The Unreliable $M/M/1$ Retrial Queue in a Random Environment

James D. Cordeiro
Department of Mathematics and Statistics
Air Force Institute of Technology
2950 Hobson Way (AFIT/ENC)
Wright Patterson AFB, OH 45433
Ph: (937) 255-3636 Ext. 4398
Email: James.Cordeiro@afit.edu

and

Jeffrey P. Kharoufeh[1]
Department of Industrial Engineering
University of Pittsburgh
1048 Benedum Hall
3700 O'Hara Street
Pittsburgh, PA 15261 USA
Ph: (412) 624-9832; Fax: (412) 624-9831
Email: jkharouf@pitt.edu

Final version appears in
Stochastic Models, 28 (1), pp. 29-48, 2012.

Abstract

We examine an $M/M/1$ retrial queue with an unreliable server whose arrival, service, failure, repair, and retrial rates are all modulated by an exogenous random environment. Provided are conditions for stability, the (approximate) orbit size distribution, and mean queueing performance measures which are obtained via matrix-analytic methods. Additionally, we consider the problem of choosing arrival and service rates for each environment state with the objective of minimizing the steady state mean time spent in orbit by an arbitrary customer, subject to cost and revenue constraints. Two numerical examples illustrate the main results.

Keywords: Retrial queue, LDQBD process, unreliable server.

[1]Author to whom correspondence should be addressed.

Report Documentation Page

Form Approved
OMB No. 0704-0188

1. REPORT DATE	2. REPORT TYPE	3. DATES COVERED
2012		**00-00-2012 to 00-00-2012**

4. TITLE AND SUBTITLE	5a. CONTRACT NUMBER
The Unreliable M/M/1 Retrial Queue in a Random Environment	5b. GRANT NUMBER
	5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)	5d. PROJECT NUMBER
	5e. TASK NUMBER
	5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **Air Force Institute of Technology,Department of Mathematics and Statistics,2950 Hobson Way (AFIT/ENC),Wright Patterson AFB,OH,45433**	8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR'S ACRONYM(S)
	11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited

13. SUPPLEMENTARY NOTES

14. ABSTRACT

We examine an M=M=1 retrial queue with an unreliable server whose arrival, service, failure repair, and retrial rates are all modulated by an exogenous random environment. Provided are conditions for stability, the (approximate) orbit size distribution, and mean queueing perfor- mance measures which are obtained via matrix-analytic methods. Additionally, we consider the problem of choosing arrival and service rates for each environment state with the objective of minimizing the steady state mean time spent in orbit by an arbitrary customer, subject to cost and revenue constraints. Two numerical examples illustrate the main results.

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT **unclassified**	b. ABSTRACT **unclassified**	c. THIS PAGE **unclassified**	**Same as Report (SAR)**	**16**	

1 Introduction

This paper examines an $M/M/1$ retrial queue with an unreliable server whose arrival, service, failure, repair, and retrial rates are all modulated by an external environment. For this system, arriving customers who find the server busy or failed, or customers whose service is interrupted by a server failure, join a retrial queue (or orbit) from which they persistently attempt to gain (or regain) access to the server at random intervals. Retrial customers can gain access to the server only when it is found operational and idle, and they repeat service until their service requirement has been satisfied. The server experiences both active and idle failures; the former correspond to failures that occur when the server is processing a customer, and the latter occur when the server is idle. Once a failure occurs, the server immediately commences a repair cycle whose duration is stochastic. The server cannot fail when it is under repair. While the arrival, service, failure, repair, and retrial times are assumed to be mutually independent, they are all modulated by a common random environment. Like many queueing models of this type, we assume the environment is an ergodic continuous-time Markov chain (CTMC) on a finite state space. We analyze this system using classical matrix-analytic methods (MAM) developed originally by Neuts [35], focusing on the stability analysis and (numerical) computation of the steady state distribution of the orbit size from which we obtain steady state performance measures. Additionally, we consider the problem of choosing the modulated arrival and service rates that minimize the mean time spent in orbit by an arbitrary customer who arrives in steady state.

Of particular relevance to the present paper are single server retrial models with unreliable servers (i.e., those with servers that experience failures at random intervals). Three seminal papers related to this topic include [3, 6, 26]. For retrial systems with an unreliable server and no queue for primary arrivals, customers who arrive to find a busy or failed server are (generally) routed to the orbit in order that they may retry their service later (cf. [4, 5, 6, 32, 42]). Other models include both a primary queue and a retrial queue so that arrivals who are initially blocked from service wait in the primary queue (cf. [7, 17, 18, 31, 37, 39]). All of the aforementioned models assume that the queueing system operates in a static environment that does not influence the system in any way.

The literature related to retrial queues operating in random environments is now beginning to emerge. Klimenok [24] studied a $BMAP/SM/1$ system whose operating mechanism is governed by a random environment. In that article, the limiting distribution at embedded and arbitrary instants, as well as the main steady state performance measures, were examined. Roszik and Sztrik [36] used the Modelling, Specification and Evaluation Language (MOSEL) to analyze a finite-source retrial queue in a random environment when all random variables are assumed to be exponentially distributed. Other important models include the $BMAP/PH/1$ and $BMAP/PH/N$ systems analyzed by Kim et al. [22, 23] which encompass a very broad class of queueing systems with randomly varying rates, including the $M/M/1$ queue analyzed in [33, 34, 35]. To analyze these complex systems, the authors show that the system state process can be viewed as an asymptotically quasi-Toeplitz Markov chain. Using results from [25], they determine the stability condition and devise an algorithm for computing the steady state distribution. Other recent contributions include an examination of the finite-source $MAP/PH/N$ retrial system with negative arrivals operating in a random environment due to Wu et al. [43]. All of the systems described here exhibit complex arrival and service processes; however, these models do not explicitly consider the interplay between a fully-modulated system and the impact of an unreliable server.

The model we consider here, namely the unreliable $M/M/1$ retrial queue in a random environment, could be analyzed as a special case of $BMAP/PH/1$ retrial queue of [22] but for the failure mechanism of the server. We compromise some model complexity in the arrival and service

processes with the intent of (1) explicitly accounting for an unreliable server, and (2) considering optimization of the arrival and server rates for designing stable, efficient systems that meet quality-of-service guarantees. Like most complex retrial queueing models, ours exhibits the level-dependent quasi-birth-and-death (LDQBD) structure (see [14, 20, 21, 28]). We examine the stability condition of the system using classical techniques, namely Lyapunov functions. Furthermore, we employ matrix-analytic methods, via the algorithms of Bright and Taylor [14, 15], to obtain the (approximate) steady state distribution and important performance measures. Finally, we consider the problem of choosing environment-dependent arrival and service rates that minimize the mean waiting time in orbit, subject to a limited budget and a minimum threshold for the revenue generated by the system.

The remainder of the paper is organized as follows. Section 2 provides a formal description of the retrial queueing model and the modulating environment and shows that the queueing system exhibits the LDQBD structure. Section 3 briefly discusses the form of the limiting distribution and examines a sufficient stability condition. Section 4 reviews an algorithm to compute the steady state distribution and mean performance measures, and provides a numerical illustration. Finally, in Section 5, we consider the problem of choosing arrival and service rates that minimize the steady state mean time spent in orbit.

