CARPATHIAN J. MATH.
Volume **39** (2023), No. 3,
Pages 745 - 758

Online version at https://www.carpathian.cunbm.utcluj.ro/
Print Edition: ISSN 1584 - 2851; Online Edition: ISSN 1843 - 4401
DOI: https://doi.org/10.37193/CJM.2023.03.13

Queueing and Reliability Analysis of Unreliable Multi-server Retrial queue with Bernoulli feedback

SHWETA UPADHYAYA[1], DIVYA AGARWAL[1], RADHIKA AGARWAL[1], IZHAR AHMAD[2,3], ABDULKARIM HASSAN IBRAHIM[4] and S. AL-HOMIDAN[2,4]

ABSTRACT. The study consists of an unreliable multiserver retrial queue with feedback where clients may resist to join the orbit and the service provider is erratic. In this model we have considered c service providers at a time and if all the c servers are busy, the incoming clients wait in the retrial orbit. The model is unfolded using Matrix Geometric Method (MGM) to obtain the rate matrix and stationary probabilities. The system with retrials is modelled and the performance measures and reliability indices are procured. These derived quotients are then visualised and validated with the help of tables and graphs. Further, the cost analysis of the model is carried out and the optimal cost for the system is obtained using Particle Swarm Optimization (PSO) algorithm for matrix method. The optimal repair and retrial rates are obtained using PSO technique which helps the computer and manufacturing industry to work towards better decision policies.

1. INTRODUCTION

Retrial queueing systems are the systems in which arriving jobs who find the server busy wait and retry to enter the system. Queueing model with retrials find their application in most modern computer networking systems including Wide Area Network (WAN), Local Area Network (LAN) protocols and network packet-switching. In day-to-day life, it is frequently noticed that people resist to join long queues. This state of affair can be viewed as *balking* behavior where impatience of clients can be observed. All of us are familiar with the fact that until the failed service provider is restored, the performances of the system such as the waiting time of the clients, the system size, queue length of the system, etc., may be heavily affected and cannot be reduced to some extent. In recent past, exploration of retrial queues facing breakdowns of the server has been a field of building interest. Taleb [21] studied preventive maintenance in a new version of an unreliable $M/G/1$ retrial queue with persistent and impatient customers. Lan [13] studied a discrete-time $Geo/G/1$ retrial queueing system with probabilistic preemptive priority and balking customers, in which the server is subject to starting failures and replacements in the repair times may occur with some probability. Lakaour [12] has discussed $M/M/1$ retrial queue with collisions, transmission errors and unreliable server. Detailed applications of retrial queueing models with breakdown can be seen in [16, 20, 24, 25].

The multi-server retrial line raises fascinating numerical and computational inquiries aside from its functional interest due to its more precise portrayal of several phenomena of congestion. Most of the several server retrial queues are designed by a level-dependent quasi-birth-and death (QBD) process. The primary component of this infinitesimal generator is the spatial heterogeneity that occurs due to repeated changes. This absence of homogeneity causes the insightful intricacy of retrial models. Haddad and Belarbi [5]

Received: 14.05.2022. In revised form: 08.03.2023. Accepted: 15.03.2023
2010 *Mathematics Subject Classification.* 60K25, 68M20, 90B22.
Key words and phrases. *Matrix-Analytic Method, $M/M/c$, retrial queue, unreliable server, feedback, balking, queueing model, reliability, cost optimization.*
Corresponding author: S. Al-Homidan; homidan@kfupm.edu.sa

analysed an unreliable $M/M/c$ retrial queue with $c \geq 3$ in which all servers are subject to breakdowns and repairs using phase merging algorithm. Kadi et al. [8] studied a multi-server $M/M/c$ queueing system with Bernoulli feedback and impatient customers (balking and reneging) under synchronous multiple and single vacation policies. Relevant literature on several-server retrial queueing system can be found in [2, 3, 7, 9, 15, 18, 23]. These authors have used the well known Matrix Geometric Method (MGM) for solving the queueing model. A practical application of multiserver queueing system can be seen in [11] where the authors have optimized the service level of queueing system of reception and the outpatient department during COVID-19 pandemic using the data of an outpatient department (OPD) of a public hospital in Hyderabad.

Scarcity of work done in multiserver queueing models including erratic service provider has motivated us to model an erratic $M/M/c$ queue with retrials, balking and bernoulli feedback where service provider is subjected to normal breakdowns. The model sees its application in telephone switching system, computer and communication networks, machinery and manufacturing sectors and in various sectors like cyber cafes, car wash services, ATMs, ticket vending machines at metro stations and other automated multinational companies. The highlights of the novelty of the work presented in the paper are

- To develop a queueing model considering erratic $M/M/c$ retrial queue with characteristic features defined above.
- To fabricate a Markov chain using Matrix Geometric Method.
- To find steady state solution to determine the steady state probabilities and queue length of the system.
- To determine the reliability indices and busy period.
- To perform a sensitivity analysis through numerical values and an example.
- To obtain the optimal cost of the system using PSO.

The remaining paper is composed as follows: Section 2 depicts the narration of our proposed queueing model. The detailed matrix model is discussed in section 3 where the state diagram is also provided for $c = 3$ to give better insight of the model. The steady state study and discussion of matrix geometric property in depth have been done in Section 4. Some special cases are also given here, which depicts that the results of our model can be reduced to the results of earlier studied models under special conditions. Section 5 presents the derivation and analysis of the performance measures along with reliability indices and busy period. Section 6 includes the numerical example with some mathematical outcomes where the effect of important performance measures on system size can also be seen. Further, the expected cost function is fabricated to accomplish the optimal ideal estimations of certain parameters using PSO in section 7 and lastly, section 8 incorporates concluding remark along with future scope of the paper.

