Discrete & Continuous Models
& Applied Computational Science
ISSN 2658-7149 (online), 2658-4670 (print)

2022, 30 (1) 5–20
http://journals.rudn.ru/miph

*Research article*

UDC 519.872:519.217
PACS 07.05.Tp, 02.60.Pn, 02.70.Bf
DOI: 10.22363/2658-4670-2022-30-1-5-20

# Performance analysis of queueing system model under priority scheduling algorithms within 5G networks slicing framework

**Kpangny Yves Berenger Adou,**
**Ekaterina V. Markova, Elena A. Zhbankova**

*Peoples' Friendship University of Russia (RUDN University)*
*6, Miklukho-Maklaya St., Moscow, 117198, Russian Federation*

(received: December 30, 2021; revised: January 21, 2022; accepted: February 18, 2022)

**Abstract.** A new era is opening for the world of information and communication technologies with the 5G networks' release. Indeed 5G networks appear in modern wireless systems as solutions to "traditional" networks' inflexibility and lack of radio resources problems. Using these networks the operators can expand their services' range at will and, therefore, manage daily operations by monitoring 'key performance indicators' (KPIs) — helping meet the quality of service (QoS) requirements much easily. To meet the QoS requirements 5G networks can be implemented alongside priority scheduling algorithms. This paper considers the operation of a wireless network slicing model under two scheduling algorithms. A comparative analysis of main performance measures is provided.

**Key words and phrases:** 5G networks, slicing, QoS, KPIs, priority scheduling, retrial queueing, iteration method

## 1. Introduction

The advent of new generation `5G` networks with their flagship `slicing` technology have highly influenced the telecommunications sector in the best way. Network operators have now the latitude to manage their assets and therefore, are able to propose new types of services to customers [1]–[3]. Businesses and enterprises can now access network connectivity that fits their specific needs [4]–[6]. `3GPP` defines `slicing` as a technology that offers on shared infrastructures the advantageous option to build fully dedicated logical networks, known as 'network slices', with very diverse quality of service (`QoS`) capabilities and requirements [7], [8]. Normally, meeting `QoS` requirements and extending capabilities are difficult tasks for network operators who can be helped by monitoring the 'key performance indicators' (`KPIs`) [9]–[12]. Essentially, monitoring the `KPIs` can allow network operators to significantly reduce service interruptions or even prevent them in the best cases [13], [14]. Since the first release of `slicing` technology few years ago, the vast majority of researchers, scientists and organizations in the telecommunications industry is focused on developing methods and techniques to flexibly and efficiently share available radio resources within its framework [15]–[19]. In modern wireless networks, one of the possible solutions to meet the `QoS` requirements is the implementation of `priority scheduling` algorithms [20]–[23]. Models implementing such algorithms within `slicing` framework could be described using the mathematical apparatus of `retrial queueing` theory [24]–[26], where retrial queues, also known as 'orbits', can be used to address service's interruptions problem.

In this paper we consider one of the possible models for implementing `slicing` with `priority scheduling` algorithms. More precisely, we provide a comparative analysis of model's performance measures under `preemptive` and `non-preemptive scheduling` algorithms. For that we use the mathematical apparatus of `queueing` theory and describe the model as a retrial queueing system coupled with a buffer [27]–[29].

The paper is organized as follows. Section 2 provides the system's general description and proposes a mathematical model for its construction. Section 3 suggests formulas to compute the stationary probability distributions under `preemptive` and `non-preemptive scheduling` algorithms respectively. Section 4 proposes formulas to calculate the main performance measures under each `priority scheduling` algorithm. Section 5 provides a numerical example of system's model operation. Section 6 concludes the paper.

## 2. Mathematical model

Let us consider a single server retrial queueing system [25] coupled with a buffer. We assume two types of requests arrival in system according to Poisson process with rates $\lambda_1$ and $\lambda_2$ respectively. The average service times are exponentially distributed with means $\mu_1$ and $\mu_2$.

Let us assume that *first* type requests have access to server and buffer, while *second* type requests — to server and orbit. Let us consider two types of `priority scheduling` algorithms — `preemptive` and `non-preemptive scheduling` [20], [21], [29], [30].

The `radio admission control` (`RAC`) mechanism for *first* type requests is organized differently depending on the `priority scheduling` algorithm.

`Preemptive scheduling`. The `RAC` mechanism for *first* type requests is organized in such a way that:

1) when server is "vacant" or "occupied" by one *second* type request, the *first* type request immediately obtains service, i.e. the *second* type request occupying server at such moments automatically joins the orbit;
2) otherwise, the *first* type request awaits server's non-utilization in buffer with first-come, first-served (`FCFS`) service discipline [24]–[26].

`Non-preemptive scheduling`. The `RAC` mechanism for *first* type requests is organized in such a way that:

1) when server is "vacant", the request immediately obtains service;

2) otherwise, the request awaits server's non-utilization in buffer with `FCFS` service discipline.

Whether `preemptive` or `non-preemptive scheduling` algorithm, awaiting in buffer *first* type requests are always given priority when it comes to service once server is "vacant".

The `RAC` mechanism for *second* type requests is organized in such a way that:

1) when server is "vacant", the request immediately obtains service;
2) otherwise, the request either *leaves* the system with probability $\pi$ or *joins* the orbit with probability $1 - \pi$.

A *second* type request that joined the orbit becomes a "retrial" *second* type request. A retrial *second* type request, as the name stipulates, *retries* to obtain service after some amount of time. The number of retrials is unlimited and time interval between two consecutive ones is exponentially distributed with rate $\sigma^{-1}$. Note that, as the "primary" *second* type request, the retrial *second* type request either *leaves* the system with probability $\pi$ or *returns* to the orbit with probability $1 - \pi$ after an *unsuccessful* attempt to occupy server.

The scheme model of considered single server retrial queueing system coupled with a buffer is given in figure 1.

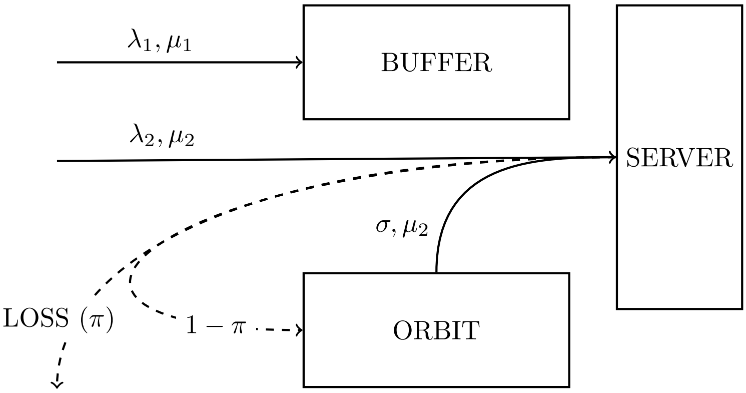

Figure 1. Scheme model of considered single server retrial queueing system coupled with an unlimited buffer

We describe system behavior using a three-dimensional vector $\mathbf{n} := (i, j, k)$ over "infinite" state spaces $\mathscr{X}$ and $\mathscr{Y}$ under `preemptive` and `non-preemptive scheduling` algorithms respectively:

$$\mathscr{X} = \left\{\mathbf{n} \in \mathbb{N}^3 : (i = 0 \wedge k \in \{0, 2\}) \vee k = 1\right\}, \tag{1a}$$

$$\mathscr{Y} = \left\{\mathbf{n} \in \mathbb{N}^3 : (i = 0 \wedge k = 0) \vee k \in \{1, 2\}\right\}, \tag{1b}$$

where $\mathbb{N}^3$ represents the state space of all three-dimensional vectors with natural elements; $i$ — the current number of *first* type requests in buffer; $j$ — the current number of *second* type requests in orbit; and $k$ — the current *state* of server (i.e., value "0" means server is "vacant"; value "1" — server is "occupied" by one *first* type request; and value "2" — server is "occupied" by one *second* type request).