2 Model Description

Consider a single-server $M/M/1$ retrial queue operating in a random environment whose server is subject to both active and idle failures. That is, the server experiences failure whether it is idle or busy, but cannot fail if it is under repair. If a primary arrival finds the server operational (i.e., not failed) and idle, it seizes the server and immediately begins its service cycle. Barring a server failure during this service cycle, the customer completes service and departs the system. However, should the server fail during the service cycle, the customer is immediately removed from the server and sent to an infinite-capacity retrial queue (or orbit) to retry service later. On the other hand, an arriving customer who finds the server busy or failed is routed directly to the orbit since there is no queue for primary arrivals; however, these customers are not lost. Retrial customers persistently attempt to regain access to the server at random intervals, and each behaves independently of the primary arrivals and all other retrial customers. However, a retrial customer can only gain (or regain) access to the server if it is found up and idle at the time of a retrial attempt. The service discipline is preemptive-repeat (i.e., an interrupted customer's service cycle is repeated following a successful retrial attempt). All customers are assumed to persist until they gain access to the server and complete their service.

Our model is distinguished from other unreliable retrial queueing models in that its arrival, service, failure, repair, and retrial rates all vary randomly over time in the spirit of the $M/M/1$ queue in a random environment studied by Neuts [34]. Specifically, the arrival, service, failure, repair, and retrial rates are all modulated by an external process $\{Z(t) : t \geq 0\}$ – an irreducible, continuous-time Markov chain (CTMC) with finite state space $S = \{1, \ldots, m\}$, infinitesimal generator $Q = [q_{ij}]$, $i, j \in S$, and invariant probability vector $\boldsymbol{\pi} = (\pi_1, \pi_2, \ldots, \pi_m)$ that uniquely solves $\boldsymbol{\pi} Q = \mathbf{0}$ and $\boldsymbol{\pi}\, \boldsymbol{e} = 1$ where $\mathbf{0}$ is the zero vector of dimension m and $\boldsymbol{e}$ is a column vector of ones. When the environment is in state $j \in S$, primary customers arrive according to a Poisson process with rate λ_j, and the service time is exponentially distributed with mean $1/\mu_j$. When the server is not failed and is either idle or busy, server failures occur according to a Poisson process with rate ξ_j. Repair of the server is initiated immediately following a failure, and the duration of the repair time (or down period) is exponentially distributed with mean $1/\alpha_j$.

For any m-dimensional (row) vector $\boldsymbol{x} = (x_1, x_2, \ldots, x_m)$, denote its transpose by $\boldsymbol{x}'$, and let the diagonal matrix of the elements of $\boldsymbol{x}$ be $\Delta(\boldsymbol{x}) = \text{diag}(x_1, x_2, \ldots, x_m)$. Next, define the m-dimensional vectors $\boldsymbol{\lambda} = (\lambda_1, \lambda_2, \ldots, \lambda_m)$, $\boldsymbol{\mu} = (\mu_1, \mu_2, \ldots, \mu_m)$, $\boldsymbol{\xi} = (\xi_1, \xi_2, \ldots, \xi_m)$, and $\boldsymbol{\alpha} = (\alpha_1, \alpha_2, \ldots, \alpha_m)$. Each retrial customer attempts to gain access to the server independently of all other customers (primary or retrial), at exponentially-distributed time intervals with mean $1/\theta_j$ when $Z(t) = j$. Therefore, if $Z(t) = j$ and there are i customers in the orbit, the total retrial rate is $r(i, j) \equiv i\,\theta_j$. Denote the vector of retrial rates by $\boldsymbol{\theta} = (\theta_1, \theta_2, \ldots, \theta_m)$. The row vectors $\boldsymbol{\lambda}$, $\boldsymbol{\mu}$, $\boldsymbol{\xi}$, $\boldsymbol{\alpha}$, and $\boldsymbol{\theta}$ are all strictly positive. In the usual way, we assume that the arrival, service, failure, repair, and retrial processes are mutually independent; however, each is modulated by the random environment, $\{Z(t) : t \geq 0\}$.

For each $t \geq 0$, let $R(t)$ be the orbit size, $Z(t)$ be the state of the random environment, and $X(t)$ be the status of the server so that $X(t) = 0$ if the server is failed, $X(t) = 1$ if the server is up and idle, and $X(t) = 2$ if the server is up and busy at time t. The state of the queueing system is described by the continuous-time stochastic process, $(R, Z, X) \equiv \{(R(t), Z(t), X(t)) : t \geq 0\}$, with state space $E = \{(i, j, k) : i \geq 0,\ j \in S,\ k \in \{0, 1, 2\}\}$. Note that E contains one denumerable dimension (the orbit size) and two finite dimensions (the environment state and server's status). Because all inter-arrival, service, inter-failure, repair, and inter-retrial times are exponentially distributed, it is easy to see that (R, Z, X) is a continuous-time Markov chain (CTMC) on E. Proposition 1 asserts that this CTMC has a well-structured, block diagonal infinitesimal generator matrix.

Proposition 1 *The process (R, Z, X) with state space E is a level-dependent quasi-birth-and-death (LDQBD) process with block diagonal infinitesimal generator matrix*

$$Q^* = \begin{bmatrix} \Gamma_0 & \Lambda & 0 & 0 & 0 & \cdots \\ \Theta_1 & \Gamma_1 & \Lambda & 0 & 0 & \cdots \\ 0 & \Theta_2 & \Gamma_2 & \Lambda & 0 & \cdots \\ 0 & 0 & \Theta_3 & \Gamma_3 & \Lambda & \cdots \\ 0 & 0 & 0 & \Theta_4 & \Gamma_4 & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix} \tag{1}$$

whose $3m \times 3m$ block diagonal elements are given by

$$\Gamma_i = \begin{bmatrix} C_i & \Delta(\boldsymbol{\xi}) & \Delta(\boldsymbol{\lambda}) \\ \Delta(\boldsymbol{\alpha}) & D_1 & 0 \\ \Delta(\boldsymbol{\mu}) & 0 & D_2 \end{bmatrix}, \quad \Theta_i = \begin{bmatrix} 0 & 0 & i\Delta(\boldsymbol{\theta}) \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \quad \Lambda = \begin{bmatrix} 0 & 0 & 0 \\ 0 & \Delta(\boldsymbol{\lambda}) & 0 \\ 0 & \Delta(\boldsymbol{\xi}) & \Delta(\boldsymbol{\lambda}) \end{bmatrix},$$

where $C_i = Q - \Delta(\boldsymbol{\lambda} + \boldsymbol{\xi}) - i\Delta(\boldsymbol{\theta})$, $i \geq 0$, $D_1 = Q - \Delta(\boldsymbol{\lambda} + \boldsymbol{\alpha})$, and $D_2 = Q - \Delta(\boldsymbol{\lambda} + \boldsymbol{\mu} + \boldsymbol{\xi})$.

The LDQBD process (cf. [14, 21, 28]) is a natural extension of the QBD process wherein some or all of the matrices comprising the ith level are explicitly dependent on the level i. For our purposes here, the limiting distribution of the LDQBD process is needed to compute the steady state queueing performance measures and to formulate and solve optimization problems to determine optimal (or near optimal) operating rates for each of the m distinct environment states. Assuming its existence, define the limiting distribution of (R, Z, X) as the row vector $\boldsymbol{p} = (\boldsymbol{p}_0, \boldsymbol{p}_1, \boldsymbol{p}_2, \ldots)$ where for $i \geq 0$, $\boldsymbol{p}_i$ is a $3m$-dimensional row vector of limiting probabilities restricted to level i. If the process is ergodic, $\boldsymbol{p}$ is the unique positive solution to $\boldsymbol{p}Q^* = \boldsymbol{0}$ and $\boldsymbol{p}\boldsymbol{e} = 1$. A closed-form, necessary and sufficient condition for the positive recurrence of *general* LDQBD processes is not available; however, it is known that if $\boldsymbol{p}$ exists, then it has the matrix-geometric property.

