AIMS Mathematics, 11(9): 30403–30436.
DOI: 10.3934/math.20261205
Received: 05 July 2026
Revised: 06 September 2026
Accepted: 09 September 2026
Published: 18 September 2026

http://www.aimspress.com/journal/Math

Research article

Steady-state analysis of an ordered hunting scheduled multiserver queueing system with Markovian arrival process, infinite buffer, heterogeneous servers, and phase-type distribution of service

Ciro D'Apice[1], Alexander N. Dudin[2], Sergey A. Dudin[2] and Rosanna Manzo[3,*]

[1] Dipartimento di Scienze Aziendali–Management & Information Systems, University of Salerno, Via Giovanni Paolo II, 132, Fisciano (SA), 84084, Italy

[2] Department of Applied Mathematics and Computer Science, Belarusian State University, Minsk, Belarus

[3] Dipartimento di Scienze Politiche e della Comunicazione, University of Salerno, Via Giovanni Paolo II, 132, Fisciano (SA), 84084, Italy

* **Correspondence:** Email: rmanzo@unisa.it.

Abstract: This paper analyzes a infinite-capacity queueing system operating with multiple distinct servers. Phase-type distributions with different irreducible representations characterize the service times. The servers are indexed in a fixed order. The entry of customers is modeled according to a Markovian arrival process. Upon arrival, a customer is placed in the infinite-capacity buffer if all servers are busy. Conversely, if multiple servers are idle, the arriving customer occupies the free server with the lowest index. Jockeying between servers during service is prohibited. This system is a natural generalization of the classical system with Markovian arrival process, phase-type service distribution and N servers (MAP/PH/N) to the case where the servers can be distinct. Such a system frequently arises as a model for various real-world systems with heterogeneous equipment. A multidimensional Markov chain with an infinitesimal generator having a special block structure characterizes the state dynamics of the system. We present an explicit expression for the generator and derive the tractable ergodicity condition. The system states exhibit a matrix-geometric stationary distribution. Additionally, analytical expressions to evaluate the core system performance metrics are established. Closed-form formulations for the Laplace–Stieltjes transforms and the initial moments of the distribution regarding both sojourn and waiting times of a generic customer are derived. Finally, numerical examples are presented. These examples illustrate the dependence of the system's key performance measures on the arrival rate and the service rate of the slowest server. They also confirm the importance of accounting for correlation in the arrival process and volatility of service times and demonstrate how the results can be used for optimization purposes.

Keywords: Markovian arrival process; heterogeneous servers; phase-type distribution; ergodicity;

performance measures
Mathematics Subject Classification: 60J28, 60K25, 90B22

1. Introduction

Queueing theory serves as a fundamental mathematical framework for evaluating, designing, and optimizing resource-sharing systems across various engineering and scientific domains. Over the past several decades, multiserver queueing models have been extensively studied due to their capability to represent concurrent processing. The servers were assumed to be identical. Although mathematically convenient, this idealization fails to address contemporary engineering challenges and creates a major gap between theoretical models and operational realities.

In modern industrial and computational ecosystems, heterogeneity among servers is an inherent and unavoidable feature driven by continuous lifecycle upgrades, architectural diversity, and variable human skills. In cloud computing platforms, legacy nodes inevitably coexist with next-generation blades possessing distinct CPU/GPU topologies and RAM capacities. In 5G/6G multitier networks, traffic must be dynamically dispatched across an asymmetric mix of broad-coverage macrocells and ultra-high-speed localized mmWave femtocells. Similarly, smart factories (Industry 4.0) integrate standard computer numerical control (CNC) machinery alongside automated robotic arms operating at entirely different processing speeds and operational phases.

To maximize operational efficiency and equipment lifetime, operators rarely utilize these diverse resources at random. Instead, they deploy an ordered hunting discipline (minimal-index rule), where inbound traffic automatically routes to the fastest, most cost-effective specialists first, engaging slower or energy-intensive legacy hardware strictly as backup capacity during peak traffic periods.

Despite its practical dominance, the mathematical modeling of such systems remains notoriously intractable. Moving beyond the two-server (fast/slow) paradigm introduces severe analytical complications. Because the servers are distinct, classical state-reduction techniques, such as server re-indexing based on the total number of busy devices, are fundamentally inapplicable. The precise state of each individual server (idle, busy, or its exact phase of service) must be permanently monitored. This causes the underlying multidimensional state space to expand exponentially as the server count scales, inducing a severe curse of dimensionality.

Furthermore, capturing the volatile, bursty, and highly correlated nature of modern data traffic requires abandoning simplistic Poisson and exponential assumptions in favor of robust Markovian arrival processes (MAP) and versatile phase-type (PH) distributions.

To bridge this critical gap, this paper addresses the following core research question: How can we construct a computationally tractable, exact analytical framework to evaluate the steady-state performance, joint state dynamics, and exact customer waiting and sojourn time distributions in an N-server queueing system operating under an arbitrary ordered hunting discipline with general MAP arrivals and PH service times?

By constructing an explicit block-tridiagonal infinitesimal generator and deriving matrix-geometric stationary distributions, this study provides a scalable algorithmic foundation. Crucially, we demonstrate how these analytical metrics can be directly mapped to managerial and economic trade-offs, such as minimizing nonlinear hourly operational expenses against steep customer delay penalties,

thereby transforming complex queueing theory into a practical tool for systemic infrastructure optimization.

As already noted, to accurately model these realistic and highly structured environments, it is essential to move beyond the classic exponential service time assumption. The PH distribution offers an exceptionally versatile tool for this purpose, as discussed in foundational monographs [1–3]. Its algebraic properties are discussed in [4, 5]. Further developments on computational frameworks and comprehensive application guides can be found in [6]. Because PH distributions are dense in the class of non-negative distributions, they allow one to approximate any continuous distribution with arbitrary precision while preserving the mathematically tractable Markovian structure via the method of phases. Motivated by these considerations, this paper provides a rigorous steady-state analysis of an N-server queueing system characterized by an unlimited capacity buffer, heterogeneous servers under an ordered hunting discipline, and distinct PH distributions of service times.

The arrival flow is modeled via a versatile MAP, which effectively captures the correlation and burstiness typical of modern network traffic. Early structural overviews and survey literature on these processes are provided in [7, 8], whereas extensive analytical characterizations are detailed in general queueing books [9, 10] and the recent paper [11]. More recently, specific multi-server policies incorporating hysteresis strategies and processing priority rules have been evaluated in [12–14], while energy-aware configurations, rating dependencies, and semi-open structures are explored in [15–17] and [18, 19]. Application to analysis of cognitive communication networks is discussed, along with [14], in [20]. The practically important problem of fitting real-world flow traces with MAPs is intensively discussed in the literature. General fitting principles and data trace methods are addressed in [3, 21, 22], while specific mathematical properties and parameter matrices are detailed in [23–25]. Furthermore, computational tools and custom parallel estimation algorithms are developed in [26–28].

When multiple servers are available, arriving customers are dispatched according to some discipline. A crucial operational feature of the considered model is the specific routing mechanism used to distribute incoming traffic among the heterogeneous servers. In systems with identical processors, the choice of a free server does not influence the overall system performance. However, when servers are heterogeneous, the dispatching discipline significantly impacts efficiency, capacity utilization, and waiting times. In this paper, we employ an ordered hunting discipline based on a minimal index rule. The servers are labeled with fixed indexes from 1 to N in a specific order, e.g, arranged in descending order of their efficiency or mean service rates ($\mu_1 \geq \mu_2 \geq \cdots \geq \mu_N$). Under this policy, an arriving customer who finds the system partially idle does not choose a free server at random. Instead, the idle server possessing the lowest index automatically receives the arriving customer. Only when all higher-index (faster) servers are simultaneously occupied will a customer be assigned to a lower-index (slower) server. Finally, if all N heterogeneous servers are busy upon a customer's arrival, the customer enters the infinite buffer and waits to be served. Once a server becomes free, it selects the next customer from the queue according to the standard first-in, first-out (FIFO) discipline. This nonrandom assignment policy creates a strong dependency between the states of different servers, which reflects real-world operations where supervisors prefer to utilize highly efficient or cost-effective equipment first, keeping slower or more expensive units as backup capacity for peak traffic periods.

The existing literature on the multiserver queues with heterogeneous servers is not very rich, especially in the case of more than two (fast and slow) available servers. Since the pioneering work [29], the relevant models assume a stationary arrival process of Poisson type and an exponential

distribution of service times. General overviews of historical queueing systems can be found in [30–32], while specific models focusing on customer delays and asymmetric processing capacities are addressed in [33–35]. Furthermore, dynamic routing policies and network fairness criteria are analyzed in [36–38], with additional performance optimizations for multi-resource allocation discussed in [39, 40]. Essential contributions regarding monotone control and stability are found in the seminal works of V. V. Rykov and D. V. Efrosinin, such as [41–43]. Extensions exploring finite-source systems, unreliable equipment, and specific threshold policies are further investigated in [44–46].

As mentioned above, we consider here a much more general Markovian arrival process and phase-type distribution of service times. A similar model with service control was considered in [47]. The version of the model with homogeneous servers (without analysis of waiting and sojourn time distribution) was briefly considered in [10]. The case with the batch arrivals but an exponential service time distribution in identical servers was studied in [48].

Analysis of the system with homogeneous servers is much simpler because, due to the identity of servers, the servers can be re-indexed whenever the number of customers changes, and the service process is completely defined by the number of currently busy servers and phases of service at each busy server (or the number of servers providing service at any existing phase; see, for example, [49]). In the system with distinct servers, the re-indexing of servers is impossible, and the state of each server (idle or busy, the phase of service in the latter case) has to be permanently monitored.

The subsequent sections of this paper are arranged as follows. Section 2 formally defines the mathematical model and its parameters. Section 3 constructs the underlying multidimensional Markov chain and derives the explicit block-tridiagonal structure of its infinitesimal generator. The system's ergodicity criterion and the corresponding equations for computing the steady-state distribution are developed in Section 4. The Laplace–Stieltjes transforms and formulas for the initial moments regarding the distributions of sojourn and waiting times are derived in Sections 5 and 6, respectively. Section 7 outlines the essential system performance metrics, which are subsequently accompanied by numerical examples in Section 8. The dependence of the core performance metrics on the arrival rate and the service rate of the slowest server is demonstrated. Furthermore, an example of applying the obtained results to optimize the economic cost criterion with respect to the service rate of the slowest server is presented. The importance of accounting for the correlation in the arrival process and variance of service times is also briefly discussed. Finally, Section 9 concludes the paper by summarizing the main findings, study limitations, and directions for future research.

2. The mathematical model

This study focuses on an N-server queueing system with an unlimited buffer, where the influx of customers is governed by the MAP. Let ν_t, $t \geq 0$, represent the directing process of the MAP. The state space associated with this irreducible continuous-time Markov chain ν_t is given by $\{1, \ldots, W\}$. The transition intensities of ν_t are characterized by the entries of two square matrices, D_0 and D_1, both of size W. Specifically, D_0 captures the transition intensities that occur without customer arrivals, whereas D_1 collects the intensities corresponding to transitions that trigger a customer arrival. Consequently, the infinitesimal generator of the process ν_t is given by the matrix $D(1) = D_0 + D_1$. The stationary distribution of ν_t is determined by the vector $\boldsymbol{\theta}$, which uniquely satisfies the linear system $\boldsymbol{\theta} D(1) = \mathbf{0}, \boldsymbol{\theta}\mathbf{e} = 1$. Throughout this paper, $\mathbf{e}$ and $\mathbf{0}$ denote column and row vectors of suitable dimensions,

composed entirely of ones and zeros, respectively.

We define the mean (fundamental) arrival rate λ of the MAP using the relation $\lambda = \boldsymbol{\theta} D_1 \mathbf{e}$. The formula $c_{var}^2 = 2\lambda\boldsymbol{\theta}(-D_0)^{-1}\mathbf{e} - 1$ yields the squared coefficient of variation c_{var}^2 of the customer inter-arrival intervals. Analogously, the correlation coefficient c_{cor} tracking consecutive arrival gaps is evaluated through $c_{cor} = (\lambda\boldsymbol{\theta}(-D_0)^{-1}D_1(-D_0)^{-1}\mathbf{e} - 1)/c_{var}^2$.

The servers are independent of each other and are indexed in a certain order. Server indexing is an interesting, practically important, and challenging task. In this paper, we consider an arbitrary indexing policy. The obtained results are expected to be useful in the future for solving this problem based on information about the service rates and costs at each server, as well as the system efficiency criteria.

The service time of a customer by the nth server, $n = \overline{1, N}$, is governed by the continuous-time Markov chain (directing process) $\eta_t^{(n)}$, $t \geq 0$. This process has an absorbing state 0 and the set $\left\{1, \cdots, M^{(n)}\right\}$ of transient states.

The initial state of the process $\eta_t^{(n)}$ at the epoch of starting the service is chosen among the transient states with the probabilities defined by the row-vector $\boldsymbol{\beta}^{(n)} = (\beta_1^{(n)}, ..., \beta_{M^{(n)}}^{(n)})$. The irreducible matrix $S^{(n)}$ of order $M^{(n)}$ defines the transition intensities of $\eta_t^{(n)}$ within the transient state space, which do not trigger the termination of a service. The diagonal entries of this matrix are negative. Their moduli define the rates of the exit of the process $\eta_t^{(n)}$ from its transient states. The off-diagonal elements specify the transition intensities occurring within the transient state space. The column vector $\mathbf{S}_0^{(n)} = -S^{(n)}\mathbf{e}$ characterizes the transition rates into the absorbing state that trigger a service completion.