2. Depiction of the model

Considering an erratic $M/M/c$ retrial line with balking and bernoulli feedback. The basic assumptions for the model are:

- *Arrivals and balking*: The arrival rate of a client is λ following Poisson process. If the event is that the server is occupied or is uncertain, then, at that point the arriving client may get annoyed and resist joining the queue and leaves the system permanently with probability ν or may join the queue with probability $\bar{\nu} = 1 - \nu$.
- *Service and retrial policy*: The entering client, on seeing the server unoccupied gets the service straight away. Else on seeing the server occupied or inoperative, the clients will have to retry later from a virtual orbit following first come first serve (FCFS) principle. Once the client is completely served, he may be unsatisfied and

join the retrial orbit to get served again with probability γ or may leave the system with probability $\bar{\gamma} = 1 - \gamma$, where every client in the orbit is considered to be as new client. The client on the top of the virtual track (orbit) retries to get served if the service provider is unoccupied with exponential rate α.

- *Breakdowns and repairs*: The service providers in the developed system are subject to breakdowns. They may face normal breakdowns with exponential rate θ while the repair process for the broken servers follow exponential rate r. Once the server is repaired, the service of the client is done.

Considering an upper limit N as the total sum of clients in the virtual orbit that are allowed to take retrials. This directly gives us the probability of repeating attempts during $(t,\ t+dt)$, such that there are j clients in virtual track at sometime t is $\alpha_j dt + o(dt)$, where $\alpha_j = min\{j,\ N\}\alpha$. Also presuming that the task of clients arrival, service times and inter-retrial times are mutually independent.

3. Matrix model description

Referring to the above described $M/M/c$ retrial model with normal breakdown, the random process Ξ is expressed as

$$\Xi = \{(X(t), Y(t), Z(t)) : X(t) = 0, 1, 2, ..., c; Y(t) = 0, 1, 2, ...; Z(t) = 0, 1, 2, ..., c\}.$$

Here $X(t)$ is the possible busy service providers, $Y(t)$ is the client's total number that are in retrials and $Z(t)$ is possible number of servers under breakdown. Figure 1 presents the state diagram for the model corresponding to $c = 3$.

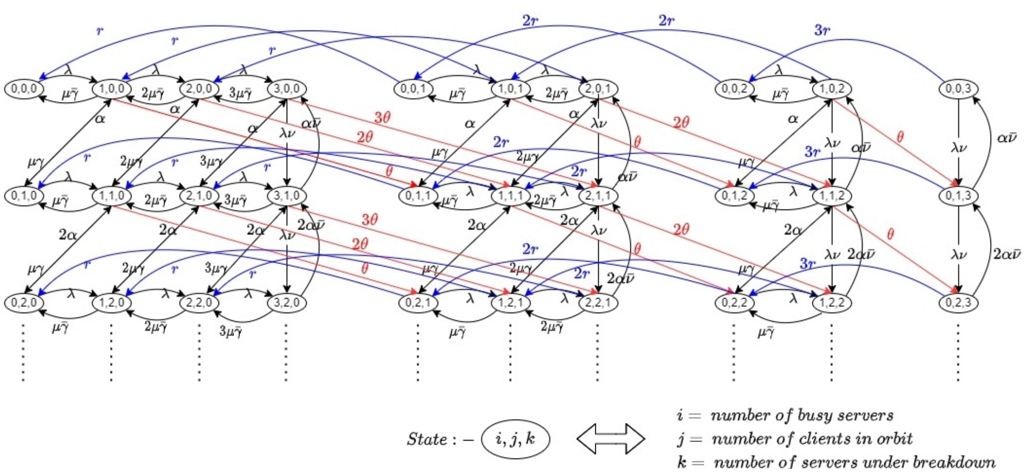

FIGURE 1. State Transition Diagram

The probable state space is given by

$$\xi = \{(i, j, k) :\ i = 0, 1, 2, ..., c;\ j = 0, 1, 2, ...;\ k = 0, 1, 2, ..., c - i\}.$$

Under the steady state, the state probabilities are defined as follows:

- $p_{i,j,k}$: the uncertainty (probability) of i servers being busy, k servers being in breakdown and j clients being in virtual track, where $j \geq 0, 0 \leq i + k \leq c$.
- $p_{c,j,0}$: the probability that all c servers are occupied, no disrupted server and j clients are in virtual track.

- $p_{n,j,k}$: the probability that $n \leq c$ servers are occupied, k disrupted servers and j clients are in the orbit.
- $p_{0,j,c}$: the probability that all c servers are disrupted and are under repair, no busy server and j clients are in the orbit.
- $p_{i,j,n}$: the probability that there are i occupied servers, j clients are in virtual track and $n \leq c$ disrupted servers.
- $p_{i,0,k}$: the probability that there are i occupied servers, no client is in the virtual track and k disrupted servers.
- $p_{i,n,k}$: the probability that there are i occupied servers, n clients are in the virtual track and k disrupted servers.

The infinitesimal generator A of the Markov chain, using the lexicographical sequence of different states is given in the form

$$A = \begin{bmatrix} S_0 & U & 0 & 0 & 0 & 0\dots & 0 & \dots \\ T_1 & S_1 & U & 0 & 0 & 0\dots & 0 & \dots \\ 0 & T_2 & S_2 & U & 0 & 0\dots & 0 & \dots \\ \vdots & \vdots & \vdots & \ddots & \vdots & \dots & 0 & \dots \\ 0 & 0 & 0 & T_{N-1} & S_{N-1} & U & 0 & \dots \\ 0 & 0 & 0 & 0 & T & S & U & \dots \\ 0 & 0 & 0 & 0 & 0 & T & S & \dots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix}.$$

The matrices $S_j (j \geq 0)$, $T_j (j \geq 1)$ and U are of order $(c+1)(c+2)/2$.

