British Journal of Mathematics & Computer Science
4(10): 1301-1317, 2014

SCIENCEDOMAIN *international*
www.sciencedomain.org

SCIENCEDOMAIN

MMPP/G/m/m+r Queuing System Model to Analytically Evaluate Cloud Computing Center Performances

Fatima Oumellal[1], Mohamed Hanini[1,2*] and Abdelkrim Haqiq[1,2,3]

[1]FST, Hassan 1st University, Settat, Morocco.
[2]e-NGN research group, Africa and Middle East, ENSIAS Rabat, Morocco.
[3]IR2M laboratory, FST, Hassan 1st University, Settat, Morocco.

Original Research Article

Received: 14 December 2013
Accepted: 25 February 2014
Published: 14 March 2014

Abstract

In the last decades cloud computing has been the focus of a lot of research in both academic and industrial fields, however, implementation-related issues have been developed and have received more attention than performance analysis which is an important aspect of cloud computing and it is of crucial interest for both cloud providers and cloud users. Successful development of cloud computing paradigm necessitates accurate performance evaluation of cloud data centers. Because of the nature of cloud centers and the diversity of user requests, an exact modeling of cloud centers is not practicable; in this work we report an approximate analytical model based on an approximate Markov chain model for performance evaluation of a cloud computing center. Due to the nature of the cloud environment, we considered, based on queuing theory, a MMPP task arrivals, a general service time for requests as well as large number of physical servers and a finite capacity. This makes our model more flexible in terms of scalability and diversity of service time. We used this model in order to evaluate the performance analysis of cloud server farms and we solved it to obtain accurate estimation of the complete probability distribution of the request response time and other important performance indicators such as: the Mean number of Tasks in the System, the distribution of Waiting Time, the Probability of Immediate Service, the Blocking Probability and Buffer Size…

Keywords: Cloud computing, performance analysis, waiting and response times, queuing theory, embedded Markov chain, MMPP processes.

1 Introduction

In 1969, Leonard Kleinrock said: “As of now, computer networks are still in their infancy, but as they grow up and become sophisticated, we will probably see the spread of ‘computer utilities’ which, like present electric and telephone utilities, will service individual homes and offices across the country” [1]. Innovations are necessary to ride the inevitable tide of change. Most of enterprises are striving to reduce their computing cost through the means of virtualization. This demand of reducing the computing cost has led to the innovation of Cloud Computing.

**Corresponding author: haninimohamed@gmail.com;*

Cloud computing is a technology that uses the internet and central remote servers to maintain data and applications. This technology allows consumers and businesses to use applications without installation and access their personal files at any computer with internet access. It allows for much more efficient computing by centralizing storage, memory, processing and bandwidth.

The analogy of cloud computing is, "If you need milk, would you buy a cow?" All the users or consumers need is to get the benefits of using the software or hardware of the computer like sending emails etc. Just to get this benefit (milk) why should a consumer buy a (cow) software and/or hardware? [2].

There is no single definition for cloud computing. The US National Institute of Standards and Technology (NIST) defines cloud computing as "a model for user convenience, on-demand network access contributes the computing resources (e.g. networks, storage, applications, servers, and services) that can be rapidly implemented with minimal management effort or service provider interference" [3]. We can also define the cloud, or a "computing on demand" as a "new" computer model which provides IT services on-demand, affordable and accessible from anywhere at any time and by any user. Cloud Computing offers better computing through improved utilization and reduced administration and infrastructure costs.

Cloud Computing has become one of the most talked about technologies in recent times and has got lots of attention from media as well as analysts because of the opportunities it is offering.
Cloud Computing encompasses different types of services. The cloud has a service-oriented architecture, and there are three classes of technology capabilities that are being offered as a service: Infrastructure -as-a -Service (IaaS), where equipment such as hardware, storage, servers and network components are accessible via the Internet, the platform -as-a -Service (PaaS), which is a central component of the Cloud: the PaaS is responsible for developing applications for the cloud. It includes hardware with operating systems, virtualized servers, etc; and finally the Software-as- a-Service (SaaS) (resources software), which includes applications and other hosted services [4].

Due to benefits offered by Cloud computing, Quality of Service (QoS) is a broad topic in this technology and is most often referred as the mechanisms in place to guarantee a certain level of performance and availability of a service [5]. QoS includes availability, throughput, reliability, security, and many other parameters, but also performance indicators such as response time, task blocking probability, probability of immediate service, and mean number of tasks in the system [6]. To model networks and estimate its QoS parameters, the queuing theory models are classic and powerful tools [7].

In this paper, and in order to analyze QoS in cloud computing, we model a cloud center as a MMPP/G/m/m+r queuing system which indicates that the arrival process is a Markov Modulated Poisson Process (MMPP), while task service times are independent and identically distributed random variables that follow a general distribution. The system under consideration contains m servers which render service in a first in first out (FIFO) manner. Because of the nature of the cloud environment, our mathematical assumptions make the proposed model more flexible in terms of conformance with reality, scalability and diversity of service time. Due to the introduction of the finite capacity, the system may experience blocking of task requests. In other words, the necessary assumptions that are required for a true performance model of a center of cloud computing are integrated.

We used this model in order to evaluate the performance analysis of cloud server farms in order to get the complete probability distribution of the request response time and other important performance indicators such as: the mean number of tasks in the system, the distribution of waiting time, the probability of immediate service, the blocking probability and buffer size.

