI, LQI – Link Quality Indicator is an integer value (from 0 to 255) used to estimate the link quality. The LQI value is defined by raido-chip manufacturer. Another metric is SNR – signal-to-noise ratio, which defines the level of received signal to the level of background noise. All three metrics, RSSI, LQI, and SNR are estimated based on hardware implementation. Moreover, the metric PRR – packet reception rate presents the ratio of received to sent packets over a defined window size, which can be implemented in the software. However, the LQI metric depends a lot on the hardware, and therefore, in this paper, we propose the LEQ method using RSSI, SNR, and PRR to evaluate the link quality. Furthermore, we will use a machine learning model trained by the input data collected by experiments in the laboratory environment [12] to evaluate and classify the links into different groups . From that result, we can choose the optimal link for routing tasks in each particular scenario.

2. Basic Estimation Metrics

2.1. Hardware-based estimation methods. Based on hardware implementation, the following parameters such as RSSI, LQI, SNR can be obtained directly from the wireless transceiver without additional calculations. These parameters are saved in registers after receiving a packet.

Received signal strength indicator. RSSI is an estimated measure of power strength that an RF device received from and another RF device. At long transmission distances, the signal gets weaker, leading to the probability of packet losses since the receiver can not decode the signal correctly. The signal is measured by the received signal strength indicator, which in most cases indicates how well a particular radio link can hear the remote wireless node. In the open space, the received signal power can be estimated using the following equation:

$$P_{rx} = P_{tx} + G_{tx} + G_{rx} + PL \tag{1}$$

where P_{rx} is the expected received power or the received signal strength indicator RSSI, P_{tx} is devoted to the transmission power, G_{tx} and G_{rx} are the transmitting and receiving antenna gains, PL is represented as path loss.

The transmission power is reduced according to the increasing distance between receiver and transmitter. Therefore, if the obtained RSSI value is less than the receiver sensitivity, the signal could be incorrectly decoded. Moreover, the signal strength decreases due to the obstacles and environment.

Signal to noise ratio. SNR is devoted to the differences in level between the received signal strength.

$$SNR(dB) = 10 \log_{10} \frac{P_{signal}}{P_{noise}} = P_{signal}(dB) - P_{noise}(dB) \tag{2}$$

Signal to noise ratio defines the difference in level between the signal and the noise for a given signal level. The less noise is generated by the receiver, the better the signal-to-noise ratio.

Link Quality Indicator. LQI is a value used to quickly determine whether the link belongs to the reliable reception range. For example, this indicator is implemented in some transceivers (CC2420) used in multi-hop networks such as Zigbee to assess the link cost. LQI is required and described in the Zigbee and IEEE 802.15.4 standards. After receiving each packet, the LQI measurement results as an integer ranging from 0 to 255. The minimum and maximum LQI values correspond to the lowest and highest quality IEEE 802.15.4 signals detectable by the receiver.

However, these values can only be obtained from successfully received packets. In the case of packet loss, the link quality might be overestimated. Moreover, the

measurement results vary unstable; hence, it is difficult to estimate the exact link quality if only one measurement is performed. Despite fast and cheap implementation in the hardware, these methods allow obtaining limited information about quality links for stable channels. Thus, using a combination of hardware and software metrics will improve the accuracy of link quality estimation.

2.2. Software-based estimation methods. Software implementation allows to estimate some values such as packet reception rate (PRR) or packet delivery ratio (PDR), throughput, expected transmission number.

In a certain transmission period, PRR can be obtained by either direct calculation or approximation. PRR represents the ratio of the number of successfully received packets to the total number of packets transmitted. Therefore, a term of window size is considered to choose an interval time to calculate PRR. As a result, an accurate estimate can be obtained in a short time with high or low link quality. On the other hand, larger window size is required to obtain PRR more accurately.

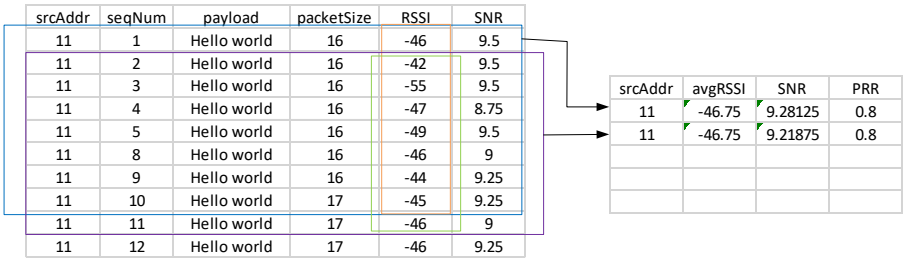
Another method considers a number of expected transmissions as a metric for the link quality estimation. That means how many transmissions are required to transmit a packet successfully while considering the number of lost packets.

3. LoRa Link Quality Estimation

3.1. Experimental measurement and preprocessing. Series of experiments were performed at the "Internal Research, Development and Testing Center for new equipment, technologies and services" supported by "Rostelecom" and International Telecommunication Union. We used several LoRa Nodes (`https://heltec.org/project/htcc-ab01/`) in the experimental measurement, including sending nodes and a sink node. The following parameters such as frequency = 868 MHz, bandwidth = 250 kHz, transmission power = 5 dBm, spreading factor = 7, coding rate = 4/5 were configured to LoRa nodes. Interval time between packet transmission was set randomly in range (500, 2000) ms. The sink node is fixed and connected to a computer to save the experimental data. On the other hand, the sending node moves far away from the sink node at a low speed to measure signal power and noise changes related to the packet reception rate. After this experiment, we received 2500 packets for further preprocessing.

For each received packet, we obtained the RSSI and SNR values from the hardware. Moreover, the sequence number indexed to packet number also is collected to calculate the PRR within a certain window size. As shown in Fig. 1 for an example of preprocessing within the window size = 10, we calculated the average RSSI and SNR values, and PRR according to the sequence number.

Based on data analysis, we expect to use the average of RSSI and SNR values that could help us to estimate the link quality level. As shown in Fig. 2a, the RSSI

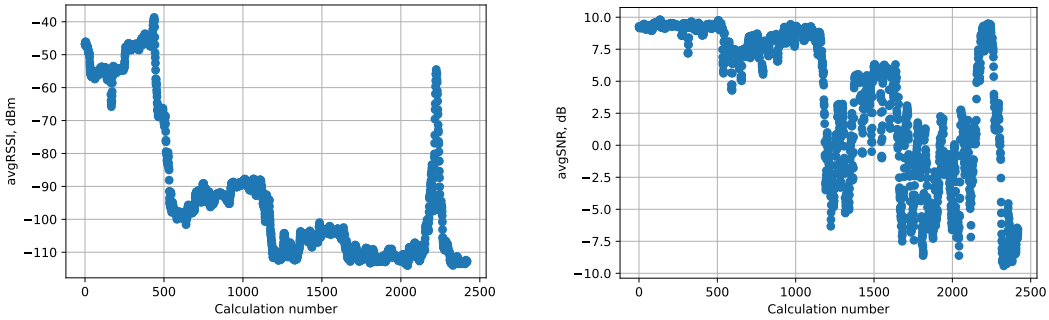


srcAddr	seqNum	payload	packetSize	RSSI	SNR
11	1	Hello world	16	-46	9.5
11	2	Hello world	16	-42	9.5
11	3	Hello world	16	-55	9.5
11	4	Hello world	16	-47	8.75
11	5	Hello world	16	-49	9.5
11	8	Hello world	16	-46	9
11	9	Hello world	16	-44	9.25
11	10	Hello world	17	-45	9.25
11	11	Hello world	17	-46	9
11	12	Hello world	17	-46	9.25

srcAddr	avgRSSI	SNR	PRR
11	-46.75	9.28125	0.8
11	-46.75	9.21875	0.8

Fig. 1. Data prepossessing with the window size = 10

values vary from -120 to -40 dBm and can be divided into four groups. On the other hand, the SNR values also change from -10 to 10 dB (Fig. 2b). Thus, approximately from this range, there are four levels that we expect to consider as link quality levels.



(a) The average RSSI

(b) The average SNR

Fig. 2. Preprocessed RSSI and SNR values

We consider four levels to assess the link quality, as shown in Tab. 1. The packet reception rate was used to label the link quality level [13].

Link Quality Level	PRR
Very good	$PRR \geq 0.9$
Good	$0.75 \leq PRR < 0.9$
Intermediate	$0.45 \leq PRR < 0.75$
Bad	$PRR < 0.45$

Table 1. LoRa link estimation classification range

In order to reduce the impact of range and model error, the avgRSSI and avgSNR are normalized so that the data are between 0 and 1. The normalization process corresponds to equation (3) as follows:

$$\begin{aligned} avgRSSI^* &= \frac{avgRSSI - min(avgRSSI)}{max(avgRSSI) - min(avgRSSI)} \\ avgSNR^* &= \frac{avgSNR - min(avgSNR)}{max(avgSNR) - min(avgSNR)} \end{aligned} \tag{3}$$

where $avgRSSI^*$ and $avgSNR^*$ are the normalized data that are used in the machine learning model.

3.2. LQE model based on SVM. The link quality estimation is converted to the multi-class classification problem, which can be solved using supervised learning models. Support vector machine (SVM) [14] is known as one of the efficient supervised learning models for solving multi-class classification. This paper proposes using SVM to train the collected data and predict the link quality level. Fig. 3 present the SVM structure using three parameters (avgRSSI, avgSNR, PRR) as the input data and Gaussian kernel for training. We used the programming language Python, and the library scikit-learn [15] to train and evaluate the proposed model. The dataset was divided into 10% for training and 90% for testing. We performed 50 training times to received the mean accuracy of the proposed model. The obtained results are shown in Tab. 2.

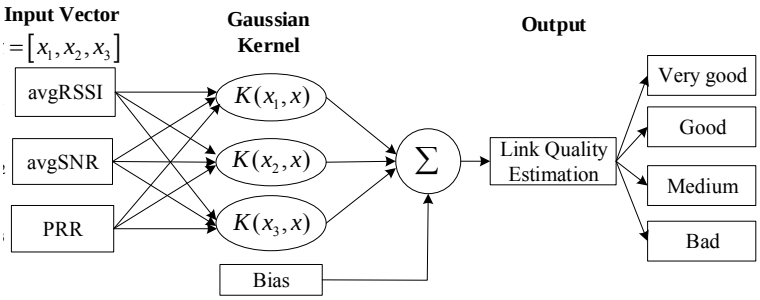


Fig. 3. The LQE model based on SVM

According to the evaluation results, the SVM model presents a high prediction accuracy with the mean = 95%. Thus, using three parameters such as {RSSI, SNR, PRR} the LoRa link quality can be estimated accurately using 10 probe packets,

LQE	Precision	Recall	F1-Score
Very good	0.96	1.00	0.98
Good	0.93	0.93	0.93
Medium	0.98	0.92	0.95
Bad	1.00	0.87	0.93
Accuracy		95%	

Table 2. The performance of link quality estimation model based on SVM

even though our dataset size is not large enough for training as usually can be seen in machine learning problems.

Moreover, it should be noted that we considered only one direction transmission (uplink). So the link quality level can be used to find the reliable uplink path in the LoRa mesh network.

4. Conclusion

In this paper, we considered a method for estimating the LoRa link quality. Link quality estimation is developed based on some metrics such as RSSI, SNR, PRR. The estimation problem was converted to a multi-class classification problem, which can be solved by using a kernelized support vector machine. An SVM-based LQE model was proposed and trained with collected data from the experiments. Despite using the small dataset for training and the number of probe packets, the proposed model has shown a high prediction accuracy (mean = 95%).

Acknowledgment

The publication has been prepared with the support of the grant from the President of the Russian Federation for state support of leading scientific schools of the Russian Federation according to the research project SS-2604.2020.9.

REFERENCES

1. A. Koucheryavy, A. Prokopiev, Y. Koucheryavy, Self-organizing networks, SPb.: Lyubavich 312 (2011).
2. D. Kandris, C. Nakas, D. Vomvas, G. Koulouras, Applications of wireless sensor networks: an up-to-date survey, Applied System Innovation 3 (1) (2020) 14.
3. S. Bertoldo, L. Carosso, E. Marchetta, M. Paredes, M. Allegretti, Feasibility analysis of a lora-based wsn using public transport, Applied System Innovation 1 (4) (2018) 49.

4. A. Proskochylo, A. Vorobyov, M. Zriakhov, A. Kravchuk, A. Akulynichev, V. Lukin, Overview of wireless technologies for organizing sensor networks, in: 2015 Second International Scientific-Practical Conference Problems of Infocommunications Science and Technology (PIC S&T), IEEE, 2015, pp. 39–41.
5. R. Kirichek, V. Vishnevsky, V. D. Pham, A. Koucheryavy, Analytic model of a mesh topology based on lora technology, in: 2020 22nd International Conference on Advanced Communication Technology (ICACT), IEEE, 2020, pp. 251–255.
6. V. D. Pham, V. Kisel, R. Kirichek, A. Koucheryavy, A. Shestakov, Evaluation of a mesh network based on LoRa technology, in: 2021 23rd International Conference on Advanced Communication Technology (ICACT), IEEE, 2021. doi:10.23919/icact51234.2021.9370792.
7. J. Luo, L. Yu, D. Zhang, Z. Xia, W. Chen, A new link quality estimation mechanism based on lqi in wsn, Information Technology Journal 12 (8) (2013) 1626.
8. N. Reijers, G. Halkes, K. Langendoen, Link layer measurements in sensor networks, in: 2004 IEEE International Conference on Mobile Ad-hoc and Sensor Systems (IEEE Cat. No. 04EX975), IEEE, 2004, pp. 224–234.
9. Z.-Q. Guo, Q. Wang, M.-H. Li, J. He, Fuzzy logic based multidimensional link quality estimation for multi-hop wireless sensor networks, IEEE Sensors Journal 13 (10) (2013) 3605–3615.
10. N. Baccour, D. Puccinelli, T. Voigt, A. Koubaa, C. Noda, H. Fotouhi, M. Alves, H. Youssef, M. A. Zuniga, C. A. Boano, K. Römer, Overview of link quality estimation, in: SpringerBriefs in Electrical and Computer Engineering, Springer International Publishing, 2013, pp. 65–86. doi:10.1007/978-3-319-00774-8_3.
11. P. Sun, H. Zhao, D. Luo, X.-y. Zhang, J. Zhu, Study on measurement of link communication quality in wireless sensor networks, Journal-China Institute of Communications 28 (10) (2007) 14.
12. R. Kirichek, A. Koucheryavy, Internet of things laboratory test bed, in: Wireless Communications, Networking and Applications, Springer India, 2015, pp. 485–494. doi:10.1007/978-81-322-2580-5_44.
13. J. Shu, S. Liu, L. Liu, L. Zhan, G. Hu, Research on link quality estimation mechanism for wireless sensor networks based on support vector machine, Chinese Journal of Electronics 26 (2) (2017) 377–384. doi:10.1049/cje.2017.01.013.
14. A. Patle, D. S. Chouhan, SVM kernel functions for classification, in: 2013 International Conference on Advances in Technology and Engineering (ICATE), IEEE, 2013. doi:10.1109/icadte.2013.6524743.
15. L. Parisi, m-arcsinh: An efficient and reliable function for svm and mlp in scikit-learn, arXiv preprint arXiv:2009.07530 (2020).

UDC: 519.872

Development of Radio Admission Scheme Model for 5G Network Slicing Framework as a Retrial Queue

Faina Moskaleva[1], Ekaterina Lisovskaya[1], Lyubov Lapshenkova[1], Sergey Shorgin[2], Yuliya Gaidamaka[1,2]

[1]Peoples' Friendship University of Russia (RUDN University), 6 Miklukho-Maklaya St, Moscow, 117198, Russian Federation

[2]Federal Research Center "Computer Science and Control" of the Russian Academy of Sciences (FRC CSC RAS), 44-2 Vavilov St, Moscow, 119333, Russian Federation

moskaleva-fa@rudn.ru, lisovskaya-eyu@rudn.ru, 1032172790@rudn.ru, sshorgin@ipiran.ru, gaydamaka-yuv@rudn.ru

Abstract

To improve the efficiency of using network resources, fifth-generation 5G networks propose to use the technology of network slicing. This feature consists of creating multiple logical, self-contained networks on top of a common shared physical infrastructure, and, therefore, it can be used to support multi-tenancy on the 5G network. Each of these logical networks is referred to as a network slice and can be tailored to provide a particular system behavior to best support specific service/application domains. This work is devoted to the development of a mathematical model of resource allocation in network slicing. Using the first-order asymptotic analysis method, we will find basic numerical and probabilistic characteristics.

Keywords: access control, radio resource slicing, performance measure, queuing system

1. Introduction

Network slicing is defined as one of the main components of fifth-generation mobile communications that can solve the problem of colossal growth in data volume traffic in cellular networks [1, 2]. The key feature of network slicing for ensuring performance and high quality of service is isolation, which limits the influence of slices on each other. Isolation is a fundamental property of network slicing that

This paper has been supported by the RUDN University Strategic Academic Leadership Program. The reported study was funded by RFBR, project numbers 19-07-00933 and 20-07-01064.

provides performance and security guarantees for each client when different clients use network segments for services with conflicting performance requirements [3]. Using isolation and resource sharing strategies on the radio interface is a rather entangled process [4].

The paper proposes one of the possible schemes for dividing radio resources on a single slice between two arrival processes. To describe and analyze the efficiency indicators of the proposed scheme, a retrial queueing system was used.

2. Mathematical Model

Consider one network segment receiving two arrival processes of service requests. Let two Poisson processes, corresponding to requests for data transmission from users of two different slices, arrive at the retrial queueing system (RQ system).The total capacity of the system is C resource units. The arrivals intensities are constant and equal to λ_1 and λ_2, respectively. Each request arriving from the k-process, $k = 1, 2$, requires b_k resource units to serve. If there is enough free resource in the system, then the request gets up for service, the duration of which is exponentially distributed with the parameter μ_k. If the free resource in the system turns out to be insufficient for servicing, then the request goes to the orbit, where it carries out an exponentially distributed random delay with the parameter σ_k, after which it makes the next attempt to get up for service. The orbits for each of the incoming streams are independent and have unlimited capacity.

We define a stochastic process $\mathbf{X}(t) = \{N_1(t), N_2(t), I_1(t), I_2(t)\}$, where $N_k(t)$ is the customers number of the k-process in the service at the time t, $t > 0$, and $I_k(t)$ is the customers number of the k-process in the orbit at the time t, $t > 0$, $k = 1, 2$. Then the states space of the process has the form:

$$\mathbb{X} = \{(n_1, n_2, i_1, i_2) : b_1 n_1 + b_2 n_2 \leq C,\, i_k \geq 0,\, k = 1, 2\},$$

where n_k is current state of the process $N_k(t)$, i_k is current state of the process $I_k(t)$.

The state spaces of customers blocking (i.e. the arrivals go to the orbit): $\mathbb{B}_1 = \{(n_1, n_2, i_1, i_2) :\, b_1(n_1+1)+b_2 n_2 \geq C\}$, $\mathbb{B}_2 = \{(n_1, n_2, i_1, i_2) :\, b_1 n_1+b_2(n_2+1) \geq C\}$, accordingly, the state spaces of accepting customers are $\tilde{\mathbb{B}}_k = \mathbb{X}\backslash\mathbb{B}_k$.

The condition for the existence of a steady-state regime in the system under consideration has the form:

$$\frac{\lambda_1}{\mu_1} b_1 + \frac{\lambda_2}{\mu_2} b_2 < C.$$

3. System of Equilibrium Equations

Let us write a system of equilibrium equations to obtain the stationary probability distribution of states $P(n_1, n_2, i_1, i_2)$, $(n_1, n_2, i_1, i_2) \in \mathbb{X}$ of the process $\mathbf{X}(t)$:

$$[\lambda_1 + \lambda_2 + n_1\mu_1 + n_2\mu_2 + i_1\sigma_1 \cdot I(b_1(n_1+1) + b_2n_2 \le C, i_1 > 0) +$$
$$+ i_2\sigma_2 \cdot I(b_1n_1 + b_2(n_2+1) \le C, i_2 > 0)] \cdot P(n_1, n_2, i_1, i_2) =$$
$$= \lambda_1 \cdot I(n_1 > 0) \cdot P(n_1 - 1, n_2, i_1, i_2) + \lambda_2 \cdot I(n_2 > 0) \cdot P(n_1, n_2 - 1, i_1, i_2) +$$
$$+ \lambda_1 \cdot I(b_1(n_1+1) + b_2n_2 > C, i_1 > 0) \cdot P(n_1, n_2, i_1 - 1, i_2) +$$
$$+ \lambda_2 \cdot I(b_1n_1 + b_2(n_2+1) > C, i_2 > 0) \cdot P(n_1, n_2, i_1, i_2 - 1) +$$
$$+ (n_1+1)\mu_1 \cdot P(n_1+1, n_2, i_1, i_2) \cdot I(b_1(n_1+1) + b_2n_2 \le C) +$$
$$+ (n_2+1)\mu_2 \cdot P(n_1, n_2+1, i_1, i_2) \cdot I(b_1n_1 + b_2(n_2+1) \le C) +$$
$$+ (i_1+1)\sigma_1 \cdot I(n_1 > 0) \cdot P(n_1 - 1, n_2, i_1 + 1, i_2) +$$
$$+ (i_2+1)\sigma_2 \cdot I(n_2 > 0) \cdot P(n_1, n_2 - 1, i_1, i_2 + 1).$$

At the next step, we write separately the form of the system of equilibrium equations for the subsets of states:

- $(1) : b_1n_1 + b_2n_2 = 0$, when the system is empty,
- $(2) : (b_1(n_1+1) + b_2n_2 \le C) \cap (b_1n_1 + b_2(n_2+1) \le C)$, when it is possible to accept customers from both processes,
- $(3) : (b_1(n_1+1) + b_2n_2 > C) \cap (b_1n_1 + b_2(n_2+1) > C)$, when it is not possible to accept customers from both processes,
- $(4) : (b_1(n_1+1) + b_2n_2 > C) \cap (b_1n_1 + b_2(n_2+1) \le C)$, when it is not possible to accept customers from 1-process, but possible for 2-process,
- $(5) : (b_1(n_1+1) + b_2n_2 \le C) \cap (b_1n_1 + b_2(n_2+1) > C)$, when it is possible to accept customers from 1-process, but not possible for 2-process.

