Performance Optimization of M/M/1 Queues with Working Vacations and Server Breakdowns in Wireless Communication Systems

S. Muthukumar[1,2], J. Ebenesar Anna Bagyam[1], and K. Basarikodi[2]

[1]*Karpagam Academy of Higher Education, Coimbatore, Tamil Nadu, India,*
[2]*Ramco Institute of Technology, Rajapalayam, Tamil Nadu, India*

https://doi.org/10.26636/jtit.2025.3.2195

Abstract — This paper presents a unified analytical and simulation framework for optimizing the performance of M/M/1 queueing systems that incorporate differentiated working vacations, server breakdowns, and customer balking behavior. Other features of the solution include dynamical transitions between full-service mode, two levels of working vacation (with reduced service rates) phases, and random breakdown-repair cycles. Customers arrive via a Poisson process and decide to join or balk based on the server's current state. Embedded Markov chains, probability generating functions, and Matlab based discrete event simulation are applied to analyze key performance metrics, including average waiting time, queue length, and server utilization. A particle swarm optimization (PSO) algorithm is used to identify parameter configurations that minimize congestion and delay. Application scenarios in 5G/6G networks and service platforms demonstrate how adaptive vacation scheduling and resilience strategies improve energy efficiency and throughput. The results offer valuable information for performance tuning in resource-constrained telecommunication systems.

Keywords — 5G/6G networks, queueing performance analysis, server reliability, single-server queues, working vacations

1. Introduction

Modern wireless communication systems, including 5G, 6G and Internet of Things (IoT) networks, face increasing levels of demand for high-quality services, yet have limited computational and energy resources at their disposal. The dynamic nature of network traffic, combined with energy constraints and unpredictable system behavior, calls for deploying adaptive queueing mechanisms that are capable of managing service delays, optimizing resource usage, and ensuring operational resilience. The queue theory provides a robust analytical framework for addressing such challenges and has become a key tool in the process of modeling and optimizing telecommunication networks [1], [2].

Among the various queueing models, the working vacation queue, originally proposed in [3], has gained attention for its practical applicability in energy-aware systems. Unlike traditional vacation models where the server becomes completely idle, the working vacation model assumes that the server continues to operate at a reduced rate, closely resembling energy-saving or degraded service modes known from wireless infrastructures. This framework has been extended to include re-trials, server unreliability, customer impatience, and heterogeneous vacation behaviors [4], [5], improving its relevance to complex environments such as sensor networks and cloud-edge systems [6], [7].

Recent advances in network management include intelligent queue control in 5G, hybrid optimization models for traffic handling, and delay-tolerant service designs [8]–[10]. However, few studies jointly address such phenomena as server breakdowns, multiple working vacation phases, and balking behavior, especially under the constraints of wireless systems, where these conditions often coexist.

In our previous work [5], we analyzed the steady state behavior of an M/M/1 queueing system with differentiated working vacations and customer balking. This paper extends that model by introducing the following improvements:

- Two distinct working vacation phases, each with its own reduced service rate.
- Random server breakdowns and repair dynamics, modeling real-world hardware unreliability.
- State-dependent customer balking, where join/balk decisions are influenced by the server's operating mode.
- Integration of discrete-event simulation in Matlab to validate the analytical findings.
- Application of particle swarm optimization (PSO) to identify optimal system parameters while minimizing delays and improving performance metrics.

The inclusion of PSO in this study is a significant methodological enhancement. PSO is a robust, population-based metaheuristic that efficiently explores complex, non-linear search spaces and converges quickly. It is particularly well-suited for optimizing queueing systems with stochastic behavior, where traditional gradient-based methods may struggle. By applying PSO to tune service rates, vacation parameters, breakdown rates, and balking thresholds, this paper offers

Tab. 1. Model parameters and their descriptions.