The remainder of the paper is organized as follows: In section 2 we give a brief overview of related work on cloud performance evaluation and performance characterization of queuing systems. Section 3 discusses our analytical model in details. In Section 4, we solve our model in order to obtain analytically desired performance metrics. Our findings are summarized in Section 5.

2 Related Work

Cloud computing is the subject of several researches, but only few are the works that are interested in QoS problems in this technology, and rigorous analytical approach has been adopted by only a handful among them.

Xiong et al. [8] suggest a queuing network based model for analyzing the working level and throughput of cloud resources, and Laplace transform is used to determine the response time distribution with particular reference to resource utilization in a cloud.

Yang et al. [9] modeled the cloud center as a M/M/m/m+r queuing system from which the distribution of response time was determined. Inter-arrival and service times were both assumed to be exponentially distributed, and the system had a finite buffer of size m+r. The response time was decomposed into three independent periods: waiting period, service period, and execution period.

A large queuing system (number of servers is large, e.g., more than 100) that is assumed to have Poisson arrival process and generally distributed service time can be a proper model for performance evaluation of a cloud center. However analyzing queuing system with generally distributed service time is not an easy task.

Theoretical analyses have mostly relied on extensive research in performance evaluation of M/G/m queuing systems, as outlined in [10-15]. As solutions for mean response time and queue length in M/G/m systems cannot be obtained in closed form, suitable approximations were sought.

However, most of the studies using M/G/m queuing system provide reasonably accurate estimations of the performances only when number of servers is comparatively small, (say, less than ten or so), but fail for large number of servers [16-19].

An approximate solution for steady-state queue length distribution in a M/G/m system with finite waiting space was described in [20].

A similar approach in the context of M/G/m queues, 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, was described in [21]. An interesting finding is that the optimal buffer size depends on the order of convexity for the service time; the higher this order is, the larger the buffer size should be.

An approximation for the average queuing delay in a M/G/m/m+r queue, based on the relationship of joint distribution of remaining service time to the equilibrium service distribution, was proposed in [13].

Recently Khazaei et al. [22] modeled the cloud center as a M/G/m/m+r queuing system with single task arrivals and a task buffer of finite capacity. They evaluated its performance using a combination of a transform-based analytical model and an approximate Markov chain model, which allows them to obtain a complete probability distribution of response time and number of tasks in the system. They also discussed the probability of immediate service (i.e., no waiting in the input buffer) and blocking probability, and determine the size of the buffer needed for the blocking probability to remain below a predefined value. Analytical results are validated through discrete-event simulation.

However, in cloud computing and because of the diversity of cloud users, analysis in the case where inter-arrival is not exponential is a more relevant model, but its analysis is more complex. In [23] authors developed and employed a cloud computing model for allocation of resources to the jobs that enter into the cloud by using queuing models. They considered that arrival of jobs follows non-homogeneous Poisson process; the service time is assumed to be exponential. They studied the transient analysis of the model by using various performance measures such as Mean number of job requests, Utilization, Throughput and Mean Delay.

In this work we report an approximate analytical model based on an approximate Markov chain model for performance evaluation of a cloud computing center. Due to the nature of the cloud environment, especially the nature of arrivals, we consider, based on queuing theory, a MMPP task arrivals which is more realistic in modeling networks, a general service time for requests as well as large number of physical servers and a finite capacity. This makes our model more flexible in term of depiction of cloud computing reality.

3 Proposed Analytical Model

3.1 Arrivals and Service Time in a MMPP/G/m/m+r Queuing System

Cloud computing technology is designed to support a wide range of users. One major feature is that the arrivals are usually highly bursty, and there is usually a correlation between arrivals. Thus, the traditional Poisson traffic arrival model cannot be applied because the presence of correlation between arrivals violates the independence assumption associated with the Poisson processes. Markov modulated Poisson Process (MMPP), that is a subclass of the doubly stochastic Poisson processes, can be used to model time-varying arrival rates and important correlations between inter-arrival times [24]. Despite these abilities MMPP processes are still tractable by analytical methods [25]. Hence the opportunities using of MMPP processes for the modeling of arrivals in cloud computing is considered in this paper.

In MMPP model, the arrivals follow a Poisson process with an arrival rate $\lambda\left[J(t)\right]$ where $J(t)$ is an irreducible Markov process with finite states. In other words, the intensity of the Poisson process depends only of the Markov process state. When the Markov chain is in state i the arrival rate is λ_i [26].

In general, a m-state MMPP (MMPP-m) process is parameterized by the infinitesimal generator Q of the Markov process and the m Poisson arrival rates: $\lambda_1, \lambda_{2,} \ldots \lambda_m$, with:

$$Q = \begin{pmatrix} -\sigma_1 & \cdots & \sigma_{1m} \\ \vdots & \ddots & \vdots \\ \sigma_{m1} & \cdots & -\sigma_m \end{pmatrix} where \;\; \sigma_i = \sum_{j=1}^{m} \sigma_{ij}$$

$\frac{\sigma_{ij}}{\sigma_i}$ represents the rate of the transition between the states i and j for the Markov process.

The steady-state probability distribution π can be gained by the solution of the linear equation $\pi^t Q = 0$ when the additional normalization condition $\sum_{i=1}^{m} \pi(i) = 1$ is imposed.

