QUALITY TECHNOLOGY & QUANTITATIVE MANAGEMENT
2020, VOL. 17, NO. 3, 307–318
https://doi.org/10.1080/16843703.2019.1618535

Taylor & Francis
Taylor & Francis Group

ARTICLE

Optimization analysis of management operation for a server farm

Fu-Min Chang

Department of Finance, Chaoyang University of Technology, Taichung, Taiwan

ABSTRACT

This paper analyses a server farm with a simple management operation, which is desirable to reduce power consumption. A block of servers is named as 'reserves'. Depending on the number of jobs staying in the system, power-up and power-down thresholds are employed to control the state of the reserves. In addition, the process of power-up is not immediately. During the period, the servers cannot serve jobs but still consuming power. The server may be subject to be breakdown. This system was modelled by an infinite capacity queueing system and analysed by matrix-geometric technique. The matrix-geometric method is applied to compute the rate matrix and the stationary probabilities. A cost model is formulated to search the optimum number of permanent server and the optimum power-down threshold. The optimization tasks are carried out by the direct search method. Both analytic processes and numerical results provide very useful and helpful information for decision-makers.

ARTICLE HISTORY
Accepted 8 May 2019

KEYWORDS
Server farm; matrix-geometric; cost model; optimization; unreliable

1. Introduction

Server farm containing a large number of server machines is crucial for data storage and computation in various network applications. Growing demand of cloud computing makes the number of server farms increasing significantly, which results in the amount of power consumption of server farms extremely huge. Schwartz, Pries, and Tran-Gia (2012) indicated that a server farm consumes about 65% of the maximum power consumption even with low load. The efficient way to keep power consumption low is to turn off the unused server. This study considered a simple management operation applying to a server farm, which a block of available servers is designated as 'reserves'. Depending on the number of jobs staying in the system, the state of the reserves is controlled by power-up and power-down thresholds. Noted that the process of power-up is not immediately. During the power-up period, the servers cannot serve jobs but consume power. In particular, we consider the server may be subject to be breakdown. Regarding to such a server farm with simple management policy, we employ an unreliable multi-server queueing system with queue-dependent servers to model such a system. Mangers or decision makers may be interesting on how to deploy the number of permanent servers and the power-down thresholds which minimizes the average cost. To do this, a cost function is formulated to search the optimum number of permanent servers and the optimum power-down threshold which minimizes the average cost.

CONTACT Fu-Min Chang fmchang@cyut.edu.tw Department of Finance, Chaoyang University of Technology, Taichung, Taiwan

Several researchers have been payed attention on the power management of a data farm. Li et al. (2017) devised the energy cost optimization architecture with job security guarantee for Internet server farm. Bai, Chen, Chen, and Wu (2016) presented ECN for multi-service multi-queue data centre network. For the analysis of queueing system where the number of servers change depending on the number of requests in the system, Singh (1973) investigated both $M/M/2/\infty$ and $M/M/3/\infty$ queueing system. Under a cost structure, Garg and Singh (1993) tried to search the optimum queue size to maximize the profit for an infinite-capacity M/M/2 queueing system. They constructed a relationship among the cost elements, and then searched the optimum queue length at which the second server is turned on. The works of Garg and Singh (1993) was extended by Yamashiro (1996). In addition, Wang and Tai (2000) considered a finite capacity M/M/3 queue, where the server is heterogeneous. They derived the stationary features of system such as the expected number of customers and the expected number of waiting customers. Jain (2005) examined a finite capacity $M/M/s$ queue and obtained the optimum threshold parameters for turning on the servers by developing a cost relationship among various cost elements. Efrosinin and Sztrik (2011) dealt with a Markovian queueing system with two heterogeneous servers which operate under a threshold policy. The faster server is activated when it is idle and a customer tries to occupy it. The slower server can be activated only when the number of waiting customers exceeds a threshold level. Ke, Ke, and Lin (2010) and Lin and Ke (2011) performed an optimization analysis on an $M/M/s$ queueing system by using genetic algorithm.

On the other hand, Yamashiro and Yuasa (1996) considered both M/M/2 and M/M/3 machine repair system where the number of repairmen changes depending on the number of failed machines in the system. Lately, Ke, Liu, and Wu (2015) studied a machine repair system with queue-dependent heterogeneous repairman and derived the analytically explicit expressions for the stationary probability of the number of failed machines. They also implemented the direct search method to search the optimum threshold values and adjust the corresponding the service rate. In addition, Lin and Ke (2010) proposed a genetic algorithm to decide the optimum threshold values for an infinite capacity Markovian queueing system with triadic policy. Huang, Hsu, and Ke (2011) used a genetic algorithm to optimize the controlling arrival and service problem for a two-removable-server system. Liou, Wang, and Liou (2013) investigated the controllable M/M/2 machine repair system with finite capacity operating under the triadic policy. They used a genetic algorithm to search the optimum threshold value and corresponding service rate.

Queueing systems wherein the service channel is subject to breakdowns is a popular subject that has received a lot of attention. The readers can refer to the paper of Choudhury and Deka (2008). Choudhury and Deka (2018) dealt with a batch arrival unreliable queue with two phases of service and Bernoulli vacation schedule under multiple vacation policy. Choudhury and Kalita (2018) studied an M/G/1 queue with two types of general heterogeneous service and optional repeated service subject to server breakdowns. In their model, the customer can choose any one type of service. After completion of either type of service, the customer has the further option to repeat the same type of service. Ke, Wu, and Pearn (2017) investigated an infinity capacity M/M/2 queueing system under a dynamic operating policy, where two identical removable servers are assumed to be unreliable. They developed the equilibrium condition of the system and performed an optimization analysis.

