Multiserver Queues with Finite Capacity and Setup Time

Tuan Phung-Duc(✉)

Department of Mathematical and Computing Sciences,
Tokyo Institute of Technology, Ookayama, Meguro-ku, Tokyo, Japan
tuan@is.titech.ac.jp

Abstract. Multiserver queues with setup time have been extensively studied because they have application in modelling of power-saving data centers. Although the infinite buffer models are extensively investigated, less attention has been paid to finite buffer models. This paper considers an M/M/c/K queue with setup time for which we suggest a simple and numerically stable recursion for the stationary distribution of the system state. Numerical experiments show various insights into the performance of the system such as performance-energy tradeoff as well as the effect of the capacity on the blocking probability and the mean queue length.

Keywords: Multiserver queue · Setup time · Finite capacity

1 Introduction

The core part of cloud computing is data center where a huge number of servers are available. These servers consume a large amount of energy. Thus, the key issue for the management of these server farms is to minimize the power consumption while keeping acceptable service level for users. It is reported that under the current technology an idle server still consumes about 60% of its peak processing jobs [1]. Thus, the only way to save power is to turn off idle servers. However, off servers need some setup time to be active during which they consume energy but cannot process jobs. Thus, there exists a trade-off between power-saving and performance which could be analyzed by queueing models with setup time.

Recently, motivated by applications in data centers, multiserver queues with setup times have been extensively investigated in the literature. In particular, Gandhi et al. [3] extensively analyze multiserver queues with setup times. They obtain some closed form approximations for the ON-OFF policy where any number of servers can be in the setup mode at a time. As is pointed out in Gandhi et al. [3], from an analytical point of view the most challenging model is the ON-OFF policy where the number of servers in setup mode is not limited. Gandhi et al. [4,5] analyze the M/M/c/Setup model with ON-OFF policy using a recursive renewal reward approach. Phung-Duc [11] obtains exact solutions for the

© Springer International Publishing Switzerland 2015
M. Gribaudo et al. (Eds.): ASMTA 2015, LNCS 9081, pp. 173–187, 2015.
DOI: 10.1007/978-3-319-18579-8_13

same model via generating functions and via matrix analytic methods. Slegers et al. [6] propose a heuristic method to decide the timing for the servers to be powered up or down.

Although, the infinite model has been investigated [4,5,11], results for systems with a large number (several hundreds) of servers are not obtained. This motivates us to develop models for large-scale server farms. Furthermore, less attention has been paid on finite buffer multiserver queue with setup time. It should be noted that the results for the latter could be used for the former by letting the capacity tend to infinity. The main aim of our current paper is to present a simple recursion for the stationary distribution of the M/M/c/K/Setup model which is more realistic for data centers which typically have a finite buffer. The computational complexity of the scheme is significantly reduced in comparison with that of direct methods. As a result, models with several hundreds of servers are easily analyzed. This allows us to explore new insight into the performance of large scale systems that has not been observed in literature. Recently, we become aware of a closely related paper [2], where the authors suggest a recursive scheme for finite buffer model with threshold control. However, the stability of the numerical scheme is not discussed. In contrast to [2], we suggest here a new recursive scheme whose numerical stability is rigorously proved.

The rest of this paper is organized as follows. Section 2 presents the model in details while Section 3 is devoted to derivation of a recursion for the joint stationary distribution. Section 4 presents some numerical examples showing insights into the performance of the system. Concluding remarks are presented in Section 5.

2 Model

We consider a queueing system with c servers and a capacity of K, i.e., the maximum of K customers can be accommodated in the system. Jobs arrive at the system according to a Poisson process with rate λ. In this system, a server is turned off immediately if it has no job to do. Upon arrival of a job, an OFF server is turned on if any and the job is placed in the buffer. However, a server needs some setup time to be active so as to serve waiting jobs. We assume that the setup time follows an exponential distribution with mean $1/\alpha$. Let j denotes the number of customers in the system and i denotes the number of active servers. The number of servers in setup process is $\min(j-i, c-i)$. Under these assumptions, the number of active servers is smaller than or equal to the number of jobs in the system. Therefore, in this model a server is in either BUSY or OFF or SETUP. We assume that the service time of jobs follows an exponential distribution with mean $1/\mu$. We assume that waiting jobs are served according to a first-come-first-served (FCFS) manner. We call this model an M/M/c/K/Setup queue.

The exponential assumptions for the inter-arrival, setup time and service time allow to construct a Markov chain whose stationary distribution is recursively obtainable. It should be noted that we can easily construct a Markov chain for a more general model with MAP arrival and phase-type service and setup time distributions. However, the state space of the resulted Markov chain explodes and thus the analysis is complex.

3 Analysis

In this section, we present a recursive scheme to calculate the joint stationary distribution. Let $C(t)$ and $N(t)$ denote the number of active servers and the number of customers in the system, respectively. It is easy to see that $\{X(t) = (C(t), N(t)); t \geq 0\}$ forms a Markov chain on the state space:

$$\mathcal{S} = \{(i, j); 0 \leq i \leq c, j = i, i+1, \ldots, K-1, K\}.$$

See Figure 1 for transition among states for the case $c = 2$ and $K = 5$.

