New Trends in Mathematical Sciences
http://dx.doi.org/10.20852/ntmsci.2016318801

Approximate anlysis of an unreliable $M/M/c$ retrial queue with phase merging algorithm

Meriem El Haddad and Faiza Belarbi

Laboratory Statistics and Stochastic Processes, B.P. 89, Djillali Liabes University, Sidi Bel Abbes, Algeria.

Received: 22 November 2015, Accepted: 28 March 2016
Published online: 19 June 2016.

Abstract: In this paper, we investigate an approximate analysis of unreliable $M/M/c$ retrial queue with $c \geq 3$ in which all servers are subject to breakdowns and repairs. Arriving customers that are unable to access a server due to congestion or failure can choose to enter a retrial orbit for an exponentially distributed amount of time and persistently attempt to gain access to a server, or abandon their request and depart the system. Once a customer is admitted to a service station, he remains there for a random duration until service is complete and then depart the system. However, if the server fails during service, i.e., an active breakdown, the customer may choose to abandon the system or proceed directly to the retrial orbit while the server begins repair immediately. In the unreliable model, there are no exact solutions when the number of servers exceeds one. Therefore, we seek to approximate the steady-state joint distribution of the number of customers in orbit and the status of the c servers for the case of Markovian arrival and service times. Our approach to deriving the approximate steady-state probabilities employs a phase-merging algorithm.

Keywords: Retrial queue, Multi server, Breakdown and repair of service, Phase merging algorithm

1 Introduction

Queueing systems in which arriving customers who find all servers and waiting positions (if any) occupied may retry for service after a period of time are called retrial queues. Retrial queues have been widely used to model many problems in telephone switching systems, telecommunication networks, computer networks and computer systems. For detailed survey of retrial queues and bibliographical information see Falin [8], Artalejo [3], [4], [5], monograph by Falin and Templeton [9] and Gomez-Corral [2]. Retrial queues with unreliable servers have been studied by Kulkarni and Choi [11], Aissani and Artalejo [1] and Brian [6]. There are a great number of numerical and approximations methods available. In this paper we will place more emphasis on the solutions by phase merging algorithm outlined by Korolyuk [10].

Our paper is organized as follows. In section 2, we provide the formal model description and state the assumptions that are needed to implement the approximation procedure. In sections 3 and 4, the algorithm is formally reviewed. In section 5, we give an illustrate example. Applying it to our model, we derive approximations for the steady-state probabilities and several standard queueing performance measures. In section 6, we assess the quality of the approximations by comparing results with those obtained using direct truncation method by M.G. Subramanian, Ayyappan and G. Sekar [12].

2 Model description

We consider an unreliable $M/M/c$ retrial queuing system in which customers arrive according to a Poisson process with rate λ ($\lambda > 0$). If upon arrival, the customer finds one of the servers idle and not failed, he occupies him immediately.

* Corresponding author e-mail: faiza_belarbi@yahoo.fr

However, if customer does not find any available servers (busy or failed) may join the retrial orbit with probability q_a or abandons the system with probability $1-q_a$ $(0 \leq q_a \leq 1)$. Customers who enter the orbit wait for an exponentially distributed time with rate $\theta(\theta > 0)$ before attempting to access a server again. The service times are assumed to be exponentially distributed with mean $1/\mu$. Failures for the c servers occur independently via a Poisson process with rate $\xi(\xi > 0)$ and the repair times for each server are exponentially distributed with rate $\alpha(\alpha > 0)$. Furthermore, interarrival times, service times, retrial times, interfailure times and repair times are mutually independent.

This model accounts for both active and idle breakdowns. For active breakdowns, the customer that is preempted by a server failure enters the retrial orbit with probability q_f or abandons the service with probability $1-q_f$.

The state of the system can be described by a trivariate stochastic process in continuous time, $\{(N(t),B(t),F(t)) : t \geq 0\}$, where $N(t)$ is the number of customers in the orbit at time t, $B(t)$ is the number of busy servers at time t and $F(t)$ is the number of failed servers at time t.

Since all the random times are exponentially distributed, the stochastic process is a continuous-time Markov chain (CTMC) on the state space $S = \{(i, j,k) : i \geq 0, j+k \leq c,\ j,k \in \{0,1,2,...c\}\}$. We assume that as $t \to \infty$ the steady-state distribution of $\{(N(t),B(t),F(t)) : t \geq 0\}$ exists.

Define $p(i, j,k)$ as the limiting probability that the system is in the state (i, j,k) where $(i, j,k) \in S$. Defined mathematically,

$$p(i, j,k) = \lim_{t\to\infty} P(N(t) = i,B(t) = j,F(t) = k).$$

Note that a set of only $\sum_{n=1}^{c-1} n+\ (2c+1)$ pairs of (j,k) are needed to completely characterize the status of the servers at any time. Let us introduce the following algorithm which gives the ordered pairs represents the number of busy servers and the number of failed servers, respectively.

```
Var c: Integer;
begin
read (c) ;
for j = 0 to c Do
for k = 0 to c − i Do
write (j,k);
end.
```

3 Main results

• *Mean Orbit Length*

For the retrial queueing model with c servers, it was shown that the steady-state distribution is Poisson with parameter $\hat{\lambda}/\hat{\theta}$. Therefore, the long-run mean orbit length is approximately the expected value of this Poisson random variable. Denoting N as the steady-state number of customers in orbit, the mean orbit size is approximated by

$$E[N] \approx \frac{\hat{\lambda}}{\hat{\theta}} = \frac{\lambda q_a p_{0,c} + \sum\limits_{\substack{j+k<c \\ j\neq 0}} j\xi q_f p_{j,k} + \sum\limits_{\substack{j+k=c \\ j\neq 0}} (\lambda q_a + j\xi q_f) p_{j,k}}{\theta \sum\limits_{j+k\neq c} p_{j,k}}.$$

• *Mean Number of customer in Service*

Let N_s be defined as the random number of customers at the servers, the approximate expression for the expected number of customers at the servers can be computed using the approximate steady-state joint probabilities derived in the last step of the algorithm and it is given by

$$E[N_s] = \sum_{i=0}^{\infty}(\sum_{j\neq 0} p(i,j,k)).$$

• *Steady-State System Size and Sojourn Time*

Let us define L as the steady-state number of customers in the system, to calculate it, we simply sum the expressions for $E[N]$ and $E[N_s]$.
The steady-state mean sojourn time, W, follows directly from Little's law.

$$\begin{aligned} L &\approx E[N]+E[N_s] \\ W &\approx \tfrac{L}{\lambda}. \end{aligned}$$

• *Total Expected Time in Orbit*

Due to server failure and blocking when making a retrial attempt, customers may enter the orbit more than once. Therefore, the expected time a customer spends in orbit is $1/\theta$ times the expected number of retrial attempts before gaining access to the server. Define Y as the random number of retrials a customer performs until it gains access to a server. Then Y is a geometric random variable with parameter p_u, the steady-state probability that at least one server is available. The approximation for p_u is given by

$$p_u = \sum_{i=0}^{\infty}(\sum_{j+k\neq c} p(i,j,k)).$$

The expected number of retrials performed, $E[Y]$, is therefore, $1/p_u$ and letting W_r be the random time spent in orbit once they are there we have,

$$E[W_r] \approx (\theta p_u)^{-1}.$$

Proof. To prove these results, a phase-merging algorithm developed by Korolyuk [10] and Courtois [7] will be employed and is summarized in the following sections.