S_j is partitioned into block sub-diagonal matrix as

$$S_j = \begin{bmatrix} C_j^0 & 0 & 0 & \dots & 0 & 0 & 0 \\ B^1 & C_j^1 & 0 & \dots & 0 & 0 & 0 \\ 0 & B^2 & C_j^2 & \dots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & B^{c-1} & C_j^{c-1} & 0 \\ 0 & 0 & 0 & \dots & 0 & B^c & C_j^c \end{bmatrix}$$

where B^k is $(c+1-k, c+2-k)$ ordered matrix s.t. $B^k[i,i] = k\beta$; $0 \leq i \leq c-k,\ k = 1, 2, ..., c$ and C_j^k is a $(c+1-k)*(c+1-k)$ matrix with entries

$$C_j^k = \begin{cases} C_j^k[i,i+1] = \lambda, & 0 \leq i \leq c-k,\ j \geq 0,\ 0 \leq k \leq c \\ C_j^k[i+1,i] = i\bar{\gamma}\mu, & 0 \leq i \leq c-k,\ j \geq 0,\ 0 \leq k \leq c \\ C_j^k[k,k-1] = kr, & 0 \leq i \leq c-k,\ j \geq 0,\ 1 \leq k \leq c \\ C_j^k[0,0] = -(\lambda + kr + j\alpha), & j \geq 0,\ 0 \leq k \leq c \\ C_j^k[i,i] = -\{\lambda + i(\mu+\theta) + kr + j\alpha\}, & 1 \leq i \leq c-k-1,\ j \geq 0,\ 0 \leq k \leq c \\ C_j^k[c+1-k, c+1-k] = & \\ -\{\lambda\nu + (c+1-k)(\mu+\theta) + kr + j\alpha\bar{\nu})\}, & j \geq 0,\ 0 \leq k \leq c \end{cases}$$

T_j is partitioned into block diagonal matrix as

$$T_j = \begin{bmatrix} D_j^0 & 0 & 0 & \dots & 0 & 0 \\ 0 & D_j^1 & 0 & \dots & 0 & 0 \\ 0 & 0 & D_j^2 & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & D_j^{c-1} & 0 \\ 0 & 0 & 0 & \dots & 0 & D_j^c \end{bmatrix}$$

where D_j^k is a $(c+1-k)*(c+1-k)$ matrix s.t $j \to j-1$ and $k \to k-1$ with elements to be

$$D_j^k = \begin{cases} D_j^k[i,i+1] = j\alpha, & 0 \le i \le c-k-1,\ j \ge 1,\ 0 \le k \le c \\ D_j^k[c+1-k, c+1-k] = j\alpha\bar{\nu}, & j \ge 1,\ 0 \le k \le c \end{cases}$$

U is partitioned into block super-diagonal matrix as

$$U = \begin{bmatrix} E_j^0 & F_j^0 & 0 & 0 & \dots & 0 & 0 \\ 0 & E_j^1 & F_j^1 & 0 & \dots & 0 & 0 \\ 0 & 0 & E_j^2 & F_j^2 & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \dots & E_j^{c-1} & F_j^{c-1} \\ 0 & 0 & 0 & 0 & \dots & 0 & E_j^c \end{bmatrix}$$

where E_j^k is a $(c+1-k)*(c+1-k)$ matrix s.t. $j \to j+1$ and $k \to k+1$ with entries of elements defined as

$$E_j^k = \begin{cases} E_j^k[i,i-1] = i\gamma\mu, & 1 \le i \le c-k,\ j \ge 0,\ 0 \le k \le c-1 \\ E_j^k[c+1-k, c+1-k] = \lambda\nu, & j \ge 0,\ i+k = c \end{cases}$$

and F_j^k is a matrix of order $(c+1-k)*(c-k)$ with $F_j^k[i+1,i] = i\theta,\ 1 \le i \le c-k.$

4. Matrix-Geometric Property and Steady State Solution

In order to solve the infinitesimal matrix A, we have different methods like Matrix analytic method and Matrix Geometric Method (MGM). Clearly matrix A is a repetitive matrix and MGM is a method to analyse the continuous quasi birth-death process whose transition matrix has a repetitive structure. In order to truncate the infinite matrix, a rate matrix R is deduced using cyclic reduction and then solving the generated finite matrix which is a replica of the infinite matrix A. This method helps in simplification of the matrix model, thus motivating us to obtain the results of the developed model using MGM to get the steady state solution.

4.1. **Steady State Solution.** In this sub-section, the steady state probabilities are to be obtained. Let $\pi = [\pi_0, \pi_1, \pi_2, ...]$ be its steady state probability vector which satisfies

$$\pi A = 0 \text{ and } \pi e = 1$$

where each $\pi_j = [p_{0,j,0}, p_{1,j,0}, ..., p_{c,j,0}, p_{0,j,1}, p_{1,j,1}, ..., p_{c-1,j,1}, ..., p_{0,j,c-1}, p_{1,j,c-1}, p_{0,j,c}], j \ge 0$ and e is a column vector with all entries as 1. As defined earlier, clearly the Markov chain A has a repetitive structure. The stationary probability vector $\pi = \{\pi_0, \pi_1, \pi_2, ...\}$ of A exists under the equilibrium condition. So the equations of steady-state in the matrix form are presented to evaluate vector π as follows:

$$\begin{aligned} \pi_0 S_0 + \pi_1 T_1 &= 0, \\ \pi_{i-1} U + \pi_i S_i + \pi_{i+1} T_{i+1} &= 0,\ 1 \le i \le N, \\ \pi_{N-1} U + \pi_N S_N + \pi_{N+1} T_N &= 0, \\ \pi_N R^{i-1-N} U + \pi_N R^{i-N} S_N + \pi_N R^{i+1-N} T_N &= 0,\ i \ge N+1. \end{aligned} \tag{4.1}$$

$$\sum_{i=0}^{\infty} \pi_i e = 1. \tag{4.2}$$

Evaluating the value of $\pi_0,\ \pi_1, ..., \pi_N$ using equation (4.1), we get

$$\begin{aligned}\pi_0 &= \pi_1 T_1(-S_0)^{-1} = \pi_1\xi_1,\\ \pi_{i-1} &= \pi_i T_i[-(\xi_{i-1}U + S_{i-1})]^{-1} = \pi_i\xi_i,\ 2 \leq i \leq N,\end{aligned} \tag{4.3}$$

