

Original Research Article

MMPP/G/m/m+r Queuing System Model to analytically evaluate Cloud Computing Center Performances

Fatima Ou-mellal¹, Mohamed Hanini^{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

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.

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.

* Tel.: +212662117066

E-mail address: haninimohamed@gmail.com

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

37 There is no single definition for cloud computing. The US National Institute of Standards and
38 Technology (NIST) defines cloud computing as "a model for user convenience, on-demand
39 network access contributes the computing resources (e.g. networks, storage, applications,
40 servers, and services) that can be rapidly implemented with minimal management effort or
41 service provider interference" [3]. We can also define the cloud, or a "computing on
42 demand" as a "new" computer model which provides IT services on-demand, affordable and
43 accessible from anywhere at any time and by any user. Cloud Computing offers better
44 computing through improved utilization and reduced administration and infrastructure costs.
45 Cloud Computing has become one of the most talked about technologies in recent times and
46 has got lots of attention from media as well as analysts because of the opportunities it is
47 offering.

48 Cloud Computing encompasses different types of services. The cloud has a service-oriented
49 architecture, and there are three classes of technology capabilities that are being offered as
50 a service: Infrastructure -as-a -Service (IaaS), where equipment such as hardware, storage,
51 servers and network components are accessible via the Internet, the platform -as-a -Service
52 (PaaS), which is a central component of the Cloud: the PaaS is responsible for developing
53 applications for the cloud. It includes hardware with operating systems, virtualized servers,
54 etc; and finally the Software-as- a-Service (SaaS) (resources software), which includes
55 applications and other hosted services [4].

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

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

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

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

2. RELATED WORK

Cloud computing is the subject of several research, 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 transformation 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, 11, 12, 13, 14, 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.

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].

However, most of these 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, 17, 18, 19].

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 are 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 $\lambda[J(t)]$ 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: $\lambda_1, \lambda_2, \dots, \lambda_m$, with:

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

$\frac{-\sigma_{ij}}{\sigma_i}$ Represent 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: $A(x) = \text{prob}[A < x]$, its density function is denoted by $a(x)$ and

its Laplace transform is: $A^*(s) = \int_0^\infty e^{-sx} a(x) dx$

172 The service time of tasks B are identically and independently distributed according to a
 173 general distribution with an average service time $\bar{b} = \frac{1}{\mu}$, its distribution function is given by

174 $B(x) = \text{prob}[B < x]$, its density function is denoted by: $b(x)$ and the Laplace transform of
 175 the service time is $B^*(s) = \int_0^\infty e^{-sx} b(x) dx$.

176 Residual task service time is the time from a random point in task execution until task
 177 completion. We will denote it as B_+ . This time is necessary for our model since it represents
 178 time distribution between a MMPP task arrival and departure of the task which was in
 179 service when task arrival occurred. It can be shown as well that probability distribution of
 180 elapsed service time (between start of the task execution and next arrival of task request B-)
 181 has the same probability distribution [27]. The Laplace transform of residual and elapsed
 182 task service times have the same probability distribution as it was calculated in [27] as

$$183 \quad B_+^*(s) = B_-^*(s) = \frac{1 - B^*(s)}{s\bar{b}}.$$

184 Results should be clearly described in a concise manner. Results for different parameters
 185 should be described under subheadings or in separate paragraph. Table or figure numbers
 186 should be mentioned in parentheses for better understanding.

187
 188 The discussion should not repeat the results, but provide detailed interpretation of data. This
 189 should interpret the significance of the findings of the work. Citations should be given in
 190 support of the findings. The results and discussion part can also be described as separate, if
 191 appropriate.