The mth initial moment $b_m^{(n)}$ of the distribution of the service time of the nth server is computed as

$$b_m^{(n)} = m!\boldsymbol{\beta}^{(n)}(-S^{(n)})^{-m}\mathbf{e}, \ m \geq 1, \ n = \overline{1, N}.$$

The value μ_n defined by the formula $\mu_n^{-1} = \boldsymbol{\beta}^{(n)}(-S^{(n)})^{-1}\mathbf{e}$ is the mean service rate in the nth server. The value $(b_2^{(n)} - (b_1^{(n)})^2)/(b_1^{(n)})^2$ is the squared coefficient of variation of the service time in the nth server, $n = \overline{1, N}$.

If all servers are busy upon arrival, the arriving customer joins the buffer and is picked up for service later on, according to the FIFO discipline. If some servers are idle at an arrival moment, then the customer occupies the server having the minimal index.

This study aims to determine a sufficient ergodicity condition for the system and to obtain the steady-state distribution itself. Furthermore, we develop expressions for the key performance measures using this stationary distribution alongside the Laplace–Stieltjes transforms governing the sojourn and waiting times of an arbitrary customer.

3. The random process defining the dynamics of the system

At any given epoch t ($t > 0$), the following hold:

- i_t denotes the total number of customers present in the buffer, where $i_t \geq 0$;
- $\eta_t^{(n)}$ is the state of the underlying process of the service in the nth server, $n = \overline{1, N}$. This state belongs to the set $\{1, \ldots, M^{(n)}\}$ if this server is busy and is assumed to be 0 if the server is idle;
- ν_t represents the current phase of the directing process associated with the MAP, where $\nu_t = \overline{1, W}$.

Let $\hat{\mathcal{R}}$ be the state space of the process $\{\eta_t^{(1)}, \ldots, \eta_t^{(N)}\}$ defining the phases of service in all servers of the system when the buffer is not empty:

$$\hat{\mathcal{R}} = \{(r^{(1)}, \ldots, r^{(N)}) : \ r^{(n)} = \overline{1, M^{(n)}}, \ n = \overline{1, N}\},$$

and let $\tilde{\mathcal{R}}$ be the state space of the process $\{\eta_t^{(1)}, \ldots, \eta_t^{(N)}\}$ defining the phases of service in all servers of the system when the buffer is empty:

$$\tilde{\mathcal{R}} = \{(r^{(1)}, \ldots, r^{(N)}) : \ r^{(n)} = \overline{0, M^{(n)}}, \ n = \overline{1, N}\}.$$

Consider the continuous-time multidimensional process

$$\xi_t = \{i_t, \eta_t^{(1)}, \ldots, \eta_t^{(N)}, \nu_t\}, \quad t \geq 0.$$

It is straightforward to verify that this process constitutes an irreducible continuous-time Markov chain.

To simplify the analysis of the Markov chain ξ_t, let us introduce the notion of a level. We refer to the set of states that share the same value i ($i \geq 0$) for the first component, ordered lexicographically, as *level* i.

Note that the cardinality of the levels $i, \ i > 0$, is equal to $\hat{K} = W\hat{M}$, where $\hat{M} = \prod_{n=1}^{N} M^{(n)}$, and the cardinality of the level 0 is equal to $\tilde{K} = W\tilde{M}$, where $\tilde{M} = \prod_{n=1}^{N}(M^{(n)} + 1)$. The cardinality of the level 0 is larger than the cardinality of other levels because some servers can be idle in some states of the chain while all servers are necessarily busy in all states of the chain belonging to levels $i, \ i > 0$. This distinction has to be taken into account in the derivation of expressions for matrices of transitions between the states of level 0 and levels $i, \ i \geq 1$.

To facilitate the subsequent analysis of the Markov chain ξ_t, we establish the following notation:

- I denotes an identity matrix of a suitable order, with a subscript explicitly indicating its size when necessary;
- $O_{n \times n'}$ denotes zero matrices with n rows and n' columns;
- Matrix Kronecker products and sums are represented by the symbols $\otimes$ and $\oplus$, respectively; for a detailed discussion, see [50];
- $J_n = \begin{pmatrix} O_{1\times 1} & O_{1\times M^{(n)}} \\ O_{M^{(n)}\times 1} & I_{M^{(n)}} \end{pmatrix}, \ \tilde{J}_n = \begin{pmatrix} O_{1\times M^{(n)}} \\ I_{M^{(n)}} \end{pmatrix}, n = \overline{1, N}$;
- $J = \bigotimes_{n=1}^{N} J_n, \ \tilde{J} = \bigotimes_{n=1}^{N} \tilde{J}_n$;
- $\tilde{I}_{M^{(n)}} = \begin{pmatrix} O & I_{M^{(n)}} \end{pmatrix}, \ G_n = \begin{pmatrix} O_{1\times 1} & O_{1\times M^{(n)}} \\ \mathbf{S}_0^{(n)} & S^{(n)} \end{pmatrix}, \ n = \overline{1, N}$;
- $\mathcal{G}_N = \sum_{n=1}^{N} \left(I_{\prod_{l=1}^{n-1}(M^{(l)}+1)} \otimes G_n \otimes I_{\prod_{l=n+1}^{N}(M^{(l)}+1)} \right) = G_1 \oplus \cdots \oplus G_N$;

- $\mathcal{S}_N = \sum\limits_{n=1}^{N} \left(I_{\prod\limits_{l=1}^{n-1} M^{(l)}} \otimes S^{(n)} \otimes I_{\prod\limits_{l=n+1}^{N} M^{(l)}} \right) = S^{(1)} \oplus \cdots \oplus S^{(N)};$
- $\mathcal{T}_N = \sum\limits_{n=1}^{N} \left(I_{\prod\limits_{l=1}^{n-1} M^{(l)}} \otimes (\mathbf{S}_0^{(n)} \boldsymbol{\beta}^{(n)}) \otimes I_{\prod\limits_{l=n+1}^{N} M^{(l)}} \right) = (\mathbf{S}_0^{(1)} \boldsymbol{\beta}^{(1)}) \oplus \cdots \oplus (\mathbf{S}_0^{(N)} \boldsymbol{\beta}^{(N)});$
- $B_n = \begin{pmatrix} O_{1\times 1} & \boldsymbol{\beta}^{(n)} \\ O_{M^{(n)}\times 1} & O_{M^{(n)}\times M^{(n)}} \end{pmatrix}, n = \overline{1, N};$
- $\mathcal{B}_N = \sum\limits_{n=1}^{N} \left(\bigotimes\limits_{k=1}^{n-1} J_k \otimes B_n \otimes I_{\prod\limits_{k=n+1}^{N} (M^{(k)}+1)} \right).$

The following statement holds true:

Lemma 1. The infinitesimal generator of the chain $\xi_t, t \geq 0$, given by the matrix $\mathbf{Q}$, exhibits a block tridiagonal structure:

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

where the component blocks $\mathbf{Q}_{i,j}$ collect the intensities governing transitions from states within level i to those within level j, specified as follows:

$$\mathbf{Q}_{0,0} = \mathcal{G}_N \oplus D_0 + \mathcal{B}_N \otimes D_1,$$

$$\mathbf{Q}_{0,1} = \tilde{J} \otimes D_1,$$

$$\mathbf{Q}_{i,i} = \mathbf{Q}^0 = \mathcal{S}_N \oplus D_0, \ i \geq 1,$$

$$\mathbf{Q}_{i,i+1} = \mathbf{Q}^+ = I_{\hat{M}} \otimes D_1, \ \ i \geq 1,$$

$$\mathbf{Q}_{1,0} = \sum_{n=1}^{N} \bigotimes_{k=1}^{n-1} \tilde{I}_{M^{(k)}} \otimes \left(O \quad \mathbf{S}_0^{(n)} \boldsymbol{\beta}^{(n)} \right) \bigotimes_{k=n+1}^{N} \tilde{I}_{M^{(k)}} \otimes I_W,$$

$$\mathbf{Q}_{i,i-1} = \mathbf{Q}^- = \mathcal{T}_N \otimes I_W, \ i \geq 2.$$

The proof of Lemma 1 is established by investigating the possible transition intensities of the Markov chain $\xi_t, t \geq 0$. Briefly, the derivation proceeds as follows.

The block $\mathbf{Q}_{0,0}$ describes transition rates within level 0. It combines the rates of service phase changes without the change of the underlying process of arrivals (defined by the matrix $\mathcal{G}_N \otimes I_W$), rates of the underlying process of the MAP transitions without customer arrivals and changes in the service processes (defined by the matrix $I_{\tilde{M}} \otimes D_0$), and the rates of arrivals that occupy free servers (defined by the matrix $\mathcal{B}_N \otimes D_1$).

The block $\mathbf{Q}_{0,1}$ describes transition rates from level 0 to level 1. A customer arrives (with rates defined by the matrix D_1) when all servers are busy, starting the queue. The matrix $\tilde{J}$ selects the

corresponding states of the service processes and performs the required reduction of their state spaces from the set $\tilde{\mathcal{R}}$ to the set $\hat{\mathcal{R}}$.

The block $\mathbf{Q}_{i,i}$, denoted also as Q^0, describes transition rates within level $i \geq 1$. It represents transitions of the underlying process of the MAP without customer arrival or service progression without departures defined by the matrix $(\mathcal{S}_N \oplus D_0)$.

The block $\mathbf{Q}_{i,i+1}$, denoted also as Q^+, describes transition rates from level i to $i+1$ $(i \geq 1)$. A new customer arrives (with rates of transitions defined by the matrix D_1) and enters the buffer while all servers remain busy. The matrix $I_{\hat{M}}$ reflects the absence of changes of the underlying processes of services.

The block $\mathbf{Q}_{i,i-1}$, denoted also as Q^-, describes transition rates from level i to $i-1$ $(i \geq 2)$. A server finishes service and immediately picks up a customer from the buffer. The matrix $\mathcal{T}_N$ defines the respective transition rates.

The block $\mathbf{Q}_{1,0}$ describes transition rates from level 1 to level 0. A server finishes the service of a customer and starts service of the single waiting customer (the corresponding rates are given by the matrix $\mathbf{S}_0^{(n)}\boldsymbol{\beta}^{(n)}$ if the service was finished in the nth server, $n = \overline{1,N}$), leaving the buffer entirely empty. The extension of the state space of the components describing service processes from the set $\hat{\mathcal{R}}$ to the set $\tilde{\mathcal{R}}$ occurs.

Lemma 1 is proven.

4. Steady-state analysis of the customer number within the system

Theorem 1. The Markov chain $\xi_t, t \geq 0$ possesses a stationary distribution if and only if the condition

$$\lambda < \sum_{n=1}^{N} \mu_n \tag{1}$$

holds.

Proof: As detailed in [1], the Markov chain $\xi_t, t \geq 0$ constitutes a level-independent quasi-birth-and-death process. Therefore, the criterion of its ergodicity is the fulfillment of the inequality

$$\mathbf{u}\mathbf{Q}^-\mathbf{e} > \mathbf{u}\mathbf{Q}^+\mathbf{e}, \tag{2}$$

where the vector $\mathbf{u}$ is determined as the unique solution of the system

$$\mathbf{u}(\mathbf{Q}^- + \mathbf{Q}^0 + \mathbf{Q}^+) = \mathbf{0},\ \mathbf{u}\mathbf{e} = 1. \tag{3}$$

By the direct substitution into (3) and by using the properties of the Kronecker product, it is possible to verify that the vector $\mathbf{u}$ has the following form:

$$\mathbf{u} = \bigotimes_{n=1}^{N} \boldsymbol{\psi}_n \otimes \boldsymbol{\theta}, \tag{4}$$

where

$$\boldsymbol{\psi}_n = \mu_n \boldsymbol{\beta}^{(n)}(-S^{(n)})^{-1},\ n = \overline{1,N}.$$

Inequality (1) is directly obtained from (2) by substitution of expression (4). Theorem 1 is proven.

Inequality (1) is easy to interpret: The stationary distribution of the Markov chain exists if and only if the mean customer arrival rate is less than the sum of service rates at all servers of the system.

Let condition (1) be fulfilled. Then, the Markov chain ξ_t is ergodic.

Consequently, the steady-state probabilities of this Markov chain, characterized as the limits

$$\pi(i, r^{(1)}, \dots, r^{(N)}, \nu) = \lim_{t\to\infty} P\{i_t = i,\ (r^{(1)}, \dots, r^{(N)}) \in \hat{\mathcal{R}}, \nu_t = \nu\},\ i \geq 1,$$

$$\pi(0, r^{(1)}, \dots, r^{(N)}, \nu) = \lim_{t\to\infty} P\{i_t = 0,\ (r^{(1)}, \dots, r^{(N)}) \in \tilde{\mathcal{R}}, \nu_t = \nu\},\ \nu = \overline{1, W},$$

exist.