Also

$$\pi_N\xi_N U + \pi_N S_N + \pi_N R T_N = 0. \tag{4.4}$$

Clearly, from equation (4.3), π_i $(0 \leq i \leq N-1)$ can be re-written in the from of π_N and the rest of the vectors $\{\pi_N, \pi_{N+1}, \pi_{N+2}, ...\}$ can be obtained using $\pi_i = \pi_N R^{i-N}, i \geq N$. Finally getting the steady state solution to be $\{\pi_0, \pi_1, \pi_2, ..., \pi_{N-1}, \pi_N, \pi_{N+1}, \pi_{N+2}, ...\}$. The normalizing condition to be used to get π_N is as follows:

$$\begin{aligned}\sum_{i=0}^{\infty}\pi_i e &= \{\pi_0 + \pi_1 + \pi_2 + ... + \pi_{N-1} + \pi_N + \pi_{N+1} + \pi_{N+2} + ...\}e\\ &= \{\pi_N\prod_{i=N}^{1}\xi_i + \pi_N\prod_{i=N}^{2}\xi_i + ... + \pi_N\prod_{i=N}^{N}\xi_i + \pi_N + \pi_N R + \pi_N R^2 + ...\}e\\ &= \pi_N\left[\sum_{k=1}^{N}\prod_{i=N}^{k}\xi_i + (I-R)^{-1}\right]e = 1.\end{aligned} \tag{4.5}$$

Following Cramer's rule in solving equation (4.4) and (4.5), one can get the steady-state probability vector π_N easily.

4.2. **Matrix-Geometric Property.** This sub-section deals with the basics of matrix analysis method and study about the rate matrix. The rate matrix R is the minimal non negative solution of the equation

$$T + SR + UR^2 = 0. \tag{4.6}$$

Referring to Neuts [17] and Latouche and Ramaswami [14], R is specified by $\lim_{n\to\infty} R_n$. The sequence R_n is represented as

$$R_0 = 0 \ \&\ R_{n+1} = -[US^{-1} + TS^{-1}R_n^2];\ n = 0, 1, 2, ... \tag{4.7}$$

The rate matrix R can be determined from above equation by successive substitutions as R_n converges monotonically. The probabilities π_j can be evaluated using matrix geometric solution i.e. $\pi_{j+N} = \pi_N R^j$, for $j \geq 1$.

Using Theorem 3.1.1. from Neuts [17], the balance situation of the model exists iff

$$XUe < XTe \tag{4.8}$$

where X is invariant probability of matrix $G = T + S + U$ s.t. X satisfies $XG = 0$ and $Xe = 1$. On substituting T and U in equation (4.8) we get,

$$\alpha N(1 - P_G) > \lambda P_G \tag{4.9}$$

where P_G is the probability of all service providers being occupied (i.e. $i + k = c$). This implies that the designed model is stable iff expected rate of retrials is larger than the expected rate of entries in virtual track. For more detailed explanation of this method and solution one can refer the work of Jain and Jain [6].

4.3. **Special Cases.** Here we present some particular cases. It can be seen that under certain conditions, our developed model reduces to those of previously studied models by various authors.

(i) When $\theta = 0$ and $r = 0$ (no breakdown and no repair), our modelled system reduces to the model developed by Lin and Ke [15].

(ii) When $\nu = 1$ and $\gamma = 0$ (no balking and no feedback), the results of the model studied by Subramanian et al. [23] can be obtained.

(iii) When $c = 2$, $\nu = 1$ and $\gamma = 0$ (two server, no balking and no feedback) our model reduces to a $M/M/2$ queue with retrial and normal failure which coexist with the study done by Raiah and Oukid [19].

(iv) When $c = 1$, $\nu = 1$ and $\gamma = 0$ (single server, no balking and no feedback), the model reduces to $M/M/1$ unreliable queue with retrial which coexist with the study done by Lakaour et al. [12].

5. System performance measures

The qualitative behavior of the queuing model is thoroughly studied using the performance measures of the system. With the help of probability steady state vectors, the following long run probabilities and average queue length can be obtained for different values of system parameters λ, μ, α, ν, γ, θ and r. The next subsections provide the long run probabilities of different states, reliability indices and system busy period as follows:

5.1. **Probabilities and queue length.** The long run probability of different system states along with queue length, mean number of occupied and broken-down service provider are given as:

- The long run probability of the number of server's occupied

$$\text{Prob (all } c \text{ servers are occupied)} = \sum_{j=0}^{\infty} p(c, j, 0).$$

$$\text{Prob (} n \text{ occupied servers where } n \leq c) = \text{P[O]} = \sum_{j=0}^{\infty} \sum_{k=0}^{n-c} p(n, j, k).$$

- The long run probability of the number of server under breakdown

$$\text{Prob (all } c \text{ servers are under breakdown)} = \sum_{j=0}^{\infty} p(0, j, c).$$

$$\text{Prob (} n \text{ servers under breakdown where } n \leq c) = \text{P[B]} = \sum_{j=0}^{\infty} \sum_{i=0}^{n-c} p(i, j, n).$$

- The long run probability of the number of clients in virtual orbit

$$\text{Prob (no client is in orbit)} = \sum_{k=0}^{c} \sum_{i=0}^{c-k} p(i, 0, k).$$

$$\text{Prob (} n \text{ clients are in orbit} = \sum_{k=0}^{c} \sum_{i=0}^{c-k} p(i, n, k).$$

