

Performance Modeling for a Cloud Computing Center Using GE/G/m/k Queuing System

Mohamed Ben el aattar^{1,2}, Abdelkrim Haqiq^{1,2}

¹Computer, Networks, Mobility and Modeling Laboratory FST, Hassan 1st University, Settat, Morocco

²e-NGN Research Group, Africa and Middle East

Abstract: Cloud computing is a new paradigm for the provision of computing infrastructure; it aims to change the location of the computer on the network infrastructure to reduce the maintenance costs of hardware and software resources. Increasing demand in cloud computing, in recent time, pushes researchers to improve access to services by developing techniques that allow users in cloud computing networks to get better services in terms of optimum performances with lowest possible cost. To evaluate the performances of a cloud computing center, most related prior studies were motivated by using analytical models based on queuing theory. However, and to the best of our knowledge, a model taking into account the assumption of batch arrivals in the center has never been studied before. In this paper, this case is studied by modeling the cloud data center as a (GE/G/m/k) queuing system with GE distribution task arrivals, a general service time for requests as well as large number of physical servers and a task buffer of finite capacity. We used this model to evaluate the performance analysis of cloud server frames and we get an accurate estimate 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 and the blocking probability.

Keywords: cloud computing, queuing system, performance analysis, GE distribution, Markov chain.

1. Introduction

Cloud computing has been used to define applications delivered as services over the Internet (as well as the hardware and middleware that reside in data centers that are used to provide those services) [1]. In this technology three main services are provided by the Cloud computing architecture according to the needs of IT customers [2].

Firstly, Software as a Service (SaaS) provides access to complete applications as a service [3]. Secondly, Platform as a Service (PaaS) provides a platform for developing other applications on top of it [4]. Finally, Infrastructure as a Service (IaaS) provides an environment for deploying, running and managing virtual machines and storage. Technically, IaaS offers incremental scalability (scale up and down) of computing resources and on-demand storage [5].

Public cloud refers to situations where the cloud, and in particular infrastructure as-a-service, is made available publicly to individuals and organizations and is charged using metered billing (i.e. pay for what you use). Public cloud allows different end users to share hardware resources and network infrastructure and examples include Amazon and Rackspace.

The private cloud is targeted at large organizations, and generally provides more flexible billing models as well as the ability for these users to define secure zones within which only their company has access to the hardware and network (e.g. Rackspace private cloud, IBM). Hybrid clouds often refer to situations where organizations are making use of both public and private cloud for their infrastructures. The concept of cloud computing also assumes that resources are theoretically available on demand, whereby a user of the cloud can scale their cloud infrastructure immediately when the need arises, i.e. during a traffic surge.

There are a number of areas where results from performance engineering of software systems could benefit the area of cloud computing. Examples include SaaS performance design; autonomies; performance monitoring; resource utilization; and data analysis. Figure 1 shows a computer service scenario in cloud computing [6]. Due to benefits offered by Cloud computing, Quality of Service (QoS) is a broad topic in this technology, and some important quality measures in cloud's users prospective have to be evaluated [7]. Quantifying and characterizing such performance measures requires appropriate models. To model networks and estimate its QoS parameters, the queuing theory models are a classic and powerful tool [8].

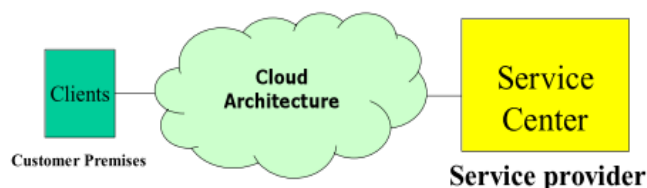


Figure 1: Computer Service Scenario in Cloud Computing

However, and due to the diversity of services in cloud computing different tasks can be treated in a cloud center and queuing modeling of cloud services has to take into consideration the type of tasks that is studied. There have been earlier studies which investigate the modeling and analyzing of QoS in cloud computing center. However, an aspect that has not received enough interest in the research is the possibility of batched arrivals. The model proposed in this paper consists of a queue where the task arrivals follow a generalized exponential (GE) distribution that can be used to model distributions where mean and the standard deviation are not equal. The service times of tasks are assumed to be independent and identically distributed random variables following a general distribution.