Parameter	Description	Typical values
λ	Customer arrival rate (Poisson process)	Example: 1.0
μ	Service rate during busy period	Must satisfy $\mu > \mu_1 > \mu_2$
μ_1	Service rate during type I working vacation	Reduced service rate
μ_2	Service rate during type II working vacation	Further reduced service rate
γ_1	Probability of entering type I vacation	Example: 0.1
γ_2	Probability of entering type II vacation	Example: 0.05
α	Probability of server breakdown	Example: 0.03
β	Server repair rate after breakdown	Exponential repair rate
b_1	Balking probability during type I vacation	Set to 0.3 (heuristic)
b_2	Balking probability during type II vacation	Set to 0.5 (heuristic)
b	Balking probability during server breakdown	Set to 0.7 (heuristic)

an actionable optimization framework for improving system responsiveness and energy efficiency in dynamic wireless communication environments.

Using embedded Markov chains, probability-generating functions, and discrete event simulation, we analyze key performance indicators such as average waiting time, server utilization, and system throughput. Our results provide practical design insights for customer service platforms, edge computing nodes, and base stations in energy- and reliability-constrained systems.

2. Model Description

This study analyses a single-server M/M/1 queueing system that incorporates differentiated working vacations, server breakdowns, and customer balking behavior. The system is structured under the following assumptions:

- **Arrival process**. Customers arrive following a Poisson process with rate λ.
- **Service mechanism**. During regular busy periods, the server operates at a service rate of μ. In type I vacation mode, the server continues to operate at a reduced rate μ_1, and in type II vacation mode, at an even slower rate μ_2, where $\mu > \mu_1 > \mu_2$.

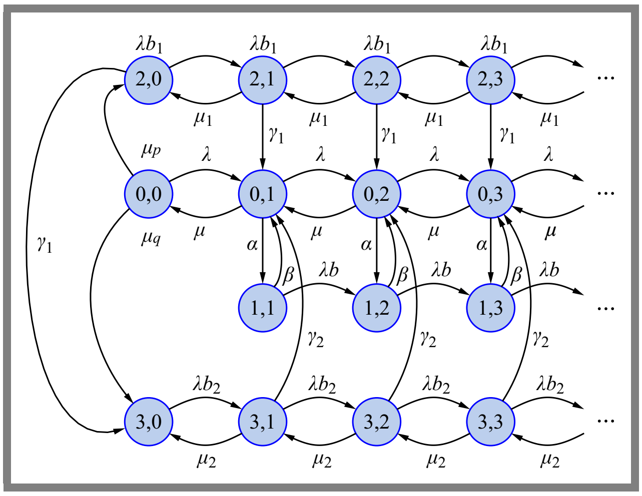

Fig. 1. State transition diagram.

- **Server behavior**. After completing a busy period, the server may enter a type I vacation with probability γ_1 or a type II vacation with probability γ_2.
- **Server breakdowns**. Random breakdowns occur with probability α, and the server undergoes repair at an exponential rate β.
- **Customer balking**. Customers may choose to balk depending on the server's state, with a 30% balking probability during type I vacation, 50% during type II vacation, and 70% during server breakdowns.

This model effectively captures real-world operational constraints such as fatigue, partial service availability, failure events, and customer impatience, making it highly relevant for the analysis and optimization of both traditional service systems and next-generation wireless networks.

2.1. Model Framework

This study builds on the queueing model presented in [5], which explored a steady-state M/M/1 system with differentiated working vacations, breakdowns, and balking. The current model retains the structural foundation of that system but introduces refined interpretations and supports simulations required for validating performance.

Let $N(t)$ denote the number of customers in the system at time t, and let $S(t) \in \{0, 1, 2, 3\}$ represent the server's state, where:

- $S(t) = 0$ – server is busy,
- $S(t) = 1$ – server is under breakdown,
- $S(t) = 2$ – server is on a type I working vacation,
- $S(t) = 3$ – server is on a type II working vacation.

The system is modeled as a continuous-time Markov process $S(t), N(t), t \geqslant 0$ with state space $\Lambda = \{(i, j) : i = 0, 1, 2, 3; j \geqslant 0\}$. Transition probabilities and the governing balance equations are retained from the earlier model, with minor notational refinements. For completeness, the main steady-state equations and performance metrics, including the expected number of customers in system and average waiting time, are summarized in Appendix A.

3. Practical Applications in Wireless Networks

This section demonstrates how the proposed M/M/1 queueing model with differentiated working vacations, server breakdowns, and customer balking can be effectively applied to modern service systems. Two representative domains are considered: customer service centers and next-generation wireless networks (NGWN).