- Average number of occupied server (ANOS)

$$\text{ANOS} = \sum_{n=0}^{c} n \left\{ \sum_{j=0}^{\infty} \sum_{k=0}^{n-c} p(n, j, k) \right\}.$$

- Average number of broken-down server (ANBS)

$$\text{ANBS} = \sum_{n=0}^{c} n\left\{\sum_{j=0}^{\infty}\sum_{i=0}^{n-c} p(i,j,n)\right\}.$$

- Average number of clients in the virtual orbit (E[L]) are

$$\text{E[L]} = \sum_{n=0}^{\infty} n\left\{\sum_{k=0}^{c}\sum_{i=0}^{c-k} p(i,n,k)\right\}.$$

5.2. **Reliability indices and busy period.** System manufacturers and software developers need a certain level of reliability in a model for the smooth functioning of their systems. The reliability measures of the system for the steady state will give the data which is expected for the system's improvement in the queueing model. Taleb and Aissani [21] have focused on the server's first failure time and its reliability. Upadhyaya [22] has studied a $M^X/G/1$ retrial line model with negative arriving clients. In [22], a point wise and steady state availability of the service provider in the queueing model is given. With the help of these studies, we incorporate the following measures:

(a) Reliability Indices:

To validate and confirm the analytical results of our model, we give the availability (A_s) and failure frequency (F_s) of the service provider are given by

$$A_s = 1 - \text{P[B]} = 1 - \left\{\sum_{j=0}^{\infty}\sum_{i=0}^{n-c} p(i,j,n)\right\};$$

(5.10)

$$F_s = \theta\text{P[B]} = \theta\left\{\sum_{j=0}^{\infty}\sum_{i=0}^{n-c} p(i,j,n)\right\}.$$

(b) Busy Period:

Let E[U] be the awaited length of the period that include beginning of the epoch when the new arrival finds the system and server unoccupied, to the epoch when the system and server are unoccupied again after service completion. Furthermore, by using the alternating renewal process, as the arrival parameter (λ) follows the geometric distribution, we get

(5.11) $$E[U] = \frac{1}{\lambda}\left[\frac{1}{p_{0,0,0}} - 1\right].$$

6. Numerical Illustration

A queueing system that is applicable in day-to-day life is worth studying. A model's authenticity is validated using numerical evaluation where we verify the trends of some basic parameters on queue length and system size evaluated for the system. Now, giving the practical application of our model in real world in sub-section 6.1 and providing the impact of some specific parameters on performance measures of the model in sub-section 6.2.

6.1. **Working of Automated car wash/service systems.** The practical application of the developed model can be seen in operational models. We are referring to a empirical problem consisting of a car wash/service system with number of machines to be three (i.e. $c = 3$). The incoming (arrival) rates when the garage/workshop is idle is considered to be $(\lambda) = 7$. The incoming cars get their data entered in the computer system and once the system allows, they may either get served immediately if the garage/workshop is idle with service rate $(\mu) = 14$ or they may have two options. Firstly, they can join the queue of cars and the owner will have to retry again (retrials) with retrial rate $(\alpha) = 6$ to get served. Secondly, they might resist joining a queue and can leave the workshop permanently $(\nu = 0.25)$.

As the garage/workshop is automated, the machines may face some sort of breakdown $(\theta = 1)$ while the car is served. We assume that the machines are mended immediately with repair rate $(r) = 3.5$ and once they are repaired, the service process resumes. Often these automated service stations share online feedback forms and provide re-service to unsatisfied clients (feedback). The probability for the server to get re-service is $\gamma = 0.2$.

We have framed a code to model this situation in MATLAB software and hence obtained the following results: $P[O] = 0.8196$, $P[B] = 0.1851$, $E[L] = 2.0390$, $A_s = 0.8149$, $F_s = 0.1851$. The results and values obtained here tells us that our model is much reliable and its failure rate is less. Thus the results are helpful in increasing the performance of the operational unit and pauses the overloading of production system. The system designers can rely on the values obtained above and can take advantage for their respective systems.

6.2. **Sensitivity Analysis.** A queueing model is efficient and helpful enough if it provides accurate results under stable condition. Thus the parametric values are selected in a way that the system remains under equilibrium. Now considering the same default parametric values as used in subsection 6.1. We plot curves for different number of servers in this subsection. In all the graphs, it can be noticed that with the increasing number of servers, the number of clients in the system decreases and this situation is closely related to the reality. By doing this, we aim to present the effect on mean system size by changing various critical system parameters λ, μ, α and θ.

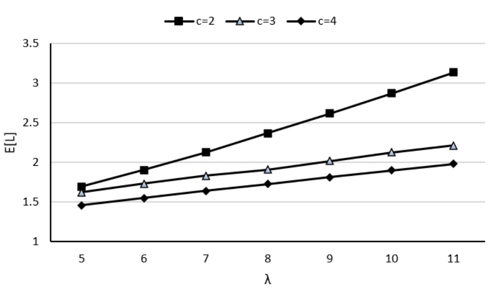

FIGURE 2. L_s vs λ for varying c.

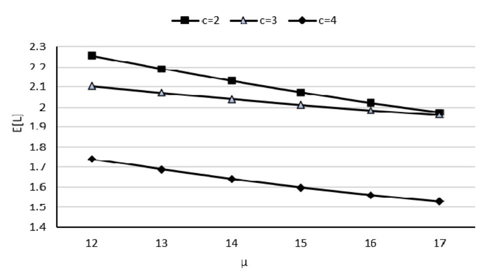

FIGURE 3. L_s vs μ for varying c.

Figure 2 reflects the increasing pattern in system size as the entering rate increases, while in figure 3 the system size slopes down with the increase in service rate. Clearly, figure 4 shows the decrease in system size with the increment in the retrial rate but in Figure 5 an increment in the system size with increase in the breakdown rate can be seen. All the graphs are in consistent to real life models and hence authenticate the results obtained for the model under study. Thus the system designers and decision makers should make efforts and choose these parameters in such a way that their respective systems become economical.