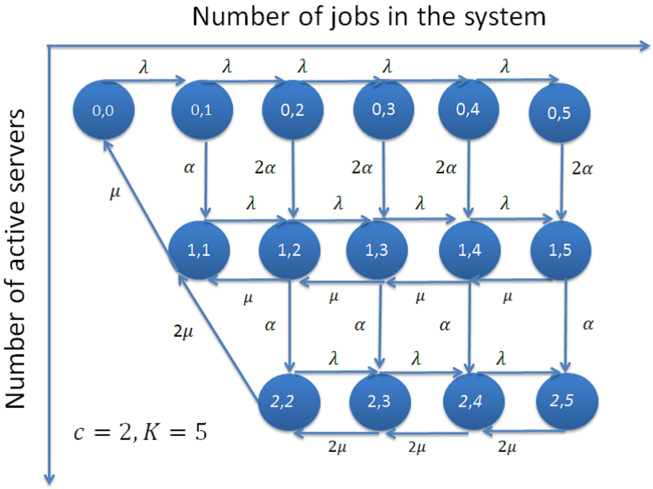

Fig. 1. Transition among states ($c = 2, K = 5$)

Let $\pi_{i,j} = \lim_{t\to\infty} \mathrm{P}(C(t) = i, N(t) = j)$ $((i, j) \in \mathcal{S})$ denote the joint stationary distribution of $\{X(t)\}$. In this section, we derive a recursion for calculating the joint stationary distribution $\pi_{i,j}$ $((i, j) \in \mathcal{S})$. The balance equations for states with $i = 0$ read as follows.

$$\begin{aligned} \lambda\pi_{0,0} &= \mu\pi_{1,1}, \\ (\lambda + \min(j, c)\alpha)\pi_{0,j} &= \lambda\pi_{0,j-1}, \qquad j = 1, 2, \ldots, K-1, \\ c\alpha\pi_{0,K} &= \lambda\pi_{0,K-1}. \end{aligned}$$

leading to $\pi_{0,j} = b_j^{(0)}\pi_{0,j-1}$ where $b_j^{(0)} = \lambda/(\lambda + \min(j, c)\alpha)$ $(j = 1, 2, \ldots, K-1)$ and $b_K^{(0)} = \lambda/(c\alpha)$. Furthermore, it should be noted that $\pi_{1,1}$ is calculated using the local balance equation in and out the set $\{(0, j); j = 0, 1, \ldots, K\}$ as follows.

$$\mu\pi_{1,1} = \sum_{j=1}^{K} \min(j,c)\alpha\pi_{0,j}.$$

Remark 1. We have expressed $\pi_{0,j}$ $(j = 1, 2, \ldots, K)$ and $\pi_{1,1}$ in terms of $\pi_{0,0}$.

Next, we consider the case $i = 1$.

Lemma 1. *We have*

$$\pi_{1,j} = a_j^{(1)} + b_j^{(1)}\pi_{1,j-1}, \qquad j = 2, 3, \ldots, K-1, K,$$

where

$$a_j^{(1)} = \frac{\mu a_{j+1}^{(1)} + \min(j,c)\alpha\pi_{0,j}}{\mu + \lambda + \min(j-1,c-1)\alpha - \mu b_{j+1}^{(1)}}, \tag{1}$$

$$b_j^{(1)} = \frac{\lambda}{\mu + \lambda + \min(j-1,c-1)\alpha - \mu b_{j+1}^{(1)}}, \tag{2}$$

for $j = K-1, K-2, \ldots, 2$ *and*

$$a_K^{(1)} = \frac{c\alpha\pi_{0,K}}{\mu + (c-1)\alpha}, \qquad b_K^{(1)} = \frac{\lambda}{\mu + (c-1)\alpha}.$$

Proof. We prove using mathematical induction. Balance equations are given as follows.

$$(\lambda + \mu + \min(j-1,c-1)\alpha)\pi_{1,j} = \lambda\pi_{1,j-1} + \mu\pi_{1,j+1} + \min(j,c)\alpha\pi_{0,j}, \quad 2 \le j \le K-1, \tag{3}$$

$$(\mu + \min(K-1,c-1)\alpha)\pi_{1,K} = \lambda\pi_{1,K-1} + c\alpha\pi_{0,K}. \tag{4}$$

It follows from (4) that

$$\pi_{1,K} = a_K^{(1)} + b_K^{(1)}\pi_{1,K-1},$$

leading to the fact that Lemma 1 is true for $j = K$. Assuming that Lemma 1 is true for $j+1$, i.e., $\pi_{1,j+1} = a_{j+1}^{(1)} + b_{j+1}^{(1)}\pi_{1,j}$. It then follows from (3) that Lemma 1 is also true for j, i.e., $\pi_{1,j} = a_j^{(1)} + b_j^{(1)}\pi_{1,j-1}$.