3.1 The Phase-Merging Algorithm

Beginning with a CTMC on a state space that completely describes a retrial queueing system. The objective of the phase-merging algorithm is to approximate the steady-state probability distribution of $\{(N(t),B(t),F(t)) : t \geq 0\}$ by approximating the conditional probability distribution of the status of the servers and by approximating the marginal probability distribution of the number of customers in orbit. The algorithm proceeds by partitioning the state space into disjoint and mutually exhaustive sets that correspond to levels of the orbit. Each level is analysed as a CTMC from which the approximate conditional probabilities are obtained. Each level itself is subsequently considered as a state of an aggregated CTMC where the transition rates between levels correspond to customers entering or leaving the orbit. Analyzing this system of 'macrostates' yields the approximate marginal probability distribution of the number of customers in the orbit. The product of the conditional and marginal probabilities is, therefore, the approximate joint probability distribution of the level of the orbit and status of the servers. Using this joint distribution, we then approximate standard queueing performance measures.

To begin, we reduce the dimensionality of the state space by defining $X(t)$ as the status of the servers at time t (outlined in Table 1), such that $X(t) \in \{1,2,3,4,..., \sum_{n=1}^{c-1} n + (2c+1)\}$.

State (j,k)	Index
(0,0)	1
(1,0)	2
(2,0)	3
.	.
.	.
.	.
(c,0)	$c+1$
.	.
.	.
.	.
(0,1)	
(0,2)	
.	.
.	.
.	.
(0,c)	$\sum_{n=1}^{c-1} n + (2c+1)$

Table 1: Substitution for server status.

The algorithm proceeds in the following manner. First, partition the state space, S, into disjoint sets that are conditional upon i such that,

$$S \equiv \bigcup_{i=0}^{\infty} S_i \ , \quad S_i \cap S_j = \emptyset, \quad i \neq j$$

where $S_i = (i,l) : l = 1,2,..., \sum_{n=1}^{c-1} n + (2c+1), i \geq 0$. This step results in an infinite number of classes (or levels) which can be analyzed individually.

Next we obtain the steady-state distribution of each class or level, S_i, by determining the infinitesimal generator matrix, Q_i defined by

$$q_{l,m} = \begin{cases} q_{(i,l),(i,m)}, & l \neq m \\ -\sum_{l \neq m} q_{(i,l)(i,m)}, & l = m \\ 0, & \text{otherwise} \end{cases}$$

Denote by $q_{l/i}$ the steady-state conditional probability that the status of the servers is state l, given there are i customers in orbit, $i \geq 0, l = 1,2,..., \sum_{n=1}^{c-1} n + (2c+1)$. Letting $p_i = [p_{l|i}]$, we solve the system of equations $p_i Q_i = 0$ and $p_i e = 1$ (where e is a column vector of ones) to obtain the approximate probability distribution.

Following this, we merge, or aggregate, all states within the class S_i, into one state corresponding to the level of orbit, i. These 'macrostates ' form the overall state space of the merged model which are defined as $\hat{S} \equiv \{i \ : i \geq 0\}$. The infinitesimal generator, Q_M, of the merged model is

$$q_{i,j} \equiv \sum_{(i,l)\in S_i} p_{l/i}\Big(\sum_{(j,m)\in S_j} q_{(i,l),(i,m)}\Big).$$

Denote π_i as the marginal probability that there are i customers in the orbit. Letting the infinite-dimensional vector $\pi \equiv [\pi_0, \pi_1, \pi_2, \ldots]$, we solve the system of equations $\pi Q_M = 0$ and $\pi e = 1$ to obtain the approximate steady-state marginal probabilities.

$$p(i,l) \approx \hat{p}(i,l) = p_{l|i} \times \pi_i, \qquad i \geq 0,\ \ l = 0,1,2,\ldots,\sum_{n=1}^{c-1} n + \ (2c+1).$$

Making use of these joint probabilities, we can obtain approximations for the performance measures for the unreliable c-servers retrial queue.

To illustrate the algorithm, we consider a reliable $M/M/c$ retrial queue whose customers arrive according to a Poisson process with rate λ. Each customer brings an exponential service requirement with mean time $1/\mu$.

Customers who find the c servers busy enter the orbit with probability s or leave the system with probability $1-s$. The time between retrials is exponentially distributed with mean $1/\theta$. All times are assumed to be mutually independent. Define the continuous-time stochastic process as $\{(N(t), B(t)) : t \geq 0\}$ where $N(t)$ is the number of customers in the orbit at time t and $B(t)$ is the number of busy servers at time t. The process is a CTMC on the state space $S = \{(i, j) : i \geq 0, j = 0,1,2,\ldots c\}$. For the purpose of illustrating the phase-merging algorithm, we will assume the system is stable and denote $p(i, j) = \lim_{t\to\infty} P(N(t) = i, B(t) = j)$ as the limiting probability that the system is in state $(i, j), i \geq 0, j = 0,1,2,\ldots c$. Figure 1.

To make use of the algorithm we proceed as follows: First, partition the state space into individual levels where the index of each level corresponds to the number of customers in the orbit. Denote this as class S_i for level $i, i \geq 0$. Note that each class has an identical structure and, therefore, the generator matrices, $\mathbf{Q}_i$ are identical for all $i \geq 0$. This fact will be extremely useful for analyzing the case of unreliable servers.