The proposed model can be notated based on Kendall's notation for queuing models as a GE/G/m/m+r queue, which indicates that system under consideration contains m servers where the tasks are served in a first come first served (FCFS) order, and the waiting queue capacity is finite and equal to r. These mathematical assumptions make the proposed model more convenient for the cloud environment nature, and it confer to the model the quality of being close to reality and the quality of scalability. Moreover and due to the introduction of the finite capacity, the system may experience blocking of task requests.

The rest of the paper is organized as follows: In the next section we give a brief overview of related work on cloud performance evaluation and performance characterization of queuing systems. A brief presentation of generalized exponential (GE) distribution is done in section 3. In section 4 we present the proposed model and discuss our analytical model in details. In Section 5 we solve our model in order to evaluate analytically desired performance metrics. Section 6 concludes our discussion.

2. Related Work

Cloud computing provides user a complete software environment. It provides resources such as computing power, bandwidth and storage capacity. It has engrossed considerable investigate attention, but only a diminutive portion of the work done so far has addressed performance issues, and rigorous analytical approach has been adopted by only a handful among these, particularly that adopting queuing theory models.

Yang et al. [9] proposed a fault recovery system scheduling for cloud services and analyzed the system as an open queue problem using a M/M/m/m+r queuing system; the results showed that addition of fault recovery increases average response time. Xiong et al. [6] modeled a cloud center as the classic open network, from which the distribution of response time is obtained, assuming that both inter-arrival and service times are exponential. Using the distribution of response time, the relationship among the maximal number of tasks, the minimal service resources and the highest level of services was found.

The cases of queuing system with generally distributed service time were the subject of most theoretical analyses. However, in these cases the steady state probability, the distributions of response time and the queue length have yet to be solved exactly. Consequently, researchers have developed many methods for approximating its solution.

The authors in [10] refined a diffusion approximation model for a M/G/m queue; they solved the equations incorporating some known results of the queue into the model. Their numerical studies indicate that the refined model provides significantly improved performance. However, in many practical cloud service situations, the arrivals of jobs are to be considered on time dependent in order to have accurate prediction of the performance measure of the cloud computing [11], and a number of measurements studies have reveal that the traffic generated by many real world applications exhibit a high degree of burstness and poses

correlation in the number of request arrivals [12], therefore the traditional Poisson process models cannot capture the burst nature of request arrival process. Hence it is needed to develop queuing model taking into account the characteristics of the type of modeled traffic.

The generalization of the M/G/m/m+r model given by Khazaei in [13], has been improved in [7] using an MMPP model for the task of arriving in the center, thus because the diversity and burstness of user requests was made in this paper. Modeling arrival process in a queue system using GE distribution is a useful tool, as it is a generalization that can take into consideration the batch arrivals, and several features of the models in which customers arrive singly are maintained in this generalization.

There are few works using GE distribution in network modeling. Kouvatso et al. in [14] have extended the model proposed in [15] by relaxing the assumption of Poisson arrival process using a GE/G/1/K queue to model finite input buffer. This paper proposes an analytical model for predicting the average worm latency in the hypercube with deterministic routing, wormhole switching and finite size input buffers. In this work we propose to use a queue system modeling for a cloud computing center with generalized exponential distribution for arrival tasks, and a general distribution for the service times. The system is assumed to be with finite capacity; and considering the properties of a cloud computing center, we assume that we have a multi-server system.

3. Generalized Exponential (GE) distribution

The generalized exponential (GE) distribution is a two parameter distribution which may be used to approximate distributions by matching two moments (as opposed to just one moment with the exponential distribution). The probability density function $f(x)$ is given by:

$$f(x) = \begin{cases} 1-\alpha & \text{for } x=0 \\ \alpha^2 \lambda e^{-\lambda \alpha x} & \text{for } x > 0 \end{cases} \quad (1)$$

with $\alpha \in [0,1]$

The mean of this distribution is $1/\lambda$ and the squared coefficient of variation (SqV), the ratio of the variance to the square of the mean, is $(2-\alpha)/\alpha$. When modeling inter-arrival times, the GE distribution can model batch Poisson arrivals with geometrically distributed batch sizes with mean $1/\alpha$.

The α term gives an impulse at the origin ($f(0)=1-\alpha$) giving a non-zero probability to a zero inter-arrival time. This allows a sequence of one or more zero inter-arrival times, and hence non-unit (geometric) batches. A deterministic unit batch size is given by setting the geometric distribution parameter to zero.