Based on the chosen lexicographical ordering for the states within each level, we construct the row-vectors

$$\boldsymbol{\pi}(i, r^{(1)}, \dots, r^{(N)}) = (\pi(i, r^{(1)}, \dots, r^{(N)}, 1), \dots, \pi(i, r^{(1)}, \dots, r^{(N)}, W))$$

of the stationary probabilities $\pi(i, r^{(1)}, \dots, r^{(N)}, \nu)$, and the row-vectors $\boldsymbol{\pi}_i$, consisting of the vectors $\boldsymbol{\pi}(i, r^{(1)}, \dots, r^{(N)})$, $i \geq 0$. Define also the infinite-dimensional probability vector $\boldsymbol{\pi} = (\boldsymbol{\pi}_0, \boldsymbol{\pi}_1, \boldsymbol{\pi}_2, \dots)$.

The stationary probability vector $\boldsymbol{\pi}$ satisfies the system of global balance equations

$$\boldsymbol{\pi}\mathbf{Q} = \mathbf{0}, \boldsymbol{\pi}\mathbf{e} = 1.$$

The vectors $\boldsymbol{\pi}_i,\ i \geq 0$, defining the stationary distribution of the states of the Markov chain ξ_t, can be computed as follows.

Theorem 2. The vectors $\boldsymbol{\pi}_i$ follow the so-called matrix-geometric distribution:

$$\boldsymbol{\pi}_i = \boldsymbol{\pi}_1 \mathcal{A}^{i-1},\ i \geq 2, \tag{5}$$

where $\mathcal{A}$ is the minimal non-negative solution to the quadratic matrix equation

$$\mathcal{A}^2\mathbf{Q}^- + \mathcal{A}\mathbf{Q}^0 + \mathbf{Q}^+ = O. \tag{6}$$

The boundary vectors $\boldsymbol{\pi}_0$ and $\boldsymbol{\pi}_1$ are uniquely determined by solving the system

$$\boldsymbol{\pi}_0\mathbf{Q}_{0,0} + \boldsymbol{\pi}_1\mathbf{Q}_{1,0} = \mathbf{0},$$

$$\boldsymbol{\pi}_0\mathbf{Q}_{0,1} + \boldsymbol{\pi}_1(\mathbf{Q}^0 + \mathcal{A}\mathbf{Q}^-) = \mathbf{0},$$

$$\boldsymbol{\pi}_0\mathbf{e} + \boldsymbol{\pi}_1(I - \mathcal{A})^{-1}\mathbf{e} = 1.$$

The so-called logarithmic reduction algorithm for quasi-birth-death processes from [51] can be used for numerically solving Eq (6).

Remark 1. The results obtained here for the stationary distribution of the system states with an infinite buffer can be easily extended to the case of a system with a finite buffer of capacity, say, J. In this case, the generator of the multidimensional Markov chain describing the system behavior is not a matrix of infinite size like the matrix $\mathbf{Q}$ defined in Lemma 1. Instead, it is a finite matrix with $J + 1$ block rows. The first J block rows coincide with the respective block rows of the generator $\mathbf{Q}$. The last $(J + 1)$-th block row consists of zero blocks except for the last two blocks, which are defined as $\mathbf{Q}_{J,J-1} = \mathcal{T}_N \otimes I_W$ and $\mathbf{Q}_{J,J} = \mathcal{S}_N \oplus (D_0 + D_1)$. The vectors of the stationary probabilities of the system states can be computed in this case using the known numerically stable algorithms given, for example, in [52, 53].

5. Evaluation of the customer sojourn time distribution within the system

As specified in the model description, an incoming customer enters the buffer and is subsequently selected for service under the FIFO discipline if all servers are occupied upon arrival. Alternatively, if any servers are vacant, the customer is assigned to the one holding the minimum index.

Let $V(x)$ denote the cumulative distribution function of the sojourn time for a generic customer, and let $v(s) = \int_0^\infty e^{-sx}dV(x)$, $Re\ s > 0$ represent its Laplace-Stieltjes transform (LST). The known probabilistic interpretation of values of the LST for real values of s is as follows. Independently of the operation of the considered system, let a virtual stationary Poisson flow of so-called catastrophes arrive with the rate of s. The probability that a generic customer completes the sojourn in the system without experiencing any catastrophe corresponds exactly to the LST $v(s)$.

Based on the customer admission policy, it is evident that the sojourn time of a generic, tagged customer is completely independent of future customer arrivals. Consequently, tracking the background arrival process can be discontinued immediately following the entry of the tagged customer.

Introduce also the following notation.

Let $\mathbf{v}_i(s)$ denote a column vector of dimension $\hat{M}$, whose entries represent the probability that no catastrophe occurs during the sojourn time of a tagged customer. Here, the customer arrives to find i ($i \geq 1$) individuals already waiting in the buffer (excluding the tagged customer itself), conditioned on the specific values of the components $\{\eta_t^{(1)}, \ldots, \eta_t^{(N)}\}$ that specify the service phases across all servers at the arrival epoch.

Similarly, the elements of the $\hat{M}$-dimensional column vector $\mathbf{v}_0(s)$ specify the probability that no catastrophe takes place during the sojourn of a tagged customer who arrives when the buffer is completely empty but when all servers are occupied.

Furthermore, we utilize $\boldsymbol{\rho}(s)$ to represent the column vector of size $\tilde{M}$, whose components define the probability of no catastrophe during the sojourn time of a tagged customer who initiates service immediately upon arrival, which happens when at least one server is vacant at the arrival epoch.

Lemma 2. The column vectors $\mathbf{v}_i(s)$ are computed from the recursion

$$\mathbf{v}_i(s) = (sI - \mathcal{S}_N)^{-1}\mathcal{T}_N\mathbf{v}_{i-1}(s),\ i \geq 1, \tag{7}$$

with the initial condition

$$\mathbf{v}_0(s) = (sI - \mathcal{S}_N)^{-1}\Big[\sum_{n=1}^{N}\bigotimes_{k=1}^{n-1}\mathbf{e}_{M^{(k)}} \otimes \mathbf{S}_0^{(n)}\boldsymbol{\beta}^{(n)}(sI - S^{(n)})^{-1}\mathbf{S}_0^{(n)}) \otimes \bigotimes_{k=n+1}^{N}\mathbf{e}_{M^{(k)}}\Big]. \tag{8}$$

Proof: Evidently, for a tagged customer arriving to find i ($i \geq 1$) individuals in the buffer to experience no catastrophe during their sojourn, it is necessary and sufficient that no catastrophe occurs until the queue length reduces to $i-1$, and subsequently, that no catastrophe arrives during the remaining sojourn of a customer facing a buffer size of $i - 1$.

Therefore,

$$\mathbf{v}_i(s) = \int_0^\infty e^{-sx}e^{\mathcal{S}_N x}\mathcal{T}_N dx\mathbf{v}_{i-1}(s),$$

which implies formula (7). Formula (8) is derived analogously, taking into account that, once service is completed in one, say, the nth, a server catastrophe must not occur during a new service there (the probability of this event is the LST of service time $\boldsymbol{\beta}^{(n)}(sI - S^{(n)})^{-1}\mathbf{S}_0^{(n)}$).

Lemma 3. The column vector $\boldsymbol{\rho}(s)$ is computed by the formula

$$\boldsymbol{\rho}(s) = \Big[\sum_{n=1}^{N}\bigotimes_{k=1}^{n-1} J_k \otimes \begin{pmatrix} O_{1\times 1} & \boldsymbol{\beta}^{(n)}(sI - S^{(n)})^{-1}(-S^{(n)}) \\ O_{M^{(n)}\times 1} & O_{M^{(n)}\times M^{(n)}} \end{pmatrix} \otimes \bigotimes_{k=n+1}^{N} I_{M^{(k)}+1}\Big]\mathbf{e}_{\tilde{M}}. \tag{9}$$

Proof: Because the vector $\boldsymbol{\rho}(s)$ characterizes the probability that no catastrophe occurs during the sojourn of a tagged customer who initiates service immediately due to the presence of vacant servers, the customer selects the nth server, $n = \overline{1,N}$, where n represents the smallest index among all idle units. Service commences in this specific server, and the absence of catastrophes must be maintained throughout its duration. This line of reasoning directly yields Eq (9).

Now, we are able to formulate the following result.

Let $\boldsymbol{\pi}_0(J)$ be the vector of size $\hat{M}W$ obtained from the vector $\boldsymbol{\pi}_0 J$ of size $\tilde{M}W$ by removal of all zero components.

Theorem 3. The LST $v(s)$ is defined by the formula

$$v(s) = \lambda^{-1}\Big[\sum_{i=1}^{\infty}\boldsymbol{\pi}_i(I_{\hat{M}}\otimes D_1\mathbf{e}_W)\mathbf{v}_i(s) + \boldsymbol{\pi}_0(J)(I_{\hat{M}}\otimes D_1\mathbf{e}_W)\mathbf{v}_0(s) + \boldsymbol{\pi}_0(I_{\tilde{M}}\otimes D_1\mathbf{e}_W)(I-J)\boldsymbol{\rho}(s)\Big],$$

where the vectors $\mathbf{v}_i(s),\ i \geq 0$, and $\boldsymbol{\rho}(s)$ are defined by Lemmas 2 and 3, correspondingly.

The proof of Theorem 3 immediately follows from the formula of total probability.

6. Analysis of the distribution of the waiting time of a customer in the system

A similar approach can be applied to evaluate the waiting time distribution of a generic customer, leading to the following result.

Theorem 4. The LST $w(s)$ of the distribution of the waiting time of a customer in the system is defined by the formula

$$w(s) = \lambda^{-1}\Big[\sum_{i=1}^{\infty}\boldsymbol{\pi}_i(I_{\hat{M}}\otimes D_1\mathbf{e}_W)\mathbf{w}_i(s) + \boldsymbol{\pi}_0(J)(I_{\hat{M}}\otimes D_1\mathbf{e}_W)\mathbf{w}_0(s) + \boldsymbol{\pi}_0(I_{\tilde{M}}\otimes D_1\mathbf{e}_W)(I-J)\mathbf{e}_{\tilde{M}}\Big],$$

where the vectors $\mathbf{w}_i(s),\ i \geq 0$ are computed from the recursion

$$\mathbf{w}_i(s) = (sI - \mathcal{S}_N)^{-1}\mathcal{T}_N\mathbf{w}_{i-1}(s),\ i \geq 1$$

with the initial condition

$$\mathbf{w}_0(s) = (sI - \mathcal{S}_N)^{-1}\mathcal{T}_N\mathbf{e}_{\hat{M}}.$$

Having the procedure for computing the LST $w(s)$, we can compute the moments of waiting time distribution, in particular, an average waiting time.

Corollary 1. The mean waiting time W_{buffer} is calculated by the formula

$$W_{buffer} = -\lambda^{-1}\Big[\sum_{i=1}^{\infty}\boldsymbol{\pi}_i(I_{\hat{M}}\otimes D_1\mathbf{e}_W)\mathbf{w}_i'(0) + \boldsymbol{\pi}_0(J)(I_{\hat{M}}\otimes D_1\mathbf{e}_W)\mathbf{w}_0'(0)\Big],$$

where the vectors $\mathbf{w}'_i(0),\ i \geq 0$ are computed from the recursion

$$\mathbf{w}'_i(0) = (-\mathcal{S}_N)^{-1}\Big(\mathcal{T}_N\mathbf{w}'_{i-1}(0) - \mathbf{e}_{\hat{M}}\Big),\ i \geq 1,$$

with the initial condition

$$\mathbf{w}'_0(0) = -(\mathcal{S}_N)^{-2}\mathcal{T}_N\mathbf{e}_{\hat{M}}.$$

The proof of Corollary 1 is evident because it is well-known that $W_{buffer} = -w'(0)$, and the recursion for computing the vectors $\mathbf{w}'_i(0),\ i \geq 0$, obviously follows from the recursion for computing the vectors $\mathbf{w}'_i(s),\ i \geq 0$ given in Theorem 4.

Corollary 2. The kth moment of the waiting time distribution is calculated by the formula

$$(-1)^k\lambda^{-1}\Big[\sum_{i=1}^{\infty}\boldsymbol{\pi}_i(I_{\hat{M}} \otimes D_1\mathbf{e}_W)\mathbf{w}_i^{(k)}(0) + \boldsymbol{\pi}_0(J)(I_{\hat{M}} \otimes D_1\mathbf{e}_W)\mathbf{w}_0^{(k)}(0)\Big],\ k \geq 2,$$

where the derivatives of order $k,\ k \geq 2$ of the vectors $\mathbf{w}_i(s),\ i \geq 0$ at point $s = 0$ are obtained using the following recursive formulas:

$$\mathbf{w}_i^{(k)}(0) = (-\mathcal{S}_N)^{-1}\Big(\mathcal{T}_N\mathbf{w}_{i-1}^{(k)}(0) - k\mathbf{w}_i^{(k-1)}(0)\Big),\ i \geq 1$$

with the initial condition

$$\mathbf{w}_0^{(k)}(0) = k(-\mathcal{S}_N)^{-1}\mathbf{w}_0^{(k-1)}(0).$$

Remark 2. Note that the matrix $\mathcal{S}_N$ is nonsingular, as it is the Kronecker sum of the irreducible subgenerators. Therefore, no difficulties arise in the straightforward derivation of Corollaries 1 and 2 from Theorem 4.

Formulas for computation of the average sojourn time of a customer in the system and its higher moments are derived similarly.

7. Performance measures of the system

Following the derivation of the steady-state probabilities for the system states, we are able to evaluate several essential performance metrics.

