Performance Analysis of the M/G/c/c+r Queuing System for Cloud Computing Data Centers

Assia Outamazirt, Mohamed Escheikh, Djamil Aissani, Kamel Barkaoui, Ouiza Lekadir

Performance Analysis of the $M/G/c/c+r$ Queuing System for Cloud Computing Data Centers

Assia Outamazirt
Research Unit LaMOS
Faculty of Exact Sciences
University of Bejaia, Algeria
outamazirt.assia@gmail.com

Mohamed Escheikh
SYS'COM ENIT Tunis
Tunis, Tunisia
mohamed.escheikh@gmail.com

Djamil Aïssani
Research Unit LaMOS
Faculty of Exact Sciences
University of Bejaia, Algeria
lamos_bejaia@hotmail.com

Kamel Barkaoui
CEDRIC, CNAM
Paris, France
kamel.barkaoui@cnam.fr

Ouiza Lekadir
Research Unit LaMOS
Faculty of Exact Sciences
University of Bejaia, Algeria
ouizalekadir@gmail.com

In this paper, we propose $M/G/c/c+r$ queuing system as a model for performance evaluation of cloud server farms. Analytical resolution of this queuing system remains, to this day, an open and challenging issue because an exact analytical solution is difficult to reach. Therefore, we provide new approximate formulas to compute the transition-probability matrix of this system. In order to examine the accuracy of our approximate formulas, we test them numerically on some examples. Then, we compute the steady-state probabilities and some performance indicators such as blocking probability, mean response time, probability of immediate service and delay probability.

Cloud Computing, Performance Analysis, $M/G/c/c+r$ Queue, Embedded Markov Chain, Transition-probability Matrix.

1. INTRODUCTION

With accelerated advancement of cloud computing and the advent of new networking technologies, a wide spectrum of cloud services have been nowadays developed. The success of the large deployment of such services by cloud service providers closely depend on guaranteeing the advertised quality characteristics expressed toward diverse performance and quality of service (QoS) attributes and achieved through adopting appropriate resource provisioning strategies. The performance of cloud service is a key enabler of the overall performance of the next generation information infrastructure. Thorough assessment of cloud service is undoubtedly beneficial to both consumers and cloud service providers; thus representing an open issue in active research area. In this regard a tremendous and valuable effort has been devoted by the research community to tackle these challenges and a substantial progress has been made. Among the fundamental approaches enabling to evaluate cloud performance are those based on stochastic modeling. This latter allows to develop complex models and to capture composite behaviors usually involved in describing value added and reliable services.

Model complexity is usually expressed in terms of versatile queues taking into consideration multiple servers with general distribution and buffer limitation. These considerations may be apprehended through queuing systems such that M/G/c/c+r queue.

However providing an exact analytical solution to this queuing system is not obvious and often leads to cumbersome and computational overhead. In order to circumvent such problem usually several approximations may be used. This subject has been tackled by extensive researches described in the literature (see Boxma et al., (1979); Kimura, (1983); Nozaki and Ross, (1978); Tijms et al., (1981); Yao, (1985)). Most models proposed in these researches, as we will see in Section 2, are not applicable to performance analysis of cloud computing center. Therefore, Khazaei et al., (2012) proposed an approximate approach by using a combination of a

transform-based analytical model and an embedded Markov chain model to compute the transition-probability matrix of the $M/G/c/c+r$ queue. This matrix is divided into four regions, and the authors proposed an approximate formula for computing the transition probabilities of every region. However, transitions between system's states expressed toward conditional probabilities are not accurately described in this work. The problem of computing transition probability-matrix of the above queue has also been addressed by Chang et al., (2016). Authors in this work proposed other approximate formulas for transition probabilities computation in the regions 3 and 4. This resulted in a stochastic matrix for the $M/G/c/c+r$ queuing system only when the service time follows a gamma distribution. Such assumption is inappropriate, since the matrix should be stochastic regardless the service law (since the service is generally distributed).

In the previous work (Outamazirt et al., (2016)), the authors presented the shortcomings of the approximate formulas proposed by Khazaei et al., (2012) and those proposed by Chang et al., (2016). Furthermore, they proposed some refinement improving the above approximations. In this paper, we provide a detailed analysis of the $M/G/c/c+r$ queuing system in order to define in a first step the process describing the behavior of this system explicitly, and to propose in a second step new approximate formulas to compute the transition probabilities. In order to examine the accuracy of our approximate formulas, we test them numerically to verify that the transition probability-matrix is effectively stochastic regardless of the service distribution. From this matrix, we can calculate the steady-state probabilities and hence the performance indicators such as blocking probability, mean response time, probability of immediate service and the probability that an arrival task has to wait before beginning service. Finally, we validate our analytical results by simulation.

The remainder of the paper is organized as follows: Section 2 presents a brief overview of related work on cloud performance evaluation and performance characterization of queuing systems. In Section 3, we present in details the proposed analytical model. Section 4 is devoted to give some application examples for which the transition probability-matrix is computed for different service time distributions. In the same section, we perform a comparison between our obtained matrix and that obtained using the approximate formulas of Khazaei et al., (2012). In Section 5 some performance indicators for cloud center are defined and then evaluated. Finally, Section 6 concludes the paper.

2. RELATED WORK

Although cloud computing has attracted research attention, only a small portion of the work done so far addressed performance issues by analytical models. In this context, Xiong and Perros, (2009) modeled the cloud center as a classic open network, from which the distribution of response time was obtained by using the Laplace transformation. Using the distribution of response time, the authors found the relationship between the maximum number of customers, the minimal service resources and the highest level of services. Yang et al., (2009) proposed the $M/M/c/c+r$ queuing system for modeling the cloud center, from which the distribution of response time was determined. Inter-arrival and service times were both assumed to be exponentially distributed, and the system has a finite capacity.
However, the assumption of the service time being exponentially distributed is inappropriate for cloud center. Therefore, Khazaei et al., (2012) adopted the system with general service times as the abstract model for performance evaluation of cloud center. Then, they modeled a cloud center as an $M/G/c/c+r$ queuing system.

The analysis of queuing systems in the case where the inter-arrival time and/or the service time is not exponential is more complex. Various theoretical analyses were done on the performance evaluation of these queuing systems (see Ma, and Mark, (1995); Takahashi, (1977); Yao, (1985)). However, for these latter, the steady state probability, the distributions of response time and the queue length cannot be obtained in closed form. Consequently, several researchers developed many methods to approximate its solutions.

Kimura, (1983) applied the method of diffusion approximation to provide approximate formulas for the distributions of the number of customers, the waiting time and the busy period in the $M/G/c$ queue. Also, an approximation for the steady state queue length distribution in $M/G/c$ queue with finite waiting space was described by Kimura, (1996a).

A similar approach in the context of $M/G/c$ queue was described by Kimura, (1996b), but extended so as to approximate the blocking probability and, thus, to determine the smallest buffer capacity such that the rate of lost tasks remains under predefined level.

Nozaki and Ross, (1978) proposed an approximation for the average queuing delay in $M/G/c/c+r$ queue based on the relationship of joint distribution of remaining service time to the equilibrium service distribution. Smith, (2003) proposed a different approximation for the blocking probability based on

the exact solution for finite capacity $M/M/c/c+r$ queues. The estimate of the blocking probability is used to guide the allocation of buffers so that the blocking probability remains below a specific threshold.

