



## Retrial System with Three Retrial Policies Subject to Repairable Starting Failures

Tzu-Hsin Liu<sup>1</sup>, Zhe-George Zhang<sup>2,3,4</sup> and Jau-Chuan Ke<sup>1,\*</sup>

<sup>1</sup>Department of Applied Statistics

National Taichung University of Science and Technology, Taichung 404, Taiwan

<sup>2</sup>Department of Decision Sciences

Western Washington University, Bellingham, WA 98225, USA

<sup>3</sup>Beedie School of Business

Simon Fraser University, Burnaby, BC, Canada, V5A 1S6

<sup>4</sup>School of Management

Lanzhou University, Lanzhou, Gansu, 730000, P. R. China

*(Received April 2019; accepted November 2019)*

---

**Abstract:** Motivated by the computer and telecommunication networks, this study deals with a multi-server retrial system with customer geometric loss (balking is possible for each service request attempt) and feedback. Moreover, the server may fail when it begins preparing to serve the customers in the startup period. An arriving customer who finds all servers are busy or broken either joins the retrial orbit or balks. If the customer gets the service, then after the service, he may either leave the system or enter the orbit for another service. The system is investigated as a quasi- birth-and-death process and interesting system performance measures are calculated with the long-term distribution. The optimum parameter settings which would minimize the cost function are discussed numerically. Some numerical experiments are presented to compare three retrial policies in terms of the major performance measures. It is demonstrated that our study can be utilized for not only evaluating performance but also generating some counter-intuitive results about system behavior.

**Keywords:** Cost, feedback, geometric loss, retrial queue, starting failure.

---

### 1. Introduction

The retrial queueing model has wide applications in practice such as telephone switching systems, telecommunication networks, computer and communication systems, call centers, and others. In computer networks, for instance, a packet can be retransmitted later by a retransmission mechanism if the packet is lost. For a literature review on retrial systems and their applications, see Falin [15], Kulkarni and Liang [23], Falin and Templeton [16], Artalejo and Gomez-Corral [3], Artalejo [1].

In the retrial systems, the inter-retrial times can be modeled according to different disciplines depending on each particular situation. In telephone networks, each source in

---

\*Corresponding author  
Email : jauchuan@nutc.edu.tw

orbit repeats its call after an exponentially distributed time with parameter  $\sigma$ . This type of retrial policy is called the classical retrial policy in which the total retrial rate is  $n\sigma$  when the orbit queue size is  $n \geq 0$  and was studied by Krishna Kumar *et al.* [21], Purohit and Rani [31], Tuan [37], and Choudhury and Deka [12]. In some practical applications such as communication networks, the retrial of customers may be controlled, and it seems that the retrial rate does not depend on the number of customers in the orbit. Hence, Fayolle [17] introduced another type of retrial policy called the constant retrial policy. Under such policy, the time between two successive attempts is independent of the number of customers in the orbit, i.e., the retrial rate is  $(1 - \delta_{n0})\nu$ , where  $\delta_{ij}$  denotes Kronecker delta. The constant retrial policy was used to describe the ALOHA protocol in communication systems, local area networks, communication protocols, mobile systems (Fayolle [17], Choi *et al.* [11], Shikata *et al.* [34]), and so on. Artalejo and Gomez-Corral [2] combined both policies by defining a linear retrial policy with rate  $(1 - \delta_{n0})\nu + n\sigma$ . Artalejo and Gomez-Corral [3] applied this policy in several computer and telecommunication networks. This paper focuses on the comparison of three retrial policies in a multi-server system involving feedback customers with possible server starting failure.

Retrial queueing systems with feedbacks can model many real-life situations such as multiple access telecommunication systems where messages turned out as errors are sent again and call center in which customers may call again if their problems are not completely solved after the service. The first work on the feedback retrial system is due to Takacs [36]. An M/G/1 retrial system with feedback was investigated by Choi and Kulkarni [10]. Choi *et al.* [9] studied an M/M/1 and M/M/2 retrial system including customer geometric loss and feedback. They derived the queue size distribution by the confluent hypergeometric equation and the method of series solution. Several extensions of the feedback retrial queue have been made by Atencia and Moreno [5], Lee [25], Krishna Kumar and Raja [20], Makkadis *et al.* [28], Ke and Chang [18], Krishna Kumar *et al.* [21], Do [13], Lin and Ke [27], Yang *et al.* [40] and Chang *et al.* [7].

In many practical applications such as telecommunication networks and computer manufacturing systems, retrial queues with server subject to breakdown and repair are often encountered. Studying the retrial queueing system with server breakdowns and repairs is not only important for theoretical research but also necessary for practical applications because server breakdowns and the limitation of repair capacity can severely deteriorate system performance. The existing literature on retrial systems with server breakdowns, can refer to Kulkarni and Choi [22], Yang and Li [42], Wang *et al.* [39], Wang and Zhou [38], Sumitha and Udaya Chandrika [35], Efrosinin and Sztrik [14], Chang and Wang [8], and Zirem *et al.* [43]. Li *et al.* [26] summarized that there are four important categories of important results concerning the queues with server subject to

breakdowns and repairs during the service. Here we consider the server is subject to starting failures that is different from the four classes indicated in Li *et al.* [26]. We assume that the arrival customer has to turn the server on when the server is idle on arrival. If the server is started successfully, the customer gets service immediately. Otherwise, the repair for the server commences immediately, and the customer must leave and make a retrial later. Assume that the server is reliable during the service. Such systems with the server subject to starting failures have been analyzed by Yang and Li [41], Krishna Kumar *et al.* [24], Mokaddis *et al.* [28], Atencia *et al.* [4], Ke and Chang [19], Sumitha and Udaya Chandrika [35], Rajadurai *et al.* [32], Ayyappan and Sathiya [6].

Motivated by the above situations, we study a multi-server retrial system with customer geometric loss and Bernoulli feedback, in which the server is subject to starting failure. In the proposed model, the server may meet an unpredictable breakdown subject to starting failure when a customer requires his service. The model analyzed in this paper can be a very suitable tool for modeling a computer system in which a message is transmitted from a source to a destination through several devices such as computers, routers, and switches. Analytical results could provide very useful and helpful management information for decision makers and practitioners with to design management policy.

The remainder of the paper is organized as follows. We first in Section 2 describe the system and give basic assumptions. Then, we construct a quasi-birth-and-death model for the system and propose an efficient algorithm for the long-term probability vector for queue size in Section 3. After that, we compute system performance measures based on the long-term distribution in Section 4. Numerical examples are implemented to illustrate the use of the algorithms. In Section 5, we construct a cost model for analyzing various retrial policies and searching for the optimum patterns. Finally, we conclude this paper in Section 6 with a brief summary.

## 2. System Description

In this study, we deal with a multi-server retrial system with Bernoulli feedback and loss under three retrial policies (mentioned earlier), where the servers may fail when it starts to prepare providing service for customers in the startup period. To analyze this retrial system, we make some assumptions given by:

1. The primary customers arrive in accordance with a Poisson process with rate  $\lambda$ . An arriving customer finding one or more servers available obtains service immediately; otherwise (finding all servers busy or broken), leaves the system with probability  $\bar{b}$  ( $= 1 - b$ ) or joins the orbit with probability  $b$ .