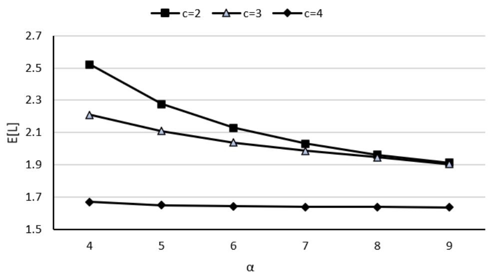

FIGURE 4. L_s vs α for varying c.

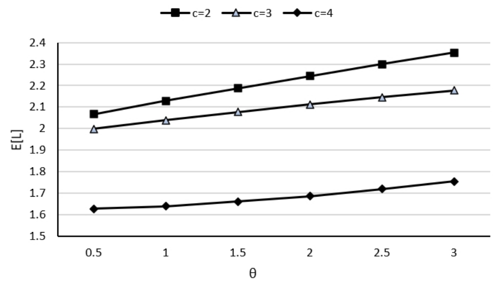

FIGURE 5. L_s vs θ for varying c.

7. COST OPTIMIZATION

Practically, the total operating cost of a system assumes a vital part in the investigation of many industrial systems. System architects and supervisors are typically keen on limiting the working expense per unit time to make the system more productive and profitable. To exhibit the relevance of the outcomes acquired in the past conversation, we foster an expected operating cost function per unit time for the queueing model which is defined as

$$TC = C_h E[L] + C_b P[O] + C_r P[B] + C_d \mu + C_s c \tag{7.12}$$

where the cost symbols corresponding to the cost function (per unit time) are:
C_h : Cost for holding each client,
C_b : Cost of service provider being occupied,
C_r : Cost of service provider being under repair,
C_d : Cost to provide a service,
C_s : Fixed cost to buy a service provider.
Consider the following minimization problem:

$$(CP)\; TC(r^*, \alpha^*) = \min TC(r, \alpha) \tag{7.13}$$

where $S : \{(r, \alpha) : r, \alpha \in (0.5, 6)\} \subset R^2$.

Solving a cost minimization problem of this kind using analytical methods is a tough task, as cost function TC is highly complex and non-linear. So we are going to adopt a soft computing technique to get the optimal cost.

7.1. **Particle Swarm Optimization (PSO).** Introduction of Particle swarm optimization (PSO) was provided by Kennedy [10] after inspecting the migratory behavior of swarms and birds. This method has got popular since the last decade in optimizing the cost of many queueing models. This algorithm works on the principle that the best positioned solution attract other possible values to get the optimal result in a particular search space. For a review of PSO on queueing models one can refer to Agarwal et al. [1]. In this survey paper they have discussed the application of PSO in optimization of unreliable retrial server queueing systems.

Taking into account the MATLAB algorithm for PSO discussed by Malik et al. [16], we have made required changes in the code and worked it for matrix geometric method for our system. An optimal cost for $c = 2, 3, 4$ with different values of the pair (θ, γ) to get the optimal point (r^*, α^*) is determined. For this purpose, four different cost sets (1-4) have been considered for the evaluation of minimum cost as

CostSet1: C_h= \$6; C_b= \$15; C_r= \$10; C_d= \$8; C_s= \$4;

CostSet2: C_h= \$6; C_b= \$15; C_r= \$10; C_d= \$4; C_s= \$8;
CostSet3: C_h= \$6; C_b= \$12; C_r= \$10; C_d= \$8; C_s= \$4;
CostSet4: C_h= \$6; C_b= \$12; C_r= \$10; C_d= \$4; C_s= \$8;

To obtain the optimal solution of the problem, the PSO technique has been executed. We get the optimal rates to be $(\mathbf{r}^*, \alpha^*) = [\mathbf{5.9226}, \mathbf{6}]$ where the minimum cost for the system corresponding to Cost Set1 is $\mathbf{TC}^* = \$\mathbf{147.5509}$ while that for Cost Set2 is $\mathbf{TC}^* = \$\mathbf{99.5509}$. Figures 6 and 7 represent the solution obtained using PSO for Cost Sets 1 and 2 respectively. Similarly, the convergence graphs for Cost Sets 3 and 4 can be obtained using the same MATLAB algorithm. Considering different cost sets with varying different cost factors to obtain the best cost for the system will help the system designers and manufacturers to develop a reliable system which cost-efficient too.

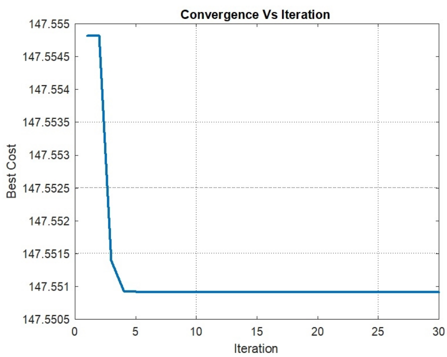

FIGURE 6. Cost vs Iteration using PSO for Cost Set 1.

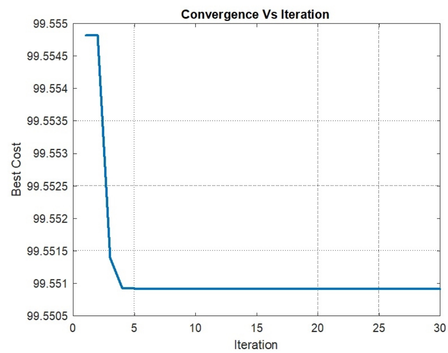

FIGURE 7. Cost vs Iteration using PSO for Cost Set 2.