However, most of these approximations are not suitable for performance evaluation of cloud center due to accuracy limitation related to several reasons. In what follows, we detail why the considered approximations in some previous works fail in providing acceptable accuracy in specific contexts and characterized by particular attribute (number of servers, coefficient of variation of the service time (CoV) and traffic intensity) (Khazaei et al., (2012)):

- for example, approximations proposed by Kimura, (1996a); Smith, (2003) are reasonably accurate when the number of servers is small enough, below 10 or so. They are not suitable for the cloud computing centers with more than 100 servers;
- approximations proposed by Nozaki and Ross, (1978); Yao, (1985) are inaccurate when the CoV is above 1.0;
- approximation errors are particularly pronounced when the traffic intensity ρ is small, and/or when both the number of servers c and the CoV of the service time are large (Boxma et al., (1979); Kimura, (1983); Tijms et al., (1981)).

Therefore, Khazaei et al., (2012) proposed suitable approach to approximate queues with large number of servers and service time distribution with CoV higher than one. This approach provides approximated formulas for the different transition probabilities expressions included in transition probability-matrix representing the $M/G/c/c+r$ queue. The main shortcoming of the above approach is that it is unable to fully describe the underlying stochastic process. More recent work (Chang et al., (2016)) proposed other approximate formulas that are only valid for gamma service distribution. This restriction had been made in order to keep relatively tractable the system resolution.

In this paper, we present a novel enhanced approximation enabling to better describe the stochastic process of the $M/G/c/c+r$ queuing system. For this purpose, we propose new explicit formulas to compute the different elements of the transition-probability matrix of this system. This allows to obtain a stochastic matrix for any service time distribution. Consequently, our work completes the cited previous works and permit an accurate performance analysis of cloud center.

3. THE PROPOSED MODEL

We consider the same modelling for cloud center as that proposed proposed by Khazaei et al., (2012) using the same $M/G/c/c+r$ queuing system. However, we perform a mathematical analysis of this system by introducing the stochastic process (the number of tasks present in the system) which better describes the system state transition behavior. Indeed, the description proposed by Khazaei et al., (2012) for this system is close to ours, since these both descriptions lead to a transition-probability matrix with four regions and then compute the steady-state probabilities by computing the transition probabilities in each region. The difference between these two descriptions lies in the computation of the transition probabilities for Region 3 and Region 4. In this section, we will give our description and a detailed analysis of this system.

3.1. The $M/G/c/c+r$ system description

As it has been pointed out, we model a cloud sever farms using the $M/G/c/c+r$ queuing system. In this system, task request arrivals follow a Poisson process. That is, the task inter-arrival time $A(x) \triangleq P[X \leq x]$ is exponentially distributed with rate λ, its Probability Density Function (PDF) is $a(x) = \lambda e^{-\lambda x}$ and its Laplace transform is

$$A^*(s) = \int_0^\infty e^{-sx} a(x) dx = \frac{\lambda}{\lambda + s}.$$

Task service times are identically and independently distributed according to a general distribution $H(y) \triangleq P[Y \leq y]$ with a mean service time equal to $\overline{h} = \frac{1}{\mu}$, its PDF is $h(y)$ and its Laplace transform is

$$H^*(s) = \int_0^\infty e^{-sy} h(y) dy.$$

The traffic intensity is $\rho \triangleq \frac{\lambda}{c\mu}$. The Residual task service time, H_+, is the time interval from an arbitrary point during a service time to the end of the service time. This time is necessary for our model since it represents time distribution between a task arrival and departure of the task which was in service when task arrival occurred. The Laplace transform of H_+ is given by Takagi, (1991) as:

$$H_+^*(s) = \frac{1 - H^*(s)}{s\,\overline{h}}. \tag{1}$$

In this model, we assume that:

- All c servers render service in order of task request arrivals (First Come First Served);
- The service time associated to the different servers is stochastically independent;

- If the waiting queue is empty and there is no new task request arrival, the server enters in the idle state;
- If the task arrives while the system capacity has already been attained, this task will depart immediately without service;
- Any task request goes through a facility node and then leaves the cloud center. A facility node may contain different computing resources such as web servers, database servers, etc.

3.2. The $M/G/c/c+r$ system analysis

3.2.1. The embedded Markov chain

The $M/G/c/c+r$ is a semi-markovian queuing system (Heyman and Sobel, (2004)) which can be analyzed by using the Embedded Markov Chain (EMC) technique similar to one adopted by Khazaei et al., (2012). This technique consists in selecting the Markov points at the instants of a new task arrival to the system. If we enumerate these instances as $0, 1, ..., c+r$, we obtain a homogeneous Markov chain with state space $S = \{0, 1, 2, ..., c+r\}$. This Markov chain is ergodic (Khazaei et al., (2012)), so its steady-state exists. Therefore, we can calculate the distribution of number of tasks in the system as well as the performance indicators. Due to the PASTA (Poisson Arrivals See Time Averages) property, the distribution of the number of tasks in the system at the time of a task arrival is identical to the distribution of the number of tasks in the system at an arbitrary time.

Let t_n (resp. t_{n+1}) the moment of the n^{th} (resp. $(n+1)^{\text{th}}$) arrival to the system, and X_n (resp. X_{n+1}) the number of tasks found in the system immediately before t_n (resp. t_{n+1}). While T_n (resp. B_{n+1}) denotes the inter-arrival time (resp. the number of tasks which depart from the system) between t_n and t_{n+1} (see Figure 1). In the rest of this paper we use T to denote any inter-arrival time.

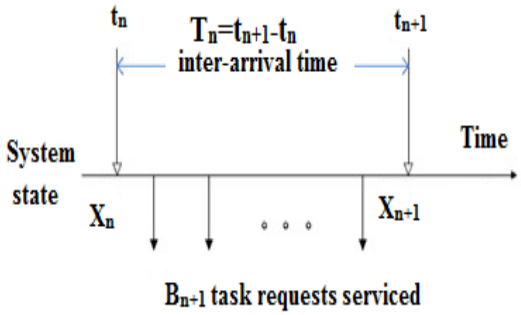

Figure 1: *The embedded Markov points.*

Thus, our Embedded Markov Chain is defined as follows:

$$X_{n+1} = \begin{cases} X_n + 1 - B_{n+1}, & \text{if } X_n < c+r; \\ X_n - B_{n+1}, & \text{if } X_n = c+r. \end{cases} \quad (2)$$

Now, we can compute the transition-probability matrix associated to this EMC.

3.2.2. The transition probability matrix

The transition probabilities of our transition matrix are defined by:

$$p_{ij} \triangleq P(X_{n+1} = j | X_n = i). \quad (3)$$

Taking into account the following points:

(i) Given the definition of X_n, obviously,

$$p_{ij} = 0 \quad \text{for all } \; j > i+1; \quad (4)$$

(ii) p_{ij} presents the probability of having exactly $(i+1-j)$ tasks are serviced during T, when we have $i < c+r$;

(iii) Because the n^{th} arrival finds the system in state $c+r$ (the system full), p_{c+rj} is the probability that exactly $(i-j)$ tasks are serviced during T. Similarly, because the n^{th} arrival finds the system in state $c+r-1$, p_{c+r-1j} presents the probability that exactly $(i-j)$ tasks are serviced during T. Thus, $p_{c+rj} = p_{c+r-1j}$ for all j;

and if we define $b_\omega = P(B_{n+1} = \omega)$ as the probability of having ω tasks are serviced during T, then $b_\omega \triangleq p_{ij}$. Thus, the state transition matrix of this EMC is given as follows:

$$P = \begin{array}{c} \\ 0 \\ 1 \\ 2 \\ \vdots \\ c-1 \\ c \\ \vdots \\ c+r-2 \\ c+r-1 \\ c+r \end{array} \begin{array}{c} \begin{array}{cccccccccc} 0 & 1 & 2 & \dots & c-1 & c & \dots & c+r-2 & c+r-1 & c+r \end{array} \\ \left(\begin{array}{cccccc|ccc} b_1 & b_0 & 0 & \dots & 0 & 0 & \dots & 0 & 0 & 0 \\ b_2 & b_1 & b_0 & \dots & 0 & 0 & \dots & 0 & 0 & 0 \\ b_3 & b_2 & b_1 & \dots & 0 & 0 & \dots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & & \vdots & \vdots & & \vdots & \vdots & \vdots \\ b_c & b_{c-1} & b_{c-2} & \dots & b_1 & b_0 & \dots & 0 & 0 & 0 \\ \hline b_{c+1} & b_c & b_{c-1} & \dots & b_2 & b_1 & \dots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & & \vdots & \vdots & & \vdots & \vdots & \vdots \\ b_{c+r-1} & b_{c+r-2} & b_{c+r-3} & \dots & b_r & b_{r-1} & \dots & b_1 & b_0 & 0 \\ b_{c+r} & b_{c+r-1} & b_{c+r-2} & \dots & b_{r+1} & b_r & \dots & b_2 & b_1 & b_0 \\ b_{c+r} & b_{c+r-1} & b_{c+r-2} & \dots & b_{r+1} & b_r & \dots & b_2 & b_1 & b_0 \end{array} \right) \end{array}$$