3.1. Customer Service Centers and Call Centers

In service-oriented platforms like call centers, help desks or support chat systems, human agents act as servers processing customer requests. The proposed model offers several relevant analogies:

- **Server breakdowns** correspond to sudden unavailability of agents due to technical issues, fatigue, or shift changes.
- **Working vacations** represent scheduled breaks or periods of reduced service effort, e.g. multitasking, handling low-priority tasks.
- **Customer balking** models real-world impatience such as callers hanging up or users exiting queues when facing perceived delays.

By implementing the proposed model, organizations may design dynamic staffing policies to:

- Reduce call abandonment rates and improve response times.
- Optimize agent workload while avoiding burnout.
- Adapt service capacity based on real-time traffic.

Strategically timed low-effort periods (type I vacations) can preserve service quality while allowing recovery time, as long as breakdown probability and balking are carefully managed.

3.2. Next Generation Wireless Networks

In wireless systems, particularly 5G/6G networks, IoT gateways, and edge computing nodes, the model maps directly to network elements and protocols in the following manner:

- Servers represent base stations or edge nodes that process data packets or user requests.
- Working vacations correspond to energy savings or degraded operating modes during off-peak hours or congestion periods.
- Breakdowns simulate hardware failures, link failures, or shutdowns caused by overheating.
- Balking models packet drops or user session termination due to degraded quality of service (QoS).

The model enables adaptive resource allocation and energy-aware operations. It supports the following:

- Sleep/wake cycles in small cells or relay nodes for energy efficiency,
- Fault tolerance mechanisms through predictive repair and redundancy,

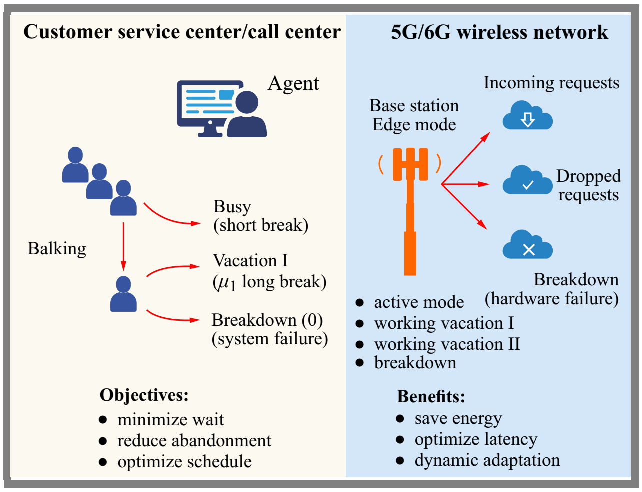

Fig. 2. Analogies of the proposed queueing model used in call centers and wireless network systems.

- Dynamic load balancing to reduce service delays and user drop rates.

Benefits for network operators include improved resilience and responsiveness, lower energy consumption without sacrificing throughput, scalable performance modeling for smart city infrastructure, vehicular networks, and cloud-edge orchestration.

4. Simulation Setup and Dataset Description

A simulation was performed to evaluate the performance of the system under the following parameters:

- Arrival rate $\lambda = 1.0$,
- Service rates: $\mu = 1.5$ (normal), $\mu_1 = 1.0$ (type I vacation), $\mu_2 = 0.5$ (type II vacation),
- Vacation probabilities: $\gamma_1 = 0.1$ (type I), $\gamma_2 = 0.05$ (type II),
- Breakdown probability $\alpha = 0.03$,
- Repair rate $\beta = 0.2$,
- Balking probabilities: $b_1 = 0.3$ (type I), $b_2 = 0.5$ (type II), and $b = 0.7$ (breakdowns).

The simulation ran for a total of 10 000 customer events, ensuring statistically significant results that capture typical system behavior under varying server states.

This representative subset highlights how customer arrivals, service commencement, and balking behavior are influenced by the server's state, providing practical insights into the system's dynamics.

5. Numerical Results and Validation

To evaluate the performance of the proposed M/M/1 queueing system with differentiated working vacations, server breakdowns, and customer balking, we developed a Matlab-based simulation. The simulation model mimics system behavior

Tab. 2. Sample simulation output (first five events).