The mean queue length L_{buffer} of customers waiting in the buffer is evaluated via

$$L_{buffer} = \sum_{i=1}^{\infty} i\boldsymbol{\pi}_i\mathbf{e} = \boldsymbol{\pi}_1(I - \mathcal{A})^{-2}\mathbf{e}.$$

The probability $P_{empty-buffer}$ that the buffer is empty at an arbitrary moment is computed by

$$P_{empty-buffer} = \boldsymbol{\pi}_0\mathbf{e}.$$

The stationary probability $P_0^{(n)}$ that the nth server, $n = \overline{1, N}$, is vacant at a generic epoch is evaluated via

$$P_0^{(n)} = \boldsymbol{\pi}_0\Big(\bigotimes_{r=1}^{n-1}\mathbf{e}_{M^{(r)}+1} \otimes \mathbf{f}^{(n)} \otimes \bigotimes_{r=n+1}^{N}\mathbf{e}_{M^{(r)}+1} \otimes \mathbf{e}_W\Big),$$

where $\mathbf{f}^{(n)}$ is the column vector of size $(M^{(n)}+1)$ having the first entry equal to 1 and other entries equal to 0.

The average number N_{busy} of busy servers at an arbitrary moment is given by

$$N_{busy} = \sum_{n=1}^{N}(1 - P_0^{(n)}).$$

The average number L_{system} of customers in the system is computed by

$$L_{system} = L_{buffer} + N_{busy}.$$

The output rate φ_n of customers from the nth server is defined by

$$\varphi_n = \boldsymbol{\pi}_0\Big[\bigotimes_{k=1}^{n-1}\mathbf{e}_{M^{(k)}+1}\otimes\begin{pmatrix}0\\ \mathbf{S}_0^{(n)}\end{pmatrix}\otimes\bigotimes_{k=n+1}^{N}\mathbf{e}_{M^{(k)}+1}\otimes\mathbf{e}_W\Big]$$

$$+\boldsymbol{\pi}_1(I-\mathcal{A})^{-1}\Big[\bigotimes_{r=1}^{n-1}\mathbf{e}_{M^{(r)}}\otimes\mathbf{S}_0^{(n)}\otimes\bigotimes_{r=n+1}^{N}\mathbf{e}_{M^{(r)}}\otimes\mathbf{e}_W\Big],\ n=\overline{1,N}.$$

The relation $\lambda = \sum_{n=1}^{N}\varphi_n$ can be used for control of accuracy of computation of the stationary distribution of the system states.

The fraction F_n of customers that receive service by the nth server is calculated as

$$F_n = \frac{\varphi_n}{\sum_{k=1}^{N}\varphi_k} = \frac{\varphi_n}{\lambda},\ n=\overline{1,N}.$$

The average service rate by a server is computed by

$$\bar{\mu} = \sum_{n=1}^{N}F_n\mu_n = \lambda^{-1}\sum_{n=1}^{N}\varphi_n\mu_n.$$

The probability $P_0^{(serv)}$ that all servers are idle at an arbitrary moment is given by

$$P_0^{(serv)} = \boldsymbol{\pi}_0\Big(\bigotimes_{n=1}^{N}\mathbf{f}^{(n)}\otimes\mathbf{e}_W\Big).$$

The probability $P_0^{(arr)}$ that all servers are idle at an arbitrary customer arrival moment is computed by

$$P_0^{(arr)} = \boldsymbol{\pi}_0\Big(\bigotimes_{n=1}^{N}\mathbf{f}^{(n)}\otimes\frac{D_1\mathbf{e}_W}{\lambda}\Big).$$

The probability P_{imm} that an arbitrary customer will succeed to enter the service immediately upon arrival is calculated by

$$P_{imm} = \boldsymbol{\pi}_0\Big(\mathcal{B}_N\otimes\frac{D_1}{\lambda}\Big)\mathbf{e}.$$

The share, Z_n of customers, who start service immediately upon arrival by the nth server, is obtained using

$$Z_n = \frac{1}{\lambda}\boldsymbol{\pi}_0\Big[\Big(\bigotimes_{k=1}^{n-1}J_k\otimes B_n\otimes I_{\prod_{k=n+1}^{N}(M^{(k)}+1)}\Big)\otimes D_1\Big]\mathbf{e},\ n=\overline{1,N}.$$

8. Numerical results

The queueing framework investigated in this study is extremely versatile, as it captures both the correlation and volatility of customer inte-arrival intervals along with the variance and higher-order moments characterizing the service duration distribution. The obtained algorithmic results establish a foundation for the exact quantitative evaluation of how the system is influenced by factors such as system load (varied via scaling the mean arrival rate or service times), the correlation coefficient of interarrival times, the coefficients of variation of interarrival and service times, server indexing rules, and the distinct capacity configuration of each server. Because evaluating various combinations of these factors is quite complex, we leave such an exhaustive analysis for future studies. Instead, we restrict this work to a brief illustration of the feasibility and outcomes of the presented algorithms, demonstrating the specific impacts of the mean arrival rate and the service rate of the slowest server. Additionally, we formulate and solve one possible optimization problem, the solution to which can be practically implemented based on our results.

8.1. Example of study of the impacts of the mean arrival rate and the service rate of the slowest server

Let initially the MAP-input be characterized by the matrices

$$D_0 = \begin{pmatrix} -1.35164 & 0 \\ 0 & -0.04387 \end{pmatrix}, \quad D_1 = \begin{pmatrix} 1.34265 & 0.00899 \\ 0.02443 & 0.01944 \end{pmatrix}.$$

For this arrival process, the correlation coefficient between two consecutive interarrival times is set to $c_{cor} = 0.2$, and the corresponding squared coefficient of variation is given by $c_{var}^2 = 13.4$. Throughout the illustrated experiment, the fundamental arrival rate, λ of the MAP will be varied; this is achieved by scaling both matrices D_0 and D_1 by an identical scalar factor.

Let us assume that the total number N of servers is equal to 3, and $M^{(1)} = 2, M^{(2)} = 2, M^{(3)} = 3$. We will denote the PH-distributions of service times on three devices as $PH_1^{(serv)}$, $PH_2^{(serv)}$, $PH_3^{(serv)}$.

$PH_1^{(serv)}$ is the second-order hyperexponential distribution with $c_{var}^2 = 4.55$, characterized by the following vector and matrix:

$$\boldsymbol{\beta}^{(1)} = (0.1, 0.9), S^{(1)} = \begin{pmatrix} -2 & 0 \\ 0 & -18 \end{pmatrix}.$$

The mean service rate is $\mu_1 = 10$.

$PH_2^{(serv)}$ is the second-order hyperexponential distribution with $c_{var}^2 = 4.54$, characterized by the following vector and matrix:

$$\boldsymbol{\beta}^{(2)} = (0.1, 0.9), S^{(2)} = \begin{pmatrix} -1.5 & 0 \\ 0 & -13.5 \end{pmatrix}.$$

The mean service rate is $\mu_2 = 7.5$.

$PH_3^{(serv)}$ is the third-order hyperexponential distribution with $c_{var}^2 = 4.28$. Initially, it is characterized by the following vector and matrix:

$$\boldsymbol{\beta}^{(3)} = \left(\frac{1}{15}, \frac{2}{15}, \frac{12}{15}\right), S^{(3)} = \begin{pmatrix} -0.2 & 0 & 0 \\ 0 & -0.4 & \\ 0 & 0 & -2.4 \end{pmatrix}.$$

The mean service rate corresponding to this choice of the matrix $S^{(3)}$ is $\mu_3 = 1$.

In the presented Figures, dependencies of several performance measures of the system on the mean arrival rate λ and the mean service rate μ_3 by the slowest, third, server are shown.

Variation of the value of μ_3 is achieved via the multiplication of the entries of the matrix $S^{(3)}$ by the corresponding constant. The value of μ_3 is varied from 0.2 to 7 with a step of 0.2. The value of λ is varied in the range from 1 to 20. Note that because $\mu_1 + \mu_2 = 17.5$, the rate μ_3 can have an arbitrary value in the indicated range when $\lambda < 17.5$. For larger values of λ, the value of μ_3 has to be larger than $\lambda - 17.5$. Otherwise, the ergodicity condition fails to hold, and the values of L_{buffer} and W_{buffer} are infinite.

Figure 1 shows the behavior of the value L_{buffer}.

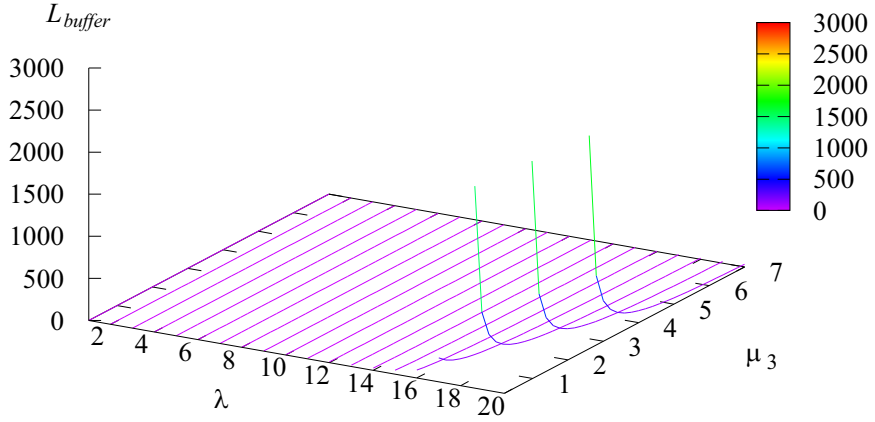

Figure 1. Dependence of the average number of customers in the buffer L_{buffer}.

From Figure 1, we observe that the number of customers in the buffer remains less than 1 when $\lambda \leq 7$, after which rapid queue growth occurs abruptly at higher arrival rates. This sharp increase appears as λ approaches the total service capacity, driving the system toward the boundary of stability. Increasing μ_3 has a significant stabilizing effect, dramatically suppressing buffer accumulation.

Figure 2 shows the behavior of the value W_{buffer}.

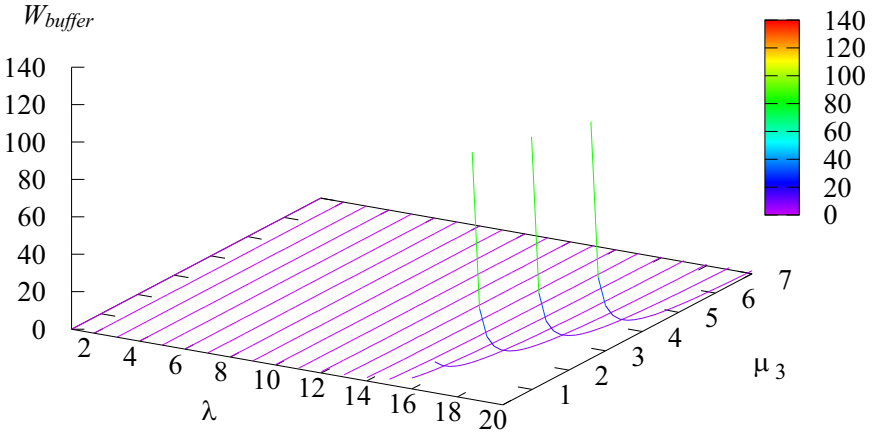

Figure 2. Dependence of the average waiting time of customers in the buffer W_{buffer}.

The overall surface topology is highly similar to the buffer size plot (Figure 1). This directly

confirms the validity of Little's formula

$$W_{buffer} = \lambda^{-1} L_{buffer}$$

for this heterogeneous system.

The key observations from Figure 2 are as follows. When the arrival rate is low, the average waiting time remains virtually negligible across all values of μ_3. As λ increases toward the total service capacity of the system, the waiting time experiences an asymptotic explosion, forming sharp vertical peaks. Increasing the service rate μ_3 of the slowest server extends the ergodicity boundary, which significantly suppresses the peaks and keeps average waiting times low even at higher load levels.

From Figure 3, we get that the value of N_{busy} is primarily governed by the arrival intensity λ. As traffic grows, it rapidly climbs from near-zero toward the absolute maximum system capacity of three servers. Impact of ordered hunting can be commented as follows: Under the minimal index rule, incoming customers fill the fast servers first. As a result, the third server remains mostly idle at lower traffic loads, keeping $N_{busy} \leq 2$. Modifying μ_3 itself has a very small impact on the number of busy servers when the system is under stable or low load conditions. It only begins to matter significantly during high-intensity periods, when all three servers must run continuously to prevent an infinite queue explosion.

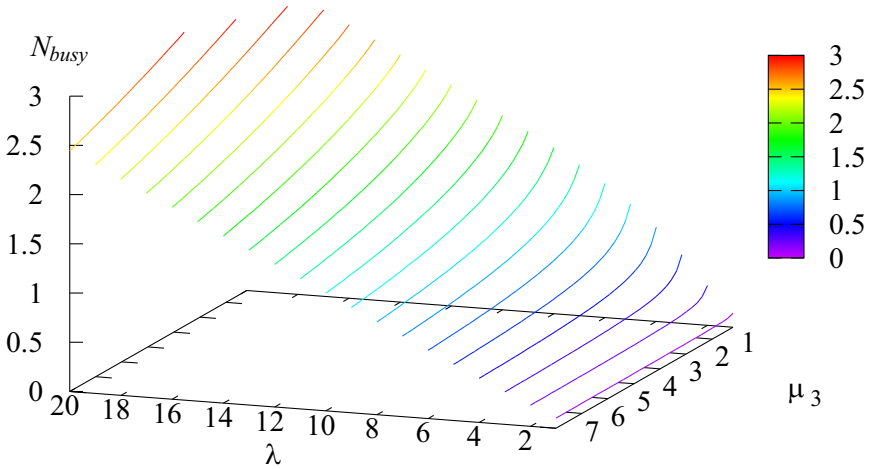

Figure 3. Dependence of the average number, N_{busy}, of busy servers.