Region 2 | Region 1 | Region 4 | Region 3

As can be see, this matrix has four regions. Before computing b_ω in each region, we first define the departure probabilities P_x, P_y and $P_{z,k}$ as follows:

$$P_x \triangleq P(A > H_+) = H_+^*(\lambda), \quad (5)$$

$$P_y \triangleq P(A > H) = H^*(\lambda), \quad (6)$$

$$P_{z,k} = \Big[\prod_{i=2}^{k} P\Big(A > H | A > (k-i)H + H_+\Big)\Big] \times P(A > H_+), \quad (7)$$

where:

- P_x is the probability of completing the service of a task, which has already been in service during the previous observation interval and is completed in the current interval.
- P_y is the probability of completing the service of a task, which begins to be serviced in the current interval and is finished within the same interval.
- If a server completes the service of a task which has already begun during the previous observation interval, in the current interval, this server will be idle. If the waiting queue is nonempty, that server as well may complete a second service in the current interval, and if the waiting queue is still nonempty, a new service may be completed, and so on until the waiting queue gets empty. Thus, the probability of k services are completed by a single server is given by the formula (7).

With these departure probabilities, we can describe the four different regions of our transition-probability matrix P.

Region 1: From the formula (4), we have $p_{ij} = 0$ for $i+1 < j$.

Region 2: For $i < c$, $j \leq c$, and $i+1 \geq j$, all tasks are in the service (no waiting). The probability that ω tasks are serviced during T is given by:

$$p_{ij} = \binom{i}{i-j} P_x^{i-j}(1-P_x)^j P_y + \binom{i}{i-j+1} P_x^{i-j+1}(1-P_x)^{j-1}(1-P_y). \quad (8)$$

Region 3: For $c \leq i \leq c+r$, $c \leq j \leq c+r$, and $i+1 \geq j$, all servers are busy during T.
In order to minimize the error while keeping the model tractable, we assume, as it is assumed by Khazaei et al., (2012), that each single server completes no more than three services of tasks between two successive task arrivals. Then, the probability that ω tasks are serviced during T in this region is given by:

$$p_{ij} = \sum_{s_1=min(\omega,1)}^{min(\omega,c)} \binom{c}{s_1} P_x^{s_1}(1-P_x)^{c-s_1} \sum_{s_2=min(\omega-s_1,1)}^{min(\omega-s_1,s_1)} \binom{s_1}{s_2} P_{z,2}^{s_2}(1-P_{z,2})^{s_1-s_2} \binom{s_2}{\omega-s_1-s_2} P_{z,3}^{\omega-s_1-s_2} (1-P_{z,3})^{s_2-(w-s_1-s_2)}\Phi, \quad (9)$$

where Φ is the indicator function:

$$\Phi = \begin{cases} 1, & if \ \ \omega - s_1 - s_2 \leq s_2; \\ 0, & if \ \ \omega - s_1 - s_2 > s_2. \end{cases} \quad (10)$$

As it is pointed out in (iii), we have $p_{c+rj} = p_{c+r-1j}$ for all j when $i = c+r$.

Region 4: For $c \leq i \leq c+r$, $j < c$, and $i+1 \geq j$, all servers are busy at the beginning of T, and $c-j$ servers are idle at the end of T. Then, the probability that ω tasks are serviced during T is given by:

$$p_{ij} = \sum_{s_1=c-j}^{min(\omega,c)} \binom{c}{s_1} P_x^{s_1}(1-P_x)^{c-s_1} \sum_{s_2=min(\omega-s_1,c-j)}^{min(\omega-s_1,min(s_1,i-c+1))} \binom{s_1}{s_2} P_{z,2}^{s_2} (1-P_{z,2})^{min(s_1,i-c+1)-s_2} \binom{s_2}{\omega-s_1-s_2} P_{z,3}^{\omega-s_1-s_2} (1-P_{z,3})^{max(i-c+1-s_1,0)-(\omega-s_1-s_2)}\Phi. \quad (11)$$

Taking into account also the point (iii) in this region, we will have $p_{c+rj} = p_{c+r-1j}$ for all j when $i = c+r$.

3.2.3. Discussion

Our proposed formulas for both regions 1 and 2 are identical to those of Khazaei et al., (2012). However, due to the particularity of the model behavior in the regions 3 and 4, we adopted a more accurate analysis which gives new approximate formulas for these regions. Let us now examine in detail these new approximate formulas and compare them with those proposed by Khazaei et al., (2012).

Region 3 $(\mathbf{c \leq i \leq c+r, c \leq j \leq c+r, i+1 \geq j})$:
In this region, the n^{th} arrival finds all c servers are busy and $(i-c)$ tasks in the waiting queue. If the number of tasks in the system is strictly less than $c+r$ (i.e. $i < c+r$), then the n^{th} arrival will be allowed entry. Therefore, there should be $(i-c+1)$ tasks in the waiting queue at the beginning of T. Among these c busy servers, s_1 of them complete at least one service during T. Among these s_1 servers,

s_2 of them will complete a second service during T. As each single server completes no more than three services of tasks between two successive task arrivals, then the remaining $(\omega - s_1 - s_2)$ services must be completed before the end of the inter-arrival time; these services will complete by a subset of s_2 servers. The number of servers in this subset is equal to $(\omega - s_1 - s_2)$, it means, there are $(\omega - s_1 - s_2)$ servers each of which completes exactly three services in T, and the number of servers that are still busy processing the third service should be set to $s_2 - (\omega - s_1 - s_2)$. Note that this number of servers is set to s_2 in the approximate formula proposed by Khazaei et al., (2012), with the probability $(1 - P_{z,3})^{s_2}$. Thus, according to these authors this approximate formula is given as:

$$\begin{aligned} p_{ij} &= \sum_{s_1=min(\omega,1)}^{min(\omega,c)} \binom{c}{s_1} P_x^{s_1}(1-P_x)^{m-s_1} \\ &\quad \sum_{s_2=min(\omega-s_1,1)}^{min(\omega-s_1,s_1)} \binom{s_1}{s_2} P_{z,2}^{s_2}(1-P_{z,2})^{s_1-s_2} \\ &\quad \binom{s_2}{\omega-s_1-s_2} P_{z,3}^{\omega-s_1-s_2}(1-P_{z,3})^{s_2}. \quad (12) \end{aligned}$$

However, it is impossible to find exactly c busy servers at the end of T in this formula since the number of servers that are still busy processing the third service is set to s_2. Consequently, Chang et al., (2016) proposed a new approximate formula which is defined as:

$$\begin{aligned} p_{ij} &= \sum_{s_1=min(\omega,1)}^{min(\omega,c)} \binom{c}{s_1} P_x^{s_1}(1-P_x)^{c-s_1} \\ &\quad \sum_{s_2=min(\omega-s_1,1)}^{min(\omega-s_1,s_1)} \binom{s_1}{s_2} P_{z,2}^{s_2}(1-P_{z,2})^{s_1-s_2} \\ &\quad \binom{s_2}{\omega-s_1-s_2} P_{z,3}^{\omega-s_1-s_2} \\ &\quad (1-P_{z,3})^{s_2-(w-s_1-s_2)}. \quad (13) \end{aligned}$$

Furthermore, in formulas (12) and (13), all the remaining $(\omega - s_1 - s_2)$ tasks that must leave the system should be serviced by the subset of s_2 servers. When the number of s_2 servers is small, it can happen that the number of remaining tasks $(\omega - s_1 - s_2)$ exceeds s_2, this is due to the assumption that "`each single server completes no more than three services of tasks during` T". So, to take into account the effect of this assumption in our formula, we have defined the indicator function Φ which is given in the formula (10).