This paper contributes on two important issues. Firstly, we model a server farm with simple management policy by using an infinite-capacity multi-server queueing system with queue-dependent heterogeneous servers subject to breakdowns and repairs, where the power-up process is not instantaneous. Secondly, the optimum number of permanent server and the optimum power-down threshold can be obtained by developing a cost function. The optimization task is implemented by the direct search method.

2. Model description

The discussed server farm consists of N servers which including n permanent servers and (N-n) standby servers (reserves). Jobs or requests arrive according to a Poisson process with rate λ. The working server serves one job at a time. Job's service times are distributed exponentially with rate μ. An unbounded first-in first-out (FIFO) queue is employed for those jobs waiting for a server. The server is subject to be breakdowns. Server breaks down according to exponentially distributed with rate α. When a breakdown occurs, the server goes for repair immediately, while the job being served is transferred instantaneously to another server if one is available. If not, the job being served moves to the head of the queue. The repairing time is exponentially distributed with rate β. Once the number of jobs staying in the system reach to the power-up threshold U, the reserves are turned on as a block. After a powering up interval, where the time of powering up is distributed exponentially with rate v, they become operational together. During the period of powering up, they cannot serve jobs but still consume power. On the other hand, to save power and cost, once the number of jobs staying in the system decreases to a given power-down threshold D, the reserves are turned off. Assume the operation of power-down is immediately. Any jobs, whose service is interrupted due to its server is turned off, is instantaneously and immediately moved to another available server. If not, the interrupted job moves to the head of the queue. The interrupted job will resume its service from the point of interruption.

A potential application for the discussed model is the deployment of web application on a cloud. When a web application is deployed on a cloud, a controller, as the portal of cloud, will establish a queue to hold the client requests. For the reason of cost, a control policy is applied. A certain number of Virtual Machines (VMs) will be created by the controller on cloud nodes and be permanently available in the system for performance consideration. The number of initially created VMs can be specified by Service Level Agreement (SLA). When a client sends a request to a web application on cloud, the request will be sent to the controller. The dispatcher in cloud controller forwards the request to the queue of the target web application. The instances of the target web application running into VMs act as service centres to process the requests in the queue. The inter-arrival times between any two successive client requests are independent of each other and have a common probability distribution. The client requests are served in first-come-first-served orders. Besides, a VM is subject to be breakdowns. When a breakdown occurs, the VM will be recreated after a time period, while the job being served is transferred instantaneously to another VM if one is available. If not, the client request being served moves to the head of the queue. As soon as there are U client requests waiting in the system, the rest of VMs will be activated to provide service. But it will be removed from the system if the queue length becomes less than D. An interesting issue raised by this case is to determine the optimal number of permanent VMs and the optimum D at minimum cost.

3. Mathematical model

The states of investigated system is described by (i, j, k), where i represents the current state of the block of reserves, i = 0, 1, 2; j means the number of jobs present, j = 0, 1, 2, …; and k represents the number of failed servers, k = 0,1, 2, …,n for i = 0,1 and k = 0, 1, 2, …, N for i = 2. It is noted that i = 0 means the reserves are turned off; i = 1 and 2 represent the reserves are turned on. Figure 1 illustrates the transition diagram of the discussed system. In steady state, the steady-state probabilities can be presented as $P^i_{j,k}$ and $P^2_{j,k}$, where $P^i_{j,k}$ represents the probability that there are j jobs present, the current state of reserves is i and k failed servers, where j = 0, 1, 2, … and k = 0, 1, 2, …, n for i = 0,1; $P^2_{j,k}$ represents the probability that there are j jobs present, the reserves are turned on and k failed servers, where j = 0, 1, 2, … and k = 0, 1, 2, …, N.

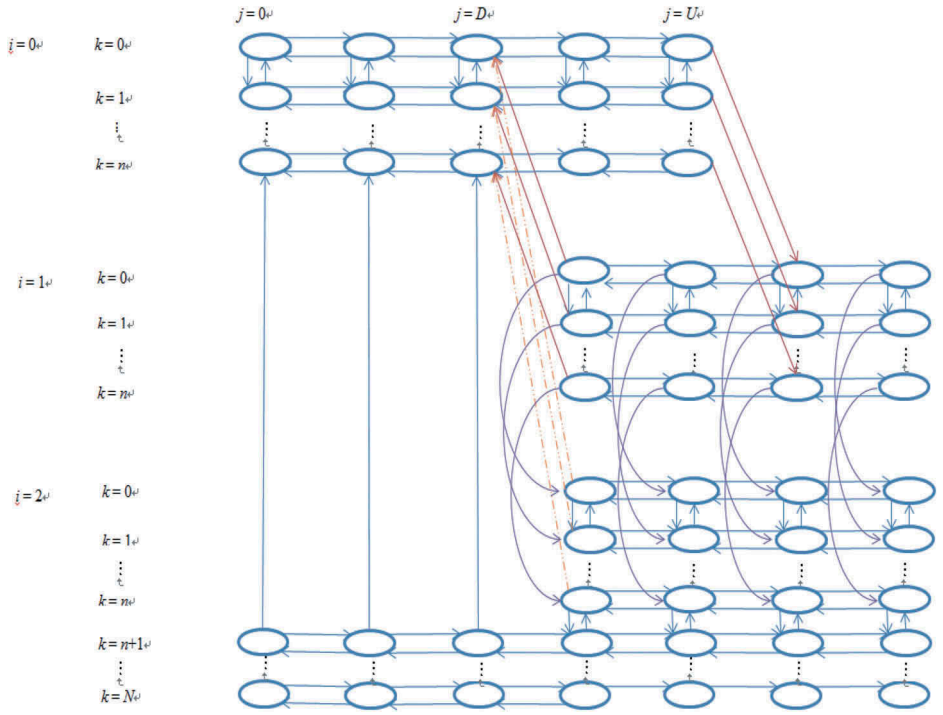

Figure 1. Steady-transition-rate diagram (*i*: the current state of the block of reserves, *j*: the number of jobs present, *k*: the number of failed servers).