Arrival time	Service start	Service end	Server state	Decision
0.47	0.47	1.08	Busy	Joined
0.64			Vacation I	Balked
2.65	2.65	2.66	Busy	Joined
6.15	6.15	6.29	Busy	Joined
6.36	6.36	6.73	Busy	Joined

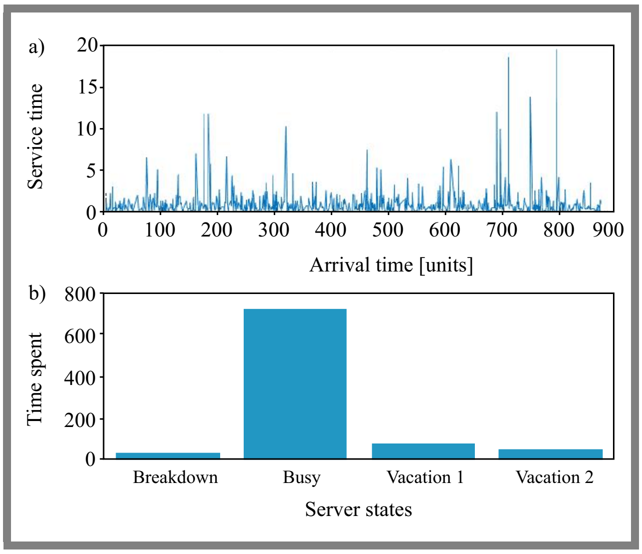

Fig. 3. Simulation outputs: a) service times for customers and b) cumulative time spent in different server states.

over 10 000 customer arrivals, ensuring statistical robustness and replicability. The key outputs are visualized in Fig. 3.

In Fig. 3a, the x-axis represents individual customer arrival times, while the y-axis shows their corresponding service durations. The figure illustrates how service times vary depending on server state: they are the shortest during busy periods, longer during type I vacations, and the longest during type II vacations or breakdown periods. Clusters of elevated service times reflect transitions into low-efficiency or failure states.

Figure 3b illustrates the time spent in different states. This bar chart quantifies the cumulative time spent in each server state (busy, vacation I, vacation II, breakdown). It clearly shows how working vacations and breakdowns reduce effective service capacity, helping identify bottlenecks and guide parameter tuning (e.g. reducing γ_2 or improving β).

These visualizations offer actionable insights into system dynamics and underscore the impact that server fatigue and failures exert on customer experience. The simulation provides a practical validation layer to the analytical model, confirming its relevance for real-world service systems and telecommunication networks, where non-ideal behaviors like balking and degradation are common.

5.1. Simulation Environment

All simulations, including queue dynamics and particle swarm optimization (PSO), were implemented in Matlab R2023b. Matlab's built-in functions and custom scripts were used to model queue states, implement PSO algorithms, and generate figures. Random number seeds were set to ensure consistency across repeated runs.

6. Optimization Framework Using PSO

Particle swarm optimization is employed to minimize cost functions and optimize key performance metrics, expected queue length, waiting time, and server utilization, in an M/M/1 queue with working vacations, breakdowns, and customer balking. PSO is particularly suitable for this problem due to its fast convergence, simplicity, and robustness in non-linear, high-dimensional search spaces. Compared to other metaheuristics (e.g. GA, ACO), PSO requires fewer parameters and is more computationally efficient.

The objective of the PSO algorithm is to minimize cost function C that represents the trade-off between queueing performance metrics, i.e. the average number of customers in the system, and the average waiting time.

The cost function is defined as:

$$C = w_1 . E(L) + w_2 . E(W) \,, \tag{1}$$

where:

- $E(L)$ – expected number of customers in the system,
- $E(W)$ – expected waiting time from Little's law: $E(W) = \frac{E(L)}{\lambda}$,
- $w_1, w_2 \in [0, 1]$ are user-defined weights such that: $w_1 + w_2 = 1$, reflecting the relative importance of queue length and delay.

For example, setting $w_1 = w_2 = 0.5$ gives equal weight to both performance criteria.