The queue can accommodate geometrically batched occurrences of both arrivals and processing completions. These batched streams cause transitions within the queue whose horizontal component is zero and with vertical component (increasing or decreasing queue length) given by

a, possibly truncated, geometric distribution showed in Figure 2. An arrival stream is truncated when it operates in a finite queue. When the arrival of a batch of customers would overflow the queue, i.e. the resulting number of customers within the queue would exceed the maximum queue length, we discard the excess customers and make the transition to the top of the queue. Hence, the arrival batch size s is distributed as follows:

$$P(i = S / j = J) = \begin{cases} \alpha(1 - \alpha)^{s-1} & \text{if } S + J < r \\ (1 - \alpha)^{s-1} & \text{if } S + J = r \\ 0 & \text{otherwise} \end{cases} \quad (2)$$

It follows that arrival streams to infinite queues, where $r = \infty$, are never truncated.

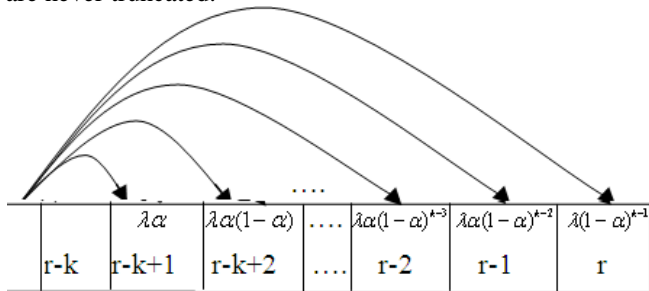


Figure 2: Arrival process truncation from level $r-k$ for a finite queue

GE distribution is useful for modeling random variables with SqV greater than 1, it is still a useful and versatile tool in the field of systems modeling. If a GE distribution is used to describe the time between arrivals task in a queue and service times, it has similar properties to that of the exponential distribution. Furthermore, in [16], Kouvatso has shown that the GE distribution is a robust two moment approximation for any service time distribution by showing that the mean queue length distribution of GE/G/1 queue subject to utilization and mean queue length constraints is identical to exact equilibrium solution when service distribution is represented by a moment-matched GE distribution.

4. The Proposed Analytical Model

4.1 Model Description

To model a center of cloud computing receiving user's traffic, we propose to use a $GE/G/m/k$ queuing system, with $k = m + r$. In this model we assume that:

- The time of arrival follows a GE process with parameters λ and α ;
- The service time is assumed to be generally distributed with a mean value μ ;
- The system has m servers;
- The buffer has a finite capacity of size r (the total system capacity is then $k = m + r$).

Batch arrivals with geometric size

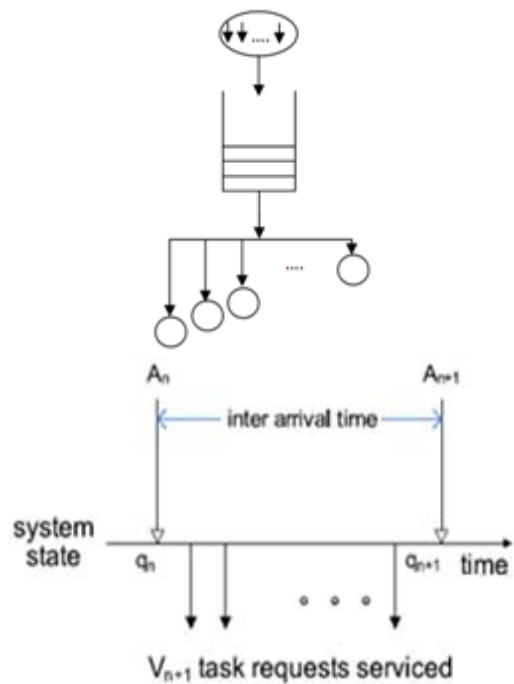


Figure 3: System behavior between two GE arrivals.

To analyze this queue we use the embedded Markov chain technique which is based on the selection of Markov points where the system receives an arrival batch. So we model the total number of tasks in the system, immediately before the arrival of a new batch of tasks. It is clear that this process is an homogeneous Markov chain with state space: $E = \{0, 1, 2, \dots, m + r\}$.