Using the lexicographical order for the system states, the infinitesimal generator $\mathbf{Q}$ of the presented queueing system is of the form

$$\mathbf{Q} = \begin{bmatrix} \mathbf{A}_0 & \mathbf{C}_0 & & & & & & & \\ \mathbf{B}_1 & \mathbf{A}_1 & \mathbf{C}_1 & & & & & & \\ & \ddots & \ddots & \ddots & & & & & \\ & & \mathbf{B}_U & \mathbf{A}_U & \mathbf{C}_U & & & & \\ & & & \mathbf{B}_{U+1} & \mathbf{A}_{U+1} & \mathbf{C}_{U+1} & & & \\ & & & & \mathbf{B}_{U+2} & \mathbf{A}_{U+1} & \mathbf{C}_{U+1} & & \\ & & & & & \mathbf{B}_{U+2} & \mathbf{A}_{U+1} & \mathbf{C}_{U+1} & \\ & & & & & & \ddots & \ddots & \ddots \end{bmatrix} \tag{1}$$

The entries $\mathbf{A}_j$, $\mathbf{B}_j$ and $\mathbf{C}_j$ are sub-matrices of $\mathbf{Q}$. To obtain these sub-matrices effectively, we define the following notations.

$$\mathbf{M}_A(n,j,z) = \begin{bmatrix} a_{z,0} & \alpha & & & \\ \beta & a_{z,1} & \alpha & & \\ & \ddots & \ddots & \ddots & \\ & & \beta & a_{z,n-1} & \alpha \\ & & & \beta & a_{z,n} \end{bmatrix}_{(n+1)\times(n+1)} \quad \text{and}$$

$$\mathbf{M}_{A_0}(n,N,j)=\begin{bmatrix} a_{0,0} & \alpha & & & & & & & \\ \beta & a_{0,1} & \alpha & & & & & & \\ & \ddots & \ddots & \ddots & & & & & \\ & & & a_{0,n-1} & \alpha & & & & \\ & & & \beta & a_{0,n} & & & & \\ & & & & \beta & b_{n+1} & \alpha & & \\ & & & & & \beta & b_{n+2} & \alpha & \\ & & & & & & & \ddots & \ddots \\ & & & & & & & \beta & b_N \end{bmatrix}_{(N+1)\times(N+1)} \tag{2}$$

where z=0,1, $a_{z,0}$=-(λ+α+zν+min{n, j}μ), $a_{z,n}$=-(λ+β+zν), b_n=-(λ+β), $a_{z,l}$=-(λ+α+β+zν+min{n-l, j}μ), 1≤l≤n-1, b_l=-(λ+α+β+min{N-l, j}μ), n+1≤l≤N-1.

$$\mathbf{M}_B(n,j)=\begin{bmatrix} \min\{j,n\}\mu & & & \\ & \min\{j,n-1\}\mu & & \\ & & \ddots & \\ & & & 0 \end{bmatrix}_{(n+1)\times(n+1)},$$

$$\mathbf{M}_{B_0}(n,N,j)=\begin{bmatrix} x_0 & & & & & & \\ & x_1 & & & & & \\ & & \ddots & & & & \\ & & & x_n & & & \\ & & & & x_{n+1} & & \\ & & & & & \ddots & \\ & & & & & & x_N \end{bmatrix}_{(N+1)\times(N+1)},$$

$$\mathbf{M}_C(n,m)=\begin{bmatrix} \lambda & & & & & & \\ & \lambda & & & & & \\ & & \ddots & & & & \\ & & & \lambda & 0 & \cdots & 0 \end{bmatrix}_{n\times m} \text{ and}$$

$$\mathbf{M}_d(n,m)=\begin{bmatrix} \nu & & & & & & \\ & \nu & & & & & \\ & & \ddots & & & & \\ & & & \nu & 0 & \cdots & 0 \end{bmatrix}_{(n+1)\times(m+1)}, \tag{3}$$

where

$$x_l=\begin{cases} \min\{n-l,j\}\mu, \text{for} l=0,1,\ldots,n \\ \min\{N-l,j\}\mu, \text{for} l=n+1,n+2,\ldots,N. \end{cases}$$

Finally, $\mathbf{M}_0(n, m)$ denotes the zero-matrix of dimension $(n + 1)\times(m + 1)$. The size for sub-matrices $\mathbf{A}_j$, $\mathbf{B}_j$ and $\mathbf{C}_j$ are tabulated in Table 1.

In Table 1, the notation $(n)^2$ means the matrix is a square matrix of dimension $n \times n$. Based on Table 1, we establish the sub-matrices $\mathbf{A}_j$ as:

$$\mathbf{A}_j=\mathbf{M}_{A_0}(n,N,j), \quad 0\le j\le D$$

Table 1. The sizes for various sub-matrices in the matrix **Q**.

j	A_j	B_j	C_j
$j = 0$	$(N + 1)^2$	-	$(N + 1)^2$
$1 \le j \le D$-1	$(N + 1)^2$	$(N + 1)^2$	$(N + 1)^2$
$j = D$	$(N + 1)^2$	$(N + 1)^2$	$(N + 1)$ x $(2n+N + 3)$
$j = D + 1$	$(2n+N + 3)^2$	$(2n+N + 3)$ x $(N + 1)$	$(2n+N + 3)^2$
$D + 2 \le j \le U$-1	$(2n+N + 3)^2$	$(2n+N + 3)^2$	$(2n+N + 3)^2$
$j = U$	$(2n+N + 3)^2$	$(2n+N + 3)^2$	$(2n+N + 3)$ x $(n + N + 2)$
$j = U + 1$	$(n + N + 2)^2$	$(n + N + 2)$ x $(2n+N + 3)$	$(n + N + 2)^2$
$U + 2 \le j$	$(n + N + 2)^2$	$(n + N + 2)^2$	$(n + N + 2)^2$