192 3.2 Embedded Markov Chain

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

201 P_x, P_y and $P_{z,k}$.

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

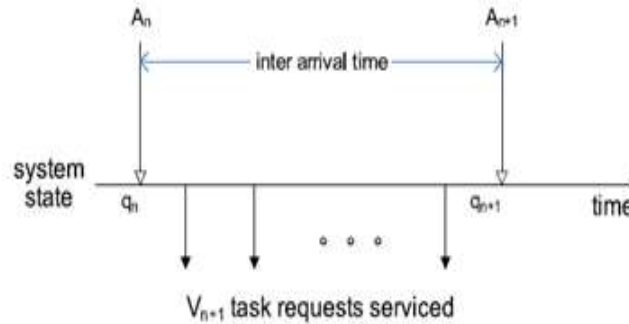


Fig. 1. Embedded Markov chain

207
 208 Let A_n and A_{n+1} the moments of n th and $(n+1)$ th arrivals to the system respectively, while q_n
 209 and q_{n+1} indicate the number of tasks found in the system immediately before these arrivals.
 210 If v_{n+1} indicates the number of tasks which depart from the system between A_n and A_{n+1} , then
 211 we need to calculate the transition probabilities associated with the embedded Markov chain
 212 defined as: $q_{n+1} = q_n - v_{n+1} + 1$. We need to calculate the transition probabilities associated
 213 with the embedded Markov chain defined as: $p_{ij} = \text{prob}[q_{n+1} = j / q_n = i]$ which is the
 214 probability that $i+1-j$ tasks are served during the inter-arrival time between the arrivals of
 215 two successive tasks. It is obvious that for $j > i+1$, $p_{ij} = 0$.
 216 Since the Markov chain is ergodic, a steady state probability distribution exists
 217 $\pi = [\pi_0, \pi_1, \pi_2, \dots, \pi_{m+r}]$ with $\pi_k = \lim_{n \rightarrow \infty} \text{prob}[q_n = k]$, $0 \leq k \leq m+r$ which is the
 218 probability that k tasks exist in the system immediately before the new arrival. To obtain this
 219 distribution it is required to solve the equation $\pi = \pi P$, where P is the matrix whose elements
 220 are the transition probabilities p_{ij} .
 221
 222 To find the elements of the transition probability matrix P , we need to count the number of
 223 tasks departing from the system between two successive arrivals, as shown in Fig.2.
 224

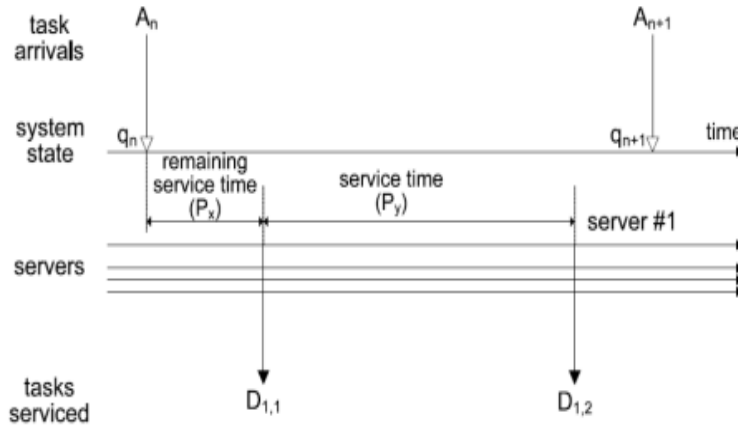


Fig. 2. System behavior between two MMPP arrivals

225
 226
 227

228 3.3 Departure Probabilities

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

$$231 \quad P = \begin{pmatrix} p_{11} & p_{12} \\ p_{21} & p_{22} \end{pmatrix}$$