In the rest of this paper we use the following notations:

- A: The inter-arrival time; $A(x)$ its distribution function; $f(x)$ its density function; and its Laplace transform is noted:

$$A^*(s) = \int_0^\infty e^{-sx} f(x) dx \quad (3)$$

- B: The service times of tasks which are identically and independently distributed according to a general distribution with an average service time $\bar{b} = \frac{1}{\mu}$;

$B(x)$ the distribution function of B; $b(x)$ its density function; and the Laplace transform of the service time is:

$$B^*(s) = \int_0^\infty e^{-sx} b(x) dx \quad (4)$$

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 GE batch arrival and departure of the task which was in service when task arrival occurred. The probability distribution of Laplace transform of residual and elapsed task service times is calculated in [17] as:

$$B_-^*(s) = B_+^*(s) = \frac{1 - B^*(s)}{sb} \quad (5)$$

4.2 Model Analysis

As pointed above GE/G/m/m+r queuing system can be analyzed by applying the embedded Markov chain technique. Therefore we model the number of the tasks in the system (both in service and queued) at the moments immediately before the arrival of the batch tasks. This Markov chain is ergodic, thus, its stationary probability exists. To get the latter we need to calculate p_x, p_y and $p_{z,l}$ which represent the departure probabilities in the system. The Figure 4 shows the state-transition-probability diagram for such embedded Markov chain.

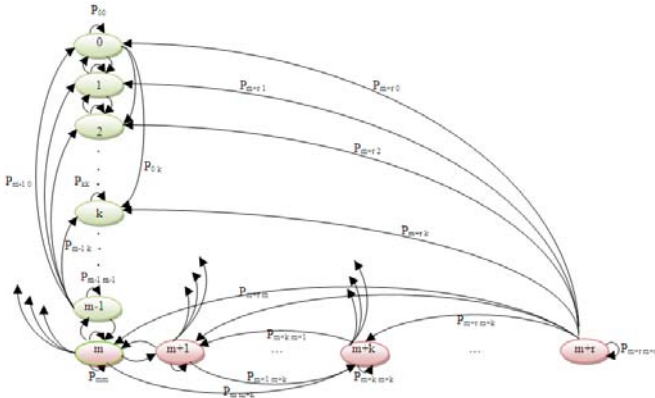


Figure 4: State-transition-probability diagram for the GE/G/m/m+r embedded Markov chain

Let A_n and A_{n+1} the moments of n^{th} and $(n+1)^{th}$ batch 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} indicate 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: $q_{n+1} = q_n - v_{n+1} + k$, with $0 < k$.

As for a geometric distribution X we have $\lim_{k \rightarrow +\infty} p(X = k) = \lim_{k \rightarrow +\infty} \alpha(1-\alpha)^{k-1} = 0$, we can assume that k is less than a given value k_{max} ($k \leq k_{max}$).

These transition probabilities are defined by $p_{ij} = prob[q_{n+1} = j / q_n = i]$.

p_{ij} represents the probability that $(i+k-j)$ tasks are served during the interval between the arrivals of two successive batches. It is obvious that for $j > i + k$, $p_{ij} = 0$.

The stationary probability distribution $\pi = [\pi_0, \pi_1, \pi_2, \dots, \pi_{m+r}]$ is defined by $\pi_l = \lim_{n \rightarrow \infty} prob[q_n = l]$ for $0 \leq l \leq m+r$. And it is the solution of the equation $\pi = \pi P$ where P is the matrix $P = (p_{ij}); 0 \leq i, j \leq m+r$.

To find the elements of the transition probability matrix P , we need to count the number of tasks departing from the system between two successive arrivals, as shown in Figure 3.

4.2.1 Departure Probabilities

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, this result presents the probability P_x of no task arrivals during residual task service time. This probability can be calculated as:

$$\begin{aligned} p_x &= Prob[B_+ < A] \\ &= \int_0^{+\infty} Prob[B_+ < A / B_+ = x] dB_+(x) \\ &= \int_0^{+\infty} \left(\int_{y=x}^{+\infty} \alpha^2 \lambda e^{-\lambda \alpha y} dy \right) dB_+(x) \\ &= \alpha \int_0^{+\infty} \left(\int_{y=x}^{+\infty} \lambda \alpha e^{-\lambda \alpha y} dy \right) dB_+(x) \\ &= \alpha \int_0^{+\infty} e^{-\lambda \alpha x} dB_+(x) \end{aligned}$$