The corresponding state transition diagrams are shown in figures 2, 3. The transition diagrams from random state are clarified in figures 4, 5.

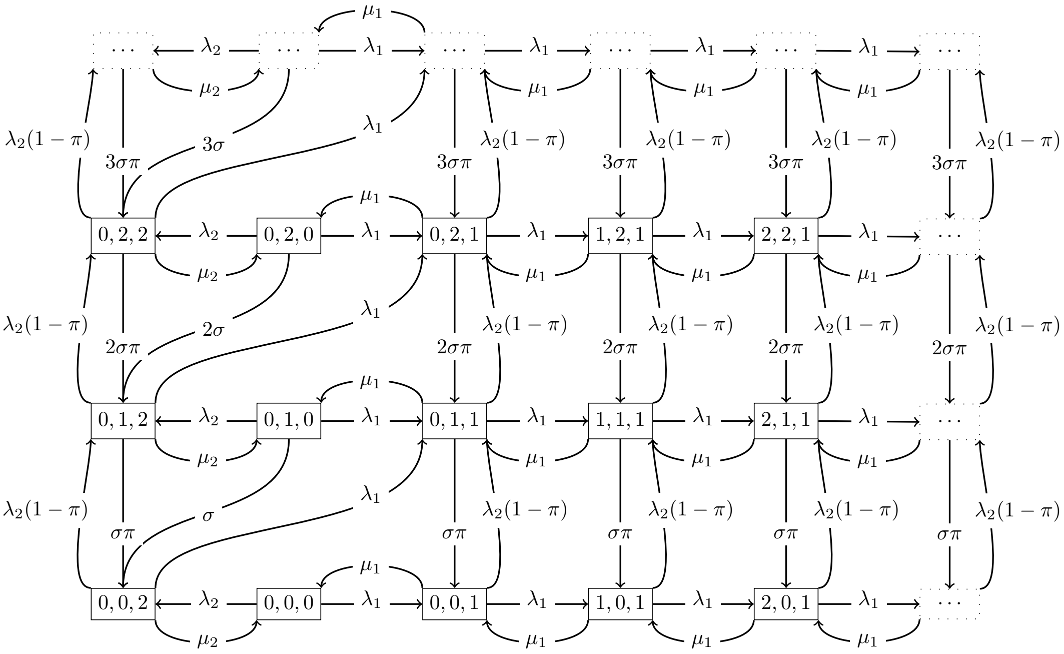

Figure 2. State transition diagram of considered single server retrial queueing system coupled with a buffer under `preemptive scheduling` algorithm

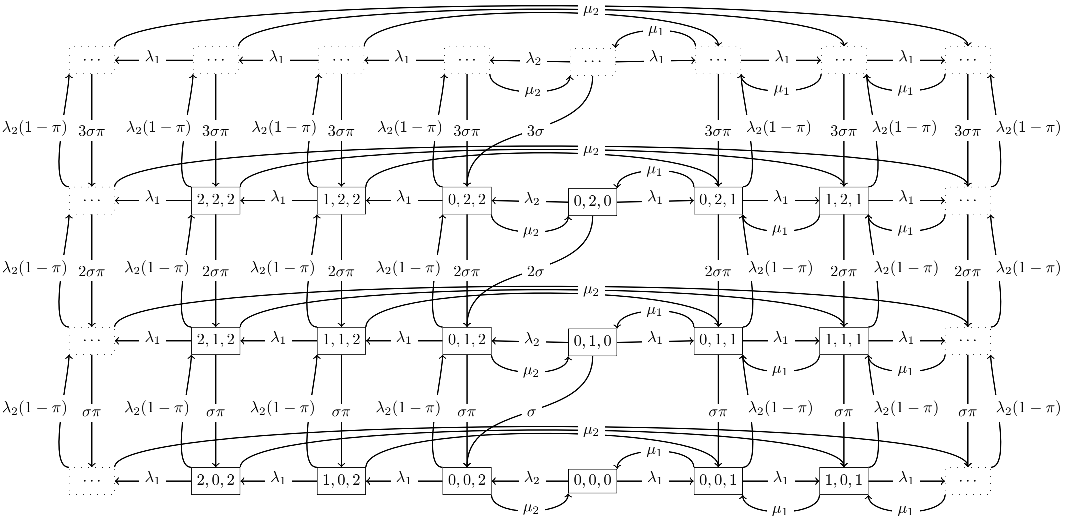

Figure 3. State transition diagram of considered single server retrial queueing system coupled with a buffer under `non-preemptive scheduling` algorithm