Moreover, when the n^{th} arrival finds the system full (i.e. $i = c + r$), then it will be lost. Therefore, there will be $(i - j)$ tasks that will leave the system between two successive task arrivals instead of $(i + 1 - j)$ tasks. This has not been taken into account in the formulas (12) and (13). To remedy this, we defined our EMC by the formula (2).

The considerations taken into account during our analysis of region 3, will be maintained in our analysis of region 4.

Region 4 $(\mathbf{c \leq i \leq c + r, 0 \leq j < c, i + 1 \geq j})$:
In this region, at the beginning of T there are $(i - c + 1)$ tasks in the waiting queue and all c servers are busy, while at the end of T, the waiting queue is empty and there are $(c - j)$ servers are idle.

The approximate formula proposed by Khazaei et al., (2012) for this region is given by:

$$\begin{aligned} p_{ij} &= \sum_{s_1=c-j}^{min(\omega,c)} \binom{c}{s_1} P_x^{s_1}(1-P_x)^{c-s_1} \\ &\quad \sum_{s_2=min(\omega-s_1,c-j)}^{min(w-s_1,s_1)} \binom{s_1}{s_2} P_{z,2}^{s_2}(1-P_{z,2})^{s_1-s_2} \\ &\quad \binom{s_2}{\omega-s_1-s_2} P_{z,3}^{w-s_1-s_2}(1-P_{z,3})^{s_2}. \quad (14) \end{aligned}$$

In this approximate formula, Khazaei et al., (2012) have not taken into account the number of tasks in the waiting queue at the beginning of T. Consequently, at the end of T, the number of idle servers differs from $(c - j)$, that is, there are some servers that are busy, while there is no task to serve, which is contradictory. Taking into account the number of tasks in the waiting queue at the beginning of T, we will distinguish two cases to be studied, namely:

- Case 1: If $s_1 \geq i - c + 1$, then all the $(i - c + 1)$ tasks in the waiting queue will begin service at a subset of the s_1 servers that have completed theirs first services in T, and there will be $\big(s_1 - (i - c + 1)\big)$ servers remain idle because the waiting queue is empty.
- Case 2: If $s_1 < i - c + 1$, there will be s_1 tasks which begin service at a subset of the s_1 servers that have completed one service in T, and there will be $\big((i - c + 1) - s_1\big)$ tasks waiting to be serviced.

Chang et al., (2016) proposed a new approximate formula for this region:

$$\begin{aligned} p_{ij} &= \frac{1}{(1-P_{z,3})^{c-j}} \sum_{s_1=c-j}^{min(\omega,c)} \binom{c}{s_1} P_x^{s_1}(1-P_x)^{c-s_1} \\ &\quad \sum_{s_2=min(\omega-s_1,c-j)}^{min(\omega-s_1,s_1)} \binom{min(i-c+1,s_1)}{s_2} P_{z,2}^{s_2} \\ &\quad (1-P_{z,2})^{s_1-s_2} \binom{min(max(i-c+1-s_1,0),s_2)}{\omega-s_1-s_2} \\ &\quad P_{z,3}^{\omega-s_1-s_2}(1-P_{z,3})^{s_2-max(i-c+1-s_1-s_2,0)}. \quad (15) \end{aligned}$$

In this formula, the authors considered the number of servers that are still busy processing the third

service is equal to $(s_2 - max(i-c+1-s_1-s_2, 0))$, because they assumed that all servers that enter in the idle state during T they do not complete the service, even though the servers are idle. In order to correctly account for the $(c-j)$ idle servers at the end of T, they multiplied their formula by $\frac{1}{(1-P_{z,3})^{c-j}}$.

In the case where $s_1 \geq i-c+1$, the formula (15) will be equal to:

$$\begin{aligned} p_{ij} &= \frac{1}{(1-P_{z,3})^{c-j}} \sum_{s_1=c-j}^{min(\omega,c)} \binom{c}{s_1} P_x^{s_1}(1-P_x)^{c-s_1} \\ &\sum_{s_2=min(\omega-s_1,c-j)}^{min(\omega-s_1,s_1)} \binom{i-c+1}{s_2} P_{z,2}^{s_2}(1-P_{z,2})^{s_1-s_2} \\ &\binom{min(max(i-c+1-s_1,0),s_2)}{\omega-s_1-s_2} P_{z,3}^{\omega-s_1-s_2} \\ &(1-P_{z,3})^{s_2-max(i-c+1-s_1-s_2,0)}. \end{aligned} \quad (16)$$

Thus, $max(i-c+1-s_1,0) = 0$, $min(max(i-c+1-s_1,0),s_2) = 0$ and $max(i-c+1-s_1-s_2,0) = 0$. As the waiting queue is empty, because all $(i-c+1)$ tasks in waiting queue enter service as $s_1 \geq i-c+1$, then the number of servers that will complete three services during T is equal to 0 ($\omega - s_1 - s_2$=0). Therefore, the number of servers that will be idle at the end of T is equal to s_2, i.e., $c-j = s_2$. Thus, the formula (16) will be equal to:

$$\begin{aligned} p_{ij} &= \sum_{s_1=c-j}^{min(\omega,c)} \binom{c}{s_1} P_x^{s_1}(1-P_x)^{c-s_1} \\ &\sum_{s_2=min(\omega-s_1,c-j)}^{min(\omega-s_1,s_1)} \binom{i-c+1}{s_2} P_{z,2}^{s_2} \\ &(1-P_{z,2})^{s_1-s_2}. \end{aligned} \quad (17)$$

In this formula, if the number of servers that complete the second service is equal to $(i-c+1)$, then at the end of T, the number of servers remain busy processing the second service is equal to $(i-c+1-s_2)$, while in formula (17), this number is equal to (s_1-s_2), consequently, the number of tasks in the system at the end of T exceeds j. Therefore, the coefficient $\frac{1}{(1-P_{z,3})^{c-j}}$ is not valid when $s_1 \geq i-c+1$, but it is valid just when $s_1 < i-c+1$.

