Modelling of a cloud platform via $M/M_1 + M_2/1$ queues of a Jackson network

R. Sivasamy*

Department of Mathematics and Statistical Sciences,
Botswana International University of Science and Technology,
Palapye, P. Bag 16, Botswana
Email: rssamy@yahoo.com
*Corresponding author

N. Paranjothi

Department of Statistics,
Annamalai University,
Annamalainagar, Tamil Nadu, India
Email: jothi_stat@yahoo.co.in

Abstract: Modelling of a cloud platform that can provide the best quality of service (QoS) to minimise the average response times of its clients is investigated via an open Jackson network. Compact expressions for the input and output parameters and measures of the proposed model are presented. Designing of the model involves the performance measures of $M/M_1 + M_2/1$ queues with a K policy. This new cloud system is able to control virtual machines dynamically and to implement its operations to promote effectiveness in most of the commercial applications.

Keywords: cloud computing; open Jackson network; M/M/1 queue; response time; quality of service; QoS.

Reference to this paper should be made as follows: Sivasamy, R. and Paranjothi, N. (2023) 'Modelling of a cloud platform via $M/M_1 + M_2/1$ queues of a Jackson network', *Int. J. Cloud Computing*, Vol. 12, No. 1, pp.63–71.

Biographical notes: R. Sivasamy is a Visiting Senior Lecturer of Statistics in the Department of Mathematics and Statistical Sciences at the Botswana International University of Science and Technology (BIUST), Botswana. He has four decades of teaching and research experience at university levels, (2008 to 2018 at the University of Botswana and 1978 to 2008, Annamalai University, India). He has been assisting and supervising researchers and scholars interested in the study of stochastic processes and their applications' and helping them to solve problems relating to queue, reliability and inventory theory. His research interests span over computational methods and data science. Much of his work has been on improving the performance of service facilities through the application of matrix analytic methods and performance evaluation. He has given numerous invited talks and tutorials. He is an author and co-author of more than 60 publications in international journals.

Copyright © 2023 Inderscience Enterprises Ltd.

N. Paranjothi is an Assistant Professor in the Department of Statistics Annamalai University, India. He has over fifteen years of teaching experience, published more than twenty research articles in the national and international journals and authored two books. He received his Ph.D. in Statistics from Annamalai University in 2009. His research interests include stochastic processes and their applications in queueing and optimization problems.

1 Introduction

This paper proposes a simple cloud architecture which is modelled via an open Jackson network of interconnected servers operating $M/M_1 + M_2/1$ queues to access a single database server. The main aim is to measure quality of service (QoS) performance of the cloud computing by an optimal scaling technique. This is so designed to identify and manage the users' response time for services and to plan the proper deployment of virtual machines, according to the system load. Further, the resulting model is capable of determining the required number of virtual machines to reach the QoS (i.e., the response time of the applications entering the system). Also, the cloud system is able to add or remove virtual machines dynamically according to the estimated or expected outputs.

1.1 Components of a simple cloud computing architecture

A 'cloud' is designed with a group of networked computer hardware. Users can access many aspects of computing through online services and control it remotely via web interfaces. Cloud storage centre is a place that spans multiple servers in multiple locations in which the digital data are stored in logical pools. Such an environment is generally managed by cloud storage providers.

1.1.1 Cloud storage versus cloud computing

Cloud computing involves the power of the internet and a vast remote network of interconnected machines to outsource tasks which we conventionally perform on our personal laptops and desktops.

People of organisations who wish to store their application data use to buy or lease storage capacity from the cloud providers. Cloud storage services are frequently accessed through a web service application programming interface (API) or by applications that utilise API. Hence, cloud computing allows people to run their computing applications through the internet, saving them time, space and lots of money.

A typical cloud system consists of two primary components, i.e., front-end and back-end:

- *The front-end:* It consists of software components like remote client applications and interfaces. Availing these applications, customers approach standard web protocols to access the system and then an authentication protocol to connect to the cloud platform until it grants access. If the response time exceeds a deadline for few customers, then they may leave from the queuing area.

- *The back-end* (called a cloud): It is a service facility with a data service centre which receives jobs from different virtual machines (called processing servers) that are routing all incoming jobs. Its functions include management of the job queue, the servers and their virtual machines and the storage servers with their database system.

Simple cloud storage refers to a service centre with a single access point at the back-end for the customers. This service centre is a host to process service applications and hence collects all service resources of customers processed by a service provider. Each service request of customers/clients is transmitted from the front-end to the web server with a service layer agreement (SLA) policy. The SLA makes a contract agreement through negotiation(s) between a customer and the service provider.

Thus, the customers called users generate service requests at a given rate for processing by the service provider through the service centre of the cloud according to the agreements made with regard to QoS requirements.

Even though simulated predictions based on the model are not in real scale, they must be very reliable, and depending on the number of servers and the arrival speed, these could be very useful for generating an accurate approximation of the task response times subject to the SLA.

1.2 Recent developments on cloud models

To study the QoS in cloud computing, Vilaplana et al. (2014) have developed a model for cloud computing involving open Jackson network, see Figure 2. They evaluate the QoS of the cloud performance in terms of response times of jobs measured by nodes that are modelled as an M/M/1 queue or an M/M/m system.