$$\mathbf{A}_j = \begin{bmatrix} \mathbf{M}_A(n,j,0) & \mathbf{M}_0(n,n) & \mathbf{M}_0(n,N) \\ \mathbf{M}_0(n,n) & \mathbf{M}_A(n,j,1) & \mathbf{M}_d(n,N) \\ \mathbf{M}_0(N,n) & \mathbf{M}_0(N,n) & \mathbf{M}_A(N,j,0) \end{bmatrix}, \quad D+1 \le j \le U$$

$$\mathbf{A}_j = \begin{bmatrix} \mathbf{M}_A(n,j,1) & \mathbf{M}_d(n,N) \\ \mathbf{M}_0(N,n) & \mathbf{M}_A(N,j,0) \end{bmatrix}, \quad U+1 \le j$$

and $\mathbf{B}_j$ as:

$$\mathbf{B}_j = \mathbf{M}_{B_0}(n,N,j), \quad 1 \le j \le D$$

$$\mathbf{B}_{D+1} = \begin{bmatrix} \mathbf{M}_B(n,D+1) & \mathbf{M}_0(n,N-n-1) \\ \mathbf{M}_B(n,D+1) & \mathbf{M}_0(n,N-n-1) \\ & \mathbf{M}_B(N,D+1) \end{bmatrix}$$

$$\mathbf{B}_j = \begin{bmatrix} \mathbf{M}_B(n,j) & \mathbf{M}_0(n,n) & \mathbf{M}_0(n,N) \\ \mathbf{M}_0(n,n) & \mathbf{M}_B(n,j) & \mathbf{M}_0(n,N) \\ \mathbf{M}_0(N,n) & \mathbf{M}_0(N,n) & \mathbf{M}_B(N,j) \end{bmatrix}, \quad D+2 \le j \le U$$

$$\mathbf{B}_{U+1} = \begin{bmatrix} \mathbf{M}_0(n,n) & \mathbf{M}_B(n,U+1) & \mathbf{M}_0(n,N) \\ \mathbf{M}_0(N,n) & \mathbf{M}_0(N,n) & \mathbf{M}_B(N,U+1) \end{bmatrix}$$

$$\mathbf{B}_j = \begin{bmatrix} \mathbf{M}_B(n,j) & \mathbf{M}_0(n,N) \\ \mathbf{M}_0(N,n) & \mathbf{M}_B(N,j) \end{bmatrix}, \quad U+2 \le j$$

and $\mathbf{C}_j$ as:

$$\mathbf{C}_j = \mathbf{M}_C(N+1,N+1), \quad 0 \le j \le D-1$$

$$\mathbf{C}_D = \begin{bmatrix} \mathbf{M}_C(n+1,3n+3) & \mathbf{M}_0(n,N-n-1) \\ \mathbf{M}_0(N-n-1,3n+2) & \mathbf{M}_C(N-n,N-n) \end{bmatrix},$$

$$\mathbf{C}_j = \mathbf{M}_C(2n+N+3,2n+N+3), \quad D+1 \le j \le U-1$$

$$\mathbf{C}_U = \begin{bmatrix} \mathbf{M}_C(n+1,n+N+2) \\ \mathbf{M}_C(n+N+2,n+N+2) \end{bmatrix},$$

$$\mathbf{C}_j = \mathbf{M}_C(n+N+2,n+N+2), \quad U+1 \le j$$

The steady-state equations are given by $\mathbf{PQ} = \mathbf{0}$ in which $\mathbf{P}$ denotes the steady-state probability vector and $\mathbf{0}$ is the zero column vector. We partition the vector $\mathbf{P}$ as $\mathbf{P} = [\mathbf{P}_0, \mathbf{P}_1, \mathbf{P}_2, \ldots]$ where

$$\mathbf{P}_j = \left[P^0_{j,0}, P^0_{j,1}, \ldots, P^0_{j,n}, P^2_{j,n+1}, \ldots, P^2_{j,N}\right], 0 \le j \le D$$

$$\mathbf{P}_j = \left[P^0_{j,0}, P^0_{j,1}, \ldots, P^0_{j,n}, P^1_{j,0}, P^1_{j,1}, \ldots, P^1_{j,n}, P^2_{j,0}, P^2_{j,1}, \ldots, P^2_{j,N}\right], \quad D+1 \le j \le U$$

$$\mathbf{P}_j = \left[P^1_{j,0}, P^1_{j,1}, \ldots, P^1_{j,n}, P^2_{j,0}, P^2_{j,1}, \ldots, P^2_{j,N}\right], U+1 \le j.$$

The stability condition should be developed to ensure that the system is stable. Let $\mathbf{F}$ be equal to $\mathbf{C}_{U+1} + \mathbf{A}_{U+1} + \mathbf{B}_{U+2}$. It is observed that the matrix $\mathbf{F}$ is the infinitesimal generator. Referring to Theorem 3.1.1 of Neuts (1981), we know the steady-state probability vector exists if and only if $\mathbf{xB}_{U+2}\mathbf{e} > \mathbf{xC}_{U+1}\mathbf{e}$. Assume that $\mathbf{x} = \left[x^1_0, x^1_1, \ldots, x^1_n, x^2_0, x^2_1, \ldots, x^2_N\right]$ is a row vector of the steady-state probability $\mathbf{F}$. Thus, $\mathbf{x}$ satisfies the linear equations $\mathbf{xF} = \mathbf{0}$ and $\mathbf{xe} = 1$. After some routine manipulations, the stability condition for discussed model can be expressed as

$$\lambda < \sum_{k=0}^{n} \min\{U+2, n-k\}\mu x^1_k + \sum_{k=0}^{N} \min\{U+2, N-k\}\mu x^2_k.$$