2. The system facility is comprised of  $m$  identical servers and service time of each server is exponentially distributed with parameter  $\mu$ . After the customer is served, he may either leave the system with probability  $\theta$  or join the retrial orbit for another service with probability  $\bar{\theta} (= 1 - \theta)$ .
3. Each server can serve only one customer at a time. Idle servers are shut down. If one or more servers are idle, an arriving customer will make one idle server to start up. The start-up time is assumed to be negligible. Moreover, the server may fail during the start-up with probability  $\bar{p} (= 1 - p)$ . If the server is started successfully, the customer gets service immediately. Otherwise, the repair starts immediately, and the customer must leave for the orbit and make a retrial later. The repair time of the failed server is assumed to be exponentially distributed with parameter  $\beta$ .
4. Customers in orbit make repeated attempts to get service. The length of the time interval between two consecutive attempts is exponentially distributed with linear intensity  $\sigma_j = (1 - \delta_{0j})\nu + j\sigma$ , where  $j$  is the number of customers in the orbit and  $\delta_{ij}$  is the Kronecker's delta. If  $\sigma = 0$  and  $\nu > 0$ , we obtain the constant retrial policy with parameter  $\nu$ . Alternatively, if  $\nu = 0$  and  $\sigma > 0$ , we get the case of classical retrial policy with parameter  $\sigma$ . The orbit customer who makes a service request and sees all  $m$  servers busy, may either leave the system with probability  $\bar{b} (= 1 - b)$  or back/enter the orbit again with probability  $b$  (this leads to geometric loss).
5. The arrival process, service process, retrial process and repair process are all assumed to be independent.

### 3. Mathematical Model

We define  $\{N_1(t), N_2(t), N_3(t); t \geq 0\}$  as the state of the system at time  $t$ , where  $N_1(t)$  denotes the number of busy servers,  $N_2(t)$  represents the number of customers in the orbit, and  $N_3(t)$  is the number of failed servers. Due to the exponentially distributed random variables in the model, the stochastic process is a three-dimensional continuous-time Markov Chain with state space  $S = \{(i, j, k); 0 \leq i \leq m, j \geq 0, 0 \leq k \leq m - i\}$ . Let

$$P_{j,k}^i = \lim_{t \rightarrow \infty} P\{N_1(t) = i, N_2(t) = j, N_3(t) = k\},$$

where  $0 \leq i + k \leq m$  and  $j \geq 0$  be the long-term probability of the retrial queueing system.

By the lexicographical sequence for the states, the infinitesimal generator  $\mathbf{Q}$  of the process describing the M/M/m retrial queue with Bernoulli feedback, loss and starting failures is of the form





and Rao, [30]) and assume that for a specified value (sufficiently large)  $M$  for all  $n \geq M$ , the retrial rate remain constant at  $(1 - \delta_{0n})\nu + M\sigma$ . We therefore assume that

$$\begin{aligned} \mathbf{A}_n &= \mathbf{A}_M, \text{ for } n \geq M \\ \mathbf{C}_n &= \mathbf{C}_M, \text{ for } n \geq M. \end{aligned}$$

It is observed that the matrix  $\mathbf{G} = \mathbf{B} + \mathbf{A}_M + \mathbf{C}_M$  is the infinitesimal generator. Let  $\mathbf{x} = [x_0^0, x_0^1, \dots, x_0^m, x_1^0, \dots, x_1^{m-1}, \dots, x_m^0]$  is  $(m+1)(m+2)/2$  elements row vector of the long-term probability  $\mathbf{G}$ . Solving the equation systems of  $\mathbf{xG} = \mathbf{0}$  and  $\mathbf{x}\mathbf{e} = 1$ , we obtain

$$x_k^i = \frac{p^i \bar{p}^k (\lambda + \sigma_M)^{i+k}}{i!k! \mu^i \beta^k} x_0^0, \quad 0 \leq i+k \leq m$$

where  $x_0^0 = \left[ \sum_{k=0}^m \sum_{i=0}^{m-k} \frac{p^i \bar{p}^k (\lambda + \sigma_M)^{i+k}}{i!k! \mu^i \beta^k} \right]^{-1}$ .

It is well known (Neuts [29], Theorem 3.1.1) that the standard drift condition  $\mathbf{xBe} < \mathbf{x}\mathbf{C}_M\mathbf{e}$  is a sufficient condition for the stable retrial system. After some algebraic manipulation, the stability condition for our model can be expressed as

$$\begin{aligned} \sum_{i+k \leq m} \frac{m!}{i!k!} (\lambda - p(\lambda + \sigma_M) + i\bar{\theta}\mu) \left( \frac{p(\lambda + \sigma_M)}{\mu} \right)^i \left( \frac{\bar{p}(\lambda + \sigma_M)}{\beta} \right)^k < \\ (1 - p - b)(\lambda + \sigma_M) \left( \frac{p(\lambda + \sigma_M)}{\mu} + \frac{\bar{p}(\lambda + \sigma_M)}{\beta} \right)^m. \end{aligned}$$

Because of the many parameters involved, it is difficult to have intuitive explanations behind the stability condition. Hence, we only attempt to give an explanation of the stability condition of a single server orbit queue with customer Bernoulli feedback, and geometric loss in which the server is subject to starting failure. The sufficient stability condition of a single server orbit queue with starting failures, feedback, and geometric loss is

$$p \left( \frac{b\lambda - \bar{b}\sigma_M}{\mu} + \bar{\theta} \right) + \bar{p} \left( 1 + \frac{b\lambda - \bar{b}\sigma_M}{\beta} \right) < \frac{\sigma_M}{\lambda + \sigma_M}.$$

According to the mean drift  $\varphi_j = p \left( \frac{b\lambda - \bar{b}\sigma_M}{\mu} + \bar{\theta} \right) + \bar{p} \left( 1 + \frac{b\lambda - \bar{b}\sigma_M}{\beta} \right) - \frac{\sigma_M}{\lambda + \sigma_M}$ , for  $j \geq 1$ , we could draw the following reasonable conclusions. The term  $p \left( \frac{b\lambda - \bar{b}\sigma_M}{\mu} + \bar{\theta} \right)$  contains components on successfully started service with probability

$p$  by any customer: the served customer joining the orbit with probability  $\bar{\theta}$  and arrivals during the busy period of the server ( $b\lambda - \bar{b}\sigma_M$ ). The term  $\bar{p}\left(1 + \frac{b\lambda - \bar{b}\sigma_M}{\beta}\right)$  also contains two components when the service fails during the start-up with probability  $\bar{p}$ : the arrivals during the breakdown ( $b\lambda - \bar{b}\sigma_M$ ) and the approached customer (resulting in breakdown of server) joining the orbit. In addition,  $\sigma_M / (\lambda + \sigma_M)$  gives the expected number of orbiting customers who enter service successfully, given that the previous service time leaves  $j$  customers in the orbit. We require, for stability, that customers have to arrive during service time and repair time more slowly than orbiting customers seeking service, at the commencement of service.