In this work, we model a center of cloud computing using $MMPP/G/m/m+r$ queuing system in which the time of arrival follows a MMPP process, service time generally distributed with a mean value μ , the system contains m servers and had a finite capacity of size m + r (buffer size is r).

We can analyze this queue by exploiting the embedded Markov chain technique discussed in the next.

The task arrival in the system follows a MMPP process, the distribution function of the inter-arrival time A is given by: $A(x) = p[A < x]$, its density function is denoted by $a(x)$ and its Laplace transform is:

$$A^*(s) = \int_0^\infty e^{-sx} a(x)\,dx$$

The service time of tasks B is identically and independently distributed according to a general distribution with an average service time $\bar{b} = \frac{1}{\mu}$, its distribution function is given by $B(x) = p[B < x]$, its density function is denoted by: $b(x)$ and the Laplace transform of the service time is

$$B^*(s) = \int_0^\infty e^{-sx} b(x)\,dx .$$

Residual task service time is the time from a random point in task execution until task completion. We will denote it as B+. This time is necessary for our model since it represents time distribution between a MMPP task arrival and departure of the task which was in service when MMPP task arrival occurred. For a Poisson process it can be shown that the probability distribution of residual and elapsed service times have the same probability distribution (where elapsed service time B- is the time between start of the task execution and next arrival of task request) [27]. The probability distribution of Laplace transform of residual and elapsed task service times is calculated in [27] as:

$$B_+^*(s) = B_-^*(s) = \frac{1 - B^*(s)}{s\bar{b}} .$$

3.2 Embedded Markov Chain

MMPP/G/m/m+r queuing system may be considered as a non-Markov process which can be analyzed by applying the embedded Markov chain technique which requires selection of Markov points in which the state of the system is observed. Therefore we model the number of the tasks in the system (both in service and queued) at the moments immediately before the task request arrival. This Markov chain is ergodic because it is irreducible (that is to say all states communicate with each other) and has a finite number of states, so the chain is recurrent non-nil, hence it is ergodic; thus, this chain admits a steady state probability. To get the latter we need to calculate the departure probabilities in the system:

$$p_x, p_y \ \ and \ \ p_{z,k} .$$

The arrival times of tasks are selected as points of Markov Fig. 1. Note that the number of departures may be anywhere between 0 and ∞. When the system is in the steady state, there will be on the average a single departure between every two successive arrivals.

Let A_n and A_{n+1} be the moments of nth and (n+1)th arrivals to the system respectively, while q_n and q_{n+1} indicate the number of tasks found in the system immediately before these arrivals. If v_{n+1} indicates the number of tasks which depart from the system between A_n and A_{n+1}, then we need to calculate the transition probabilities associated with the embedded Markov chain defined as: $q_{n+1} = q_n - v_{n+1} + 1$. We need to calculate the transition probabilities associated with the embedded Markov chain defined as: $p_{ij}^{s} = p\left[q_{n+1} = j \mid q_n = i\right]$which is the probability that i +1- j tasks are served during the inter-arrival time between the arrivals of two successive tasks. It is obvious that for $j > i+1,\ p_{ij}^{s} = 0.$

Since the Markov chain is ergodic, a steady state probability distribution exists $\pi^s = \left[\pi_0^s, \pi_1^s, \pi_2^s, ..., \pi_{m+r}^s\right]$ with $\pi_k^s = \lim\limits_{n\to+\infty} p\left[q_n = k\right]$, $0 \le k \le m+r$ which is the probability that k tasks exist in the system immediately before the new arrival. To obtain this distribution it is required to solve the equation $\pi^s = \pi^s P^s$, where P^s is the matrix whose elements are the transition probabilities p_{ij}^{s} .

To find the elements of the transition probability matrix P^s , we need to count the number of tasks departing from the system between two successive arrivals, as shown in Fig.2.

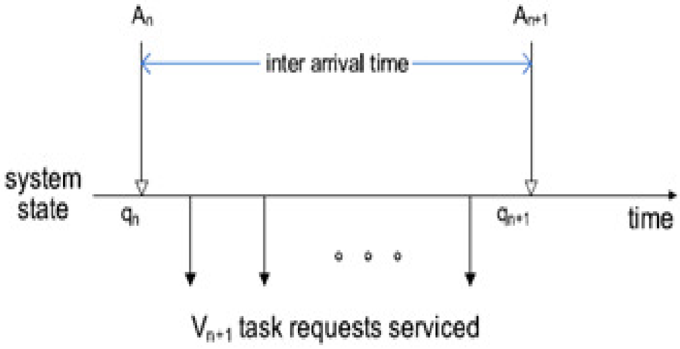

Fig. 1. Embedded Markov chain

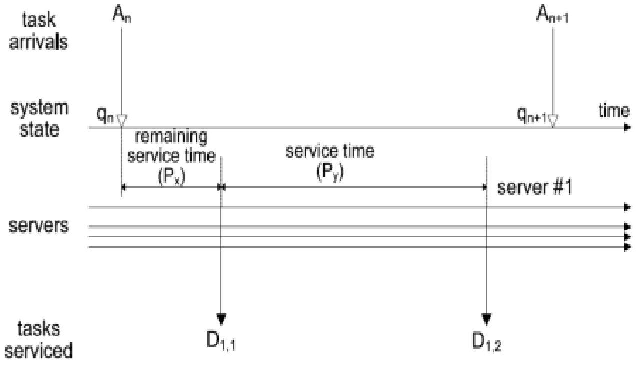

Fig. 2. System behavior between two MMPP arrivals

3.3 Departure Probabilities

First consider the case of a two-state process MMPP (MMPP-2), P is the transition matrix of the Markov chain associated with the MMPP-2 defined as follows:

$$P = \begin{pmatrix} p_{11} & p_{12} \\ p_{21} & p_{22} \end{pmatrix}$$

λ is the arrival rate of the MMPP-2: $\lambda = (\lambda_1, \lambda_2)$. For a task to be served and leaves the system during the inter-arrival time, its remaining duration (residual service time B+) must be shorter than the task inter-arrival time, which results in the probability p_x. This probability can be calculated as:

Let $t_{n=} A_n$ *and* $t_{n+1=} A_{n+1}$

$$
\begin{aligned}
p_x &= p\left[B_+ < A\right] \\
&= p\left[B_+ < A \middle/ J(t_n)=\lambda_1, J(t_{n+1})=\lambda_1\right] \times p\left[J(t_n)=\lambda_1, J(t_{n+1})=\lambda_1\right] \\
&\quad + p\left[B_+ < A \middle/ J(t_n)=\lambda_1, J(t_{n+1})=\lambda_2\right] \times p\left[J(t_n)=\lambda_1, J(t_{n+1})=\lambda_2\right] \\
&\quad + p\left[B_+ < A \middle/ J(t_n)=\lambda_2, J(t_{n+1})=\lambda_1\right] \times p\left[J(t_n)=\lambda_2, J(t_{n+1})=\lambda_1\right] \\
&\quad + p\left[B_+ < A \middle/ J(t_n)=\lambda_2, J(t_{n+1})=\lambda_2\right] \times p\left[J(t_n)=\lambda_2, J(t_{n+1})=\lambda_2\right] \\
&= p_{11} p\left[J(t_n)=\lambda_1\right] \int_0^{+\infty} p\left[B_+ < A \middle/ J(t_n)=\lambda_1, J(t_{n+1})=\lambda_1, B_+ = x\right] dB_+(x) \\
&\quad + p_{12} p\left[J(t_n)=\lambda_1\right] \int_0^{+\infty} p\left[B_+ < A \middle/ J(t_n)=\lambda_1, J(t_{n+1})=\lambda_2, B_+ = x\right] dB_+(x) \\
&\quad + p_{21} p\left[J(t_n)=\lambda_2\right] \int_0^{+\infty} p\left[B_+ < A \middle/ J(t_n)=\lambda_2, J(t_{n+1})=\lambda_1, B_+ = x\right] dB_+(x) \\
&\quad + p_{22} p\left[J(t_n)=\lambda_2\right] \int_0^{+\infty} p\left[B_+ < A \middle/ J(t_n)=\lambda_2, J(t_{n+1})=\lambda_2, B_+ = x\right] dB_+(x)
\end{aligned}
$$

In the stationary state $p\left[J(t_n)=\lambda_k\right]$ will be time independent and it is equal to π_k : the k^{th} component of MMPP-2 steady-state probability is defined by:

$$
\begin{cases}
\pi_1 = \dfrac{P_{21}}{P_{21}+P_{12}} \\
\pi_2 = \dfrac{P_{12}}{P_{21}+P_{12}}
\end{cases}
$$

Hence:

$$
\begin{aligned}
p_x &= p_{11}\pi_1 \int_0^{+\infty} e^{-\lambda_1 x} dB_+(x) + p_{12}\pi_1 \int_0^{+\infty} e^{-\lambda_2 x} dB_+(x) + p_{21}\pi_2 \int_0^{+\infty} e^{-\lambda_1 x} dB_+(x) + p_{22}\pi_2 \int_0^{+\infty} e^{-\lambda_2 x} dB_+(x) \\
&= p_{11}\pi_1 B_+^*(\lambda_1) + p_{12}\pi_1 B_+^*(\lambda_2) + p_{21}\pi_2 B_+^*(\lambda_1) + p_{22}\pi_2 B_+^*(\lambda_2)
\end{aligned}
$$

Fig. 3 and Fig. 4 show the evolution of p_x and p_y as a function of λ_1 and λ_2. The two figures illustrate the decrease of p_x and p_y when λ_1 and λ_2 increase.

$p_x = p\left[B_+ < A\right] = \sum_{i=1}^{N}\sum_{j=1}^{N} p_{ij}\pi_i^N B_+^*(\lambda_j)$ Where π^N is the steady-state probability of MMPP-N

Physically this result presents the probability of no task arrivals during residual task service time.

In the case when arriving task can be accommodated immediately by an idle server (and therefore queue length is zero) we have to evaluate the probability that such task will depart before next task arrival.