Figure 4 allows us to draw the following conclusions. The buffer remains nearly empty when $\lambda \leq 4$. The probability P_{empty} of the buffer being empty drops sharply to 0 as the system approaches saturation. Increasing μ_3 expands the system stability boundary, and a higher μ_3 delays buffer accumulation under heavy workloads.

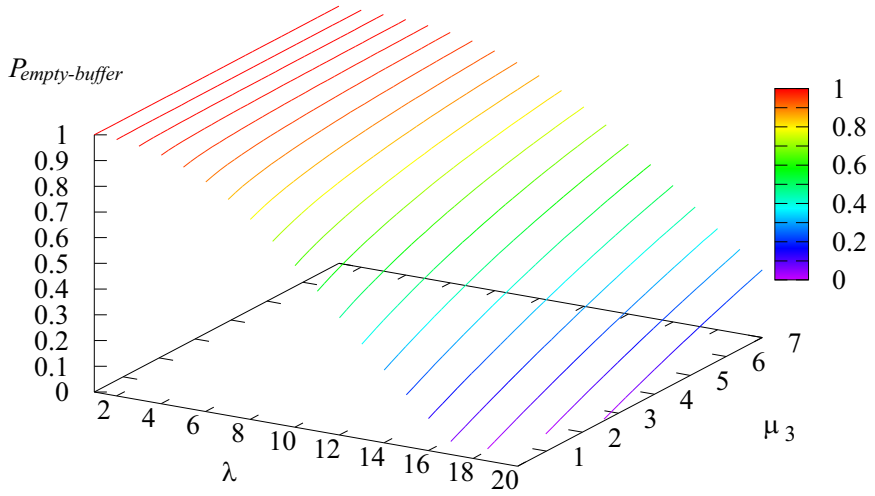

Figure 4. Dependence of the probability $P_{empty-buffer}$ of an empty buffer.

It is seen in Figure 5 that when arrival intensity is low ($\lambda \leq 4$), the probability of immediate service remains close to 1 across all values of μ_3, meaning that arriving customers almost never wait. As λ escalates toward full capacity, the probability of immediate access rapidly drops to 0, indicating that arriving customers are forced straight into the queue buffer. Increasing the service speed of the slowest server successfully stretches the boundary of stable performance, delaying the sudden drop in P_{imm} under medium-to-high workloads.

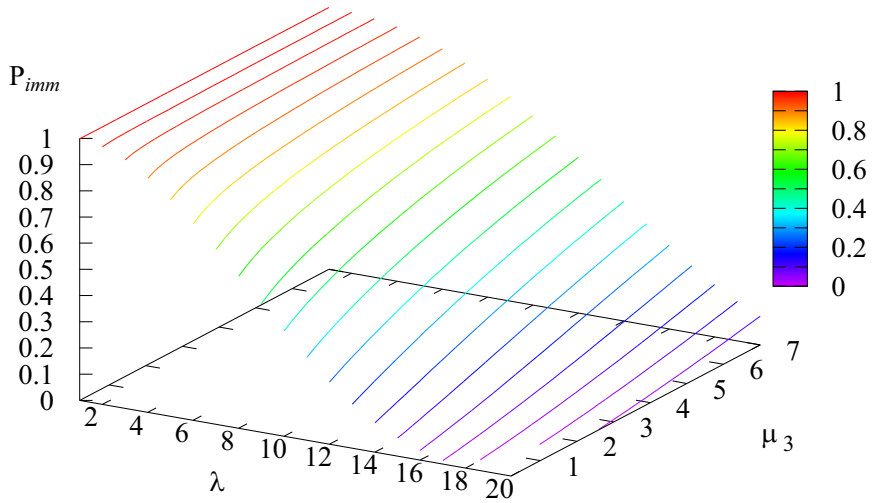

Figure 5. Dependence of the probability P_{imm} of immediate access of an arriving customer to servers.

Figures 6–8 present the dependencies of fractions F_n of customers that receive service by the nth server, $n = \overline{1, N}$.

At low traffic ($\lambda \to 0$), F_1 is highest (≈ 0.9) due to the minimal index rule. As λ increases, F_1 drops because the first server becomes fully occupied. Increasing μ_3 slightly decreases F_1 at high loads as the third server absorbs more traffic. The second server fraction (F_2) starts near 0.1 and grows as traffic overflows from the first server. It reaches a maximum under moderate loads before stabilizing

as the third server activates. The third server fraction (F_3) stays near 0 when $\lambda \leq 4$ because the first two servers handle the load. Unlike F_1 and F_2, F_3 is highly sensitive to μ_3. At $\lambda = 20$, F_3 grows significantly as μ_3 scales from 2.6 to 7.

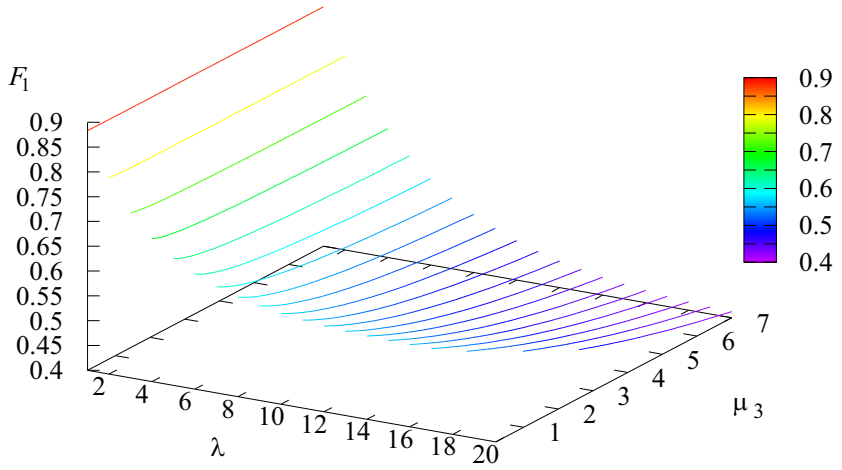

Figure 6. Dependence of the fraction F_1 of customers that receive service by the first server.

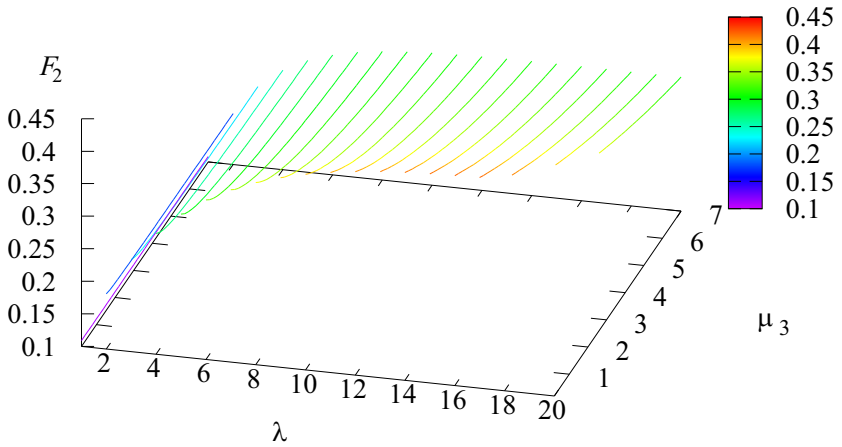

Figure 7. Dependence of the fraction F_2 of customers that receive service by the second server.

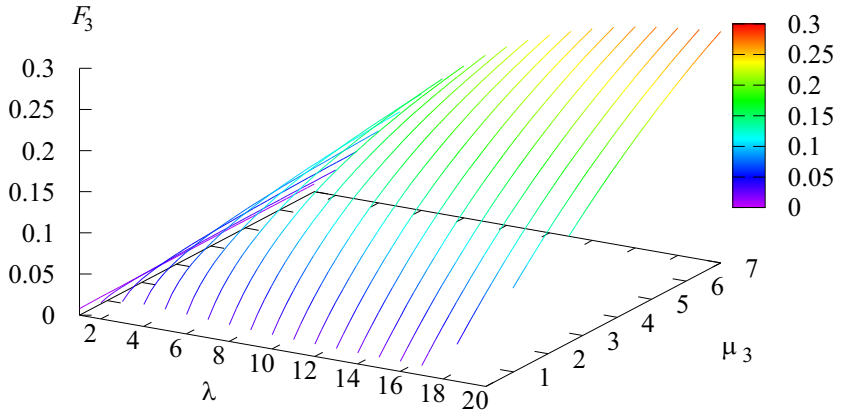

Figure 8. Dependence of the fraction F_3 of customers that receive service by the third server.

Figures 9–11 present the dependencies of the share, Z_n of customers, which start service immediately upon arrival by the nth server, $n = \overline{1, N}$.

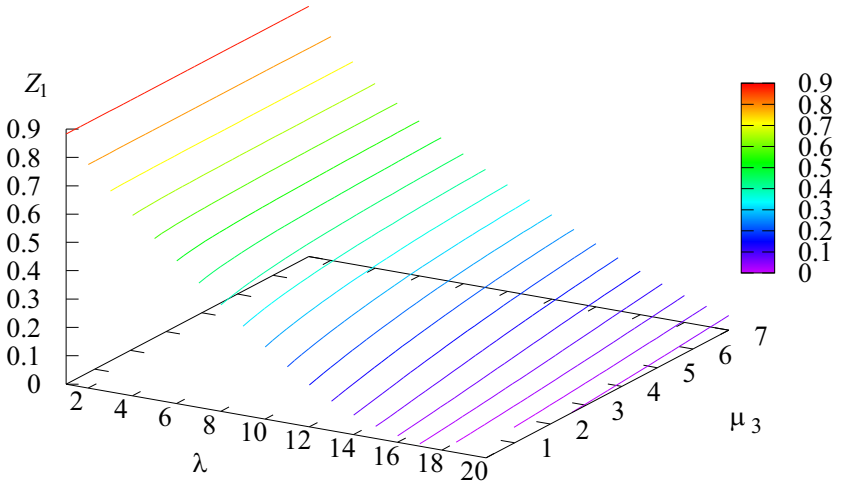

Figure 9. Dependence of the fraction Z_1 of customers that receive service immediately upon arrival by the first server.

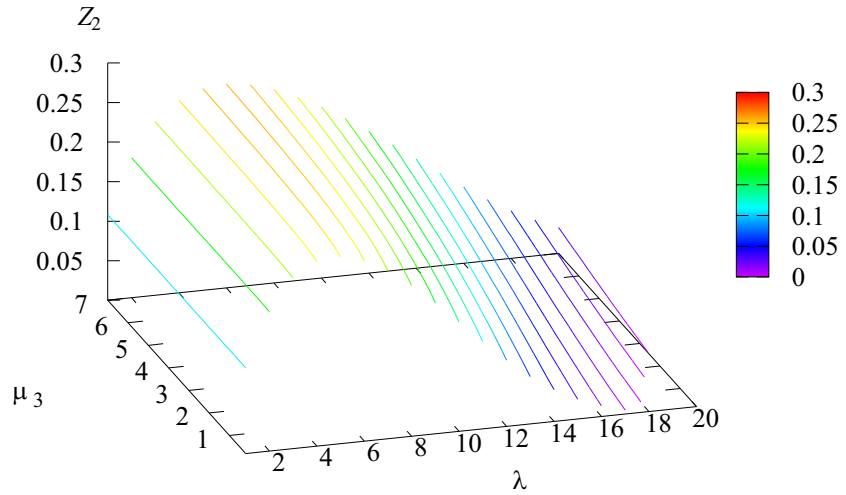

Figure 10. Dependence of the fraction Z_2 of customers that receive service immediately upon arrival by the second server.

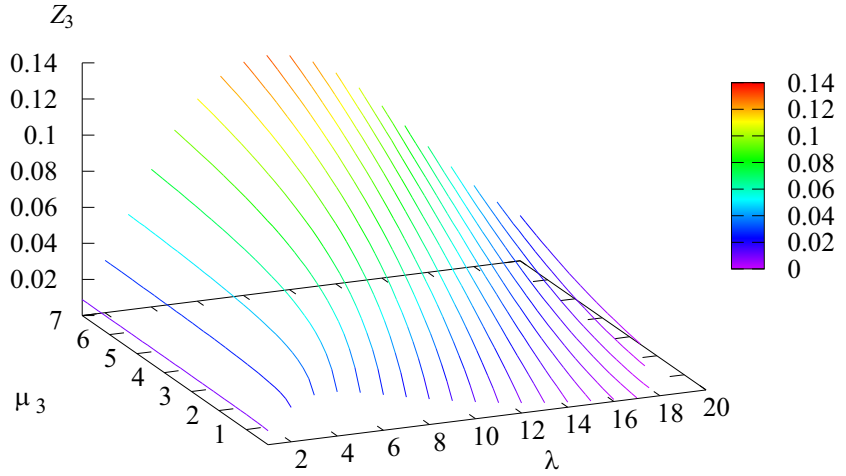

Figure 11. Dependence of the fraction Z_3 of customers that receive service immediately upon arrival by the third server.