4. Computing rate matrix and stationary probabilities

The stationary probability vector $\mathbf{P}$ of $\mathbf{Q}$ exists under the stability condition. Expanding the equations $\mathbf{PQ} = \mathbf{0}$ yields

$$\mathbf{P}_0\mathbf{A}_0 + \mathbf{P}_1\mathbf{B}_1 = \mathbf{0} \tag{4a}$$

$$\mathbf{P}_{j-1}\mathbf{C}_{j-1} + \mathbf{P}_j\mathbf{A}_j + \mathbf{P}_{j+1}\mathbf{B}_{j+1} = \mathbf{0}, \quad 1 \le j \le U+1 \tag{4b}$$

$$\mathbf{P}_{U+1}\mathbf{C}_{U+1} + \mathbf{P}_{U+2}\mathbf{A}_{U+1} + \mathbf{P}_{U+2}\mathbf{R}\mathbf{B}_{U+2} = \mathbf{0}, \tag{4c}$$

$$\mathbf{P}_{U+2}\mathbf{R}^{j-U-3}\mathbf{C}_{U+1} + \mathbf{P}_{U+2}\mathbf{R}^{j-U-2}\mathbf{A}_{U+1} + \mathbf{P}_{U+2}\mathbf{R}^{j-U-1}\mathbf{B}_{U+2} = \mathbf{0}, \quad U+3 \le j \tag{4d}$$

where $\mathbf{R}$ is the minimal non-negative solution of the matrix quadratic equation given below:

$$\mathbf{C}_{U+1} + \mathbf{R}\mathbf{A}_{U+1} + \mathbf{R}^2\mathbf{B}_{U+2} = \mathbf{0} \tag{5}$$

Since the steady-state probabilities $[\mathbf{P}_{U+2}, \mathbf{P}_{U+3}, \mathbf{P}_{U+4}, \ldots]$ have the following property:$\mathbf{P}_{U+2+j} = \mathbf{P}_{U+2}\mathbf{R}^j$ for $j \ge 1$, $\mathbf{P}_j (j = U+3, U+4, \ldots)$ can be determined recursively. It is known (Neuts (1981)) that $\mathbf{R}$ is given by $\lim_{n\to\infty} \mathbf{R}_n$, where $\{\mathbf{R}_n\}$ is defined by

$$\mathbf{R}_0 = \mathbf{0} \text{ and } \mathbf{R}_{n+1} = -\mathbf{C}_{U+1}\mathbf{A}^{-1}_{U+1} - \mathbf{R}^2_n\mathbf{B}_{U+2}\mathbf{A}^{-1}_{U+1} \text{ for } n \ge 0. \tag{6}$$

Because $\{\mathbf{R}_n\}$ is certified that it converges to rate matrix $\mathbf{R}$ monotonically, the rate matrix $\mathbf{R}$ could be evaluated from above sequences by successive substitutions.

Next, we compute the stationary probabilities. Combining equations (4a)-(4d) recursively, we obtain

$$\mathbf{P}_0 = \mathbf{P}_1\mathbf{B}_1(\mathbf{A}_0)^{-1} = \mathbf{P}_1\varphi_0 \tag{7}$$

$$\mathbf{P}_j = \mathbf{P}_{j+1}\mathbf{B}_{j+1}\left[-\left(\varphi_{j-1}\mathbf{C}_{j-1} + \mathbf{A}_j\right)\right]^{-1} = \mathbf{P}_{j+1}\varphi_j, \quad 1 \le j \le U+1 \tag{8}$$

$$\mathbf{P}_{U+2}\varphi_{U+1}\mathbf{C}_{U+1} + \mathbf{P}_{U+2}\mathbf{A}_{U+1} + \mathbf{P}_{U+2}\mathbf{R}\mathbf{B}_{U+2} = \mathbf{0} \tag{9}$$

Consequently, the steady-state probabilities $\mathbf{P}_j (0 \le j \le U+1)$ in equations (7) and (8) can be written in terms of $\mathbf{P}_{U+2}$ as $\mathbf{P}_{U+2}\prod_{i=U+1}^{j}\varphi_i$ and the remaining portion of steady-state probabilities $[\mathbf{P}_{U+2}, \mathbf{P}_{U+3}, \mathbf{P}_{U+4}, \ldots]$ can be determined using $\mathbf{P}_{U+2+j} = \mathbf{P}_{U+2}\mathbf{R}^j$ for $j \ge 1$. Once $\mathbf{P}_{U+2}$ is gained, the steady-state solutions $[\mathbf{P}_0, \mathbf{P}_1, \mathbf{P}_2, \ldots]$ can be derived. It is noted that $\mathbf{P}_{U+2}$ can be solved by using Equation (9) and the following normalization condition:

$$\begin{aligned}\sum_{j=0}^{\infty}\mathbf{P}_j\mathbf{e} &= \left\{\mathbf{P}_{U+2}\prod_{k=U+1}^{0}\varphi_k + \mathbf{P}_{U+2}\prod_{k=U+1}^{1}\varphi_k + \cdots + \mathbf{P}_{U+2}\varphi_{U+1} + \mathbf{P}_{U+2} + \mathbf{P}_{U+2}\mathbf{R} + \mathbf{P}_{U+2}\mathbf{R}^2 + \cdots\right\}\mathbf{e} \\ &= \mathbf{P}_{U+2}\left[\sum_{j=1}^{U+1}\prod_{k=U+1}^{j}\varphi_k + (\mathbf{I}-\mathbf{R})^{-1}\right]\mathbf{e} = 1\end{aligned} \tag{10}$$

The algorithm for computing the steady-state probability is presented as follow:

Algorithm: Recursive solver

Step 1: Compute $\varphi_0 = \mathbf{B}_1(\mathbf{A}_0)^{-1}$ and $\varphi_j = \mathbf{B}_{j+1}\left[-\left(\varphi_{j-1}\mathbf{C}_{j-1} + \mathbf{A}_j\right)\right]^{-1}$, $1 \le j \le U+1$.