Theorem 2. *We have the following bound.*

$$a_j^{(1)} \ge 0, \qquad 0 \le b_j^{(1)} \le \frac{\lambda}{\mu + \min(j-1,c-1)\alpha},$$

for $j = 2, 3, \ldots, K-1, K$.

Proof. We use mathematical induction. It is easy to see that the theorem is true for $j = K$. Assuming that the theorem is true for $j+1$, i.e.,

$$a_{j+1}^{(1)} \geq 0, \qquad 0 \leq b_{j+1}^{(1)} \leq \frac{\lambda}{\mu + \min(j, c-1)\alpha}, \qquad j = 1, 2, \ldots, K-1.$$

Thus, we have $\mu b_{j+1}^{(1)} < \lambda$. From this inequality, (1) and (2), we obtain

$$b_j^{(1)} \leq \frac{\lambda}{\mu + \min(j-1, c-1)\alpha},$$

and $a_j^{(1)} \geq 0$.

It should be noted that $\pi_{2,2}$ can be calculated using the local balance between the flows in and out the set of states $\{(i,j); i = 0, 1, j = i, i+1, \ldots, K\}$ as follows.

$$2\mu\pi_{2,2} = \sum_{j=2}^{K} \min(j-1, c-1)\alpha\pi_{1,j}.$$

Remark 2. We have expressed $\pi_{1,j}$ $(j = 1, 2 \ldots, K)$ and $\pi_{2,2}$ in terms of $\pi_{0,0}$.

We consider the general case where $2 \leq i \leq c-1$. Similar to the case $i = 1$, we can prove the following result by mathematical induction.

Lemma 3. *We have*

$$\pi_{i,j} = a_j^{(i)} + b_j^{(i)}\pi_{i,j-1}, \qquad j = i+1, i+2, \ldots, K-1, K,$$

where

$$a_j^{(i)} = \frac{i\mu a_{j+1}^{(i)} + \min(c-i+1, j-i+1)\alpha\pi_{i-1,j}}{\lambda + \min(c-i, j-i)\alpha + i\mu - i\mu b_{j+1}^{(i)}}, \tag{5}$$

$$b_j^{(i)} = \frac{\lambda}{\lambda + \min(c-i, j-i)\alpha + i\mu - i\mu b_{j+1}^{(i)}}, \tag{6}$$

and

$$a_K^{(i)} = \frac{(c-i+1)\alpha\pi_{i-1,K}}{(c-i)\alpha + i\mu}, \qquad b_K^{(i)} = \frac{\lambda}{(c-i)\alpha + i\mu}.$$

Proof. The balance equation for state (i, K) is given as follows.

$$((c-i)\alpha + i\mu)\pi_{i,K} = \lambda\pi_{i,K-1} + (c-i+1)\alpha\pi_{i-1,K},$$

leading to the fact that Lemma 3 is true for $j = K$. Assuming that

$$\pi_{i,j+1} = a_{j+1}^{(i)} + b_{j+1}^{(i)}\pi_{i,j}, \qquad j = i+1, i+2, \ldots, K-1.$$

It then follows from

$$\begin{aligned}
&(\lambda + \min(c-i, j-i)\alpha + i\mu)\pi_{i,j} \\
&= \lambda\pi_{i,j-1} + i\mu\pi_{i,j+1} + \min(c-i+1, j-i+1)\alpha\pi_{i-1,j}, \\
&j = K-1, K-2, \ldots, i+1,
\end{aligned}$$

that

$$\pi_{i,j} = a_j^{(i)} + b_j^{(i)}\pi_{i,j-1}.$$

Theorem 4. *We have the following bound.*

$$a_j^{(i)} > 0, \qquad 0 < b_j^{(i)} < \frac{\lambda}{i\mu + \min(j-i, c-i)\alpha},$$

for $j = i+1, i+2, \dots, K-1, i = 1, 2, \dots, c-1$.

Proof. We also prove using mathematical induction. It is clear that Theorem 4 is true for $j = K$. Assuming that Theorem 4 is true for $j+1$, i.e.,

$$a_{j+1}^{(i)} > 0, \qquad 0 < b_{j+1}^{(i)} < \frac{\lambda}{i\mu + \min(j+1-i, c-i)\alpha},$$

for $j = i+1, i+2, \dots, K-1, i = 1, 2, \dots, c-1$. It follows from the second inequality that $i\mu b_{j+1}^{(i)} < \lambda$. This together with formulae (5) and (6) yield the desired result.

It should be noted that $\pi_{i+1,i+1}$ is calculated using the following local balance equation in and out the set of states:

$$\{(k,j); k = 0, 1, \dots, i; j = k, k+1, \dots, K\}$$

as follows.

$$(i+1)\mu\pi_{i+1,i+1} = \sum_{j=i+1}^{K} \min(j-i, c-i)\alpha\pi_{i,j}.$$

Remark 3. We have expressed $\pi_{i,j}$ $(i = 0, 1, \dots, c-1, j = i, i+1, \dots, K)$ and $\pi_{i+1,i+1}$ in terms of $\pi_{0,0}$.