Then : $p_x = \alpha B_+^*(\lambda \alpha)$ (6)

In the case when arriving task can be accommodated immediately by an idle server we have to evaluate the probability that such task will depart before next batch tasks arrival. We will denote this probability as P_y and it is defined for GE, as follows:

$$\begin{aligned} p_y &= Prob[B < A] \\ &= \int_0^{+\infty} Prob[B < A / B = x] dB(x) \\ &= \int_0^{+\infty} \left(\int_{y=x}^{+\infty} \alpha^2 \lambda e^{-\lambda \alpha y} dy \right) dB(x) \\ &= \alpha \int_0^{+\infty} \left(\int_{y=x}^{+\infty} \lambda \alpha e^{-\lambda \alpha y} dy \right) dB(x) \\ &= \alpha \int_0^{+\infty} e^{-\lambda \alpha x} dB(x) \end{aligned}$$

Then : $p_y = \alpha B^*(\lambda \alpha)$ (7)

The probability that k tasks depart from a server before the arrival of a new batch 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".

$$\begin{aligned} p_{z,l} &= Prob[A \cap B^{(l-1)}] \\ &= Prob(A) \times (Prob(B))^{(l-1)} \\ &= p_x \times (p_y)^{(l-1)} \\ &= [\alpha B_+^*(\lambda \alpha)] \times [\alpha B^*(\lambda \alpha)]^{(l-1)} \end{aligned}$$

Then : $p_{z,l} = \alpha^l [B_+^*(\lambda \alpha)] \times [B^*(\lambda \alpha)]^{(l-1)}$ (8)

Using these values we can compute the transition probabilities matrix.

4.2.2 Transition Matrix

After calculating the departure probabilities p_x, p_y and $p_{z,l}$,

in the embedded Markov chain, we may identify four different regions of operation for which different conditions hold.

- 1 – For $i + k < j$, $p_{ij} = 0$ with $0 < k \leq k_{\max} < r$.
- 2 – For $i < m$ and $j \leq m$ (no waiting), between two successive arrivals the probability that $i - j + k$ tasks are served with $0 < k \leq k_{\max} < r$ is:

$$p_{ij} = \sum_{k=1}^{m-i} [C_{i-j}^k p_x^{j-k} (1-p_x)^j p_y^k + C_{i+k-j}^k p_x^{i+k-j} (1-p_x)^{j-k} (1-p_y)^k] \alpha (1-\alpha)^{k-1} + \sum_{k=m-i+1}^{k_{\max}} [\sum_{s_1=\min(w,1)}^{\min(w,m)} C_m^{s_1} p_x^{s_1} (1-p_x)^{m-s_1} \times \sum_{s_2=\min(w-s_1,1)}^{\min(w-s_1,s_1)} [C_{s_1}^{s_2} p_{z,2}^{s_2} (1-p_{z,2})^{s_1-s_2} \times C_{s_2}^{w-s_1-s_2} p_{z,3}^{w-s_1-s_2} (1-p_{z,3})^{2s_2-w+s_1}]] \alpha (1-\alpha)^{k-1} \quad (9)$$

3 – For $i \geq m$ and $j \geq m$, i.e. all servers are busy during the inter-arrival time. Let $w = i + k - j$ with $0 < k \leq k_{\max}$, 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. that there are no more than three tasks served between two successive arrivals.

if $k_{\max} < m + r - i$ then :

$$p_{ij} = \sum_{k=1}^{k_{\max}} [\sum_{s_1=\min(w,1)}^{\min(w,m)} C_m^{s_1} p_x^{s_1} (1-p_x)^{m-s_1} \times \sum_{s_2=\min(w-s_1,1)}^{\min(w-s_1,s_1)} [C_{s_1}^{s_2} p_{z,2}^{s_2} (1-p_{z,2})^{s_1-s_2} \times C_{s_2}^{w-s_1-s_2} p_{z,3}^{w-s_1-s_2} (1-p_{z,3})^{2s_2-w+s_1}]] \alpha (1-\alpha)^{k-1} \quad (10)$$

if $k_{\max} \geq m + r - i$ then :