Let us assume for momentarily that $(R(t), Z(t), X(t)) \to (\widetilde{R}, \widetilde{Z}, \widetilde{X})$ in distribution as $t \to \infty$. In such a case, $\boldsymbol{p} = [p(i,j,k)]_{(i,j,k)\in E}$, where $p(i,j,k) = \mathbb{P}(\widetilde{R} = i, \widetilde{Z} = j, \widetilde{X} = k)$, $(i,j,k) \in E$. The marginal distribution of the steady state orbit size is given by

$$\mathbb{P}(\widetilde{R} = i) = \sum_{j=1}^{m} \sum_{k=0}^{2} p(i,j,k) = \boldsymbol{p}_i \, \boldsymbol{e}, \quad i \geq 0,$$

and likewise, the steady state status of the server has probability mass function (p.m.f.)

$$\mathbb{P}(\widetilde{X} = k) = \sum_{i=0}^{\infty} \sum_{j=1}^{m} p(i,j,k), \quad k \in \{0,1,2\}.$$

In Section 3, we provide the stability condition of the queueing system using Lyapunov functions to establish a sufficient condition for the ergodicity of (R, Z, X).

3 Stability Analysis

In this section we discuss necessary and sufficient conditions for positive recurrence of the process (R, Z, X). The following theorem can be stated using results in Bright and Taylor [14].

Theorem 1 *The LDQBD process (R, Z, X) with infinitesimal generator matrix Q^* is positive recurrent if and only if there exists a strictly positive solution to the system of equations*

$$\boldsymbol{p}_0 \, (\Gamma_0 + R_0 \, \Theta_1) = \boldsymbol{0}, \tag{2}$$

subject to the normalization condition

$$\boldsymbol{p}_0 \left(\sum_{i=0}^{\infty} \prod_{n=0}^{i-1} R_n \right) \boldsymbol{e} = 1. \tag{3}$$

In such a case, the $3m$-order row vector $\boldsymbol{p}_i$ is given by

$$\boldsymbol{p}_i = \boldsymbol{p}_0 \prod_{n=0}^{i-1} R_n, \quad i \geq 0. \tag{4}$$

In equations (3) and (4), when $i = 0$, the empty product results in the identity matrix I. It is well known (cf. [14, 28]) that the sequence $\{R_i : i \geq 0\}$ is the minimal non-negative solution of the set of equations

$$\Lambda + R_i \, \Gamma_{i+1} + R_i \, (R_{i+1} \Theta_{i+2}) = 0, \quad i \geq 0 \tag{5}$$

which must be determined numerically.

In general, it is difficult to assert positive recurrence using the conditions of Theorem 1. For our retrial queueing system, Theorem 2 establishes a sufficient condition for stability using Lyapunov functions and a classical result due to Tweedie [41], which is stated here as Lemma 1.

Lemma 1 *A continuous-time Markov chain with generator matrix $Q^* = [q^*_{xx'}]$, $x, x' \in E$ is regular and ergodic if there exists a function $v : E \to \mathbb{R}_+$ which is bounded below, a finite set $H \subset E$, and some $\epsilon > 0$ such that*

$$d(x) \equiv \sum_{x' \in E \setminus \{x\}} q^*_{xx'} [v(x') - v(x)] < \infty, \quad \textit{for all } x \in E, \tag{6}$$

and

$$d(x) \leq -\epsilon, \quad \textit{for all } x \in E \setminus H. \tag{7}$$

Equipped with Lemma 1, we are now prepared to state a sufficient condition for the ergodicity of the continuous-time process (R, Z, X). Theorem 2 represents a somewhat strong condition.

Theorem 2 *The process (R, Z, X) is ergodic if*

$$\boldsymbol{\lambda}' < [\Delta(\boldsymbol{\alpha} + \boldsymbol{\xi})]^{-1} \Delta(\boldsymbol{\alpha})\, \boldsymbol{\mu}', \tag{8}$$

where the inequality holds componentwise.

Proof. Since all the inter-event times are assumed to be exponential, and the environment is a CTMC, we consider the following Lyapunov function. For $i \geq 0$, $j \in S$, and some real numbers $a, b > 0$, define

$$v(i, j, k) = \begin{cases} b + a\, i, & \text{if } k = 0, \\ 1 + a\, i, & \text{if } k = 1, \\ a\, i, & \text{if } k = 2. \end{cases}$$

Substituting the function v in (6), and using the transition rates in Q^* of (1), we obtain

$$\begin{aligned} d(i, j, 0) &= a\, \lambda_j - \alpha_j\, b, \\ d(i, j, 1) &= \lambda_j + \xi_j\, b + i\theta_j(1 - a), \\ d(i, j, 2) &= a\, \lambda_j - \mu_j + \xi_j(b - 1 + a). \end{aligned}$$

It is not difficult to see that $d(i, j, k) < \infty$ for all (i, j, k), so we need only to determine if the drift functions satisfy (7). Clearly, for all $j \in S$, there exists a positive integer N such that $d(i, j, 1) < 0$ for each $i \geq N$ if $a > 1$. Therefore, inequality (7) is verified if there exist positive constants a and b satisfying the set of linear inequalities,

$$1 - a < 0, \tag{9}$$

$$a\lambda_j - \alpha_j b < 0, \tag{10}$$

$$a\lambda_j - \mu_j + \xi_j(b - 1 + a) < 0. \tag{11}$$

By inequality (10), we must have $b > a\lambda_j/\alpha_j$. Using this fact in inequality (11), we can eliminate b to obtain

$$a < \frac{\alpha_j \xi_j + \alpha_j \mu_j}{\lambda_j \xi_j + \alpha_j \xi_j + \lambda_j \alpha_j}.$$

But by (9), we must have $a > 1$; therefore, the set of linear inequalities (9)–(11) has a solution if and only if there is a positive number a such that $a > 1$ and

$$a < \frac{\alpha_j(\xi_j + \mu_j)}{\lambda_j(\alpha_j + \xi_j) + \alpha_j \xi_j},$$

or equivalently, if

$$\lambda_j < \left(\frac{\alpha_j}{\alpha_j + \xi_j} \right) \mu_j, \quad j = 1, 2, \ldots, m. \tag{12}$$

Rewriting this expression in vector/matrix form, we obtain $\boldsymbol{\lambda}' < \Delta(\boldsymbol{\alpha} + \boldsymbol{\xi})^{-1} \Delta(\boldsymbol{\alpha}) \boldsymbol{\mu}'$, where the inequality holds componentwise. ∎

The right-hand side of (12) can be viewed as the effective service rate when the environment is in state j as the quantity $\alpha_j/(\alpha_j + \xi_j)$ is the effective proportion of time the server is not failed in environment state j. Likewise, λ_j is the effective arrival rate of customers in state j. Under

the strong condition that the effective arrival rate is less than the effective service rate for *all* environment states, stability is to be expected; however, condition (12) need not hold for *every* $j \in S$. To see this, note that the *average* effective arrival rate is given by $\lambda = \boldsymbol{\pi}\boldsymbol{\lambda}'$, as the arrival process mirrors that of the standard $M/M/1$ queue in a random environment (see Neuts [34]). Similarly, using a Markov reward argument, it can be shown that the *average* effective service rate is given by $\mu = \boldsymbol{\pi}\Delta(\boldsymbol{\alpha}+\boldsymbol{\xi})^{-1}\Delta(\boldsymbol{\alpha})\boldsymbol{\mu}'$. Standard results for stability conditions of single-server retrial queues (see [8, 19]) show that $\lambda < \mu$ is necessary for stability of the system. Hence, the system can be stable only if

$$\boldsymbol{\pi}\boldsymbol{\lambda}' < \boldsymbol{\pi}\Delta(\boldsymbol{\alpha}+\boldsymbol{\xi})^{-1}\Delta(\boldsymbol{\alpha})\boldsymbol{\mu}'. \tag{13}$$