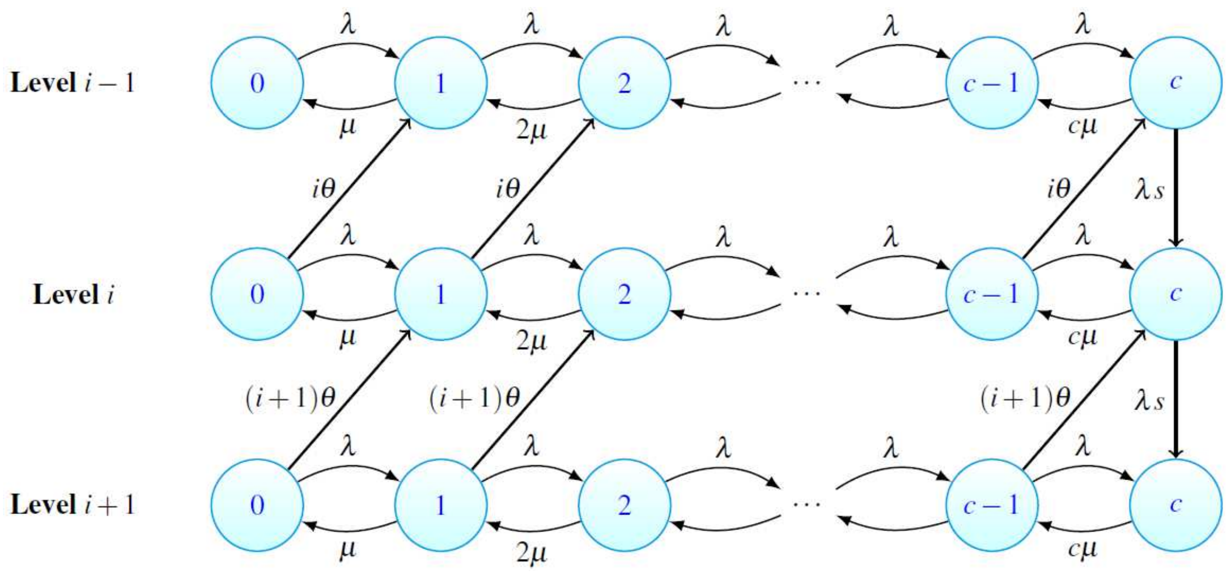

Fig. 1: Transition rate diagram for a reliable $M/M/c$ retrial queue.

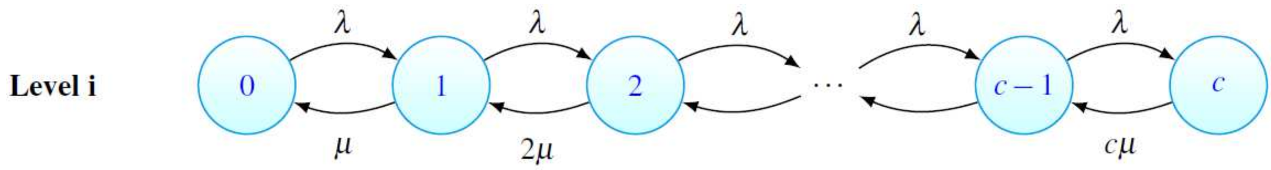

Fig. 2: Transition rate diagram for level i: reliable $M/M/c$ retrial queue .

Next we compute the steady-state conditional distribution of the status of the servers given there are i customers in orbit. Denote these probabilities by $p_{j|i}, j = 0,1,2,...c$, and it is easy to obtain the following conditional probabilities for all $i \geq 0$ by solving the balance equation system using the Formal Maple Calculation Software.

The next step is to aggregate the states of each class to form a series of merged states, i where $i \geq 0$ and investigate the transitions between them. The elements of the infinitesimal generator matrix for the merged states are.

$$q_{i,j} = \begin{cases} \lambda s p_{c|i}, & i \geq 0, \quad j = i+1 \\ i\theta(\sum_{j=0}^{c-1} p_{j/i}), & i \geq 0, \quad j = i-1 \\ -[\lambda s p_{c|i} + i\theta(\sum_{j=0}^{c-1} p_{j/i})], & i = j \\ 0, & \text{otherwise.} \end{cases}$$

Using the substitutions, $\hat{\lambda} = \lambda s p_{c|i}$ and $\hat{\theta} = \theta(\sum_{j=0}^{c-1} p_{j,k})$, we see that the analysis of this system is analogous to the $M/M/\infty$ queueing system. Thus, defining the steady-state marginal probability vector as $\pi = [\pi_0, \pi_1, \pi_2, ...]$ we have,

$$\pi_i = \frac{1}{i!}(\frac{\lambda}{\theta})^i e^{-\lambda/\theta}, \qquad i \geq 0.$$

Finally, the approximate steady-state distribution of $\{(N(t), B(t)) : t \geq 0\}$ is given by

$$p(i,j) \approx \hat{p}(i,j) = p_{j|i} \times \pi_i = \frac{p_{j|i}}{i!}(\frac{\lambda}{\theta})^i e^{-\lambda/\theta}, \quad i \geq 0, \quad j = 0,1,2,...c.$$

3.2 Approximation Using the Phase-Merging Algorithm

We apply the phase-merging algorithm described in [10] and [7] to the unreliable $M/M/c$ retrial queue. Recall that the interarrival times, service and repair times, time between failures and time between retrials are all exponentially distributed with the parameters defined previously. Since the number of customers in the orbit can theoretically reach infinity, the state space of the system can be partitioned into a countable number of classes. As noted previously, the state space S is partitioned as the countable union

$$S = \bigcup_{i=0}^{\infty} S_i \;, \quad S_i \cap S_j = \emptyset, \quad i \neq j$$

where $S_i = (i,l) : l = 1,2,..., \sum_{n=1}^{c-1} n + \ (2c+1), i \geq 0$. Just as in the reliable $M/M/c$ retrial queue, each class is identical in structure so that only one class needs to be analyzed.

To determine the steady-state probabilities for the class S_i define the stochastic process $\{(B(t), F(t)) : t \geq 0\}$ where $B(t)$

represents the number of busy servers and $F(t)$ represents the number of failed servers at time t. Clearly, the process is a **CTMC** on the state space E defined previously. Using the notation defined in Table 1 we denote $p_{l|i}$ as the limiting conditional probability of the servers being in state l given that there are i customers in orbit,

$$p_{l|i} = \lim_{t\to\infty} P(X(t) = l/N(t) = i), l = 1,2,..., \sum_{n=1}^{c-1} n + \ (2c+1).$$

For each $i \geq 0$, the transition rates for this process are described in the generator matrix Q_i.