Accounting for the above analysis for this region, we study the two cases cited above separately:

- In the case 1, s_2 tasks among $(i-c+1)$ tasks leave the system with a probability $P_{z,2}^{s_2}$, and $(i-c+1-s_2)$ tasks remain in service with a probability $(1-P_{z,2})^{i-c+1-s_2}$. In other words, the s_2 servers that complete a second service must be selected from the s_1 servers that have completed theirs services in T (i.e. $\binom{s_1}{s_2}$), and the maximum number of these servers is equal to $min(\omega - s_1, i-c+1)$.
  Thus, we proposed an approximate formula to compute the transition probabilities in this case as follows:

$$\begin{aligned} p_{ij} &= \sum_{s_1=c-j}^{min(\omega,c)} \binom{c}{s_1} P_x^{s_1}(1-P_x)^{c-s_1} \\ &\sum_{s_2=min(\omega-s_1,c-j)}^{min(\omega-s_1,i-c+1)} \binom{s_1}{s_2} P_{z,2}^{s_2} \\ &(1-P_{z,2})^{i-c+1-s_2}. \end{aligned} \quad (18)$$

- In this case 2, as $s_1 < i-c+1$, then s_2 tasks among s_1 tasks leave the system with a probability $P_{z,2}^{s_2}$, (s_1-s_2) tasks remain in service with a probability $(1-P_{z,2})^{s_1-s_2}$, and $\big((i-c+1)-s_1\big)$ tasks waiting to be serviced. These latter will begin service at a subset of the s_2 servers that have completed tow services during T. Among these $\big((i-c+1)-s_1\big)$ tasks, $(\omega-s_1-s_2)$ tasks leave the system with a probability $P_{z,3}^{\omega-s_1-s_2}$ and $\big((i-c+1-s_1)-(\omega-s_1-s_2)\big)$ tasks remain in service with a probability $(1-P_{z,3})^{(i-c+1-s_1)-(\omega-s_1-s_2)}$.
  Thus, we proposed an approximate formula to compute the transition probabilities in this case as follows:

$$\begin{aligned} p_{ij} &= \sum_{s_1=c-j}^{min(\omega,c)} \binom{c}{s_1} P_x^{s_1}(1-P_x)^{c-s_1} \\ &\sum_{s_2=min(\omega-s_1,c-j)}^{min(\omega-s_1,s_1)} \binom{s_1}{s_2} P_{z,2}^{s_2}(1-P_{z,2})^{s_1-s_2} \\ &\binom{s_2}{\omega-s_1-s_2} P_{z,3}^{\omega-s_1-s_2} \\ &(1-P_{z,3})^{(i-c+1-s_1)-(\omega-s_1-s_2)}\Phi. \end{aligned} \quad (19)$$

  Combining the two formulas (18) and (19), we obtained the formula (11) for computing the transition probabilities in region 4.

This analysis allowed us to propose a more appropriate new matrix for the $M/G/c/c+1$ queue.

4. APPLICATION EXAMPLE

In order to confirm our theoretical results, we consider in this section the example model "$M/G/2/4$ queue" for which we will compute the matrix P using both approximate formulas, those of Khazaei et al., (2012) and ours.

Let $\widetilde{P}$ the transition-probability matrix of the $M/G/2/4$ queue found using our approximate

formulas:

$$\widetilde{P} = \begin{array}{c} \\ 0 \\ 1 \\ 2 \\ 3 \\ 4 \end{array} \begin{array}{c} \begin{array}{ccccc} 0 & 1 & 2 & 3 & 4 \end{array} \\ \begin{pmatrix} & \widetilde{A} & & 0 & 0 \\ & & & 0 & 0 \\ \widetilde{B} & & & \widetilde{C} & \end{pmatrix} \end{array},$$

where:

$$\widetilde{\boldsymbol{A}} = \begin{pmatrix} P_y & 1-P_y & 0 \\ P_x P_y & \left[(1-P_x)P_y + P_x(1-P_x)\right] & (1-P_x)(1-P_y) \end{pmatrix},$$

$$\widetilde{\boldsymbol{B}} = \begin{pmatrix} 2P_x^2 P_{z,2} & \left[2P_x(1-P_x)P_{z,2} + P_x^2 \times(1-P_{z,2})\right] \\ P_x^2 P_{z,2}^2 & \left[2P_x(1-P_x)P_{z,2}P_{z,3} + 2P_x^2 \times P_{z,2}(1-P_{z,2})\right] \\ P_x^2 P_{z,2}^2 & \left[2P_x(1-P_x)P_{z,2}P_{z,3} + 2P_x^2 \times P_{z,2}(1-P_{z,2})\right] \end{pmatrix},$$

$$\widetilde{\boldsymbol{C}} = \begin{pmatrix} 2P_x(1-P_x)(1-P_{z,2}) & (1-P_x)^2 & 0 \\ \left[2P_x(1-P_x)P_{z,2} \times(1-P_{z,3})\right] + \left[P_x^2(1-P_{z,2})^2\right] & \left[2P_x(1-P_x) \times(1-P_{z,2})\right] & (1-P_x)^2 \\ \left[2P_x(1-P_x)P_{z,2} \times(1-P_{z,3})\right] + \left[P_x^2(1-P_{z,2})^2\right] & \left[2P_x(1-P_x) \times(1-P_{z,2})\right] & (1-P_x)^2 \end{pmatrix}.$$

Let $\widetilde{\widetilde{P}}$ the transition-probability matrix of the $M/G/2/4$ queue found using the approximate formulas proposed by Khazaei et al., (2012)):

$$\widetilde{\widetilde{P}} = \begin{array}{c} \\ 0 \\ 1 \\ 2 \\ 3 \\ 4 \end{array} \begin{array}{c} \begin{array}{ccccc} 0 & 1 & 2 & 3 & 4 \end{array} \\ \begin{pmatrix} & \widetilde{\widetilde{A}} & & 0 & 0 \\ & & & 0 & 0 \\ \widetilde{\widetilde{B}} & & & \widetilde{\widetilde{C}} & \end{pmatrix} \end{array},$$

where:

$$\widetilde{\widetilde{\boldsymbol{A}}} = \begin{pmatrix} P_y & 1-P_y & 0 \\ P_x P_y & (1-P_x)P_y + P_x(1-P_x) & (1-P_x)(1-P_y) \end{pmatrix},$$

$$\widetilde{\widetilde{\boldsymbol{B}}} = \begin{pmatrix} \left[2P_x^2 P_{z,2}(1-P_{z,2})\right] \times\left[(1-P_{z,3})\right] & \left[2P_x(1-P_x)P_{z,2}(1-P_{z,3})\right] + \left[P_x^2(1-P_{z,2})^2\right] \\ P_x^2 P_{z,2}^2(1-P_{z,3})^2 & \left[2P_x(1-P_x)P_{z,2}P_{z,3} \times(1-P_{z,3})\right] + \left[2P_x^2 P_{z,2}(1-P_{z,2})(1-P_{z,3})\right] \\ 2P_x^2 P_{z,2}^2 P_{z,3}(1-P_{z,3})^2 & \left[2P_x^2 P_{z,2}(1-P_{z,2})P_{z,3} \times(1-P_{z,3})\right] + \left[P_x^2 P_{z,2}^2(1-P_{z,3})^2\right] \end{pmatrix},$$

$$\widetilde{\widetilde{\boldsymbol{C}}} = \begin{pmatrix} \left[2P_x(1-P_x) \times(1-P_{z,2})\right] & (1-P_x)^2 & 0 \\ \left[2P_x(1-P_x) \times P_{z,2}(1-P_{z,3})\right] + \left[P_x^2(1-P_{z,2})^2\right] & \left[2P_x(1-P_x) \times(1-P_{z,2})\right] & (1-P_x)^2 \\ \left[2P_x(1-P_x)P_{z,2} \times P_{z,3}(1-P_{z,3})\right] + \left[2P_x^2 P_{z,2} \times(1-P_{z,2}) \times(1-P_{z,3})\right] & \left[2P_x(1-P_x) \times P_{z,2}(1-P_{z,3})\right] + \left[P_x^2(1-P_{z,2})^2\right] & \left[2P_x(1-P_x) \times(1-P_{z,2})\right] \end{pmatrix}.$$