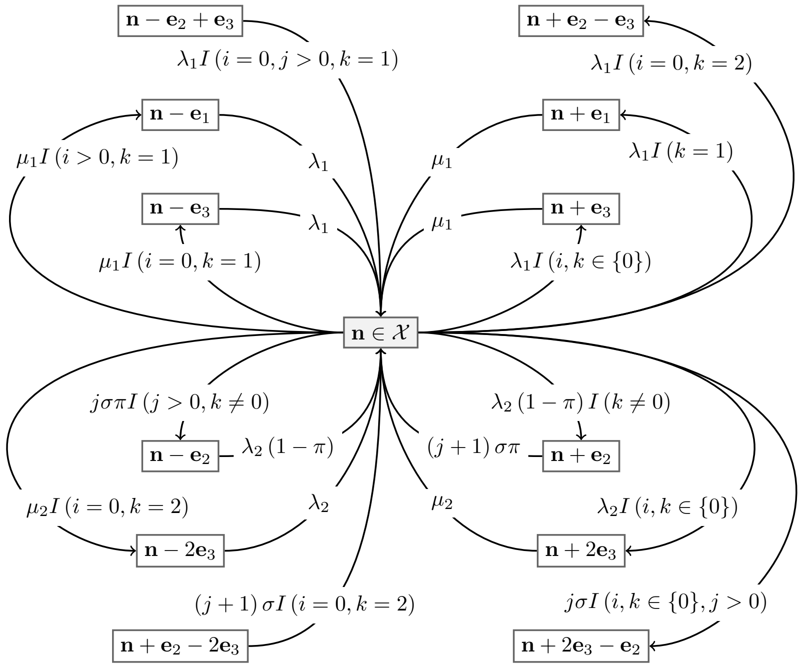
$\mathbf{n}-\mathbf{e}_2+\mathbf{e}_3$
$\lambda_1 I\,(i=0, j>0, k=1)$
$\mathbf{n}+\mathbf{e}_2-\mathbf{e}_3$
$\lambda_1 I\,(i=0, k=2)$
$\mathbf{n}-\mathbf{e}_1$
$\mu_1 I\,(i>0, k=1)$
$\lambda_1$
$\mu_1$
$\mathbf{n}+\mathbf{e}_1$
$\lambda_1 I\,(k=1)$
$\mathbf{n}-\mathbf{e}_3$
$\lambda_1$
$\mu_1$
$\mathbf{n}+\mathbf{e}_3$
$\mu_1 I\,(i=0, k=1)$
$\lambda_1 I\,(i, k\in\{0\})$
$\mathbf{n}\in\mathcal{X}$
$j\sigma\pi I\,(j>0, k\neq 0)$
$\lambda_2\,(1-\pi)\,I\,(k\neq 0)$
$\mathbf{n}-\mathbf{e}_2$
$\lambda_2\,(1-\pi)$
$(j+1)\,\sigma\pi$
$\mathbf{n}+\mathbf{e}_2$
$\mu_2 I\,(i=0, k=2)$
$\lambda_2$
$\mu_2$
$\lambda_2 I\,(i, k\in\{0\})$
$\mathbf{n}-2\mathbf{e}_3$
$\mathbf{n}+2\mathbf{e}_3$
$(j+1)\,\sigma I\,(i=0, k=2)$
$j\sigma I\,(i, k\in\{0\}, j>0)$
$\mathbf{n}+\mathbf{e}_2-2\mathbf{e}_3$
$\mathbf{n}+2\mathbf{e}_3-\mathbf{e}_2$

Figure 4. Transition diagram from random state for considered single server retrial queueing system coupled with a buffer under `preemptive scheduling` algorithm

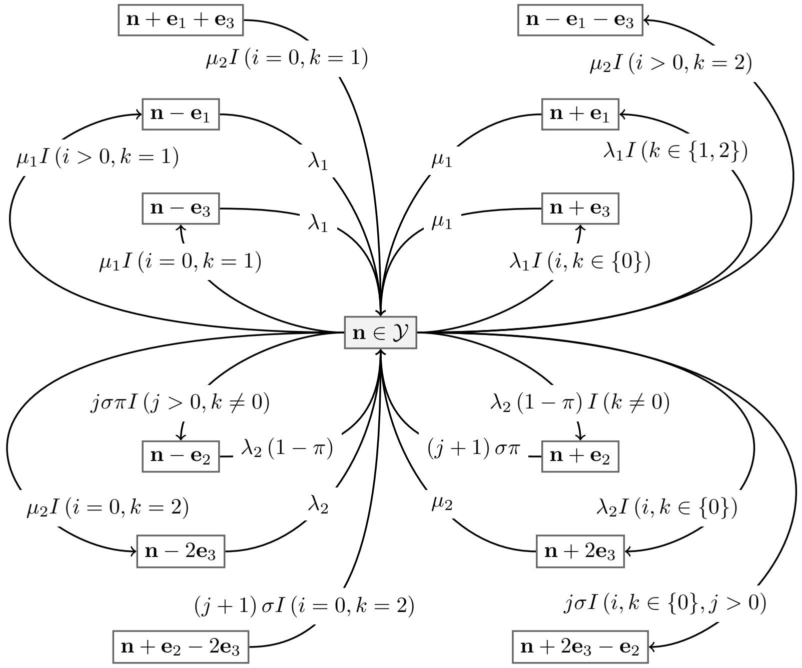
$\mathbf{n}+\mathbf{e}_1+\mathbf{e}_3$
$\mu_2 I\,(i=0, k=1)$
$\mathbf{n}-\mathbf{e}_1-\mathbf{e}_3$
$\mu_2 I\,(i>0, k=2)$
$\mathbf{n}-\mathbf{e}_1$
$\mu_1 I\,(i>0, k=1)$
$\lambda_1$
$\mu_1$
$\mathbf{n}+\mathbf{e}_1$
$\lambda_1 I\,(k\in\{1,2\})$
$\mathbf{n}-\mathbf{e}_3$
$\lambda_1$
$\mu_1$
$\mathbf{n}+\mathbf{e}_3$
$\mu_1 I\,(i=0, k=1)$
$\lambda_1 I\,(i, k\in\{0\})$
$\mathbf{n}\in\mathcal{Y}$
$j\sigma\pi I\,(j>0, k\neq 0)$
$\lambda_2\,(1-\pi)\,I\,(k\neq 0)$
$\mathbf{n}-\mathbf{e}_2$
$\lambda_2\,(1-\pi)$
$(j+1)\,\sigma\pi$
$\mathbf{n}+\mathbf{e}_2$
$\mu_2 I\,(i=0, k=2)$
$\lambda_2$
$\mu_2$
$\lambda_2 I\,(i, k\in\{0\})$
$\mathbf{n}-2\mathbf{e}_3$
$\mathbf{n}+2\mathbf{e}_3$
$(j+1)\,\sigma I\,(i=0, k=2)$
$j\sigma I\,(i, k\in\{0\}, j>0)$
$\mathbf{n}+\mathbf{e}_2-2\mathbf{e}_3$
$\mathbf{n}+2\mathbf{e}_3-\mathbf{e}_2$

Figure 5. Transition diagram from random state for considered single server retrial queueing system coupled with a buffer under `non-preemptive scheduling` algorithm

According to investigated `priority scheduling` algorithms and considering the transition diagrams from random state (i.e., figures 4, 5) one can obtain the equilibrium equations systems given below that describe the discussed Markov processes $X(t)$ and $Y(t)$, where $t > 0$:

$$\begin{gathered}
[\lambda_1 + \lambda_2 I\left(i,k \in \{0\}\right) + \lambda_2\left(1-\pi\right) I\left(k \neq 0\right) + \mu_k I\left(k \neq 0\right) + \\
+ j\sigma I\left(i,k \in \{0\}\right) + j\sigma\pi I\left(k \neq 0\right)]P(\mathbf{n}) = \lambda_1 I\left(i=0,k=1\right) P(\mathbf{n}-\mathbf{e}_3) + \\
+ \lambda_1 I\left(i>0,k=1\right) P(\mathbf{n}-\mathbf{e}_1) + \lambda_2 I\left(i=0,k=2\right) P(\mathbf{n}-2\mathbf{e}_3) + \\
+ \lambda_2\left(1-\pi\right) I\left(j>0,k\neq 0\right) P(\mathbf{n}-\mathbf{e}_2) + \mu_1 I\left(i,k \in \{0\}\right) P(\mathbf{n}+\mathbf{e}_3) + \\
+ \mu_1 I\left(k=1\right) P(\mathbf{n}+\mathbf{e}_1) + \mu_2 I\left(i,k \in \{0\}\right) P(\mathbf{n}+2\mathbf{e}_3) + \\
+ \left(j+1\right)\sigma I\left(i=0,k=2\right) P(\mathbf{n}+\mathbf{e}_2-2\mathbf{e}_3) + (j+1)\sigma\pi I\left(k\neq 0\right) P(\mathbf{n}+\mathbf{e}_2) + \\
+ \lambda_1 I\left(i=0,j>0,k=1\right) P(\mathbf{n}-\mathbf{e}_2+\mathbf{e}_3), \quad \text{(2a)}
\end{gathered}$$