The first server immediate access probability, Z_1, is highest (≈ 0.9) under low traffic loads. Then, it drops steadily as the arrival rate λ increases. It is unaffected by changes to the slowest server speed μ_3. The second server immediate access share, Z_2, peaks at lower traffic as server 1 fills up. Then, it rapidly falls to 0 under heavy traffic loads. It shows slight variations based on the value of μ_3. The third server immediate access share, Z_3, stays exceptionally low, peaking at only around 0.14. It is highly sensitive to the third server's own rate μ_3 and completely vanishes to 0 when system saturation occurs.

The results presented above were obtained under the assumption that the service times at all available servers follow a hyperexponential distribution, with a coefficient of variation greater than 4. The impact of the service time distribution deserves special attention. Moreover, service times across different servers can have coefficients of variation that are greater than, equal to, or less than 1. Here, we restrict ourselves to providing brief information on the system's performance measures for several values of the arrival rate λ and three types of service time distributions across all servers: the hyper-

exponential distribution considered above (denoted here as $Hexp$), an exponential distribution with a coefficient of variation equal to 1 (denoted here as Exp), and a second-order Erlang distribution with a coefficient of variation equal to 0.5 (denoted here as Erl). The mean service rates in the corresponding servers are assumed to be as defined above for the hyper-exponential distribution: $\mu_1 = 10$, $\mu_2 = 7.5$, and $\mu_3 = 1$. The results are presented in Table 1.

Table 1. System performance metrics as a function of the arrival rate λ and type of service time distribution.

λ, type	L_{buffer}	$\bar{\mu}$	N_{busy}	P_{imm}	W_{buffer}	$P_{empty-buffer}$
5, $Hexp$	0.116051	8.67578	0.838513	0.877504	0.023210	0.955216
5, Exp	0.0479343	8.67192	0.842314	0.886787	0.009587	0.969402
5, Erl	0.0364282	8.66985	0.843763	0.890313	0.007286	0.97432
10, $Hexp$	2.31564	8.45505	1.73046	0.474704	0.231564	0.68319
10, Exp	1.01211	8.46721	1.71686	0.465909	0.101211	0.711302
10, Erl	0.78335	8.47039	1.7131	0.466123	0.078335	0.729016
15, $Hexp$	30.2942	8.47426	2.48722	0.104348	2.019616	0.261821
15, Exp	21.901	8.48415	2.46901	0.0677035	1.460066	0.244084
15, Erl	20.3252	8.48765	2.46251	0.0580588	1.355011	0.240681

Table 1 demonstrates how the customer arrival rate (λ) and the type of service time distribution impact the key performance metrics of the queueing system. The main insights and structural patterns from the data are highlighted below:

i) The impact of the arrival rate (λ): Nonlinear queue growth: As λ scales from 5 to 15, the mean queue length (L_{buffer}) and the average waiting time (W_{buffer}) explode nonlinearly. System saturation: At $\lambda = 15$, the system approaches its total boundary capacity ($\mu_1 + \mu_2 + \mu_3 = 18.5$), triggering an asymptotic performance degradation. Server utilization: The average number of busy servers (N_{busy}) climbs predictably toward the absolute system limit of three devices. Plunging availability: The probability of immediate service access (P_{imm}) and the probability of an empty buffer ($P_{empty-buffer}$) drop sharply to minimal values under heavy workloads.

ii) The impact of service time variance (volatility): The experiment compares three distinct distributions with identical mean service rates but different coefficients of variation (c_{var}): Hexp ($c_{var} > 4$), Exp ($c_{var} = 1$), and Erl ($c_{var} = 0.5$). Volatility penalty: Higher irregularity in service durations (moving from Erl $\rightarrow$ Exp $\rightarrow$ Hexp) systematically compromises system performance. Queue accumulation: At a medium load of $\lambda = 10$, the queue length L_{buffer} is only 0.78335 for the stable Erlang distribution (Erl), rises to 1.01211 for Exponential (Exp), and spikes up to 2.31564 for Hyperexponential (Hexp). Inflated waiting times: The average waiting delay (W_{buffer}) for highly unpredictable workloads (Hexp) is nearly three times longer than that of well-regulated schedules (Erl) under moderate workloads. Stable throughput: The averaged service rate ($\bar{\mu}$) remains remarkably consistent across different distributions within the same arrival tier.

Key takeaway: Table 1 provides concrete numerical verification of a fundamental queueing principle—system congestion is driven not just by raw traffic volume but heavily by the mathematical

volatility and burstiness of the underlying processes.

8.2. Economic cost function and optimization framework

Let us now demonstrate the possibility of application of the obtained results for managerial goals, in particular, to determination of the optimal configuration of the queueing system. To evaluate the trade-off between the requirements to customer service quality and the operational expenses of server capacities, let us introduce an economic cost function. The objective function, which represents the expected total charge paid by the system hourly, is formulated as follows:

$$E = E(\lambda, \mu_3) = C_{wait} W_{buffer} + C_{serv-cost} \mu_3,$$

where

- W_{buffer} is the average waiting time of a customer in the buffer;
- $C_{wait} = 20$ is the penalty cost per unit of time a customer spends waiting in the queue. This coefficient penalizes excessive delays and queue accumulation;
- $C_{serv-cost} = 1$ is the cost coefficient associated with maintaining the service rate μ_3 of the third server per unit of time;
- μ_3 is the control parameter under optimization, evaluated within the technological bounds $\mu_3 \in [\max\{0.2, \lambda - \mu_1 - \mu_2\}, 7]$ with an incremental step of 0.2.

For each given mean arrival rate $\lambda \in \{1, 2, \ldots, 20\}$, the global optimization task is to find the optimal service capacity μ_3^* that minimizes the economic criterion under the ergodicity constraint:

$$E^{opt} = \min_{\mu_3} E(\lambda, \mu_3) \quad \text{subject to} \quad \lambda < \sum_{n=1}^{3} \mu_n.$$

It is worth noting that in this optimization scenario, the service rates of the primary servers (μ_1 and μ_2) are kept fixed, and only the capacity of the slowest server (μ_3) is treated as a decision variable. This setup is strongly motivated by real-world operational environments where high-priority, efficient servers represent an existing baseline infrastructure that operates at a fixed maximum capacity. Managers typically cannot easily alter these core assets. Instead, the lowest-priority server serves as a flexible, scalable backup unit or an on-demand resource deployed specifically to handle peak traffic overflows. Thus, optimizing μ_3 captures the critical managerial decision of sizing auxiliary backup capacity to balance waiting time penalties against incremental operational costs. Although a joint multivariable optimization over all server rates $(\mu_1, \ldots, \mu_N)$ can be straightforwardly implemented using the analytical performance metrics derived in Section 7, the single-variable optimization presented here serves as a clear, traceable illustration of the fundamental economic trade-offs and capacity thresholds inherent in the system.

Table 2 shows the optimal value μ_3^* of the mean service rate μ_3 and minimum economic cost function values under the fixed choice 10 and 7.5 of the rates μ_1 and μ_2, correspondingly, for different mean arrival rates λ.

Table 2. The optimal value μ_3^* and minimum economic cost function values for different mean arrival rates λ.

λ	μ_3^*	E^{opt}	λ	μ_3^*	E^{opt}
1	0.2	0.205846	11	3.0	6.748900
2	0.2	0.251449	12	4.2	8.150660
3	0.2	0.372313	13	5.4	9.564690
4	0.2	0.587805	14	6.6	10.977200
5	0.2	0.917595	15	7.0	12.508800
6	0.2	1.393550	16	7.0	14.633600
7	0.2	2.067100	17	7.0	17.668100
8	0.4	2.984860	18	7.0	22.070800
9	1.0	4.107940	19	7.0	28.578500
10	2.0	5.384040	20	7.0	38.498600

The results presented in Table 2 can be briefly summarized as follows.

When the traffic is low ($\lambda = 1, \ldots, 7$), the arrivals are easily handled by the faster servers. The slowest server is kept at its minimum technological bound ($\mu_3^* = 0.2$) to save operational costs.

When the traffic is medium ($\lambda = 8, \ldots, 14$), as arrival intensity grows, queue starts to build up. The system systematically increases μ_3^* from 0.4 to 6.6 to avoid heavy waiting penalties ($C_{wait} = 20$).

When the traffic becomes high ($\lambda = 15, \ldots, 20$), the arrival rate approaches or exceeds primary capacity. The third server is forced to run at its absolute maximum capacity ($\mu_3^* = 7$).

The economic cost (E^{opt}) scales nonlinearly. It grows slowly during low traffic but accelerates significantly when the system runs at full capacity.

Table 2 presents the optimal service rate μ_3^* and the corresponding minimum economic cost function values for various mean arrival rates λ under the baseline cost coefficients $C_{wait} = 20$ (the waiting time penalty per unit of time) and $C_{serv-cost} = 1$ (the operational cost coefficient for maintaining the third server's capacity). To highlight how these optimal values depend on the cost structure, we additionally analyze the system behavior across different values of the cost ratio $r = C_{wait}/(C_{serv-cost})$. For this sensitivity analysis, the capacity cost coefficient is kept constant ($C_{serv-cost} = 1$), and the ratio r is scaled by setting C_{wait} to 5, 10, 20, 40, and 50, respectively.

Table 3 presents the optimal value μ_3^* and minimum economic cost function values for different mean arrival rates λ and ratio r.

Table 3. The optimal value μ_3^* and minimum economic cost function values for different mean arrival rates λ and ratio r.

λ	r	μ_3^*	E^{opt}	λ	r	μ_3^*	E^{opt}
5	5	0.2	0.379399	10	40	3.8	7.99195
5	10	0.2	0.558797	10	50	4.6	8.94849
5	20	0.2	0.917595	15	5	4	7.15889
5	40	0.2	1.63519	15	10	5.8	9.52913
5	50	0.4	1.96784	15	20	7	12.5088
10	5	0.2	1.75252	15	40	7	18.0176
10	10	0.6	3.26321	15	50	7	20.772
10	20	2	5.38404				

Key observations and insights are as follows.

i) Low traffic scenario ($\lambda = 5$): When arrival intensity is low, the faster primary servers ($\mu_1 = 10, \mu_2 = 7.5$) easily handle the incoming workload. Consequently, even as the delay penalty r increases from 5 to 40, the system keeps the third server at its absolute minimum technological bound ($\mu_3^* = 0.2$) to save operational costs. It is only when the penalty becomes extreme ($r = 50$) that the system minimally increases the capacity to $\mu_3^* = 0.4$.

ii) Medium traffic scenario ($\lambda = 10$): As the load increases, queue accumulation begins to threaten the system with higher delay charges. We see a direct correlation between the severity of the waiting penalty (r) and the optimal backup capacity: At $r = 5$, the server stays at 0.2, but as r increases to 50, the system systematically scales up μ_3^* from 0.2 to 4.6. This suppresses buffer wait times to avoid steep financial punishments.

iii) High traffic scenario ($\lambda = 15$): When the arrival flow approaches the limits of the primary servers, the infrastructure faces near-saturation. At a low penalty ($r = 5$), the system aggressively compromises on waiting times to keep the server speed low ($\mu_3^* = 4$). However, as the cost ratio reaches $r = 20$ and above, the system is forced to run the third server at its maximum capacity ($\mu_3^* = 7.0$) to maintain ergodicity (stability) and prevent an infinite queue explosion.

iv) Managerial takeaway is: Table 3 successfully demonstrates a core managerial trade-off: Auxiliary backup capacity (μ_3) should act as a flexible asset. When waiting delays are inexpensive, managers should minimize backup utilization to save costs; however, as customer time becomes more valuable or as traffic spikes, scaling up backup processing speeds is financially necessary to avoid steep penalties.

8.3. Larger values of N

The previous results illustrate the system's performance for $N = 3$ servers. To demonstrate the feasibility of the proposed algorithms for larger systems, we briefly analyze configurations with higher values of N.

First, let us fix $N = 5$ with hyperexponential service time distributions specified by the vectors $\beta^{(n)} = \beta = (0.1, 0.9)$ for $n = \overline{1,5}$.

The matrix $S^{(1)}$ is given by $S^{(1)} = \begin{pmatrix} -2.4 & 0 \\ 0 & -21.6 \end{pmatrix}$. This yields a mean service rate of $\mu_1 = 12$ at the first server, with a squared coefficient of variation of approximately 4.55. The matrices $S^{(n)}$ for $n = \overline{2,5}$ are obtained by scaling $S^{(1)}$ to achieve the mean service rates $\mu_2 = 10$, $\mu_3 = 7$, $\mu_4 = 4$, and $\mu_5 = 1$. The squared coefficient of variation of service times by all servers is also approximately 4.55.

Table 4 demonstrates how the system's key performance metrics (L_{buffer}, $\bar{\mu}$, N_{busy}, P_{imm}, W_{buffer}, and $P_{empty-buffer}$) change as the mean arrival rate, λ, increases from 20 to 30 under this five- server configuration. As seen from Table 4, the buffer size increases sharply from 2.6469 to 66.691, and the mean waiting times grow nonlinearly from 0.1323 to 2.223. Simultaneously, the average number of busy servers rises from 2.9994 to 4.4517, and the immediate entry probability plunges from 0.5202 to 0.0748, and the empty buffer probability drops from 0.6941 to 0.1794. Meanwhile, the average service rate remains stable around 9.10–9.18.