232

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

$$\begin{aligned}
p_x &= \text{Prob}[B_+ < A] \\
&= \text{Prob}\left[\frac{B_+ < A}{J(t_n) = \lambda_1, J(t_{n+1}) = \lambda_1}\right] \times \text{Prob}[J(t_n) = \lambda_1, J(t_{n+1}) = \lambda_1] \\
&+ \text{Prob}\left[\frac{B_+ < A}{J(t_n) = \lambda_1, J(t_{n+1}) = \lambda_2}\right] \times \text{Prob}[J(t_n) = \lambda_1, J(t_{n+1}) = \lambda_2] \\
&+ \text{Prob}\left[\frac{B_+ < A}{J(t_n) = \lambda_2, J(t_{n+1}) = \lambda_1}\right] \times \text{Prob}[J(t_n) = \lambda_2, J(t_{n+1}) = \lambda_1] \\
&+ \text{Prob}\left[\frac{B_+ < A}{J(t_n) = \lambda_2, J(t_{n+1}) = \lambda_2}\right] \times \text{Prob}[J(t_n) = \lambda_2, J(t_{n+1}) = \lambda_2] \\
&= p_{11} \int_0^{+\infty} \text{Prob}\left[\frac{B_+ < A}{J(t_n) = \lambda_1, J(t_{n+1}) = \lambda_1, B_+ = x}\right] dB_+(x) + p_{12} \int_0^{+\infty} \text{Prob}\left[\frac{B_+ < A}{J(t_n) = \lambda_1, J(t_{n+1}) = \lambda_2, B_+ = x}\right] dB_+(x) \\
&+ p_{21} \int_0^{+\infty} \text{Prob}\left[\frac{B_+ < A}{J(t_n) = \lambda_2, J(t_{n+1}) = \lambda_1, B_+ = x}\right] dB_+(x) + p_{22} \int_0^{+\infty} \text{Prob}\left[\frac{B_+ < A}{J(t_n) = \lambda_2, J(t_{n+1}) = \lambda_2, B_+ = x}\right] dB_+(x) \\
&= p_{11} \int_0^{+\infty} e^{-\lambda_1 x} dB_+(x) + p_{12} \int_0^{+\infty} e^{-\lambda_2 x} dB_+(x) + p_{21} \int_0^{+\infty} e^{-\lambda_1 x} dB_+(x) + p_{22} \int_0^{+\infty} e^{-\lambda_2 x} dB_+(x) \\
&= p_{11} B_+^*(\lambda_1) + p_{12} B_+^*(\lambda_2) + p_{21} B_+^*(\lambda_1) + p_{22} B_+^*(\lambda_2)
\end{aligned}$$

Fig. 3 shows the evolution of P_x as a function of λ_1 and fixed λ_2 values ($\lambda_2 = 2, 10, 20$ and 50). Fig. 4 shows the evolution of P_x as a function of λ_1 and λ_2 . The two figures illustrate the decrease of P_x when λ_1 and λ_2 increase.

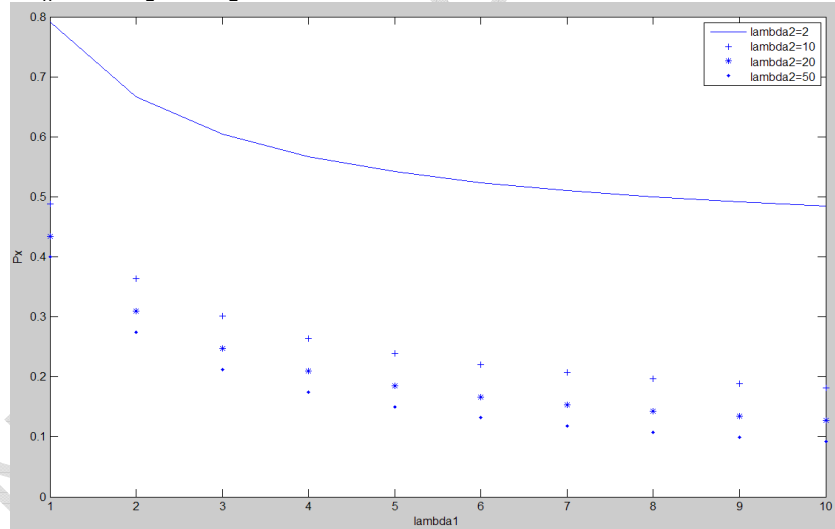


Fig. 3. Evolution of P_x as a function of λ_1 with fixed values of λ_2 .

$$p_x = \text{Prob}[B_+ < A] = \sum_{i=1}^N \sum_{j=1}^N p_{ij} B_+^*(\lambda_j)$$

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.