$$\begin{gathered}
[\lambda_1 + \lambda_2 I\left(i,k \in \{0\}\right) + \lambda_2\left(1-\pi\right) I\left(k \neq 0\right) + \mu_k I\left(k \neq 0\right) + \\
+ j\sigma I\left(i,k \in \{0\}\right) + j\sigma\pi I\left(k \neq 0\right)]Q(\mathbf{n}) = \lambda_1 I\left(i=0,k=1\right) Q(\mathbf{n}-\mathbf{e}_3) + \\
+ \lambda_1 I\left(i>0,k=1\right) Q(\mathbf{n}-\mathbf{e}_1) + \lambda_2 I\left(i=0,k=2\right) Q(\mathbf{n}-2\mathbf{e}_3) + \\
+ \lambda_2\left(1-\pi\right) I\left(j>0,k\neq 0\right) Q(\mathbf{n}-\mathbf{e}_2) + \mu_1 I\left(i,k \in \{0\}\right) Q(\mathbf{n}+\mathbf{e}_3) + \\
+ \mu_1 I\left(k \in \{1,2\}\right) Q(\mathbf{n}+\mathbf{e}_1) + \mu_2 I\left(i,k \in \{0\}\right) Q(\mathbf{n}+2\mathbf{e}_3) + \\
+ \left(j+1\right)\sigma I\left(i=0,k=2\right) Q(\mathbf{n}+\mathbf{e}_2-2\mathbf{e}_3) + (j+1)\sigma\pi I\left(k\neq 0\right) Q(\mathbf{n}+\mathbf{e}_2) + \\
+ \mu_2 I\left(i=0,k=1\right) Q(\mathbf{n}+\mathbf{e}_1+\mathbf{e}_3), \quad \text{(2b)}
\end{gathered}$$

where $P(\mathbf{n})_{\mathbf{n}\in\mathcal{X}}$ and $Q(\mathbf{n})_{\mathbf{n}\in\mathcal{Y}}$ are the stationary probability distributions under `preemptive` and `non-preemptive scheduling` algorithms respectively; $\mathbf{e}_{s\in\{1,2,3\}}$ — the $s$-th row of identity matrix of size 3 × 3; and $I\left(\cdot\right)$ — the function indicator equaling value “1” when condition is met, and value “0” otherwise.

## 3. Stationary probability distribution

Due to the “infinite” sizes of buffer and orbit, the stationary probability distributions $\mathbf{P} = \left(P(\mathbf{n})\right)_{\mathbf{n}\in\mathcal{X}}$ and $\mathbf{Q} = \left(Q(\mathbf{n})\right)_{\mathbf{n}\in\mathcal{Y}}$ should be computed through `generating function`-based approaches [25], [27], [29]. However, one can compute them using `iteration` methods [31], [32] by simply adding limitations to the storage sizes, setting these to random maximum values. Thus, we set buffer’s maximum size to $i_{\max}$ and orbit’s to $j_{\max}$. Therefore, we obtain the “finite” state spaces $\widetilde{\mathcal{X}}$ and $\tilde{\mathcal{Y}}$ under `preemptive` and `non-preemptive scheduling` algorithms respectively:

$$\widetilde{\mathcal{X}} = \left\{\mathbf{n}\in\mathcal{X} : i \leqslant i_{\max} \wedge j \leqslant j_{\max}\right\}, \quad \tilde{\mathcal{Y}} = \left\{\mathbf{n}\in\mathcal{Y} : i \leqslant i_{\max} \wedge j \leqslant j_{\max}\right\}.$$

The process describing considered system is not a reversible Markov process whether under `preemptive` or `non-preemptive scheduling` algorithm.

Therefore, one can compute either stationary probability distribution $\mathbf{P}$ or $\mathbf{Q}$ using `iteration` method on respective equilibrium's equations system, i.e.

$$\mathbf{P}\cdot\mathbf{A}_{[|\widetilde{\mathscr{X}}|\times|\widetilde{\mathscr{X}}|]}=\mathbf{0}_{[1\times|\widetilde{\mathscr{X}}|]},\quad \mathbf{Q}\cdot\mathbf{B}_{[|\widetilde{\mathscr{Y}}|\times|\widetilde{\mathscr{Y}}|]}=\mathbf{0}_{[1\times|\widetilde{\mathscr{Y}}|]},$$

where $\mathbf{A}$ and $\mathbf{B}$ are the infinitesimal generators of Markov process under `preemptive` and `non-preemptive scheduling` algorithms respectively.

The elements $A_{\mathbf{n},\hat{\mathbf{n}}}$ of the infinitesimal generator $\mathbf{A}$ are computed using (3a). Equation (3b) calculates the elements $B_{\mathbf{n},\hat{\mathbf{n}}}$ of the infinitesimal generator $\mathbf{B}$.

$$A_{\mathbf{n},\hat{\mathbf{n}}}=\begin{cases}\lambda_1,\ \text{if}\ \ \hat{\mathbf{n}}=\mathbf{n}+\mathbf{e}_3,\ \text{s.t.}\ i,k\in\{0\},\\ \ \text{or}\ \ \hat{\mathbf{n}}=\mathbf{n}+\mathbf{e}_1,\ \text{s.t.}\ i<i_{\max}\wedge k=1,\\ \ \text{or}\ \ \hat{\mathbf{n}}=\mathbf{n}+\mathbf{e}_2-\mathbf{e}_3,\ \text{s.t.}\ i=0\wedge j<j_{\max}\wedge k=2,\\ \lambda_2,\ \text{if}\ \ \hat{\mathbf{n}}=\mathbf{n}+2\mathbf{e}_3,\ \text{s.t.}\ i,k\in\{0\},\\ \lambda_2\left(1-\pi\right),\ \text{if}\ \ \hat{\mathbf{n}}=\mathbf{n}+\mathbf{e}_2,\ \text{s.t.}\ j<j_{\max}\wedge k\in\{1,2\},\\ \mu_1,\ \text{if}\ \ \hat{\mathbf{n}}=\mathbf{n}-\mathbf{e}_3,\ \text{s.t.}\ i=0\wedge k=1,\\ \ \text{or}\ \ \hat{\mathbf{n}}=\mathbf{n}-\mathbf{e}_1,\ \text{s.t.}\ i>0\wedge k=1,\\ \mu_2,\ \text{if}\ \ \hat{\mathbf{n}}=\mathbf{n}-2\mathbf{e}_3,\ \text{s.t.}\ i=0\wedge k=2,\\ j\sigma,\ \text{if}\ \ \hat{\mathbf{n}}=\mathbf{n}+2\mathbf{e}_3-\mathbf{e}_2,\ \text{s.t.}\ j>0\wedge i,k\in\{0\},\\ j\sigma\pi,\ \text{if}\ \ \hat{\mathbf{n}}=\mathbf{n}-\mathbf{e}_2,\ \text{s.t.}\ j>0\wedge k\in\{1,2\},\\ 0,\ \text{otherwise},\end{cases}\tag{3a}$$

with $\mathbf{n}\in\widetilde{\mathscr{X}}$, and $A_{\mathbf{n},\mathbf{n}}=-\sum\limits_{\hat{\mathbf{n}}\in\widetilde{\mathscr{X}}\{\mathbf{n}\}}A_{\mathbf{n},\hat{\mathbf{n}}}$.