As we know, a transition-probability matrix is a stochastic matrix by definition. Thus, our main contribution lies in obtaining a stochastic matrix, which cannot be obtained using the approximate formulas proposed by Khazaei et al., (2012), which we will confirm in what follows.
In a stochastic matrix, the sum of transition probabilities from state i to all other states must be 1. So, if there is at least one state i such that this sum is different from 1 in a matrix, then this latter is no longer stochastic.
Let's calculate for example the sum of row 4 for $\widetilde{P}$ (resp. $\widetilde{\widetilde{P}}$):

$$\begin{aligned} \sum_{j=0}^{4} \widetilde{p}_{4j} &= 1 - 2P_x P_{z,2} P_{z,3} + 2P_x^2 P_{z,2} P_{z,3} + 2P_x P_{z,2} P_{z,3} - \\ &\quad 2P_x^2 P_{z,2} P_{z,3} = 1. \\ \sum_{j=0}^{4} \widetilde{\widetilde{p}}_{4j} &= 2P_x - P_x^2 - 2P_{z,2} P_{z,3} + 2P_x^2 P_{z,2} P_{z,3} - 2P_x^2 P_{z,2} P_{z,3}^2 - \\ &\quad P_x^2 P_{z,2}^2 P_{z,3}^2 + 2P_x^2 P_{z,2}^2 P_{z,3}^3 + 2P_{z,2} P_{z,3} - 2P_{z,2} P_{z,3}^2 - \\ &\quad 2P_x^2 P_{z,2} P_{z,3} + 2P_x^2 P_{z,2} P_{z,3}^2. \\ &= 2P_x - P_x^2 - P_x^2 P_{z,2}^2 P_{z,3}^2 + 2P_x^2 P_{z,2}^2 P_{z,3}^3 - 2P_{z,2} P_{z,3}^2. \end{aligned}$$

Despite that our formulas are approximates, we obtained in this case $\sum_{j=0}^{4} \widetilde{p}_{4j}$ is exactly equal to 1, which explains that our approximate formulas are more accurate. $\sum_{j=0}^{4} \widetilde{\widetilde{p}}_{4j}$ can be only verified through numerical analysis, for this we give some numerical examples where we assume different service time distributions with the same traffic intensity ρ assumed by Khazaei et al., (2012).

Example 1 *Exponential service times with rate of service $\mu = 0.6$ and $\rho = 0.85$:*

$$\widetilde{P} = \begin{array}{c} \\ 0 \\ 1 \\ 2 \\ 3 \\ 4 \end{array} \begin{array}{c} \begin{array}{ccccc} 0 & 1 & 2 & 3 & 4 \end{array} \\ \begin{pmatrix} 0.37 & 0.62 & 0 & 0 & 0 \\ 0.13 & 0.46 & 0.39 & 0 & 0 \\ 0.03 & 0.18 & 0.40 & 0.39 & 0 \\ 0.00 & 0.03 & 0.16 & 0.40 & 0.39 \\ 0.00 & 0.03 & 0.16 & 0.40 & 0.39 \end{pmatrix} \end{array} \begin{array}{l} \\ \sum_j = 0.99 \\ \sum_j = 0.98 \\ \sum_j = 1.00 \\ \sum_j = 0.98 \\ \sum_j = 0.98 \end{array}$$

$$\tilde{\tilde{P}} = \begin{array}{c} \\ 0 \\ 1 \\ 2 \\ 3 \\ 4 \end{array} \begin{array}{c} \begin{array}{ccccc} 0 & 1 & 2 & 3 & 4 \end{array} \\ \begin{pmatrix} 0.37 & 0.62 & 0 & 0 & 0 \\ 0.13 & 0.46 & 0.39 & 0 & 0 \\ 0.03 & 0.16 & 0.40 & 0.39 & 0 \\ 0.00 & 0.03 & 0.16 & 0.40 & 0.39 \\ 0.00 & 0.00 & 0.03 & 0.16 & 0.40 \end{pmatrix} \end{array} \begin{array}{c} \\ \sum_j = 0.99 \\ \sum_j = 0.98 \\ \sum_j = 0.98 \\ \sum_j = 0.98 \\ \sum_j = 0.59 \end{array}$$

Example 2 *Erlang service times with shape parameter $a = 4$, scale parameter $\mu = 0.6$, and $\rho = 0.85$:*

$$\tilde{P} = \begin{array}{c} \\ 0 \\ 1 \\ 2 \\ 3 \\ 4 \end{array} \begin{array}{c} \begin{array}{ccccc} 0 & 1 & 2 & 3 & 4 \end{array} \\ \begin{pmatrix} 0.24 & 0.75 & 0 & 0 & 0 \\ 0.10 & 0.47 & 0.41 & 0 & 0 \\ 0.04 & 0.23 & 0.44 & 0.30 & 0 \\ 0.00 & 0.04 & 0.20 & 0.44 & 0.30 \\ 0.00 & 0.04 & 0.20 & 0.44 & 0.30 \end{pmatrix} \end{array} \begin{array}{c} \\ \sum_j = 0.99 \\ \sum_j = 0.98 \\ \sum_j = 1.01 \\ \sum_j = 0.98 \\ \sum_j = 0.98 \end{array}$$

$$\tilde{\tilde{P}} = \begin{array}{c} \\ 0 \\ 1 \\ 2 \\ 3 \\ 4 \end{array} \begin{array}{c} \begin{array}{ccccc} 0 & 1 & 2 & 3 & 4 \end{array} \\ \begin{pmatrix} 0.24 & 0.75 & 0 & 0 & 0 \\ 0.10 & 0.47 & 0.41 & 0 & 0 \\ 0.03 & 0.20 & 0.44 & 0.30 & 0 \\ 0.00 & 0.03 & 0.20 & 0.44 & 0.30 \\ 0.00 & 0.00 & 0.03 & 0.20 & 0.44 \end{pmatrix} \end{array} \begin{array}{c} \\ \sum_j = 0.99 \\ \sum_j = 0.98 \\ \sum_j = 0.97 \\ \sum_j = 0.97 \\ \sum_j = 0.67 \end{array}$$

Example 3 *Weibull service times with shape parameter $a = 0.8$, scale parameter $\mu = 0.6$, and $\rho = 0.85$:*

$$\tilde{P} = \begin{array}{c} \\ 0 \\ 1 \\ 2 \\ 3 \\ 4 \end{array} \begin{array}{c} \begin{array}{ccccc} 0 & 1 & 2 & 3 & 4 \end{array} \\ \begin{pmatrix} 0.35 & 0.64 & 0 & 0 & 0 \\ 0.09 & 0.43 & 0.47 & 0 & 0 \\ 0.01 & 0.10 & 0.35 & 0.53 & 0 \\ 0.00 & 0.01 & 0.09 & 0.35 & 0.53 \\ 0.00 & 0.01 & 0.09 & 0.35 & 0.53 \end{pmatrix} \end{array} \begin{array}{c} \\ \sum_j = 0.99 \\ \sum_j = 0.99 \\ \sum_j = 0.99 \\ \sum_j = 0.98 \\ \sum_j = 0.98 \end{array}$$

$$\tilde{\tilde{P}} = \begin{array}{c} \\ 0 \\ 1 \\ 2 \\ 3 \\ 4 \end{array} \begin{array}{c} \begin{array}{ccccc} 0 & 1 & 2 & 3 & 4 \end{array} \\ \begin{pmatrix} 0.35 & 0.64 & 0 & 0 & 0 \\ 0.09 & 0.43 & 0.47 & 0 & 0 \\ 0.01 & 0.09 & 0.35 & 0.53 & 0 \\ 0.00 & 0.01 & 0.09 & 0.35 & 0.53 \\ 0.00 & 0.00 & 0.01 & 0.09 & 0.35 \end{pmatrix} \end{array} \begin{array}{c} \\ \sum_j = 0.99 \\ \sum_j = 0.99 \\ \sum_j = 0.98 \\ \sum_j = 0.98 \\ \sum_j = 0.45 \end{array}$$