Under this stability condition, the long-term probability vector  $\mathbf{\Pi}$  of  $\mathbf{Q}$  exists. The long-term probability vector  $\mathbf{\Pi}$  partitioned as  $\mathbf{\Pi} = [\mathbf{\Pi}_0, \mathbf{\Pi}_1, \mathbf{\Pi}_2, \dots]$ , where the sub-vectors  $\mathbf{\Pi}_j = [P_{j,0}^0, P_{j,0}^1, \dots, P_{j,0}^m, P_{j,1}^0, P_{j,1}^1, \dots, P_{j,1}^{m-1}, \dots, P_{j,m-1}^0, P_{j,m-1}^1, P_{j,m}^0]$ , is given by

$$\mathbf{\Pi}_0 = \mathbf{\Pi}_1 \mathbf{C}_1 (-\mathbf{A}_0)^{-1} = \mathbf{\Pi}_1 \boldsymbol{\psi}_1,$$

$$\mathbf{\Pi}_{j-1} = \mathbf{\Pi}_j \mathbf{C}_j \left[ -(\boldsymbol{\psi}_{j-1} \mathbf{B} + \mathbf{A}_{j-1}) \right]^{-1} = \mathbf{\Pi}_j \boldsymbol{\psi}_j, \quad j = 2, 3, \dots, M$$

$$\mathbf{\Pi}_M \boldsymbol{\psi}_M \mathbf{B} + \mathbf{\Pi}_M \mathbf{A}_M + \mathbf{\Pi}_M \mathbf{R} \mathbf{C}_M = \mathbf{0},$$

$$\sum_{j=0}^{\infty} \mathbf{\Pi}_j \mathbf{e} = \mathbf{\Pi}_M \left[ \sum_{j=1}^M \prod_{i=M}^j \boldsymbol{\psi}_i + (\mathbf{I} - \mathbf{R})^{-1} \right] \mathbf{e} = \mathbf{1},$$

where  $\mathbf{e}$  is a column vector of suitable size with all elements equal to 1,  $\mathbf{I}$  is an identity matrix and  $\mathbf{R}$  is the unique non-negative solution with spectral radius less than one of the equation  $\mathbf{R}^2 \mathbf{C}_M + \mathbf{R} \mathbf{A}_M + \mathbf{B} = \mathbf{0}$ . In the following, we present a solution procedure to compute the long-term probability vectors.

*Algorithm for computing the steady-state probability vectors*

**INPUT**  $m, M, \mathbf{B}, \mathbf{A}_j (0 \leq j \leq M), \mathbf{C}_j (1 \leq j \leq M), \mathbf{R}, \mathbf{e}$  and  $\mathbf{I}$

**OUTPUT** approximate solution  $\mathbf{\Pi}_0, \mathbf{\Pi}_1, \mathbf{\Pi}_2, \dots$

Step 1: Set  $\boldsymbol{\psi}_1 = \mathbf{C}_1 (-\mathbf{A}_0)^{-1}$ ;

Step 2: For  $i = 2$  to  $M$ , set  $\boldsymbol{\psi}_i = \mathbf{C}_i \left[ -(\boldsymbol{\psi}_{i-1} \mathbf{B} + \mathbf{A}_{i-1}) \right]^{-1}$ ;

Step 3: For  $i = 1$  to  $M$ , set  $\boldsymbol{\Psi}_j = \prod_{i=N}^j \boldsymbol{\psi}_i$ ;

Step 4: Solve  $\mathbf{\Pi}_M \boldsymbol{\psi}_M \mathbf{B} + \mathbf{\Pi}_M \mathbf{A}_M + \mathbf{\Pi}_M \mathbf{R} \mathbf{C}_M = \mathbf{0}$  and  $\mathbf{\Pi}_M \left[ \sum_{j=1}^M \boldsymbol{\Psi}_j + (\mathbf{I} - \mathbf{R})^{-1} \right] \mathbf{e} = \mathbf{1}$ ;



Step 5: For  $i=1$  to  $M$ , set  $\mathbf{\Pi}_i = \mathbf{\Pi}_M \mathbf{\Psi}_{i+1}$ , and

for  $i = M + 1$  to  $\dots$ , set  $\mathbf{\Pi}_{i+1} = \mathbf{\Pi}_i \mathbf{R}$ ;

Step 6: OUTPUT.

#### 4. System Performance Measures

The major system performance measure of such a retrial queue system are listed as follows:

$E[B] \equiv$  expected number of busy servers;

$E[I] \equiv$  expected number of idle servers;

$E[D] \equiv$  expected number of broken servers;

$\sigma_1^* \equiv$  overall rate of retrials;

$\sigma_2^* \equiv$  successful rate of retrials;

$F \equiv$  fraction of retrials that are successful;

$L \equiv$  expected number of customers in orbit;

$L_c \equiv$  expected number of customers in orbit when all servers are busy or broken.

Using the long-term distribution, we obtain the following expressions:

$$\begin{aligned}
 E[B] &= \sum_{j=0}^{\infty} \mathbf{\Pi}_j \mathbf{f}_1 = \sum_{j=0}^M \mathbf{\Pi}_j \mathbf{f}_1 + \mathbf{\Pi}_M \mathbf{R} (\mathbf{I} - \mathbf{R})^{-1} \mathbf{f}_1; \\
 E[I] &= \sum_{j=0}^{\infty} \mathbf{\Pi}_j \mathbf{f}_2 = \sum_{j=0}^M \mathbf{\Pi}_j \mathbf{f}_2 + \mathbf{\Pi}_M \mathbf{R} (\mathbf{I} - \mathbf{R})^{-1} \mathbf{f}_2; \\
 E[D] &= \sum_{j=0}^M \mathbf{\Pi}_j \mathbf{f}_3 + \mathbf{\Pi}_M \mathbf{R} (\mathbf{I} - \mathbf{R})^{-1} \mathbf{f}_3; \\
 \sigma_1^* &= \sum_{j=1}^{\infty} \sigma_j \sum_{i+k < m} P_{j,k}^i + \sum_{j=1}^{\infty} \bar{b} \sigma_j \sum_{i+k=m} P_{j,k}^i \\
 &= \sum_{j=1}^{\infty} \sigma_j \mathbf{\Pi}_j (\mathbf{e} - \mathbf{f}_4) + \sum_{j=1}^{\infty} \bar{b} \sigma_j \mathbf{\Pi}_j \mathbf{f}_4 \\
 &= \sum_{j=1}^M \sigma_j \mathbf{\Pi}_j (\mathbf{e} - \mathbf{f}_4) + \sigma_M \mathbf{\Pi}_M \mathbf{R} (\mathbf{I} - \mathbf{R})^{-1} (\mathbf{e} - \mathbf{f}_4) \\
 &\quad + \sum_{j=0}^M \bar{b} \sigma_j \mathbf{\Pi}_j \mathbf{f}_4 + \bar{b} \sigma_M \mathbf{\Pi}_M \mathbf{R} (\mathbf{I} - \mathbf{R})^{-1} \mathbf{f}_4;
 \end{aligned}$$

$$\begin{aligned}\sigma_2^* &= \sum_{j=1}^{\infty} p\sigma_j \sum_{i+k<m} P_{j,k}^i = \sum_{j=1}^{\infty} p\sigma_j \Pi_j (\mathbf{e} - \mathbf{f}_4) \\ &= \sum_{j=1}^M p\sigma_j \Pi_j (\mathbf{e} - \mathbf{f}_4) + p\sigma_M \Pi_M \mathbf{R} (\mathbf{I} - \mathbf{R})^{-1} (\mathbf{e} - \mathbf{f}_4); \end{aligned}$$