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

$$251 \quad p_y = \text{Pr ob}[B < A]$$

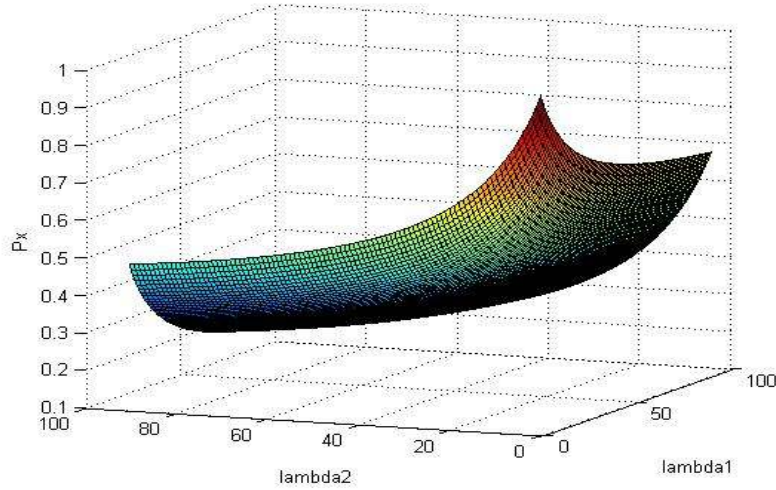
252 Using the same analysis in computing p_x we can show that:

$$253 \quad p_y = p_{11}B^*(\lambda_1) + p_{12}B^*(\lambda_2) + p_{21}B^*(\lambda_1) + p_{22}B^*(\lambda_2)$$

254

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

$$256 \quad p_y = \text{Pr ob}[B < A] = \sum_{i=1}^N \sum_{j=1}^N p_{ij}B^*(\lambda_j)$$



257 **Fig. 4. Tridimensional view of the evolution of P_x as a function of λ_1 and λ_2**

259 For the case of a MMPP-N we can generalize this probability as follows:

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

262 Let A and B be two events such as:

263 A: "The task in service is complete and leaves the system during the inter-arrival"

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

$$\begin{aligned} 266 \quad p_{z,k} &= \text{Pr ob}[A \cap B^{(k-1)}] \\ &= \text{Pr ob}(A) \times (\text{Pr ob}(B))^{(k-1)} \\ &= p_x \times (p_y)^{(k-1)} \\ &= [p_{11}B^*(\lambda_1) + p_{12}B^*(\lambda_2) + p_{21}B^*(\lambda_1) + p_{22}B^*(\lambda_2)] \times [p_{11}B^*(\lambda_1) + p_{12}B^*(\lambda_2) + p_{21}B^*(\lambda_1) + p_{22}B^*(\lambda_2)]^{(k-1)} \end{aligned}$$

267

268 The general expression for this probability is given by:

$$269 \quad p_{z,k} = \left[\sum_{i=1}^N \sum_{j=1}^N p_{ij}B^*(\lambda_j) \right] \times \left[\sum_{i=1}^N \sum_{j=1}^N p_{ij}B^*(\lambda_j) \right]^{(k-1)} \quad \text{with } p_{z,1} = p_x.$$