$$B_{\mathbf{n},\hat{\mathbf{n}}}=\begin{cases}\lambda_1,\ \text{if}\ \ \hat{\mathbf{n}}=\mathbf{n}+\mathbf{e}_3,\ \text{s.t.}\ i,k\in\{0\},\\ \ \text{or}\ \ \hat{\mathbf{n}}=\mathbf{n}+\mathbf{e}_1,\ \text{s.t.}\ i<i_{\max}\wedge k\in\{1,2\},\\ \lambda_2,\ \text{if}\ \ \hat{\mathbf{n}}=\mathbf{n}+2\mathbf{e}_3,\ \text{s.t.}\ i,k\in\{0\},\\ \lambda_2\left(1-\pi\right),\ \text{if}\ \ \hat{\mathbf{n}}=\mathbf{n}+\mathbf{e}_2,\ \text{s.t.}\ j<j_{\max}\wedge k\in\{1,2\},\\ \mu_1,\ \text{if}\ \ \hat{\mathbf{n}}=\mathbf{n}-\mathbf{e}_3,\ \text{s.t.}\ i=0\wedge k=1,\\ \ \text{or}\ \ \hat{\mathbf{n}}=\mathbf{n}-\mathbf{e}_1,\ \text{s.t.}\ i>0\wedge k=1,\\ \mu_2,\ \text{if}\ \ \hat{\mathbf{n}}=\mathbf{n}-2\mathbf{e}_3,\ \text{s.t.}\ i=0\wedge k=2,\\ \ \text{or}\ \ \hat{\mathbf{n}}=\mathbf{n}-\mathbf{e}_1-\mathbf{e}_3,\ \text{s.t.}\ i>0\wedge k=2,\\ j\sigma,\ \text{if}\ \ \hat{\mathbf{n}}=\mathbf{n}+2\mathbf{e}_3-\mathbf{e}_2,\ \text{s.t.}\ j>0\wedge i,k\in\{0\},\\ j\sigma\pi,\ \text{if}\ \ \hat{\mathbf{n}}=\mathbf{n}-\mathbf{e}_2,\ \text{s.t.}\ j>0\wedge k\in\{1,2\},\\ 0,\ \text{otherwise},\end{cases}\tag{3b}$$

with $\mathbf{n}\in\widetilde{\mathscr{Y}}$, and $B_{\mathbf{n},\mathbf{n}}=-\sum\limits_{\hat{\mathbf{n}}\in\widetilde{\mathscr{Y}}\{\mathbf{n}\}}B_{\mathbf{n},\hat{\mathbf{n}}}$.

## 4. Performance measures

After computing the stationary probability distributions **P** and **Q** one can calculate system's performance measures under `preemptive` and `non-preemptive scheduling` algorithms respectively. Let us consider following main performance measures:

1. The mean number of *first* type requests in buffer

$$\sum_{\mathbf{n}\in\widetilde{\mathscr{X}}} i\cdot P(\mathbf{n})\,,\quad \sum_{\mathbf{n}\in\widetilde{\mathscr{Y}}} i\cdot Q\left(\mathbf{n}\right),\tag{4}$$

2. The mean number of *second* type requests in orbit

$$\sum_{\mathbf{n}\in\widetilde{\mathscr{X}}} j\cdot P(\mathbf{n})\,,\quad \sum_{\mathbf{n}\in\widetilde{\mathscr{Y}}} j\cdot Q\left(\mathbf{n}\right),\tag{5}$$

3. The server's vacancy probability

$$\sum_{\mathbf{n}\in\widetilde{\mathscr{X}}:k=0} P(\mathbf{n})\,,\quad \sum_{\mathbf{n}\in\widetilde{\mathscr{Y}}:k=0} Q\left(\mathbf{n}\right),\tag{6}$$

4. The server's occupancy probability by one *first* type request

$$\sum_{\mathbf{n}\in\widetilde{\mathscr{X}}:k=1} P(\mathbf{n})\,,\quad \sum_{\mathbf{n}\in\widetilde{\mathscr{Y}}:k=1} Q\left(\mathbf{n}\right),\tag{7}$$

5. The server's occupancy probability by one *second* type request

$$\sum_{\mathbf{n}\in\widetilde{\mathscr{X}}:k=2} P(\mathbf{n})\,,\quad \sum_{\mathbf{n}\in\widetilde{\mathscr{Y}}:k=2} Q\left(\mathbf{n}\right).\tag{8}$$

Since limitations were applied to storage sizes, i.e. buffer and orbit, one may find it necessary to also compute following performance measures:

1. The buffer's saturation probability

$$\sum_{\mathbf{n}\in\widetilde{\mathscr{X}}:i=i_{\max}} P(\mathbf{n})\,,\quad \sum_{\mathbf{n}\in\widetilde{\mathscr{Y}}:i=i_{\max}} Q\left(\mathbf{n}\right),\tag{9}$$

2. The orbit's saturation probability

$$\sum_{\mathbf{n}\in\widetilde{\mathscr{X}}:j=j_{\max}} P(\mathbf{n})\,,\quad \sum_{\mathbf{n}\in\widetilde{\mathscr{Y}}:j=j_{\max}} Q\left(\mathbf{n}\right).\tag{10}$$

## 5. Numerical example

Let us illustrate the behavior of performance measures, computed in previous section 4, depending on various system's parameters. To implement `iteration` method one must set the `error tolerance` $\varepsilon$ and, for ergonomic features, limit the `number of iterations` $MaxIters$. Since *second* type requests are apparently more affected by implemented `priority scheduling` algorithms, one may build the example around performance measures "directly" related to them:

— the mean number of *second* type requests in orbit, i.e. equations (5);
— the server's vacancy probability, i.e. equations (6);
— the server's occupancy probability by one *second* type request, i.e. equations (8);
— the orbit's saturation probability, i.e. equations (10).

Summaries of the numerical examples results are provided in tables 1 to 4.