4. Partial Characteristic Functions

Then, we introduce the partial characteristic functions:

$$H(n_1, n_2, u_1, u_2) = \sum_{i_1=0}^{\infty} \sum_{i_2=0}^{\infty} e^{ju_1i_1} e^{ju_2i_2} P(n_1, n_2, i_1, i_2), \text{ where } j = \sqrt{-1},$$

and rewrite the system for partial characteristic functions for subsets of states. Then, we denote $\mathbf{H}(u_1, u_2)$ as matrix of functions $H(n_1, n_2, u_1, u_2)$ and rewrite the system as operator equation:

$$(\mathbf{A} + \lambda_1 e^{ju_1}\mathbf{B_1} + \lambda_2 e^{ju_2}\mathbf{B_2})\mathbf{H}(u_1, u_2) +$$
$$+ j\sigma_1(\mathbf{C_1} - e^{-ju_1}\mathbf{D_1})\frac{\partial \mathbf{H}(u_1, u_2)}{\partial u_1} + j\sigma_2(\mathbf{C_2} - e^{-ju_2}\mathbf{D_2})\frac{\partial \mathbf{H}(u_1, u_2)}{\partial u_2} = 0, \quad (1)$$

where $\mathbf{A}, \mathbf{B_1}, \mathbf{B_2}, \mathbf{C_1}, \mathbf{C_2}, \mathbf{D_1}, \mathbf{D_2}$ are operators, which are specified in the following form:

$$\mathbf{B_1H}(u_1,u_2)=\begin{cases} 0, (1), (2), (5) \\ H(n_1,n_2,u_1,u_2), (3), (4) \end{cases} \mathbf{B_2H}(u_1,u_2)=\begin{cases} 0, (1), (2), (4) \\ H(n_1,n_2,u_1,u_2), (3), (5) \end{cases}$$

$$\mathbf{C_1H}(u_1,u_2) = \begin{cases} H(n_1,n_2,u_1,u_2), & (1), (2), (5) \\ 0, & (3), (4) \end{cases}$$

$$\mathbf{C_2H}(u_1,u_2) = \begin{cases} H(n_1,n_2,u_1,u_2), & (1), (2), (4) \\ 0, & (3), (5) \end{cases}$$

$$\mathbf{D_1H}(u_1,u_2) = \begin{cases} 0, & (1) \\ H(n_1-1,n_2,u_1,u_2), & (2)-(5) \end{cases}$$

$$\mathbf{D_2H}(u_1,u_2) = \begin{cases} 0, & (1) \\ H(n_1,n_2-1,u_1,u_2), & (2)-(5) \end{cases}$$

$$\mathbf{AH}(u_1,u_2) = \begin{cases} -(\lambda_1+\lambda_2)H(n_1,n_2,u_1,u_2)+\mu_1 H(n_1+1,n_2,u_1,u_2)+ \\ +\mu_2 H(n_1,n_2+1,u_1,u_2), & (1) \\ -(\lambda_1+\lambda_2+n_1\mu_1+n_2\mu_2)H(n_1,n_2,u_1,u_2)+ \\ +\lambda_1 H(n_1-1,n_2,u_1,u_2)+\lambda_2 H(n_1,n_2-1,u_1,u_2)+ \\ +(n_1+1)\mu_1 H(n_1+1,n_2,u_1,u_2)+ \\ +(n_2+1)\mu_2 H(n_1,n_2+1,u_1,u_2), & (2) \\ -(\lambda_1+\lambda_2+n_1\mu_1+n_2\mu_2)H(n_1,n_2,u_1,u_2)+ \\ +\lambda_1 H(n_1-1,n_2,u_1,u_2)+\lambda_2 H(n_1,n_2-1,u_1,u_2), & (3) \\ -(\lambda_1+\lambda_2+n_1\mu_1+n_2\mu_2)H(n_1,n_2,u_1,u_2)+ \\ +\lambda_1 H(n_1-1,n_2,u_1,u_2)+\lambda_2 H(n_1,n_2-1,u_1,u_2)+ \\ +(n_2+1)\mu_2 H(n_1,n_2+1,u_1,u_2), & (4) \\ -(\lambda_1+\lambda_2+n_1\mu_1+n_2\mu_2)H(n_1,n_2,u_1,u_2)+ \\ +\lambda_1 H(n_1-1,n_2,u_1,u_2)+\lambda_2 H(n_1,n_2-1,u_1,u_2)+ \\ +(n_1+1)\mu_1 H(n_1+1,n_2,u_1,u_2), & (5) \end{cases}$$

Let us define $\mathbf{E}$ as an operator that sums functions overall available values of n_1, n_2 and represents the following additional scalar equation:

$$\mathbf{E}(\lambda_1(e^{ju_1}-1)\mathbf{B_1}+\lambda_2(e^{ju_2}-1)\mathbf{B_2})\mathbf{H}(u_1,u_2)+$$
$$+j\sigma_1(1-e^{-ju_1})\mathbf{ED_1}\frac{\partial \mathbf{H}(u_1,u_2)}{\partial u_1}+j\sigma_2(1-e^{-ju_2})\mathbf{ED_2}\frac{\partial \mathbf{H}(u_1,u_2)}{\partial u_2}=0. \quad (2)$$

5. First Order Approximation

We find the solution of (1)–(2) using the asymptotic analysis [5] under the condition of proportional to the increasing delay time in orbits (i.e. $\sigma_k = \sigma \cdot \gamma_k$, $\sigma \to 0$, $k = 1, 2$). As a result of first order asymptotic analysis we obtain the first order approximation of matrix characteristic function:

$$\mathbf{H}(u_1, u_2) = \mathbf{R} \cdot \exp\left\{ju_1 \frac{x_1}{\sigma} + ju_2 \frac{x_2}{\sigma}\right\},$$

where $\mathbf{R}$ is a matrix of elements $R(n_1, n_2)$ – a stationary joint probability distribution of the two-dimensional process $\{N_1(t), N_2(t)\}$, and x_1, x_2 are the normalized customers numbers means in orbits. The parameters are defined from:

$$[(\mathbf{A} + \lambda_1 \mathbf{B_1} + \lambda_2 \mathbf{B_2}) - x_1\gamma_1(\mathbf{C_1} - \mathbf{D_1}) - x_2\gamma_2(\mathbf{C_2} - \mathbf{D_2})]\,\mathbf{R} = 0,$$

$$\mathbf{E}\,[\lambda_1 \mathbf{B_1} - x_1\gamma_1 \mathbf{D_1}]\,\mathbf{R} = 0, \quad \mathbf{E}\,[\lambda_2 \mathbf{B_2} - x_2\gamma_2 \mathbf{D_2}]\,\mathbf{R} = 0, \quad \mathbf{ER} = 1.$$

At this step, we obtain the marginal probability distribution of customers number in the service $\mathbf{R}$ and the means of customers number in the orbits $\frac{x_k}{\sigma}$, $k = 1, 2$.

6. Numerical Example

We can calculate the main numerical characteristics of system performance. Let the system parameters has the form: $C = 5$, $\lambda_1 = \lambda_2 = 2$, $\mu_1 = \mu_2 = 2$, $\sigma_1 = \sigma_2 = 3 \cdot \sigma\,(\sigma \to 0)$, $b_1 = 1$, $b_2 = 2$. Figure 1 shows the changes in means of customers' numbers in the orbits and their relative errors with simulation mean. Other characteristics can be easily calculated using the stationary probability distribution:

- probability that arrival goes to the orbit (probabilistic characteristic)

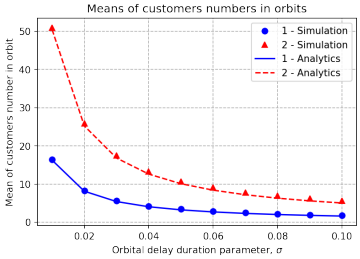
$$p_k = \sum_{(n_1, n_2, i_1, i_2) \in \mathbb{B}_k} R(n_1, n_2),\ k = 1, 2,$$

- means of the total resource amounts occupied (numerical characteristic)

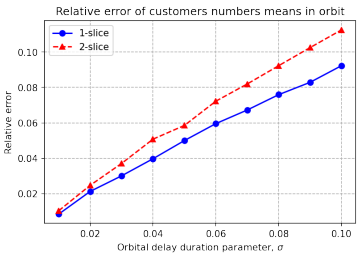
$$\mathrm{E}[B_k] = \sum_{(n_1, n_2, i_1, i_2) \in \mathbb{X}} b_k n_k R(n_1, n_2),\ k = 1, 2,$$

- utility coefficient (numerical characteristic)

$$R = (\mathrm{E}[B_1] + \mathrm{E}[B_2])/C.$$



(a) Means of customers numbers in orbits



(b) Relative error

Fig. 1. Visualization of approximation accuracy

7. Conclusion

In this paper, we formalized the mathematical model of resource allocation in network slicing. Using the first-order asymptotic analysis method, we found basic numerical and probabilistic characteristics. Next, we plan to perform a second-order asymptotic analysis to obtain a four-dimensional probability distribution that will allow us to calculate a wider range of performance metrics for network slicing technology. In addition, it is necessary to consider a model with three or more arrival processes (slices).

REFERENCES

1. E. Mokrov, A. Ponomarenko-Timofeev, I. Gudkova, P. Masek, J. Hosek, S. Andreev, Y. Koucheryavy, Y. Gaidamaka, Modeling Transmit Power Reduction for a Typical Cell With Licensed Shared Access Capabilities, IEEE Transactions on Vehicular Technology 67 (6) (2018) 5505–5509. doi:10.1109/TVT.2018.2799141.
2. A. Ometov, D. Kozyrev, V. Rykov, S. Andreev, Y. Gaidamaka, Y. Koucheryavy, Reliability-Centric Analysis of Offloaded Computation in Cooperative Wearable Applications, Wireless Communications and Mobile Computing 2017 (2017) 1–15. doi:10.1155/2017/9625687.
3. N. Yarkina, Y. Gaidamaka, L. M. Correia, K. Samouylov, An Analytical Model for 5G Network Resource Sharing with Flexible SLA-Oriented Slice Isolation, Mathematics 8 (7) (2020). doi:10.3390/math8071177.
4. M. Richart, J. Baliosian, J. Serrat, J. Gorricho, Resource Slicing in Virtual Wireless Networks: A Survey, IEEE Transactions on Network and Service Management 13 (3) (2016) 462–476. doi:10.1109/TNSM.2016.2597295.
5. A. Nazarov, T. Phung-Duc, Y. Izmailova, Multidimensional Central Limit Theorem of the Multiclass M/M/1/1 Retrial Queue, Lecture Notes in Computer Science 12563 (4) (2020) 298–310. doi:10.1007/978-3-030-66471-8_23.

UDC: 519.217

Reliability Model of a Homogeneous Hot-Standby k-out-of-n System

H.G.K. Houankpo[1], D.V. Kozyrev [1,2], E. Nibasumba[1], M.N.B. Mouale[1]

[1]Department of Applied Probability and Informatics, Peoples' Friendship University of Russia (RUDN University), 6 Miklukho-Maklaya St, Moscow, 117198, Russian Federation

[2]V.A.Trapeznikov Institute of Control Sciences of Russian Academy of Sciences, 65 Profsoyuznaya street, Moscow, 117997, Russia

gibsonhouankpo@yahoo.fr, kozyrev-dv@rudn.ru, ema.patiri2015@yandex.ru, bmouale@mail.ru

Abstract

We propose a mathematical model of a closed homogeneous redundant hot-standby system. The system consists of a single repair unit and an arbitrary number of unreliable data sources with an exponential distribution function (DF) of uptime and a general independent (GI) DF of the repair time. Explicit analytic expressions have been obtained for the steady-state probabilities (SSP) of the system and the SSP of failure-free system operation.

Keywords: system reliability, steady-state probabilities, sensitivity analysis, mathematical modeling, hybrid data transmission systems

Notations

A – random variable, time to failure of the main component,
B – random variable, recovery time of a failed component,
$A(x)$ – DF of the random variable A,
$B(x)$ – DF of the random variable B,
$b(x)$ – probability density function (PDF) of the random variable B,
$\tilde{b}(\lambda_i) = \int_0^\infty e^{-sx} b(x)dx$ – Laplace transform of the PDF $b(x); i = \overline{0, k-1}$,
EA – mean uptime of a working component ,
$EB = b$ – mean repair time of a failed component,
DB – recovery time variance,
$c = \frac{\sqrt{DB}}{EB}$ – coefficient of variation,
$\rho = \frac{EA}{EB}$ – relative recovery rate,
$\delta(x) = \frac{b(x)}{1-B(x)}$ – conditional PDF of the residual repair duration of the element being repaired at time t (recovery rate) [15],

$\lambda_i = (n-i)\alpha$ – parameter of the exponential distribution of the uptime of components; $i = \overline{0, k-1}$.

1. Introduction

Computer and communications networks are constantly evolving due to the results of theoretical and practical problems aimed at improving the availability and reliability of networks and data transmission systems. Researchers are often faced with the development of complex systems including the k-out-of-n type systems.

There are a number of important studies on different k-out-of-n systems. Cao Wang [1] proposed the performance of civil infrastructure using k-out-of-n systems with identical component deterioration. Xinchen Zhuang, Tianxiang Yu, Zhongchao Sun and Kunling Song [2] focused on reliability and capacity evaluation of multi-performance multi-state weighted k-out-of-n systems. A recursive approach was developed to evaluate the system reliability more efficiently, and comparison with the existing approach was carried out in various aspects. Tetsushi Yuge [3] investigated the reliability of systems with simultaneous and consecutive failures, where the reliability of a k-out-of-n system and a consecutive k-out-of-n system subjected to shock and considering the simultaneous failures were discussed. Eunkyung Chae, Chan-woo Park, Jeon-gwon Kang [4] studied the reliability analysis of a M-out-of-N system with common cause failures for railway, where the reliability of a hot-standby sparing system considering common cause failure has been analyzed and used to improve the accuracy of the system reliability evaluation. Huyang Xu, Yuanchen Fan, Nasser Fard [5] performed the reliability assessment of repairable load-sharing K-out-of-N systems with flowgraph model.

In our previous works [6, 7, 8, 9, 10], we focused on respectively a simulation approach to reliability assessment of a redundant system with arbitrary input distributions, the reliability analysis of a homogeneous hot standby data transmission system, the mathematical model for reliability analysis of a heterogeneous redundant data transmission system, the reliability model of a homogeneous warm-standby data transmission system with general repair time distribution and the sensitivity analysis of steady state reliability characteristics of a repairable cold standby data transmission system to the shapes of lifetime and repair time distributions of its elements. In these papers, we utilized the supplementary variable method to investigate the sensitivity of steady-state reliability characteristics of some special cases of systems. Further, in [11] we applied the k-out-of-n system model to the reliability study of a hexacopter-based flight module of a tethered unmanned high-altitude platform.

In the current paper, we study the mathematical model of a closed homogeneous hot-standby k-out-of-n system. The purpose of this work is to carry out mathematical

modeling of the system, obtain the analytical expressions for its SSP and carry out numerical analysis for a specific example.

2. Mathematical model of the system

2.1. Description, assumptions and problem statement. Let's consider a closed homogenous redundant hot standby system. The system consists of an arbitrary number of data sources with an exponential distribution function (DF) of uptime and a general independent DF of the repair time of its components with a single repair unit, which according to a modified Kendall's notation [12], we denote as $\langle M_{k<n}/GI/1\rangle$.

We study the dependence of the failure-free operation probability of the system on the relative recovery rate (RRR) with different values of the coefficient of variation. We introduce the following assumptions regarding the system's operation:

Assumption 1: initially backup elements participate in the functioning of the system on a par with the main element.

Assumption 2: failed elements are sent to repair one at a time.

Assumption 3: the system with n components fails if k components fail.

The aim is to find the explicit analytical expressions for the steady-state probabilities distribution of the system and for the steady-state probability of the failure-free system operation, both in the general case and for some special cases of distributions.

2.2. Explicit analytic expressions. Consider a stochastic process $v(t)$ — the number of failed elements at time t, with a set of states $E = \{0, 1, 2, \ldots, k\}$.

To solve the stated problem, we use an approach based on the Markovization principle [13]. To describe the behavior of the system using a Markov process, we introduce an additional variable $x(t) \in \mathbb{R}^2_+$ — the overall duration spent at time t for recovery of the failed element. We obtain a two dimensional [14] process $(v(t), x(t))$, with an extended phase space $\epsilon = \{(0), (1, x), (2, x), \ldots, (k, x)\}$.

We denote by $p_0(t)$ the probability that at time t the system is in state $i = 0$, and by $p_i(t, x)$ the probability density function (in continuous component) that at time t the system is in state $i\,(i = 1, 2, \ldots, k)$, and the time taken to repair the failed element is in the range $(x, x + dx)$.

$$p_0(t) = \mathbb{P}\{v(t) = 0\}, \tag{1}$$

$$p_i(t, x)dx = \mathbb{P}\{v(t) = i, x < x(t) < x + dx\}, i = \overline{1, k}. \tag{2}$$

Theorem 1. The steady-state probabilities of the considered repairable redundant system are:

$$p_0 = C_1 \cdot \frac{\tilde{b}(\lambda_1)}{\lambda_0};\ p_1 = C_1 \cdot \Phi_1;$$

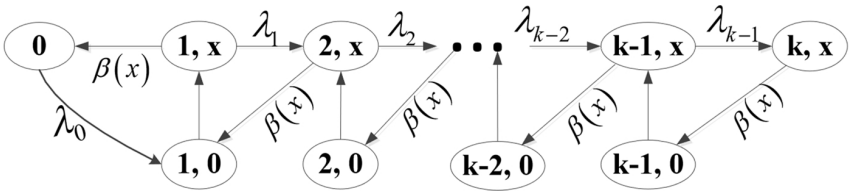


Fig. 1. State transition graph

$$p_i = C_1 \left(A_i \Phi_i + \sum_{j=1}^{i-1} (-1)^{i-j} \left(\prod_{k=j}^{i-1} \frac{\lambda_k}{\lambda_j - \lambda_{k+1}} \right) A_j \Phi_j \right);\ i = \overline{2, k-1}; k \geqslant 3;$$

$$p_k = \begin{cases} C_1 \left(A_k b - A_{k-1} \Phi_{k-1} \right); k = 2; \\ C_1 \left(A_k b - A_{k-1} \Phi_{k-1} + \sum_{j=1}^{k-2} (-1)^{k-j} \left(\prod_{i=j}^{k-2} \frac{\lambda_{i+1}}{\lambda_j - \lambda_{i+1}} \right) A_j \Phi_j \right); k \geqslant 3; \end{cases}$$

Proof. From (1) and (2), with the help of the total probability rule we move to the Kolmogorov's forward system of differential equations, which makes it possible to find the stationary state probabilities of the considered system. Using the total probability formula, and passing to the limit as $\Delta \to 0$, we derive the following Kolmogorov differential equations systems:

$$\begin{cases} \lambda_0 p_0 = \int_0^\infty p_1(x)\delta(x)dx \\ \frac{dp_1(x)}{dx} = -(\lambda_1 + \delta(x))p_1(x) \\ \frac{dp_i(x)}{dx} = -(\lambda_i + \delta(x))p_i(x) + \lambda_{i-1}p_{i-1}(x);\ i = \overline{2, k-1} \\ \frac{dp_k(x)}{dx} = -p_k(x)\delta(x) + \lambda_{k-1}p_{k-1}(x) \end{cases} \tag{3}$$

with boundary condition

$$p_1(0)dx = \lambda_0 p_0 + \int_0^\infty p_2(x)\delta(x)dx, \tag{4}$$

$$p_i(0)dx = \int_0^\infty p_{i-1}(x)\delta(x)dx;\ i = \overline{2, k-1}. \tag{5}$$

This system allows us to find the probabilities of the states [16] of the system in question. We suppose that for the described process, there exists a stationary probability distribution as $t \to \infty$.

We proceed to solving the resulting system of balance equations using the constant variation method [17]. From here we find the stationary probabilities for macro-states. As a result, we obtain the following analytical expressions for the SSP of the repairable system in the following form:

$$p_0 = C_1 \cdot \frac{\tilde{b}(\lambda_1)}{\lambda_0}; p_1 = C_1 \cdot \Phi_1;$$

$$p_i = C_1 \left(A_i \Phi_i + \sum_{j=1}^{i-1} (-1)^{i-j} \left(\prod_{k=j}^{i-1} \frac{\lambda_k}{\lambda_j - \lambda_{k+1}} \right) A_j \Phi_j \right); i = \overline{2, k-1}; k \geqslant 3;$$

$$p_k = \begin{cases} C_1 \left(A_k b - A_{k-1} \Phi_{k-1} \right); k = 2; \\ C_1 \left(A_k b - A_{k-1} \Phi_{k-1} + \sum_{j=1}^{k-2} (-1)^{k-j} \left(\prod_{i=j}^{k-2} \frac{\lambda_{i+1}}{\lambda_j - \lambda_{i+1}} \right) A_j \Phi_j \right); k \geqslant 3; \end{cases}$$

Where

$$C_1 = \left(\frac{\tilde{b}(\lambda_1)}{\lambda_0} + \Phi_1 + A_k b - A_{k-1} \Phi_{k-1} \right)^{-1}; k = 2;$$

$$C_1^{-1} = \frac{\tilde{b}(\lambda_1)}{\lambda_0} + \Phi_1 + \sum_{i=2}^{k-1} \left(A_i \Phi_i + \sum_{j=1}^{i-1} (-1)^{i-j} \left(\prod_{k=j}^{i-1} \frac{\lambda_k}{\lambda_j - \lambda_{k+1}} \right) A_j \Phi_j \right) +$$

$$+ \left(A_k b - A_{k-1} \Phi_{k-1} + \sum_{j=1}^{k-2} (-1)^{k-j} \left(\prod_{i=j}^{k-2} \frac{\lambda_{i+1}}{\lambda_j - \lambda_{i+1}} \right) A_j \Phi_j \right); k \geqslant 3;$$

$$\Phi_i = \frac{1 - \tilde{b}(\lambda_i)}{\lambda_i}$$

$$A_2 = A_1; k = 2$$

$$A_2 = \left(1 - \left(1 - \frac{\lambda_1}{\lambda_1 - \lambda_2} \right) \lambda_1 \right) \right) \frac{1}{\tilde{b}(\lambda_2)}; k \geqslant 3$$

$$A_{i+1} = (A_i + \sum_{j=1}^{i-1} (-1)^{i-j} \left(\prod_{k=j}^{i-1} \frac{\lambda_k}{\lambda_j - \lambda_{k+1}} \right) A_j -$$

$$- \sum_{j=1}^{i} (-1)^{i+1-j} \left(\prod_{k=j}^{i} \frac{\lambda_k}{\lambda_j - \lambda_{k+1}} \right) A_j \tilde{b}(\lambda_j)) \frac{1}{\tilde{b}(\lambda_{i+1})}; i = \overline{2, k-2}; k \geqslant 4$$

$$A_k = A_{k-1}(1 + \tilde{b}(\lambda_{k-1})) + \sum_{j=1}^{k-2} (-1)^{i-1-j} \left(\prod_{i=j}^{k-2} \frac{\lambda_i}{\lambda_j - \lambda_{i+1}} \right) A_j +$$

$$+ \sum_{j=1}^{k-2} (-1)^{k-j} \left(\prod_{i=j}^{k-2} \frac{\lambda_{i+1}}{\lambda_j - \lambda_{i+1}} \right) A_j \tilde{b}(\lambda_j); k \geqslant 3 \qquad \blacksquare$$

From these expressions it is evident there is a dependence of the SSP of the system states on the type of the repair time distribution. However, this dependence becomes vanishingly small with the "rapid" repair of the failed elements, i.e. with the increase in relative recovery rate [6, 7, 8, 9, 10, 11].

3. Example. Numerical analysis

To analyze the numerical results of the model of a redundant data transmission system with different known distributions of repair time, where $k = 3$ and $n = 5$, we consider special cases when $EB = b = 1$ with different coefficients of variation $c = \{0.5; 1; 2\}$; and $EA = 25$.

Table 1 shows the values of probability $1 - p_k$ of failure-free operation of the considered system .

GI \ c	0.5	1	2
Weibull	0,9985355	0,9968928	0,9911761
Pareto	0,9986520	0,9968809	0,9947930
Gamma	0,9988354	0,9968928	0,9861888
LogNormal	0,9988074	0,9965752	0,9869367

Table 1. Values of probability $1 - p_3$ of the system's failure-free operation

Obviously, for all the distributions under consideration, the greater is the coefficient of variation, the less is the probability of failure-free operation of the system.

Figure 1 shows the graphs of the probabilities of failure-free system operation. For graphical results, we consider models with $\rho = 25$

The graphical results also confirm the above conclusion that the greater is the coefficient of variation, the less is the probability of failure-free operation of the system, and the dependence of the SSP of the system states on the type of the repair time distribution becomes vanishingly small with the "rapid" repair of the failed elements.

4. Conclusion

In the current work, we obtained the explicit analytical expressions for the steady-state probabilities distribution of the system states and for the stationary probability of system uptime in the general case.

The obtained formulas show the presence of an explicit dependence of reliability measures of the system on the type of the distribution function of the repair time of its elements.