But it is possible that $\lambda_j \geq \alpha_j\,\mu_j/(\alpha_j+\xi_j)$ for some $j \in S$ while (13) still holds. Therefore, we see that while (12) implies (13), it is not necessary for stability of the system. For convenience, and for use in Sections 4 and 5, let us define the overall traffic intensity by

$$\rho = \frac{\boldsymbol{\pi}\boldsymbol{\lambda}'}{\boldsymbol{\pi}\Delta(\boldsymbol{\alpha}+\boldsymbol{\xi})^{-1}\Delta(\boldsymbol{\alpha})\boldsymbol{\mu}'}. \tag{14}$$

Remarks: Note that if $\boldsymbol{\xi} = \mathbf{0}$, the server never fails, and the stability condition reduces to $\boldsymbol{\pi}\boldsymbol{\lambda}' < \boldsymbol{\pi}\boldsymbol{\mu}'$. This is precisely the stability condition for the standard $M/M/1$ queue in a random environment analyzed by Neuts [34]. Moreover, if $\lambda_j = \lambda$, $\mu_j = \mu$, $\xi_j = \xi$, and $\alpha_j = \alpha$ for all $j \in S$, the (necessary and sufficient) stability condition is $\lambda < \alpha\mu/(\alpha+\xi)$. The same result has been derived for other single-server, exponential retrial models with an unreliable server (cf. Falin [17]), or it can be derived from general service time models by assuming exponential service times (cf. Sherman et al. [38, 39], Kulkarni and Choi [26], and others).

4 Evaluating Performance Measures

The most widely used algorithms for approximating the limiting distribution of a LDQBD process with an infinite number of levels or phases are due to Bright and Taylor [14, 15]. Because these algorithms play a central role in solving the optimization problem of Section 5, we next summarize their essential elements. These techniques extend the well-known logarithmic reduction algorithm of Latouche and Ramaswami [27] QBD processes to the level-dependent case.

The generator matrix of (1) possesses two nice properties that facilitate relatively easy implementation of the algorithms. First, the number of phase states in each level is fixed at $3m$, and second, the matrix Λ is independent of the level i. Essentially, the main algorithm of [14] truncates the infinite series of (3) at some level K and then re-normalizes to compute an approximate subvector of the form $\boldsymbol{p}_i(K) = \boldsymbol{p}_0(K)\prod_{n=0}^{i-1} R_n$, $i \geq 1$, where $\boldsymbol{p}_0(K)$ satisfies (2) and the normalization condition, $\boldsymbol{p}_0(K)\sum_{i=0}^{K}\left[\prod_{n=0}^{i-1} R_n\right]\boldsymbol{e} = 1$. The subvectors, $\{\boldsymbol{p}_i(K) : i \geq 0\}$, represent an invariant measure for the limiting distribution of all states at or below level K; therefore, $\boldsymbol{p}_i \leq \boldsymbol{p}_i(K)$ componentwise for any $K \geq 0$, and $\boldsymbol{p}_i(K) \to \boldsymbol{p}_i$ componentwise as $K \to \infty$. For a given truncation point K, Bright and Taylor [14] examine the discrete-time Markov chain embedded at the jump epochs of the process to obtain the family of matrices $\{R_i : i \geq 0\}$ using a recursive scheme. Lemma 2 is a direct consequence of (1) and Lemma 1 of [14].

Lemma 2 *If (R, Z, X) is positive recurrent, the matrix R_i is given by*

$$R_i = \sum_{\ell=0}^{\infty} U_i^{\ell} \prod_{n=0}^{\ell-1} D_{i+2^{\ell-n}}^{\ell-1-n}, \quad i \geq 0 \tag{15}$$

where for $i \geq 1$, U_i^ℓ and D_i^ℓ are $3m \times 3m$ matrices recursively defined by

$$\begin{aligned} U_i^0 &= \Lambda\,(-\Gamma_{i+1})^{-1}, \\ D_i^0 &= \Theta_i\,(-\Gamma_{i-1})^{-1}, \\ U_i^{\ell+1} &= U_i^\ell U_{i+2^\ell}^\ell \left[I - U_{i+2^{\ell+1}}^\ell D_{i+3\cdot 2^\ell}^\ell - D_{i+2^{\ell+1}}^\ell U_{i+2^\ell}^\ell\right]^{-1}, \\ D_i^{\ell+1} &= D_i^\ell D_{i-2^\ell}^\ell \left[I - U_{i-2^{\ell+1}}^\ell D_{i-2^\ell}^\ell - D_{i-2^{\ell+1}}^\ell U_{i-3\cdot 2^\ell}^\ell\right]^{-1}. \end{aligned}$$

The infinite series of equation (15) can be truncated using a simple scheme (see Algorithms 2 and 3 of [14]). By rearranging the terms in (5), the family of matrices, $\{R_i : i \geq 0\}$, are recursively computed by

$$R_i = \Lambda\,(-\Gamma_{i+1} - R_{i+1}\,\Theta_{i+2})^{-1} \tag{16}$$

(assuming the inverse exists) so that (15) need not be computed repeatedly. For all of the numerical results that follow, Algorithms 1–3 of [14] are used to select the integer K and compute $\boldsymbol{p}_i(K)$.

Assuming $\rho < 1$, the approximate steady state performance measures of the queueing system (R, Z, X) may be obtained using the approximation of the steady state distribution given by the vector $\boldsymbol{p} = [p(i,j,k)]$, where $p(i,j,k) = \mathbb{P}(\widetilde{R} = i, \widetilde{Z} = j, \widetilde{X} = k)$, $(i,j,k) \in E$. More specifically, the steady state orbit size distribution is

$$\mathbb{P}(\widetilde{R} = i) = \sum_{j=1}^{m}\sum_{k=0}^{2} p(i,j,k) = \boldsymbol{p}_i\,\boldsymbol{e}, \quad i \geq 0. \tag{17}$$

Likewise, the steady state distribution of the server's status is

$$\gamma_k \equiv \mathbb{P}(\widetilde{X} = k) = \sum_{i=0}^{\infty}\sum_{j=1}^{m} p(i,j,k), \quad k = 0, 1, 2. \tag{18}$$

(Of course, the distribution of $\widetilde{Z}$ is the invariant vector $\boldsymbol{\pi}$, which is independent of $(\widetilde{R}, \widetilde{X})$.)

From (17) and (18), we can obtain the steady state delay and congestion parameters in the usual way using Little's Law. Specifically, using the (approximate) distribution $\boldsymbol{p}$, the steady state mean orbit size is

$$\mathbb{E}(\widetilde{R}) = \sum_{i=1}^{\infty} i\,(\boldsymbol{p}_i\boldsymbol{e}). \tag{19}$$

Let $\widetilde{N}$ denote the steady state number of customers in the system (in the retrial queue and in service). The mean of $\widetilde{N}$ is easily computed by noting that $\widetilde{N} = \widetilde{R}$ if $\widetilde{X} = 0$ or $\widetilde{X} = 1$, and $\widetilde{N} = \widetilde{R} + 1$ if $\widetilde{X} = 2$. Therefore, by conditioning on $\widetilde{X}$, we see that

$$\mathbb{E}(\widetilde{N}) = \mathbb{E}(\widetilde{R})\,(\gamma_0 + \gamma_1) + \mathbb{E}(\widetilde{R} + 1)\gamma_2 = \mathbb{E}(\widetilde{R}) + \gamma_2. \tag{20}$$