Finally, we consider the case $i = c$. Balance equation for state (c, K) yields,

Lemma 5. *We have*

$$\pi_{c,j} = a_j^{(c)} + b_j^{(c)}\pi_{c,j-1}, \qquad j = c+1, c+2, \dots, K-1,$$

where

$$a_j^{(c)} = \frac{c\mu a_{j+1}^{(c)} + \alpha\pi_{c-1,j}}{\lambda + c\mu - c\mu b_{j+1}^{(c)}}, \qquad j = K-1, K-2, \dots, c+1, \tag{7}$$

$$b_j^{(c)} = \frac{\lambda}{\lambda + c\mu - c\mu b_{j+1}^{(c)}}, \qquad j = K-1, K-2, \dots, c+1, \tag{8}$$

and

$$a_K^{(c)} = \frac{\alpha\pi_{c-1,K}}{c\mu}, \qquad b_K^{(c)} = \frac{\lambda}{c\mu}.$$

Proof. The global balance equation at state (c, K) is given by

$$c\mu\pi_{c,K} = \alpha\pi_{c-1,K} + \lambda\pi_{c,K-1},$$

leading to

$$\pi_{c,K} = a_K^{(c)} + b_K^{(c)}\pi_{c,K-1}.$$

Assuming that $\pi_{c,j+1} = a_{j+1}^{(c)} + b_{j+1}^{(c)}\pi_{c,j}$, it follows from the global balance equation at state (c, j),

$$(\lambda + c\mu)\pi_{c,j} = \lambda\pi_{c,j-1} + c\mu\pi_{c,j+1} + \alpha\pi_{c-1,j}, \qquad j = c+1, c+2, \ldots, K-1,$$

that $\pi_{c,j} = a_j^{(c)} + b_j^{(c)}\pi_{c,j-1}$ for $j = c+1, c+2, \ldots, K$.

Theorem 6. *We have the following bound.*

$$a_j^{(c)} > 0, \qquad 0 < b_j^{(c)} < \frac{\lambda}{c\mu}, \qquad j = c+1, c+2, \ldots, K-1.$$

Proof. We also prove using mathematical induction. It is clear that Theorem 6 is true for $j = K$. Assuming that Theorem 6 is true for $j + 1$, i.e.,

$$a_{j+1}^{(c)} > 0, \qquad 0 < b_{j+1}^{(c)} < \frac{\lambda}{c\mu}, \qquad j = c+1, c+2, \ldots, K-1.$$

It follows from the second inequality that $c\mu b_{j+1}^{(c)} < \lambda$. This together with formulae (7) and (8) yield the desired result.

We have expressed all the probability $\pi_{i,j}$ $((i,j) \in \mathcal{S})$ in terms of $\pi_{0,0}$ which is uniquely determined by the normalizing condition.

$$\sum_{(i,j)\in\mathcal{S}} \pi_{i,j} = 1.$$

Remark 4. We see that the computational complexity order for $\{\pi_{i,j}; (i,j) \in \mathcal{S}\}$ is $O(cK)$. A direct method for solving the set of balance equations requires the complexity of $O(c^3K^3)$ while a level-dependent QBD approach (See Phung-Duc et al. [8]) needs the computational complexity of $O(Kc^3)$. We also observe that the recursion scheme of this paper is numerically stable since it manipulates only positive numbers (See Theorems 2, 4 and 6).

4 Performance Measures and Numerical Examples

4.1 Performance Measures

Let P_B denote the blocking probability. We have

$$P_B = \sum_{i=0}^{c} \pi_{i,K}.$$

Let π_i denote the stationary probability that there are i active servers, i.e., $\pi_i = \sum_{j=i}^{K} \pi_{i,j}$. Let $\mathbb{E}[A]$ and $\mathbb{E}[S]$ denote the mean number of active servers and that in setup mode, respectively. We have

$$\mathbb{E}[A] = \sum_{i=1}^{c} i\pi_i, \qquad \mathbb{E}[S] = \sum_{i=0}^{c} \sum_{j=i}^{K} \min(j-i, c-i)\pi_{i,j}.$$

The power consumption per a unit time for the model with setup time is given by

$$Cost_{on-off} = C_a\mathbb{E}[A] + C_s\mathbb{E}[S], \tag{9}$$

where C_a and C_s are the cost per a unit time for an active server and a server in setup mode, respectively.

For comparison, we also find the power consumption per a unit time for the corresponding ON-IDLE model, i.e., M/M/c/K without setup times. Letting p_i $(i = 0, 1, \dots, K-1, K)$ denote the stationary probability that there are i customers in the system, we have

$$p_i = \left(\frac{\lambda}{\mu}\right)^i \frac{1}{i!} p_0, \quad i = 0, 1, \dots, c,$$
$$p_i = p_c \left(\frac{\lambda}{c\mu}\right)^{i-c}, \quad i = c, c+1, \dots, K-1, K,$$

where p_0 is determined by the normalization condition $\sum_{i=0}^{K} p_i = 1$. Let $\mathbb{E}[\widehat{A}]$ denote the mean number of active servers, we have

$$\mathbb{E}[\widehat{A}] = \sum_{i=0}^{K} \min(i, c) p_i = \frac{\lambda(1-p_K)}{\mu},$$

where the second equality is due to Little's law. Therefore, the mean number of idle servers is given by $c - \mathbb{E}[\widehat{A}]$. Thus, for this model, the power consumption per a unit time is given by

$$Cost_{on-idle} = C_a\mathbb{E}[\widehat{A}] + (c - \mathbb{E}[\widehat{A}])C_i. \tag{10}$$

where C_i is the cost per a unit time for an idle server.