$$F = \frac{\sigma_2^*}{\sigma_1^*};$$

$$L = \sum_{j=0}^{\infty} j \Pi_j \mathbf{e} = \sum_{j=0}^M j \Pi_j \mathbf{e} + M \Pi_M \mathbf{R} (\mathbf{I} - \mathbf{R})^{-1} \mathbf{e} + \Pi_M \mathbf{R} (\mathbf{I} - \mathbf{R})^{-2} \mathbf{e};$$

$$\begin{aligned}L_c &= \sum_{j=0}^{\infty} j \sum_{i+k=m} P_{j,k}^i = \sum_{j=0}^{\infty} j \Pi_j \mathbf{f}_4 \\ &= \sum_{j=0}^M j \Pi_j \mathbf{f}_4 + M \Pi_M \mathbf{R} (\mathbf{I} - \mathbf{R})^{-1} \mathbf{f}_4 + \Pi_M \mathbf{R} (\mathbf{I} - \mathbf{R})^{-2} \mathbf{f}_4, \end{aligned}$$

where

$$\begin{aligned}\mathbf{f}_1 &= \left[ \underbrace{0, 1, \dots, m}_{\# = m+1}, \underbrace{0, 1, \dots, m-1}_{\# = m}, \dots, \underbrace{0, 1}_{\# = 2}, 0 \right]^T, \\ \mathbf{f}_2 &= \left[ \underbrace{m, m-1, \dots, 0}_{\# = m+1}, \underbrace{m-1, m-2, \dots, 0}_{\# = m}, \dots, \underbrace{1, 0}_{\# = 2}, 0 \right]^T, \\ \mathbf{f}_3 &= m \left[ \underbrace{0, 0, \dots, 0}_{\# = m+1}, \underbrace{1, 1, \dots, 1}_{\# = m}, \dots, \underbrace{m-1, m-1}_{\# = 2}, m \right]^T \text{ and} \\ \mathbf{f}_4 &= \left[ \underbrace{0, 0, \dots, 0, 1}_{\# = m+1}, \underbrace{0, 0, \dots, 0, 1}_{\# = m}, \dots, \underbrace{0, 1}_{\# = 2}, 1 \right]^T. \end{aligned}$$

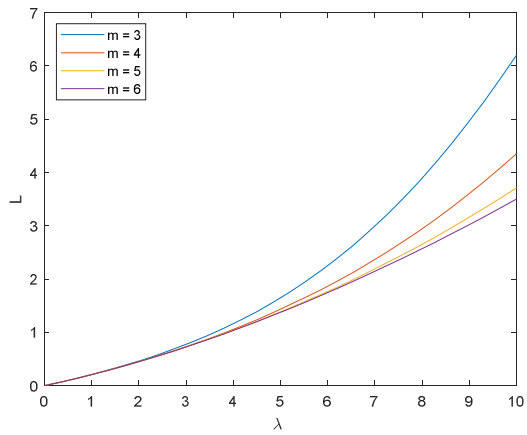
To explore the effects of the major system parameters on the expected number of customers in the orbit (a congestion measure), we carry out the numerical analysis based on the following cases with different values of  $m$  (the number of servers):

**Case 1:**  $\mu = 10$ ,  $\theta = 0.9$ ,  $p = 0.75$ ,  $b = 0.8$ ,  $\sigma = 1$ ,  $\nu = 1.5$ ,  $\beta = 6$ ,  $M = 20$ , change the values of  $\lambda$  from 0.01 to 10.

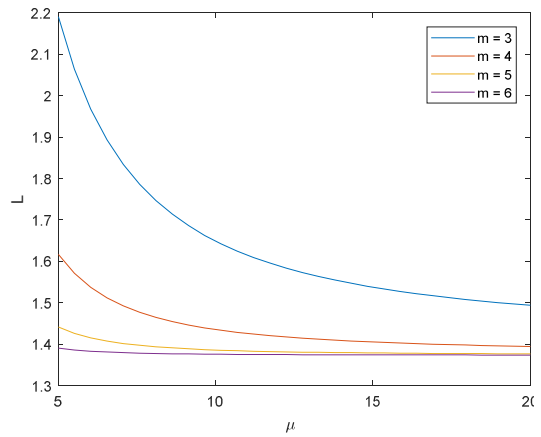
**Case 2:**  $\lambda = 5$ ,  $\theta = 0.9$ ,  $p = 0.75$ ,  $b = 0.8$ ,  $\sigma = 1$ ,  $\nu = 1.5$ ,  $\beta = 6$ ,  $M = 20$ , change the values of  $\mu$  from 5 to 20.

- Case 3:**  $\lambda=5$ ,  $\mu=10$ ,  $\theta=0.9$ ,  $p=0.75$ ,  $b=0.8$ ,  $\sigma=1$ ,  $\beta=6$ ,  $M=20$ , change the values of  $\nu$  from 0.1 to 5.
- Case 4:**  $\lambda=5$ ,  $\mu=10$ ,  $\theta=0.9$ ,  $p=0.75$ ,  $b=0.8$ ,  $\nu=1.5$ ,  $\beta=6$ ,  $M=20$ , change the values of  $\sigma$  from 1 to 20.
- Case 5:**  $\lambda=5$ ,  $\mu=10$ ,  $\theta=0.3$ ,  $p=0.75$ ,  $\sigma=1$ ,  $\nu=1.5$ ,  $\beta=6$ ,  $M=20$ , change the values of  $b$  from 0 to 0.8.
- Case 6:**  $\lambda=5$ ,  $\mu=10$ ,  $\theta=0.9$ ,  $p=0.75$ ,  $b=0.8$ ,  $\sigma=1$ ,  $\nu=1.5$ ,  $M=20$ , change the values of  $\beta$  from 0.5 to 20.
- Case 7:**  $\lambda=5$ ,  $\mu=10$ ,  $\theta=0.9$ ,  $b=0.5$ ,  $\sigma=1$ ,  $\nu=1.5$ ,  $\beta=6$ ,  $M=20$ , change the values of  $p$  from 0.1 to 1.
- Case 8:**  $\lambda=5$ ,  $\mu=10$ ,  $p=0.75$ ,  $b=0.5$ ,  $\sigma=1$ ,  $\nu=1.5$ ,  $\beta=6$ ,  $M=20$ , change the values of  $\theta$  from 0.1 to 1.