We note that the mean number in system is not $\mathbb{E}(\widetilde{R}) + \rho$ because there are periods during which the orbit is not empty, but the server is idle. Next, let $\widetilde{W}$ be the sojourn time (time in service and in orbit) of an arbitrary customer who arrives in steady state. It is well known that, for an ordinary (non-modulated), single-server retrial queue with Poisson arrivals and exponential inter-retrial times, the mean sojourn time is the mean number in system divided by the arrival rate. Analogously, we can apply Little's Law to obtain

$$\mathbb{E}(\widetilde{W}) = (\boldsymbol{\pi}\boldsymbol{\lambda}')^{-1}\mathbb{E}(\widetilde{N}), \tag{21}$$

where $(\boldsymbol{\pi}\boldsymbol{\lambda}')^{-1}$ is the average effective arrival rate of customers to the system. In a similar manner, the steady state mean time spent in the orbit is given by

$$\mathbb{E}(\widetilde{W}_r) = (\boldsymbol{\pi}\boldsymbol{\lambda}')^{-1}\mathbb{E}(\widetilde{R}). \tag{22}$$

Finally, to compare various environment states, we define the traffic intensity in state j by

$$\rho_j = \lambda_j \left[\left(\frac{\alpha_j}{\alpha_j + \xi_j} \right) \mu_j \right]^{-1}. \tag{23}$$

Next, we provide a numerical example to illustrate the steady state orbit size distribution, among other steady state performance measures.

Example: Suppose the environment has state space $S = \{1, 2, 3, 4, 5, 6, 7\}$ and infinitesimal generator matrix

$$Q = \begin{bmatrix} -7.6 & 2.0 & 3.0 & 1.0 & 1.0 & 0.1 & 0.5 \\ 0.5 & -8.0 & 2.5 & 1.0 & 2.0 & 0.8 & 1.2 \\ 0.3 & 1.5 & -5.8 & 1.0 & 1.0 & 1.0 & 1.0 \\ 2.0 & 3.0 & 5.0 & -11.2 & 0.1 & 1.0 & 0.1 \\ 0.8 & 2.5 & 2.0 & 1.1 & -7.9 & 0.7 & 0.8 \\ 1.5 & 1.0 & 1.6 & 1.2 & 0.5 & -6.3 & 0.5 \\ 2.0 & 2.5 & 2.0 & 1.0 & 1.8 & 0.9 & -10.2 \end{bmatrix}.$$

Table 1 reveals that the system is critically loaded when the environment is in state 3 ($\rho_3 = 0.96$), and it is overloaded in environment state 2 ($\rho_2 > 1$). The environment spends a relatively short amount of time in the overloaded condition (less than 20%) and about 30% of the time in a critically loaded condition. However, the overall traffic intensity is only $\rho = 0.6895$. On first glance, this result seems counterintuitive until we notice the relatively small arrival and failure rates of state 2. In environment state 3, the service rate is comparatively high, so even though the system is critically loaded in this state, the system will experience recovery periods when occupying the less detrimental states.

Table 1: Summary of parameter values and traffic intensities.

Environment (j)	λ_j	μ_j	ξ_j	α_j	θ_j	π_j	ρ_j	ρ
1	3.0	7.0	0.5	2.0	1.0	0.1021	0.5357	0.6895
2	1.0	3.0	1.1	0.5	11.0	0.1917	1.0667	
3	3.0	12.5	1.5	0.5	2.0	0.3086	0.9600	
4	2.0	12.5	4.0	2.0	2.0	0.0848	0.4800	
5	2.0	4.5	1.0	6.0	5.0	0.1256	0.5185	
6	0.5	2.0	0.7	3.0	1.0	0.1130	0.3083	
7	0.5	4.0	1.5	0.5	0.5	0.0740	0.5000	

The values of the approximated performance measures are summarized in Table 2. It is noteworthy that the limiting probability that the server is busy ($\mathbb{P}(\widetilde{X} = 2)$) is less than the traffic intensity ρ. This is attributed to the fact that there are periods in which the retrial queue is not empty, but the server remains idle waiting for either a primary customer arrival or the next retrial attempt.

Table 2: Performance measures for the numerical example.

$\mathbb{E}(\widetilde{R})$	$\mathbb{E}(\widetilde{N})$	$\mathbb{E}(\widetilde{W}_r)$	$\mathbb{E}(\widetilde{W})$	$\mathbb{P}(\widetilde{X}=0)$	$\mathbb{P}(\widetilde{X}=1)$	$\mathbb{P}(\widetilde{X}=2)$
5.9904	6.2963	3.0902	3.2480	0.4685	0.2256	0.3059

For the sake of completeness, Figure 1 graphs the (approximate) probability distribution of $\widetilde{R}$, namely $\mathbb{P}(\widetilde{R}=i)=\boldsymbol{p}_i\,\boldsymbol{e}$ for $i=0,1,,\ldots,75$. Note the geometric rate of decay exhibited by the steady state distribution, which is expected because $\boldsymbol{p}$ is a matrix geometric distribution.

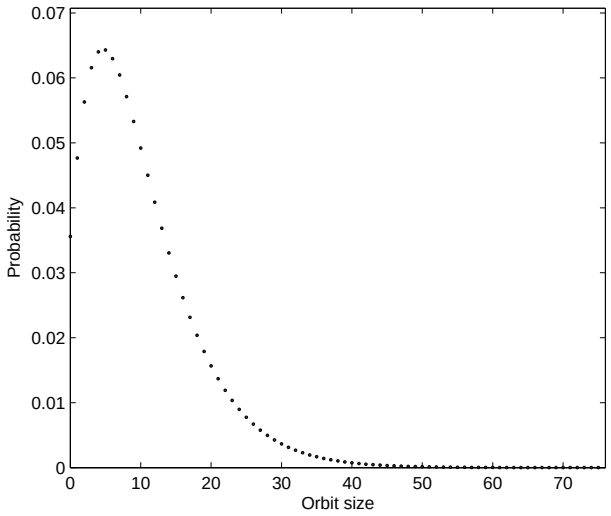

Figure 1: Approximate steady state orbit size distribution.

In Section 5 we use the approximated steady state orbit size distribution and queueing performance measures to determine optimal arrival and service rates to minimize the mean time spent in orbit.

5 Optimizing Arrival and Service Rates

In this section, we consider the problem of choosing the arrival and service rates, as a function of the environment state, to minimize the mean time spent in orbit by an arbitrary customer in steady state. This problem can be viewed as a (static) design problem in which the system designer chooses $\lambda_j \in \left[\underline{\lambda},\, \overline{\lambda}\right]$ and $\mu_j \in \left[\underline{\mu},\, \overline{\mu}\right]$ for each $j \in S$ where $0 \le \underline{\lambda} < \overline{\lambda} < \infty$ and $0 \le \underline{\mu} < \overline{\mu} < \infty$. The rate setting is done only once so that whenever $Z(t)=j$, a controller limits the arrival rate to the optimal λ_j value and tunes the service rate to the optimal μ_j value, $j=1,2,\ldots,m$. For every unit of arrival rate, the system gains a reward (revenue) r_j, and for every unit of service rate, the system incurs a cost c_j. Let $\boldsymbol{r}=(r_1,r_2,\ldots,r_m)$ and $\boldsymbol{c}=(c_1,c_2,\ldots,c_m)$ be the revenue and cost vectors, respectively. The total revenue generation rate (over all environment states) must meet or exceed a minimum threshold value R $(0<R<\infty)$, while the total cost rate is subject to an upper limit B $(0<B<\infty)$. It is reasonable to assume that only admission and service rates can be chosen at our discretion as the failure and repair rates are dictated by the inherent limitations of the equipment, etc. Similarly, the retrial rates may be dictated by customer behavior or attitudes

and are assumed to be outside of the controller's purview.