Let p_i be the steady-state conditional probability vector where $p_i = [p_{l|i}],\ l = 1,2,..., \sum_{n=1}^{c-1} n + \ (2c+1)$.
Solving the equations $p_iQ_i = 0$ and $p_ie = 1$ yields the following system,

$$\begin{aligned}
(\lambda + c\xi)p_{(0,0)|i} &= \mu p_{(1,0)|i} + \alpha p_{(0,1)|i} \\
(\lambda + (c-1)\xi + \mu)p_{(1,0)|i} &= \lambda p_{(0,0)|i} + 2\mu p_{(2,0)|i} + \alpha p_{(1,1)|i} \\
(\lambda + (c-2)\xi + 2\mu)p_{(2,0)|i} &= \lambda p_{1,0|i} + 3\mu p_{(3,0)|i} + \alpha p_{(2,1)|i} \\
&\vdots \\
(\lambda + \xi + (c-1)\mu)p_{(c-1,0)|i} &= \lambda p_{(c-2,0)|i} + c\mu p_{(c,0)|i} + \alpha p_{(c-1,1)|i} \\
c\mu p_{(c,0)|i} &= \lambda p_{(c-1,0)|i} \\
(\lambda + (c-(j+k))\xi + j\mu + k\alpha)p_{j,k|i} &= (j+1)\mu p_{j+1,k|i} + (k+1)\alpha p_{j,k+1|i} + \lambda p_{j-1,k|i} \\
&\quad + (c-(j+k-1))\xi p_{j,k-1|i} \qquad \text{with } j+k<c,\ j\neq 0,\ k\neq 0 \\
(j\mu + k\alpha)p_{j,k|i} &= \lambda p_{j-1,k|i} + (c-(j+k-1))\xi p_{j,k-1|i} \text{with } j+k=c,\ j\neq 0,\ k\neq 0 \\
(\lambda + (c-1)\xi + \alpha)p_{(0,1)|i} &= c\xi p_{(0,0)|i} + \mu p_{(1,1)|i} + 2\alpha p_{(0,2)|i} \\
(\lambda + (c-2)\xi + 2\alpha)p_{(0,2)|i} &= (c-1)\xi p_{0,1|i} + \mu p_{(1,2)|i} + 3\alpha p_{(0,3)|i} \\
(\lambda + (c-3)\xi + 3\alpha)p_{(0,3)|i} &= (c-2)\xi p_{0,2|i} + \mu p_{(1,3)|i} + 4\alpha p_{(0,4)|i} \\
&\vdots \\
(\lambda + \xi + (c-1)\alpha)p_{(0,c-1)|i} &= 2\xi p_{0,c-2|i} + \mu p_{(1,c-1)|i} + c\alpha p_{(0,c)|i} \\
c\alpha p_{(0,c)|i} &= \xi p_{(0,c-1)|i} \\
\sum_{l=1}^{10} p_{l|i} &= 1
\end{aligned}$$

The solutions to the conditional probabilities are obtained for all $i \geq 0$ by solving the balance equations system using the Formal Maple Calculation Software.

Aggregating the states of each class S_i yields a system of macro-states which denote as $i, i \geq 0$. The rates of transitions between the "macro-states" are expressed in the infinitesimal generator matrix with elements,

$$q_{i,j} = \begin{cases} \lambda q_a p_{0,c} + \sum\limits_{\substack{j+k<c \\ j\neq 0}} j\xi q_f p_{j,k} + \sum\limits_{\substack{j+k=c \\ j\neq 0}} (\lambda q_a + j\xi q_f)p_{j,k}, & i \geq 0,\ \ j = i+1 \\ i\theta(\sum\limits_{j+k\neq c} p_{j,k}), & i \geq 0,\ \ j = i-1 \\ \lambda q_a p_{0,c} + \sum\limits_{\substack{j+k<c \\ j\neq 0}} j\xi q_f p_{j,k} + \sum\limits_{\substack{j+k=c \\ j\neq 0}} (\lambda q_a + j\xi q_f)p_{j,k} + i\theta(\sum\limits_{j+k\neq c} p_{j,k}), & i = j \\ 0 & \text{otherwise}. \end{cases}$$

To simplify the analysis of the merged states we use the following substitutions for $i, i \geq 0$

$$\hat{\lambda} = \lambda q_a p_{0,c} + \sum_{\substack{j+k<c \\ j\neq 0}} j\xi q_f p_{j,k} + \sum_{\substack{j+k=c \\ j\neq 0}} (\lambda q_a + j\xi q_f) p_{j,k}$$
$$\hat{\theta} = \theta \sum_{j+k\neq c} p_{j,k}.$$

Making the substitutions, the elements of the generator matrix are,

$$q_{i,j} = \begin{cases} \hat{\lambda}, & i \geq 0, \quad j = i+1 \\ i\hat{\theta}, & i \geq 0, \quad j = i-1 \\ -[\hat{\lambda} + i\hat{\theta}], & i = j \\ 0, & \text{otherwise} \end{cases}.$$

This new model is a state dependent birth-and-death process, the analysis of which is analogous to the $M/M/\infty$ queueing system. Using the method of arc cuts, we recursively solve for the steady-state probability vector, $\pi = [\pi_0, \pi_1, \pi_2, ...]$

$$\begin{array}{l} \hat{\lambda}\pi_0 = \hat{\theta}\pi_1 \Rightarrow \pi_1 = \frac{\hat{\lambda}}{\hat{\theta}}\pi_0 \\ \hat{\lambda}\pi_1 = 2\hat{\theta}\pi_2 \Rightarrow \pi_2 = \frac{1}{2}(\frac{\hat{\lambda}}{\hat{\theta}})^2\pi_0 \\ \hat{\lambda}\pi_2 = 3\hat{\theta}\pi_3 \Rightarrow \pi_3 = \frac{1}{6}(\frac{\hat{\lambda}}{\hat{\theta}})^3\pi_0 \\ \hat{\lambda}\pi_3 = 4\hat{\theta}\pi_4 \Rightarrow \pi_4 = \frac{1}{24}(\frac{\hat{\lambda}}{\hat{\theta}})^4\pi_0 \end{array}.$$

Continuing inductively, it can easily be shown that,

$$\pi_i = \frac{1}{i!}(\frac{\hat{\lambda}}{\hat{\theta}})^i \pi_0, \; i \geq 0. \tag{1}$$

Using the normalization equation , $\sum_{j=0}^{\infty} \pi_j = 1$, the solution for π_0 is obtained by

$$\begin{array}{r} \pi_0 + \frac{1}{2}(\frac{\hat{\lambda}}{\hat{\theta}})^2\pi_0 + \frac{1}{6}(\frac{\hat{\lambda}}{\hat{\theta}})^3\pi_0 + ... = 1 \\ \pi_0(1 + \frac{\hat{\lambda}}{\hat{\theta}} + \frac{1}{2}(\frac{\hat{\lambda}}{\hat{\theta}}) + \frac{1}{6}(\frac{\hat{\lambda}}{\hat{\theta}})^3\pi_0 + ...) = 1 \end{array}$$

$$\pi_0(\sum_{j=0}^{\infty} \frac{1}{j!}(\frac{\hat{\lambda}}{\hat{\theta}})^j) = 1. \tag{2}$$

The infinite series of (2) is the Maclaurin power series expansion for $e^{-\hat{\lambda}/\hat{\theta}}$. Thus, we see that

$$\pi_0 = e^{\hat{\lambda}/\hat{\theta}}.$$

Substituting π_0 into Equation (1) we have the following expression,

$$\pi_i = \frac{1}{i!}(\frac{\hat{\lambda}}{\hat{\theta}})^i e^{-\hat{\lambda}/\hat{\theta}}$$

which is the probability mass function for a poisson distributed random variable with rate parameter $\frac{\hat{\lambda}}{\hat{\theta}}$.