$$p_{ij} = \sum_{k=1}^{m+r-i-1} [\sum_{s_1=\min(w,1)}^{\min(w,m)} C_m^{s_1} p_x^{s_1} (1-p_x)^{m-s_1} \times \sum_{s_2=\min(w-s_1,1)}^{\min(w-s_1,s_1)} [C_{s_1}^{s_2} p_{z,2}^{s_2} (1-p_{z,2})^{s_1-s_2} \times C_{s_2}^{w-s_1-s_2} p_{z,3}^{w-s_1-s_2} (1-p_{z,3})^{2s_2-w+s_1}]] \alpha (1-\alpha)^{k-1} + \sum_{s_1=\min(m+r-j,1)}^{\min(m+r-j,m)} C_m^{s_1} p_x^{s_1} (1-p_x)^{m-s_1} \times \sum_{s_2=\min(m+r-j-s_1,1)}^{\min(m+r-j-s_1,s_1)} [C_{s_1}^{s_2} p_{z,2}^{s_2} (1-p_{z,2})^{s_1-s_2} \times C_{s_2}^{m+r-j-s_1-s_2} p_{z,3}^{m+r-j-s_1-s_2} (1-p_{z,3})^{2s_2-(m+r-j)+s_1}]] (1-\alpha)^{k-1} \quad (11)$$

4 - Finally for $i \geq m$ and $j < m$, i.e. that 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 tasks, there are exactly j tasks in the system, no one will be in the queue. The transition probability is expressed by:

if $k_{\max} < m + r - i$ then:

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

$$] \alpha (1-\alpha)^{k-1} \quad (12)$$

if $k_{\max} \geq m + r - i$ then:

$$p_{ij} = \sum_{k=1}^{m+r-i-1} [\sum_{s_1=\min(w,1)}^{\min(w,m)} C_m^{s_1} p_x^{s_1} (1-p_x)^{m-s_1} \times \sum_{s_2=\min(w-s_1,m-j)}^{\min(w-s_1,s_1)} [C_{s_1}^{s_2} p_{z,2}^{s_2} (1-p_{z,2})^{s_1-s_2} \times C_{s_2}^{w-s_1-s_2} p_{z,3}^{w-s_1-s_2} (1-p_{z,3})^{2s_2-w+s_1}]] \alpha (1-\alpha)^{k-1} + \sum_{s_1=\min(w,1)}^{\min(m+r-j,m)} C_m^{s_1} p_x^{s_1} (1-p_x)^{m-s_1} \times \sum_{s_2=\min(m+r-j-s_1,m-j)}^{\min(m+r-j-s_1,s_1)} [C_{s_1}^{s_2} p_{z,2}^{s_2} (1-p_{z,2})^{s_1-s_2} \times C_{s_2}^{m+r-j-s_1-s_2} p_{z,3}^{m+r-j-s_1-s_2} (1-p_{z,3})^{2s_2-(m+r-j)+s_1}]] (1-\alpha)^{k-1} \quad (13)$$

5. Performance Evaluation at Steady State

Once we find the matrix P , we can establish the balance equations in order to obtain the steady state distribution. We need to solve the system formed by the balance equations:

$$\pi_i = \sum_{j=0}^{m+r} \pi_j p_{ji} \quad 0 \leq i \leq m+r \quad (14)$$

with the normalization equation $\sum_{i=0}^{m+r} \pi_i = 1$.

It is still difficult to find an analytic solution for this system and therefore a numerical solution is required. Once this distribution defined, some performance parameters can be computed.

Number of Tasks in the System:

Using the steady state probabilities, we can establish the generating function of the number of tasks

$$\Pi(z) = \sum_{k=0}^{m+r} \pi_k z^k \quad (15)$$

and therefore we can deduce the average number of tasks in the system by deriving the above expression, we thus find $\bar{p} = \Pi'(1)$.

Waiting and Response Times:

Response time is a metric that includes any delay that the task routed to the center as the traffic may suffer while waiting for service.

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

$$\bar{t} = \frac{\bar{p}}{\lambda(1-\pi_{m+r})} = \frac{\bar{p}}{(\lambda\alpha)(1-\pi_{m+r})} \quad (16)$$

Where $\bar{\lambda}$ is the average intensity of the GE distribution.