The PSO algorithm searches for the parameter set $\theta = [\mu_1, \mu_2, \gamma_1, \gamma_2, \alpha, \beta, b_1, b_2, \text{and}\, b]$ that minimizes this cost function:

$$\theta^* = \arg\min_{\theta} c(\theta) \,. \tag{2}$$

It is defined based on parameters such as arrival/service rates $\lambda, \mu, \mu_1, \mu_2$, breakdown and repair rates α, β, vacation probabilities γ_1, γ_2, and balking probabilities b_1, b_2, and b.

Initialization of the algorithm includes swarm size, inertia weight w, and coefficients c_1, c_2. Parameter bounds are defined, e.g., $\mu_1 \in [0.1, \mu]$. Each particle encodes queue parameters:

$$Particle_i = [\mu_1, \mu_2, \gamma_1, \gamma_2, \alpha, \beta, balking\ par, \ldots] \,. \tag{3}$$

At iteration $t+1$, particle velocities and positions are updated as:

$$v_i^{t+1} = w\, v_i^t + c_1\, r_1 (pBest_i - x_i^t) + c_2\, r_2 (gBest - x_i^t) \,, \tag{4}$$

$$x_i^{t+1} = x_i^t + v_i^{t+1} \,, \tag{5}$$

Tab. 3. Baseline vs. optimized performance.

Metric	Baseline	PSO optimized	Improvement
Average queue length	6.12	3.48	↓~43.1%
Average waiting time [units]	5.01	2.23	↓~55.5%
Server utilization	78.4%	85.7%	↑ +7.3%

where $r_1, r_2 \sim U(0, 1)$ are random factors, $pBest_i$ is the best position of particle i, and $gBest$ is the global best found so far. The algorithm terminates when a maximum number of iterations is reached or the improvement falls below a threshold.

6.1. PSO Results

To assess the impact of the PSO algorithm, we compared performance metrics before and after optimization (Tab. 3). The PSO was applied to tune key parameters, e.g. $\mu_1, \mu_2, \gamma_1, \gamma_2, \alpha, \beta$, and balking probabilities, with the objective of minimizing average queue lengths and waiting times.

These results confirm that PSO effectively identifies superior parameter configurations reducing congestion and improving response times.

The plot shown in Fig. 4 shows a smoothed trend of service durations aligned with arrival times. It illustrates how optimized parameters help maintain service times within a tighter range, thus avoiding spikes observed under baseline settings. This uniformity leads to greater predictability and reduced customer waiting time.

6.2. PSO Parameter Settings and Cost Convergence

To validate the performance of the PSO algorithm in optimizing the M/M/1 queue system with working vacations and server breakdowns, Tab. 4 summarizes the parameter settings used in the simulation.

The cost function used in optimization reflects a weighted combination of average queue length and average waiting time. These metrics capture the overall efficiency of the system and customer satisfaction under varying conditions, for example, vacations, breakdowns, and balking.

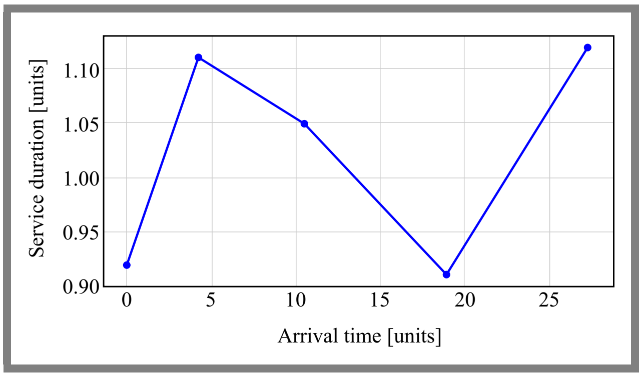

Fig. 4. Optimized service times across arrival times.

Tab. 4. PSO parameter settings.