Finally, we approximate the steady-state distribution of $\{(N(t), X(t)) : t \geq 0\}$ by

$$\begin{aligned} p(i,l) \approx \hat{p}(i,l) &= p_{l|i} \times \pi_i \\ &= \frac{p_{l|i}}{i!}(\frac{\lambda}{\theta})^i e^{-\lambda/\theta}, \qquad i \geq 0,\ \ l = 0,1,2,\ldots,\sum_{n=1}^{c-1} n+\ (2c+1). \end{aligned}$$

4 Illustrative example: $M/M/3$.

We reduce the dimensionality of the state space by defining $X(t)$ as the status of the servers at time t, such that $X(t) \in \{1,2,3,4,5,6,7,8,9,10\}$.

State (j,k)	Index
(0,0)	1
(1,0)	2
(2,0)	3
(3,0)	4
(1,1)	5
(2,1)	6
(1,2)	7
(0,1)	8
(0,2)	9
(0,3)	10

Table 2: Substitution for server status.

For each $i \geq 0$, the transition rates for this process are described in the following generator matrix Q_i.

$$Q_i = \begin{pmatrix} -(\lambda+3\xi) & \lambda & 0 & 0 & 0 & 0 & 0 & 3\xi & 0 & 0 \\ \mu & -(\lambda+\mu+2\xi) & \lambda & 0 & 2\xi & 0 & 0 & 0 & 0 & 0 \\ 0 & 2\mu & -(\lambda+2\mu+\xi) & \lambda & 0 & \xi & 0 & 0 & 0 & 0 \\ 0 & 0 & 3\mu & -3\mu & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \alpha & 0 & 0 & -(\lambda+\mu+\alpha+\xi) & \lambda & \xi & \mu & 0 & 0 \\ 0 & 0 & \alpha & 0 & 2\mu & -(\alpha+2\mu) & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 2\alpha & 0 & -(2\alpha+\mu) & 0 & \mu & 0 \\ \alpha & 0 & 0 & 0 & \lambda & 0 & 0 & -(\lambda+2\xi) & 2\xi & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \lambda & 2\alpha & -(\lambda+\xi+2\alpha) & \xi \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 3\alpha & -3\alpha \end{pmatrix}.$$

Let p_i be the steady-state conditional probability vector where $p_i = [p_{l|i}],\ l = 1,2,\ldots,10$. Solving the equations $p_i Q_i = 0$ and $p_i e = 1$ yields the following system,

$$(\lambda+3\xi)p_{1|i} = \mu p_{2|i} + \alpha p_{8|i} \tag{3}$$

$$(\lambda+2\xi+\mu)p_{2|i} = \lambda p_{1|i} + 2\mu p_{3|i} + \alpha p_{5|i} \tag{4}$$

$$(\lambda+\xi+2\mu)p_{3|i} = \lambda p_{2|i} + 3\mu p_{4|i} + \alpha p_{6|i} \tag{5}$$

$$3\mu p_{4|i} = \lambda p_{3|i} \tag{6}$$

$$(\lambda+\xi+\mu+\alpha)p_{5|i} = 2\mu p_{6|i} + 2\alpha p_{7|i} + \lambda p_{8|i} + 2\xi p_{2|i} \tag{7}$$

$$(2\mu+\alpha)p_{6|i}=\lambda p_{5|i}+\xi p_{3|i} \tag{8}$$

$$(\mu+2\alpha)p_{7|i}=\lambda p_{9|i}+2\xi p_{5|i} \tag{9}$$

$$(\lambda+2\xi+\alpha)p_{8|i}=3\xi p_{1|i}+\mu p_{5|i}+2\alpha p_{9|i} \tag{10}$$

$$(\lambda+\xi+2\alpha)p_{9|i}=2\xi p_{8|i}+\mu p_{7|i}+3\alpha p_{10|i} \tag{11}$$

$$3\alpha p_{10|i}=\xi p_{9|i} \tag{12}$$

$$\sum_{l=1}^{10} p_{l|i}=1. \tag{13}$$

Replacing Equation (11) with the normalization equation (13), the solution to the conditional probabilities are obtained for all $i\geq 0$, using the Maple Calculation Software

$$\begin{aligned}p_{1/i}=D^{-1}(6(&-4\alpha\xi^3+2\mu\xi^3-2\xi^2\alpha\lambda+5\mu\xi^2\lambda-4\alpha^2\xi^2-4\alpha\mu\xi^2+7\mu^2\xi^2+7\mu^3\xi+4\lambda^2\mu\xi+9\lambda\alpha\mu\xi\\&+10\alpha\mu^2\xi+4\alpha^2\mu\xi+11\lambda\mu^2\xi+4\lambda\mu^3+3\lambda^2\mu^2+7\alpha^2\mu^2+9\alpha\mu^2\lambda+2\alpha^3\mu+7\alpha\mu^3+\lambda^3\mu\\&+4\lambda^2\alpha\mu+5\alpha^2\mu\lambda+2\mu^4)\alpha^3\mu^2)\end{aligned}$$

$$\begin{aligned}p_{2/i}=D^{-1}(6(&2\mu^3+9\mu^2\xi+7\alpha\mu^2+4\lambda\mu^2+16\mu\xi^2+15\alpha\mu\xi+15\lambda\mu\xi+9\lambda\alpha\mu+7\mu\alpha^2+3\mu\lambda^2+6\xi^3\\&+\lambda^3+11\lambda\xi^2+10\alpha\xi^2+14\alpha\lambda\xi+6\lambda^2\xi+6\alpha^2\xi+5\alpha^2\lambda+4\lambda^2\alpha+2\alpha^3)\lambda\alpha^3\mu^2)\end{aligned}$$