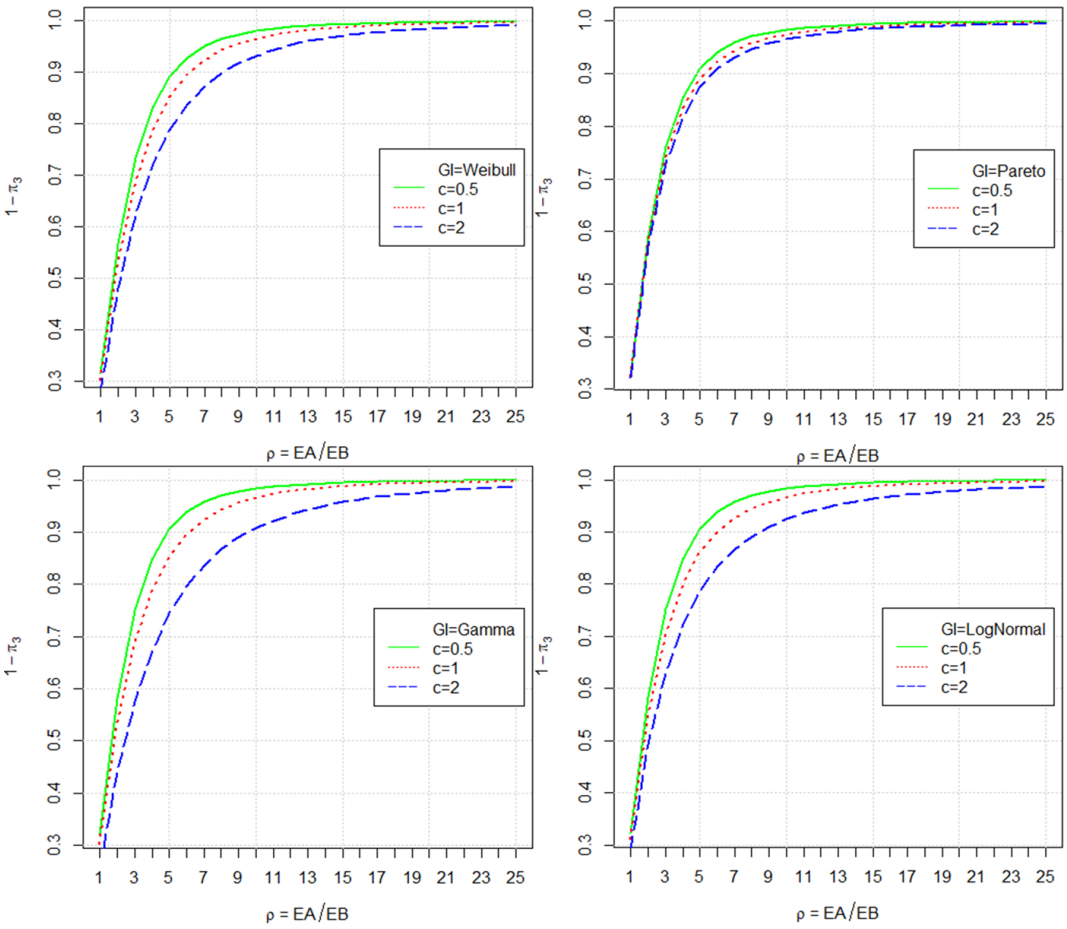


Fig. 2. Graphs of the probability of failure-free system operation $1 - p_3$ versus ρ

Numerical analysis showed that the greater is the coefficient of variation, the less is the probability of failure-free operation of the system.

The graphical results confirm the conclusion about the inverse relationship between the coefficient of variation and the probability of failure-free operation of the system, and about the dependence of the SSP of the system states on the type of the repair time distribution which becomes vanishingly small with the "rapid" repair of the failed elements.

Acknowledgments

The publication has been supported by the RUDN University Strategic Academic Leadership Program. The reported study was funded by RFBR, project No. 20-37-90137 (recipient Dmitry Kozyrev, formal analysis, validation, and recipient H.G.K. Houankpo, methodology and numerical analysis)

REFERENCES

1. Cao W. Time-dependent reliability of (weighted) k-out-of-n systems with identical component deterioration //J Infrastruct Preserv Resil 2, 3 (2021). DOI: 10.1186/s43065-021-00018-1
2. Zhuang X., Yu T., Sun Z., Song K. Reliability and capacity evaluation of multi-performance multi-state weighted k-out-of-n systems //Communications in Statistics - Simulation and Computation. DOI: 10.1080/03610918.2020.1788590
3. Tetsushi Y. Reliability of Systems with Simultaneous and Consecutive Failures // In: 2019 4th International Conference on System Reliability and Safety (ICSRS). DOI: 10.1109/ICSRS48664.2019.8987614
4. Eunkyung Chae, Chan-woo Park, Jeon-gwon Kang Reliability Analysis of M out of N System with Common Cause Failures for Railway // DOI: 10.7782/JKSR.2018.21.10.969
5. Xu H., Fan Y., Fard N. Reliability Assessment of Repairable Load-Sharing K-out-of-N: System with Flowgraph Model // DOI: 10.1109/RAM.2018.8463109
6. Houankpo H. G. K., Kozyrev D. V., Nibasumba E., Mouale M. N. B., Sergeeva I. A. A Simulation Approach to Reliability Assessment of a Redundant System with Arbitrary Input Distributions //In: Vishnevskiy V.M., Samouylov K.E., Kozyrev D.V. (eds) Distributed Computer and Communication Networks. DCCN 2020. Lecture Notes in Computer Science, vol 12563, pp.380–392. Springer, Cham. DOI: 10.1007/978-3-030-66471-8_29.
7. Houankpo H. G. K., Kozyrev D. V., Nibasumba E., Mouale M. N. B. Reliability Analysis of a Homogeneous Hot Standby Data Transmission System //In: Proceedings of the 30th European Safety and Reliability Conference and 15th Probabilistic Safety Assessment and Management Conference (ESREL2020 PSAM15), 2020, pp. 1–8.
8. Houankpo H. G. K., Kozyrev D. V., Nibasumba E., Mouale M. N. B. Mathematical Model for Reliability Analysis of a Heterogeneous Redundant Data Transmission System //2020 12th International Congress on Ultra Modern Telecommunications and Control Systems and Workshops (ICUMT), Brno, Czech Republic, 2020, pp. 189–194, DOI: 10.1109/ICUMT51630.2020.9222431

9. Houankpo H. G. K., Kozyrev D. V. Reliability Model of a Homogeneous Warm-Standby Data Transmission System with General Repair Time Distribution //In: Vishnevskiy V., Samouylov K., Kozyrev D. (eds) Distributed Computer and Communication Networks. DCCN 2019. Lecture Notes in Computer Science, vol 11965, pp.443–454. Springer, Cham. DOI: 10.1007/978-3-030-36614-8_34
10. Houankpo H. G. K., Kozyrev D. V. Sensitivity Analysis of Steady State Reliability Characteristics of a Repairable Cold Standby Data Transmission System to the Shapes of Lifetime and Repair Time Distributions of its Elements // In: K. E. Samouilov, L. A. Sevastianov, D. S. Kulyabov (eds.): Selected Papers of the VII Conference "Information and Telecommunication Technologies and Mathematical Modeling of High-Tech Systems", Moscow, Russia, 24-Apr-2017, CEUR Workshop Modelings 1995 (2017), p. 107–113. published at http://ceur-ws.org/Vol-1995/paper-15-970.pdf.
11. Kozyrev D. V., Phuong N. D., Houankpo H. G. K., Sokolov A. Reliability Evaluation of a Hexacopter-Based Flight Module of a Tethered Unmanned High-Altitude Platform // In: Vishnevskiy V., Samouylov K., Kozyrev D. (eds) Distributed Computer and Communication Networks. DCCN 2019. Communications in Computer and Information Science, vol 1141, pp.646–656. Springer, Cham. DOI: 10.1007/978-3-030-36625-4_52
12. Kendall D. G. Stochastic processes occurring in the theory of queues and their analysis by the method of embedded Markov chains // In: Ann. Math. Stat. 1953, 24, pp. 338–354.
13. Parshutina S., Bogatyrev V. Models to support design of highly reliable distributed computer systems with redundant processes of data transmission and handling // In: 2017 International Conference "Quality Management, Transport and Information Security, Information Technologies" (IT&QM&IS) (2017). DOI: 10.1109/ITMQIS.2017.8085772.
14. Teh T., Lai C. M., Cheng, Y. H. Impact of the real-time thermal loading on the bulk electric system reliability // IEEE Trans. Reliab. 66(4), 11101119 (2017). DOI: 10.1109/TR.2017.2740158.
15. Lisnianski A., Laredo D., Haim H. B. Multi-state Markov model for reliability analysis of a combined cycle gas turbine power plant // In: 2016 Second International Symposium on Stochastic Models in Reliability Engineering, Life Science and Operations Management (SMRLO) (2016). DOI: 10.1109/SMRLO.2016.31.
16. Sevastyanov B. A. An Ergodic theorem for Markov processes and its application to telephone systems with refusals // Theor. Probab. Appl. 1957, pp. 104–112.
17. Petrovsky I. G. Lectures on the theory of ordinary differential equations (Lektsii po teorii obyknovennykh differentsialnykh uravneniy), Moscow, GITTL, 232 p., 1952. (In Russian)

UDC: 519.6

Full Version for Algorithmic Analysis of Finite-Source Multi-Server Heterogeneous Queues.

D. Efrosinin[1,2,3], N. Stepanova[2], J. Sztrik[4]

[1]Johannes Kepler University Linz, Altenbergerstrasse 69, 4040 Linz, Austria
[2]V.A. Trapeznikov Institute of Control Sciences of RAS, Profsoyuznaya 65, 117997 Moscow, Russia
[3]Peoples' Friendship University of Russia (RUDN University), Miklukho-Maklaya 6, 117198 Moscow, Russia
[4]University of Debrecen, Egyetem tér 1, 4032 Debrecen, Hungary

dmitry.efrosinin@jku.at,natalia0410@rambler.ru,sztrik.janos@inf.unideb.hu

Abstract

The paper deals with a finite-source queueing system serving one class of customers and consisting of heterogeneous servers with unequal service intensities and of one common queue. The main model has a non-preemptive service when the customer can not change the server during its service time. The optimal allocation problem is formulated as a Markov-decision one. We show numerically that the optimal policy which minimizes the long-run average number of customers in the system has a threshold structure. We derive the matrix expressions for the mean performance measures and compare the main model with alternative simplified queuing systems which are analysed for the arbitrary number of servers. We observe that the preemptive heterogeneous model operating under a threshold policy is a good approximation for the main model by calculating the mean number of customers in the system. Moreover,using the preemptive and non-preemptive queueing models with the faster server first policy the lower and upper bounds are calculated for this mean value

Keywords: Finite-source queueing system, Preemptive and non-preemptive service, Markov-decision process, Policy-iteration algorithm, Performance analysis

1. Introduction

The finite-source or finite-population queueing systems comparing to the ordinary markovian queues have no longer a Poisson arrival stream as in systems with

Research is supported by the Austro-Hungarian Cooperation (OMAA) Grant No 106öu4, 2021. (recipient D. Efrosinin and J. Sztrik). This paper has been supported by the RUDN University Strategic Academic Leadership Program (recipient D. Efrosinin)

an infinite source of customers, but rather have a finite source capacity N of possible customers. In such systems a customer can be inside the system, consisting in our case of one common queue with capacity N and K heterogeneous servers or outside the system in so-called arriving state. It is assumed that each customer outside arrives to the system in exponentially distributed time. After receiving the service a customer returns to the arriving area. Much attention by the study of the finite-source queueing systems has been paid in terms of the machine repairman problem, see e.g. [8, 10]. The customers outside the queueing system can be interpreted as unreliable machines with independent exponentially distributed life times. The queueing system represents then the repair facility where the failed machines must be recovered. Such systems are also used in various dispatching problems, they are appropriate queueing models for telephone registration systems, call centers, Ethernet systems, local-area networks, mobile communications, magnetic disk memory systems and so on.

The main model of the paper is a non-preemptive controlled finite-source queueing system with one class of customers and heterogeneous servers. In such a system, a customer that receives service on a slower server cannot change it if a faster server becomes available in the course of service. Unfortunately, performance analysis of this system in analytical form is limited firstly by the need to have a known allocation mechanism between the servers or control policy and secondly by the dimensionality of the corresponding Markov process, which is affected by the number of servers. To calculate the optimal allocation policy with the aim to minimize the long-run average number of customers in the system we formulate the Markov decision process (MDP) and apply the policy-iteration algorithm. This algorithm can be used not only for the optimal allocation policy calculation but also to obtain the mean number of customers in the system operating under that policy. Numerical experiments confirm our expectations that the optimal policy is of threshold type as in the models with an infinite source capacity [3]. According to this policy the fastest server must be activated whenever there is a customer in the system while the slower servers must be used only if the number of customers in the queue reaches some prespecified threshold level. The model of the non-preemptive queue operating under the optimal threshold policy will referred to in the paper as the OTP-model.

The task of calculating other system performance characteristics for a given control policy remains unresolved. Furthermore, it should be taken into account that despite of advantages the policy-iteration algorithm has a limitation on the dimensionality of the random process for an arbitrary number of servers. In case of a threshold control policy for a particular states' ordering the corresponding Markov chain is a quasi-birth-and-death (QBD) process with a three-diagonal block infinitesimal matrix, where the blocks depend on the values of thresholds as it was shown in

[4] for the infinite population system. In this case, matrix-analytic solution methods can be applied, but for a limited number of servers. This led us to discuss here in addition some simplified variants of the main model. The non-preemptive queueing system operating under a Fastest Server First (FSF) policy which prescribes for service the usage of the fastest idle server in each state and the preemptive queueing system (PS), where the service in a slower server can be interrupted if during the service time the faster server becomes idle. This system will operates according to a threshold policy, when the slower servers are activated or deactivated if the number of customers in the queue respectively exceeds or falls below a certain threshold level. Although these systems are simpler than the main model and have a low dimensions of the state-space, there are very few publications on such specific systems, especially those with analytical results.

Description of standard finite-source models with classical results, motivation examples and literature overview can be found in [12]. In [7] the author obtained the product form solution for the stationary state distribution for the finite population queueing model with a queue-dependent servers. A non-preemptive finite-population queueing system with heterogeneous classes of customers and a single server was studied in [6]. The problem of the throughput maximizing in a finite-source system with parallel queues was analysed in [2], where some structural properties of the optimal control policy was proved. Heterogeneous multi-server finite-source queues with a FSF-policy and retrial phenomenon have been studied in [11], where numerical results were carried out by the help of the MOSEL tool. The model with machines having non-identical exponential service times was analysed in [1], where the repair policies which minimize the mean processing cost were considered. For the FSF- and PS-models we obtain analytical results for an arbitrary number of servers. Moreover, as will be shown in the paper, the performance characteristics of these systems in certain operation modes are the same or very close to those of the main system functioning under the optimal policy. Thus, these simplified models can be used under certain conditions to calculate upper and lower bounds for some performance characteristics and also as approximating models.

The main contributions of paper are as follows. It is shown numerically the structural properties of the optimal allocation policy. We derive for the main model the matrix expressions used further by calculating different performance measures such as the mean number of waiting customers, the mean number of busy servers, the mean length of a busy period. The matrix-analytic solution for the stationary state distribution and mean performance measures is obtained for the FSF-model. Here we used the recurrent definition of some blocks in the infinitesimal matrix. The stationary state distribution for the PS-model is obtained in a product form. We develop also the first step analysis to study the mean number of customers served in

the system or by the kth server in a busy period and the probability of the maximum queue length observed during this period.

The rest of the paper is organized as follows. In Section 2, we describe the Markov-decision process of the main model and show that the system has a threshold-based optimal allocation policy. In this section we develop also the computational analysis for the mean performance measures and the measures characterizing the behaviour of the system in a busy period. The FSF-Model is presented and analysed in Section 3. Section 4 is devoted to the PS-model. Comparison analysis of the proposed models and illustrative examples are summarized in Section 5.

The following notations will be used throughout this paper: $\mathbf{e}(n)$, $\mathbf{e}_j(n)$ and I_n stands respectively for the unit vector of dimension n, for the basis vector of dimension n in $\mathbb{R}^n$ with $0 \leq j \leq n-1$ or $1 \leq j \leq n$ depending on the context, and for the identity matrix of dimension n. If it is not necessary to specify a vector dimension, we will omit the corresponding argument. For example, $\mathbf{e}$ denotes a column unit vector of an appropriate dimension. The notation $'$ is used for the transpose. The notation $\otimes$ stands for the Kronecker product of two matrices, $1_{\{A\}}$ – for the indicator function, where $1_{\{A\}} = 1$ if the condition A holds, and 0 otherwise. The notation $|A|$ is used for the magnitude of a finite set A.

2. OTP-Model

Here we discuss the main model of the non-preemptive finite-source controlled queueing system of the type $M/M/K/N//N$ illustrated in Figure 1. The system has K heterogeneous servers with different rates $\mu_1 \geq \mu_2 \geq \cdots \geq \mu_K > 0$ and N customers in a source. It operates under the optimal allocation policy which minimizes the mean number of customers in the system. It will be shown that this policy is defined through a sequence of threshold levels $1 = q_1 \leq q_2 \leq \cdots \leq q_K < \infty$ for the queue lengths which prescribe the activation of slower servers. The analysed system can be treated as a model for the machine-repairman problem, where N unreliable machines in a working area with exponential distributed life times and equal rates $\lambda > 0$ must be repaired by K heterogeneous repair stations. The machines fail independently of each other. The stream of failed machines can be treated as an arrival stream of customers to the queueing system. Hereafter, we will refer to the customer as a failed machine which enters the repair system and gets there a repair service. After the repair the machine becomes as good as a new one and it returns to the working area. The aim is to dynamically allocate the customers to the servers in order to minimize the long-run average number of customers in the system and to calculate the corresponding mean performance measures.

2.1. MDP formulation. We formulate the optimal allocation problem in this machine-repairman system as a Markov Decision Process (MDP) in the following

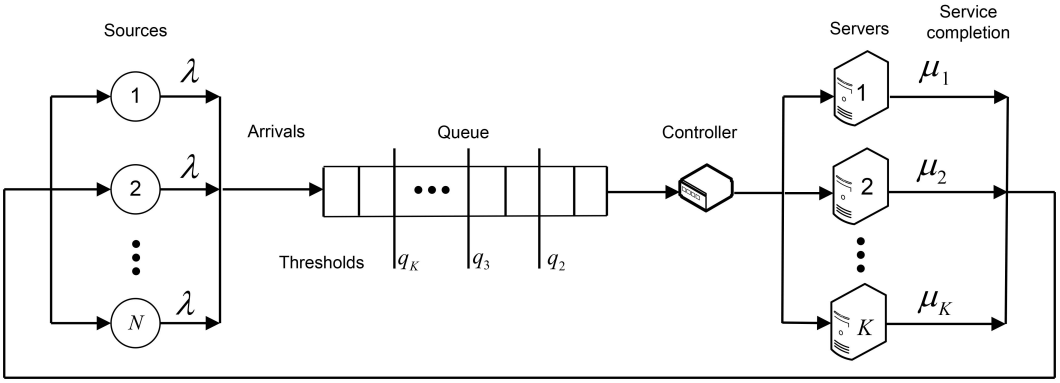


Fig. 1. The schema of the finite-source queueing system

way. The behaviour of the system is described by a multi-dimensional continuous-time Markov-chain

$$\{X(t)\}_{t\geq 0} = \{Q(t), D_1(t), \ldots, D_K(t)\}_{t\geq 0}, \tag{1}$$

where $Q(t)$ stands for the number of customers waiting in the queue at time t and $D_j(t)$ specifies the state of the jth server at time t, where

$$D_j(t) = \begin{cases} 0 & \text{if the server } j \text{ is idle} \\ 1 & \text{if the server } j \text{ is busy.} \end{cases}$$

State space: The set E_X consists of $K+1$ dimensional row vectors,

$$E_X = \{x = (q(x), d_1(x), \ldots, d_K(x)) : \\ q(x) \in \{0, 1, 2, \ldots, N - \sum_{j=1}^{K} d_j(x)\},\ d_j(x) \in \{0, 1\},\ j = 1, \ldots, K\},$$

where $q(x)$ denotes the number of customers in the queue and $d_j(x)$ – the status of the jth server in state x. The total number of states in the set E_X is equal to $|E_X| = \sum_{j=0}^{K} \binom{K}{j}(N - j + 1)$.

Decision epochs: The arrival and service completion epochs in the system with waiting customers.

Action space: $A = \{0, 1, \ldots, K\}$. To identify the group of idle and busy servers, the following sets are defined,

$$J_0(x) = \{j : d_j(x) = 0\},\ J_1(x) = \{d_j(x) = 1\}.$$

With this notations the set of admissible control actions $A(x) \subseteq A$ in state $x \in E_X$ can be defined as $A(x) = J_0(x) \cup \{0\}$. The action $a \in J_0(x)$ means that in state x a customer must be allocated to an idle server, while $a = 0$ means that the customer must be routed to the queue. At an arrival epoch, which occurs only if the number of customers in the system is less than N, the arrived customer joins the queue and simultaneously another one from the head of the queue must be routed to some idle server or returned back to the queue. At a service completion epoch the same happens, i.e. the customer from the head of the queue is routed either to one of idle servers or to the queue again. By service completion in a system without waiting customers no actions have to be performed.

Immediate cost: The function $l(x)$ specifies the number of customers in a state $x \in E_X$, i.e.

$$l(x) = q(x) + \sum_{j=1}^{K} d_j(x),$$

which is in fact independent of a control action a.

Transition rates: The policy-dependent infinitesimal matrix $\Lambda^f = [\lambda_{xy}(a)]_{x,y\in E_X}$ of the Markov-chain (1) includes the rates to go from state x to state y given the control action is a defined as

$$\lambda_{xy}(a) = \begin{cases} (N - l(x))\lambda & y = S_a x,\ 0 \le l(x) \le N,\ a \in A(x), \\ \mu_j & y = S_j^{-1} x,\ j \in J_1(x),\ q(x) = 0, \\ \mu_j & y = S_0^{-1} S_j^{-1} S_a x,\ j \in J_1(x),\ q(x) > 0,\ a \in A(S_0^{-1} S_j^{-1} x), \\ -((N - l(x)) & \\ + \sum\limits_{j \in J_1(x)} \mu_j) & y = x \\ 0 & \text{otherwise} \end{cases} \tag{2}$$

with $\lambda_x(a) = -\lambda_{xx}(a) = -\sum_{y \neq x} \lambda_{xy}(a)$, where S_a and S_j^{-1} stand for the shift operators applied to the vector state x in the following way,

$$S_a x = x + \mathbf{e}_a(K+1),\ a \in J_0(x) \text{ and } S_j^{-1} x = x - \mathbf{e}_j(K+1),\ j \in J_1(x).$$

Due to the finiteness of the state space E_X and boundedness of the immediate cost function $l(x) \le N$, a stationary average-cost optimal policy $f : E \to A$ exists with a finite constant gain

$$g^f = \limsup_{t\to\infty} \frac{1}{t} \mathbb{E}^f \Big[\int_0^t \Big(Q(t) + \sum_{j=1}^{K} D_j(t) \Big) dt \Big| X(0) = x \Big] = \sum_{x \in E_X} l(x) \pi_x^f < \infty$$

which is independent of the initial state x. In this case the policy-iteration algorithm introduced in Algorithm 1 converges. This algorithm consists of two main parts:

Algorithm 1 Policy-iteration algorithm

1: **procedure** PIA$(K, N, \lambda, \mu_j, j = 1, 2, \ldots, K)$

2: $f^{(0)}(x) = \text{argmax}_{j \in J_0(x)} \left\{ \mu_j \right\}$ ▷ Initial policy

3: $n \leftarrow 0$

4: $g^{f^{(n)}} = N\lambda v^{f^{(n)}}(\mathbf{e}_1(K+1))$ ▷ Policy evaluation

5: **for** $x = (0, 1, 0, \ldots, 0)$ **to** $(N-K, 1, 1, \ldots, 1)$ **do**

$$v^{f^{(n)}}(x) = \frac{1}{(N - l(x))\lambda + \sum_{j \in J_1(x)} \mu_j} \Big[l(x) - g^{f^{(n)}} + (N - l(x))\lambda v^{f^{(n)}}(S_{f^{(n)}(x)}x) + \sum_{j \in J_1(x)} \mu_j v^{f^{(n)}}(S_j^{-1}x) 1_{\{q(x)=0\}} + \sum_{j \in J_1(x)} \mu_j v^{f^{(n)}}(S_0^{-1}S_j^{-1}S_{f^{(n)}(S_0^{-1}S_j^{-1}x)}x) 1_{\{q(x)>0\}} \Big]$$

6: **end for**

7: ▷ Policy improvement

$$f^{(n+1)}(x) = \text{argmin}_{a \in A(x)}\, v^{f^{(n)}}(S_a x)$$

8: **if** $f^{(n+1)}(x) = f^{(n)}(x)$, $x \in E^f$ **then return** $f^{(n+1)}(x), v^{f^{(n)}}(x), g^{f^{(n)}}$

9: **else** $n \leftarrow n+1$, **go to step 4**

10: **end if**

11: ▷ Threshold evaluation

$$q_k : f^{(n+1)}(q, 1, \ldots, 1, 0, d_{k+1}, \ldots, d_K) = \begin{cases} 0 & q \le q_k - 2 \\ k & q > q_k - 2 \end{cases}, \; k = 2, \ldots, K$$

12: **end procedure**

Policy evaluation and Policy improvement. In the first part, a system of linear equations with immediate costs $l(x)$

$$v^f(x) = -\frac{1}{\lambda_{xx}(a)} \Big(l(x) + \sum_{y \neq x} \lambda_{xy}(a) v^f(y) - g^f \Big) \tag{3}$$

is solved for the unknown real-valued dynamic-programming value function $v^f : E_X \to \mathbb{R}$ and gain g^f given a control policy is f. The second part of the algorithm is responsible for improving the previous policy, which for a given system consists in determining, for each system state, a control action a that minimizes the value function $v(S_a x)$. The improved control action in state x is defined then as $f^*(x) = \text{argmin}_{a\in A(x)}\, v(S_a x)$ for $x \in E_X \setminus \{x : l(x) = N\}$. Thus, the algorithm constructs a sequence of improved control policies until it finds one that minimizes the gain g^f.