Let $\mathbb{E}[N]$ denote the mean number of customers in the system. We have

$$\mathbb{E}[N] = \sum_{i=0}^{c} \sum_{j=i}^{K} \pi_{i,j} \times j.$$

Let $\mathbb{E}[T]$ denote the mean response time of a customer. We have

$$\mathbb{E}[T] = \frac{\mathbb{E}[N]}{\lambda(1-P_B)}.$$

4.2 M/M/c/c System

We consider the following parameter setting: $c = K$, $\mu = \alpha = 1$. Furthermore, we set the cost for an active server and that for a setup server as $C_a = C_s = 1$ as in [7]. The cost for an idle server is $C_i = 0.6$ because an idle server still consumes 60% energy of its peak processing a job [1]. We investigate the power consumption for the M/M/c/K/Setup queue and its corresponding M/M/c/K model by (9) and (10), respectively. Figures 2 and 4 represent the blocking probability and power consumption against $\rho = \lambda/(c\mu)$ for the case $c = K = 50$ while Figures 3 and 5 represent those for the case $c = K = 500$. We observe that the blocking probability P_B decreases with α and is bounded from below by that of the corresponding ON-IDLE model (p_K). We also observe that our numerical scheme is stable since it can calculate the blocking probability of order 10^{-17}.

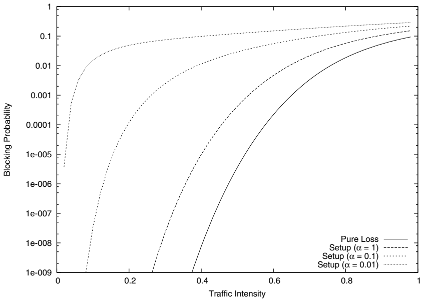

Fig. 2. Blocking probability against ρ ($c = 50$)

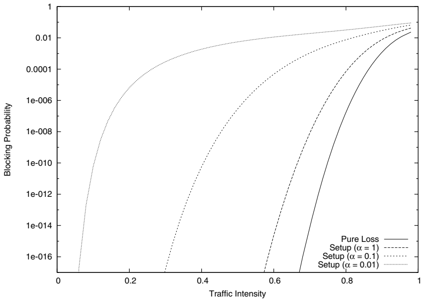

Fig. 3. Blocking probability against ρ ($c = 500$)

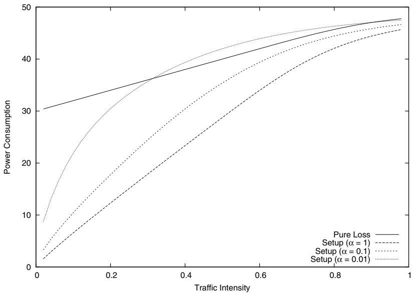

Fig. 4. Power consumption against ρ ($c = 50$)

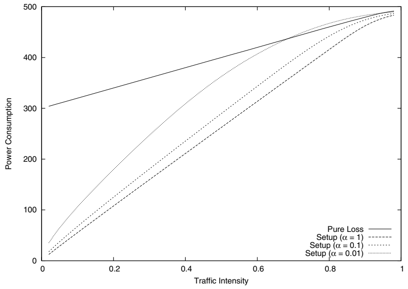

Fig. 5. Power consumption against ρ ($c = 500$)

We observe from Figures 4 and 5 that the power consumption increases with the traffic intensity ρ as expected. Furthermore, for the case $\alpha = 1, 0.1$, the ON-OFF policy outperforms the ON-IDLE one for any value of ρ. As for the case

$\alpha = 0.01$ there exist a range in which the power consumption of the ON-IDLE model is smaller than that of the ON-OFF model. Furthermore, the range for $c = 50$ is larger than that of $c = 500$. This suggests that the ON-OFF policy is more advanced in large-scale systems.

Figures 6 and 7 represent the mean number of setup servers $\mathbb{E}[S]$ against traffic intensity for the case $c = 50$ and $c = 500$, respectively. We observe that there exists some $\widehat{\rho}_\alpha$ such that $\mathbb{E}[S]$ increases with ρ in the range $(0, \widehat{\rho}_\alpha)$ while $\mathbb{E}[S]$ decreases with ρ for the range $(\widehat{\rho}_\alpha, 1)$. This is because when the traffic intensity is small, many servers are turned off. As a result, increasing the traffic intensity (number of arriving customers) incurs in the increase in the mean number of servers in setup. However, when the traffic intensity is large enough, almost the servers are likely on for all the time. Thus, the effect of setup is less and then the mean number of servers in setup time decreases with the traffic intensity.