$$\begin{aligned}p_{3/i}=D^{-1}(3(&2\mu^3+4\lambda\mu^2+11\mu^2\xi+7\alpha\mu^2+9\lambda\alpha\mu+18\mu\lambda\xi+20\alpha\mu\xi+3\mu\lambda^2+7\mu\alpha^2+24\mu\xi^2+5\alpha^2\lambda\\&+8\alpha^2\xi+17\alpha\lambda\xi+7\lambda^2\xi+4\lambda^2\alpha+16\lambda\xi^2+18\alpha\xi^2+2\alpha^3+\lambda^3+12\xi^3)\lambda^2\alpha^3\mu)\end{aligned}$$

$$\begin{aligned}p_{4/i}=D^{-1}((&2\mu^3+4\lambda\mu^2+11\mu^2\xi+7\alpha\mu^2+9\lambda\alpha\mu+18\mu\lambda\xi+20\alpha\mu\xi+3\mu\lambda^2+7\mu\alpha^2+24\mu\xi^2+5\alpha^2\lambda\\&+8\alpha^2\xi+17\alpha\lambda\xi+7\lambda^2\xi+4\lambda^2\alpha+16\lambda\xi^2+18\alpha\xi^2+2\alpha^3+\lambda^3+12\xi^3)\lambda^3\alpha^3)\end{aligned}$$

$$\begin{aligned}p_{5/i}=D^{-1}(6(&6\mu^3+27\mu^2\xi+19\alpha\mu^2+10\lambda\mu^2+44\alpha\mu\xi+18\lambda\alpha\mu+28\mu\lambda\xi+5\mu\lambda^2+16\mu\alpha^2+36\mu\xi^2\\&+16\alpha^2\xi+\lambda^3+24\alpha\xi^2+16\lambda\xi^2+12\xi^3+22\alpha\lambda\xi+7\lambda^2\xi+8\alpha^2\lambda+5\lambda^2\alpha+4\alpha^3)\alpha^2\mu^2\xi\lambda\end{aligned}$$

$$\begin{aligned}p_{6/i}=D^{-1}(3(&6\mu^3+10\lambda\mu^2+27\mu^2\xi+17\alpha\mu^2+28\mu\lambda\xi+36\alpha\mu\xi+36\mu\xi^2+5\mu\lambda^2+15\alpha\lambda\mu+11\mu\alpha^2\\&+5\alpha^2\lambda+\lambda^3+8\alpha^2\xi+17\alpha\lambda\xi+7\lambda^2\xi+4\lambda^2\alpha+16\lambda\xi^2+18\alpha\xi^2+2\alpha^3+12\xi^3)\xi\mu\lambda^2\alpha^2\end{aligned}$$

$$\begin{aligned}p_{7/i}=D^{-1}(3(&12\mu^3+42\mu^2\xi+14\lambda\mu^2+42\alpha\mu^2+34\mu\alpha^2+33\mu\lambda\xi+42\mu\xi^2+84\alpha\mu\xi+33\lambda\alpha\mu+6\mu\lambda^2\\&+8\alpha^3+32\alpha\lambda\xi+7\lambda^2\xi+36\alpha\xi^2+14\alpha^2\lambda+32\alpha^2\xi+7\lambda^2\alpha+12\xi^3+16\lambda\xi^2+\lambda^3)\alpha\lambda\xi^2\mu^2\end{aligned}$$

$$\begin{aligned}p_{8/i}=D^{-1}(6\alpha^2(&13\alpha\mu\lambda\xi+24\mu^2\lambda\xi+6\xi^3\mu+6\lambda^2\mu\xi+11\lambda\xi^2\mu-2\lambda^2\xi\alpha-10\alpha\lambda\xi^2+21\mu^3\xi+5\lambda^2\mu^2\\&+21\mu^2\xi^2+7\mu\lambda^2\alpha-12\alpha\xi^3+\mu\lambda^3+6\mu^4+10\lambda\mu^3+30\xi\mu^2\alpha-12\alpha\mu\xi^2+22\alpha\mu^2\lambda+21\alpha\mu^3\\&-4\lambda\xi\alpha^2-12\alpha^2\xi^2+12\alpha^2\mu\xi+13\mu\alpha^2\lambda+21\mu^2\alpha^2+6\alpha^3\mu)\mu^2\xi)\end{aligned}$$

$$p_{9/i} = D^{-1}(3\alpha\mu^2\xi^2(-14\alpha\mu\lambda\xi + 33\mu^2\lambda\xi + 12\mu\xi^3 - 2\lambda^3\alpha + 7\lambda^2\xi\mu + 16\lambda\xi^2\mu - 14\lambda^2\xi\alpha - 32\alpha\lambda\xi^2$$
$$+42\mu^3\xi + 6\mu^2\lambda^2 + 42\xi^2\mu^2 - \mu\lambda^2\alpha - 24\alpha\xi^3 + \mu\lambda^3 + 12\mu^4 + 14\lambda\mu^3 + 60\mu^2\alpha\xi - 24\mu\alpha\xi^2$$
$$+21\mu^2\alpha\lambda + 42\mu^3\alpha - 24\xi\alpha^2\lambda - 6\alpha^2\lambda^2 - 24\xi^2\alpha^2 + 24\alpha^2\mu\xi + 8\lambda\mu\alpha^2 + 42\alpha^2\mu^2 - 4\lambda\alpha^3$$
$$+12\mu\alpha^3))$$

$$p_{10/i} = D^{-1}(\mu^2\xi^3(-14\xi\mu\alpha\lambda + 33\xi\lambda\mu^2 + 12\xi^3\mu - 2\lambda^3\alpha + 7\lambda^2\mu\xi + 16\lambda\xi^2\mu - 14\alpha\xi\lambda^2 - 32\xi^2\alpha\lambda$$
$$+42\mu^3\xi + 6\lambda^2\mu^2 + 42\xi^2\mu^2 - \lambda^2\alpha\mu - 24\alpha\xi^3 + \mu\lambda^3 + 12\mu^4 + 14\lambda\mu^3 + 60\xi\alpha\mu^2 - 24\xi^2\mu\alpha$$
$$+21\alpha\mu^2\lambda + 42\alpha\mu^3 - 24\lambda\alpha^2\xi - 6\lambda^2\alpha^2 - 24\xi^2\alpha^2 + 24\xi\alpha^2\mu + 8\lambda\alpha^2\mu + 42\alpha^2\mu^2 - 4\alpha^3\lambda$$
$$+12\alpha^3\mu)$$