Table 1

Mean number of *second* type requests in orbit depending on triplet $(j_{\max}, \lambda_1, \lambda_2)$ with $i_{\max} = 10$, $\mu_1 = \mu_2 = 2$, $\pi = 0.001$, $\sigma = 1$, $\varepsilon = 10^{-12}$ and $MaxIters = 1000$

| - | - | `Preemptive scheduling` | | | `Non-preemptive scheduling` | | |
|---|---|---|---|---|---|---|---|
| $j_{\max}$ | $\lambda_2$ <br> $\lambda_1$ | 1 | 2 | 3 | 1 | 2 | 3 |
| 5 | 1 | 2.5438 | 3.3625 | 3.8375 | 2.4659 | 3.4162 | 3.9503 |
| | 2 | 3.9805 | 4.1998 | 4.3897 | 4.0846 | 4.3046 | 4.4961 |
| | 3 | 4.5566 | 4.6908 | 4.7835 | 4.6314 | 4.7437 | 4.8230 |
| 10 | 1 | 4.9052 | 7.2944 | 8.3800 | 4.7192 | 7.3611 | 8.5276 |
| | 2 | 8.5649 | 8.9173 | 9.2121 | 8.7040 | 9.0528 | 9.3429 |
| | 3 | 9.4234 | 9.5984 | 9.7193 | 9.5149 | 9.6616 | 9.7651 |
| 15 | 1 | 6.9305 | 11.4591 | 13.1148 | 6.6439 | 11.5360 | 13.2783 |
| | 2 | 13.3191 | 13.7555 | 14.1114 | 13.4738 | 13.9025 | 14.2497 |
| | 3 | 14.3427 | 14.5381 | 14.6738 | 14.4387 | 14.6034 | 14.7205 |

Table 1 shows that when the arrival rate $\lambda_1$ of *first* type requests or $\lambda_2$ of *second* type requests increases, the mean number of *second* type requests in orbit also increases. That performance measure is greater under `non-preemptive scheduling` algorithm. This may be explained by the fact that, we have more *second* type requests in system, and consequently, the orbit tends to saturation. This situation is also illustrated by table 2 showing the increase of orbit's saturation probability under the same circumstances.

Table 3 shows that when the arrival rate $\lambda_1$ of *first* type requests or $\lambda_2$ of *second* type requests increases, the server's vacancy probability decreases. As one can see from that table, and according to table 1, that performance measure is less under `non-preemptive scheduling` algorithm. This may be explained by the fact that the more requests we have in system, the less server will be "vacant".

Table 4 shows that when fixing arrival rate $\lambda_1$ of *first* type requests to value "1" and increasing arrival rate $\lambda_2$ of *second* type requests, the server's occupancy probability increases.

Table 2

Saturation probability of orbit depending on triplet $(j_{\max}, \lambda_1, \lambda_2)$ with $i_{\max} = 10$, $\mu_1 = \mu_2 = 2$, $\pi = 0.001$, $\sigma = 1$, $\varepsilon = 10^{-12}$ and $MaxIters = 1000$

| - | - | `Preemptive scheduling` | | | `Non-preemptive scheduling` | | |
|---|---|---|---|---|---|---|---|
| $j_{\max}$ | $\lambda_2$ \ $\lambda_1$ | 1 | 2 | 3 | 1 | 2 | 3 |
| 5 | 1 | 0.2229 | 0.3647 | 0.4686 | 0.2256 | 0.3990 | 0.5262 |
| | 2 | 0.5322 | 0.6105 | 0.6796 | 0.5862 | 0.6654 | 0.7372 |
| | 3 | 0.7602 | 0.8236 | 0.8687 | 0.8015 | 0.8537 | 0.8919 |
| 10 | 1 | 0.1197 | 0.2889 | 0.4174 | 0.1223 | 0.3216 | 0.4750 |
| | 2 | 0.4801 | 0.5653 | 0.6425 | 0.5346 | 0.6210 | 0.7007 |
| | 3 | 0.7247 | 0.7913 | 0.8401 | 0.7664 | 0.8215 | 0.8633 |
| 15 | 1 | 0.0690 | 0.2473 | 0.3889 | 0.0702 | 0.2766 | 0.4436 |
| | 2 | 0.4490 | 0.5363 | 0.6171 | 0.5007 | 0.5893 | 0.6729 |
| | 3 | 0.6978 | 0.7649 | 0.8154 | 0.7375 | 0.7936 | 0.8374 |

Table 3

Vacancy probability of server depending on triplet $(j_{\max}, \lambda_1, \lambda_2)$ with $i_{\max} = 10$, $\mu_1 = \mu_2 = 2$, $\pi = 0.001$, $\sigma = 1$, $\varepsilon = 10^{-12}$ and $MaxIters = 1000$

| - | - | `Preemptive scheduling` | | | `Non-preemptive scheduling` | | |
|---|---|---|---|---|---|---|---|
| $j_{\max}$ | $\lambda_2$ \ $\lambda_1$ | 1 | 2 | 3 | 1 | 2 | 3 |
| 5 | 1 | 0.1394 | 0.0803 | 0.0556 | 0.1242 | 0.0630 | 0.0395 |
| | 2 | 0.0465 | 0.0361 | 0.0285 | 0.0310 | 0.0223 | 0.0162 |
| | 3 | 0.0198 | 0.0137 | 0.0098 | 0.0107 | 0.0071 | 0.0048 |
| 10 | 1 | 0.0923 | 0.0351 | 0.0220 | 0.0847 | 0.0269 | 0.0148 |
| | 2 | 0.0183 | 0.0143 | 0.0118 | 0.0114 | 0.0083 | 0.0063 |
| | 3 | 0.0082 | 0.0058 | 0.0043 | 0.0042 | 0.0028 | 0.0020 |
| 15 | 1 | 0.0735 | 0.0197 | 0.0128 | 0.0685 | 0.0146 | 0.0084 |
| | 2 | 0.0106 | 0.0086 | 0.0073 | 0.0064 | 0.0048 | 0.0039 |
| | 3 | 0.0051 | 0.0037 | 0.0027 | 0.0026 | 0.0018 | 0.0013 |

But, when fixing $\lambda_1$ to values “2” or “3” that probability decreases. That performance measure is less under `non-preemptive scheduling` algorithm. This may be explained by the fact that the more *first* type requests we have in system, the less server will be occupied by one *second* type request, since `RAC` mechanism suggests that priority is always given to *first* type requests once server is “vacant”. Furthermore, when fixing $\lambda_2$ and increasing $\lambda_1$ the server’s occupancy probability decreases generally except under `preemptive scheduling` algorithm for one case, where orbit’s maximum size $j_{\max}$ equals value “5” and $\lambda_2$ equals value “1”. In that case, that probability increases to a maximum value and then decreases.