Parameter	Value	Description
Swarm size	30	Number of particles in the swarm
Number of iterations	100	Maximum iterations of the algorithm
Inertia weight w	0.7	Controls influence of previous velocity
Cognitive coefficient c_1	1.5	Weight toward personal best position
Social coefficient c_2	1.5	Weight toward global best position

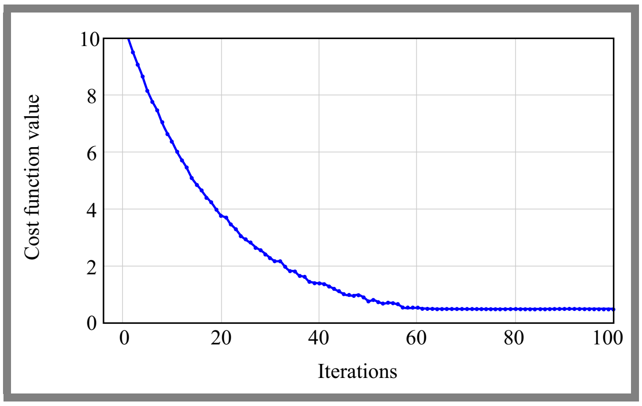

Fig. 5. Convergence of the PSO cost function.

A cost convergence plot was generated to visualize the performance of the PSO algorithm in the individual iterations. As shown in Fig. 5, the cost steadily decreases and stabilizes as the algorithm converges toward an optimal parameter set.

The convergence curve demonstrates that the algorithm reliably reduces the cost within 100 iterations, indicating successful optimization of the queue parameters under the defined constraints.

7. Conclusions

This article presents an enhanced M/M/1 queueing model that integrates differentiated working vacations, server breakdowns, and state-dependent customer balking, offering a realistic framework for analyzing and optimizing performance in modern telecommunication systems. The model captures essential operational dynamics often encountered in environments such as call centers, wireless base stations, and edge computing nodes, where service degradation, unreliability, and impatient user behavior are common.

The model demonstrates significant improvements in performance, offers a reduction in average waiting time (up to 55%) and a 43% decrease in queue length. These results underscore the potential of intelligent queue management and metaheuristic optimization approaches, as these are capable of improving responsiveness and energy efficiency in resource-constrained networks. The proposed framework offers actionable insights for designing adaptive scheduling policies, optimizing energy usage, and enhancing user satisfaction in 5G/6G, IoT gateways, and customer-facing service platforms.

Although PSO has proven effective in optimizing the proposed queueing model due to its fast convergence and simplicity, it is beneficial to briefly consider alternative metaheuristic approaches. Genetic algorithms (GAs), for instance, offer robustness and flexibility, particularly for discrete optimization problems, but often require more computational effort and parameter tuning.

Reinforcement learning (RL), on the other hand, enables adaptive learning in dynamic environments and is well-suited for online decision making. However, its applicability is limited in settings where training data is sparse or where system states evolve slowly.

PSO was selected for this study because of its ease of implementation, compatibility with simulation-based optimization, and lower computational overhead in static system scenarios. Future comparative studies may further explore the trade-offs among these techniques to guide optimization choices across various application domains.

Acknowledgments

Data supporting the findings of this study were reported in our previous publication [5]. Additionally derived data supporting this study are available from the authors upon reasonable request.

Appendix A. Local Balance Equations

Let $p_{i,j}$ denote the steady-state probability of being in server state i with j customers in the system.

The balance equations for the continuous-time Markov process are as follows:

$$(\lambda+\mu)\,p_{0,0}=\mu\,p_{0,1}\ , \tag{6}$$

$$(\lambda+\mu+\alpha)\,p_{0,n}=\gamma_1\,p_{2,n}+\gamma_2\,p_{3,n}+\beta\,p_{1,n} + \mu\,p_{0,n+1}+\lambda\,p_{0,n-1},\ n\geqslant 2\ , \tag{7}$$

$$(\lambda b+\beta)\,p_{1,1}=\alpha\,p_{0,1}\ , \tag{8}$$

$$(\lambda b+\beta)\,p_{1,n}=\alpha\,p_{0,n}+\lambda b\,p_{1,n-1},\ n\geqslant 1\ , \tag{9}$$

$$(\lambda b_1+\gamma_1)\,p_{2,0}=\mu_1\,p_{2,1}+\mu p\,p_{0,0}\ , \tag{10}$$

$$(\lambda b_1+\gamma_1+\mu_1)\,p_{2,n}=\mu_1\,p_{2,n+1}+\lambda b_1\,p_{2,n-1},\ n\geqslant 1\ , \tag{11}$$

$$\lambda b_2\,p_{3,0}=\gamma_1\,p_{2,0}+\mu q\,p_{0,0}+\mu_2\,p_{3,1}\ , \tag{12}$$

$$(\lambda b_2+\gamma_2+\mu_2)\,p_{3,n}=\lambda b_2\,p_{3,n-1}+\mu_2\,p_{3,n+1},\ n\geqslant 1\ , \tag{13}$$