Let W denotes the waiting time in the steady state, $W(x)$ the distribution function and $W^*(x)$ its LST. It has been demonstrated in [18] 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 GE distribution, we have demonstrated that Q and W have the following distribution:

$$Q(z) = \alpha w^* (\lambda\alpha(1-z)) \quad (17)$$

Proof:

$$\begin{aligned}
 Q(z) &= E(z^{U_{n+1}}) = \sum_{k=0}^{+\infty} \text{Pr } ob(U_{n+1} = k) z^k \\
 &= \sum_{k=0}^{+\infty} [\text{Pr } ob(U_{n+1} = k)] z^k \\
 &= \sum_{k=0}^{+\infty} \left[\int_0^{+\infty} \alpha \frac{(\lambda \alpha t)^k}{k!} e^{-\lambda \alpha t} dw(t) \right] z^k \\
 &= \alpha \sum_{k=0}^{+\infty} \left[\int_0^{+\infty} \frac{(\lambda \alpha tz)^k}{k!} e^{-\lambda \alpha t} dw(t) \right] \\
 &= \alpha \int_0^{+\infty} \sum_{k=0}^{+\infty} \frac{(\lambda \alpha zt)^k}{k!} e^{-\lambda \alpha t} dw(t) \\
 &= \alpha \int_0^{+\infty} e^{z \lambda \alpha t} e^{-\lambda \alpha t} dw(t) \\
 &= \alpha \int_0^{+\infty} e^{-(1-z)\lambda \alpha t} dw(t)
 \end{aligned}$$

Then : $Q(z) = \alpha w^*(\lambda \alpha (1 - z))$

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

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

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

$$\lambda_e = \lambda \alpha (1 - \pi_{m+r})$$

Hence we have:

$$W^*(s) = Q(z) \Big|_{z=1-\frac{s}{\lambda_e}} = Q\left(1 - \frac{s}{\lambda_e}\right) = Q\left(1 - \frac{s}{\lambda \alpha (1 - \pi_{m+r})}\right) \quad (18)$$

Moreover, we demonstrated that the LST of response time can be written as:

$$T^*(s) = W^*(s) B^*(s) \quad (19)$$

in which the $W^*(s)$ and $B^*(s)$ are the LST of waiting time and the service time, respectively.

Proof:

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

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

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

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

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

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

Immediate service in the system:

Here we are interested in the probability that the traffic

routed from the user to the cloud center will get into service immediately upon arrival, without any queuing. In this case the routing decision of the task will not cause any extra delay.

Get into service immediately upon arrival means that there is at least an ideal server and thus the response time would be equal to the service time:

$$P_{nq} = \sum_{i=0}^{m-1} \pi_i \quad (20)$$

Blocking probability:

The blocking probability is a common measure of network performance. It is defined as the probability that the task cannot be accepted in the center. It is a very important parameter, and it can be decisive in the decision of routing traffic, especially when this decision is based on criteria such looking for faster service for example. Since GE 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:

$$P b_r = \pi_{m+r} \quad (21)$$

6. Conclusion

In the context of the evolution of cloud computing network, to develop realistic models which represent centers of such a network appears as a great challenge for researchers. In this paper we have proposed an analytical model for performance evaluation of a cloud computing data centre using the queue GE/G/m/m+r. Owing to the nature of the cloud environment and the diversity of needs and demands of users, the proposed model uses a Generalized exponential (GE) arrival process that reflects the nature of arrivals in the cloud with batch Poisson with geometrically distributed batch sizes, a general service time, a number of servers and a finite buffer capacity. In this 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] Armbrust, M., Fox, A., Griffith, R., Joseph, A. D., Katz, R., Konwinski, A., Lee, G., Patterson, D., Rabkin, A., Stoica, I, Zaharia M.: A View of Cloud Computing. Communications of the ACM 53 (4) pp. 50-58 (2010).
- [2] Saurabh Kumar Garg, Steve Versteeg, Rajkumar Buyya; A framework for ranking of cloud computing services. Future Generation Computer Systems 29 (2013) 1012-1023.
- [3] M. Cusumano, Cloud computing and SaaS as new computing platforms, Communications of the ACM 53 (4) (2010) 27-29.
- [4] E. Ciurana, Developing with Google App Engine, Apress, Berkeley, CA, USA, 2009.
- [5] R. Buyya, C. Yeo, S. Venugopal, J. Broberg, I. Brandic, Cloud computing and emerging IT platforms: vision, hype, and reality for delivering computing as the 5th utility, Future Generation Computer Systems 25 (6) (2009) 599-616.