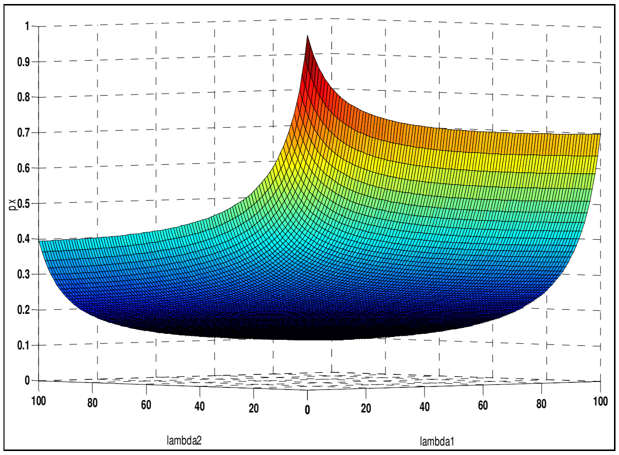

Fig. 3. Tridimensional view of the evolution of p_x as a function of λ_1 and λ_2

We will denote this probability as p_y and it is defined for MMPP-2, as follows:

$$p_y = p\left[B < A\right]$$

Using the same analysis in computing p_x we can show that:

$$p_y = p_{11}\pi_1 B^*(\lambda_1) + p_{12}\pi_1 B^*(\lambda_2) + p_{21}\pi_2 B^*(\lambda_1) + p_{22}\pi_2 B^*(\lambda_2)$$

Similarly, we can give the general expression for MMPP-N as follows:

$$p_y = p\left[B < A\right] = \sum_{i=1}^{N}\sum_{j=1}^{N} \pi_i^N p_{ij} B^*(\lambda_j)$$

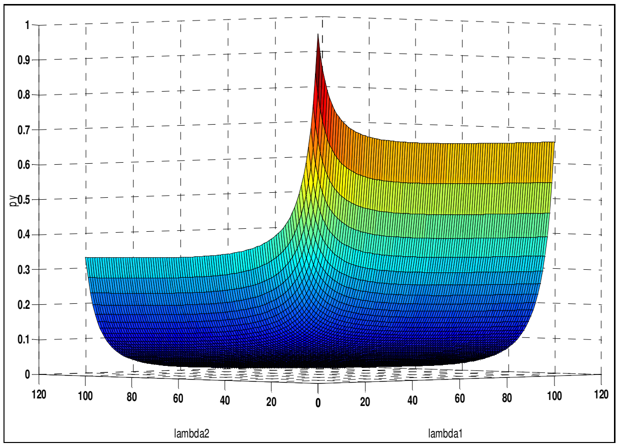

Fig. 4. Tridimensional view of the evolution of p_y as a function of λ_1 and λ_2

For the case of a MMPP-N we can generalize this probability in the following.

The probability that k tasks depart from a server before the arrival of a new task is derived from the two previous expressions p_x and p_y.

Let A and B be two events such as:

A: "The task in service is complete and leaves the system during the inter-arrival"
B: "The task which is waiting enters the service, completes its service and leaves the system during the inter-arrival"

$$\begin{aligned} p_{z,k} &= p\left[A \cap B^{(k-1)}\right] \\ &= p(A) \times (p(B))^{(k-1)} \\ &= p_x \times (p_y)^{(k-1)} \\ &= \left[p_{11}\pi_1 B_+^*(\lambda_1) + p_{12}\pi_1 B_+^*(\lambda_2) + p_{21}\pi_2 B_+^*(\lambda_1) + p_{22}\pi_2 B_+^*(\lambda_2)\right] \times \\ &\quad \left[p_{11}\pi_1 B^*(\lambda_1) + p_{12}\pi_1 B^*(\lambda_2) + p_{21}\pi_2 B^*(\lambda_1) + p_{22}\pi_2 B^*(\lambda_2)\right]^{(k-1)} \end{aligned}$$

The general expression for this probability is given by:

$$p_{z,k} = \left[\sum_{i=1}^{N}\sum_{j=1}^{N} p_{ij}\pi_i^N B_+^*(\lambda_j)\right] \times \left[\sum_{i=1}^{N}\sum_{j=1}^{N} \pi_i^N p_{ij} B^*(\lambda_j)\right]^{(k-1)} \quad \text{with } p_{z,1} = p_x \; .$$

Using these values we can compute the transition probabilities matrix.

Fig. 5 shows the evolution of $p_{z,k}$ as a function of λ_1 and λ_2. This figure shows the decrease of $p_{z,k}$ when increasing λ_1 and λ_2.

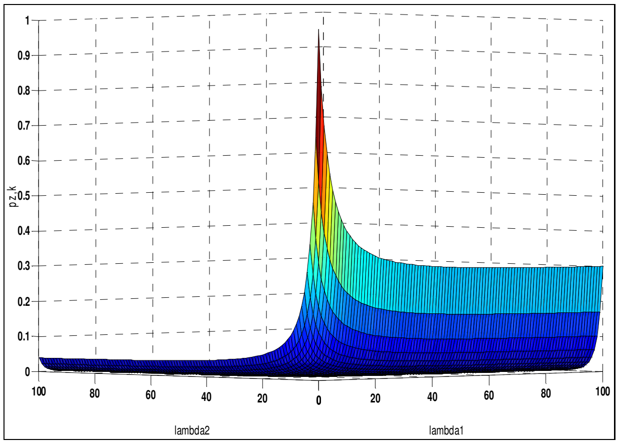

Fig. 5. Evolution of $p_{z,k}$ (for k =3) as a function of λ_1 and λ_2

3.4 Transition Matrix

After calculating the departure probabilities p_x, p_y *and* $p_{z,k}$, in the embedded Markov chain, we may identify four different regions of operation for which different conditions hold; these regions are schematically shown in Fig. 6, where the numbers on horizontal and vertical axes correspond to the number of tasks in the system immediately before a task request arrival (i) and immediately upon the next task request arrival (j), respectively.