Generating functions

Using corresponding probability generating functions

$$P_i(z)=\sum_{n=0}^{\infty}p_{i,n}z^n$$

the expressions are:

$$P_0(z)=\frac{(\mu-\alpha z)\,p_{0,0}-\big[P_1(z)\big]\beta z+\big[p_{2,0}-P_2(z)\big]\gamma_1 z}{\lambda z^2-(\lambda+\mu+\alpha)z+\mu} + \frac{\big[p_{3,0}-P_3(z)\big]\gamma_2 z}{\lambda z^2-(\lambda+\mu+\alpha)z+\mu}\ , \tag{14}$$

$$P_1(z)=\frac{\alpha\big[p_{0,0}-P_0(z)\big]}{\lambda bz-(\lambda b+\beta)}\ , \tag{15}$$

$$P_2(z)=\frac{\mu_1\,p_{2,0}-\mu_1 z\,p_{2,0}-\mu p\,z\,p_{0,0}}{\lambda b_1 z^2-(\lambda b_1+\mu_1+\gamma_1)z+\mu_1}\ , \tag{16}$$

$$P_3(z)=\frac{\mu_2\,p_{3,0}-z\,p_{3,0}(\mu_2+\gamma_2)-\gamma_1 z\,p_{2,0}-\mu q z\,p_{0,0}}{\lambda b_2 z^2-(\lambda b_2+\mu_2+\gamma_2)z+\mu_2}\ , \tag{17}$$

Steady-state probability at idle state

$$p_{0,0}=\frac{\gamma_1\gamma_2}{\gamma_1\gamma_2+(\gamma_2\,\mu\,p+\gamma_1\gamma_2\,s+\gamma_1^2\,t+\gamma_1\,\mu\,q)}\ , \tag{18}$$

where:

$$s=\frac{\mu p\,z_1}{\mu_1(z_1-1)}\ ,$$

$$t=-\frac{z_2\,(s\gamma_1+\mu q)}{z_2\,(\gamma_2+\mu_2)-\mu_2}\ ,$$

and:

z_1 is the positive root of

$$\lambda b_1 z^2-(\lambda b_1+\mu_1+\gamma_1)z+\mu_1\ ,$$

z_2 is the positive root of

$$\lambda b_2 z^2-(\lambda b_2+\mu_2+\gamma_2)z+\mu_2\ ,$$

Expected system lengths by state

Busy state:

$$E(L_B)=\frac{\gamma_1\gamma_2\,\lambda b_2\,p_{3,0}+[\gamma_1\gamma_2\,\mu_1+\gamma_1^2(\lambda b_2-\mu_2)]\,p_{2,0}}{\gamma_1\gamma_2\,\alpha} + \frac{[\gamma_2\,\mu p(\lambda b_1-\mu_1)+\gamma_1\,\mu q(\lambda b_2-\mu_2)+\gamma_1\gamma_2\lambda]\,p_{0,0}}{\gamma_1\gamma_2\,\alpha}\ , \tag{19}$$

Breakdown state:

$$E(L_1)=\frac{\alpha\,\big[E(L_B)-p_{0,0}\big]}{\beta}\ , \tag{20}$$

Type-I vacation:

$$E(L_2)=\frac{\big[\mu p\,(\lambda b_1-\mu_1)\big]\,p_{0,0}+\gamma_1\mu_1 p_{2,0}}{\gamma_1^2}\ , \tag{21}$$

Type-II vacation:

$$E(L_3) = \frac{\gamma_2 \lambda b_2 \, p_{3,0} + \gamma_1 (\lambda b_2 - \mu_2) \, p_{2,0} + \mu q (\lambda b_2 - \mu_2) \, p_{0,0}}{\gamma_2^2}, \tag{22}$$

Expected waiting time

Using Little's law:

$$E(W) = \frac{E(L_B) + E(L_1) + E(L_2) + E(L_3)}{\lambda}. \tag{23}$$

References

[1] S.A. Afolalu *et al.*, "A Short Review on Queuing Theory as a Deterministic Tool in Sustainable Telecommunication System", *Materials Today: Proceedings*, vol. 44, pp. 2884–2888, 2021 (https://doi.org/10.1016/j.matpr.2021.01.092).