In Algorithm 1 we perform a conversion of the $K+1$-dimensional state space E_X of the Markov chain (1) to one-dimensional equivalent state space using the function $\Delta : E_X \to \mathbb{N}_0$, where

$$\Delta(x) = q(x)2^K + \sum_{i=1}^{K} d_i(x)2^{i-1}. \tag{4}$$

In one-dimensional state space the transitions due to arrivals and service completions can be defined then as

$$\Delta(x \pm \mathbf{e}_0(K+1)) = (q(x) \pm 1)2^K + \sum_{i=1}^{K} d_i(x)2^{i-1} = \Delta(x) \pm 2^K,$$

$$\Delta(x \pm \mathbf{e}_j(K+1)) = q(x)2^K + \sum_{i=1}^{K} d_i(x)2^{i-1} \pm 2^{j-1} = \Delta(x) \pm 2^{j-1},\ 1 \le j \le K.$$

For more details about derivation of the optimality equation for heterogeneous queueing systems the interested reader is referred to relevant publications, e.g. [3].

Numerical analysis confirms our expectation that the optimal control policy in heterogeneous systems for a finite number of customers also belongs to a class of threshold policies, as in infinite population case. Theoretical justification of this statement is still difficult. For this purpose it is necessary to prove that the dynamic-programming operator B defined for our queueing model as

$$\begin{aligned} v(x) &= \frac{1}{(N-l(x))\lambda + \sum\limits_{j\in J_1(x)} \mu_j} \Big[l(x) + (N-l(x))\lambda T_0 v(x) + \sum_{j\in J_1(x)} \mu_j T_j v(x) - g \Big] \\ &= Bv(x), \end{aligned} \tag{5}$$

where T_0 and T_j are the events operators in case of a new arrival and a service completion at server $j \in J_1(x)$,

$$T_0 v(x) = \min_{a \in A(x)} v(S_a x),$$
$$T_j v(x) = v(S_j^{-1} x),\ q(x) = 0,$$
$$T_j v(x) = T_0 v(S_0^{-1} S_j^{-1} x),\ q(x) > 0,$$

preserves the monotonicity properties of the increments of the value function v:

$$v(S_0 x) - v(S_2 x) - v(S_0^2 x) + v(S_0 S_2 x) \le 0,\ x \in E_X,\ d_1(x) = 1,\ d_2(x) = 0, \quad (6)$$
$$v(S_0 x) - v(x) - v(S_0 S_2 x) + v(S_2 x) \le 0,\ x \in E_X,\ d_1(x) = 1,\ d_2(x) = 0. \quad (7)$$

In proving the inequality (7) we encounter difficulty. This is due primarily to the form of the operator B in (5). There is a term describing arriving customers whose coefficient $(N - l(x))\lambda$ depends on the system state x. Bringing the terms in inequality (7) to a common denominator by introducing fictitious transitions, we get terms which cannot be proved to be negative. We hope that we will be able to overcome these difficulties in our next paper, but to date we're basing our statement about a threshold structure of the optimal control policy f exclusively on the performed numerical experiments. The following example makes the case vividly.

Example 1. Consider the system with $K = 5, N = 60$ and $\lambda = 0.3$. The service rates take the following values: $(\mu_1, \mu_2, \mu_3, \mu_4, \mu_5) = (20, 8, 4, 2, 1)$. The table of optimal control actions $f(x)$ for selected system states x is of the form:

System state x	Queue length $q(x)$													
$d = (d_1, d_2, d_3, d_4, d_5)$	0	1	2	3	4	5	6	7	8	9	10	11	12	...
(0,*,*,*,*)	**1**	1	1	1	1	1	1	1	1	1	1	1	1	1
(1,0,*,*,*)	**2**	2	2	2	2	2	2	2	2	2	2	2	2	2
(1,1,0,*,*)	0	**3**	3	3	3	3	3	3	3	3	3	3	3	3
(1,1,1,0,*)	0	0	0	**4**	4	4	4	4	4	4	4	4	4	4
(1,1,1,1,0)	0	0	0	0	0	0	0	0	**5**	5	5	5	5	5
(1,1,1,1,1)	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Threshold levels q_k, $2 \le k \le K$, are evaluated by comparing the optimal actions $f(x) = 0$ and $f(S_0 x) = k$ for $x = (q(x), 1, \ldots, 1, 0, d_{k+1}(x), \ldots, d_K(x))$, $0 \le q(x) \le N - \sum_{j=1}^{K} d_j(x), d_j(x) \in \{0, 1\}$. In this example the optimal policy f is defined here through a sequence of threshold levels $(q_2, q_3, q_4, q_5) = (1, 2, 4, 9)$ and $g^f = 4.91549$.

2.2. Evaluation of system performance measures. We are concerned in calculation of the system performance measures for a given policy f. The state probabilities and performance characteristics defined here will refer to some particular fixed control policy f, so we will use in notations the corresponding upper index.

The states x of the set E_X with $q(x) = 0$ are ordered according to the number of busy servers $|J_1(x)|$ while the states for $q(x) > 0$ are ordered with respect to the queue length, so that the infinitesimal matrix Λ^f has a block three-diagonal structure for the fixed policy f. First we define the performance characteristics:

- The probability that the kth server $1 \le k \le K$ is busy, $\bar{U}_k^f = \sum_{x \in E_X} d_k(x)\pi_x^f$;
- The mean number of busy servers, $\bar{C}^f = \sum_{k=1}^K \bar{U}_k^f$;
- The mean number of customers in the queue, $\bar{Q}^f = \sum_{x \in E_X} q(x)\pi_x^f$.
- The mean number of customers in the system, $\bar{N}^f = \bar{C}^f + \bar{Q}^f$.

The following vectors of dimension $|E_X| - 1$ comprise the policy-dependent values $a^f(x)$ and policy-independent values $b(x)$,

$$\mathbf{a}^f = (a^f(x) : x \in E_X \setminus \{x_0\}),\ \mathbf{b} = (b(x) : x \in E_X \setminus \{x_0\}),\ x_0 = \mathbf{0}.$$

where the first elements of the vectors are respectively $a^f(\mathbf{e}_1(K+1))$ and $b(\mathbf{e}_1(K+1))$.

Proposition 1. The performance measure $\bar{M}_1^f$ satisfies the following system

$$\begin{aligned} &\bar{M}_1^f = N\lambda \mathbf{e}_1'(|E_X| - 1)\mathbf{a}^f, \\ &(\tilde{\Lambda}^f + N\lambda \mathbf{e}'(|E_X| - 1) \otimes \mathbf{e}_1(|E_X| - 1))\mathbf{a}^f = -\mathbf{b}, \end{aligned} \tag{8}$$

where the matrix $\tilde{\Lambda}^f$ is obtained from Λ^f by removing the first column and the first row, and

$$\bar{M}_1^f = \begin{cases} \bar{U}_k^f & b(x) = d_k(x),\ x \in E_X, \\ \bar{C}^f & b(x) = \sum_{k=1}^K d_k(x),\ x \in E_X, \\ \bar{Q}^f & b(x) = q(x),\ x \in E_X, \\ \bar{N}^f & b(x) = l(x),\ x \in E_X. \end{cases} \tag{9}$$

Proof. We multiply the both sides of the second equality in (8) by the row-vector of the stationary state probabilities $\tilde{\boldsymbol{\pi}}^f = (\pi_x^f : x \in E \setminus \{x_0\})$,

$$\tilde{\boldsymbol{\pi}}^f(\tilde{\Lambda}^f - N\lambda \mathbf{e}'(|E_X| - 1) \otimes \mathbf{e}_1(|E_X| - 1))\mathbf{a}^f = -\tilde{\boldsymbol{\pi}}^f \mathbf{b},$$

where $\tilde{\boldsymbol{\pi}}^f \mathbf{b} = \sum_{x \in E_X \setminus \{x_0\}} b(x)\pi_x^f$ for the corresponding function $b(x)$ is obviously equal to the performance measure $\bar{M}_1^f$. The following sequence of relations

$$\begin{aligned} &\tilde{\boldsymbol{\pi}}(\tilde{\Lambda}^f - N\lambda \mathbf{e}'(|E_X| - 1) \otimes \mathbf{e}_1(|E_X| - 1))\mathbf{a}^f = \\ &\tilde{\boldsymbol{\pi}}^f \tilde{\Lambda}^f \mathbf{a}^f - N\lambda \tilde{\boldsymbol{\pi}}^f \mathbf{e}'(|E_X| - 1) \otimes \mathbf{e}_1(|E_X| - 1)\mathbf{a}^f = \\ &- \pi_{x_0}^f (N\lambda, 0, \ldots, 0)\mathbf{a}^f - N\lambda(1 - \pi_{x_0}^f, 0, \ldots, 0)\mathbf{a}^f = -N\lambda \mathbf{e}_1'(|E_X| - 1)\mathbf{a}^f = -\bar{M}_1^f. \end{aligned}$$

validates the statement. ■

The following measures characterize the behaviour of the system in a busy period which we define as a duration starting when the arrived customer enters the empty system in state x_0 and finishes when the system visits x_0 again after a service completion.

- The mean length of a busy period, $\bar{L}^f = \frac{1}{N\lambda}\Big(\frac{1}{\pi^f_{x_0}} - 1\Big)$;
- The mean number of customers served in a busy period by the kth server, $\bar{N}^f_{L,k}$;
- The total mean number of customers served in a busy period, $\bar{N}^f_L = \sum_{k=1}^K \bar{N}^f_{L,k} = \frac{1}{\pi^f_{x_0}}$.

In the following proposition we describe a general way to calculate these characteristics for the fixed control policy f.

Proposition 2. The performance measure $\bar{M}^f_2$ satisfies the following system

$$\begin{aligned} &\bar{M}^f_2 = \mathbf{e}'_1(|E_X| - 1)\mathbf{a}^f, \\ &\tilde{\Lambda}^f \mathbf{a}^f = -\mathbf{b}, \end{aligned} \tag{10}$$

where

$$\bar{M}^f_2 = \begin{cases} \bar{L}^f & b(x) = 1 + \sum_{k=1}^K d_k(x)\mu_k 1_{\{|J_1(x)|=1\}},\ x \in E_X, \\ \bar{N}^f_{L,k} & b(x) = d_k(x)\mu_k,\ x \in E_X, \\ \bar{N}^f_L & b(x) = \sum_{k=1}^K d_k(x)\mu_k,\ x \in E_X. \end{cases}$$

Proof. Denote by $\tilde{\varphi}^f_x(s) = \int_0^\infty \varphi^f_x(t)e^{-st}dt, Re[s] > 0$, the Laplace-Stiltjes transform (LST) of the probability density function (PDF) $\varphi^f_x(t)$ for the first passage time to state x_0 given that the initial state is $x \in E_X$, the control policy is f and by $\bar{L}^f_x = \int_0^\infty t\varphi^f_x(t)dt$ the corresponding first moment. According to the first step analysis we get for the LST the system

$$\begin{aligned} &\tilde{\varphi}^f_{x_0}(s) = 0, \\ &\tilde{\varphi}^f_x = \sum_{y \neq x} \frac{\lambda_{xy}(a)}{s + \lambda_x(a)} \tilde{\varphi}^f_y(s),\ x \in E_X \setminus \{x_0\}. \end{aligned} \tag{11}$$

We take into account that $\bar{L}^f(x) = -\frac{d}{ds}\tilde{\varphi}^f_x(s)\Big|_{s=0}$, we can obtain from (11) the system for the conditional moments

$$\begin{aligned} &\bar{L}^f(x_0) = 1, \\ &\bar{L}^f(x) = \frac{1}{\lambda_x(a)}\Big[1 + \sum_{y \neq x} \lambda_{xy}(a)\bar{L}^f(y)\Big],\ x \in E_X \setminus \{x_0\}. \end{aligned} \tag{12}$$

The system (12) for the transition rates (2) is of the form

$$\Big((N-l(x))\lambda+\sum_{j\in J_1(x)}\mu_j\Big)\bar{L}^f(x)=1+\sum_{j\in J_1(x),|J_1(x)|=1}\mu_j 1_{\{q(x)=0\}}+ \tag{13}$$
$$(N-l(x))\lambda\bar{L}^f(S_{f(x)}x)+\sum_{j\in J_1(x),|J_1(x)|>1}\mu_j\bar{L}(S_j^{-1}x)1_{\{q(x)=0\}}+$$
$$\sum_{j\in J_1(x)}\mu_j\bar{L}^f(S_0^{-1}S_j^{-1}S_{f(S_0^{-1}S_j^{-1}x)}x)1_{\{q(x)>0\}},\ x\in E_X\setminus\{x_0\}$$

By expressing relations (13) in matrix form and taking into account that $\bar{L}^f := \bar{L}^f(\mathbf{e}_1(K+1))$ we obtain the expressions (10) for $a^f(x)=\bar{L}^f(x)$.

Denote now by $\tilde{\psi}^f_{x,k}=\sum_{i=0}^{\infty}\psi^f_{x,k}(i)z^i$, $|z|\le 1$, the probability generating function (PGF) of the PDF $\psi^f_{x,k}(i)$ of the number of service completion at server k up to the end of busy period given that the initial state is $x\in E_X\setminus\{x_0\}$. With respect to the law of the total probability we get the following relations for the function $\psi^f_{x,k}(i)$,

$$\psi^f_{x,k}(i)=\frac{\lambda_{xu}(a)}{\lambda_x(a)}\psi^f_{u,k}(i-1)+\sum_{y\neq x,u}\frac{\lambda_{xy}(a)}{\lambda_x(a)}\psi^f_{y,k}(i),\ i\ge 1. \tag{14}$$

The first term on the right hand side of (14) represents the transition to state u accompanied with an event we count, that is a service completion at server k. The second term stands for other possible transitions. The system (14) can be rewritten in terms of the PGF in the following form,

$$\tilde{\psi}^f_{x,k}(z)=\frac{z\lambda_{xu}(a)}{\lambda_x(a)}\tilde{\psi}^f_{u,k}+\sum_{y\neq x,u}\frac{\lambda_{xy}(a)}{\lambda_x(a)}\tilde{\psi}^f_{y,k}(z). \tag{15}$$

The expressions (15) can be modified using the property $\bar{N}^f_{L,k}(x)=\frac{d}{dz}\tilde{\psi}_{x,k}(z)\Big|_{z=1}$ in such a way that we get a system for the corresponding first moments,

$$\bar{N}^f_{L,k}(x_0)=1, \tag{16}$$
$$\bar{N}^f_{L,k}(x)=\frac{1}{\lambda_x(a)}\Big[\lambda_{xu}(a)+\sum_{y\neq x}\lambda_{xy}(a)\bar{N}^f_{L,k}(y)\Big],\ x\in E_X\setminus\{x_0\}.$$

For the model under study the system (16) is of the form

$$\Big((N-l(x))\lambda+\sum_{j\in J_1(x)}\mu_j\Big)\bar{N}^f_{L,k}(x)=d_k(x)\mu_k+ \tag{17}$$
$$(N-l(x))\lambda\bar{N}^f_{L,k}(S_{f(x)}x)+\sum_{j\in J_1(x)}\mu_j\bar{N}^f_{L,k}(S_j^{-1}x)1_{\{q(x)=0\}}+$$
$$\sum_{j\in J_1(x)}\mu_j\bar{N}^f_{L,k}(S_0^{-1}S_j^{-1}S_{f(S_0^{-1}S_j^{-1}x)}x)1_{\{q(x)>0\}},\ x\in E_X\setminus\{x_0\}.$$

The last system can be also expressed in form (10) for $a^f(x)=\bar{N}^f_{L,k}(x)$ and $\bar{N}^f_{L,k}=\bar{N}_{L,k}(\mathbf{e}_1(K+1))$. For the mean total number of customers served $\bar{N}_L$ the term $d_k(x)\mu_k$ on the right hand side of (17) must be replaced by $\sum_{k=1}^K d_k(x)\mu_k$. ■

Finally, one more performance measure in this section is of our interest, namely, the distribution of the maximal queue length in a busy period for the given control policy f. Denote by Q^f_{max} the maximum number of customers waiting in the queue during a busy period. For each fixed value $n\geq 0$ the event $\{Q^f_{max}\leq n\}$ is equivalent to the event that the process $\{X(t)\}_{t\geq 0}$ starting in state $\mathbf{e}_1(K+1)$, where the first server is busy, hits the empty state x_0 before visiting the subset of states

$$E_{max,n}=\{x=(q(x),d_1(x),\ldots,d_K(x)):$$
$$q(x)\in\{n+1,n+2,\ldots,N-\sum_{j=1}^K d_j(x)\},\ d_j(x)\in\{0,1\},\ j=1,\ldots,K\}$$

The probability $\bar{Q}^f_{max,n}=\mathbb{P}[Q^f_{max}\leq n]$ will be calculated by means of absorption probabilities for states in a set of absorbing states $E_{max,n}\cup\{x_0\}$ given that the initial state is $x\in E_{X,n}=E_X\setminus E_{max,n}\cup\{x_0\}$. Denote by

$$\mathbf{a}^f(n)=(a^f(x):x\in E_{X,n})\ \text{ and }\ \mathbf{b}(n)=(b(x):x\in E_{X,n})$$

the column-vectors of dimension $|E_{X,n}|=|E_X|-|E_{max,n}|-1=\sum_{j=0}^K\binom{K}{j}(n+1)-1$.

Proposition 3. The performance measure $\bar{M}^f_3$ satisfies the following system

$$\bar{M}_3(n)^f=\mathbf{e}_1'(|E_{X,n}|)\mathbf{a}^f(n), \tag{18}$$
$$\tilde{\Lambda}^f(n)\mathbf{a}^f(n)=-\mathbf{b}(n),$$

where the matrix $\tilde{\Lambda}^f(n)$ is obtained from $\tilde{\Lambda}^f$ by removing all columns and rows starting with the $n+1$, and

$$\bar{M}^f_3(n)=\bar{Q}^f_{max,n},\ b(x)=\sum_{k=1}^K d_k(x)\mu_k 1_{\{|J_1(x)|=1\}},\ x\in E_{X,n}. \tag{19}$$

Proof. Denote by $\bar{Q}^f_{max,n}(x)$ the probability of absorption into empty state x_0 starting in $x \in E_{X,n}$, where $\bar{Q}^f_{max,n} = \bar{Q}^f_{max,n}(\mathbf{e}_1(K+1))$, where $\mathbf{e}_1(K+1)$ as before is the state after an arrival to an empty state x_0. The following system can be obtained by conditioning on the next visited state Using again the first principles,

$$\begin{aligned}
&\bar{Q}^f_{max,n}(x_0) = 1, \\
&\bar{Q}^f_{max,n}(x) = \frac{1}{\lambda_x(a)} \sum_{y \neq x} \lambda_{xy}(a) \bar{Q}^f_{max,n}(y),\ x \in E_{X,n}, \\
&\bar{Q}^f_{max,n}(x) = 0,\ x \in E_{max,n}.
\end{aligned} \tag{20}$$

For the queueing system operation under the control policy f the system (20) is of the form,

$$\begin{aligned}
&\Big((N - l(x))\lambda + \sum_{j \in J_1(x)} \mu_j\Big) \bar{Q}^f_{max,n}(x) = (N - l(x))\lambda \bar{Q}^f_{max,n}(S_{f(x)}x) + \\
&\sum_{j \in J_1(x)} \mu_j \bar{Q}^f_{max,n}(S_j^{-1}x) 1_{\{q(x)=0\}} + \\
&\sum_{j \in J_1(x)} \mu_j \bar{Q}^f_{max,n}(S_0^{-1} S_j^{-1} S_{f(S_0^{-1}S_j^{-1}x)}x) 1_{\{q(x)>0\}},\ x \in E_{X,n}.
\end{aligned} \tag{21}$$

Then after a routine of (block) identification the system (21) can be expressed in form (18), where $a^f(x) = \bar{Q}^f_{max,n}(x), x \in E_{X,n}$. ■

As we can see, calculating the performance characteristics requires solving very similar systems of equations. Thus, the same algorithm can be used for this purpose by substituting appropriate values into vectors $\mathbf{a}^f$ and $\mathbf{b}$, This versatility of the proposed approach greatly simplifies the application of algorithmic types of analysis of complex controlled queueing systems. In principle, we assume that for a fixed control threshold policy, the structure of the infinitesimal matrix can be even fully defined for an arbitrary number of servers, as will be proposed in the next section for the special case of the control policy where all thresholds are equal to 1. Thus we believe that matrix expressions can be derived explicitly from the presented matrix systems for performance characteristics. We leave this problem for our research in the near future.

3. FSF-Model

Here we discuss the FSF-Model which is a special case of the OTP-model, where $q_1 = q_2 = \cdots = q_K = 1$. The Markov-chain $\{X(t)\}_{t \geq 0}$ operating under the FSF-

policy has a state space

$$E_X = \{x : q(x) = 0, |J_1(x)| < K\} \cup \{x : q(x) \geq 1, |J_1(x)| = K\}.$$

The states in E_X are divided in to levels y in the following way,

$$\mathbf{y} = \{x \in E : q(x) = 0, |J_1(x)| = y\},\ 0 \leq y \leq K,$$
$$\mathbf{y} = \{x \in E : q(x) = y, |J_1(x)| = K\},\ K+1 \leq y \leq N.$$

Denote by $s_{i,j} = \binom{K-j+i}{i}$ for $K \geq j$, then $|\mathbf{y}| = s_{y,y}$ for $1 \leq y \leq K$ and $|\mathbf{y}| = 1$ for $K+1 \leq y \leq N$. Within each level $y, 1 \leq y \leq K$, the states are ordered in the lexicographic order, where the rank of x in the level y with $|J_1(x)| = y$ and $|J_0(x)| = K - y$ can be evaluated by

$$\Delta_y(x) = \sum_{i=1, i\in J_0(x)}^{K-1} \frac{n_i(x)(K-i)!}{\Big(\sum_{j=i}^{K} d_j(x)\Big)!\Big(K - \sum_{j=i}^{K} d_j(x)\Big)!} + 1, \tag{22}$$

where $n_i(x) = |\{j : d_j(x) = 1, d_i(x) = 0, j > i\}|$ is the number of slower busy servers as the ith idle one. Note that this ordering of states differs from that defined in (4) and used in the policy iteration algorithm. In the lexicographic ordering within each level of states it is possible to obtain explicit matrix expressions for state probabilities in case of an arbitrary number of servers K. Denote further by $L_y, 1 \leq y \leq K$, matrices whose rows consist of ordered elements of level y.