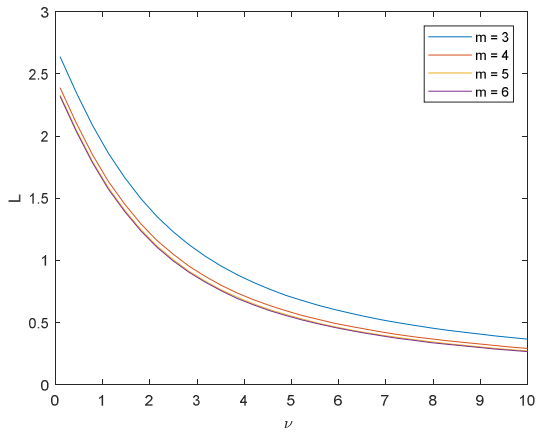
The results are illustrated in Figure 1 for Cases 1-8, respectively. Each case focuses on each system parameter. Although the general patterns of these relations are intuitive, our algorithms based on the QBD model offer a powerful tool for evaluating the system performance quantitatively. It reveals from Figure 1 that  $L$  increases as  $\lambda$  or  $b$  increases, and decreases as the other parameters ( $\mu$ ,  $\nu$ ,  $\sigma$ ,  $\beta$ ,  $p$ ,  $\theta$ ) increases. In particular, we find that the effect of each parameter on the  $L$  (congestion measure) depends on the number of servers. For example, for some parameters such as  $\nu$  or  $\sigma$  (retrial rate), the impact on  $L$  remains almost the same for  $c \geq 3$  as illustrated in Cases 3 and 4, respectively, in Figure 1. However, the impact on  $L$  of other system parameters such as  $b$  (balking rate),  $p$  (server starting failure rate), and  $\theta$  (leaving probability after service) may depend on the number of servers significantly as shown in Cases 5, 7, and 8, respectively, in Figure 1. These observations offer important insights for practitioners in designing the system. Another interesting finding is that there exists an intersection in Case 5 of Figure 1. This implies that when the balking rate is too high, the fewer servers may lead to smaller number of customers in the orbit, a little counter-intuitive result. This mainly because that balking does not take time but reduce the number of customers in the orbit. Of course, such a high balking rate may not be reasonable in a practical system.



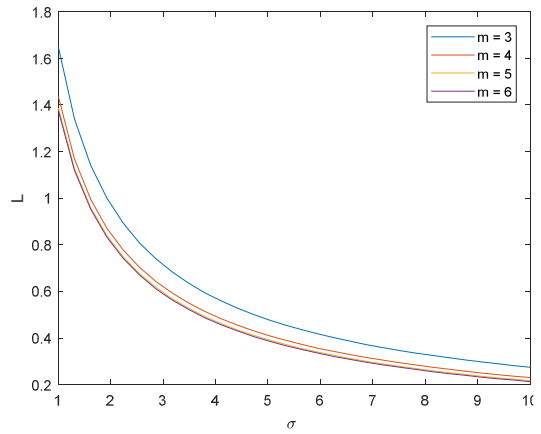
Case 1.  $0.01 \leq \lambda \leq 10$



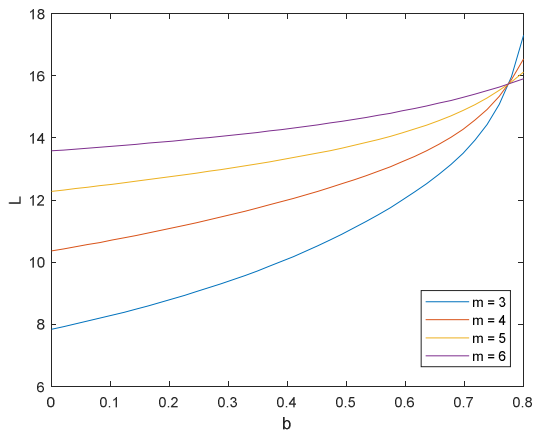
Case 2.  $5 \leq \mu \leq 20$



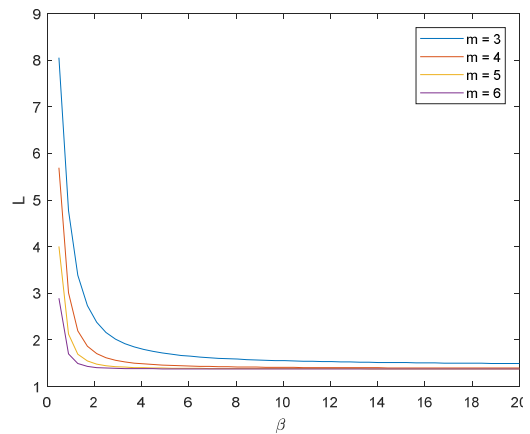
Case 3.  $0.1 \leq \nu \leq 10$



Case 4.  $1 \leq \sigma \leq 10$



Case 5.  $0 \leq b \leq 0.8$



Case 6.  $0.5 \leq \beta \leq 20$

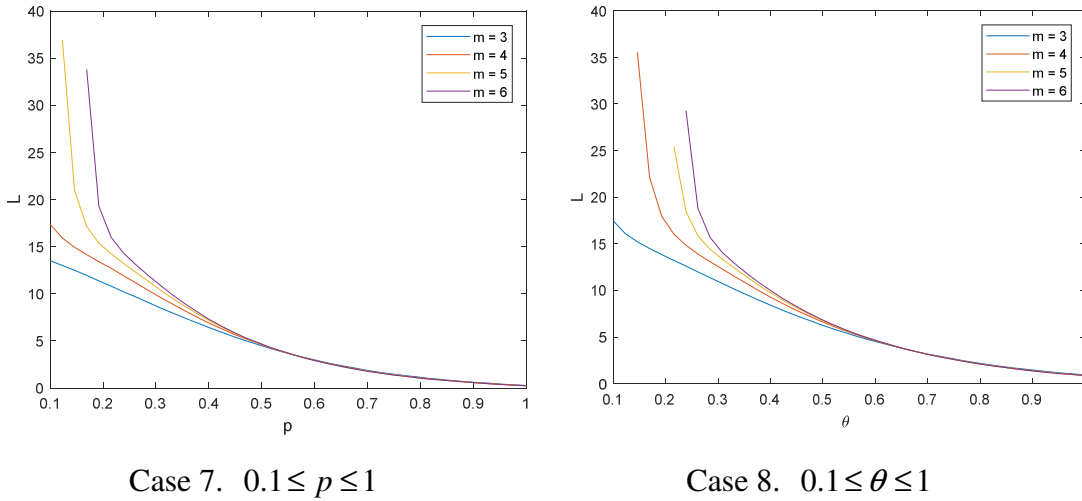


Figure 1. The expected number of customers in the orbit versus various system parameters.

In Figure 2, we compare classical retrial policy, constant retrial policy, and linear retrial policy by considering the following cases.

- Case 9:**  $\mu = 10, \theta = 0.9, p = 0.7, b = 0.5, \beta = 6, m = 3, M = 20$ , change the values of  $\lambda$  from 0.01 to 5.
- Case 10:**  $\lambda = 5, \theta = 0.9, p = 0.7, b = 0.5, \beta = 6, m = 3, M = 20$ , change the values of  $\mu$  from 5 to 20.
- Case 11:**  $\lambda = 5, \mu = 10, \theta = 0.9, p = 0.7, \beta = 6, m = 3, M = 20$ , change the values of  $b$  from 0 to 1.
- Case 12:**  $\lambda = 5, \mu = 10, \theta = 0.9, p = 0.7, b = 0.5, m = 3, M = 20$ , change the values of  $\beta$  from 1 to 10.
- Case 13:**  $\lambda = 3, \mu = 10, p = 0.9, b = 0.5, \beta = 6, m = 3, M = 20$ , change the values of  $\theta$  from 0.1 to 1.
- Case 14:**  $\lambda = 3, \mu = 10, \theta = 0.95, b = 0.5, \beta = 6, m = 3, M = 20$ , change the values of  $p$  from 0.1 to 1.

It is observed that  $L_{\text{linear retrial policy}} < L_{\text{constant retrial policy}} < L_{\text{classical retrial policy}}$  holds when  $\lambda, \mu, b$  or  $\beta$  varies as shown in Cases 9 to 12 of Figure 2. This is intuitive as the linear retrial policy is more flexible (more general) than the other two policies. However, again the  $L$  curves against  $\theta$  and  $p$  shown in Cases 13 and 14 intersect, indicating that the dominating relation  $L_{\text{linear retrial policy}} < L_{\text{constant retrial policy}} < L_{\text{classical retrial policy}}$  fails. This is also a useful insight for practitioners in selecting a retrial policy for a given feedback rate or server starting failure rate.