- [6] K. Xiong and H. Perros. Service performance and analysis in cloud computing. In IEEE 2009 World Conference on Services, pages 693–700, February 2009.
- [7] F. Oumellal, M. Hanini, and A. Haqiq, “MMPP/G/m/m+r Queuing System Model to analytically evaluate Cloud Computing Center Performances”, British Journal of mathematics and computer science. 4(10): 1301-1317, 2014
- [8] L. Kleinrock, Queueing Systems: Theory, vol. 1, Wiley-Interscience, 1975.
- [9] B. Yang, F. Tan, Y. Dai, and S. Guo. Performance evaluation of cloud service considering fault recovery. In First Int'l Conference on Cloud Computing (CloudCom) 2009, 571–576, December 2009
- [10] D. D. Yao, “Refining the diffusion approximation for the M/G/m queue,” Operations Research, vol. 33, pp. 1266–1277, 1985.
- [11] Satyanarayana .A Dr. P. Suresh Varma Dr. M.V.Rama Sundari Dr. P Sarada Varma. Performance Analysis of Cloud Computing under Non Homogeneous Conditions. International Journal of Advanced Research in Computer Science and Software Engineering Volume 3, Issue 5, May 2013.
- [12] T.Sai Sowjanya, D.Praveen, K.Satish, A.Rahmain, “The Queueing Theory in Cloud Computing to Reduce the waiting Time”, in IJCSET ,April-2011.
- [13] H. Khazaei, J. Misic, and V. B. Misic. Performance Analysis of Cloud Computing Centers Using M/G/m/m+r Queuing Systems. IEEE Transactions on parallel and distributed systems, 23(5): 936-943, 2012.
- [14] D.D. Kouvatsos, S. Assi, M. Ould-Khaoua, Performance modelling of hypercubes with deterministic wormhole routing, Proc. 1st International Working Conference Performance Modelling and Evaluation of Heterogeneous Networks (HET-NETs'03), D. D. Kouvatsos (Ed.), ISBN 0-9540151-3-4, pp. 77/1-77/10
- [15] P.-C. HU, L. Kleinrock, An Analytical model for wormhole routing with finite size input Buffers, 15th International Telegraphic Congress, 1997
- [16] D. D. Kouvatsos. Mem for arbitrary queueing networks with multiple general servers and repetitive-service blocking. Performance Evaluation, 10:169–195, 1989.
- [17] Takagi H. Queueing Analysis, volume 1: Vacation and Priority Systems. North- Holland; 1991.
- [18] Marshall KT and Wolff RW. Customer Average and Time Average Queue Lengths and Waiting Times. J. Applied Probability. 1971; 8.



Abdelkrim Haqiq has a High Study Degree (DES) and a PhD (Doctorat d'Etat), both in Applied Mathematics, option modeling and performance evaluation of computer communication networks, from the University of Mohamed V, Rabat, Morocco. Since September 1995, he has been working as a Professor at the Faculty of Sciences and Techniques, Settat, Morocco. He is the Director of IR2M laboratory. He is also a General Secretary of e-Next Generation Networks (e-NGN) Research Group, Moroccan section. Abdelkrim Haqiq's interests lie in the areas of applied stochastic processes, stochastic control, queueing theory, game theory, and their applications for modeling/simulation and performance analysis of computer communication networks. He is the author and co-author of more than 50 papers (international journals and conferences/workshops). He was the Chair of the NGNS'2010, held in Marrakech, July, 8- 10, 2010 and the TPC Chair of the NGNS'2012 international conference, held in Portugal, December, 2 - 4, 2012. He was also the TPC Chair of the iCEER2013 international conference, held in Marrakesh July, 1st –5th, 2013. Dr. Abdelkrim Haqiq is also a TPC member and a reviewer for many international conferences. He is also a Guest Editor of a special issue on Next Generation Networks and Services of the International Journal of Mobile Computing and Multimedia Communications, July-Septembe 2012, Vol. 4, No. 3.

Author Profile



Mohamed Ben el aattar received the B.Sc. degree in Applied Mathematics from the University of Hassan 2nd, Faculty of Sciences Ben msik, Casablanca, Morocco, in 1998, and M.Sc. degree in Mathematical and Applications engineering from the Hassan 1st University, Faculty of Sciences and Techniques (FSTS), Settat, Morocco, in 2011. Currently, he is working toward his Ph.D. at FSTS. His current research interests include performance evaluation and control of telecommunication networks and cloud computing.