The joint optimal values of parameters (r^*, α^*) obtained for all four cost sets by changing parameter values are summed up in Table 1 where it can be noted that the optimal cost slightly increases with increase in number of servers and increase in parameter values. We observe from Table 1 that the minimum cost is obtained for cost set 4 as compared with other cost sets for different parameters. Also, when we decrease the cost of occupied service provider, the overall cost of the system decreases irrespective of the server's fixed buying cost or cost to provide service. Thus a real life scenario can be interpreted where one is advised to make a one time investment by buying a good product (irrespective of high fixed cost) rather than getting its re-service regularly (increase overall cost).

Most of the authors working with multi-server queues like [2, 3, 4, 5, 6, 19, 23] have not studied the optimum cost for their systems. Very few authors have used Direct Search Method and Quasi-Newton method to optimize the system cost which can be seen in [7, 8, 9, 15]. These methods are time consuming and are prone to errors as they are based on larger calculations. In this paper, we have not only done the sensitivity and reliability analysis of the parameters but also the cost optimization of the system is done using a well known meta-heuristic technique PSO. Finding the optimal cost of the system using PSO makes it trustworthy as the technique, in larger space, is helpful to the mathematicians and scientists in getting the best optimal results. Its convergence rate is relatively faster and accurate compared to the other techniques.

TABLE 1. Evaluation of the optimal cost (in \$) and optimal values for (r^*, α^*) by varying the number of servers using PSO.

Number of server	Parameters	Cost Set1	Cost Set2	Cost Set3	Cost Set4
	(θ, γ)	(r^*, α^*), TC when $\lambda = 7,\ \mu = 14,\ \nu = 0.25$			
$c = 2$	(1,0.2)	[5.9223 6], 147.5509	[5.9226 6], 99.5509	[6 6], 144.9125	[6 6], 96.9125
	(2,0.2)	[3.8894 6], 148.0795	[3.8893 6], 100.0795	[6 6], 145.6270	[6 6], 97.6270
	(1,0.25)	[3.0583 6], 148.2037	[3.0583 6], 100.2037	[6 6], 145.6518	[6 6], 97.6518
$c = 3$	(1,0.2)	[0.8277 6], 149.6769	[0.8276 6], 105.6769	[1.3736 6], 147.7729	[1.3737 6], 103.7729
	(2,0.2)	[1.3401 6], 149.9597	[1.3397 6], 105.9597	[1.9279 6], 148.1150	[1.9277 6], 104.1150
	(1,0.25)	[0.8754 6], 150.2386	[0.8760 6], 106.2386	[1.4312 6], 148.3629	[1.4313 6], 104.3629
$c = 4$	(1,0.2)	[6 2.9836], 151.0648	[6 2.9836], 111.0648	[6 3.8364], 149.1718	[6 3.8363], 109.1718
	(2,0.2)	[6 3.7501], 151.1717	[6 3.7501], 111.1717	[6 4.9511], 149.2625	[6 4.9516], 109.2625
	(1,0.25)	[6 3.7262], 151.2821	[6 3.7258], 111.2821	[6 4.8012], 149.3997	[6 4.8009], 109.3997

8. CONCLUSION

Getting a cost efficient system helps system designers to improve profit and reduce delays in satisfying client's demand. The work done in the present investigation includes balking behavior of the entering clients. Our server, being machines, can breakdown at some point of time and are subject to repairs. The unsatisfied clients may opt for another service in the form of bernoulli feedback. This enables us to relate our model to real life situations where one desires a timely service. One can apply the results of the model to workplaces where distinct machinery systems work simultaneously such as automated vending machines, computing systems, mobile and cellular networks and many more.

Decision makers face challenges in controlling the sensitive parameters of a model. The reliability and numerical analysis of some critical parameters in this study helps system managers to develop a reliable system which reduces congestion problems. Efficient search approach is presented to obtain the optimal number of repair and retrial rates. The global optimal obtained using PSO makes our model reliable in real economical world. The model is applicable to many distinct digital systems such as packet switching networks, computing systems and many more. In future, development on repair policies i.e. considering delayed repair or active, passive breakdowns could be applied in this model. One can also think of applying characteristic behavior of clients like reneging with multi optional service on this model.

Acknowledgments. The authors would like to acknowledge the support provided by the deanship of research oversight & coordination at King Fahd University of Petroleum & Minerals, Saudi Arabia.

REFERENCES

[1] Agarwal, R.; Agarwal, D.; Upadhyaya, S. Study on Optimization of Unreliable Server Queueing Systems: A PSO Based Survey. In: Singh D. et al. (Eds.), *Proceedings of International Conference on Scientific and Natural Computing. Algorithms for Intelligent Systems*. Springer, Singapore (2021), 189–197.

[2] Aniyeri, R.; Nadar, C. R. A Multiphase Queuing System with Assorted Servers by using Matrix Geometric Method. *Int. J. Appl. Eng.* **12** (2017), no. 22, 12052-12059.

[3] Chakravarthy, S. R.; Ozkar, S.; Shruti. Analysis of $M/M/c$ retrial queue with thresholds, PH distribution of retrial times and unreliable servers. *Int. J. Appl. Math. Inform.* **39** (2021), no. 1-2, 173–196, doi:10.14317/jami.2021.173.

[4] Dudin, A.; Dudina, O.; Dudin, S.; Samouylov, K. Analysis of Multi-Server Queue with Self-Sustained Servers. *Mathematics* **9** (2021), no. 17, 2134. doi:10.3390/math9172134.

[5] Haddad, M. E.; Belarbi, F. Approximate anlysis of an unreliable M/M/c retrial queue with phase merging algorithm. *New trend math. sci.* **4** (2016), no. 3, 9–21. doi:10.20852/ntmsci.2016318801

[6] Jain, M.; Jain, A. Working vacations queueing model with multiple types of server breakdowns. *Appl. Math. Model.* **34** (2010), 1–13.