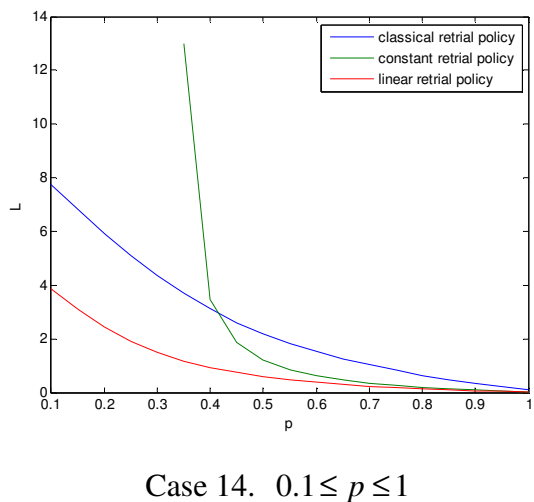
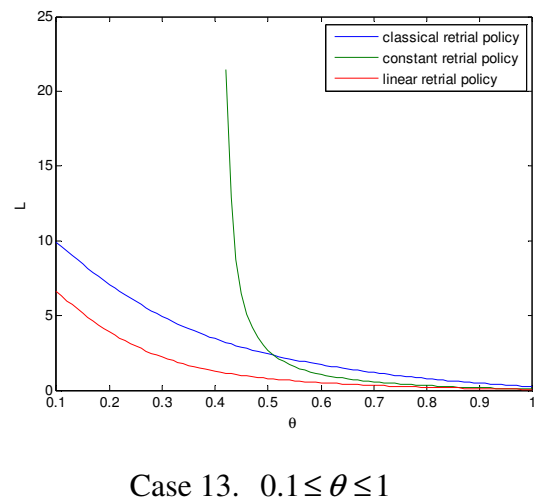
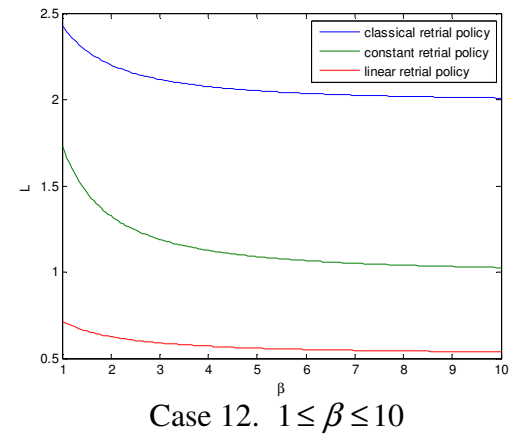
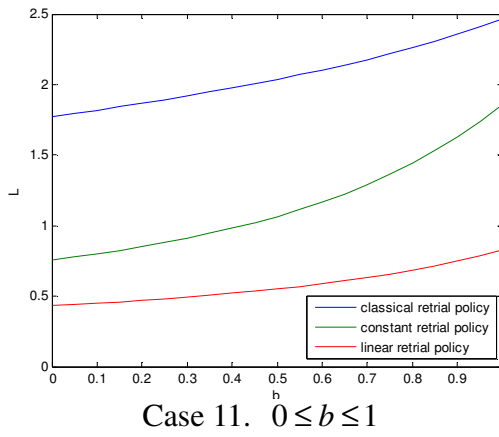
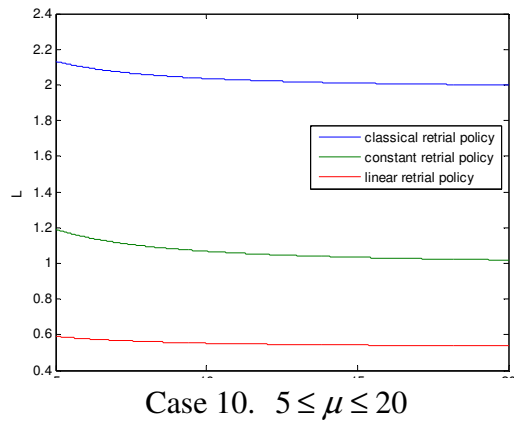
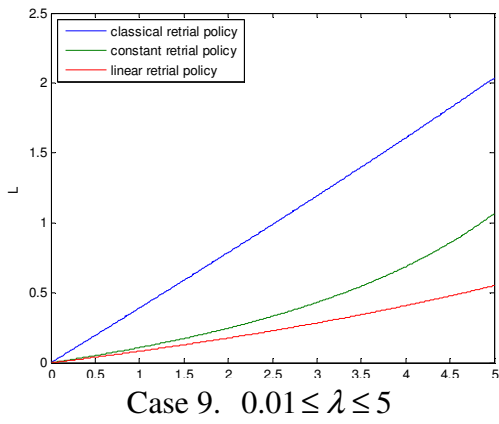


Figure 2. The effect of the retrial policy and different system parameters on  $L$ .

## 5. Optimization Analysis for Three Retrieval Policies

The long-term distribution obtained can be used to address the optimization issue. For this purpose, we construct an expected cost function per unit time in which the number of servers, the mean service rate, and the mean repair rate are decision variables. Such a cost function is based on the cost parameters defined as follows:

- $c_h \equiv$  holding cost per unit time per customer present in orbit;
- $c_d \equiv$  cost per unit time per failed server;
- $c_l \equiv$  cost per unit time per customer loss from orbit;
- $c_f \equiv$  cost per unit time of each available server;
- $c_s \equiv$  cost per customer served by a mean service rate  $\mu$ ;
- $c_r \equiv$  cost per failed server repaired by a mean repair rate  $\beta$ .

Using these cost elements, the expected cost function  $F(m, \mu, \beta)$  per customer per unit time is given by

$$F(m, \mu, \beta) = c_h L + c_d E[D] + c_l \bar{b} E[L_c] + c_f m + c_s \mu + c_r \beta$$

Because the expected cost function is highly complex and nonlinear on  $(m, \mu, \beta)$ , we apply the probabilistic global search Lausanne (PGSL) method developed by Raphael and Smith [33] to obtain the optimum value of  $(m, \mu, \beta)$ . To compare three retrieval policies, we use the following cost parameters:

$$\begin{array}{ll} c_h = 300/\text{customer/unit time}; & c_d = 20/\text{server/unit time}; \\ c_l = 15/\text{customers/unit time}; & c_f = 30/\text{server/unit time}; \\ c_s = 8/\text{customer}; & c_r = 5/\text{failed server}. \end{array}$$

Given other system parameters, we observe from Tables 1-4 that (i)  $m^*$  increases in  $\lambda$  or  $b$  but decreases in  $\theta$  or  $p$ ; and (ii) for a given  $m^*$ ,  $(\mu^*, \beta^*)$  increase in  $\lambda$  or  $b$  and  $(\mu^*, \beta^*)$  decrease in  $\theta$  or  $p$ . In addition, although these relations are intuitive and as expected, our analysis reveals the quantitative measures of operating costs under three different policies. These quantified measures help decision makers in deciding whether or not a change in retrieval policy is beneficial (performing cost and benefit analysis).