Proposition 4. The the system under FSF-policy is described by a QBD process with a block-three diagonal infinitesimal matrix of the form

$$\Lambda = \begin{pmatrix}
A_{1,0} & A_{0,1} & 0 & 0 & 0 & 0 & 0 & 0 & \dots & 0 \\
A_{2,0} & A_{1,1} & A_{0,2} & 0 & 0 & 0 & 0 & 0 & \dots & 0 \\
0 & A_{2,1} & A_{1,2} & A_{0,3} & 0 & 0 & 0 & 0 & \dots & 0 \\
\ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \\
0 & \dots & 0 & A_{2,K-2} & A_{1,K-1} & A_{0,K} & 0 & 0 & \dots & 0 \\
0 & \dots & 0 & 0 & A_{2,K-1} & A_{1,K} & A_{0,K+1} & 0 & \dots & 0 \\
0 & \dots & 0 & 0 & 0 & A_{2,K} & A_{1,K+1} & A_{0,K+2} & \dots & 0 \\
\ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \\
0 & \dots & 0 & 0 & 0 & 0 & 0 & A_{2,N-2} & A_{1,N-1} & A_{0,N} \\
0 & \dots & 0 & 0 & 0 & 0 & 0 & 0 & A_{2,N-1} & A_{1,N}
\end{pmatrix}. \tag{23}$$

The square blocks $A_{1,y}$ of dimension $s_{y,y}$ for $0 \leq y \leq K-1$ and 1 for $K \leq y \leq N$ consist of the rates to stay in the yth level, are defined as

$$A_{1,y} = I_{s_{y,y}}(\mathbf{e}'(s_{y,y}) \otimes [L_y B_{0,1} + (N-y)\lambda \mathbf{e}(s_{y,y})]),\ 0 \leq y \leq K-1, \tag{24}$$
$$A_{1,y} = (N-y)\lambda + m_K,\ K \leq y \leq N.$$

The blocks $A_{0,y}$ of dimension $s_{y-1,y-1} \times s_{y,y}$ for $1 \le y \le K$ and of dimension 1 for $K+1 \le y \le N$ consist of the rates to move upwards from the level $y-1$ to y due to arrivals and are defined as

$$A_{0,y} = (N-y+1)\lambda \begin{pmatrix} I_{s_{0,y}} & 0 & 0 & 0 & \dots & 0 \\ I_{s_{1,y}} & & 0 & 0 & \dots & 0 \\ \ddots & \ddots & \ddots & \ddots & \ddots & \\ I_{s_{y-1,y}} & & & 0 & \dots & 0 \end{pmatrix}, 1 \le y \le K, \tag{25}$$

$A_{0,y} = (N-y+1)\lambda$, $K+1 \le y \le N$.

The blocks $A_{2,y}$ of dimension $s_{y,y} \times s_{y-1,y-1}$ for $1 \le y \le K$ and of dimension 1 for $K+1 \le y \le N$ consist of the rates to move downwards from the level $y+1$ to y due to service completions and are defined as recursive matrices

$$A_{2,y} = B_{y,y+1},\ 1 \le y \le K, \text{ where} \tag{26}$$

$$B_{0,j} = (\mu_j, \mu_{j+1}, \dots, \mu_K)',$$

$$B_{i,j} = \begin{pmatrix} B_{i-1,j} & & & & & I_{s_{i,j}}\mu_{j-i} \\ 0 & B_{i-1,j+1} & & & & I_{s_{i,j+1}}\mu_{j+1-i} \\ \ddots & \ddots & \ddots & \ddots & \ddots & \\ 0 & 0 & \dots & 0 & B_{i-1,K} & I_{s_{i,K}}\mu_{K-i} \end{pmatrix},$$

$A_{2,y} = m_K$, $K+1 \le y \le N$.

Proof. Analysing the transitions of the Markov-chain $\{X(t)\}_{t\ge 0}$ we get a system of balance equations in form

$$((N - \sum_{j=1}^{K} d_j(x) - q(x))\lambda + \sum_{j=1}^{K} d_j(x)\mu_j)\pi_x = (N - \sum_{j=1}^{K} d_j(x) - q(x) + 1)\lambda \times \tag{27}$$

$$\times \sum_{k=1}^{K} 1_{\{\sum_{i=1}^{k} d_i(x) = k\}} \pi_{x-\mathbf{e}_k} + \sum_{j=1}^{K} (1 - d_j(x))\mu_j \pi_{x+\mathbf{e}_j},\ q(x) = 0,\ |J_1(x)| \le K,$$

$$((N - K - q(x))\lambda + m_K)\pi_x = (N - K - q(x) + 1)\lambda\pi_{x-\mathbf{e}_0} + m_K\pi_{x+\mathbf{e}_0},$$

$$q(x) > 0,\ |J_1(x)| = K,$$

where $\pi_x = \lim_{t\to\infty} \mathbb{P}[X(t) = x], x \in E$. Expressing equations (27) for the sub-vectors $\boldsymbol{\pi}_y$, $1 \le y \le K-1$, and the scalars $\boldsymbol{\pi}_0$ and $\boldsymbol{\pi}_y, K \le y \le N$, by means of

defined blocks and taking into account the states' ordering (22) we get the system

$$\begin{aligned}
&\boldsymbol{\pi}_0 A_{1,0} = \boldsymbol{\pi}_1 A_{0,1}, \\
&\boldsymbol{\pi}_y A_{1,y} = \boldsymbol{\pi}_{y-1} A_{0,y} + \boldsymbol{\pi}_{y+1} A_{2,y},\ 1 \le y \le K-2, \\
&\boldsymbol{\pi}_{K-1} A_{1,K-1} = \boldsymbol{\pi}_{K-2} A_{0,K-1} + \pi_K A_{2,K-1}, \\
&\boldsymbol{\pi}_K A_{1,K} = \boldsymbol{\pi}_{K-1} A_{0,K} + \pi_{K+1} A_{2,K}, \\
&\boldsymbol{\pi}_y A_{1,y} = \pi_{y-1} A_{0,y} + \pi_{y+1} A_{2,y},\ K+1 \le y \le N-1, \\
&\boldsymbol{\pi}_N A_{1,N} = \pi_{M-1} A_{0,N}.
\end{aligned} \tag{28}$$

Denote by $\boldsymbol{\pi}$ the macro-vector of the stationary state probabilities, i.e.

$$\boldsymbol{\pi} = (\boldsymbol{\pi}_0, \boldsymbol{\pi}_1, \ldots, \boldsymbol{\pi}_{K-1}, \boldsymbol{\pi}_K, \ldots, \boldsymbol{\pi}_M).$$

Compiling relations (28) to the the system $\boldsymbol{\pi}\Lambda = \mathbf{0}$ we get then the infinitesimal matrix Λ is the form (23) with blocks defined by (24)–(26). ■

Proposition 5. The elements of the stationary probability macro-vector $\boldsymbol{\pi}$ satisfy the relations

$$\boldsymbol{\pi}_0 = \prod_{j=1}^{K} M_{K-j} \boldsymbol{\pi}_K, \tag{29}$$

$$\boldsymbol{\pi}_y = \prod_{j=1}^{K-y} M_{K-j} \boldsymbol{\pi}_K,\ 1 \le y \le K-1, \tag{30}$$

$$\boldsymbol{\pi}_y = \frac{(N-K)!}{(N-y)!} \rho_K^{y-K} \boldsymbol{\pi}_K,\ K \le y \le N, \tag{31}$$

$$\boldsymbol{\pi}_K = \Big(\sum_{y=0}^{K-1} \prod_{j=1}^{K-y} M_{K-j} + \sum_{y=K}^{N} \frac{(N-K)!}{(N-y)!} \rho_K^{y-K} \Big)^{-1}, \tag{32}$$

where the matrices M_y satisfies the recursive relations

$$\begin{aligned}
&M_0 = A_{0,1} A_{1,0}^{-1}, \\
&M_y = A_{2,y} (A_{1,y} - M_{y-1} A_{0,y})^{-1},\ 1 \le y \le K-1.
\end{aligned} \tag{33}$$

Proof. The probability π_0 and sub-vectors $\boldsymbol{\pi}_y, 1 \le y \le K-2$, can be expressed from the balance equations (28) using a block forward elimination-backward substitution as

$$\boldsymbol{\pi}_0 = A_{0,1} A_{1,0}^{-1} \boldsymbol{\pi}_1 = \boldsymbol{\pi}_1 M_0,$$

$$\boldsymbol{\pi}_1 A_{1,1} = \boldsymbol{\pi}_1 M_0 A_{0,1} + \boldsymbol{\pi}_2 A_{2,1} \Rightarrow \boldsymbol{\pi}_1 = \boldsymbol{\pi}_2 A_{2,1} (A_{1,1} - M_0 A_{0,1})^{-1} = \boldsymbol{\pi}_2 M_1,$$

$$\boldsymbol{\pi}_y A_{1,y} = \boldsymbol{\pi}_y M_{y-1} A_{0,y} + \boldsymbol{\pi}_{y+1} A_{2,y} \Rightarrow \boldsymbol{\pi}_y = \boldsymbol{\pi}_{y+1} A_{2,y} (A_{1,y} - M_{y-1} A_{0,y})^{-1} = \boldsymbol{\pi}_{y+1} M_y.$$

We similarly obtain an expression for $\boldsymbol{\pi}_{K-1}$,

$$\boldsymbol{\pi}_{K-1}A_{1,K-1} = \boldsymbol{\pi}_{K-1}M_{K-2}A_{0,K-1} + \boldsymbol{\pi}_K A_{2,K-1} \Rightarrow$$
$$\boldsymbol{\pi}_{K-1} = A_{2,K-1}(A_{1,K-1} - M_{K-2}A_{0,K-1})^{-1}\boldsymbol{\pi}_K = M_{K-1}\pi_K$$

The relations (29) and (30) are obtained then through a successive substitution. The relation (31) is obtained by solving (28) for $K \le y \le N$ recursively using

$$\boldsymbol{\pi}_y = \frac{(N-y+1)\lambda}{m_K}\boldsymbol{\pi}_{y-1}$$

starting from the last equation. The relation (32) for the probability π_K is determined using the normalizing condition $\boldsymbol{\pi}\mathbf{e}(N) = 1$. ■

To calculate performance characteristics the expressions from the previous section applied to a control policy $f(x) = \text{argmax}_{j\in J_0(x)}\{\mu_j\}$ can be used. As an alternative to the policy-iteration algorithm we can use the proposed matrix-analytic solution (29)–(32) to obtain the matrix expressions for the performance characteristics in an explicit form.

Corollary 1. The probability that the kth server is busy and the mean number of busy servers,

$$\bar{U}_k = \Big(\sum_{y=1}^{K-1}\prod_{j=1}^{K-y} M_{K-j}L_y\mathbf{e}_k(s_{y,y}) + \sum_{y=K}^{N}\frac{(N-K)!}{(N-y)!}\rho_K^{y-K}\Big)\times,$$
$$\times\Big(\sum_{y=0}^{K-1}\prod_{j=1}^{K-y} M_{K-j} + \sum_{y=K}^{N}\frac{(N-K)!}{(N-y)!}\rho_K^{y-K}\Big)^{-1},\ \bar{C} = \sum_{k=1}^{K}\bar{U}_k.$$

The mean number of customers in the queue,

$$\bar{Q} = \sum_{y=K}^{N}\frac{(y-K)(N-K)!}{(N-y)!}\rho_K^{y-K}\Big(\sum_{y=0}^{K-1}\prod_{j=1}^{K-y} M_{K-j} + \sum_{y=K}^{N}\frac{(N-K)!}{(N-y)!}\rho_K^{y-K}\Big)^{-1}.$$

The mean number of customers in the system,

$$\bar{N} = \bar{C} + \bar{Q} = \Big(\sum_{y=1}^{K-1}\prod_{j=1}^{K-y} M_{K-j}L_y\mathbf{e}(s_{y,y}) + \sum_{y=K}^{N}\frac{(y-K+1)(N-K)!}{(N-y)!}\rho_K^{y-K}\Big)\times,$$
$$\times\Big(\sum_{y=0}^{K-1}\prod_{j=1}^{K-y} M_{K-j} + \sum_{y=K}^{N}\frac{(N-K)!}{(N-y)!}\rho_K^{y-K}\Big)^{-1}$$

Mean length of a busy period,

$$\bar{L} = \frac{1}{N\lambda}\Big(\Big(\prod_{j=1}^{K} M_{K-j}\Big)^{-1}\Big(\sum_{y=0}^{K-1}\prod_{j=1}^{K-y} M_{K-j} + \sum_{y=K}^{N}\frac{(N-K)!}{(N-y)!}\rho_K^{y-K}\Big) - 1\Big).$$

Mean number of customers served in a busy period,

$$\bar{N}_L = \Big(\prod_{j=1}^{K} M_{K-j}\Big)^{-1}\Big(\sum_{y=0}^{K-1}\prod_{j=1}^{K-y} M_{K-j} + \sum_{y=K}^{N}\frac{(N-K)!}{(N-y)!}\rho_K^{y-K}\Big).$$

The mean number of customers served by the kth server in a busy period and the distribution of the maximal queue length can be evaluated using the matrix systems (10) and (18) taking into account the structure (23) of the infinitesimal matrix Λ.

Proposition 6. The mean number $\bar{N}_{L,k}$ of customers served in a busy period by the kth server satisfies the relation

$$\bar{N}_{L,k} = \mathbf{e}_1'(K)\sum_{y=1}^{N}\Big(\prod_{j=1}^{y-1} T_j\Big)S_y,\ 1 \le k \le K, \tag{34}$$

where

$$\begin{aligned}
&S_1 = A_{1,1}^{-1}\mathbf{b}_1,\ T_1 = -A_{1,1}^{-1}A_{0,2}, \qquad (35)\\
&S_y = (A_{2,y-1}T_{y-1} + A_{1,y})^{-1}(\mathbf{b}_y - A_{2,y-1}S_{y-1}),\ T_y = -(A_{2,y-1}T_{y-1} + A_{1,y})^{-1}A_{0,y+1}\\
&\qquad 2 \le y \le N-1,\\
&S_N = (A_{2,N-1}T_{N-1} + A_{1,N})^{-1}(\mathbf{b}_N - A_{2,N-1}S_{N-1}).
\end{aligned}$$

The column-vector $\mathbf{b}_y = L_y\mathbf{e}_k(K+1)\mu_k$ for $1 \le y \le K$ and the scalar $\mathbf{b}_y = \mu_k$ for $K+1 \le y \le N$.

Proof. The system (10) can be rewritten for appropriate blocks in form

$$\begin{aligned}
&A_{1,1}\mathbf{a}_1 + A_{0,2}\mathbf{a}_2 = \mathbf{b}_1,\\
&A_{2,y-1}\mathbf{a}_{y-1} + A_{1,y}\mathbf{a}_y + A_{0,y+1}\mathbf{a}_{y+1} = \mathbf{b}_y,\ 2 \le y \le N-1,\\
&A_{2,N-1}\mathbf{a}_{N-1} + A_{1,N}\mathbf{a}_N = \mathbf{b}_N
\end{aligned}$$

The elements of $\mathbf{b}_y$ are equal to μ_k if for some state x of the level $\mathbf{y}$ $d_k(x) = 1$. This implies the relations for $\mathbf{b}_y$. Using a forward elimination - backward substitution we get the recursive relations

$$\mathbf{a}_y = S_y + T_y\mathbf{a}_{y+1},\ 1 \le y \le N-1,\ \mathbf{a}_N = S_N,$$

where S_y and T_y are defined by (35). This statement follows through recurrence substitution taking into account that $\bar{N}_{L,k} = \mathbf{e}_1'(K)\mathbf{a}_1$, since the level 1 consists of K states. ■

The following statement for the matrix equation (18) can be proved in a similar way taking into account the structure (23) of the infinitesimal matrix Λ.

Proposition 7. The probability of the maximum queue length in a busy period satisfies the relation

$$\bar{Q}_{max,n} = \mathbf{e}_1'(K) \sum_{y=1}^{n} \Big(\prod_{j=1}^{y-1} T_j \Big) S_y, \tag{36}$$

where

$$\begin{aligned} &S_1 = A_{1,1}^{-1} A_{2,0},\ T_1 = -A_{1,1}^{-1} A_{0,2}, \\ &S_y = -(A_{2,y-1}T_{y-1} + A_{1,y})^{-1} A_{2,y-1} S_{y-1},\ T_y = -(A_{2,y-1}T_{y-1} + A_{1,y})^{-1} A_{0,y+1} \\ &\qquad 2 \le y \le n-1, \\ &S_n = -(A_{2,n-1}T_{n-1} + A_{1,n})^{-1} A_{2,n-1} S_{n-1}. \end{aligned} \tag{37}$$

4. PS-model

In this section we discuss a queueing system with a preemption operating under a general threshold policy f defined as a sequence of threshold levels $(q_2, \ldots, q_K)$. The first server in this system is permanently available for service while the jth slower server must be used as soon as the number of customers in the system increases up to the value $q_{j-1} + j - 2$. This server must be removed from the system when the number of customers becomes again less as $q_{j-1} + j - 2$. Denote by $\{Y(t)\}_{t\ge 0}$ the continuous-time Markov-chain with a state space $E_Y = \{y : y \in \mathbb{N}_0\}$. All the rates are the same as in the model without preemption. The infinitesimal matrix $\Lambda_Y^f = \lambda_{xy}(q_2, \ldots, q_K)$ is then of the form:

$$\lambda_{xy}(q_2, \ldots, q_K) = \begin{cases} \lambda & y = x+1, \\ m_j & y = x-1, \\ & q_{j-1} + j - 2 \le y \le q_j + j - 2,\ j = 2, \ldots, K, \end{cases} \tag{38}$$

where $m_j = \sum_{i=1}^{j} \mu_i$ and $q_1 = 1$. The state transition diagram of the process $\{Y(t)\}_{t\ge 0}$ is illustrated in Figure 2.

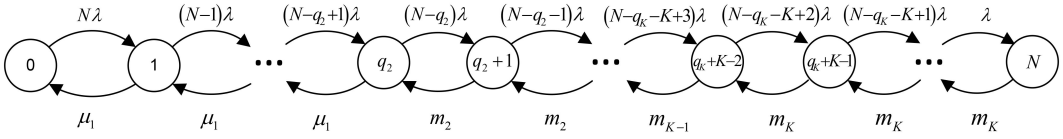


Fig. 2. The state transition diagram for the queueing system S_2

Proposition 8. The steady-state probabilities $\pi_y = \lim_{t\to\infty} \mathbb{P}[Y(t) = y]$ of the PS-Model satisfy the relations

$$\pi_y = \frac{N!}{(N-y)!} \prod_{i=1}^{j-1} \rho_i^{q_i - q_{i-1}+1} \rho_j^{y-q_j-j+2} \pi_0,\ q_j + j - 1 \le y \le q_{j+1} + j - 1, \qquad (39)$$
$$j = 1, \dots, K-1,$$

$$\pi_y = \frac{N!}{(N-y)!} \prod_{i=1}^{K-1} \rho_i^{q_i - q_{i-1}+1} \rho_K^{y-q_K-K+2} \pi_0,\ q_K + K - 1 \le y \le N,$$

$$\pi_0 = \Big(1 + \sum_{j=1}^{K} \sum_{y=q_j+j-1}^{q_{j+1}+j-1} \frac{N!}{(N-y)!} \prod_{i=1}^{j-1} \rho_i^{q_i - q_{i-1}+1} \rho_j^{y-q_j-j+2} +$$
$$\sum_{y=q_K+K-1}^{N} \frac{N!}{(N-y)!} \prod_{i=1}^{K-1} \rho_i^{q_i - q_{i-1}+1} \rho_K^{y-q_K-K+2} \Big)^{-1},$$

where $\rho_j = \frac{\lambda}{m_j}, j = 1, \dots, K$ and $\prod_{i=1}^{0} \dots = 1$.

Proof. The proposition follows by solving the following equations

$$\begin{aligned}
N\lambda\pi_0 &= \mu_1\pi_1, \\
((N - q_{j+1} - j + 1)\lambda + m_j)\pi_{q_{j+1}+j-1} &= (N - q_{j+1} - j + 2)\lambda\pi_{q_{j+1}+j-2} \\
&\quad + m_{j+1}\pi_{q_{j+1}+j}, \\
((N-y)\lambda + m_j)\pi_y &= (N - y + 1)\lambda\pi_{y-1} + m_j\pi_{y+1}, \\
&\quad q_j + j - 1 \le y \le q_{j+1} + j - 2, \\
((N-y)\lambda + m_K)\pi_y &= (N - y + 1)\lambda\pi_{y-1} + m_K\pi_{y+1}, \\
&\quad q_K + K - 1 \le y \le N - 1, \\
m_K\pi_N &= \lambda\pi_{N-1}
\end{aligned}$$

recursively for $j = 1, \dots, K-1$, where π_0 is calculated by means of the normalizing condition $\sum_{y=0}^{M} \pi_y = 1$. ■

Corollary 2. The probability that the kth server is busy and the mean number of busy servers,

$$\bar{U}_k^f = \left[\sum_{j=k}^{K} \sum_{y=q_j+j-1}^{q_{j+1}+j-1} \frac{M!}{(M-y)!} \prod_{i=1}^{j-1} \rho_i^{q_i-q_{i-1}+1} \rho_j^{y-q_j-j+2} + \sum_{y=q_K+K-1}^{N} \frac{M!}{(M-y)!} \prod_{i=1}^{K-1} \rho_i^{q_i-q_{i-1}+1} \rho_K^{y-q_K-K+2} \right] \pi_0, \ \bar{C}^f = \sum_{k=1}^{K} \bar{U}_k^f.$$

The mean number of customers in the queue,

$$\bar{Q}^f = \left[\sum_{j=1}^{K-1} \sum_{y=q_j+j-1}^{q_{j+1}+j-1} \frac{(y-j)N!}{(N-y)!} \prod_{i=1}^{j-1} \rho_i^{q_i-q_{i-1}+1} \rho_j^{y-q_j-j+2} + \sum_{y=q_K+K-1}^{N} \frac{(y-K)N!}{(N-y)!} \prod_{i=1}^{K-1} \rho_i^{q_i-q_{i-1}+1} \rho_K^{y-q_K-K+2} \right] \pi_0.$$

The mean number of customers in the system $\bar{N}^f = \bar{C}^f + \bar{Q}^f$.
The mean length of busy period,

$$\bar{L}^f = \frac{1}{N\lambda} \left[\sum_{j=1}^{K} \sum_{y=q_j+j-1}^{q_{j+1}+j-1} \frac{N!}{(N-y)!} \prod_{i=1}^{j-1} \rho_i^{q_i-q_{i-1}+1} \rho_j^{y-q_j-j+2} + \sum_{y=q_K+K-1}^{N} \frac{N!}{(N-y)!} \prod_{i=1}^{K-1} \rho_i^{q_i-q_{i-1}+1} \rho_K^{y-q_K-K+2} \right].$$

The mean number of customers served in a busy period,

$$\bar{N}_L^f = 1 + \sum_{j=1}^{K} \sum_{y=q_j+j-1}^{q_{j+1}+j-1} \frac{N!}{(N-y)!} \prod_{i=1}^{j-1} \rho_i^{q_i-q_{i-1}+1} \rho_j^{y-q_j-j+2} + \sum_{y=q_K+K-1}^{N} \frac{N!}{(N-y)!} \prod_{i=1}^{K-1} \rho_i^{q_i-q_{i-1}+1} \rho_K^{y-q_K-K+2}.$$

Further we use a similar methodology that has been employed in previous section to derive expressions for $\bar{N}_{L,k}^f$ and $\bar{Q}_{max,n}^f$ with the knowledge that all levels $\mathbf{y}$ consist now of only one state, and hence in the sequel we omit some details.