Step 2: Compute $\Phi_j = \prod_{k=U+1}^{j}\varphi_k$, $1 \le j \le U+1$.

Step 3: Solve $\mathbf{P}_{U+2}\varphi_{U+1}\mathbf{C}_{U+1} + \mathbf{P}_{U+2}\mathbf{A}_{U+1} + \mathbf{P}_{U+2}\mathbf{R}\mathbf{B}_{U+2} = \mathbf{0}$ and

$\mathbf{P}_{U+2}\left[\sum_{j=1}^{U+1}\prod_{k=U+1}^{j}\varphi_k + (\mathbf{I}-\mathbf{R})^{-1}\right]\mathbf{e} = 1$ and obtain steady-state probability $\mathbf{P}_{U+2}$.

Step 4: Obtain steady-state probability $\mathbf{P}$ by $\mathbf{P}_j = \mathbf{P}_{U+2}\Phi_j$ if $0 \le j \le U+1$ and $\mathbf{P}_j = \mathbf{P}_{j-1}\mathbf{R}$ if $j \ge U+3$.

5. Optimization analysis

We denoted the mean number of servers which are consuming power and the mean number of breakdown servers by S and S_1, respectively. The explicit expressions of S and S_1 are presented below:

$$\begin{aligned}S &= \sum_{j=0}^{U}\sum_{k=0}^{n}(n-k)P_{j,k}^0 + \sum_{j=D+1}^{\infty}\sum_{k=0}^{n}(N-k)P_{j,k}^1 + \sum_{j=D+1}^{\infty}\sum_{k=0}^{N}(N-k)P_{j,k}^2 \\ &\quad + \sum_{j=0}^{D}\sum_{k=n+1}^{N}(N-k)P_{j,k}^2 \\ &= \sum_{j=0}^{D}\mathbf{P}_j\begin{bmatrix}\mathbf{v}_{n,0}\\ \mathbf{v}_{N-n-1,0}\end{bmatrix} + \sum_{j=D+1}^{U}\mathbf{P}_j\begin{bmatrix}\mathbf{v}_{n,0}\\ \mathbf{v}_{N,N-n}\\ \mathbf{v}_{N,0}\end{bmatrix} + \mathbf{P}_{U+2}\left[\varphi_{U+2} + (\mathbf{I}-\mathbf{R})^{-1}\right]\begin{bmatrix}\mathbf{v}_{N,N-n}\\ \mathbf{v}_{N,0}\end{bmatrix}\end{aligned}$$

$$\begin{aligned}S_1 &= \sum_{j=0}^{U}\sum_{k=0}^{n}kP_{j,k}^0 + \sum_{j=D+1}^{\infty}\sum_{k=0}^{n}kP_{j,k}^1 + \sum_{j=D+1}^{\infty}\sum_{k=0}^{N}kP_{j,k}^2 + \sum_{j=0}^{D}\sum_{k=n+1}^{N}kP_{j,k}^2 \\ &= \sum_{j=0}^{D}\mathbf{P}_j\begin{bmatrix}\mathbf{u}_{0,n}\\ \mathbf{u}_{n+1,N}\end{bmatrix} + \sum_{j=D+1}^{U}\mathbf{P}_j\begin{bmatrix}\mathbf{u}_{0,n}\\ \mathbf{u}_{0,n}\\ \mathbf{u}_{0,N}\end{bmatrix} + \mathbf{P}_{U+2}\left[\varphi_{U+2} + (\mathbf{I}-\mathbf{R})^{-1}\right]\begin{bmatrix}\mathbf{u}_{0,n}\\ \mathbf{u}_{0,N}\end{bmatrix}\end{aligned}$$

where $\mathbf{v}_{l,m}$ and $\mathbf{u}_{l,m}$ denote the vector $[l, l\text{-}1, \ldots, m]^{\mathrm{T}}$ and $[l, l+1, \ldots, m]^{\mathrm{T}}$, respectively.

A cost function is formulated to make the investigated system viable economically. We assume that each available server incurs a unit cost c_1; each breakdown server incurs a unit cost c_2; d_1 means the unit cost when the block of reserves is turned on; and each power-down of the reserves incurs a unit cost d_2. Thus the average cost of the system per unit time has the form

$$C = c_1 S + c_2 S_1 + d_1 \lambda \sum_{k=0}^{n} P_{U,k}^{0} + d_2 \mu \sum_{k=0}^{n} \min\{D+1, n-k\} P_{D+1,k}^{1}$$
$$+ d_2 \mu \sum_{k=0}^{n} \min\{D+1, N-k\} P_{D+1,k}^{2} + d_2 \beta \sum_{j=0}^{D} P_{j,n+1}^{2}$$

We investigated the parameter effects on cost function and searched the optimum number of permanent server and the optimum power-down threshold which minimizes the average cost. In following cases, we set N (the number of servers) to 20 and the mean service rate μ was set to 1. The mean power-up interval v was set to 1. It is assumed that the server breakdown rate and the repairing rate were chosen as $\alpha = 3$ and $\beta = 6$, respectively. We also vary the values of arrive rate λ from 6 to 10 with increments of 2. The following cost elements are adopted: $c_1 = 30$, $c_2 = 90$, $d_1 = 150$, $d_2 = 100$.

Figure 2 shows that the relevance of expected cost C and the number of permanent servers n for different arrival rate λ. The power-up threshold U was fixed at 20 and the power-down threshold D was selected optimally in each case, by calculating the average cost for all values $0 \le D \le U$. From Figure 2 one can find that the optimum value of n exists for each value of λ and the optimum value of n increases as arrival rate λ increases.