4.3 Mean Response Time and Queue Length

In this section, we show the mean queue length ($\mathbb{E}[N]$) and the mean response time ($\mathbb{E}[T]$) of the M/M/100/K with setup time where $K =$ 200,500,1000, 2000 and 3000. We observe from Figures 8 and 9 that for $\alpha = 1, 0.1$, the mean response time and the mean queue length are unchanged for $K \geq 500$. This is because our system converges to the corresponding M/M/100/∞ as the capacity (K) tends to infinity. However, for the curves where $\alpha = 0.01$, we observe that $K = 2000$ is not large enough to approximate the infinite capacity system.

We observe in all the curves that the mean queue length increases with the traffic intensity. On the other hand, the mean response time decreases with ρ when ρ is small while it increases with ρ when ρ is large. This is because at low traffic intensity, the effect of setup time is large. Thus, increasing the traffic intensity incurs in increasing the number of setup servers. As a result the mean response time decreases. However, when the traffic intensity is large enough, it is likely that all the servers are ON for all the time. As a result, the effect of setup time decreases leading to the increase of the mean response time with the traffic intensity as in the conventional M/M/c/K system without setup time.

4.4 Effect of the Number of Servers

Figures 10 to 13 represent the ratio of the power consumption of the M/M/c/c with setup time against that of the corresponding M/M/c/c without setup time ($Cost_{on-off}/Cost_{on-idle}$) for $\rho = 0.3, 0.5, 0.7$ and 0.9. We observe that under all considered traffic intensities, the ratio is less than one for $\alpha = 1, 0.1$ meaning that the former is less power-consuming than the latter for $\alpha = 1$ and 0.1. On the other hand, for $\alpha = 0.01$, the latter outperforms the former for a wide range of c. This may be due to the fact that a large portion of customers are lost due to the slow setup ($1/\alpha = 100$). We observe in the case $\rho = 0.3, 0.5$ and 0.7 that the power consumption ratio decreases with c.

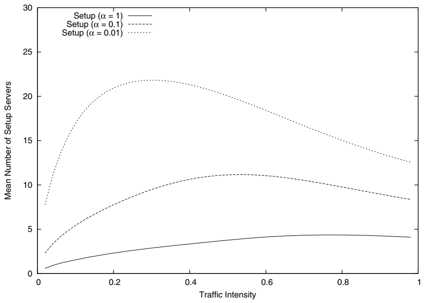

Fig. 6. Mean number of setup servers against ρ ($c = 50$)

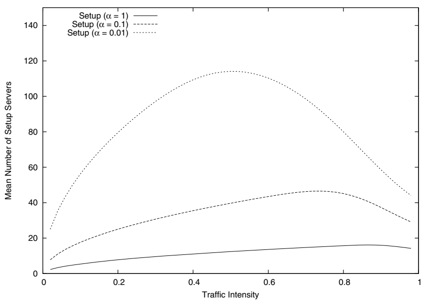

Fig. 7. Mean number of setup servers against ρ ($c = 500$)

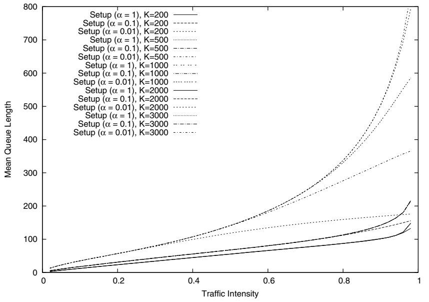

Fig. 8. Mean queue length against ρ ($c = 100$)

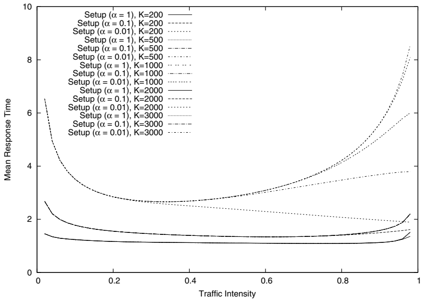

Fig. 9. Mean response time against ρ ($c = 100$)

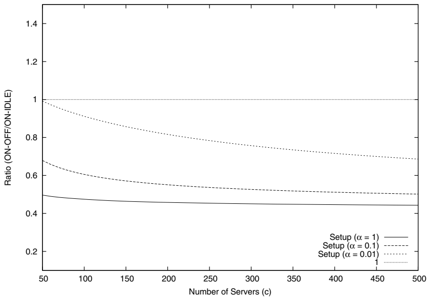

Fig. 10. Ratio of power consumption ($\rho = 0.3$)

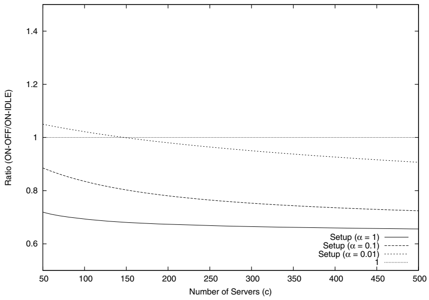

Fig. 11. Ratio of power consumption ($\rho = 0.5$)