[2] A. Roy, J.L. Pachuau, and A.K. Saha, "An Overview of Queuing Delay and Various Delay Based Algorithms in Networks", *Computing*, vol. 103, pp. 2361–2399, 2021 (https://doi.org/10.1007/s00607-021-00973-3).

[3] L.D. Servi and S.G. Finn, "M/M/1 Queues with Working Vacations (M/M/1/WV)", *Performance Evaluation*, vol. 50, pp. 41–52, 2002 (https://doi.org/10.1016/S0166-5316(02)00057-3).

[4] J.E.A. Bagyam, P. Suganthi, and P. Visali, "Unreliable Single Server Retrial Queueing Model with Repeated Vacation", *International Journal of Mathematics in Operational Research*, vol. 26, pp. 357–372, 2023 (https://doi.org/10.1504/IJMOR.2023.134838).

[5] S. Muthukumar and J.E.A. Bagyam, "A Steady-state Behavior of an M/M/1 Queue with Optional Differentiated Working Vacations, Server Breakdown, and Customer Balking", *Advances and Applications in Statistics*, vol. 92, pp. 603–631, 2025 (https://doi.org/10.17654/0972361725025).

[6] G. Bouloukakis, I. Moscholios, N. Georgantas, and V. Issarny, "Performance Analysis of Internet of Things Interactions via Simulation-based Queueing Models", *Future Internet*, vol. 13, art. no. 87, 2021 (https://doi.org/10.3390/fi13040087).

[7] X. Li, "Performance Evaluation of M/M/1 Queueing Models in Cloud Computing Environments", *Highlights in Science, Engineering and Technology*, vol. 85, pp. 841–848, 2024 (https://doi.org/10.54097/1xcw8y74).

[8] S. Jung, J. Kim, and J.H. Kim, "Intelligent Active Queue Management for Stabilized QoS Guarantees in 5G Mobile Networks", *IEEE Systems Journal*, vol. 15, pp. 4293–4302, 2020 (https://doi.org/10.1109/JSYST.2020.3014231).

[9] A.A. Rezaee and F. Pasandideh, "A Fuzzy Congestion Control Protocol Based on Active Queue Management in Wireless Sensor Networks with Medical Applications", *Wireless Personal Communications*, vol. 98, pp. 815–842, 2018 (https://doi.org/10.1007/s11277-017-4896-6).

[10] Z. Zhang *et al.*, "A Particle Swarm Optimization-based Queue Scheduling and Optimization Mechanism for Large-scale Low-Earth-orbit Satellite Communication Networks", *Sensors*, vol. 25, art. no. 1069, 2025 (https://doi.org/10.3390/s25041069).

S. Muthukumar, M.Phil., Assistant Professor
Department of Mathematics
https://orcid.org/0009-0007-1320-6082
E-mail: muthukumarmsc99@gmail.com
Karpagam Academy of Higher Education, Coimbatore, Tamil Nadu, India
https://kahedu.edu.in
Ramco Institute of Technology, Rajapalayam, Tamil Nadu, India
https://www.ritrjpm.ac.in

J. Ebenesar Anna Bagyam, Ph.D., Assistant Professor
Department of Mathematics
https://orcid.org/0000-0002-9943-5893
E-mail: ebenesar.j@gmail.com
Karpagam Academy of Higher Education, Coimbatore, Tamil Nadu, India
https://kahedu.edu.in

K. Basarikodi, Ph.D., Professor
Department of Mathematics
https://orcid.org/0000-0002-2668-7096
E-mail: basarikodi@ritrjpm.ac.in
Ramco Institute of Technology, Rajapalayam, Tamil Nadu, India
https://www.ritrjpm.ac.in