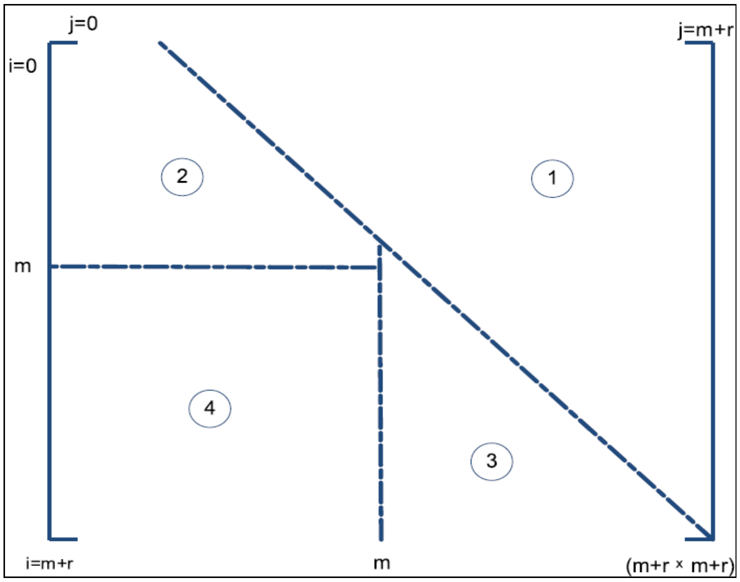

Fig. 6. Range of validity for p_{ij}^s

1 – For $i+1<j, \quad p_{ij}^s=0$

2 – For $i<m$ and $j\leq m$ (no waiting), between two successive arrivals the probability that $i-j+1$ tasks are served is:

$$p_{ij}^s = C_{i-j}^{i} p_x^{i-j}(1-p_x)^j p_y + C_{i+1-j}^{i} p_x^{i+1-j}(1-p_x)^{j-1}(1-p_y)$$

3 – For $i\geq m \, and \, j\geq m$, i.e. all servers are busy during the inter-arrival time. Let $w=i+1-j$ represents the number of tasks that leave the system between two successive Markov points. This number can be between 0 and infinity, but it is often close to 1. In this model it is assumed that w does not exceed 3 i.e. there are no more than three tasks served between two successive arrivals.

$$p_{ij}^s = \sum_{s_1=\min(w,1)}^{\min(w,m)} C_m^{s_1} p_x^{s_1}(1-p_x)^{m-s_1} \times \sum_{s_2=\min(w-s_1,1)}^{\min(w-s_1,s_1)} [C_{s_1}^{s_2} p_{z,2}^{s_2}(1-p_{z,2})^{s_1-s_2} \times C_{s_2}^{w-s_1-s_2} p_{z,3}^{w-s_1-s_2}(1-p_{z,3})^{2s_2-w+s_1}]$$

4 - Finally for $i \geq m \, and \, j < m$, i.e. all the servers are busy at the time of first arrival and $i - m$ tasks are in the queue, while after the arrival of the next task, there are exactly j tasks in the system, no one will be in the queue. The transition probability is expressed by:

$$p_{ij}^{s} = \sum_{s_1=\mathrm{m}-j}^{\min(w,m)} C_{m}^{s_1} p_{x}^{s_1} (1-p_x)^{m-s_1} \times \sum_{s_2=\min(w-s_1,m-j)}^{\min(w-s_1,s_1)} [C_{s_1}^{s_2} p_{z,2}^{s_2} (1-p_{z,2})^{s_1-s_2} \times C_{s_2}^{w-s_1-s_2} p_{z,3}^{w-s_1-s_2} (1-p_{z,3})^{2s_2-w+s_1}]$$

4 Analytical Performances Evaluation at Steady State

4.1 Equilibrium Balance Equations

Once we find the matrix P, the steady state probabilities can be computed by solving the system of the balance equations (the average flow outgoing of each state is equal to the average flow go into this state) and adding the normalization equation (the sum of all state probabilities equal to 1).

The balance equations are:

$$\pi_i^s = \sum_{j=0}^{m+r} \pi_j^s p_{ji}^s \quad 0 \leq i \leq m + r$$ to which we add the normalization equation $\sum_{i=0}^{m+r} \pi_i^s = 1$.

But it is still difficult to solve the equation of balance of the steady state and therefore a numerical solution is required.

4.2 Distribution of the Number of Tasks in the System

Once the steady-state probabilities are found, we can establish the generating function of the number of tasks $\prod(z) = \sum_{k=0}^{m+r} \pi_k^s z^k$, and therefore we can deduce the average number of tasks in the system by deriving the above expression, we thus find $\overline{p} = \prod'(1)$

4.3 Distribution of Waiting and Response Times

Using Little's Law, the response time average is given by:

$$\bar{t} = \frac{\overline{p}}{\overline{\lambda}(1-\pi_{m+r}^s)} = \frac{\overline{p}}{(\pi * \lambda^T)(1-\pi_{m+r}^s)}$$ Where $\overline{\lambda}$ is the intensity average of the MMPP process.

Let W denotes the waiting time in the steady state, W(x) the distribution function and W*(x) its Laplace transform. It has been demonstrated in [28] that the length of the queue Q has the same distribution as W and therefore the number of tasks that arrive during the waiting time is expressed as:

$$Q(z) = W^*(\lambda(1-z)) \ .$$

For MMPP-2 process we can calculate the distribution Q as a function of W distribution as follows.