Proposition 9. The mean number of customers served by the kth server in a busy period satisfies the relation

$$\bar{N}^f_{L,k} = \sum_{y=1}^{N} \Big(\prod_{j=1}^{y-1} T_j \Big) S_y, \tag{40}$$

where

$$\begin{aligned}
S_1 =& \frac{m_1 + \mu_1 1_{\{k=1\}}}{(N-1)\lambda + m_1}, \; T_1 = \frac{(N-1)\lambda}{(N-1)\lambda + m_1} \\
S_y =& \frac{m_j S_{y-1} + \mu_k 1_{\{j \geq k\}}}{(N-y)\lambda + m_j - m_j T_{y-1}}, \; T_y = \frac{(N-y)\lambda}{(N-y)\lambda + m_j - m_j T_{y-1}}, \\
& q_j + j - 1 \leq y \leq q_{j+1} + j - 1, \; 1 \leq j \leq K-1, \\
S_y =& \frac{m_j S_{y-1} + \mu_K}{(N-y)\lambda + m_j - m_j T_{y-1}}, \; T_y = \frac{(N-y)\lambda}{(N-y)\lambda + m_K - m_K T_{y-1}}, \\
& q_K + K - 1 \leq y \leq N-1, \\
S_N =& \frac{S_{N-1}}{1 - T_{N-1}}.
\end{aligned} \tag{41}$$

Proof. The proof follows directly from (10) by forward elimination - backward substitution taking into account the structure (38) of the infinitesimal matrix Λ^f. ■

Proposition 10. The probability of the maximum queue length in a busy period satisfies the relation

$$\bar{Q}^f_{max,n} = \sum_{y=1}^{n} \Big(\prod_{j=1}^{y-1} T_j \Big) S_y, \tag{42}$$

where

$$\begin{aligned}
S_1 =& \frac{m_1}{(N-1)\lambda + m_1}, \; T_1 = \frac{(N-1)\lambda}{(N-1)\lambda + m_1} \\
S_y =& \frac{m_j S_{y-1} + \mu_k 1_{\{j \geq k\}}}{(N-y)\lambda + m_j - m_j T_{y-1}}, \; T_y = \frac{(N-y)\lambda}{(N-y)\lambda + m_j - m_j T_{y-1}}, \\
& q_j + j - 1 \leq y \leq \min\{n-1, q_{j+1} + j - 1\}, \; 1 \leq j \leq K-1, \\
S_y =& \frac{m_j S_{y-1} + \mu_K}{(N-y)\lambda + m_j - m_j T_{y-1}}, \; T_y = \frac{(N-y)\lambda}{(N-y)\lambda + m_K - m_K T_{y-1}}, \\
& q_K + K - 1 \leq y \leq \min\{n-1, N-1\}, \\
S_n =& \frac{m_j S_{n-1} + \mu_K}{(N-n)\lambda + m_j - m_j T_{n-1}}, \; n < N, \; S_n = \frac{S_{n-1}}{1 - T_{n-1}}, \; n = N.
\end{aligned} \tag{43}$$

The last result can be rewritten in explicit form as well.

Proposition 11. The probability of the maximum queue length in a busy period is given by

$$\bar{Q}_{max,n} = \frac{\sum_{y=1}^{n} F(y)}{1 + \sum_{y=1}^{n} F(y)}, \tag{44}$$

where the function $F(n)$ has the following product form,

$$\begin{aligned} F(y) =& \frac{m_j^{y-q_j-j+2}}{\prod_{i=q_j+j-1}^{y}((N-i)\lambda + m_j)} \prod_{i=1}^{j-1} \frac{m_i^{q_{i+1}-q_i+1}}{\prod_{s=q_i+i-1}^{q_{i+1}+i-1}((N-s)\lambda + m_i)}, \\ & q_j + j - 1 \le y \le q_{j+1} + j - 1,\ 1 \le j \le K-1, \\ F(y) =& \frac{m_K^{y-q_K-K+2}}{\prod_{i=q_K+K-1}^{y}((N-i)\lambda + m_K)} \prod_{i=1}^{K-1} \frac{m_i^{q_{i+1}-q_i+1}}{\prod_{s=q_i+i-1}^{q_{i+1}+i-1}((N-s)\lambda + m_i)}, \\ & q_K + K - 1 \le y \le N. \end{aligned} \tag{45}$$

Proof. The function $\bar{Q}^f_{max,n}(x), x \in E_Y$ for the given policy f satisfy the following system,

$$\begin{aligned} & \bar{Q}^f_{max,n}(0) = 1, \\ & ((N-y)\lambda + m_j)\bar{Q}^f_{max,n}(y) = (N-y)\lambda \bar{Q}^f_{max,n}(y+1) + m_j \bar{Q}^f_{max,n}(y+1), \\ & \qquad q_j + j - 1 \le y \le \min\{n-1, q_{j+1} + j - 1\},\ 1 \le j \le K-1, \\ & ((N-y)\lambda + m_K)\bar{Q}^f_{max,n}(y) = (N-y)\lambda + \bar{Q}^f_{max,n}(y+1) + m_K \bar{Q}^f_{max,n}(y-1), \\ & \qquad q_K + K - 1 \le y \le \min\{n-1, N\}, \\ & ((N-y)\lambda + m_K)\bar{Q}^f_{max,n}(n) = m_K \bar{Q}^f_{max,n}(n-1),\ n < N. \end{aligned} \tag{46}$$

These difference equations can be rewritten as recurrent relation for $1 \le y \le n$,

$$\bar{Q}^f_{max,n}(y+1) - \bar{Q}^f_{max,n}(y) = \frac{m_j}{(N-y)\lambda + m_j}(\bar{Q}^f_{max,n}(y) - \bar{Q}^f_{max,n}(y-1)). \tag{47}$$

By iterating (47), taking into account the structure of difference equations for the threshold policy we obtain

$$\bar{Q}^f_{max,n}(y+1) - \bar{Q}^f_{max,n}(y) = F(y)(\bar{Q}^f_{max,n}(1) - \bar{Q}^f_{max,n}(0)), \tag{48}$$

where the function $F(y)$ has a product form (45). Summing (48) for $y = 1, \ldots, n$ yields

$$\bar{Q}^f_{max,n}(n+1) - \bar{Q}^f_{max,n}(1) = \sum_{y=1}^{n} F(n)(\bar{Q}^f_{max,n}(1) - \bar{Q}^f_{max,n}(0)), \tag{49}$$

where $\bar{Q}^f_{max,n}(n+1) = 0$ and $\bar{Q}^f_{max,n}(0) = 1$. Expressing $\bar{Q}^f_{max,n}(1)$ we obtain the explicit formula (44). ■

5. Comparison analysis

In this section we discuss the results after having computed the performance metrics for the following finite-source heterogeneous queueing models: Non-preemptive queueing system operating under the optimal threshold policy (OTP-Model), non-preemptive queueing system with a fastest server first policy (FSF-Model), preemptive queueing system (PS1-Model), where the kth server is used when at least k customers present in the system, and preemptive queueing system (PS2-Model) operating according to a given threshold policy. This policy we calculate using a similar heuristic formula obtained in [5], which can be rewritten in form

$$q_k = \max\Big\{q_{k-1}, \left\lfloor \Big(\sum_{j=1}^{k-1}\mu_j - (N-\bar{N}^{PS1})\lambda\Big)\Big(\frac{1}{\mu_k} - \frac{k-1}{\sum_{j=1}^{k-1}\mu_j}\Big)\right\rfloor + k\Big\},\ 2 \le k \le K,$$

where $q_1 = 1$ and $(N-\bar{N}^{PS1})\lambda$ is an average arrival rate in the PS1-Model which is derived in explicit form.

In our experiments we fix the number of servers $K = 5$, the source capacity $N = 60$ and service intensities $(\mu_1, \mu_2, \mu_3, \mu_4, \mu_5) = (20, 8, 4, 2, 1)$. The rate λ will be varied in the interval $[0.01, 0.7]$. The choice of this interval is not random. At higher values of λ, the analysed functions become indistinguishable, since the corresponding queueing systems will have similar stochastic behaviour in a so-called heavy-traffic mode. In Figure 3, we display the functions $\bar{N}^f$ (figure labeled by "a") and $\bar{Q}^f$ (figure

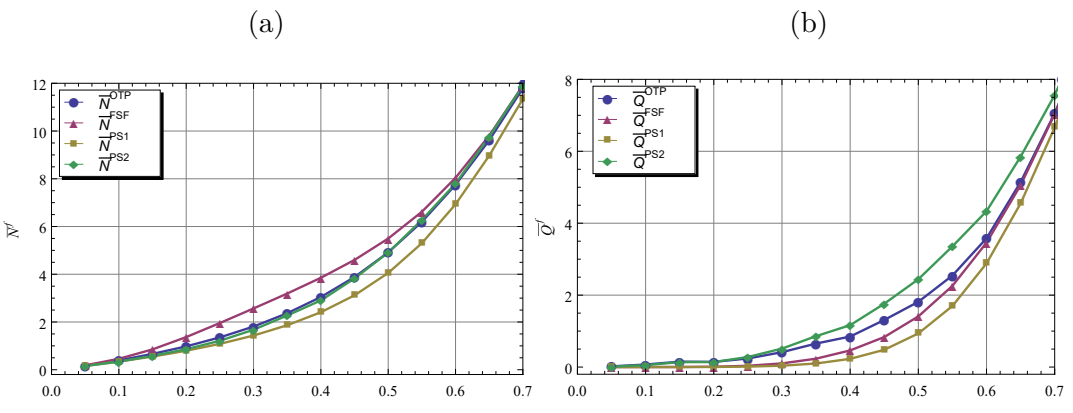


Fig. 3. $\bar{N}^f$ (a) and $\bar{Q}^f$ versus λ

labeled by "b") for different models as λ varies. We observe that the functions $\bar{N}^{FSF}$

and $\bar{Q}^{PS1}$ models are the natural upper and low bounds for $\bar{N}^{OTP}$. It is clear that the FSF-model is a particular case of the OTP and the queue with a preemption is always superior in performance comparing to the non-preemptive case. Differences

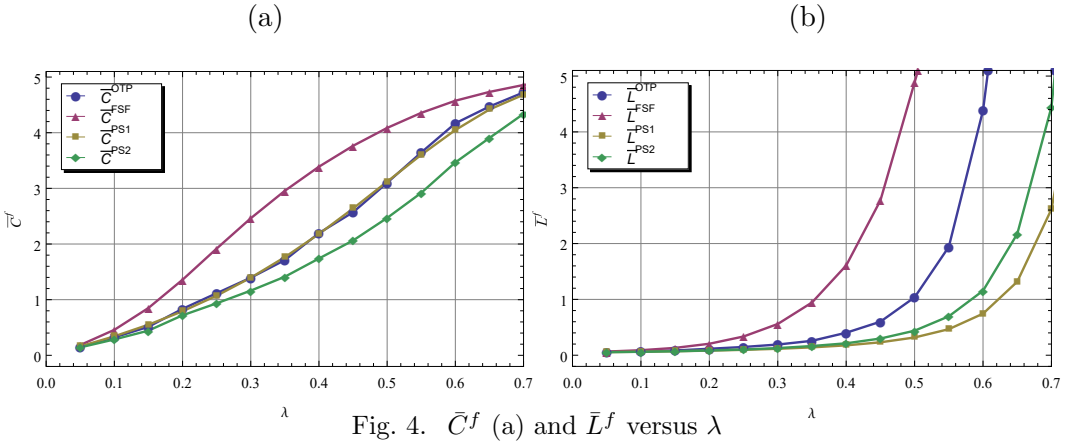


Fig. 4. $\bar{C}^f$ (a) and $\bar{L}^f$ versus λ

between the functions $\bar{N}^{OTP}$ and $\bar{N}^{PS2}$ are almost not visible. This effect is also observed for other values of system parameters. It allows the preemptive system under a threshold policy to be used as an approximation of the original OTP-model. In contrary, the PS2-model exhibits the higher values of queue lengths while the PS1-model – the shortest. The OTP-model also has in average more waiting customers as in FSF-model which is not surprising, since the optimal policy minimizes in our case the mean number of customers in the system but not in the queue. It should also be noted that the higher the degree of heterogeneity of the servers, the greater the differences in performance functions for different models become.

Figure 4 illustrates the influence of λ and model types on the functions $\bar{C}^f$ (figure labeled by "a") and $\bar{L}^f$ (figure labeled by "b"). The functions of the mean number of busy servers for the OTP- and PS1-models are very close to each other. Thus, by subtracting the mean number of busy servers in PS1-model from the mean number of customers in PS2-model, an approximation can be obtained for the mean queue length of the OTP-model. The functions $\bar{C}^{FSF}$ and $\bar{C}^{PS2}$ represent the upper and low bound for $\bar{C}^{OTP}$. The longest busy period appears in FSF-model. In this case the slower servers can be occupied with higher probability and then these servers remain busy for a very long time. As expected, the shortest busy period exhibits the preemptive PS1-model.

Figure 5 shows the effect of the service speed of kth server, $1 \leq k \leq K$, to the mean number of customers $\bar{N}_{L,k}$ served in a busy period (figures are labeled respectively by "a"–"f"). We observe that the slow servers begin to contribute

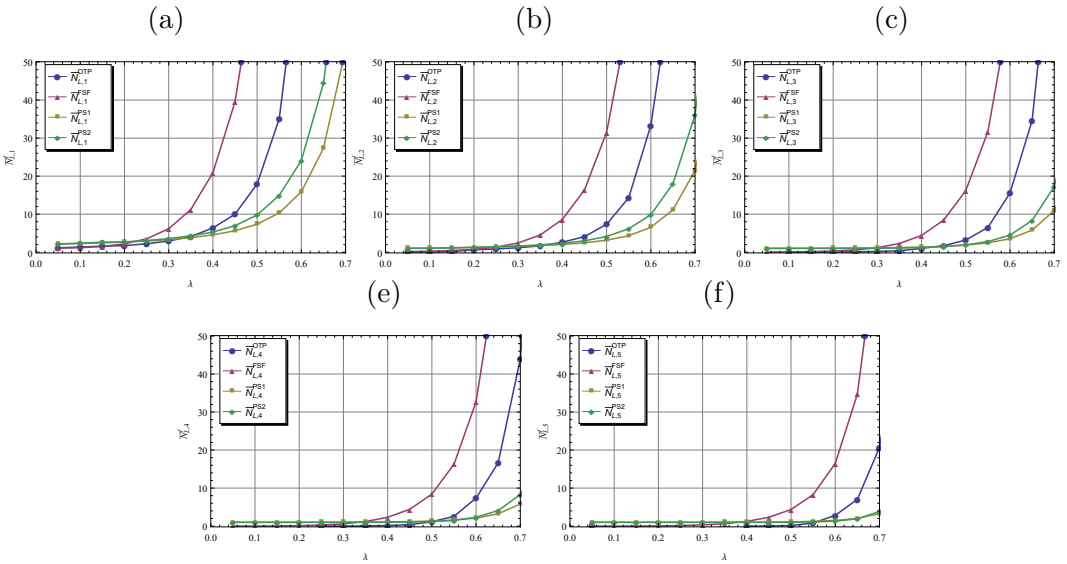


Fig. 5. The mean number of customers $\bar{N}_{L,k}$ versus λ

n	$\lambda = 0.1$	$\lambda = 0.3$	$\lambda = 0.5$	$\lambda = 0.7$
0	0.79903	0.58580	0.47423	0.40562
1	0.94864	0.75918	0.61516	0.52072
2	0.98649	0.84411	0.68819	0.58156
3	0.99963	0.94722	0.77455	0.64604
4	0.99995	0.96327	0.80985	0.67923
5	1	0.97991	0.84369	0.70681
⋮	⋮	⋮	⋮	⋮
10	1	0.99956	0.96445	0.84265
⋮	⋮	⋮	⋮	⋮
20	1	0.99987	0.99772	0.91243
⋮	⋮	⋮	⋮	⋮
40	1	1	0.98523	0.93216

n	$\lambda = 0.1$	$\lambda = 0.3$	$\lambda = 0.5$	$\lambda = 0.7$
0	0.99944	0.87367	0.60804	0.44909
1	0.99992	0.91039	0.61822	0.45087
2	0.99998	0.94535	0.63089	0.45254
3	0.99999	0.97086	0.64667	0.45413
4	1	0.98591	0.64667	0.45568
5	1	0.99361	0.69029	0.45723
⋮	⋮	⋮	⋮	⋮
10	1	0.999993	0.86795	0.46603
⋮	⋮	⋮	⋮	⋮
20	1	1	0.99910	0.53571
⋮	⋮	⋮	⋮	⋮
40	1	1	1	0.99998

Table 1. The probability of the maximum queue length $\bar{Q}^f_{max,n}$ as λ varies for OTP and FSF

to the number of customers served as the intensity of λ increases. The functions $\bar{N}^f_{L,k}$ are proportional to the rate λ, they are simply shifted to the right as λ is getting higher without changing their form. It can be observed also that the FSF-policy maximizes the number of customers served in a busy period at any server. This observation coincides with a statement in [9] that the fastest available server stochastically maximizes the number of service completions.

n	$\lambda = 0.1$	$\lambda = 0.3$	$\lambda = 0.5$	$\lambda = 0.7$
0	0.77220	0.53050	0.40404	0.32626
1	0.95182	0.74672	0.57128	0.45002
2	0.99112	0.86396	0.67401	0.52063
3	0.99851	0.92976	0.74748	0.56866
4	0.99976	0.96533	0.80376	0.60468
5	0.99996	0.98344	0.84775	0.63305
⋮	⋮	⋮	⋮	⋮
10	1	0.99971	0.96288	0.72316
⋮	⋮	⋮	⋮	⋮
20	1	1	0.99951	0.86029
⋮	⋮	⋮	⋮	⋮
40	1	1	1	0.99999

n	$\lambda = 0.1$	$\lambda = 0.3$	$\lambda = 0.5$	$\lambda = 0.7$
0	0.77220	0.53050	0.40404	0.32626
1	0.93781	0.74672	0.57128	0.45002
2	0.98639	0.85561	0.66394	0.51281
3	0.99723	0.91588	0.72366	0.54955
4	0.99945	0.95076	0.77102	0.57612
5	0.99989	0.97293	0.80967	0.59627
⋮	⋮	⋮	⋮	⋮
10	1	0.99923	0.93623	0.66395
⋮	⋮	⋮	⋮	⋮
20	1	1	0.99854	0.79730
⋮	⋮	⋮	⋮	⋮
40	1	1	1	0.99998

Table 2. The probability of the maximum queue length $\bar{Q}^f_{max,n}$ as λ varies for PS1 and PS2

We now focus on the results obtained for the maximum queue length observed during a busy period. To study the influence of system parameters and model type we summarized the results in Table 1 for OTP- and FSF-models and in Table 2 – for PS1- and PS2-models. In tables we vary the rate λ keeping as before other parameters constant. The results compiled and presented in tables correlate with the graphs for the mean length of the busy period. The longer the busy period is, the more likely there will be fewer waiting customers in the queue. In the FSF-model it is more likely that there is an empty waiting line. As λ increases, the queues grow and hence we observe that for all models that the 99th percentile increases.

We have also conducted various experiments where we analyzed the effect of the number of servers, the source capacity, the level of heterogeneity and so on to performance metrics of non-preemptive heterogeneous systems and possible approximations through their preemptive equivalents. Due to the space limitations of the paper, we omit these results. As a generalisation, we can state that the main observations we made in the presented examples remain valid also for other values of system parameters.

6. Conclusion

Finite-source multi-server heterogeneous systems without priority service interruption are described using a multivariate Markov-chains. For such a systems we have found the optimal threshold policy and calculated the corresponding performance measure. Both analytical and numerical studies of such a system face constraints on the dimensionality of the problem, i.e. on the number of servers. In this

paper we have also tried to understand, whether there are simplified variations of the main model which are appropriate for boundary values calculation or even for approximation of the main model but without constraint on the number of servers. We have analyzed non-preemptive and preemptive queues and provided comparison analysis for the performance characteristics.

REFERENCES

1. Chakka R., Mitrani I. (1994) Heterogeneous Multiprocessor Systems with Breakdowns // Theoretical Computer Science. 1994. V. 125. P 91–109.
2. Deslay M., Kolfal B., Ingolfsson A. Maximizing Throughput in Finite-Source parallel queue systems //European Journal of Operational Research. 2012. V. 217. P 554–559.
3. Efrosinin D. Controlled Queueing Systems with Heterogeneous Servers: Dynamic Optimization and Monotonicity Properties of Optimal Control Policies in Multiserver Heterogeneous Servers. VDM Verlag: Saarbr"ucken, Germany, 2008.
4. Efrosinin D. V., Rykov V. V. On Performance Characteristics for Queueing Systems with Heterogeneous Servers // Autom Remote Control. 2008. V. 69. P. 61-–75.
5. Efrosinin D., Stepanova N., Sztrik J., Plank A. Approximations in Performance Analysis of a Controllable Queueing System with Heterogeneous Servers // Mathematics. 2020. V. 8. 1803.
6. Iravani S.M. R., Krishnamurthy V., Chao G. H. Optimal Server Scheduling in Nonpreemptive finite-population queueing systems //Queueing Systems. 2007. V. 55. P. 95–105.
7. Jain M. Finite Source $M/M/r$ Queueing System with Queue-Dependent Servers // Computers and Mathematics with Applications. 2005. V. 50. P. 187–199.
8. Ke J., Wang K. Cost Analysis of the $M/M/R$ Machine Repair Problem with Balking, Reneging and Server Breakdown // Journal of the Operational Research Society. 1999. V. 50. P. 275–282.
9. Righter, R. Optimal Policies for Scheduling Repairs and Allocating Heterogeneous Servers // Journal of Applied Probability. 1996. V. 33. P. 536–547.
10. Stecke K. E. Machine Interference: Assignment of Machines to Operators. Handbook of Industrial engineering, Second edition. 1992. P. 1460–1494.
11. Sztrik J., Roszik J. Performance Analysis of Finite-Source Retrial Queueing Systems with Heterogeneous Non-Reliable Servers and Different Service Policies. Research report, University Debrecen 2001.

12. Sztrik J. Finite-Source Queueing Systems and Their Applications. In M. Ferenczi, A. Pataricza, L. Rnyai, eds. Formal Methods in Computing, Akadémia Kiadó, Budapest, Hungary 2005, P. 311–356.

УДК: 519.218

Применение теории разложимых полурегенерирующих процессов к исследованию системы k-из-n:F с частичным ремонтом*

В.В. Рыков[1,3], Д.В. Козырев[1,2], Н.М. Иванова[1,2]

[1]Российский университет дружбы народов (РУДН), ул. Миклухо-Маклая, д. 6, Москва, Россия, 117198

[2]Институт проблем управления им. В.А. Трапезникова РАН, ул. Профсюзная, д. 65, Москва, Россия, 117997

[3]Институт проблем передачи информации им. А. А. Харкевича РАН, Большой Каретный пер., 19, Москва, Россия, 127051

vladimir_rykov@mail.ru, kozyrevdv@gmail.com, nm_ivanova@bk.ru

Аннотация

В работе исследуется система k-из-n : F с помощью теории разложимых полурегенерирующих процессов. Длительности отказов элементов и всей системы имеют экспоненциальное распределение, а ремонта - произвольное. Рассматривается сценарий частичного ремонта элементов после отказа всей системы. Представлены нестационарные вероятности состояний системы в терминах преобразования Лапласа, вычислены стационарные характеристики.

Ключевые слова: система k-из-n : F, регенерирующий процесс, надежность системы, стационарные вероятности

1. Введение

Теория регенерирующих процессов была предложена Смитом [1] в 1955 г. Эта теория помогает решить множество прикладных задач, в виду чего нашла множество обобщений и применений. Иногда поведение процесса на отдельном периоде регенерации, а также соответствующие вероятности могут быть достаточно сложны для аналитических вычислений, таким образом необходимо более детально исследовать этот процесс. В работе [2] было рассмотрено одно из

*Публикация выполнена при поддержке Программы стратегического академического лидерства РУДН (Рыков В.В., постановка задачи, Козырев Д.В., вывод основных соотношений, Иванова Н.М., проверка результатов). Исследование выполнено при финансовой поддержке РФФИ в рамках научного проекта № 20-01-00575 (Рыков В.В., построение разложимого полурегенерирующего процесса, Иванова Н.М., проведение аналитических расчетов).