[7] Jain M.; Upadhyaya, S. Synchronous working vacation policy for finite-buffer multiserverqueueing system. *Appl. Math. Comput.* **217** (2011), no. 24, 9916–9932.

[8] Kadi, M.; Bouchentouf, A. A.; Yahiaoui, L. On a Multiserver Queueing System with Customers' Impatience Until the End of Service Under Single and Multiple Vacation Policies, *Appl. Appl. Math.* **15** (2020), no. 2, 740–763.

[9] Ke, J. C.; Wu, C. H.; Pearn, W. L. Multi-server retrial queue with second optional service: algorithmic computation and optimisation. *Int. J. Syst. Sci.* **42** (2011), no. 10, 1755–1769. doi: 10.1080/00207721003650209.

[10] Kennedy, J.; Eberhart, R. Particle swarm optimization. *In Proceedings of ICNN'95-international conference on neural networks, IEEE, Perth(Australia)* **4** (1995), 1942–1948.

[11] Khaskheli, S. A.; Kalwar, H. A.; Kalwar, M. A.; Marri, H. B.; Khan, M. A.; Nebhwani, M. Application of multi-server queueing model to analyze the queueing system of OPD during Covid-19 Pandemic: A case study. *Journal of Contemporary Issues in Business and Government* **27** (2021), no. 5. doi:10.47750/cibg.2021.27.05.094.

[12] Lakaour, L.; Aissani, D.; Aissanou, K. A.; Barkaoui, K.; Ziani, S. An unreliable single server retrial queue with collisions and transmission errors. *Commun. Stat. Theory Methods* **51** (2022), no. 4, 1085–1109. doi:10.1080/03610926.2020.1758943.

[13] Lan, S.; Tang, Y. An unreliable discrete-time retrial queue with probabilistic preemptive priority, balking customers and replacements of repair times. *AIMS* **5** (2020), no. 5, 4322-4344. doi:10.3934/math.2020276.

[14] Latouche, G.; Ramaswami, V. *Introduction to matrix analytic methods in stochastic modeling (ASA-SIAM Series on Statistics and Applied Probability)* PA: Society for Industrial and Applied Mathematics, Philadelphia (1999).

[15] Lin, C. H.; Ke, J. C. On the multi-server retrial queue with geometric loss and feedback: computational algorithm and parameter optimization". *Int. J. Comput. Math.* **88** (2011), no. 5, 1083–1101. doi: 10.1080/00207160.2010.498884.

[16] Malik, G.; Upadhyaya, S.; Sharma, R. Particle Swarm Optimization and Maximum Entropy Results for $M^X/G/1$ Retrial G-Queue with Delayed Repair. *Int. J. Math. Eng. Manag. Sci.* **6** (2021), no. 2, 541–563.

[17] Neuts, M. F. *Matrix geometric solutions in stochastic models: An algorithmic approach.* The John Hopkins University Press, Baltimore (1981).

[18] Prasad, S. V.; Mahaboob, B.; Donthi, R.; Praveen, J. P. A Comparative Study Between Multi Queue Multi Server And Single Queue Multi Server Queuing System. *Int. J. Sci. Res.*, **8** (2019), no. 12.

[19] Raiah, L.; Oukid, N. An $M/M/2$ retrial queue with breakdowns and repairs. *Rom. J. Math. Comput. Sci.* **7** (2017), no. 1, 11–20.

[20] Rajadurai, P.; Chandrasekaran, V. M.; Saravanarajan, M. C. Analysis of an unreliable retrial G-queue with working vacations and vacation interruption under Bernoulli schedule. *Ain Shams Eng. J.* **9** (2018), no. 4, 567–580.

[21] Taleb, S.; Aissani, A. Preventive maintenance in an unreliable M/G/1 retrial queue with persistent and impatient customers. *Ann. Oper. Res.* **247** (2016), no. 1, 291–317. doi:10.1007/s10479-016-2217-1.

[22] Upadhyaya, S. Investigating a general service retrial queue with damaging and licensed units: an application in local area networks. *Opsearch* **57** (2020), 716–745. doi:10.1007/s12597-020-00440-1.

[23] Subramanian, M. G.; Ayyappan; Sekar, G. $M/M/c$ retrial queueing with breakdown and repair of services. *Asian J. Math.* **4** (2011), no. 4, 214-223. doi: 10.3923/ajms.2011.214.223.

[24] Upadhyaya, S.; Kushwaha, C. Performance prediction and ANFIS computing for unreliable retrial queue with delayed repair under modified vacation policy. *Int. J. Math. Oper. Res.* **17** (2020), no. 4, 437–466.

[25] Wang, J.; Cao, J.; Li, Q. Reliability Analysis of the Retrial Queue with Server Breakdowns and Repairs. *Queueing Syst.* **38** (2001) 363–380. doi:10.1023/A:1010918926884.

[1]AMITY INSTITUTE OF APPLIED SCIENCES, AMITY UNIVERSITY
SECTOR 125, NOIDA, UP, INDIA
Email address: supadhyay@amity.edu, dagarwal1@amity.edu, radhika6696@gmail.com

[2]DEPARTMENT OF MATHEMATICS
KING FAHD UNIVERSITY OF PETROLEUM AND MINERALS
DHAHRAN 31261, SAUDI ARABIA

[3]CENTER FOR INTELLIGENT SECURE SYSTEMS
KING FAHD UNIVERSITY OF PETROLEUM AND MINERALS
DHAHRAN 31261, SAUDI ARABIA
Email address: drizhar@kfupm.edu.sa

[4]INTERDISCIPLINARY CENTER OF SMART MOBILITY AND LOGISTICS
KING FAHD UNIVERSITY OF PETROLEUM AND MINERALS
DHAHRAN, 31261, SAUDI ARABIA
Email address: homidan@kfupm.edu.sa,abdulkarim.ibrahim@kfupm.edu.sa