Let U_{n+1} denotes the number of task arrivals during the waiting time, we have:

$$\begin{aligned}
Q(z) &= E(z^{U_{n+1}}) \\
&= \sum_{k=0}^{+\infty} \mathrm{p}(U_{n+1}=k)z^{k} \\
&= \sum_{k=0}^{+\infty} \left[p(U_{n+1}=k,J_t=1)+p(U_{n+1}=k,J_t=2)\right]z^{k} \\
&= \sum_{k=0}^{+\infty} \left[p(U_{n+1}=k \mid J_t=1)p(J_t=1)+p(U_{n+1}=k \mid J_t=2)p(J_t=2)\right]z^{k} \\
&= \sum_{k=0}^{+\infty} \left[\pi_1 \int_0^{+\infty} \frac{(\lambda_1 t)^k}{k!} e^{-\lambda_1 t} dw(t) + \pi_2 \int_0^{+\infty} \frac{(\lambda_2 t)^k}{k!} e^{-\lambda_2 t} dw(t)\right] z^{k} \\
&= \pi_1 \int_0^{+\infty} \sum_{k=0}^{+\infty} \frac{(\lambda_1 z t)^k}{k!} e^{-\lambda_1 t} dw(t) + \pi_2 \int_0^{+\infty} \sum_{k=0}^{+\infty} \frac{(\lambda_2 z t)^k}{k!} e^{-\lambda_2 t} dw(t) \\
&= \pi_1 \int_0^{+\infty} e^{z\lambda_1 t}.e^{-\lambda_1 t} dw(t) + \pi_2 \int_0^{+\infty} e^{z\lambda_2 t}.e^{-\lambda_2 t} dw(t) \\
&= \pi_1 \int_0^{+\infty} e^{-(1-z)\lambda_1 t} dw(t) + \pi_2 \int_0^{+\infty} e^{-(1-z)\lambda_2 t} dw(t) \\
&= \pi_1 w^{*}(\lambda_1(1-z)) + \pi_2 w^{*}(\lambda_2(1-z))
\end{aligned}$$

with $\pi^t P = \pi; \pi_1 + \pi_2 = 1$

We can easily generalize this expression for MMPP-N as follows:

$$\begin{aligned}
E(z^{U_{n+1}}) &= \sum_{k=0}^{+\infty} p(U_{n+1}=k)z^{k} \\
&= \sum_{k=0}^{N} \pi_k w^{*}(\lambda_k(1-z)) \qquad \text{with}: \pi^t P = \pi \ \ and \ \ \sum_{k=0}^{N} \pi_k = 1
\end{aligned}$$

Q(z) could also be written as follows:

$$Q(z) = \sum_{k=0}^{m-1} \pi_k^s + \sum_{k=m}^{m+r} \pi_k^s z^{k-m}$$

Proof :

Let w_q the number of waiting tasks in the system.

$$
\begin{aligned}
Q(z) &= \sum_{k=0}^{r} p(w_q = k) z^k \\
&= p(w_q = 0) + \sum_{k=1}^{r} p(w_q = k) z^k \\
&= \sum_{k=0}^{m} \pi_k^s + \sum_{k=1}^{r} \pi_{m+k}^s z^k \\
&= \pi_m^s + \sum_{k=0}^{m-1} \pi_k^s + \sum_{k'=m+1}^{m+r} \pi_{k'}^s z^{k'-m} \quad \text{with:}\, k = k' - m \\
&= \pi_m^s + \sum_{k=0}^{m-1} \pi_k^s + \sum_{k=m+1}^{m+r} \pi_k^s z^{k-m} \\
&= \sum_{k=0}^{m-1} \pi_k^s + \sum_{k=m}^{m+r} \pi_k^s z^{k-m}
\end{aligned}
$$

As we have a finite capacity system (i.e., there may exist blocking), we shall use effective arrival rate as:

$$\lambda_e = \lambda^t.\pi(1 - \pi_{m+r}^s)$$

Hence we have:

$$W^*(s) = Q(z)\,|_{z=1-\frac{s}{\lambda_e}} = Q(1 - \frac{s}{\lambda_e}) = \mathrm{Q}(1 - \frac{s}{\lambda^t.\pi(1 - \pi_{m+r}^s)})$$

Moreover, the Laplace transform of response time is: $T^*(s) = W^*(s)B^*(s)$, in which the W*(s) and B*(s) are the Laplace transform of waiting time and the service time, respectively.

Proof:

$$
\begin{aligned}
T^*(s) &= \int_0^{+\infty} e^{-st} dT(t) \\
&= E(e^{-st}) \\
&= E(e^{-s(w+b)}) \quad \text{because } T = W + B \\
&= E(e^{-sw}) * E(e^{-sb}) \text{ as W and B are indépendant} \\
&= W^*(s) * B^*(s)
\end{aligned}
$$