In addition, it is also observed that the minimum expected cost follows the cost order linear retrieval policy < constant retrieval policy < classical retrieval policy due to the flexibility of the linear retrieval policy.

Table 1. The optimum value  $(m^*, \mu^*, \beta^*)$  and the minimum expected cost for various value of  $\lambda$  and different retrial policy ( $\theta = 0.6$ ,  $p = 0.7$ ,  $b = 0.8$ ,  $M = 20$ ).

$\lambda$	linear retrial policy		constant retrial policy		classical retrial policy	
	$(m^*, \mu^*, \beta^*)$	$F(m^*, \mu^*, \beta^*)$	$(m^*, \mu^*, \beta^*)$	$F(m^*, \mu^*, \beta^*)$	$(m^*, \mu^*, \beta^*)$	$F(m^*, \mu^*, \beta^*)$
1	(2,2.63,3.38)	192.558	(2,3.56,4.32)	235.031	(3,2.83,3.46)	426.503
1.5	(3,3.11,4.02)	261.547	(3,3.56,4.32)	235.031	(3,4.20,4.73)	589.725
2	(3,4.28,5.16)	325.436	(4,4.41,5.09)	461.556	(4,4.01,4.68)	751.301
2.5	(3,5.51,6.30)	394.848	(4,6.14,6.51)	639.040	(4,5.00,5.57)	906.240
3	(4,5.09,6.00)	464.145	(5,6.21,6.44)	927.613	(4,5.99,6.45)	1061.031
3.5	(4,6.05,6.85)	538.833	(6,6.49,6.55)	1531.722	(4,6.98,7.31)	1215.722
4	(4,7.04,7.71)	619.936	(7,7.70,7.03)	3762.429	(5,6.06,6.66)	1366.365

Table 2. The optimum value  $(m^*, \mu^*, \beta^*)$  and the minimum expected cost for various value of  $\theta$  and different retrial policy ( $\lambda = 3$ ,  $p = 0.95$ ,  $b = 0.8$ ,  $M = 20$ ).

$\theta$	linear retrial policy		constant retrial policy		classical retrial policy	
	$(m^*, \mu^*, \beta^*)$	$F(m^*, \mu^*, \beta^*)$	$(m^*, \mu^*, \beta^*)$	$F(m^*, \mu^*, \beta^*)$	$(m^*, \mu^*, \beta^*)$	$F(m^*, \mu^*, \beta^*)$
0.5	(3,7.02,2.70)	371.417	(4,7.41,2.71)	588.481	(4,6.17,2.37)	866.437
0.6	(3,5.98,2.36)	287.691	(4,5.52,2.18)	376.361	(4,5.35,2.10)	645.007
0.7	(3,5.21,2.09)	234.358	(3,6.27,2.46)	279.448	(4,4.72,1.89)	486.230
0.8	(3,4.62,1.87)	197.580	(3,5.29,2.14)	221.070	(3,6.05,2.28)	364.177
0.9	(3,4.15,1.70)	170.739	(3,4.60,1.89)	182.875	(3,5.49,2.09)	268.651



Table 3. The optimum value  $(m^*, \mu^*, \beta^*)$  and the minimum expected cost for various value of  $p$  and different retrial policy ( $\lambda = 3, \theta = 0.9, b = 0.8, M = 20$ ).

$p$	linear retrial policy		constant retrial policy		classical retrial policy	
	$(m^*, \mu^*, \beta^*)$	$F(m^*, \mu^*, \beta^*)$	$(m^*, \mu^*, \beta^*)$	$F(m^*, \mu^*, \beta^*)$	$(m^*, \mu^*, \beta^*)$	$F(m^*, \mu^*, \beta^*)$
0.5	(4,3.94,7.38)	425.360	(5,4.34,7.48)	719.633	(4,4.77,7.95)	963.945
0.6	(3,4.98,7.33)	334.530	(4,4.73,6.89)	443.534	(4,4.59,6.39)	727.863
0.7	(3,4.73,5.70)	272.039	(4,4.16,5.13)	327.376	(4,4.39,5.02)	556.610
0.8	(3,4.51,4.17)	226.144	(3,5.24,4.77)	257.319	(4,4.19,3.72)	425.640
0.9	(3,4.29,2.60)	188.683	(3,4.82,2.93)	205.642	(3,5.67,3.14)	317.817

Table 4. The optimum value  $(m^*, \mu^*, \beta^*)$  and the minimum expected cost for various value of  $b$  and different retrial policy ( $\lambda = 3, \theta = 0.9, p = 0.95, M = 20$ ).

$b$	linear retrial policy		constant retrial policy		classical retrial policy	
	$(m^*, \mu^*, \beta^*)$	$F(m^*, \mu^*, \beta^*)$	$(m^*, \mu^*, \beta^*)$	$F(m^*, \mu^*, \beta^*)$	$(m^*, \mu^*, \beta^*)$	$F(m^*, \mu^*, \beta^*)$
0.5	(2,3.46,1.34)	143.823	(2,4.32,1.68)	161.207	(3,4.28,1.65)	254.489
0.6	(2,4.41,1.69)	155.109	(3,3.70,1.53)	172.837	(3,4.74,1.81)	259.857
0.7	(2,5.32,2.06)	165.812	(3,4.17,1.71)	178.069	(3,5.13,1.96)	264.507
0.8	(3,4.15,1.70)	170.739	(3,4.60,1.89)	182.875	(3,5.49,2.09)	268.651
0.9	(3,4.59,1.89)	175.433	(3,5.00,2.07)	187.343	(3,5.81,2.22)	272.418

## 6. Conclusions

This paper investigates a multi-server retrial queue with geometric loss and feedback, and unreliable servers during start-up period. The sufficient condition for the stability of this system is obtained for the QBD process. For a stable system, we develop the major long-term performance measures to evaluate such a stochastic service system. The system

can be made more economical at the optimum number of servers, the optimum service rate, and the optimum repair rate simultaneously to minimize the expected cost. Finally, we compare the linear retrial policy, the constant retrial policy, and the classical retrial policy by numerical analysis. Besides the quantified performance measures, we also observe some counter-intuitive system behaviors. The results will benefit the practitioners in telecommunication systems and computer networks or other service systems that fit the model.