Example 4 *Gamma service times with shape parameter $a = 0.2$, scale parameter $\mu = 0.6$, and $\rho = 0.85$:*

$$\tilde{P} = \begin{array}{c} \\ 0 \\ 1 \\ 2 \\ 3 \\ 4 \end{array} \begin{array}{c} \begin{array}{ccccc} 0 & 1 & 2 & 3 & 4 \end{array} \\ \begin{pmatrix} 0.63 & 0.36 & 0 & 0 & 0 \\ 0.13 & 0.57 & 0.28 & 0 & 0 \\ 0.01 & 0.08 & 0.28 & 0.61 & 0 \\ 0.00 & 0.01 & 0.07 & 0.28 & 0.61 \\ 0.00 & 0.01 & 0.07 & 0.28 & 0.61 \end{pmatrix} \end{array} \begin{array}{c} \\ \sum_j = 0.99 \\ \sum_j = 0.98 \\ \sum_j = 0.98 \\ \sum_j = 0.97 \\ \sum_j = 0.97 \end{array}$$

$$\tilde{\tilde{P}} = \begin{array}{c} \\ 0 \\ 1 \\ 2 \\ 3 \\ 4 \end{array} \begin{array}{c} \begin{array}{ccccc} 0 & 1 & 2 & 3 & 4 \end{array} \\ \begin{pmatrix} 0.63 & 0.36 & 0 & 0 & 0 \\ 0.13 & 0.57 & 0.28 & 0 & 0 \\ 0.01 & 0.07 & 0.28 & 0.61 & 0 \\ 0.00 & 0.01 & 0.07 & 0.28 & 0.61 \\ 0.00 & 0.00 & 0.01 & 0.07 & 0.28 \end{pmatrix} \end{array} \begin{array}{c} \\ \sum_j = 0.99 \\ \sum_j = 0.98 \\ \sum_j = 0.97 \\ \sum_j = 0.97 \\ \sum_j = 0.36 \end{array}$$

From these examples, we notice that the sum of the elements of each row for the matrices obtained by our approximate formulas, is equal or very close to 1, in opposite to that obtained using the approximate formulas of Khazaei et al., (2012), which is very far from 1, see particularly the last row.
This application example proves that our matrices are stochastic, which confirms that our approximate formulas are more accurate.

5. PERFORMANCE EVALUATION

In this section, we will solve the balance equations of the proposed model using MATLAB to obtain the steady-state probabilities. Then, we will compute some performance indicators such as blocking probability, mean response time, probability of immediate service and delay probability. Finally, we will validate the obtained results by simulation.

5.1. Equilibrium balance equations

Due to ergodicity of the system, the equilibrium probability distribution of the number of tasks present at the arrival instants $\boldsymbol{\pi} = (\pi_0, \pi_1, ..., \pi_{c+r})$, where $\pi_i = \lim_{n\to\infty} P(X_n = i)$, $0 \leq i \leq c+r$, exists. $\boldsymbol{\pi}$ can be obtained by solving the following system of linear equations:

$$\begin{cases} \boldsymbol{\pi} P = \boldsymbol{\pi} \\ \boldsymbol{\pi} \mathbf{1} = 1, \end{cases} \tag{20}$$

where, $\mathbf{1}$ is the column vector of ones.

This system cannot be solved in closed form, therefore we have to resort to a numerical solution.

5.2. Numerical results

There is no precise statistic or empirical results on percentage of different types of instances for a real cloud provider. For instance, Amazon does not publish any information regarding average traffic intensity, buffer space and the percentage of reserved, on-demand or spot instances in their various centers (Khazaei et al., (2012)). Therefore, Khazaei et al., (2012) assumed the traffic intensity of cloud center is rather high, $\rho = 0.85$, because they supposed that a cloud provider tries to keep the traffic intensity up as much as possible in order to optimize the use of the deployed infrastructure. Thus, we perform our results with the same assumptions of Khazaei et al., (2012), that is, the task service time follows the gamma distribution; the traffic intensity $\rho = 0.85$ and $CoV = 0.5, 1.4$. In all plots, the analytical results and simulation are labelled with Ana and Sim, respectively.

• First we present the blocking probability which we illustrate in Figure 2.

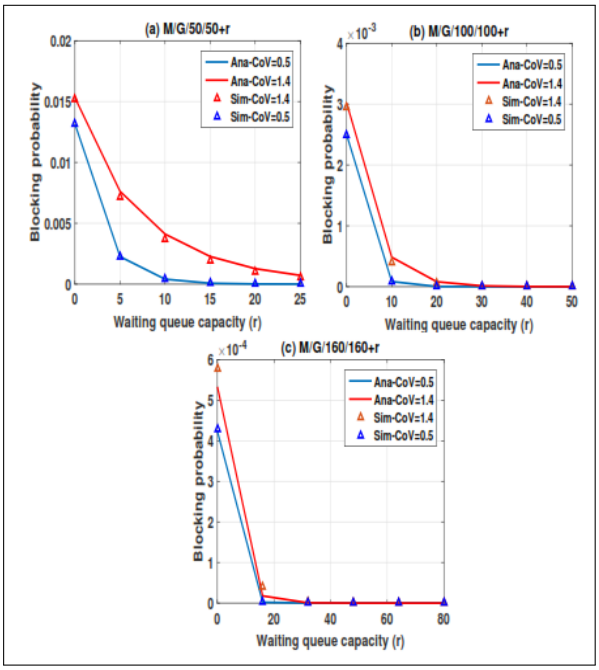

Figure 2: *Blocking probability.*

From the Figure 2, the results confirm that if the capacity of the waiting queue increased linearly the blocking probability would decrease exponentially. In the system with 50 servers, the blocking probability varies from 0.002 to 4×10^{-4} when the capacity of the waiting queue varies from 5 to 10 respectively. It equals to 0.8×10^{-4} in the system with 100 servers when the capacity of the waiting queue equals to 10. While for the system with 160 servers, the blocking probability is much lower. Thus, we can estimate the smallest capacity of waiting queue such that the blocking probability remains below a predefined value ϵ. For $\epsilon = 0.8 \times 10^{-4}$, the capacity of waiting queue should be at least 10.

• The probability of immediate service is shown in Figure 3.

As can be seen, the probability of immediate service decreases with the increase of the capacity of the waiting queue. We also notice in Figure 3, that this probability is close to the value 1 with a low value of the capacity of the waiting queue, which explains an arrival task can be directly served without joining the waiting queue.

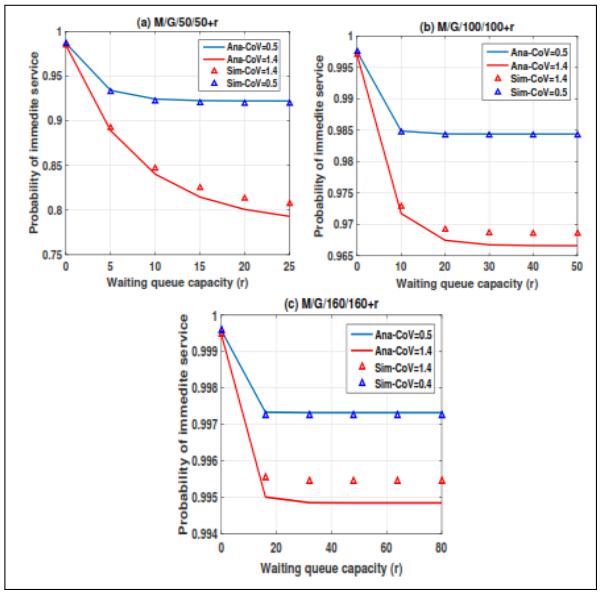

Figure 3: *Probability of immediate service.*

• It is also important to know the delay probability (the probability that an arrival task must wait because all servers are busy) by either cloud provider or its customers, because it can happen that an arrival task will leave the system without obtaining service due to long queuing length. So, we compute this probability, and the results are shown in Figure 4:

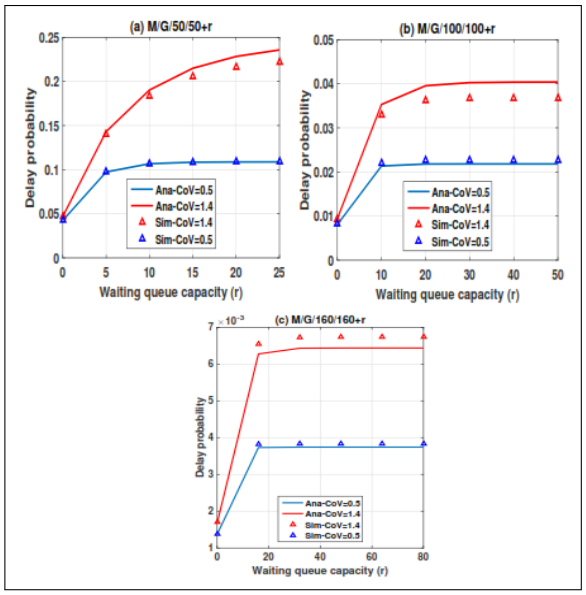

Figure 4: *Delay probability.*

As can be seen, the delay probability increases rapidly when the capacity of the waiting queue increases. This probability is low when the capacity of the waiting queue is very low, which explains that an arrival task has only to wait the service time of tasks which are in the service. We also notice in

Figure 4, that the delay probability in the system with higher number of servers is different from that in the system with a smaller number of servers.

• Finally, Figure 5 shows the mean response time:

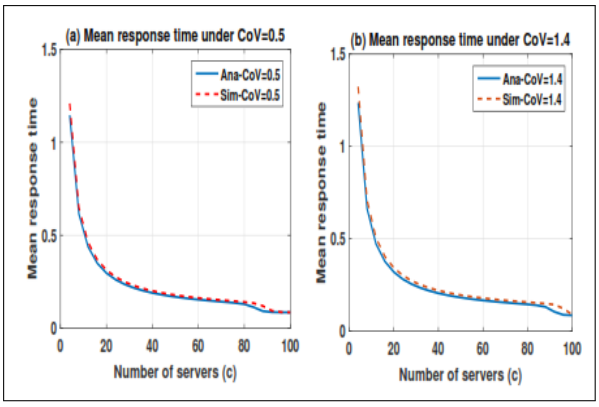

Figure 5: *Mean response time.*

As can be seen in this figure, the mean response time decreases when the number of servers increases. We notice that the mean response time under $CoV = 0.5$ is shorter than the mean response time under $CoV = 1.4$. Thus, we can conclude that it is better to submit the task requests to cloud centers that accept the same type of request than cloud centers that accept different types of requests.

Overall, the results suggest that performance is worse when the coefficient of variation of the service time is equal to 1.4, compared to performance with a CoV of the service time equals to 0.5. Finally, we note that the obtained analytical results are close to those obtained by simulation, which confirms the validity of our analytical model.

6. CONCLUSION AND FUTURE WORK

Due to the nature of environment of cloud computing, it is not feasible to obtain an adequate model for performance evaluation of cloud center. According to our related works, we observed that the $M/G/c/c+r$ queue is an abstract model to performance analysis of cloud center. In fact, considering this model, we performed a mathematical analysis of $M/G/c/c+r$ queuing system by introducing the stochastic process which is more appropriate, for the number of tasks present in the system at the arrival instants. Then, we proposed more accurate approximate formulas for computing the transition-probability matrix of this system. From this obtained matrix, we obtained the steady-state probabilities which allowed us to accurately compute the performance indicators such as the blocking probability, the mean response time, the probability of immediate service and the delay probability. Finally, we validated our analytical results by simulation.

For the future, we plan to extend the model $M/G/c/c+r$ queuing system to consider the services with different priorities and/or batch-task arrivals. It may also be interesting to extend this model taking into account the effect of impatient customers behavior on the total revenue of cloud providers.

REFERENCES

Boxma, O.J., Cohen, J.W. and Huffel, N. (1979) 'Approximations of the Mean Waiting Time in an M/G/s Queueing System', *Journal of Operations Research*, vol. 27 Issue 6, pp. 1115-1127.

Kimura, T. (1983) 'Diffusion Approximation for an M/G/m Queue', *Journal of Operations Research*, vol. 31 Issue 2, pp. 304-321.

Nozaki, S.A. and Ross, S.M. (1978) 'Approximations in Finite-Capacity Multi-Server Queues with Poisson Arrivals', *Journal of Applied Probability*, vol. 15 No. 4, pp. 826-834.

Tijms, H.C., Hoorn, M.H.V. and Federgru, A. (1981) 'Approximations for the Steady-State Probabilities in the M/G/c Queue', *Journal of Advances in Applied Probability*, vol. 13 No 1, pp. 186-206.

Yao, D.D. (1985) 'Refining the diffusion approximation for the M/G/m queue', *Journal of Operations Research*, vol. 33 Issue 6, pp. 1266-1277.

Khazaei, H., Misic, J. and Vojislav, B.M. (2012) 'Performance analysis of cloud computing centers using M/G/m/m+r queueing systems', *Journal of IEEE Transactions on Parallel and Distributed Systems*, vol. 23 Issue 5, pp. 936-943.

Chang, X., Wang, B., Muppala, J.K. and Liu, J. (2016) 'Modeling active virtual machines on IaaS clouds using an M/G/m/m+K queue', *Journal of IEEE Transactions on Services Computing*, vol. 9 Issue 3, pp. 408-420.

Outamazirt, A., Escheikh, M., Aïssani, D., Barkaoui, K. and Lekadir, O. (2016) 'On the Modeling and Performance Evaluation of Cloud Computing Centers Using M/G/c/c+r Queuing System', *Proceedings of the 10th International Workshop on VECoS 2016*, Tunis, Tunisia, pp. 77-84.

Xiong, K. and Perros, H. (2009) 'Service Performance and Analysis in Cloud Computing', *Proceedings of the 2009 Congress on Services*, Los Angeles, California, USA, pp. 693-700.

Yang, B., Tan F., Dai, Y. and Guo, S. (2009) 'Performance Evaluation of Cloud Service Considering Fault Recovery', *Proceedings of the First International Conference, CloudCom 2009*, Beijing, China, pp. 571-576.

Ma, B.N.W. and Mark, J.W. (1995) 'Approximation of the mean queue length of an M/G/c queuing system', *Journal of Operations Research*, Vol. 43 Issue 1, pp. 158165.

Takahashi, Y. (1977) 'An approximation formula for the mean waiting time of an M/G/c queue', *Journal of Operations Research Society of Japan*, Vol. 20 No. 3, pp. 150163.

Kimura, T. (1996) 'A Transform-Free Approximation for the Finite Capacity M/G/s Queue', *Journal of Operations Research*, vol. 44 No. 6, pp. 984-988.

Kimura, T. (1996) 'Optimal Buffer Design of an M/G/s Queue with Finite Capacity', *Journal of Communications in Statistics Stochastic Models*, vol. 12 Issue 1, pp. 165-180.

Smith, J.M. (2003) 'M/G/c/K Blocking Probability Models and System Performance', *Journal of Performance Evaluation*, vol. 52, pp. 237-267.

Takagi H. (1991) Queueing analysis: Vacation and Priority Systems, Vol. 1, North-Holland.

Heyman, D.F. and Sobel, M.J. (2004) Stochastic Models in Operations Research, vol. 1, Dover.