The ith central moment, t(i), of the response time distribution is given by:

$$
\begin{aligned}
T^*(z) &= \int_0^{+\infty} e^{-zt} f_T(t) dt \\
&= E(e^{-zT})
\end{aligned}
$$

$$\text{Then : } T^*{}'(z) = -E(Te^{-zT})$$

$$\text{This leads to : } T^{*(i)}(z) = (-1)^i E(T^i e^{-zT})$$

$$E(T^i) = (-1)^i T^{*(i)}(0)$$

4.4 Probability of Immediate Service

Here we are interested in the probability that tasks will get into service immediately upon arrival, without any queuing, so there is at least an idle server and thus the response time would be equal to the service time:

$$P_{nq} = \sum_{i=0}^{m-1} \pi_i^s$$

4.5 Blocking Probability

Since MMPP arrivals are independent of buffer state and the distribution of number of tasks in the system was obtained, we are able to directly calculate the blocking probability of a system with buffer size r:

$$Pb_r = \pi_{m+r}^s$$

4 Conclusion

In this paper, we proposed an approximate model based on a Markov chain to evaluate the performance of a center of cloud computing using the queue MMPP/G/m/m+r. Due to the nature of the environment of cloud computing and the diverse needs and demands of users, we considered a MMPP arrival process that reflects the nature of arrivals in the cloud, a general service time, a number of servers and a finite buffer capacity. We described this new analytical approximation for performance evaluation of a center of cloud computing and resolved it to get a very decent estimate. In this proposed model we calculated analytically the performance indicators such as the average number of tasks in the system, blocking probability, probability of immediate service and the average of response time.

Competing Interests

Authors have declared that no competing interests exist.

References

[1] Kleinrock L. A vision for the internet. ST Journal of Research. 2005;2.

[2] Abhishek Goel, Shikha Goel. Security Issues in cloud computing. International Journal of Application or Innovation in Engineering and Management. 2012;1(4).

[3] Peter Mell, Tim Grance. Draft NIST working definition of cloud computing. 2009. Available: http://csrc.nist.gov/groups/SNS/cloud-computing.

[4] Furht B. Cloud computing fundamentals. Handbook of cloud computing, Springer; 2010.

[5] Armstrong D, Djemame K. Towards quality of service in the cloud. Proceedings of the 25th UK performance engineering workshop, Leeds, UK; 2009.

[6] Wang L, Von Laszewski, Younge A, He X, Kunze M, Tao J, Fu C. Cloud computing: A perspective study. New Generation Computing. 2010;28.

[7] Kleinrock L. Queueing systems: Theory, Wiley-Interscience. 1975;1.

[8] Xiong K, Perros H. Service performance and analysis in cloud computing. IEEE 2009 World Conference on Services; 2009.

[9] Yang B, Tan F, Dai Y, Guo S. Performance evaluation of cloud service considering fault recovery. First Int'l Conference on Cloud Computing; 2009.

[10] Hokstad P. Approximations for the M/G/m queues. Operations Research. 1978;26.

[11] Ma BNW, Mark JW. Approximation of the mean queue length of an M/G/c queueing system. Operations Research. 1998;43.

[12] Miyazawa M. Approximation of the queue-length distribution of an M/GI/s queue by the basic equations. Journal of Applied Probability. 1989;23.

[13] Nozaki SA, Ross SM. Approximations in finite-capacity multi-server queues with poisson arrivals. Journal of Applied Probability.1978;15.

[14] Page E. Tables of waiting times for M/M/n, M/D/n and D/M/n and their use to give approximate waiting times in more general queues. J. Operational Research Society. 1982;33.

[15] Yao DD. Refining the diffusion approximation for the M/G/m queue. Operations Research. 1985;33.

[16] Boxma OJ, Cohen JW, and Huffel N. Approximations of the mean waiting time in an M/G/s queueing system. Operations Research. 1979;27.

[17] Kimura T. Diffusion approximation for an M/G/m queue. Operations Research. 1983;31.

[18] Takahashi Y. An approximation formula for the mean waiting time of an M/G/c queue. J Operational Research Society. 1977;20.

[19] Tijms HC, Hoorn MH, Federgru A. Approximations for the steady-state probabilities in the M/G/c queue. Advances in Applied Probability. 1981;13.

[20] Kimura T. A transform-free approximation for the finite capacity M/G/s queue. Operations Research. 1996;44(6).

[21] Kimura T. Optimal buffer design of an M/G/s queue with finite capacity. Communications in Statistics and Stochastic Models. 1996;12(6),

[22] Khazaei H, Misic J, Misic VB. Performance analysis of cloud computing centers using M/G/m/m+r queuing systems. IEEE Transactions on parallel and distributed systems. 2012;23(5).

[23] Satyanarayana A, Suresh Varma P, Rama Sundari MV, Sarada Varma P. Performance analysis of cloud computing under non homogeneous conditions. International Journal of Advanced Research in Computer Science and Software Engineering. 2013;3(5).

[24] Jelena revzina. Possibilities of MMPP processes for bursty traffic analysis. Proceedings of the 10th international conference "reliability and statistics in transportation and communication" held in riga, Latvia; 2010.

[25] Bolch G, Greiner S, Meer H, Trivedi KS. Queueing networks and Markov chains: Modeling and performance evaluation with computer science applications. John Wiley and Sons; 2006.

[26] Neuts F. Models based on the Markovian arrival process. IEICE Trans. Commun. 1992;E75-B(12).

[27] Takagi H. Queueing analysis. Vacation and priority systems. North- Holland; 1991;1.

[28] Marshall KT, Wolff RW. Customer average and time average queue lengths and waiting times. J Applied Probability. 1971;8.

Peer-review history:
The peer review history for this paper can be accessed here (Please copy paste the total link in your browser address bar)
www.sciencedomain.org/review-history.php?iid=466&id=6&aid=3991