The objective is to minimize the mean time spent in orbit by an arbitrary customer in steady state given by

$$\vartheta(\boldsymbol{\lambda}, \boldsymbol{\mu}) = (\boldsymbol{\pi}\boldsymbol{\lambda}')^{-1} \sum_{i=1}^{\infty} i\, \mathbb{P}(\widetilde{R} = i) = (\boldsymbol{\pi}\boldsymbol{\lambda}')^{-1} \sum_{i=1}^{\infty} i\, (\boldsymbol{p}_i\, \boldsymbol{e}). \tag{24}$$

Here, we include the implicit dependence on $\boldsymbol{\mu}$ through the vectors $\boldsymbol{p}_i$, $i \geq 0$, which are approximated by the algorithms summarized in Section 4. To compute $\vartheta(\boldsymbol{\lambda}, \boldsymbol{\mu})$, we truncate the infinite series in (24) at the nth term ($n \in \mathbb{N}$) if $|S_{n+1} - S_n| < \epsilon$ where

$$S_n \equiv \sum_{i=1}^{n} i\, (\boldsymbol{p}_i\, \boldsymbol{e}),$$

and $\epsilon > 0$ is a convergence threshold. With these preliminaries and notation, the nonlinear programming formulation is as follows:

$$\begin{aligned}
\min \quad & \vartheta(\boldsymbol{\lambda}, \boldsymbol{\mu}) \\
\text{s.t.} \quad & \rho < 1, && \text{(25a)} \\
& \boldsymbol{r}\, \boldsymbol{\lambda}' \geq R, && \text{(25b)} \\
& \boldsymbol{c}\, \boldsymbol{\mu}' \leq B, && \text{(25c)} \\
& \lambda_j \in \left[\, \underline{\lambda},\, \overline{\lambda} \,\right], \quad j = 1, \ldots, m, && \text{(25d)} \\
& \mu_j \in \left[\, \underline{\mu},\, \overline{\mu} \,\right], \quad j = 1, \ldots, m. && \text{(25e)}
\end{aligned}$$

Constraint (25a) enforces the necessary stability condition discussed in Section 3 while (25b) and (25c) are the revenue and cost constraints. Constraints (25d) and (25e) are box constraints that ensure realistic rate settings. For a candidate solution $(\boldsymbol{\lambda}, \boldsymbol{\mu})$, the left-hand side of (25a) is computed directly via (14).

Though easily stated, the optimization problem (25) is not easily solved for at least the following reasons. First, the objective function is expensive as it requires an approximation of $\boldsymbol{p} = (\boldsymbol{p}_0, \boldsymbol{p}_1, \boldsymbol{p}_2, \ldots)$ for each candidate solution $(\boldsymbol{\lambda}, \boldsymbol{\mu})$. Second, the feasible region is not closed (due to the strict inequality constraint (25a)), and in general, derivative information about the objective function is not available. These complications motivate our use of derivative-free, adaptive search algorithms with proven convergence properties. We next discuss a class of these algorithms.

5.1 Solving the Rate-Setting Problem

Because the steady state vector $\boldsymbol{p} = (\boldsymbol{p}_0, \boldsymbol{p}_1, \boldsymbol{p}_2, \ldots)$ is only available numerically (via the matrix-analytic methods described in Section 4), the objective function (24) is computationally expensive. Therefore, to solve problem (25), we employ *generalized pattern search* (GPS), and specifically, derivative-free, mesh-adaptive search (MADS) techniques. These techniques do not require a closed-form objective function, as long as the objective function can be evaluated numerically at points inside the feasible region. Detailed descriptions of these algorithms are summarized in [2, 12, 13].

GPS is a derivative-free optimization technique for unconstrained problems originally introduced by Torczon [40] who proved convergence of a subsequence of iterates to a first-order stationary point. It has known convergence properties for a variety of problem classes, even when the objective function is nonsmooth (see Audet and Dennis [11]). GPS methods iteratively search a set of points around the current iterate for one that improves the objective function value. Consider the general nonlinear minimization problem,

$$\min_{\boldsymbol{x} \in \Omega} \vartheta(\boldsymbol{x}), \tag{26}$$

where $\Omega = \{\boldsymbol{x} \in \mathbb{R}^n : \boldsymbol{\ell} \leq \mathbf{A}\boldsymbol{x} \leq \boldsymbol{u}\}$, $\vartheta : \mathbb{R}^n \to \mathbb{R}$ and $\mathbf{A} \in \mathbb{Q}^{m \times n}$ is a rational matrix. Moreover, $\boldsymbol{\ell}$ and $\boldsymbol{u}$ are the lower and upper bounds of the constraints where $\boldsymbol{\ell}, \boldsymbol{u} \in \{\mathbb{R}^m \cap \{\pm\infty\}\}$ and $\boldsymbol{\ell} \leq \boldsymbol{u}$. GPS algorithms generate a sequence of iterates $\{\boldsymbol{x}_k\}$ in $\mathbb{R}^n$ with nonincreasing objective function values. Each iteration is divided into an optional *search* step and a local *poll* step. Both the *search* and *poll* steps evaluate points on a mesh in order to find a mesh point that improves the objective function value. The *mesh* is constructed as a lattice of points in $\mathbb{R}^n$, based on a finite set of directions D that form a positive spanning set and a *mesh size parameter*, Δ_k ($\Delta_k > 0$), that controls the fineness of the mesh. In this case, a positive spanning set refers to a set of vectors such that any vector in the space can be represented by a nonnegative linear combination of the vectors in the set. By definition, nonnegative linear combinations of the elements of the set D span $\mathbb{R}^n$. The directions that form D can be arbitrarily chosen provided that, for each direction $d_j \in D$, $j = 1, 2, \ldots, |D|$, $d_j = \mathbf{G}\bar{z}_j$, where $\mathbf{G} \in \mathbb{R}^{n \times n}$ is a nonsingular matrix and $\bar{z}_j \in \mathbb{Z}^n$ is an integer vector. At iteration k, the mesh is centered around the current iterate $\boldsymbol{x}_k \in \mathbb{R}^n$ and its fineness is parameterized through the mesh size parameter Δ_k. The mesh can then be represented as

$$M_k = \left\{ \boldsymbol{x}_k + \Delta_k \mathbf{D}\, z : z \in \mathbb{Z}_+^{|D|} \right\} \tag{27}$$

where $\mathbb{Z}_+$ is the set of nonnegative integers. Note that in (27), the columns of the matrix $\mathbf{D}$ form the set D.

In the *search* step, GPS can evaluate any finite set of mesh points, and a number of strategies exist for generating trial points, including random search, genetic algorithms, Latin hypercube search, or orthogonal arrays. If the *search* step fails to provide an improved mesh point, the *poll* step is invoked. The *poll* step is more rigidly defined and evaluates the neighboring mesh points of the current iterate. The use of positive spanning directions in the construction of these neighboring points provides the theoretical basis for the convergence of GPS. The *poll* set at iteration k can be expressed as $\{\boldsymbol{x}_k + \Delta_k d : d \in \mathbf{D}_k\}$, where $\mathbf{D}_k \subseteq D$ is also a matrix whose columns positively span $\mathbb{R}^n$. The *poll* set is therefore composed of mesh points neighboring the current iterate $\boldsymbol{x}_k$ in the directions of the columns of $\mathbf{D}_k$, a multiple Δ_k away from the current iterate.