## Reference

- [1] Artalejo, J. R. (2010). Accessible bibliography on retrial queues: Progress in 2000-2009. *Mathematical and Computer Modelling*, 51, 1071-1081.
- [2] Artalejo, J. R., & Gomez-Corral, A. (1997). Steady-state solution of a single server queue with linear request repeated. *Journal of Applied Probability*, 34, 223-233.
- [3] Artalejo, J. R., & Gomez-Corral, A. (2008). *Retrial Queueing Systems: A Computational Approach*. Berlin, Springer.
- [4] Atencia, I., Fortes, I., & Sánchez, S. (2009). A discrete-time retrial queueing system with starting failures, Bernoulli feedback and general retrial times. *Computers and Industrial Engineering*, 57, 1291-1299.
- [5] Atencia, I., & Moreno, P. (2003). A queueing system with linear repeated attempts, Bernoulli schedule and feedback. *TOP*, 11, 285-310.
- [6] Ayyappan, G., & Sathiya, K. (2013). Transient analysis of batch arrival feedback retrial queue with starting failure and Bernoulli vacation. *Mathematical Theory and Modeling*, 3, 60-67.
- [7] Chang, F. M., Liu, T. H., & Ke, J. C. (2018). On an unreliable-server retrial queue with customer feedback and impatience. *Applied Mathematical Modelling*, 55, 171-182.
- [8] Chang, J., & Wang, J. (2018). Unreliable M/M/1/1 retrial queues with set-up time. *Quality Technology and Quantitative Management*, 15, 589-601.
- [9] Choi, B. D., Kim, Y. C., & Lee, Y. W. (1998). The M/M/c retrial queue with geometric loss and feedback. *Computers and Mathematics with Applications*, 36, 41-52.
- [10] Choi, B. D., & Kulkarni, V. G. (1992). *Feedback Retrial Queueing Systems, Stochastic Models and Related Fields*, 93-105, Oxford University Press, New York.
- [11] Choi, B. D., Shin, Y. W., & Ahn, W. C. (1992). Retrial queues with collision arriving from unslotted CSMA/CD protocols. *Queueing Systems*, 11, 335-356.

- [12] Choudhury, G., & Deka, K. (2013). A batch arrival retrial queue with two phases of service and Bernoulli vacation schedule. *Acta Mathematicae Applicatae Sinica, English Series*, 29, 15-34.
- [13] Do, T. V. (2010). An efficient computation algorithm for a multiserver feedback retrial queue with a large queueing capacity. *Applied Mathematical Modelling*, 34, 2272-2278.
- [14] Efrosinin, D., & Sztrik, J. (2018). An algorithmic approach to analyzing the reliability of a controllable unreliable queue with two heterogeneous servers, *European Journal of Operational Research*, 271, 934-952.
- [15] Falin, G. I. (1990). A survey of retrial queues. *Queueing Systems*, 1, 127-168.
- [16] Falin, G. I., & Templeton, J.G.C. (1997). *Retrial Queues*. New York, Chapman and Hall.
- [17] Fayolle, G. (1986). A simple telephone exchange with delayed feedbacks. In O. J Boxma, J. W. Cohen and H. C. Tijms (Eds.), *Teletrac Analysis and Computer Performance Evaluation*, 245-253.
- [18] Ke, J. C., & Chang, F. M. (2009). Modified vacation policy for M/G/1 retrial queue with balking and feedback. *Computers and Industrial Engineering*, 57, 433-443.
- [19] Ke, J. C., & Chang, F. M. (2009).  $M^X/G_1, G_2/1$  retrial queue with Bernoulli vacation schedules with repeated attempts and starting failures. *Applied Mathematical Modeling*, 33, 3186-3196.
- [20] Krishna Kumar, B., & Raja, J. (2006). On multiserver feedback retrial queues with balking and control retrial rate. *Annals of Operations Research*, 141, 211-232.
- [21] Krishna Kumar, B., Vijayalakshmi, G., Krishnamoorthy, A., & Sadiq Basha, S. (2010). A single server feedback retrial queue with collisions. *Computers and Operations Research*, 37, 1247-1255.
- [22] Kulkarni, V. G., & Choi, B. D. (1990). Retrial queues with server subject to breakdowns and repairs. *Queueing Systems*, 7, 191-208.
- [23] Kulkarni, V. G., & Liang, H. M. (1997). Retrial queues revisited. In: Dshalalow JH, editor. *Frontiers in queueing models and applications in science and engineering*. Boca Raton: CRC Press, 19-34.
- [24] Krishna Kumar, B., Pavai Madheswari, S., & Vijayakumar, A. (2002). The M/G/1 retrial queue with feedback and starting failures. *Applied Mathematical Modelling*, 26, 1057-1075.
- [25] Lee, Y. W. (2005). The M/G/1 feedback retrial queue with two types of customers. *Bulletin of the Korean Mathematical Society*, 42, 875-887.

- [26] Li, Q. L., Ying, Y., & Zhao, Y. Q. (2006). A BMAP/G/1 retrial queue with a server subject to breakdowns and repairs. *Annals of Operations Research*, 141, 233-270.
- [27] Lin, C. H., & Ke, J. C. (2011). On the multi-server retrial queue with geometric loss and feedback: computational algorithm and parameter optimization. *International Journal of Computer Mathematics*, 88, 1083-1101.
- [28] Mokaddis, G. S., Metwally, S. A., & Zaki, B. M. (2007). A feedback retrial queuing system with starting failures and single vacation. *Tamkang Journal of Science and Engineering*, 10, 183-192.
- [29] Neuts, M. F. (1981). *Matrix Geometric Solutions in Stochastic Models: An Algorithmic Approach*. Baltimore, The John Hopkins University Press.
- [30] Neuts, M. F., & Rao, B. M. (1990). Numerical investigation of a multi-server retrial model. *Queueing Systems*, 7, 169-190.
- [31] Purohit, G. N., & Rani, Shinu (2012). M/M/1 retrial queue with Bernoulli feedback, linear retrial rate, collisions and state-dependent arrival rates. *The IUP Journal of Operations Management*, XI(2), 54-64.
- [32] Rajadurai, P., Saravananarajan, M.C., & Chandrasekaran, V. M. (2018). A study on M/G/1 feedback retrial queue with subject to server breakdown and repair under multiple working vacation policy. *Alexandria Engineering Journal*, 57, 947-962.
- [33] Raphael, B., & Smith, I.F.C. (2003). A direct stochastic algorithm for global search. *Applied Mathematics and Computation*, 146, 729-758.
- [34] Shikata, Y., Suzuki, S., Takahashi, Y., Ihara, T., & Nakanishi, T. (1999). Loss probability evaluation of PCS call-terminating control. *IEICE Transactions Fundamentals*, E82-A, 7, 1230-1237.
- [35] Sumitha, D., & Udaya Chandrika, K. (2012). Retrial queuing system with starting failure, single vacation and orbital search. *International Journal of Computer Applications*, 40, 29-33.
- [36] Takacs, L. (1963). A single server queue with feedback. *Bell System Technical Journal*, 42, 505-519.
- [37] Tuan, P. D. (2012). An explicit solution for a tandem queue with retrials and losses. *Operational Research*, 12, 189-207.
- [38] Wang, J., & Zhou, P. F. (2010). A batch arrival retrial queue with starting failures, feedback and admission control. *Journal of Systems Science and Systems Engineering*, 19, 306-320.
- [39] Wang, J., Cao, J., & Li, Q. (2001). Reliability analysis of the retrial queue with server breakdowns and repairs. *Queueing Systems*, 38, 363-380.

- [40] Yang, D. Y., Ke, J. C., & Wu, C. H. (2015). The multi-server retrial system with Bernoulli feedback and starting failures. *International Journal of Computer mathematics*, 92, 954-969.
- [41] Yang, T., & Li, H. (1994). The M/G/1 retrial queue with the server subjected to starting failure. *Queueing System*, 16, 83-96.
- [42] Yang, T., & Li, H. (1995). On the steady-state queue size distribution of the discrete-time Geo/G/1 queue with repeated customers. *Queueing Systems*, 21, 199-215.
- [43] Zirem, D., Boualem, M., Adel-Aissanou, K., & Aïssani, D. (2019). Analysis of a single server batch arrival unreliable queue with balking and general retrial time. *Quality Technology and Quantitative Management*, 16, 672-695.