Table 4

Occupancy probability of server by one *second* type request depending on triplet $(j_{\max}, \lambda_1, \lambda_2)$ with $i_{\max} = 10$, $\mu_1 = \mu_2 = 2$, $\pi = 0.001$, $\sigma = 1$, $\varepsilon = 10^{-12}$ and $MaxIters = 1000$

| - | - | `Preemptive scheduling` | | | `Non-preemptive scheduling` | | |
|---|---|---|---|---|---|---|---|
| $j_{\max}$ | $\lambda_1$ \ $\lambda_2$ | 1 | 2 | 3 | 1 | 2 | 3 |
| 5 | 1 | 0.3779 | 0.4655 | 0.5232 | 0.3760 | 0.4372 | 0.4608 |
| | 2 | 0.3973 | 0.3372 | 0.3070 | 0.3461 | 0.2836 | 0.2446 |
| | 3 | 0.2056 | 0.1463 | 0.1112 | 0.1639 | 0.1155 | 0.0857 |
| 10 | 1 | 0.4163 | 0.4992 | 0.5457 | 0.4156 | 0.4734 | 0.4854 |
| | 2 | 0.4157 | 0.3503 | 0.3156 | 0.3657 | 0.2977 | 0.2546 |
| | 3 | 0.2113 | 0.1498 | 0.1132 | 0.1706 | 0.1199 | 0.0887 |
| 15 | 1 | 0.4312 | 0.5087 | 0.5487 | 0.4317 | 0.4856 | 0.4919 |
| | 2 | 0.4178 | 0.3509 | 0.3150 | 0.3707 | 0.3012 | 0.2571 |
| | 3 | 0.2107 | 0.1491 | 0.1124 | 0.1723 | 0.1211 | 0.0895 |

## 6. Conclusion

One considered a possible model for implementing `slicing` technology with `priority scheduling` algorithms. A comparative analysis of computed main performance measures — mean number of *first* type requests in buffer, mean number of *second* type requests in orbit, server’s vacancy probability, server’s occupancy probability by one *first* type request, server’s occupancy probability by one *second* type request, buffer’s saturation probability and orbit’s saturation probability — was provided. That analysis showed that system load is higher under `non-preemptive scheduling` algorithm with very low probability of leaving system after an *unsuccessful* attempt to occupy server.

## Acknowledgments

This paper has been supported by the RUDN University Strategic Academic Leadership Program (recipient Adou Y.). The reported study was funded by RFBR, project number 20-07-01052 (recipient Markova E. V.)

## References

[1] W. Lehr, F. Queder, and J. Haucap, "5G: A new future for Mobile Network Operators, or not?" *Telecommunications Policy*, vol. 45, no. 3, p. 102 086, Jan. 2021. DOI: 10.1016/j.telpol.2020.102086.

[2] Z. Ofir. "What will be the impact of 5g on network operators?" (Nov. 2021), [Online]. Available: https://www.forbes.com/sites/forbestechcouncil/2020/11/24/what-will-be-the-impact-of-5g-on-network-operators/.

[3] "Cloud computing services drive companies' digital transformation." (Jan. 2022), [Online]. Available: https://www.telefonica.com/en/communication-room/blog/cloud-computing-services-drive-companies-digital-transformation/.

[4] K. Budka. "AI + Augmented: Pushing the Limits of What Machines Can Do." (Sep. 2021), [Online]. Available: https://www.industryweek.com/technology-and-iiot/emerging-technologies/article/21174112/ai-augmented-pushes-the-limits-of-what-machines-can-do.

[5] C. Johnson. "What Companies And Governments Really Want From Industry 4.0." (Nov. 2021), [Online]. Available: https://www.forbes.com/sites/nokia-industry-40/2021/11/18/what-companies-and-governments-really-want-from-industry-40/.

[6] Nokia. "How Supply Chain 4.0 delivers the goods other approaches can't." (Dec. 2021), [Online]. Available: http://www.ft.com/partnercontent/nokia/how-supply-chain-4-0-delivers-the-goods-other-approaches-cant.html.

[7] A. Sultan and M. Pope, "Feasibility study on new services and markets technology enablers for network operation; Stage 1," 3rd Generation Partnership Project (3GPP), Technical report (TR) 22.864, Sep. 2016, Version 15.0.0.

[8] J. M. Meredith, F. Firmin, and M. Pope, "Release 16 Description; Summary of Rel-16 Work Items," 3rd Generation Partnership Project (3GPP), Technical report (TR) 21.916, Jan. 2022, Version 16.1.0.

[9] J. M. Meredith, M. C. Soveri, and M. Pope, "Management and orchestration; 5G end to end Key Performance Indicators (KPI)," 3rd Generation Partnership Project (3GPP), Technical specification (TS) 28.554, Dec. 2021, Version 17.5.0.

[10] "5G industry campus network deployment guideline," GSM Association (GSMA), Official Document NG.123, Oct. 2021, Version 2.0.

[11] I. Markopoulos *et al.*, "Service performance measurement methods over 5G experimental networks," 5G PPP, white paper ICT-19, Mar. 2021, Version 1.0. DOI: 10.5281/zenodo.4748482.

[12] L. Nielsen *et al.*, "Basic Testing Guide — A Starter Kit for Basic 5G KPIs Verification," 5G PPP, white paper, Nov. 2021, Version 1.0. DOI: 10.5281/zenodo.5704519.

[13] O. Ohlsson, P. Wallentin, and C.-G. Persson. "Reducing mobility interruption time in 5G networks." (Apr. 2020), [Online]. Available: https://www.ericsson.com/en/blog/2020/4/reducing-mobility-interruption-time-5g-networks.

[14] G. Sevilla. "2022 predictions: Internet and network outages will continue to get worse before they get better." (Dec. 2021), [Online]. Available: https://www.emarketer.com/content/2022-predictions-internet-network-outages-will-continue-worse-before-they-better.

[15] H. Zhu, G. Zhang, D. Hong, S. Zhang, and S. Huang, "Data Access Control Method of Power Terminal Based on 5G Technology," in *Advanced Hybrid Information Processing*, S. Liu and X. Ma, Eds., Cham: Springer International Publishing, 2022, pp. 26–39. DOI: 10.1007/978-3-030-94554-1_3.

[16] D. Alotaibi, V. Thayananthan, and J. Yazdani, "The 5G network slicing using SDN based technology for managing network traffic," *Procedia Computer Science*, vol. 194, pp. 114–121, Dec. 2021, 18th International Learning & Technology Conference 2021. DOI: 10.1016/j.procs.2021.10.064.

[17] R. Moreira, P. F. Rosa, R. L. A. Aguiar, and F. de Oliveira Silva, "NASOR: A network slicing approach for multiple Autonomous Systems," *Computer Communications*, vol. 179, pp. 131–144, Jul. 2021. DOI: 10.1016/j.comcom.2021.07.028.

[18] P. Zhu, J. Zhang, Y. Xiao, J. Cui, L. Bai, and Y. Ji, "Deep reinforcement learning-based radio function deployment for secure and resource-efficient NG-RAN slicing," *Engineering Applications of Artificial Intelligence*, vol. 106, p. 104 490, 2021. DOI: 10.1016/j.engappai.2021.104490.

[19] H. Yang, T. So, and Y. Xu, "Chapter 12 — 5G network slicing," in *5G NR and Enhancements*, J. Shen, Z. Du, Z. Zhang, N. Yang, and H. Tang, Eds., Elsevier, 2022, pp. 621–639. DOI: 10.1016/B978-0-323-91060-6.00012-X.