270 Using these values we can compute the transition probabilities matrix.

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

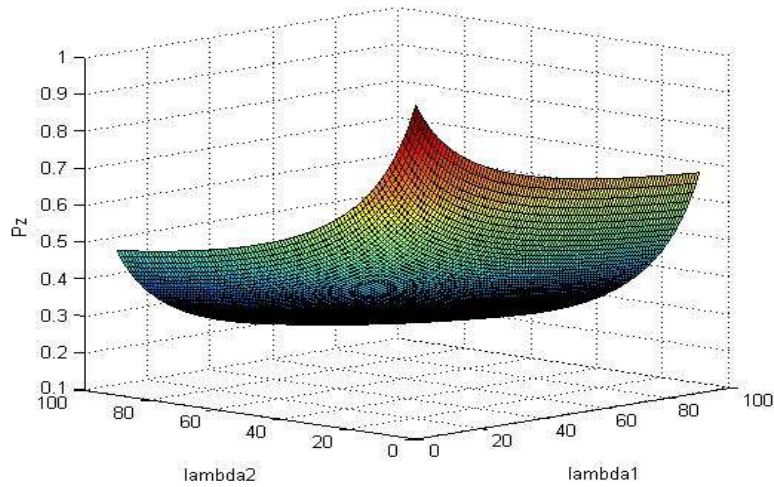


Fig. 5. Evolution of $P_{z,k}$ (for $k=3$) as a function of λ_1 and λ_2

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.

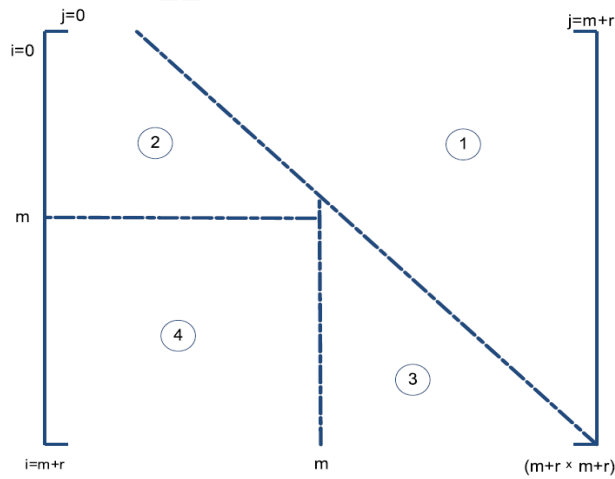


Fig. 6. Range of validity for p_{ij}

1 – For $i+1 < j$, $p_{ij} = 0$

288 2 – For $i < m$ and $j \leq m$ (no waiting), between two successive arrivals the probability that
 289 $i - j + 1$ tasks are served is:

$$290 \quad p_{ij} = C_{i-j}^i p_x^{i-j} (1-p_x)^j p_y + C_{i+1-j}^i p_x^{i+1-j} (1-p_x)^{j-1} (1-p_y)$$

291 3 – For $i \geq m$ and $j \geq m$, i.e. all servers are busy during the inter-arrival time. Let
 292 $w = i + 1 - j$ represents the number of tasks that leave the system between two successive
 293 Markov points. This number can be between 0 and infinity, but it is often close to 1. In this
 294 model it is assumed that w does not exceed 3 i.e. there are no more than three tasks served
 295 between two successive arrivals.

$$296 \quad p_{ij} = \sum_{s_1=\min(w,1)}^{\min(w,m)} C_m^{s_1} p_x^{s_1} (1-p_x)^{m-s_1} \times \sum_{s_2=\min(w-s_1,1)}^{\min(w-s_1,s_1)} [C_{s_1}^{s_2} p_{z,2}^{s_2} (1-p_{z,2})^{s_1-s_2} \times C_{s_2}^{w-s_1-s_2} p_{z,3}^{w-s_1-s_2} (1-p_{z,3})^{2s_2-w+s_1}]$$

297 4 - Finally for $i \geq m$ and $j < m$, i.e. all the servers are busy at the time of first arrival and
 298 $i - m$ tasks are in the queue, while after the arrival of the next task, there are exactly j tasks
 299 in the system, no one will be in the queue. The transition probability is expressed by:

$$300 \quad p_{ij} = \sum_{s_1=m-j}^{\min(w,m)} C_m^{s_1} p_x^{s_1} (1-p_x)^{m-s_1} \times \sum_{s_2=\min(w-s_1,m-j)}^{\min(w-s_1,s_1)} [C_{s_1}^{s_2} p_{z,2}^{s_2} (1-p_{z,2})^{s_1-s_2} \times C_{s_2}^{w-s_1-s_2} p_{z,3}^{w-s_1-s_2} (1-p_{z,3})^{2s_2-w+s_1}]$$