обобщений теории Смита, а именно – полумарковские процессы, что привело к развитию полурегенерирующих процессов. Позже эта идея нашла свое продолжение в работе [3], где была преложена теория разложимых полурегенерирующих процессов, которая основана на представлении процесса на вложенном периоде регенерации и его распределения на втором уровне регенерации.

В настоящей статье теория разложимых полурегенерирующих процессов применяется к задаче вычисления стационарных и нестационарных вероятностей состояний системы k-из-n. Эти системы представляют интерес как с теоретической точки зрения, так и практической. В настоящее время предложен ряд модификаций структуры таких систем, в зависимости от их приложений на практике [4]. Системы типа k-из-n можно встретить в областях телекоммуникаций, нефтегазовой отрасли, системах мониторинга, транспорта и т.д.

В серии недавних работ для системы k-из-$n : F$ были вычислены различные характеристики надежности при показательном времени жизни и произвольном распределении времени восстановления элементов системы. Например, в работе [5] была вычислена функция надежности и среднее время жизни системы, в [6] для случая полного восстановления системы 3-из-6 : F найдены стационарные вероятности состояний системы с помощью метода марковизации (метод введения дополнительных переменных), а также проведен анализ чувствительности вероятности безотказной работы системы к форме распределения длительности восстановления.

Целью настоящей работы является демонстрация возможностей теории разложимых полурегенерируюших процессов для вычисления стационарных и нестационарных вероятностей состояний системы k-из-n при показательном распределении времени безотказной работы и произвольном распределении времени ремонта её компонент при сценарии частичного восстановления.

2. Постановка задачи

Рассмотрим ремонтируемую систему k-из-$n : F$, которая выходит из строя при выходе из строя k ее компонент из n. Для такой системы существует как минимум два возможных сценария восстановления системы после выхода из строя:

- **частичный ремонт** наступает при отказе i-го элемента, $i = \overline{1, n-k+1}$, отказавший элемент ремонтируется, после чего начинается восстановление другого элемента (переход в предшествующее состояние);
- **полный ремонт** происходит, когда отказал $k-1$ элемент, система оказывается в состоянии k, и начинается восстановление всех отказавших элементов (переход в нулевое состояние).

В работе рассматривается случай частичного ремонта системы. Введем следующие предположения:

- Время жизни компонент системы - это независимые одинаково распределенные (н.о.р.) случайные величины (с.в.), которые имеют экспоненциальное распределение с параметром α.
- Для восстановления неисправных компонент имеется одна ремонтная единица.
- Длительности ремонта являются н.о.р.с.в. B_i $(i = 1, 2, \dots)$ с произвольной функцией распределения (ф.р.) $B(t) = \mathbb{P}\{B_i \le t\}$.

Для изучения системы введем следующие обозначения.

- $\mathbb{P}\{\cdot\}$, $\mathbb{E}[\cdot]$ — символы вероятности и математического ожидания, в то время как символы $\mathbb{P}_i\{\cdot\}$, $\mathbb{E}_i[\cdot]$ использованы для условных вероятности и среднего значения с учетом начального состояния процесса в i;
- $\lambda_i = (n - i)\alpha$ интенсивность отказа одной из компонент, когда система находится в состоянии i, иногда также используется обозначение $(n - i)\alpha$ для этого значения;
- $E = \{0, 1, \dots k\}$ - пространство состояний системы, где j означает количество отказавших компонентов, а k - состояние отказа системы;
- с этим пространством состояний определяют случайный процесс $J = \{J(t),\ t \ge 0\}$

$$J(t) = j, \text{ если в момент времени } t \text{ система находится в состоянии } j \in E;$$

- T — время до отказа системы, $T = \inf\{t :\ J(t) = k\}$.

Предположим также, что в начале своей работы все компоненты системы находятся в работоспособном состоянии, то есть начальным состоянием процесса J является нулевое, $J(0) = 0$. Также предполагается, что мгновенный ремонт компонент невозможен, а среднее время ремонта ограничено,

$$B(0) = 0, \quad \int_0^\infty (1 - B(t))dt < \infty.$$

В статье вычисляются вероятности состояний системы, зависящие от времени,

$$\pi_j(t) = \mathbb{P}\{J(t) = j\},$$

и стационарные вероятности состояний системы

$$\pi_j = \lim_{t\to\infty} \pi_j(t).$$

3. Вычисление характеристик системы k-из-$n:F$ на периоде полурегенерации

На рис. 1 представлена траектория случайного процесса J для системы k-из-$n:F$ с частичным ремонтом. Процесс J является полурегенерирующим процессом. Моменты регенерации S_n типа j — это моменты окончания ремонта, когда система попадает в состояние j, $J(S_n+0)=j$. Периоды полурегенерации обозначим как $T_n=S_n-S_{n-1}$, а пространство состояний такого процесса как $E_1=\{j:(j=\overline{0,k-1})\}$.

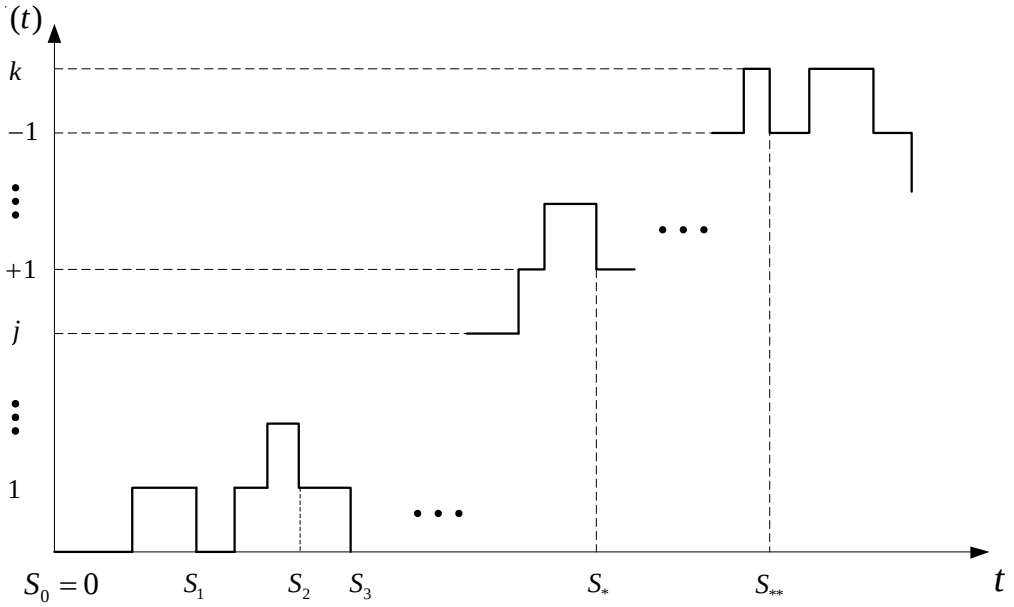


Рис. 1. Траектория процесса J для сценария частичного ремонта системы.

Обозначим через $p_{ij}(t)$ вероятность перехода системы из состояния i в состояние j за время t без восстановления какой-либо компоненты, а через $P_{ik}(t)$ вероятность того, что, за время t система, находясь в состоянии i, выйдет из подмножества рабочих состояний

$$\begin{aligned} p_{ij}(t) &= \binom{n-i}{j-i}(1-e^{-\alpha t})^{j-i}e^{-(n-j)\alpha t}, \\ P_{ik}(t) &= \sum_{j\geq k} p_{ij}(t) = 1 - \sum_{i\leq j\leq k-1} p_{ij}(t). \end{aligned} \tag{1}$$

Обозначим полумарковскую матрицу процесса J как $Q(t) = [Q_{ij}(t)]_{ij\in E_1}$

$$Q_{ij}(t) = \mathbb{P}\{J(S_n + 0) = j,\ T_n \le t \,|J(S_{n-1} + 0) = i\}.$$

Лемма 1. *Элементы матрицы $Q(t)$ можно вычислить с помощью определений (1) следующим образом*

$$\begin{aligned}
Q_{0j}(dt) &= \int_0^t \lambda_0 e^{-\lambda_0 u}\, du\ p_{1j+1}(t-u)B(dt-u),\ \ j = \overline{0, k-2} \\
Q_{0k-1}(dt) &= \int_0^t \lambda_0 e^{-\lambda_0 u}\, du\ P_{1k}(t-u)B(dt-u) \\
Q_{ij}(dt) &= p_{ij+1}(t)B(dt),\ \ (i = \overline{1, k-2},\ j = \overline{i-1, k-2}) \\
Q_{ik-1}(dt) &= P_{ik}(t)B(dt).
\end{aligned} \tag{2}$$

Согласно теории разложимых полурегенерирующих процессов [3], вероятности перехода процесса J $\Pi(t) = [\pi_{ij}(t)]_{i,j\in E}$ могут быть представлены как

$$\Pi(t) = \Pi^{(1)}(t) + H \star \Pi^{(1)}(t), \tag{3}$$

где $\Pi^{(1)}(t) = [\pi^{(1)}_{ij}(t)]_{i,j\in E_1}$ - соответствующие вероятности на вложенном периоде регенерации

$$\pi^{(1)}_{ij}(t) = \mathbb{P}\{J(S_{n-1} + t) = j,\ t \le T_n \,|\, J(S_{n-1} + 0) = i)\}\ \ (i, j \in E_1),$$

а $H(t) = [H_{ij}(t)]_{i,j\in E_1}$ — матрица марковского восстановления с элементами

$$H_{ij}(t) = \mathbb{E}\left[\sum_{n\ge 1} 1_{\{S_n\le t,\ J(S_n)=j\}} \,|\, J(0) = i\right] \equiv \mathbb{E}_i\left[\sum_{n\ge 1} 1_{\{S_n\le t,\ J(S_n)=j\}}\right].$$

Матрица марковского восстановления $H(t)$ может быть вычислена через полумарковскую матрицу $Q(t)$ следующим образом:

$$H(t) = Q(t) + Q \star H(t). \tag{4}$$

Тогда уравнения (3) и (4) в терминах преобразований Лапласа (ПЛ) и Лапласа-Стилтьеса (ПЛС) соответствующих функций принимают вид:

$$\tilde{\Pi}(s) = (I + \tilde{h}(s))\tilde{\Pi}^{(1)}(s), \quad \text{и} \quad \tilde{h}(s) = \tilde{q}(s) + \tilde{q}(s)\tilde{h}(s),$$

откуда

$$\begin{aligned}\tilde{\Pi}(s) &= (I+\tilde{h}(s))\tilde{\Pi}^{(1)}(s) = [I+(I-\tilde{q}(s))^{-1}\tilde{q}(s)]\tilde{\Pi}^{(1)}(s) = \\ &= \left[I+\left(\sum_{n\geq 0}\tilde{q}^n(s)\right)\tilde{q}(s)\right]\tilde{\Pi}^{(1)}(s) = (I-\tilde{q}(s))^{-1}\tilde{\Pi}^{(1)}(s). \qquad (5)\end{aligned}$$

Для вычисления элементов $\pi_{ij}^{(1)}(t)$ матрицы $\Pi^{(1)}(t)$ сформулируем следующую Лемму.

Лемма 2. *Вероятности состояния системы, зависящие от времени, на отдельном периоде полурегенерации принимают вид*

$$\begin{aligned}\pi_{00}^{(1)}(t) &= e^{-\lambda_0 t};\\ \pi_{0j}^{(1)}(t) &= \int_0^t \lambda_0 e^{-\lambda_0 u} p_{1j}(t-u)(1-B(t-u))du \ \ (j=\overline{1,k-1});\\ \pi_{0k}^{(1)}(t) &= \int_0^t \lambda_0 e^{-\lambda_0 u} P_{1k}(t-u)(1-B(t-u))du;\\ \pi_{ij}^{(1)}(t) &= p_{ij}(t)(1-B(t)) \ \ (1\leq i\leq j\leq k-1);\\ \pi_{ik}^{(1)}(t) &= P_{ik}(t)(1-B(t)) \ \ (i=\overline{1,k-1}). \qquad (6)\end{aligned}$$

Таким образом, путём объединения полученных результатов получим следующую теорему.

Теорема 1. *ПЛ нестационарных вероятностей состояний процесса J в матричной форме задается следующим равенством*

$$\tilde{\Pi}(s) = (I-\tilde{q}(s))^{-1}\tilde{\Pi}^{(1)}(s),$$

где компоненты $\tilde{\pi}_{ij}^{(1)}(s)$, $\tilde{q}_{ij}(s)$ и $\tilde{\pi}_{ij}(s)$ вычисляются с помощью (1), (2) и (6) и переходом к соответствующим ПЛС.

Стационарное распределение вероятностей состояний процесса можно вычислить, перейдя к пределу в последнем равенстве.

Теорема 2. *Стационарный режим рассматриваемой системы при сценарии частичного ремонта существует и вероятности состояний равны*

$$\pi_j = \frac{1}{m}\sum_{0\leq l\leq j\wedge(k-1)} \alpha_l \tilde{\pi}_{lj}^{(1)}(0) \qquad (j=\overline{0,k}), \qquad (7)$$

где $\vec{\alpha} = \{\alpha_l : \ l \in E_1\}$ *удовлетворяет следующей системе уравнений*

$$\vec{\alpha}' = \vec{\alpha}' \tilde{q}(0), \qquad \sum_{0 \le l \le k-1} \alpha_l = 1. \tag{8}$$

$\tilde{\pi}_{ij}^{(1)}(0)$ *могут быть вычислены из (6) Леммы 2, а* m *— среднее значение полурегенерирующего периода,* $m = \lambda_0^{-1}(\alpha_0 + \lambda_0 b)$*,* α_0 *— начальное распределение процесса,* $\vec{\alpha}^{(0)} = \{\alpha_i^{(0)} : \ i \in E_1\}$*.*

4. Заключение

В работе предложен новый метод вычисления характеристик систем k-из-$n : F$. Теория разложимых полурегенерирующих процессов использована для вычисления основных характеристик системы для сценария частичного ремонта системы и произвольного распределения времени ремонта отказавших компонент. Представленные результаты совпадают с полученными ранее с помощью метода марковизации, а также с теми, что вычисляются с помощью простого Марковского процесса.

Литература

1. Smith, W.L. Regenerative stochastic processes. Proc. R. Soc. Ser. A 1955, 232, DOI:10.1098/rspa.1955.0198.
2. Cinlar, E. On semi-Markov processes on arbitrary space. Proc. Camb. Philos., Math. Proc. Camb. Philos. Soc. 1969, 66, pp. 381–392.
3. Rykov, V. V. Decomposable Semi-Regenerative Processes and Their Applications. Lap Lambert Academic Publishing: Berlin, Germany, 2011, p. 75.
4. Shepherd, D.K. k-out-of-n systems. In F, Ruggeri, R. Kenett & F.W. Faltin (eds.) Encyclopedia of statistics in quality and reliability. Chichester, England: Wiley, 2008.
5. Rykov, V.; Kozyrev, D.; Filimonov, A.; Ivanova, N. On Reliability Function of a k-out-of-n System With General Repair Time Distribution. Probability in the Engineering and Informational Sciences, 2020, pp. 1–18, DOI:10.1017/S0269964820000285
6. Rykov V.V., Ivanova N.M., Kozyrev D.V. Sensitivity Analysis of a k-out-of-n:F System Characteristics to Shapes of Input Distribution. In: Vishnevskiy V.M., Samouylov K.E., Kozyrev D.V. (eds) DCCN 2020. Lecture Notes in Computer Science, vol 12563. Springer, Cham. 2020, pp. 485–496. DOI:10.1007/978-3-030-66471-8_37

UDC: 004.7

Identification of narrowband wireless communication networks systems and Internet of Things devices using Blockchain technology

A.V. Pomogalova[1], D.D. Sazonov[1], E.A. Donskov[2], A.S. Borodin[3], R.V. Kirichek[1]

[1]The Bonch-Bruevich Saint-Petersburg State University of Telecommunications, Saint-Petersburg, Russian Federation
[2]St. Petersburg Institute for Informatics and Automation of the Russian Academy of Sciences, Saint-Petersburg, Russian Federation
[3]PJSC Rostelecom, Moscow, Russian Federation

alya.pomo@gmail.com, dim-saz@yandex.ru, radion2002@gmail.com, aleksey.borodin@rt.ru, kirichek@sut.ru

Abstract

The paper explores the possibility of interaction between the Handle System identification system and the blockchain network in order to optimize the operation of the identification system. The work covered the organization of the blockchain network, the architecture of the model stand developed, as well as a number of studies reflecting the results of the interaction of the identification system with the blockchain network in terms of the time of creation of an entity in LHS and the record thereof in the blockchain network, as well as the time of response of the blockchain network in case of query of data from it. As part of this work, the blockchain network acts as a tool to log all the changes of the identification system in order to prevent counterfeiting and substitution of the identification data of devices and sensors of the Internet of Things.

Keywords: blockchain, DOA, IoT, smart contract, identification, identification system, digital object architecture, Internet of Things

1. Introduction

The growth of various devices and sensors of the Internet of Things, the necessity of their organized interaction and accounting is a topical and long-term problem since the beginning of the development of the Internet of Things as a concept. Thus, until a few years ago, the number of such sensors and devices was in the thousands, whereas

The publication has been prepared with the support of the grant from the President of the Russian Federation for state support of leading scientific schools of the Russian Federation according to the research project SS-2604.2020.9.

today it is in the millions and billions. Besides servicing Internet of Things devices and debugging their interaction, one of the key problems remains identification of such devices.

Earlier authors of the work carried out an extensive review and analysis of the existing systems of identification of narrowband wireless communication networks systems and Internet of Things devices, further confirming the need for the development of a single universal identifier for the Internet of Things. The International Telecommunication Union (ITU-T) has formulated common requirements for an identification system that would address a number of existing problems. The authors' research [1, 2] concluded that the DOA digital object architecture was suitable. The handle identifiers used in the DOA architecture are unique and consistent in the global namespace, guaranteeing the resolution of the digital object to the valid information when the client requests it [3].

But in order to meet all the requirements formulated by ITU-T, the identification system must be secure and ensure that there is no possibility of uncontrolled modification of the data of a digital entity. Part of this problem is solved by using a distributed administration system used in solutions based on the architecture of DOA digital objects. Administration is based on the use of asymmetric access keys.

However, in addition to distributed administration, it is required to add the ability to unambiguously establish a chain of changes in a digital entity at all stages of its life cycle.

For this reason, the authors of the work made an assumption about the possibility of using some of the properties and functions of distributed ledger technologies to prevent the substitution and forgery of identification data. The blockchain technology based on the Ethereum open source solution was used as a distributed ledger technology.

The main goal of this work is to study the possibility of interaction of the architecture of digital objects (DOA) with a blockchain network to control changes in the identification system. To achieve this goal, the authors set up a model stand and performed a number of experiments.

2. Related Works

The research carried out by the authors within the framework of the proposed concept showed that today there are practically no experiments related to the selected identification system. Existing works are devoted to research the possibility of combining the Internet of Things sphere and blockchain technology, including some questions of identification, but are more narrowly focused [5-7]. There is also a lot of research on the possibility of using blockchain technology to store confidential data [8-10]. Within the framework of this paper, the authors propose a more generalized

solution, since the identification system used is applicable to many existing industries and acts as a trusted source of data storage, which is the first step in protecting identification data.

3. Features of Blockchain Network Setup

Today, there are a large number of different blockchain solutions in the world, which differ in the goals of use, performance, requirements for the parameters of validator devices and other devices that support the network, various consensus algorithms and features of the network architecture. As part of this work, the geth client was chosen as a blockchain network for the study, which allows setting up a private instance of the Ethereum blockchain network. This blockchain network operates on the Proof of Work consensus algorithm and is the first platform that made it possible to fully implement the idea of smart contracts. A smart contract within the selected blockchain network means a computer algorithm written in the Solidity programming language that allows performing various actions in the blockchain network, for example, writing and reading data. Data immutability is considered a key feature of blockchain networks, which is an important factor for any identification systems. Data immutability means the fact that changing the instances of the block chain on all devices is practically impossible not only in terms of the number of devices containing the current instance of the block chain and constantly synchronizing with each other, but also because of the complexity of calculating each of the sequentially connected and cryptographically protected blocks for their complete replacement.

The Ethereum blockchain instance configured for research consists of three synchronized devices configured on the basis of three virtual machines. To configure the network, a genesis.json file was created, which contains all the key parameters of the future network, such as the network instance number, block generation complexity, forks occurring in the network and block numbers in which forks similar to the main Ethereum network will occur. Forks are required to use all existing functions of the blockchain network and smart contracts. For each virtual machine, miner accounts were also configured so that each of the three virtual machines performed not only the functions of a full node of the blockchain network, but also a miner that creates blocks and processes transactions. After configuring all three network nodes and synchronizing them, a smart contract was developed that performs the functions of recording and reading information about assigned identification numbers.

4. Model Stand Architecture

The model bench device interaction scheme is shown in Fig. 1. The internal interaction of the blockchain nodes (Geth Ethereum Node 1-3) is performed using

the static-nodes.json file located on each of the virtual machines in the blockchain network containing the addresses of all three nodes on the network and the ports to which they should be addressed. For external interaction with the blockchain network the web3.js library is used as well as Brownie - a Python-based framework for developing and testing smart contracts.Thus, an HTTP server was also configured for communication, which accepts Rest API requests and sends queries to the blockchain network to record new data or read existing data. On the Rest API client side queries are possible both to the blockchain network and to the LHS database.

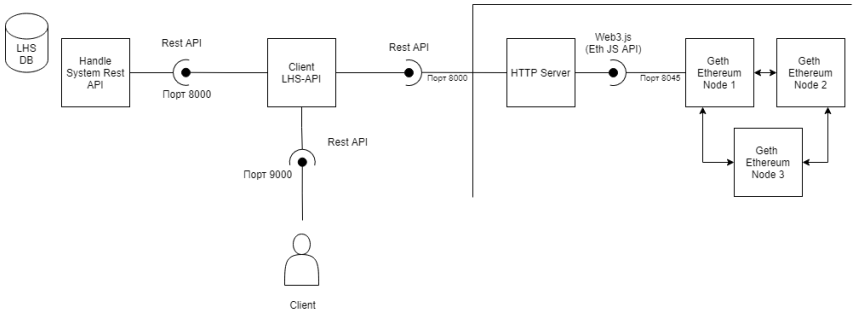


Fig. 1. Model stand interface diagram

The model stand developed makes it possible to evaluate the possibility and effectiveness of the application of blockchain network as a duplicate trusted distributed database, which acts as a reference repository.

5. Research Script

As part of the research, the authors evaluated the possibility of interaction between the blockchain network and the identification system of narrowband wireless communication devices and systems. The DOA digital object architecture has been chosen as the most efficient and effective system for identification of narrowband wireless communication networks systems and Internet of Things devices, according to previous authors' studies. The Handle System was chosen as an implementation of the DOA concept.

The key objective of the study was to assess the feasibility of validating all processes in the identification system using blockchain technology in particular to log all changes in the descriptor.

In the developed test system the main interaction of the client is via HTTP to Rest API service LHS-API, as shown in Fig. 1. This service acts as a "service-facade" providing end-to-end functionality for the client to interact with the Handle System infrastructure and the blockchain network. The LHS-API service provides

the client with the functionality to create, modify and read data of digital objects through interaction with the identifier handleID and Handle System. LHS-API also interacts via Rest Api with a blockchain platform to store data on changes in the meta-information of the digital object and the change chain. Fifty tests were run to measure the response time of the LHS-API service. Figure 2 shows a graph where shown the LHS-API response time when creating a descriptor, as well as writing to a blockchain in ms.