Table 4. System performance metrics for $N = 5$ as a function of the arrival rate λ.

λ	L_{buffer}	$\bar{\mu}$	N_{busy}	P_{imm}	W_{buffer}	$P_{empty-buffer}$
20	2.6469	9.1822	2.9994	0.5203	0.1323	0.6941
21	3.6222	9.1591	3.1542	0.463	0.1725	0.6449
22	4.9412	9.1411	3.3061	0.4062	0.2246	0.5933
23	6.7309	9.1275	3.4554	0.3512	0.2926	0.5402
24	9.1688	9.1176	3.6024	0.2988	0.382	0.4862
25	12.5054	9.1107	3.7474	0.2501	0.5002	0.4323
26	17.1024	9.1063	3.8908	0.2057	0.6578	0.379
27	23.5064	9.104	4.0327	0.1661	0.8706	0.3269
28	32.6022	9.1033	4.1733	0.1313	1.1644	0.2762
29	45.9576	9.104	4.313	0.1011	1.5847	0.2271
30	66.6911	9.106	4.4517	0.0748	2.223	0.1794

As the arrival rate λ approaches the total system service capacity 34, the primary fast servers saturate. This triggers the activation of slower backup servers, causing incoming customers to experience longer delays and a much higher probability of waiting in the buffer queue.

Table 4 presents the system performance metrics for $N = 5$ as a function of the arrival rate λ for the case where the correlation coefficient c_{cor} of successive interarrival times is relatively small, equal to 0.2. To underline the importance of accounting for the correlation and variation in the arrival process, we also present Table 5, which gives the values of the same performance metrics when the arrival process is assumed to be a stationary Poisson process with the same arrival rate and the correlation coefficient c_{cor} equal to zero and the squared coefficient of variation c_{var}^2 equal to 1.

Table 5. System performance metrics for $N = 5$ as a function of the arrival rate λ in the case of stationary Poisson arrival flow.

λ	L_{buffer}	$\bar{\mu}$	N_{busy}	P_{imm}	W_{buffer}	$P_{empty-buffer}$
20	0.7501	9.3954	2.8865	0.7604	0.0375	0.8348
21	1.0067	9.3475	3.055	0.7204	0.0479	0.8002
22	1.3404	9.3063	3.2202	0.6775	0.0609	0.7615
23	1.7749	9.271	3.3823	0.632	0.0772	0.719
24	2.3434	9.241	3.5413	0.584	0.0976	0.6724
25	3.0938	9.2156	3.6973	0.5337	0.1237	0.6219
26	4.0978	9.1942	3.8505	0.4812	0.1576	0.5675
27	5.4680	9.1765	4.0012	0.4267	0.2025	0.5093
28	7.392	9.1617	4.1494	0.3704	0.264	0.4474
29	10.2072	9.1496	4.2955	0.3123	0.352	0.3817
30	14.5886	9.1399	4.4396	0.2527	0.4863	0.3124

Comparing the values of the most important performance measures such as L_{buffer}, W_{buffer}, P_{imm}, $P_{empty-buffer}$, given in Tables 4 and 5, it can be immediately concluded that correlation and variation have a significant impact. Neglecting the correlation and variation coefficients (by using a stationary Poisson process to model the arrival flow) leads to an overly optimistic forecast of system performance. This is caused by the higher irregularity of customer arrivals in flows with positive correlation and/or large variance of interarrival times. High positive correlation implies that periods when customers arrive frequently (and the system becomes overloaded) alternate with periods when customers arrive rarely, and server starvation occurs. Large variance has a similar effect: Some intervals between the arrivals are short, and congestion can occur. Some intervals are long, which may lead to server downtime.

Note that with an increase in correlation and/or variance of interarrival times, the discrepancies between the corresponding performance metrics can become highly significant.

Regarding the computational efficiency, the running time required to obtain the stationary distributions and performance measures heavily depends on the system load, defined as the ratio $\lambda / \sum_{n=1}^{N} \mu_n$. This execution time increases significantly as the load ratio approaches 1.

It is worth noting that in this five-server case, where the phase-type distribution of service times and the arrival process both have state spaces of cardinality 2, the dimension of the block $\mathbf{Q}_{0,0}$ is $486 = 2 \times 3^5$, whereas the dimension of the critical blocks $(\mathbf{Q}^0, \mathbf{Q}^+, \mathbf{Q}^-)$ is $64 = 2 \times 2^5$.

To evaluate the execution time, we performed calculations for a five-server system using 11 values of λ (from 20 to 30 in steps of 2) and 9 values of μ_5 (from 0.2 to 1.8 in steps of 0.2), resulting in 209 combinations in total. The total computation time on a laptop (11th Gen Intel Core i7-1165G7 @ 2.80GHz, 16GB RAM) using Wolfram Mathematica 12.2 was 422 seconds, that is, about 2 seconds per one combination. The iterative computation of the stationary probability vectors $\boldsymbol{\pi}_i$ was terminated when the norm of a vector became less than 10^{-15}.

Remark 3. It is worth noting that the correctness of the computations for the stationary distribution of the system states and performance measures in this example was validated via a discrete-event simulation implemented in Wolfram Mathematica 12.2. The simulation time required to match the

analytical results up to four decimal places was, on average, four minutes per parameter combination. This confirms the high efficiency of the analytical approach compared to computer simulations for relatively small values of the number of servers N, as the computation time in this example with $N = 5$ using the analytical results is more than 120 times faster than the simulation time.

To evaluate the scalability further, the experiment is extended to an $N = 8$ server system by changing μ_5 to 3 and introducing three additional servers with rates $\mu_6 = 2$, $\mu_7 = 1.5$, and $\mu_8 = 1$.

In this scenario, the size of the block $\mathbf{Q}_{0,0}$ expands to $13122 = 2 \times 3^8$, and the blocks $\mathbf{Q}^0$, Q^+, and $\mathbf{Q}^-$ reach a dimension of $512 = 2 \times 2^8$. Fixing $\lambda = 25$ yields a system load ratio of approximately 0.617. The computational time for a single parameter combination on the same notebook is 157.95 seconds (approximately 2.63 minutes). The sharp increase in computation time compared to the $N = 5$ case is driven by the rapid growth in the size of the generator blocks when N increases. Consequently, the efficiency of analytical methods compared to simulation decreases as N grows. For large values of N analytical methods may become computationally intractable.

The computed values of the performance characteristics of the system are as follows.

- The average number of customers in the buffer L_{buffer} is 2.71608.

- The average waiting time W_{buffer} is 0.10864.

- The variance of waiting time is 0.0569019.

- The average number of customers in the system L_{system} is 7.66208.

- The average customer sojourn time is 0.306483.

- The average service time is 0.19784.

- The average service rate $\bar{\mu}$ is 8.07779.

- The average number of busy servers N_{busy} is 4.94599.

- The probability that a customer will start service immediately upon arrival P_{imm} is 0.5737.

- The probability that the buffer is empty $P_{empty-buffer}$ is 0.718489.

- The probability that all servers are idle $P_0^{(serv)}$ is 0.0828338.

- The probability that all servers are idle at an arbitrary arrival moment $P_0^{(arr)}$ is 0.0105509.

The individual characteristics of the performance of the servers are given in Table 6.

Table 6. The characteristics of the performance of the individual servers.

Server index n	Output rate φ_n	Share F_n	Idle probability $P_0^{(n)}$	Immediate access probability Z_n
1	7.7245	0.30898	0.356292	0.190558
2	6.04722	0.241889	0.395278	0.140414
3	4.14426	0.16577	0.407963	0.0919512
4	2.44778	0.0979112	0.388055	0.0536961
5	1.81786	0.0727142	0.394048	0.0384798
6	1.23888	0.049555	0.380561	0.0259565
7	0.933976	0.037359	0.377349	0.0192111
8	0.645541	0.0258217	0.354459	0.0134552
all	25	1	-	0.5737

Table 6 displays the individual performance indicators for each of the $N = 8$ heterogeneous devices at a fixed input flow rate $\lambda = 25$. This table is of crucial theoretical and practical importance for the entire article, as it experimentally confirms the mathematical properties of a queueing system with the discipline of ordered device selection (ordered hunting) with the priority given to the fastest servers.

The data clearly demonstrate the effect of this service discipline. The first servers (with the smallest indices $n = 1, 2$) take on the main load. Their output flow intensity ($\varphi_1 = 7.7245$, $\varphi_2 = 6.04722$) and the share of served customers ($F_1 \approx 31\%$, $F_2 \approx 24\%$) are the highest in the system. As the server index n increases, the values of the output rate φ_n and the share F_n decrease monotonically and rapidly. The last, eighth server, processes only $\approx 2.5\%$ of the total traffic volume ($F_8 = 0.0258$). This proves that low-priority (slower) servers are activated predominantly during peak periods, when all the previous ones are busy.

The Z_n indicator (the probability that an arriving customer will immediately be served on the nth server) drops quickly: from 0.1905 for the first server to 0.0134 for the eighth. Taken together, all the Z_n indicators yield a total probability of immediate entry $P_{imm} = 0.5737$.

Note the column of server idle probabilities $P_0^{(n)}$, $n = \overline{1,8}$. Contrary to expectations, they do not increase monotonically from 1 to 8. For the first server, $P_0^{(1)} = 0.356$, and then, the idle probability increases to 0.407 for the third server, but by the eighth server, it drops again to 0.354. Why is this happening? This is a strict consequence of heterogeneity. The first servers are the fastest, complete their work quickly, and are freed frequently. In contrast, the last servers are very slow. Due to the low service speed, if a customer reaches the server with index 8, this server will be occupied by this customer for a very long time. This drastically reduces the likelihood of a server being idle, despite its infrequent activation.

It is interesting to note the following. The marginal load of the nth server is $\rho_n = \lambda F_n/\mu_n$, $n = \overline{1,N}$. It can be seen that the values presented in Table 6, of the probability $P_0^{(n)}$ that the nth server is idle coincide with the number $1 - \rho_n$, which is the probability of the idle state of a single-server queueing system with the stationary Poisson arrival process having the rate λF_n and average service time μ_n^{-1}.

9. Conclusions

9.1. Main findings

In this paper, we have provided a rigorous steady-state analysis of a multi-server queueing system with a Markovian arrival process, heterogeneous servers, and phase-type service time distributions under an ordered hunting discipline. By applying matrix-analytic methods, we successfully constructed the explicit block-tridiagonal structure of the infinitesimal generator and derived a highly tractable exact ergodicity condition. The stationary distribution was obtained in a mathematically elegant matrix-geometric form, enabling the derivation of closed-form Laplace–Stieltjes transforms for both customer waiting and sojourn times. Our numerical experiments revealed several critical insights into the behavior of heterogeneous systems.

The impact of correlation: Neglecting traffic correlation (e.g., by approximating the influx via a stationary Poisson process) leads to an overly optimistic forecast of system performance, underestimating the average queue size and waiting times due to the high arrival irregularity caused by the positive correlation.

The impact of service time variance (volatility): Higher irregularity and variance in service durations systematically compromise system performance, even when the mean service rates remain perfectly identical. Progressing from stable, low-variance service schedules (such as the Erlang distribution) to highly unpredictable ones (such as the hyperexponential distribution) triggers an acute volatility penalty, nonlinearly spiking queue lengths and inflating average customer waiting delays in the considered example by nearly three times under moderate workloads. This confirms the fundamental queueing principle that systemic congestion is profoundly driven by the volatility of the processing durations.

The heterogeneity paradox: Slower backup servers, despite being rarely activated under low workloads, exhibit high occupancy times because they process customers much slower, resulting in unexpectedly low idle probabilities.

Computational efficiency: The proposed analytical approach proved to be computationally efficient for relatively small values of N. Validation via discrete-event simulation confirmed that our exact mathematical framework achieves identical precision (up to four decimal places) while reducing the required computation time for $N = 5$ by a factor of more than 120.

9.2. Limitations of the study

Although the developed framework is highly versatile, it possesses certain limitations that stem from the underlying modeling assumptions.

Infinite buffer capacity: The model assumes an infinite-capacity buffer. In physical infrastructure (e.g., 5G/6G finite router buffers), buffer overflows inevitably trigger packet drops or losses, which are not captured here.

Absolute server reliability: The heterogeneous servers are assumed to be perfectly reliable and permanently available. The framework does not account for random server breakdowns, scheduled maintenance, or vacation periods.

Fixed indexing policy: Although the algorithms accommodate any fixed routing order, the allocation discipline itself remains static during the system's operation.

9.3. Directions for future work

Driven by these limitations and the structural insights obtained in this study, future research can be expanded in the following vital directions.

Finite buffers and customer impatience: Incorporate finite buffer thresholds and reneging behavior (impatient customers) to better match cloud-computing and call-center dynamics. This extension breaks the level-independence of the quasi-birth-and-death (QBD) process but can be resolved using the theory of asymptotically quasi-Toeplitz Markov chains; see [54] or [55].

Customers retrials: Account for the possibility of the absence of the buffer and repeated attempts of customer to enter the service.

Server unreliability: Introduce server breakdowns, repairs, or setup delays to model realistic lifecycle constraints across heterogeneous hardware clusters.