If the *search* and *poll* step both fail, the incumbent solution is said to be a *mesh local optimizer* and the mesh is then refined by setting the mesh size parameter

$$\Delta_{k+1} = \psi^{w_k} \Delta_k, \tag{28}$$

where $\psi > 1$ is rational and $w_k \in \{w^-, w^- + 1, \ldots, -1\}$ for some w^-. An incumbent point $\boldsymbol{x}_k$ is replaced by $\boldsymbol{x}_{k+1}$ only if $\vartheta(\boldsymbol{x}_{k+1}) < \vartheta(\boldsymbol{x}_k)$, and in such case, $\boldsymbol{x}_{k+1}$ is termed an *improved mesh point*. If an improved mesh point is found in either step, then the mesh is either retained or coarsened by increasing the mesh size parameter according to equation (28) for some $w_k \in \{0, 1, \ldots, w^+\}$. It follows that for any $k \geq 0$ there exists an integer $r_k \in \mathbb{Z}$ such that $\Delta_k = \psi^{r_k} \Delta_0$.

The convergence analysis of pattern search is well-established in [10, 40] and requires a few assumptions. First, all iterates produced by GPS must lie in a compact set [16]. This very common assumption holds as long as $\{\boldsymbol{x} \in \Omega : \vartheta(\boldsymbol{x}) \leq \vartheta(\boldsymbol{x}_0)\}$ is compact. Second, if the matrix $\mathbf{G} = I$ (as is usually the case), then the constraint matrix $\mathbf{A}$ must be rational. The final necessary assumption is that $\vartheta(\boldsymbol{x}_0) < \infty$ for $\boldsymbol{x}_0 \in \mathbb{R}^n$. Torczon [40] proved that, under these assumptions, the mesh size parameter satisfies $\liminf_{k \to \infty} \Delta_k = 0$, which leads to the main directional convergence result (see [11]). However, Audet [9] proved convergence to a Clarke first-order stationary point in the one-dimensional case for unconstrained problems. Of particular relevance to our work here, GPS was extended in [29, 30] to problems with bound and linear constraints, respectively. To handle these constraints while maintaining convergence properties, infeasible points are discarded without being evaluated, and search directions are chosen so as to conform to the geometry of

the nearby constraint boundaries. The NOMADm optimization software by Abramson [1], written in the MATLAB® computing environment, was used to implement the pattern search procedure described here. NOMADm is specifically designed to numerically solve nonlinear and mixed variable optimization problems via an implementation of the class of mesh-adaptive direct search (MADS) algorithms. GPS is a subclass of MADS [12], in which *poll* directions are restricted to a uniformly bounded finite set.

5.2 Optimization Illustration

In this subsection, we formulate and solve an instance of problem (25) using generalized pattern search (namely MADS). The vectors $\boldsymbol{\xi}$, $\boldsymbol{\alpha}$, $\boldsymbol{\theta}$, $\boldsymbol{r}$, and $\boldsymbol{c}$ are specified and fixed (i.e., they are simply treated as parameters in the model). The decision variables are the arrival and service rates contained in $\boldsymbol{\lambda}$ and $\boldsymbol{\mu}$, respectively. Initial feasible vectors $\boldsymbol{\lambda}_0$ and $\boldsymbol{\mu}_0$ were specified for each of three independent replications to ensure that the algorithm produced consistent solutions. The aim is to choose the arrival and service rates that minimize the mean time customers spend in orbit in steady state.

Example: Suppose the retrial system's environment has state space $S = \{1, 2, 3, 4, 5\}$ and infinitesimal generator matrix

$$Q = \begin{bmatrix} -23 & 6 & 7 & 9 & 1 \\ 6 & -19 & 1 & 4 & 8 \\ 6 & 2 & -18 & 9 & 1 \\ 4 & 1 & 3 & -11 & 3 \\ 8 & 1 & 3 & 1 & -13 \end{bmatrix}.$$

The invariant probability vector of Q is $\boldsymbol{\pi} = (0.1946, 0.1071, 0.1697, 0.3530, 0.1754)$ while the revenue threshold value is $R = 6$, and the upper limit on the budget is $B = 20$. The box constraints (25d) and (25e) for this example are $\lambda_j \in [1, 5]$ and $\mu_j \in [0, 4]$, $j \in S$. The other input parameters are as follows: $\boldsymbol{\xi} = (0.5, 1.1, 1.5, 4.0, 1.0)$, $\boldsymbol{\alpha} = (2.0, 0.5, 8.5, 4.5, 6.0)$, $\boldsymbol{\theta} = (1.0, 4.0, 2.0, 8.0, 5.0)$, $\boldsymbol{r} = (1.0, 1.0, 0.5, 2.0, 0.75)$, and $\boldsymbol{c} = (2.0, 1.5, 0.5, 2.0, 0.5)$.

Table 3 summarizes the best obtained solutions using three distinct initial feasible solutions. The objective function values of runs 1 and 3 are very similar, as are their solutions with the exception of the first two elements of $\boldsymbol{\mu}^*$. Run 2 produced a solution that differs significantly from the others and yields a clearly inferior objective function value. The average number of iterations needed for convergence of the MADS algorithm was just over 400 (i.e., on average, 400 approximated steady state distributions were computed).

Table 3: MADS best obtained solutions for the example problem.

Run no.	Initial solution	Best solution obtained	$\vartheta(\boldsymbol{\lambda}^*, \boldsymbol{\mu}^*)$
1	$\boldsymbol{\lambda}_0 = (1.0, 1.0, 1.0, 1.0, 1.0)$	$\boldsymbol{\lambda}^* = (1.008, 1.287, 1.001, 1.211, 1.046)$	30.791
	$\boldsymbol{\mu}_0 = (2.0, 2.0, 2.0, 2.0, 2.0)$	$\boldsymbol{\mu}^* = (1.917, 3.983, 3.965, 3.105, 3.999)$	
2	$\boldsymbol{\lambda}_0 = (1.0, 1.0, 1.0, 1.0, 1.0)$	$\boldsymbol{\lambda}^* = (1.102, 1.263, 1.000, 1.171, 1.057)$	33.293
	$\boldsymbol{\mu}_0 = (1.5, 1.5, 1.5, 1.5, 1.5)$	$\boldsymbol{\mu}^* = (2.389, 3.416, 3.741, 3.476, 2.552)$	
3	$\boldsymbol{\lambda}_0 = (1.0, 1.0, 1.0, 1.0, 1.0)$	$\boldsymbol{\lambda}^* = (1.083, 1.409, 1.001, 1.123, 1.017)$	29.580
	$\boldsymbol{\mu}_0 = (2.5, 2.5, 2.5, 2.5, 2.5)$	$\boldsymbol{\mu}^* = (3.403, 1.962, 3.950, 3.139, 3.995)$	

It is worth noting that the time and computational effort to solve the 5-state example greatly exceeded the time and effort needed to solve a similar 3-state model which required just under 100 iterations. It is surmised that the increased computational effort stems from the exponential increase in the effort needed to compute $\boldsymbol{p}_0, \boldsymbol{p}_1, \boldsymbol{p}_2, \ldots$ at each objective function evaluation. Despite the computational effort, the solutions were obtained in less than 10 minutes. These optimization models can be used to help a system controller determine the appropriate set of arrival and service rates to choose, depending on the prevailing conditions.

Acknowledgements: We thank Dr. Srinivas Chakravarthy and two anonymous referees for their helpful comments. This research was sponsored in part by a grant from the U.S. Air Force Office of Scientific Research (FA9550-08-1-0004). The views expressed in this paper are those of the authors and do not reflect the official policy or position of the United States Air Force, Department of Defense, or the U.S. Government

References

[1] M.A. Abramson. MATLAB implementation of NOMADm Optimization Software, 2011. `http://www.gerad.ca/NOMAD/Abramson/NOMADm.html`.

[2] M.A. Abramson and C. Audet. Convergence of mesh adaptive direct search to second-order stationary points. *SIAM Journal on Optimization*, 17(2):606–619, 2006.