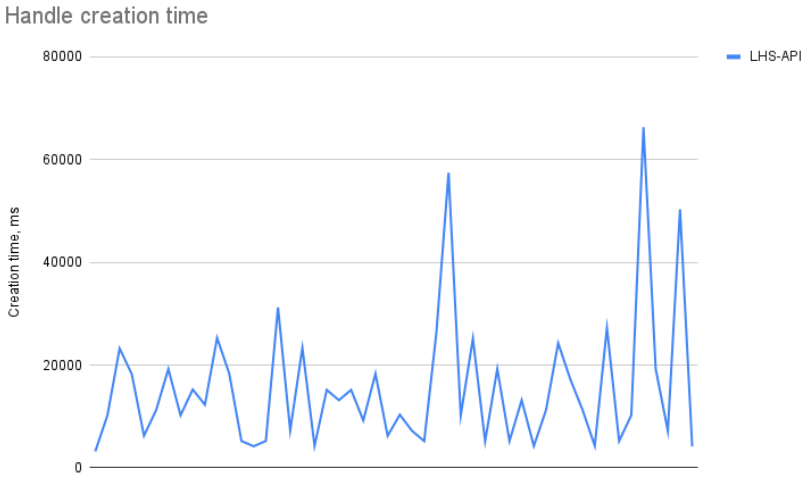


Fig. 2. LHS-API response time when creating a descriptor (ms)

Figure 3 shows a graph where shown resolution time of the descriptor as well as the capture of the chain from the blockchain in ms as well as time of chain extraction from blockchain in ms.

6. Conclusion

From the creation time schedule, it can be seen that the creation of a descriptor in LHS and then the subsequent synchronous recording of data in the chain can be as high as 60 seconds. Based on the results of testing the LHS system under similar conditions without the blockchain system [5] the service response time for creating a new descriptor is increased precisely because of synchronous writing in the blockchain. This is understandable, as the system needs to perform sufficiently resource- and time-consuming calculations to write a block. In order to speed up the response time, it is advisable to write to the system with an asynchronous query after successfully adding data to LHS.

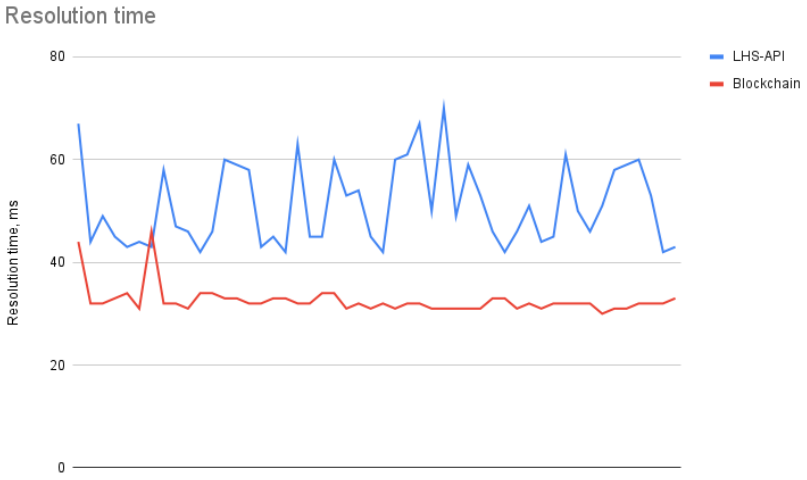


Fig. 3. Resolution time of the descriptor (ms)

The resolution time of the descriptor via the LHS-API with synchronous approach to the blockchain behind the given chain remains rather small, the read time of the blockchain is tens of milliseconds, which was sufficient performance of modern web services.

The time schedule of the blockchain response to read is presented separately. From the received time values it is evident that the greatest contribution to the total response time of the LHS-API service to the descriptor resolution is exactly the request to the blockchain.

The LHS service solves this problem by caching a descriptor resolution response to a digital object to reduce response time. At the same time, it is calculated in the system that the digital object is rarely modified and the data cached remains relevant for a sufficient period of time (up to 24 hours). With the data recorded in the blockchain, it is not advisable to do so, as the chain can be supplemented continuously and it is necessary to see the current state of the system by blockchain casting.

REFERENCES

1. Sazonov D., Kirichek R.,Borodin A. Implementation of authentication and authorization system based on digital object architecture // 11TH INTERNATIONAL CONGRESS ON ULTRA MODERN TELECOMMUNICATIONS AND CONTROL SYSTEMS AND WORKSHOPS (ICUMT). 2019.

2. Sazonov D., Kirichek R. Digital object architecture as an approach to identifying internet of things devices //COMMUNICATIONS IN COMPUTER AND INFORMATION SCIENCE. 2019.
3. Sazonov D., Kirichek R. Identification System Model for Energy-Efficient Long Range Mesh Network Based on Digital Object Architecture //In: Vishnevskiy V.M., Samouylov K.E., Kozyrev D.V. (eds) Distributed Computer and Communication Networks: Control, Computation, Communications. DCCN 2020. Communications in Computer and Information Science, vol 1337. Springer, Cham. 2020.
4. H. Wang, X. Yu, J. Peng, L. Zhang and T. Xu. Design of Power Material Management System Basing on Internet of Things Identification and Blockchain //2020 IEEE Conference on Telecommunications, Optics and Computer Science (TOCS). 2020. P. 160-163.
5. T. Kirks, T. Uhlott and J. Jost. The Use of Blockchain Technology for Private Data Handling for Mobile Agents in Human-Technology Interaction // 2019 IEEE International Conference on Cybernetics and Intelligent Systems (CIS) and IEEE Conference on Robotics, Automation and Mechatronics (RAM). 2019. P. 445-450.
6. M. H. Salih Mohammed. A Hybrid Framework for Securing Data Transmission in Internet of Things (IoTs) Environment using Blockchain Approach // 2021 IEEE International IOT, Electronics and Mechatronics Conference (IEMTRONICS). 2021. P. 1-10.
7. T. M. Roopak, R. Sumathi. Electronic Voting based on Virtual ID of Aadhar using Blockchain Technology // 2020 2nd International Conference on Innovative Mechanisms for Industry Applications (ICIMIA). 2020. P. 71-75.
8. N. Chalaemwongwan, W. Kurutach. A Practical National Digital ID Framework on Blockchain (NIDBC) // 15th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON). 2018. P. 497-500.
9. N. Choi and H. Kim. Hybrid Blockchain-based Unification ID in Smart Environment // 22nd International Conference on Advanced Communication Technology (ICACT). 2020. P. 166-170.
10. N. Kshetri, J. Voas. Blockchain-Enabled E-Voting // in IEEE Software. 2018. V. 35. N. 4. P. 95-99.

UDC: 004.7

Analysis of Network Security Issues in the Join Procedure of LoRaWAN

Duc Tran Le[1,*], Tai Duc Nguyen[1], Luong Ba Le[1], Van Dai Pham[2], Ruslan Kirichek[2]

[1]The University of Danang - University of Science and Technology, Nguyen Luong Bang 54, Danang, Vietnam
[2]Bonch-Bruevich Saint-Petersburg State University of Telecommunications, Saint Petersburg 193232, Russia
[*]Corresponding author

letranduc@dut.udn.vn, ductai26998@gmail.com, lebaluong33@gmail.com, fam.vd@spbgut.ru, kirichek@sut.ru

Abstract

In the LoRaWAN network, the end devices need to make join-request by sending packets to the Network Server during the join procedure process. This procedure has some vulnerabilities that the attackers can exploit and attack network operations. In this paper, we will review, analyze, and compare some methods of exploiting vulnerabilities that are usually performed in the join procedure. In addition, we compare the advantages and disadvantages of existing solutions to reduce the above security issues.

Keywords: LoRaWAN, LoRa, join procedure, jamming attack, replay attack

1. Introduction

Nowadays, with LoRa technology, we can transmit data from a distance up to several kilometers without power amplifier circuits in LoRaWAN network. LoRa also has advantages in battery life optimization, easy deployment, and robustness to interference.

However, besides the advantages, LoRaWAN also has a lot of limitations, one of which is security issues. Currently, there are quite a few studies interested in this topic. The studies on this topic by other researchers only present the attacks and the measures to prevent those attacks. In this paper, in addition to giving an overview of the attacks on the LoRaWAN network, we also compare these attacks, while giving an assumption about combining different types of attacks. And then, we compare some existing methods to reduce the impact of these attacks in the join procedure.

2. Security Issues in LoRaWAN

LoRaWAN uses two security layers: the network layer and the application layer. The network layer is responsible for validating the access of the end devices in the network, while the application layer ensures that the network operator does not have access to the end-user data [1]. However, it still has security issues that attackers can take advantage of accessing and destroying networks or stealing important information [2].

In LoRaWAN, there are three processes of communication and data transmission: (1) Communication between the end devices and Network Server; (2) Communication between the servers; (3) Communication between the Application Server and the user.

Each process has its security vulnerabilities [3]. However, because processes (2) and (3) use other types of networks such as LTE, Ethernet, consideration of security issues in these processes are not related to LoRa Technology. Therefore, in this paper, we only focus on analyzing the security issues of the communication process between the end devices and the Network Servers.

It should be noted that the latest version of the LoRaWAN specification was a significant advance of the protocol features and addressed many security problems previously reported. However, according to the analysis in [4], there are still a lot of security threats existing. LoRaWAN-based systems can be attacked by DOS denial attacks aimed at wireless signal transmissions [5]. In addition, data in the LoRaWAN network is not fully encrypted [6], which leads to information leakage or communication between servers is interfered through man-in-the-middle (MITM) attacks.

The main contributions of this paper are:

- Analyzing and comparing the popular types of attacks taking place in the join procedure

- Analyzing and comparing some solutions to reduce the effects of these types of attacks

3. Join Procedure in LoRaWAN

Each end device has two unique root keys, which are embedded during fabrication: Application key (**AppKey**) and Network key (**NwkKey**). This pair of keys is used to derive the session keys (application session key (**AppSKey**) and network session key (**NwkSKey**)). The AppSKey is unique for each end device. It is only known by the end device and the application server. The NwkSKey is only known by the end device and the Network Server and is unique for each end device.

When a new LoRa-end device is added to a LoRa network, it should go through an activation process. Through this process, both session keys are shared between the

end device and the Network Server. Currently, LoRa provides the following two types of activation methods: Activation-By-Personalization (**ABP**) and Over-The-Air Activation (**OTAA**), which is called Join Procedure. Because in the ABP method, an end device can belong to a particular LoRa network without performing a join procedure under certain conditions, we will not recall it now. The readers can find more details about ABP in [3].

The join procedure consists of two messages exchanged between the end device and the Network Server, namely join-request and join-accept. The join-request message is sent by the end device to the Network Server. The Network Server responds to the join-request message with a join-accept message if the end device is permitted to join a network. Figure 1 depicts the join procedure.

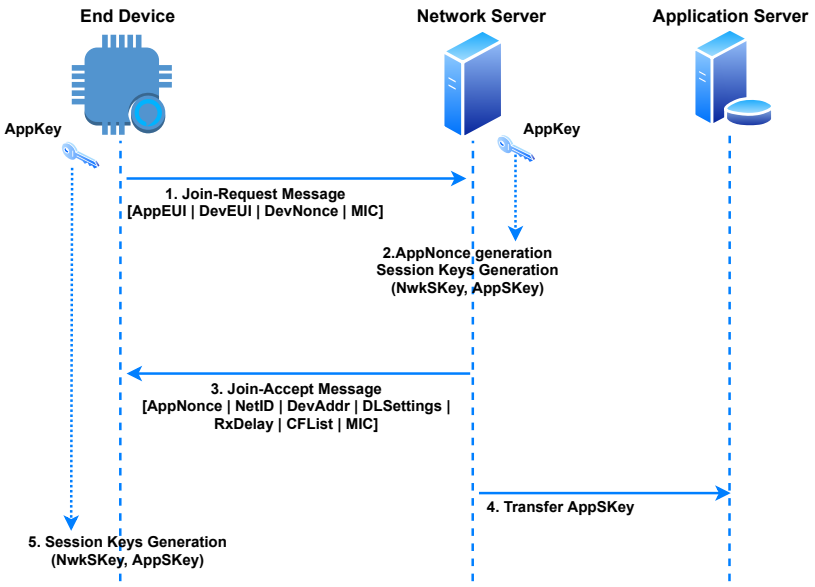


Fig. 1. Join Procedure in LoRaWAN

Join Procedure Steps:

- **Join-Request Message**: It consists of the end device identifier (**DevEUI**), the application identifier (**AppEUI**), and a nonce of 16 bits (**DevNonce**). The DevNonce is a random sequence number starting by 0 when the device is initially powered up and incremented with every join-request by the end device. A DevNonce value shall never be reused for a given AppEUI value. For each end device, the Network Server keeps track of the last DevNonce value used by the end device and ignores join-requests if DevNonce is not incremented.

A message integrity check (**MIC**) value of join request, which is calculated by end device and an AppKey, is preshared between the end device and Network Server. It should be noted that the join-request message is not encrypted (for the versions before version 1.0.3).

- **Authentication and Session Keys Generation**: At the reception, the Network Server verifies whether the end device is permitted to join the network or not based on the DevNonce. If the DevNonce in the join request is previously used, the Network Server determines that the message is invalid and that the join process will fail. If the message is valid, the Network Server authenticates the end device with the MIC value. If the end device passes the authentication, the Network Server generates a NwkSKey and an AppSKey. **AppNonce** is a unique random counter number generated by the Network Server and sent back to the end device. **NetID** is a 24-bit field network identifier to separate addresses of geographically duplicated LoRa networks.

- **Join-Accept Message**: A join-accept message contains **AppNonce**, **NetID**, **DevAddr**, **DLSettings**, **RxDelay**, and **CFList**. The **DevAddr** is a 32-bit identifier of the end device within the current network. **DLSettings** contains several values related to the downlink configuration. **RxDelay** is a delay between the transmission and reception process. **CFList** is an optional field that is about channel frequencies. Finally, the whole join-accept message is encrypted with the AppKey.

- **Transfer AppSKey**: Since the AppSKey is devised to secure end-to-end communications between the end device and the Application Server, it should be transferred from the Network Server to the Application Server.

- **Session Keys Generation**: After receiving the join-accept message, the end device decrypts it and generates session keys using extracted parameters.

From the analysis of the operation of LoRaWAN above, it can be seen that LoRaWAN opposites with quite a lot of threats. In fact, the Network Server may be a web server and a DoS attack may be launched to prevent all communications to this web server. Another attack can rely on the characteristics of AppNonce, which is not registered by the end device in the join-accept message (while DevNonce in join-request message shall be registered to avoid replay attack). An attacker may register a fake join-accept message and when the end device sends another join-request, that fake registered message is used to respond. In addition, we can see that the AppKey is used to generate NwkSKey and AppSKey. An inner attacker such as a network operator can completely decrypt the application data and even modify it. So, the Application Server must trust the Network Server to not modify the data. In the next section, we will consider in detail several typical attack types in the join procedure.

4. Jamming Attacks on Join Procedure

LoRaWAN is a wireless network, so it cannot avoid signal jamming attacks. A jamming attack is a denial of service attack on wireless channels. Its purpose is to interfere with the access between the end device and the Network Server by emitting a noise signal in the vicinity of the LoRa end devices. The scenario of this attack was simulated by researchers and presented in [5]. In the scope of this paper, we will describe the two main different techniques to perform an attack jamming.

Triggered jamming To avoid simultaneous transmissions, LoRa radio modules can scan a certain channel to detect whether there is an ongoing LoRa transmission or not. However, the attackers can take advantage of this capability to detect activity on the channel and deploy a fake LoRa device to trigger a jamming attack to the transmission. It leads to increase packet loss in the network. This type of attack is easy to deploy because there is no need to perform modulation processes or decode any signal.

Selective jamming Triggered jamming will interfere with all devices on a certain frequency so it is easily detected. To avoid detection, the attackers can deploy another type of jamming that is Selective Jamming. Selective jamming only interferes with devices or messages that have been selected before. It is difficult to distinguish between conventional incidents during transmission and purposeful attacks. However, due to its characteristics, to deploy this kind of attacks, the attackers need to perform a series of steps such as: (i) detect a LoRaWAN packet, (ii) start receiving that packet, then (iii) abort receiving that packet if received data triggers the jamming policies; (iv) immediately jam the channel if all configurations are set. This whole process increases the processing time and deployment of attacks due to processing many messages before jamming. Therefore, with a certain period, the probability of the successive jamming of the selective jamming method will be lower than the triggered jamming.

5. Replay Attacks on Join Procedure

The replay attack is a kind of attack, which is based on recording and replaying the same message transmitted by the end devices. As mentioned above, join-request messages are not encrypted (for the versions before version 1.0.3) so the attackers can take advantage of this weakness to steal the information of the packet and perform the replay attack. Through the replay attack, the attackers may have valid access to the LoRaWAN network and they will distribute malware in that network, as well as carry out other destructive activities, which leads to serious consequences. From version 1.0.3, the join-request message is registered with 4 bytes MIC and it has been encrypted. A DevNonce value is also used to prevent the replay attack. However, these nonces are produced by using a random-number-generator with

limited capabilities such as a limited pool of numbers (each time a number from this pool is selected randomly, the probability of selecting the same number increases as time advances), which might end up with repeating itself after some certain usage time. In addition, with some special kinds of jamming techniques (as presented below), this DevNonce number pool can be diminished in a short period.

To carry out a replay attack, we need to proceed in three stages: (i) install equipment to sniff packets in the vicinity of the target, (ii) analyze the packets and extract information such as AppEUI, DevEUI, DevNonce, (iii) perform the attack on the Network Server by sending continuous join-request messages. Under this attack, the Network Server will remove or do not respond to the requests from the real end devices. The authors in [7] pointed out that such an attack needs to take T_r days to take place.

$$T_r[days] = \frac{N_D + 1[DevNonce]}{f_J[\frac{DevNonce}{days}]} \tag{1}$$

where f_J is the number of valid procedures involved every day on each end device, N_D is the number of DevNonce values that were previously used and hosted by the Network Server. In most cases, T_r is a huge number, but still, there are exceptions, for example when the device is reset when participating in the network.

In addition to the mentioned-above types of attacks, there are currently some combinations of different attacks. The attackers can completely combine the selective jamming attack with the replay attack, the wormhole attack or the sinkhole attack for maximum effectiveness. The analysis of these combined attacks is out of the scope of this paper, so we will compare and analyze them in our future research.

In Tab. 1, we compare these two types of attacks in the join procedure.

6. Comparison of Existing Solutions

In this section, we will compare the existing solutions for the above-mentioned attacks. The results of this comparison are presented in Tab. 2.

Table 1. A comparison between the jamming attacks and the replay attacks

Criteria	Jamming Attact		Replay attacks
	Triggered Jamming	**Selective Jamming**	
Consequence	Causing network congestion and denial of service on a certain channel in the vicinity of the interference-causing equipment	Only causes an impact on the selected target	Spreading the malware, causing loss and misleading information
Deployment	Easy to configure and perform. It is suitable when there are many targets	It is more complex than triggered jamming, but it is simpler than a replay attack to perform	It is complex, higher technical requirements, and need more devices to perform
Success rate	High success rate	The success rate is lower than triggered jamming	The success rate is much lower than jamming attacks and waiting time can be very long
Detection	Easy to be detected	Difficult to be detected. It is similar to conventional incidents	Difficult to be detected
Popularity	Very popular	Very popular	Relatively common

Table 2. Comparison of existing solutions

Solution	Description	Attack Type	Advantages	Disadvantages
Creating dense LoRa networks with overlapping coverage regions [8]	Deploying LoRaWAN end devices within the range of multiple gateways makes the jamming hard to be performed because the jammers are difficult to prevent the end devices to connect to a gateway. When using this technique, the jamming will become more difficult because the attacker must identify all the gateways that the end devices can connect to.	Jamming attacks, combined attack.	No need to modify the application level.	High cost, inefficient for jamming attacks that have an influence range wider than the coverage.
Optimizing the channel hopping usage [8].	LoRa devices hop between multiple channels when sending messages to reduce collisions. The more channels used, the more complex the jammer has to be, as it needs to listen on all of those channels.	Jamming attacks, combined attacks	The built-in mechanism in LoRa specification, easy to configure.	High cost, high energy consumption.

Continued on next page

Table 2. Comparison of existing solutions (cont.)

Solution	Description	Attack type	Advantages	Disadvantages
Using higher spreading factor (SF) in Chirp Spread Spectrum (CSS) techniques [8].	The higher SFs require higher dB differentials between the jammer and target message.	Jamming attacks, combined attacks.	Increasing jamming process time and limiting multiple transmissions of the jammer.	High cost, high energy consumption.
Analyzing the transmission rate [8].	Performing traffic analysis and profiling (at the gateway or server level). If the sending rate of the end device is aware, the Network Server can identify abnormal traffic patterns.	Jamming attacks, replay attacks, combined attacks.	Only need to apply at the Network Server and it can react accordingly to the unplanned changes.	Difficult to determine traffic patterns and it is inappropriate when the deviation in the transmission time of the packets is small.
Encrypting the packet using the XOR algorithm [9].	Using a token generated from the previous session keys to XOR with a join message to create a new encrypted packet. Only the end device and Network Server know this token and the attacker could hardly get the token to decode.	Replay attacks, combined attacks.	Simple, high-security efficiency.	Need more time to encode and decode the data at the end device and Network Server.

7. Conclusion

In this paper, we have presented an overview and comparison of two typical types of attacks in the join procedure: Jamming Attacks and Replay Attacks. In addition, we also made a comparison of existing solutions to reduce these types of attacks.

Our research was made with LoRaWAN version 1.0.3. In the process of completing this study, a new LoRaWAN version (version 1.0.4) has been released with some changes. Therefore, there may be some characteristics outlined in the paper that would not be true for version 1.0.4. We will conduct research and update the changes in version 1.0.4 shortly and will update the changes in our next study.

Acknowledgments

The publication has been prepared with the support of the grant from the President of the Russian Federation for state support of leading scientific schools of the Russian Federation according to the research project Scientific School–2604.2020.9.

REFERENCES

1. L. Alliance, White paper: A technical overview of lora and lorawan, The LoRa Alliance: San Ramon, CA, USA (2015) 7–11.
2. M. Eldefrawy, I. Butun, N. Pereira, M. Gidlund, Formal security analysis of lorawan, Computer Networks 148 (2019) 328–339.
3. I. Butun, N. Pereira, M. Gidlund, Security risk analysis of lorawan and future directions, Future Internet 11 (1) (2019) 3.
4. I. Butun, N. Pereira, M. Gidlund, Analysis of lorawan v1. 1 security, in: Proceedings of the 4th ACM MobiHoc Workshop on Experiences with the Design and Implementation of Smart Objects, 2018, pp. 1–6.
5. E. Van Es, H. Vranken, A. Hommersom, Denial-of-service attacks on lorawan, in: Proceedings of the 13th International Conference on Availability, Reliability and Security, 2018, pp. 1–6.
6. M. Vanhoef, F. Piessens, Advanced wi-fi attacks using commodity hardware, in: Proceedings of the 30th Annual Computer Security Applications Conference, 2014, pp. 256–265.
7. S. Zulian, Security threat analysis and countermeasures for lorawan join procedure, in: Master Thesis,Universit a degli Studi di Padova, 2016.
8. E. Aras, N. Small, G. S. Ramachandran, S. Delbruel, W. Joosen, D. Hughes, Selective jamming of lorawan using commodity hardware, in: Proceedings of the 14th EAI International Conference on Mobile and Ubiquitous Systems: Computing, Networking and Services, 2017, pp. 363–372.
9. S. Na, D. Hwang, W. Shin, K.-H. Kim, Scenario and countermeasure for replay attack using join request messages in lorawan, in: 2017 international conference on information networking (ICOIN), IEEE, 2017, pp. 718–720.

Научное электронное издание

Распределенные компьютерные и телекоммуникационные сети: управление, вычисление, связь (DCCN-2021)

МАТЕРИАЛЫ XXIV МЕЖДУНАРОДНОЙ НАУЧНОЙ КОНФЕРЕНЦИИ

Под общей редакцией д.т.н. В.М. Вишневского, д.т.н. К.Е. Самуйлова

Составитель: к.ф.-м.н. Козырев Дмитрий Владимирович

Локальное электронное издание
Номер госрегистрации в НТЦ «Информрегистр» 0322103543

Мин. системные требования:
Pentium 4, Acrobat reader 4.0 и выше

Дата подписания к использованию: 01.11.2021
1 электронно-оптический диск (CD-R), 24,9 Мб, Тираж 100 экз.

Федеральное государственное бюджетное учреждение науки
Институт проблем управления им. В.А. Трапезникова
Российской академии наук
117997, Россия, Москва,
ул. Профсоюзная, д. 65
www.ipu.ru