Algorithmic optimization of server indexing: Utilize the performance metrics established in this paper as a foundation to solve the optimal server indexing sequence problem, aiming to maximize systemic throughput while minimizing hourly operational overhead and non-linear energy costs.

Author contributions

C. D'Apice: Conceptualization, methodology, validation, investigation, writing, original draft preparation, writing, review and editing, project administration; A. N. Dudin: Conceptualization, methodology, formal analysis, investigation, writing, original draft preparation, writing, review and editing, project administration; S. A. Dudin: Conceptualization, methodology, software, formal analysis, investigation, writing, original draft preparation, supervision; R. Manzo: Conceptualization, software, validation, formal analysis, investigation, writing, original draft preparation, writing, review and editing. All authors have read and approved the final version of the manuscript for publication.

Use of Generative-AI tools declaration

The authors declare they have not used Artificial Intelligence (AI) tools in the creation of this article.

Conflict of interest

Rosanna Manzo is an academic editor for AIMS Mathematics and was not involved in the review or the decision to publish this article. All authors declare no conflicts of interest in this paper.

References

1. M. F. Neuts, *Matrix-geometric solutions in stochastic models*, North Chelmsford: Courier Corporation, 1994.
2. S. Asmussen, *Applied probability and queues*, 2Eds., New York: Springer, 2003. https://doi.org/10.1007/b97236
3. P. Buchholz, J. Kriege, I. Felko, *Input modeling with phase-type distributions and Markov models: theory and applications*, Cham: Springer, 2014. https://doi.org/10.1007/978-3-319-06674-5

4. C. A. O'Cinneide, Characterization of phase-type distributions, *Stoch. Models*, **6**, (1990), 1–57. https://doi.org/10.1080/15326349908807134
5. C. A. O'Cinneide, Phase-type distributions: open problems and a few properties, *Stoch. Models*, **15** (1999), 731–757. https://doi.org/10.1080/15326349908807560
6. A. Horvath, M. Telek, *Phase type distributions: theory and application*, Hoboken: John Wiley & Sons, 2024. https://doi.org/10.1002/9781119419808
7. D. M. Lucantoni, New results on the single server queue with a batch Markovian arrival process, *Stoch. Models*, **7** (1991), 1–46. https://doi.org/10.1080/15326349108807174
8. S. R. Chakravarthy, The batch Markovian arrival process: A review and future work, In: *Advance in probability and stochastic process*, NewJersey: Notable Publications Inc., 2000, 21–49.
9. S. R. Chakravarthy, *Introduction to Matrix-matrix-analytic methods in queues 1: Analytical and simulation approach–Basics*, New York: London and John Wiley and Sons, 2022. https://doi.org/10.1002/9781394165421
10. A. N. Dudin, V. I. Klimenok, V. M. Vishnevsky, *The theory of queuing systems with correlated flows*, Cham: Springer, 2020. https://doi.org/10.1007/978-3-030-32072-0
11. M. G. Bernal, R. E. Lillo, J. Ramirez-Cobo, Call center data modeling: A queueing science approach based on Markovian arrival processe, *Qual. Technol. Quant. M.*, **22** (2025), 631–658. https://doi.org/10.1080/16843703.2024.2371715
12. A. N. Dudin, S. A. Dudin, R. Manzo, L. Rarità, Analysis of multi-server priority queueing system with hysteresis strategy of servers reservation and retrials, *Mathematics*, **10** (2022), 3747. https://doi.org/10.3390/math10203747
13. C. D'Apice, A. N. Dudin, S. A. Dudin, R. Manzo, Priority queueing system with many types of requests and restricted processor sharing, *J. Ambient Intell. Human. Comput.*, **14** (2023), 12651–12662. https://doi.org/10.1007/s12652-022-04233-w
14. C. D'Apice, M. P. D'Arienzo, A. N. Dudin, R. Manzo, Admission control in priority queueing system with servers reservation and temporal blocking admission of low priority users, *IEEE Access*, **11** (2023), 44425–44443. https://doi.org/10.1109/ACCESS.2023.3273148
15. C. D'Apice, M. P. D'Arienzo, A. N. Dudin, R. Manzo, Optimal hysteresis control via a queueing system with two heterogeneous energy-consuming servers, *Mathematics*, **11** (2023), 4515. https://doi.org/10.3390/math11214515
16. A. N. Dudin, S. A. Dudin, R. Manzo, L. Rarità, Queueing system with batch arrival of heterogeneous orders, flexible limited processor sharing and dynamical change of priorities, *AIMS Math.*, **9** (2024), 12144–12169. https://doi.org/10.3934/math.2024593
17. C. D'Apice, A. N. Dudin, O. S. Dudina, R. Manzo, Analysis of a queueing system with dynamic rating-dependent arrival process and price of service, *Mathematics*, **12** (2024), 1101. https://doi.org/10.3390/math12071101
18. S. A. Dudin, A. N. Dudin, R. Manzo, L. Rarità, Analysis of semi-open queueing network with correlated arrival process and multi-server nodes, *Oper. Res. Forum*, **5** (2024), 99. https://doi.org/10.1007/s43069-024-00383-z

19. C. D'Apice, A. N. Dudin, S. A. Dudin, R. Manzo, *Study of a semi-open queueing network with hysteresis control of service regimes*, *AIMS Math.*, **10** (2025), 3095–3123. https://doi.org/10.3934/math.2025144

20. B. Sun, M. H. Lee, S. A. Dudin, A. N. Dudin, Analysis of multiserver queueing system with opportunistic occupation and reservation of servers, *Math. Probl. Eng.*, **2014** (2014), 178108. https://doi.org/10.1155/2014/178108

21. G. Casale, E. Z. Zhang, E. Smirni, Trace data characterization and fitting for Markov modeling, *Perform. Evaluation*, **67** (2010), 61–79. https://doi.org/10.1016/j.peva.2009.09.003

22. P. Buchholz, P. Kemper, J. Kriege, Multi-class Markovian arrival processes and their parameter fitting, *Perform. Evaluation*, **67** (2010), 1092–1106. https://doi.org/10.1016/j.peva.2010.08.006

23. P. Buchholz, A. Panchenko, A two-step EM algorithm for MAP fitting, In: *Computer and information sciences-ISCIS 2004*, Berlin: Springer, 2004, 217–227. https://doi.org/10.1007/978-3-540-30182-0_23

24. H. Okamura, T. Dohi, Mapfit: An R-based tool for PH/MAP parameter estimation, In: *Quantitative evaluation of systems*, Cham: Springer, 2015, 105–112. https://doi.org/10.1007/978-3-319-22264-6_7

25. H. Okamura, T. Dohi, K. S. Trivedi, Markovian arrival process parameter estimation with group data, *IEEE/ACM T. Network.*, **17** (2009), 1326–1339. https://doi.org/10.1109/TNET.2008.2008750

26. C. Li, J. J. Zheng, H. Okamura, T. Dohi, Markovian arrival process parameter estimation of quasi-birth-death queueing systems with utilization data, 2026, arXiv: 2607.01914. https://doi.org/10.48550/arXiv.2607.01914

27. R. Wang, G. Casale, A. Filieri, Estimating multiclass service demand distributions using Markovian arrival processes, *ACM T. Model. Comput. S.*, **33** (2023), 1–26. https://doi.org/10.1145/3570924

28. M. Brazenas, G. Horvath, M. Telek, Parallel algorithms for fitting Markov arrival processes, *Perform. Evaluation*, **123** (2018), 50–67. https://doi.org/10.1016/j.peva.2018.05.001

29. H. Gumbel, Waiting lines with heterogeneous servers, *Oper. Res.*, **8** (1960), 504–511. https://doi.org/10.1287/opre.8.4.504

30. J. H. Kim, H. S. Ahn, R. Righter, Managing queues with heterogeneous servers, *J. Appl. Probab.*, **48** (2011), 435–452. https://doi.org/10.1239/jap/1308662637

31. C. J. Ancker, A. V. Cafarian, Queuing with reneging and multiple heterogeneous servers, *Naval Research Logistics Quarterly*, **10** (1963), 125–149. https://doi.org/10.1002/nav.3800100112

32. V. Saglam, A. Shahbazov, Minimizing loss probability in queuing systems with heterogeneous servers, *Iran. J. Sci. Technol. A*, **31** (2007), 199–206.

33. D. Efrosinin, N. Stepanova, J. Sztrik, A. Plank, Approximations in performance analysis of a controllable queueing system with heterogeneous servers, *Mathematics*, **8** (2020), 1803. https://doi.org/10.3390/math8101803

34. A. Gregosiewicz, E. Ratajczyk, L. Stepien, Modeling and simulation of heterogeneous multiserver queues with impatient customers, *Adv. Sci. Technol. Res. J.*, **19** (2025), 464–477. https://doi.org/10.12913/22998624/210352
35. M. Armony, Dynamic routing in large-scale service systems with heterogeneous servers, *Queueing Syst.*, **51** (2005), 287–329. https://doi.org/10.1007/s11134-005-3760-7
36. M. Armony, A. R. Ward, Fair dynamic routing in large-scale heterogeneous-server systems, *Oper. Res.*, **58** (2010), 624–637. https://doi.org/10.1287/opre.1090.0777
37. J. Khamse-Ashari, I. Lambadaris, G. Kesidis, B. Urgaonkar, Y. Q. Zhao, An efficient and fair multi-resource allocation mechanism for heterogeneous servers, *IEEE T. Parall. Distr.*, **29** (2018), 2686-2699. https://doi.org/10.1109/TPDS.2018.2841915
38. N. Li, D. A. Stanford, Multi-server accumulating priority queues with heterogeneous servers, *Eur. J. Oper. Res.*, **252** (2016), 866–878. https://doi.org/10.1016/j.ejor.2016.02.010
39. A. Z. Melikov, L. A. Ponomarenko, E. V. Mekhbaliyeva, Analyzing models of systems with heterogeneous servers, *Cybern Syst Anal*, **56** (2020), 89–99. https://doi.org/10.1007/s10559-020-00224-x
40. L. Sakalauskas, L. Kaklauskas, R. Macaitiene, Stalling in queuing systems with heterogeneous channels, *Appl. Sci.*, **14** (2024), 773. https://doi.org/10.3390/app14020773
41. V. V. Rykov, Monotone control of queueing systems with heterogeneous servers, *Queueing Syst.*, **37** (2001), 391–403. https://doi.org/10.1023/A:1010893501581
42. V. Rykov, D. Efrosinin, Optimal control of queueing systems with heterogeneous servers, *Queueing Syst.*, **46** (2004), 389–407. https://doi.org/10.1023/B:QUES.0000027992.91461.1e
43. D. V. Efrosinin, M. P. Farkhadov, N. V. Stepanova, Study of a controllable queueing system with unreliable heterogeneous servers, *Autom. Remote Control*, **79** (2018), 265–285. https://doi.org/10.1134/S0005117918020066
44. D. Efrosinin, N. Stepanova, J. Sztrik, Algorithmic analysis of finite-source multi-server heterogeneous queueing systems, *Mathematics*, **9** (2020), 2624. https://doi.org/10.3390/math9202624
45. D. V. Efrosinin, V. V. Rykov, Numerical study of the optimal control of a system with heterogeneous servers, *Automat. Remote Control*, **64** (2003), 302–309. https://doi.org/10.1023/A:1022271316352
46. D. V. Efrosinin, V. V. Rykov, On performance characteristics for queueing systems with heterogeneous servers, *Automat. Remote Control*, **69** (2008), 61–75. https://doi.org/10.1134/S0005117908010074
47. D. Efrosinin, L. Breuer, Threshold policies for controlled retrial queues with heterogeneous servers, *Ann. Oper. Res*, **141** (2006), 139–162. https://doi.org/10.1007/s10479-006-5297-5
48. M. B. Combe, Queueing models with dependence structure, PhD Thesis, Tilburg University, 1995.
49. Q. M. He, A. S. Alfa, Space reduction for a class of multidimensional Markov chains: A summary and some applications, *INFORMS J. Comput.*, **30** (2018), 1–10. https://doi.org/10.1287/ijoc.2017.0759

50. A. Graham, *Kronecker products and matrix calculus with applications*, Mineola: Courier Dover Publications, 2018.

51. G. Latouche, V. A. Ramaswami, A logarithmic reduction algorithm for quasi-birth-death processes, *J. Appl. Probab.*, **30** (1993), 650–674. https://doi.org/10.2307/3214773

52. H. Baumann, W. Sandmann, Numerical solution of level dependent quasi-birth-and-death processes, *Procedia Computer Science*, **1** (2010), 1561–1569. https://doi.org/10.1016/j.procs.2010.04.175

53. S. A. Dudin, O. S. Dudina, Call center operation model as a $MAP/PH/N/R - N$ system with impatient customers, *Probl. Inf. Transm.*, **47** (2011), 364–377. https://doi.org/10.1134/S0032946011040053

54. V. I. Klimenok, A. N. Dudin, Multidimensional asymptotically quasi-Toeplitz Markov chains and their application in queueing theory, *Queueing Syst.*, **54** (2006), 245–259. https://doi.org/10.1007/s11134-006-0300-z

55. S. A. Dudin, A. N. Dudin, O. S. Dudina, Development of a constructive method for verification of ergodicity and calculation of the stationary distribution of quasi-birth-and-death processes with spatially inhomogeneous transitions, *Mathematics*, **14** (2026), 1148. https://doi.org/10.3390/math14071148

AIMS Press