The plot in Figure 3 presents the relevance of expected cost C and the power-up threshold U for different arrival rate λ. The value of n is set to 6. The power-down threshold is again selected optimally. The other parameters are set as the same as before. It can be observed that the power-up threshold U should be postponed indefinitely. Based on this, the upper threshold U is set to N.

Next, we searched the optimum number of permanent server and the optimum power-down threshold which minimize the expected cost when the total number of servers, the power-up threshold and other system parameters are given. A sensitivity study are performed to search the optimum values based on changes in specific values of the system parameters. For various values of λ, α, μ and v, the numerical results are shown in Tables 2 and Table 3. From these tables, one

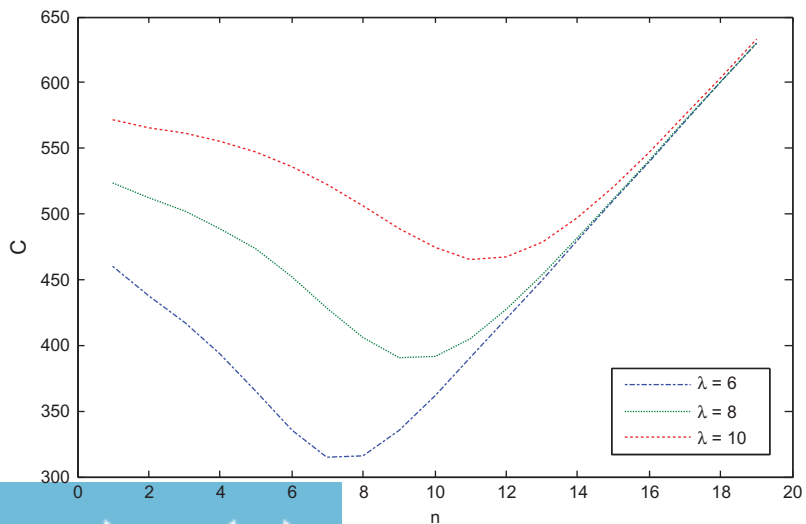

Figure 2. C versus the number of permanent server n for different arrival rate λ.

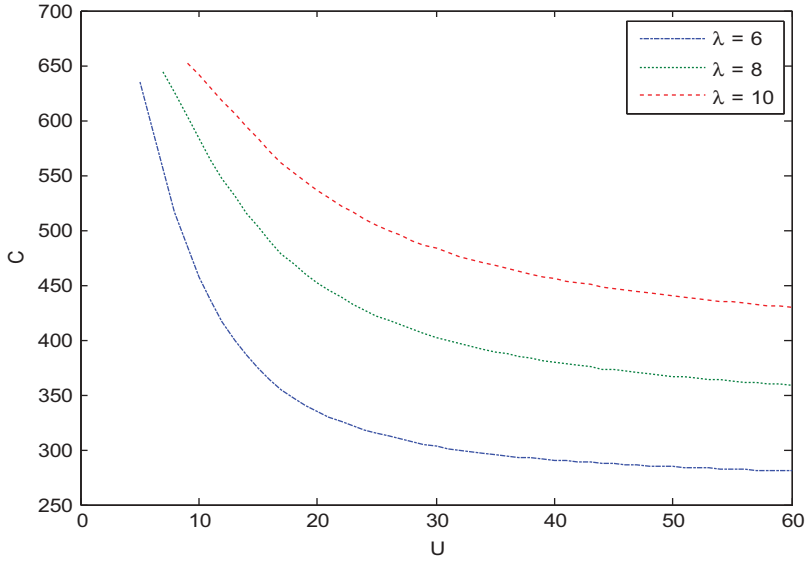

Figure 3. C versus the power-up threshold U for different arrival rate λ.

Table 2. (n^*, D^*) and corresponding minimum C for values of λ and α. ($\mu = 1$, $\nu = 1$, $\beta = 6$).

(λ,α)	(4,1)	(6,1)	(8,1)	(4,5)	(6,5)	(8,5)
(n^*, D^*)	(5,13)	(7,13)	(9,13)	(7,15)	(9,16)	(9,11)
C	167.8264	239.5941	315.0498	411.6985	520.9401	648.1969
S	4.943	7.2166	9.5581	6.5934	7.9735	9.7866
S_1	0.1997	0.2000	0.2000	2.2128	2.8939	3.6270
(λ,α)	(4,1)	(4,3)	(4,5)	(8,1)	(8,3)	(8,5)
(n^*, D^*)	(5,13)	(5,9)	(7,15)	(9,13)	(9,12)	(9,11)
C	167.8264	242.2139	411.6985	315.0498	390.9623	648.1969
S	4.9430	5.2317	6.5934	9.5581	9.5072	9.7866
S_1	0.1997	0.8716	2.2128	0.2000	0.9895	3.6270

Table 3. (n^*, D^*) and corresponding minimum C for values of μ and ν. ($\lambda = 8$, $\alpha = 3$, $\beta = 6$).

(μ, ν)	(1,0.5)	(1.5,0.5)	(2,0.5)	(1,1.5)	(1.5,1.5)	(2,1.5)
(n^*, D^*)	(10,14)	(7,12)	(6,15)	(9,12)	(7,12)	(5,9)
C	397.3926	303.1277	255.0773	385.5293	291.6925	247.3172
S	9.9164	6.9833	5.6293	9.3132	6.5903	5.1799
S_1	0.9924	0.9519	0.8995	0.9897	0.9518	0.8798
(μ, ν)	(1,0.5)	(1,1)	(1,1.5)	(2,0.5)	(2,1)	(2,1.5)
(n^*, D^*)	(10,14)	(9,12)	(9,12)	(6,15)	(6,14)	(5,9)
C	397.3926	390.9623	385.5293	255.0773	250.5332	247.3172
S	9.9164	9.5072	9.3132	5.6293	5.4809	5.1799
S_1	0.9924	0.9895	0.9897	0.8995	0.9012	0.8798

can find that the number of permanent servers n increases as λ increases or μ decreases. For fixing $\mu = 1$, $\nu = 1$ and $\beta = 6$, Table 2 reveals that both the mean number of servers which are consuming power S and the mean number of breakdown servers S_1 increase as λ increases or μ increases. As shown in Table 3, for fixing $\lambda = 8$, $\alpha = 3$ and $\beta = 6$, both S and S_1 decrease as μ increases or ν increases.