where the constant D is given by

$$D = 252\mu^5\xi^2\alpha^2 + 14\lambda\mu^5\xi^3 + 168\mu^5\xi^3\alpha + 500\alpha^2\mu^3\xi^3\lambda + 78\lambda\alpha\mu^5\xi^2 + 96\lambda\alpha^2\mu^5\xi + 42\mu^5\xi^4 + 36\alpha\mu^6\xi^2$$
$$+60\alpha^5\mu^3\xi + 36\alpha^2\mu^6\xi + 12\mu^6\xi^3 + 4\lambda\alpha\mu^2\xi^5 + 19\lambda^3\alpha\mu^2\xi^3 + 108\lambda^2\alpha^2\mu^4\xi + 72\lambda\alpha^5\mu^3 + 51\lambda^2\alpha^5\mu^2$$
$$+2\lambda^3\alpha^6 + 22\lambda^3\alpha^5\mu + 66\alpha^2\mu^3\xi\lambda^3 + 3\lambda^4\mu^2\xi^2\alpha + 34\lambda^2\mu^2\xi^4\alpha + 160\lambda\mu^3\xi^4\alpha + 48\lambda^2\alpha^3\mu^4 + 42\alpha^5\mu^4$$
$$+225\xi\alpha^4\lambda^2\mu^2 + 60\lambda^2\mu^4\xi^2\alpha + 246\lambda\mu^4\xi^3\alpha + 21\lambda^3\mu^3\xi^2\alpha + 12\mu^6\alpha^3 + 42\alpha^4\mu^5 + 42\xi^5\mu^4 + 6\lambda^2\alpha^6\mu$$
$$+4\lambda^5\alpha^4 + 33\lambda\mu^4\xi^4 + 6\lambda^2\mu^4\xi^3 + 186\alpha\mu^4\xi^4 + \lambda^3\mu^3\xi^3 + 7\lambda^2\mu^3\xi^4 + 16\lambda\mu^3\xi^5 + 12\alpha^6\lambda\mu^2 + 12\mu^3\xi^6$$
$$+12\alpha\mu^3\xi^5 - 12\alpha^2\mu^3\xi^4 - 24\alpha\mu^2\xi^6 + 348\alpha^2\mu^4\xi^3 + 24\alpha^3\mu^3\xi^3 - 96\alpha^2\mu^2\xi^5 + 348\alpha^3\mu^4\xi^2 + 8\lambda^3\alpha^5\xi$$
$$+38\lambda^3\alpha^3\mu^3 + 99\lambda^2\alpha^4\mu^3 + 84\alpha^4\mu^3\xi^2 - 144\alpha^3\mu^2\xi^4 + 36\lambda\alpha^3\mu^5 + 96\lambda\alpha^4\mu^4 + 19\lambda^4\alpha^3\mu^2 + 12\alpha^6\mu^3$$
$$+6\lambda^5\alpha^3\mu + 7\lambda^5\alpha^3\xi + 16\lambda^4\alpha^3\xi^2 + 12\lambda^3\alpha^3\xi^3 + 21\lambda^4\alpha^4\mu + 18\lambda^3\alpha^4\xi^2 + 119\lambda^2\alpha\mu^3\xi^3 + 30\xi\alpha^5\lambda^2\mu$$
$$+17\lambda^4\alpha^4\xi + 495\lambda\alpha^2\mu^4\xi^2 + 594\alpha^3\mu^3\xi^2\lambda + 252\xi^3\lambda^2\alpha^2\mu^2 + 60\alpha^2\mu^2\lambda\xi^4 + 348\lambda^2\mu^3\xi\alpha^3 + 3\lambda^5\alpha^2\mu\xi$$
$$+141\lambda^3\mu^2\xi^2\alpha^2 + 366\lambda\alpha^3\mu^4\xi + 318\lambda\alpha^4\mu^3\xi + 390\lambda^2\alpha^3\mu^2\xi^2 + 140\lambda\alpha^3\mu^2\xi^3 + 176\alpha^3\mu^2\lambda^3\xi + \lambda^6\alpha^3$$
$$+21\lambda^4\alpha^2\mu^2\xi + 36\mu\lambda^2\alpha^2\xi^4 + 48\mu\lambda^3\alpha^2\xi^3 + 123\mu\lambda^3\alpha^3\xi^2 + 90\mu\lambda^2\alpha^3\xi^3 + 21\lambda^4\alpha^2\mu\xi^2 + 51\lambda^4\alpha^3\mu\xi$$
$$+86\lambda^3\alpha^4\xi\mu + 78\lambda^2\alpha^4\xi^2\mu - 24\alpha^5\mu^2\xi^2 + 5\lambda^4\alpha^5 + 186\alpha^4\mu^4\xi + 60\alpha^5\mu^2\lambda\xi - 96\alpha^4\mu^2\xi^3 + 58\alpha^4\mu^2\lambda^3$$
$$+168\alpha^3\mu^5\xi + 381\lambda^2\alpha^2\mu^3\xi^2 + 132\lambda\alpha^4\mu^2\xi^2.$$

Aggregating the states of each class S_i , The rates of transition between the 'macrostates'are expressed in the infinitesimal generator matrix

$$q_{i,j} = \begin{cases} \lambda q_a p_{10|i} + \xi q_f p_{2|i} + 2\xi q_f p_{3|i} + \xi q_f p_{5|i} + (\lambda q_a + 3\xi q_f)p_{4|i} & \\ +(\lambda q_a + 2\xi q_f)p_{6|i} + (\lambda q_a + \xi q_f)p_{7|i}, & i \geq 0,\ j = i+1 \\ i\theta(p_{1|i} + p_{2|i} + p_{3|i} + p_{5|i} + p_{8|i} + p_{9|i}), & i \geq 0,\ j = i-1 \\ -[\lambda q_a p_{10|i} + \xi q_f p_{2|i} + 2\xi q_f p_{3|i} + \xi q_f p_{5|i} + (\lambda q_a + 3\xi q_f)p_{4|i} + (\lambda q_a + 2\xi q_f)p_{6|i} + & \\ (\lambda q_a + \xi q_f)p_{7|i} + i\theta(p_{1|i} + p_{2|i} + p_{3|i} + p_{5|i} + p_{8|i} + p_{9|i})], & j = i \\ 0, & \text{otherwise.} \end{cases}$$

5 Numerical Experiments

In this section, we assess the quality of the phase-merging approximate for the unreliable $M/M/3$ retrial queue. Using a direct truncation method [12], we will compare results for congestion.