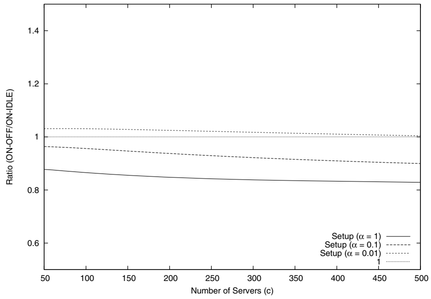

Fig. 12. Ratio of power consumption ($\rho = 0.7$)

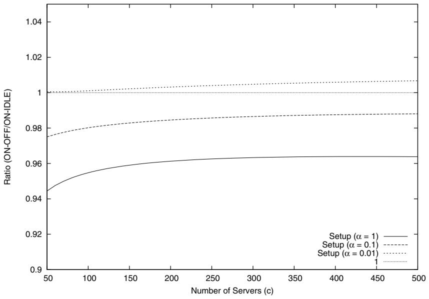

Fig. 13. Ratio of power consumption ($\rho = 0.9$)

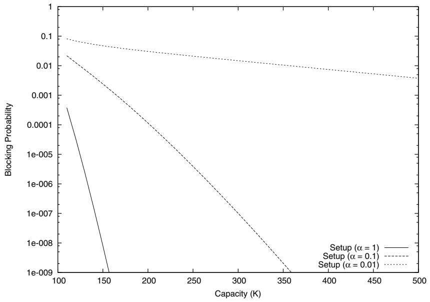

Fig. 14. Blocking probability against K ($\rho = 0.7, c = 100$)

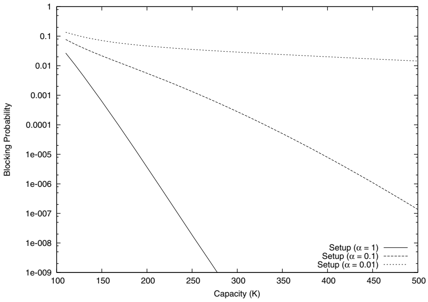

Fig. 15. Blocking probability against K ($\rho = 0.9, c = 100$)

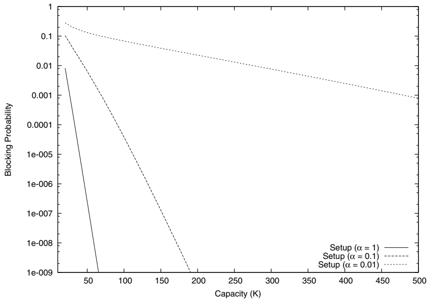

Fig. 16. Blocking probability against K ($\rho = 0.7, c = 10$)

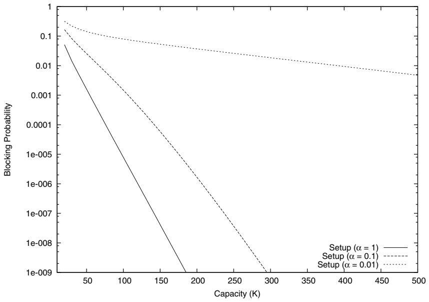

Fig. 17. Blocking probability against K ($\rho = 0.9, c = 10$)