301

302 4. ANALYTICAL PERFORMANCES EVALUATION AT STEADY STATE

303 4.1 Equilibrium balance equations

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

308 The balance equations are:

$$309 \quad \pi_i = \sum_{j=0}^{m+r} \pi_j p_{ji} \quad 0 \leq i \leq m+r \quad \text{to which we add the normalization equation} \quad \sum_{i=0}^{m+r} \pi_i = 1.$$

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

312

313 4.2 Distribution of the number of tasks in the system

314 Once the steady-state probabilities are found, we can establish the generating function of
 315 the number of tasks $\Pi(z) = \sum_{k=0}^{m+r} \pi_k z^k$, and therefore we can deduce the average number of

316 tasks in the system by deriving the above expression, we thus find $\bar{p} = \Pi'(1)$

317 **proof:**

$$318 \quad \text{we have : } \Pi(z) = \sum_{k=0}^{m+r} \pi_k z^k = E(z^{q_n}) \quad \text{Then : } \Pi'(z) = E(q_n z^{q_n-1})$$

this leads to : $\bar{p} = \Pi'(1) = E(q_n)$

319 4.3 Distribution of waiting and response times

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

321 $\bar{t} = \frac{\bar{p}}{\bar{\lambda}(1-\Pi_{m+r})} = \frac{\bar{p}}{(\Pi_{MMPP} * \lambda^T)(1-\Pi_{m+r})}$ where $\bar{\lambda}$ is the intensity average of the

322 MMPP process.

323 Let W denotes the waiting time in the steady state, $W(x)$ the distribution function and $W^*(x)$
 324 its Laplace transform. It has been demonstrated in [28] that the length of the queue Q has
 325 the same distribution as W and therefore the number of tasks that arrive during the waiting
 326 time is expressed as: $Q(z) = W^*(\lambda(1-z))$.

327 Similarly we can assert that Q and W have the same distribution for a MMPP-2 process.

328 **Proof:**

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

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

330 $= \pi_1 w^*(\lambda_1(1-z)) + \pi_2 w^*(\lambda_2(1-z))$

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

331

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

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

334 $Q(z)$ could also be written as follows:

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

336

337

338 **Proof :**

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

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

$$342 \quad \lambda_e = \lambda^t \cdot \pi_{MMPP} (1 - \pi_{m+r})$$

343 Hence we have:

$$344 \quad W^*(s) = Q(z) \Big|_{z=1-\frac{s}{\lambda_e}} = Q\left(1 - \frac{s}{\lambda_e}\right) = Q\left(1 - \frac{s}{\lambda^t \cdot \pi_{MMPP} (1 - \pi_{m+r})}\right)$$

345 Moreover, the Laplace transform of response time is: $T^*(s) = W^*(s)B^*(s)$, in which the
 346 $W^*(s)$ and $B^*(s)$ are the Laplace transform of waiting time and the service time, respectively.

347 **Proof:**

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

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

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

350 Then : $T^*(z) = -E(Te^{-zT})$

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

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

351
 352
 353

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

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 of r :

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

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.