[3] A. Aissani. On the $M/G/1/1$ queueing system with repeated orders and unreliable server. *Journal of Technology*, 6:98–123, 1988. (in French).

[4] A. Aissani. Unreliable queuing with repeated orders. *Microelectronics and Reliability*, 33(14):2093–2106, November 1993.

[5] A. Aissani. A retrial queue with redundancy and unreliable server. *Queueing Systems: Theory and Applications*, 17(3-4):431–449, 1994.

[6] J. R. Artalejo. Analysis of an $M/G/1$ queue with constant repeated attempts and server vacations. *Computers & Operations Research*, 24(6):493–504, 1997.

[7] J.R. Artalejo. New results in retrial queueing systems with breakdown of the servers. *Statistica Neerlandica*, 48(1):23–36, 1994.

[8] J.R. Artalejo and A. Gómez-Corral. *Retrial Queueing Systems: A Computational Approach*. Springer, Berlin, Germany, 2008.

[9] C. Audet. Convergence results for pattern search algorithms are tight. *Technical Report 98-24, Department of Computational and Applied Mathematics, Rice University, Houston, TX*, 1998.

[10] C. Audet and J.E. Dennis. Pattern search algorithms for mixed variable programming. *SIAM Journal on Optimization*, 11:573–594, 2000.

[11] C. Audet and J.E. Dennis. Analysis of generalized pattern searches. *SIAM Journal on Optimization*, 13:889–903, 2003.

[12] C. Audet and J.E. Dennis. Mesh adaptive direct search algorithms for constrained optimization. *SIAM Journal on Optimization*, 17(1):188–217, 2006.

[13] C. Audet and D. Orban. Finding optimal algorithmic parameters using derivative-free optimization. *SIAM Journal on Optimization*, 17(3):642–664, 2006.

[14] L.W. Bright and P. G. Taylor. Calculating the equilibrium distribution in level dependent quasi-birth-and-death processes. *Communications in Statistics: Stochastic Models*, 11(3):497–525, 1995.

[15] L.W. Bright and P.G. Taylor. Equilibrium distributions for level-dependent quasi-birth-and-death processes. In S.R. Chakravarthy and A.S. Alfa, editors, *Matrix Analytic Methods in Stochastic Models: Proceedings of the 1st International Conference*, pages 359–375, New York, 1997. Marcel Dekker.

[16] I.E. Coope and C.J. Price. On the convergence of grid-based methods for unconstrained optimization. *SIAM Journal on Optimization*, 11:859–869, 2001.

[17] G.I. Falin. The $M/M/1$ retrial queue with retrials due to server failures. *Queueing Systems: Theory and Applications*, 58:155–160, 2008.

[18] G.I. Falin. An $M/G/1$ retrial queue with an unreliable server and general repair times. *Performance Evaluation*, 67:569–582, 2010.

[19] G.I. Falin and J.G.C. Templeton. *Retrial Queues*. Chapman & Hall, New York, NY, 1997.

[20] A. Gómez-Corral. A bibliographical guide to the analysis of retrial queues through matrix analytic techniques. *Annals of Operations Research*, 141:163–191, 2006.

[21] J.P. Kharoufeh. Level-dependent quasi-birth-and-death processes. In J. Cochran, A. Cox, P. Keskinocak, J.P. Kharoufeh, and J.C. Smith, editors, *Wiley Encyclopedia of Operations Research and Management Science*, Hoboken, NJ, 2011. John Wiley & Sons, Inc.

[22] C.S. Kim, V. Klimenok, S.C. Lee, and A. Dudin. The $BMAP/PH/1$ retrial queueing system operating in random environment. *Journal of Statistical Planning and Inference*, 137:3904–3916, 2007.

[23] C.S. Kim, V. Klimenok, V. Mushko, and A. Dudin. The $BMAP/PH/N$ retrial queueing system operating in Markovian random environment. *Computers and Operations Research*, 37:1228–1237, 2010.

[24] V. Klimenok. A $BMAP/SM/1$ queueing system with hybrid operation mechanism. *Automation & Remote Control*, 66(5):779–790, May 2005.

[25] V. Klimenok and A. Dudin. Multi-dimensional asymptotically quasi-toeplitz markov chains and their application in queueing theory. *Queueing Systems: Theory and Applications*, 54:245–259, 2006.

[26] V. G. Kulkarni and Bong D. Choi. Retrial queues with server subject to breakdowns and repairs. *Queueing Systems: Theory and Applications*, 7(2):191–208, 1990.

[27] G. Latouche and V. Ramaswami. A logarithmic reduction algorithm for quasi-birth-and-death processes. *Journal of Applied Probability*, 30:650–674, 1993.

[28] G. Latouche and V. Ramaswami. *Introduction to Matrix-Analytic Methods in Stochastic Modeling*. ASA–SIAM Series on Statistics and Applied Probability. American Stat. Assoc. and the Soc. for Indust. and Applied Mathematics, Alexandria, VA and Philadelphia, PA, 1999.

[29] R.M. Lewis and V. Torczon. Pattern search algorithms for bound constrained minimization. *SIAM Journal on Optimization*, 9:1082–1099, 1999.

[30] R.M. Lewis and V. Torczon. Pattern search algorithms for linearly constrained minimization. *SIAM Journal on Optimization*, 10:917–941, 2000.

[31] H. Li and Y.Q. Zhao. A retrial queue with a constant retrial rate, server break downs and impatient customers. *Stochastic Models*, 21(2-3):531–550, 2005.

[32] E. Moutzoukis and C. Langaris. Non-preemptive priorities and vacations in a multiclass retrial queueing system. *Communications in Statistics: Stochastic Models*, 12(3):455–472, 1996.

[33] M.F. Neuts. Further results on the $M/M/1$ queue with randomly varying rates. *OPSEARCH*, 15:158–168, 1978.

[34] M.F. Neuts. The $M/M/1$ queue with randomly varying arrival and service rates. *OPSEARCH*, 15:139–157, 1978.

[35] M.F. Neuts. *Matrix-Geometric Solutions in Stochastic Models: An Algorithmic Approach*. Dover Publications, Inc., New York, NY, 1999.

[36] J. Roszik and J. Sztrik. Performance analysis of finite-source retrial queues operating in random environments. *International Journal of Operational Research*, 2(3):254–268, 2007.

[37] N.P. Sherman and J.P. Kharoufeh. An $M/M/1$ retrial queue with unreliable server. *Operations Research Letters*, 34(6):697–705, 2006.

[38] N.P. Sherman and J.P. Kharoufeh. Optimal Bernoulli routing in an unreliable $M/G/1$ retrial queue. *Probability in the Engineering and Informational Sciences*, 25(1):1–20, 2011.

[39] N.P. Sherman, J.P. Kharoufeh, and M.A. Abramson. An $M/G/1$ retrial queue with unreliable server for streaming multimedia applications. *Probability in the Engineering and Informational Sciences*, 23(2):281–304, 2009.

[40] V. Torczon. On the convergence of pattern search algorithms. *SIAM Journal on Optimization*, 7(1):1–25, 1997.

[41] R.L. Tweedie. Sufficient conditions for regularity, recurrence and ergodicity of Markov processes. In *Math. Proceedings of the Cambridge Philosophical Society*, volume 78, pages 125–136, 1975.

[42] J. Wang, J. Cao, and Q. Li. Reliability analysis of the retrial queue with server breakdowns and repairs. *Queueing Systems: Theory and Applications*, 38(4):363–380, 2001.

[43] J. Wu, Z. Liu, and G. Yang. Analysis of the finite source $MAP/PH/N$ retrial G-queue operating in a random environment. *Applied Mathematical Modeling*, 35:1184–1193, 2011.