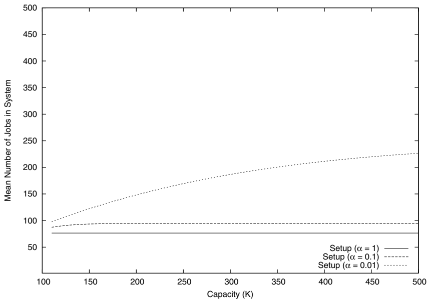

Fig. 18. Mean number of jobs in system against K $(\rho = 0.7, c = 100)$

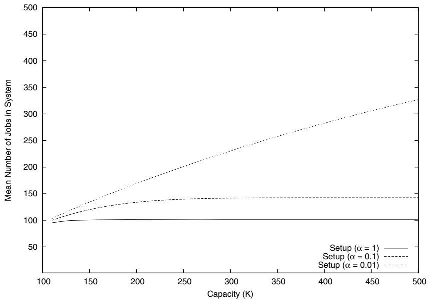

Fig. 19. Mean number of jobs in system against K $(\rho = 0.9, c = 100)$

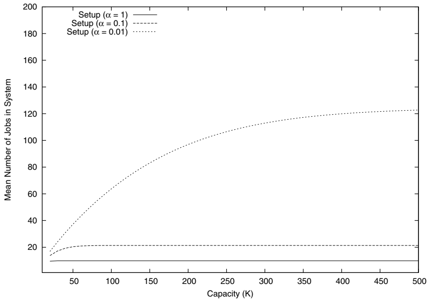

Fig. 20. Mean number of jobs in system against K $(\rho = 0.7, c = 10)$

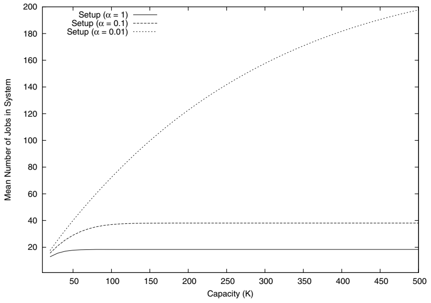

Fig. 21. Mean number of jobs in system against K $(\rho = 0.9, c = 10)$

4.5 Effect of the Capacity

In this section, we show the influence of the capacity K on the performance of the system. We consider the cases where $\rho = 0.7$ and $\rho = 0.9$ while $c = 10$ and 100. Figures 14 to 17 represent the blocking probability against K for the $c = 100, 10$ and $\rho = 0.7, 0.9$. We observe in all these graphs that the blocking probability geometrically decreases in K. We observe in the curves for $\alpha = 1, 0.1$ that the blocking probability is sensitive to K in the sense that it decreases with K at a high speed. On the other hand, we observe that the blocking blocking probability for the case $\alpha = 0.01$ is less sensitive to K in comparison with the cases $\alpha = 1, 0.1$.

Figures 18 to 21 represent the mean number of customers in the system against K for the $c = 100, 10$ and $\rho = 0.7, 0.9$. We observe in the graphs for $\rho = 0.7$ that the mean number of customers in the system increases with K and then converges to some fixed value. This is intuitive because our system converges to the M/M/c/∞ with setup time when $K \to \infty$. In the graphs for $\rho = 0.9$ we also observe that the mean number of customers in the system increases with K. Furthermore, when $\alpha = 1, 0.1$ the mean number of customers in the system converges to some fixed value for $K < 500$ however the curve for $\alpha = 0.01$ does not converge in the range $K < 500$. This suggests that in the case $\alpha = 0.01$ the queue length is very long and a large portion of customers are lost due to blocking. This is also supported from the curves for the blocking probability with $\alpha = 0.01$.

5 Concluding Remarks

We present a simple recursion to calculate the stationary distribution of the system state of an M/M/c/K queue with setup time for data centers. The computational complexity order of the algorithm is only $O(cK)$. The methodology of this paper can be applied for various variant models with setup time and finite buffer. In particular, the methodology of this paper can also be applied to the finite buffer counter part of the M/M/c queue with vacation presented in [12]. Furthermore, it is easy to extend the model in this paper to take into account the abandonment of customers [9]. This extension may be presented somewhere.

Acknowledgments. This work was supported in part by JSPS KAKENHI Grant Number 2673001. The author would like to thank the referees for constructive comments which improve the presentation of the paper.

References

1. Barroso, L.A., Holzle, U.: The case for energy-proportional computing. Computer **40**(12), 33–37 (2007)
2. Kuehn, P.J., Mashaly, M.E.: Automatic energy efficiency management of data center resources by load-dependent server activation and sleep modes. Ad Hoc Networks **25**, 497–504 (2015)
3. Gandhi, A., Harchol-Balter, M., Adan, I.: Server farms with setup costs. Performance Evaluation **67**, 1123–1138 (2010)
4. Gandhi, A., Doroudi, S., Harchol-Balter, M., Scheller-Wolf, A.: Exact analysis of the M/M/k/setup class of Markov chains via recursive renewal reward. In: Proceedings of the ACM SIGMETRICS, pp. 153–166. ACM (2013)
5. Gandhi, A., Doroudi, S., Harchol-Balter, M., Scheller-Wolf, A.: Exact analysis of the M/M/k/setup class of Markov chains via recursive renewal reward. Queueing Systems **77**(2), 177–209 (2014)
6. Slegers, J., Thomas, N., Mitrani, I.: Dynamic server allocation for power and performance. In: Kounev, S., Gorton, I., Sachs, K. (eds.) SIPEW 2008. LNCS, vol. 5119, pp. 247–261. Springer, Heidelberg (2008)
7. Mitrani, I.: Managing performance and power consumption in a server farm. Annals of Operations Research **202**(1), 121–134 (2013)
8. Phung-Duc, T., Masuyama, H., Kasahara, S., Takahashi, Y.: A simple algorithm for the rate matrices of level-dependent QBD processes. In: Proceedings of the 5th International Conference on Queueing Theory and Network Applications (QTNA2010), Beijing, China, pp. 46–52. ACM, New York (2010)

9. Phung-Duc, T.: Impatient customers in power-saving data centers. In: Sericola, B., Telek, M., Horváth, G. (eds.) ASMTA 2014. LNCS, vol. 8499, pp. 185–199. Springer, Heidelberg (2014)
10. Phung-Duc, T.: Server farms with batch arrival and staggered setup. In: Proceedings of the Fifth Symposium on Information and Communication Technology, pp. 240–247. ACM (2014)
11. Phung-Duc, T.: Exact solution for M/M/c/Setup queue (2014). http://arxiv.org/abs/1406.3084
12. Tian, N., Li, Q.L., Gao, J.: Conditional stochastic decompositions in the M/M/c queue with server vacations. Stochastic Models **15**, 367–377 (1999)