REFERENCES

1. L. Kleinrock. A vision for the internet. ST Journal of Research,2: November 2005
2. Abhishek Goel, Shikha Goel. Security Issues in Cloud Computing. International Journal of Application or Innovation in Engineering & Management Volume 1, Issue 4, December 2012
3. Peter Mell, and Tim Grance. Draft NIST Working Definition of Cloud Computing. 2009, Available: <http://csrc.nist.gov/groups/SNS/cloud-computing>
4. B. Furht, Cloud Computing Fundamentals. Handbook of Cloud Computing, Springer, 2010.
- 5.D. Armstrong and K. Djemame. Towards Quality of Service in the Cloud. Proceedings of the 25th UK Performance Engineering Workshop, Leeds, UK, July 2009
6. L. Wang, G. von Laszewski, A. Younge, X. He, M. Kunze, J. Tao, and C. Fu. Cloud Computing: A Perspective Study. New Generation Computing, vol. 28, 2010
7. L. Kleinrock, Queueing Systems: Theory, vol. 1, Wiley-Interscience, 1975.
8. K. Xiong and H. Perros. Service performance and analysis in cloud computing. IEEE 2009 World Conference on Services, February 2009.
9. B. Yang, F. Tan, Y. Dai, and S. Guo. Performance evaluation of cloud service considering fault recovery. First Int'l Conference on Cloud Computing, December 2009
10. P. Hokstad. Approximations for them/G/m queues. Operations Research, 26, 1978.
11. B. N.W. Ma and J.W. Mark. Approximation of the mean queue length of an M/G/c queueing system. Operations Research, 43, 1998.

- 396 12. M. Miyazawa. Approximation of the queue-length distribution of an M/GI/s queue by the
397 basic equations. Journal of Applied Probability, 23, 1986.
- 398 13. S. A. Nozaki and S. M. Ross. Approximations in finite-capacity multi-server queues with
399 poisson arrivals. Journal of Applied Probability, 15, 1978.
- 400 14. E. Page. Tables of waiting times for M/M/n, M/D/n and D/M/n and their use to give
401 approximate waiting times in more general queues. J. Operational Research Society, 33,
402 1982
- 403 15. D. D. Yao. Refining the diffusion approximation for the M/G/m queue. Operations
404 Research, 33, 1985
- 405 16. O. J. Boxma, J.W. Cohen, and N. Huffel. Approximations of the mean waiting time in an
406 M/G/s queueing system. Operations Research, 27, 1979
- 407 17. T. Kimura. Diffusion approximation for an M/G/m queue. Operations Research, 31, 1983.
- 408 18. Y. Takahashi. An approximation formula for the mean waiting time of an M/G/c queue. J.
409 Operational Research Society, 20, 1977
- 410 19. H. C. Tijms, M. H. V. Hoorn, and A. Federgru. Approximations for the steady-state
411 probabilities in the M/G/c queue. Advances in Applied Probability, 13, 1981.
- 412 20. T. Kimura. A transform-free approximation for the finite capacity M/G/s queue.
413 Operations Research, 44(6), 1996.
- 414 21. T. Kimura. Optimal buffer design of an M/G/s queue with finite capacity. Communications
415 in Statistics and Stochastic Models, 12(6), 1996
- 416 22. H. Khazaei, J. Misic, and V. B. Misic. Performance Analysis of Cloud Computing Centers
417 Using M/G/m/m+r Queuing Systems. IEEE Transactions on parallel and distributed systems,
418 23(5), 2012
- 419 23. Satyanarayana .A Dr. P. Suresh Varma Dr. M.V.Rama Sundari Dr. P Sarada Varma.
420 Performance Analysis of Cloud Computing under Non Homogeneous Conditions .
421 International Journal of Advanced Research in Computer Science and Software Engineering
422 Volume 3, Issue 5, May 2013
- 423 24. Jelena revzina. Possibilities of MMPP processes for bursty traffic analysis. Proceedings
424 of the 10th international conference "reliability and statistics in transportation and
425 communication", October 2010, riga, Latvia
- 426 25. Bolch, G., Greiner, S., de Meer, H., Trivedi, K.S. Queueing Networks and Markov
427 Chains: Modeling and Performance Evaluation with Computer Science Applications.
428 JohnWiley&Sons, 2006
- 429 26. F. Neuts . Models Based on the Markovian Arrival Process. IEICE Trans. Commun., vol.
430 E75-B, n°12, decembre 1992.
- 431 27. H. Takagi. Queueing Analysis, volume 1: Vacation and Priority Systems. North- Holland,
432 1991
- 433 28. K.T. Marshall and R.W. Wolff. Customer Average and Time Average Queue Lengths and
434 Waiting Times. J. Applied Probability, vol. 8, 1971.