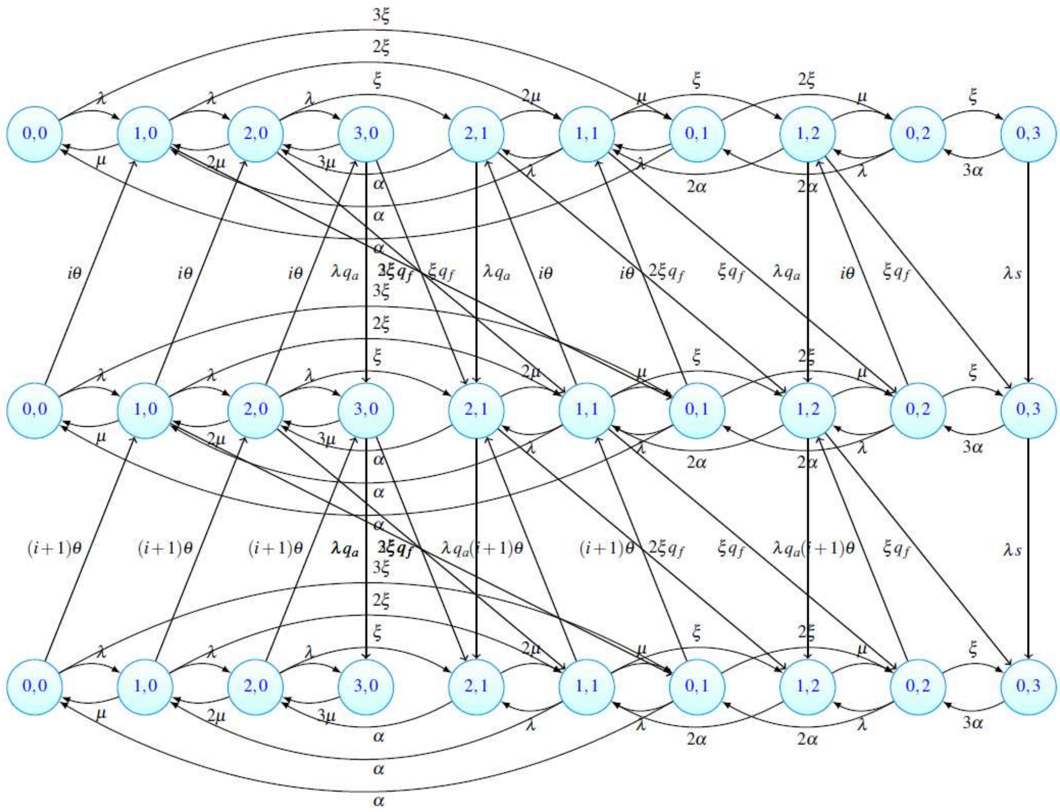

Fig. 3: Transition rate diagram for a reliable M/M/3 retrial queue .

ξ	Phase-Merging Algorithm	Direct Truncation Method
10.0000	0.0934	0.1326
5.0000	0.0448	0.0768
2.500	0.0236	0.0497
1.2500	0.0139	0.0364
0.6250	0.0083	0.0297
0.3125	0.0057	0.0264
0.1563	0.0045	0.0248
0.0781	0.0038	0.0240
0.0391	0.0035	0.0235
0.0195	0.0033	0.0233
0.0098	0.0033	0.0232

Table 3: Breakdown rate (ξ) and mean number of customers in the orbit for $c = 3,\ \lambda = 8,\ \mu = 10,\ \alpha = 100,\ \theta = 100$ and $q_a = q_f = 1$.

6 Conclusion

The primary aim of this work was to provide a formal analysis of the unreliable $M/M/c$ retrial queueing system with $c \geq 3$. Applying a phase merging algorithm due to Korolyuk [10] and Courtois [7], we showed that the steady state orbit length is approximately Poisson distributed. Using this result, we approximated the joint probability distribution of the number of customers in the orbit and the status of the servers. This enabled us to derive approximate expressions for the steady state mean orbit length, mean number of customers in service, mean number of customers in the system, the mean system sojourn time and the mean orbit sojourn time. The numerical examples show that that means number of customers

θ	phase-merging	Direct Truncation Method
10	0.935	0.9677
30	0.3117	0.3475
50	0.1870	0.2234
70	0.1336	0.1701
90	0.1042	0.1404
100	0.0935	0.1300
300	0.0311	0.0670
500	0.0187	0.0541
600	0.0155	0.0509
700	0.0133	0.0486
800	0.0116	0.0468
900	0.0103	0.0455
1000	0.0093	0.0444
2000	0.0046	0.0394
3000	0.0031	0.0378
4000	0.0023	0.0369
5000	0.0018	0.0364
6000	0.0015	0.0361

Table 4: Retrial rate (θ) and mean number of customers in the orbit for $c = 3,\ \lambda = 30,\ \mu = 40,\ \xi = 10,\ \alpha = 100$ and $q_a = q_f = 1$.

in the orbit using the phase-merging algorithm decrease more quickly than direct truncation method's. Better yet, these results remain valid if $q_a < 1$ and $q_f < 1$.

References

[1] Aissani, A and J.R. Artalejo, , On the single server retrial queue subject to breakdown, *Queueing Sys.*, **30** (1998), 309-321.

[2] Artalejo, J.R. and A. Gomez-Corral, *Retrial Queueing Systems: A Computational Approach*, Springer, Spain, pp: 318. (2008)

[3] Artalejo, J.R., A classified bibliography of research on retrial queues: Progress in 1990-1999.,*Busin. Econ.*,**7** (1999), 187-211.

[4] Artalejo, J.R., Accessible bibliography on retrial queues, *Math. Comp. Mod.*, **30** (1999), 1-6.

[5] Artalejo, J.R., Accessible bibliography on retrial queues: Progress in 2000-2009., *Math. Comp. Mod.*, **51** (2010), 1071-1081.

[6] Brian, P. Crawford, *Approximate analysis of an unreliable M/M/2 retrial queue*, thesis, (2012)

[7] Courtoi, P.J., Decomposability, instabilities, and saturation in multiprogramming systems, *Communications of the ACM*, **18 (7)** (1975), 371-377.

[8] Falin, G., , A survey of retrial queues, *Queueing Sys.*, **7**, (1990), 127-167.

[9] Falin, G.I. and J.G.C. Templeton, *Retrial queues*, Champman and Hall, London, pp: 328. (1997)

[10] Korolyuk, V.S. and V.V. Korolyuk, *Stochastic models of systems.* Kluwer Academic Publishers, Boston (1999).

[11] Kulkarni, V.G. and B.D. Choi, Retrial queues with server subject to breakdown and repairs, *Queueing Sys.*, **7** (1990), 191-208.

[12] Subramanian, M.G., G. Ayyappan and G. Sekar, M/M/c Retrial queueing system with breakdown and repair of services, *Asian Journal of Mthematics and Statistics* **4 (4)** (2011), 214-223.