[20] N. Suganthi and S. Meenakshi, "An efficient scheduling algorithm using queuing system to minimize starvation of non-real-time secondary users in Cognitive Radio Network," *Cluster Computing*, vol. 25, no. 1, pp. 1–11, Jan. 2022. DOI: 10.1007/s10586-017-1595-8.

[21] A. Fathalla, K. Li, and A. Salah, "Best-KFF: a multi-objective preemptive resource allocation policy for cloud computing systems," *Cluster Computing*, vol. 25, no. 1, pp. 321–336, Feb. 2022. DOI: 10.1007/s10586-021-03407-z.

[22] S. J. Ahmad *et al.*, "A Dynamic Priority Based Scheduling Scheme for Multimedia Streaming Over MANETs to Improve QoS," in *Distributed Computing and Internet Technology*, Cham: Springer International Publishing, 2016, pp. 122–126. DOI: 10.1007/978-3-319-28034-9_15.

[23] A. Belgacem and K. Beghdad-Bey, "Multi-objective workflow scheduling in cloud computing: trade-off between makespan and cost," *Cluster Computing*, vol. 25, no. 1, pp. 579–595, Feb. 2022. DOI: `10.1007/s10586-021-03432-y`.

[24] J. R. Artalejo and A. Gómez-Corral, *Retrial Queueing Systems*. Springer, Berlin, Heidelberg, Jan. 2008. DOI: `10.1007/978-3-540-78725-9`.

[25] P. P. Bocharov, C. D'Apice, and A. V. Pechinkin, *Queueing Theory*. De Gruyter, 2011. DOI: `10.1515/9783110936025`.

[26] G. P. Basharin, Y. V. Gaidamaka, and K. E. Samouylov, "Mathematical Theory of Teletraffic and its Application to the Analysis of Multiservice Communication of Next Generation Networks," *Automatic Control and Computer Sciences*, vol. 47, no. 2, pp. 62–69, 2013. DOI: `10.3103/S0146411613020028`.

[27] K. Y. Adou and E. V. Markova, "Methods for Analyzing Slicing Technology in 5G Wireless Network Described as Queueing System with Unlimited Buffer and Retrial Group," in *Information Technologies and Mathematical Modelling. Queueing Theory and Applications*, Cham: Springer International Publishing, Mar. 2021, pp. 264–278. DOI: `10.1007/978-3-030-72247-0_20`.

[28] E. Markova, Y. Adou, D. Ivanova, A. Golskaia, and K. Samouylov, "Queue with Retrial Group for Modeling Best Effort Traffic with Minimum Bit Rate Guarantee Transmission Under Network Slicing," in *Distributed Computer and Communication Networks*, Cham: Springer International Publishing, 2019, pp. 432–442. DOI: `978-3-030-36614-8_33`.

[29] M. Korenevskaya, O. Zayats, A. Ilyashenko, and V. Muliukha, "Retrial Queuing System with Randomized Push-Out Mechanism and Non-Preemptive Priority," *Procedia Computer Science*, vol. 150, pp. 716–725, 2019, Proceedings of the 13th International Symposium "Intelligent Systems 2018" (INTELS'18), 22-24 October, 2018, St. Petersburg, Russia. DOI: `10.1016/j.procs.2019.02.016`.

[30] A. M. Yadav, K. N. Tripathi, and S. C. Sharma, "An enhanced multi-objective fireworks algorithm for task scheduling in fog computing environment," *Cluster Computing*, Nov. 2021. DOI: `10.1007/s10586-021-03481-3`.

[31] S. N. Stepanov, *Fundamentals of Multiservice Networks [Osnovy teletraffika multiservisnykh setei]*. Moscow: Eqo-Trends, 2010, p. 392, in Russian.

[32] S. N. Stepanov, *Theory of Teletraffic: Concepts, Models, Applications [Teoriya teletraffika: kontseptsii, modeli, prilozheniya]*. Moscow: Goryachaya Liniya-Telekom, 2015, p. 868, in Russian.

**For citation:**

K. Y. Adou, E. V. Markova, E. A. Zhbankova, Performance analysis of queueing system model under priority scheduling algorithms within 5G networks slicing framework, Discrete and Continuous Models and Applied Computational Science 30 (1) (2022) 5–20. DOI: 10.22363/2658-4670-2022-30-1-5-20.

**Information about the authors**:

**Adou, Kpangny Yves Berenger** — PhD Student at the Department of Applied Probability and Informatics, Faculty of Science, Peoples' Friendship University of Russia (RUDN University) (e-mail: 1042205051@rudn.ru, ORCID: https://orcid.org/0000–0003–4669–0898)

**Markova, Ekaterina Viktorovna** — Candidate of Physical and Mathematical Sciences, Associate Professor at the Department of Applied Probability and Informatics, Faculty of Science, Peoples' Friendship University of Russia (RUDN University) (e-mail: markova-ev@rudn.ru, ORCID: https://orcid.org/0000–0002–7876–2801)

**Zhbankova, Elena Aleksandrovna** — MSc student at the Department of Applied Probability and Informatics, Faculty of Science, Peoples' Friendship University of Russia (RUDN University) (e-mail: 1032202159@rudn.ru, ORCID: https://orcid.org/0000-0003-2482-4488)

УДК 519.872:519.217
PACS 07.05.Tp, 02.60.Pn, 02.70.Bf
DOI: 10.22363/2658-4670-2022-30-1-5-20

# К анализу системы массового обслуживания для сети 5G с технологией NS и приоритетным управлением доступом к радиоресурсам

**К. И. Б. Аду, Е. В. Маркова, Е. А. Жбанкова**

*Российский университет дружбы народов*
*ул. Миклухо-Маклая, д. 6, Москва, Россия, 117198*

**Аннотация.** Переход к беспроводным сетям пятого поколения 5G ознаменовал новый этап развития информационных и коммуникационных технологий. Сети пятого поколения должны решить такие проблемы, как негибкость «традиционных» сетей и нехватка частотных радиоресурсов для качественного предоставления услуг. Предполагается, что, используя эти сети, мобильные операторы смогут значительно расширить спектр услуг и обеспечить требуемое качество их предоставления. Для удовлетворения требований к качеству обслуживания (*англ.* Quality of Service — QoS) операторам необходимо выполнение «ключевых показателей эффективности» (*англ.* Key Performance Indicators — KPI), описанных в стандартах связи. Для этой цели могут быть использованы алгоритмы приоритетного облуживания. В статье рассмотрена модель беспроводной сети 5G, поддерживающая технологию нарезки сети и реализующая управление доступом к сетевым радиоресурсам при помощи введения приоритетов. Изучена работа модели в рамках двух алгоритмов. Проведён сравнительный анализ основных показателей эффективности модели.

**Ключевые слова:** сети 5G, нарезка сети NS, QoS, KPI, приоритетное управление доступом, СМО с повторными заявками, итерационный метод