6. Conclusions

By using an infinite-capacity multi-server queueing system with queue-dependent heterogeneous servers subject to breakdowns and repairs, this paper modelled a simple mechanism applying to a server farm in which a block of available servers is reserving and powering them up and down depending on the number of jobs staying in the system. This system analysed by matrix-geometric property. We also derived the formulae for computing the rate matrix and stationary probabilities. A cost function was formulated to decide the optimum number of permanent server and the optimum power-down threshold. Some numerical cases were also performed to illustrate how to choose the control parameters. Both analytical processes and results provide very useful and helpful information for decision-makers.

Acknowledgements

The authors gratefully acknowledge the constructive comments of editors and the anonymous reviewers. This research was partially supported by the Ministry of Science and Technology of Taiwan under grants MOST 106-2221-E-324-016-.

Disclosure statement

No potential conflict of interest was reported by the author.

Notes on contributor

Fu-Min Chang is an assistant professor in the Department of Finance, ChaoYang University of Technology, Taiwan.He received his Master and PhD degrees from National Chung-Hsing University, Taiwan, ROC, in 1992 and 2005,respectively. His current research interests include the area of queueing network, network performance evaluation, andsystem and network management.

References

Bai, W., Chen, L., Chen, K., & Wu, H. (2016). Enabling ECN in multi-service multi-queue data centers. *13th USENIX Symposium on Networked Systems Design and Implementation*, Santa Clara, CA, USA, 537–549.

Choudhury, G., & Deka, K. (2008). An M/G/1 retrial queueing system with two phases of service subject to the server breakdown and repair. *Performance Evaluation*, *65*(10), 714–724.

Choudhury, G., & Deka, M. (2018). A batch arrival unreliable server delaying repair queue with two phases of service and Bernoulli vacation under multiple vacation policy. *Quality Technology & Quantitative Management*, *15*(2), 157–186.

Choudhury, G., & Kalita, C. R. (2018). An M/G/1 queue with two types of general heterogeneous service and optional repeated service subject to server's breakdown and delayed repair. *Quality Technology & Quantitative Management*, *15*(5), 622–654.

Efrosinin, D., & Sztrik, J. (2011). Performance analysis of a two-server heterogeneous retrial queue with threshold policy. *Quality Technology & Quantitative Management*, *8*(3), 211–236.

Garg, R. L., & Singh, P. (1993). Queue-dependent servers queueing system. *Microelectronic Reliability*, *33*(15), 2289–2295.

Huang, H. I., Hsu, P. C., & Ke, J. C. (2011). Controlling arrival and service of a two-removable-server system using genetic algorithm. *Expert System with Applications*, *38*(8), 10054–10059.

Jain, M. (2005). Finite capacity M/M/r queueing system with queue-dependent servers. *Computer and Mathematics with Applications*, *50*(1), 187–199.

Ke, J. B., Ke, J. C., & Lin, C. H. (2010). Cost optimization of an M/M/r queueing system with queue-dependent servers: Genetic algorithm. Proceedings, *the 5th International Conference on Queueing Theory and Network Applications*, Beijing, China, July 24–26.

Ke, J. C., Liu, T. H., & Wu, C. H. (2015). An optimum approach of profit analysis on the machine repair system with heterogeneous repairmen. *Applied Mathematics and Computation*, *253*, 40–51.

Ke, J. C., Wu, C. H., & Pearn, W. L. (2017). Dynamic operating policy for the controllable queue with two removable unreliable servers. *International Journal of Computer Mathematics: Computer Systems Theory*, *2*(3), 81–96.
Li, Z., Ge, J., Li, C., Yang, H., Hu, H., Luo, B., & Chang, V. (2017). Energy cost minimization with job security guarantee in Internet data center. *Future Generation Computer Systems*, *73*, 63–78.
Lin, C. H., & Ke, J. C. (2010). Genetic algorithm for optimal thresholds of an infinite capacity multi-server system with triadic policy. *Expert System with Application*, *37*(6), 4276–4282.
Lin, C. H., & Ke, J. C. (2011). Optimization analysis for an infinite capacity queueing system with multiple queue-dependent servers: Genetic algorithm. *International Journal of Computer Mathematics*, *88*(7), 1430–1442.
Liou, C. D., Wang, K. H., & Liou, M. W. (2013). Genetic algorithm to the machine repair problem with two removable servers operating under the triadic (0, Q, N, M) policy. *Applied Mathematical Modelling*, *37*(18), 8419–8430.
Neuts, M. F. (1981). *Matrix geometric solutions in stochastic models: An algorithmic approach*. Baltimore, USA: The John Hopkins University Press.
Schwartz, C., Pries, R., & Tran-Gia, P. (2012). A queuing analysis of an energy-saving mechanism in data centers. *International Conference on Information Networking*, Bali, Indonesia, 70–75.
Singh, V. P. (1973). Queue-dependent servers. *Journal of Engineering Mathematics*, *7*(2), 123–126.
Wang, K. H., & Tai, K. H. (2000). A queueing system with queue-dependent servers and finite capacity. *Applied Mathematical Modelling*, *24*(11), 807–814.
Yamashiro, M. (1996). A system where the number of servers changes depending on the queue length. *Microelectronic Reliability*, *36*(3), 389–391.
Yamashiro, M., & Yuasa, Y. (1996). Repair system where the repairmen changes depending on the failed machines. *Microelectronic Reliability*, *36*(2), 231–234.