There are other contributions which use general exponential distribution in the modelling of cloud network. El Aattar and Haqiq (2014) have considered a model of GE/G/m/m+r queue for performance estimation of a cloud computing data centre. The nature of arrivals follows batch Poisson with geometrically scattered batch size and uniform service time. They have calculated the probability of immediate service, mean response time, mean number of jobs in the system and blocking probability.

Khazaei et al. (2012) have proposed a model M/G/m/m+r for a cloud centre. Customers arrive singly and form a queue over a task buffer of finite capacity. They have calculated various performance measures. The same model has been extended by Oumellal et al. (2014) using Markov-modulated Poisson process (MMPP) model.

Jaiganesh et al. (2015) have analysed a cloud platform using an M/G/m/m+r priority type of queuing system. They included weighted fair queuing to achieve high profit without affecting the performance. Xiong and Perros (2009) have studied service performance of a cloud computing and obtained distribution of response times for a cloud centre modelled as an open Jackson network.

Section 2 deals with modelling of a cloud via M/M$_1$ + M$_2$/1 queuing system with a K policy and obtain average response times of requests that demanded computing. Section 3 formulates a 'benchmark cloud model (BMCM)' with a virtual node PS_1 as an M/M/1 service facility and explains the way of computing the total response time. Section 4 presents a numerical illustration for a comparative study between the total response times due to the proposed cloud modelling with that of BMCM. Section 5 provides a formal summary for derived results on cloud computing.

2 Modelling of a cloud via $M/M_1 + M_2/1$ queuing system with a 'K' policy

A simple cloud is now proposed (see Figure 2) as an open Jackson network composed of a multi-server queuing system with an entry server (ES), two processing servers PS_1 and PS_2, a single data server (DS), an output server (OS) and a client server (CS).

2.1 Distribution of computing requests and their average response times

The single ES acts as an M/M/1 service facility and as a load balancer transmits the user requests to the processing server nodes PS_j with probability p_j ($p_1 + p_2 = 1$) for $j = 1$ and 2. Inter-arrival times of such requests and their service times are assumed to be *i.i.d.* random variables and exponentially distributed with mean values $(1/\lambda)$ and $(1/\mu)$ respectively where $\rho_0 = (\lambda/\mu) < 1$. During these forwarding exercise of requests, an algorithm is used to decide the probabilities p_j for $j = 1$ and 2. Thus, the expected response time, say T_{ES}, of a request at ES is given by

$$T_{ES} = \frac{L_1}{\lambda} = \frac{1}{\mu - \lambda} \tag{1}$$

Average response time at PS_j for $j = 1$ and 2: let τ denote the output probability of leaving the cloud server CS linked with the open Jackson network represented in Figure 2. Application of the properties of open Jackson network leads to a fact that

$$\gamma = \frac{\lambda}{1 - \tau} \tag{2}$$

Let T_{PSj} represent the average response time of the requests at the process servicing node PS_j for $j = 1$ and 2. The node PS_1 is modelled as an $M/M_1 + M_2/1$ queue with mean arrival rate $p_1\ \gamma$ of the Poisson arrival process of requests. The exponential service rate is μ as and when queue length $Q = 0, 1, \ldots, (K + 1)$ and it changes to $\mu_1 + \mu_2$ for the queue length level $(K + 2)$ onwards. Here, for a given set of prefixed value of a positive integer K, and service rates $\mu > 0$ and $\mu_2 > 0$, let

$$\rho = \left(\frac{\lambda_1}{\mu + \mu_2}\right) \text{ and } \rho_1 = \left(\frac{\lambda_1}{\mu}\right) < 1. \tag{3}$$

2.2 $M/M_1 + M_2/1$ queuing system with a 'K' policy

Sivasamy et al. (2019) have obtained steady state results for an $M/M_1 + M_2/1$ queuing system with change in service rates under a 'K' policy. The assumption is that for all customers, if the number of customers Q in the system is less than or equal to a threshold as $(K + 1)$, the service rate is set as a low value $\mu > 0$ and for all customers when the system size $Q = (K + 2), (K + 3), \ldots$, the service rate is changed to $(\mu + \mu_2) > \mu > 0$. Then, $q_n = P(Q = n)$, the steady state probability that there are n customers in the system are found as below:

$$q_0 = \frac{(1-\rho)(1-\rho_1)}{1-\rho-\rho_1^{(K+1)}(\rho_1-\rho)}$$
$$q_j = q_0\rho_1^j \quad \text{for } j = 0, 1, 2, \ldots, (K+1) \tag{4}$$
$$q_j = q_{(K+1)}\rho^{(i-(K+1))} \quad \text{for } j = (K+2), (K+3), \ldots, \infty$$

Let the expected number $E(Q) = \sum_{n=0}^{\infty} nq_n$ of customers in the system be denoted by L_2. Thus

$$L_2 = q_0\left[\frac{\rho_1\left\{1+K\rho_1^{\{K+1\}}-(K+1)\varepsilon_1^K\right\}}{(1-\rho_1)^2}+\frac{\rho\rho_1^{\{K\}}[K+1-K\rho]}{(1-\rho)^2}\right] \tag{5}$$

It can be verified that the value L_2 of equation (5) becomes the expected system size of M/M/1 model since $\rho_1 = \rho$ when $\mu_2 = 0$, i.e.,

$$L_1 = \frac{\rho}{(1-\rho)} \tag{6}$$

Thus, the mean response time T_{PS_1} is given by

$$T_{PS_1} = \left(\frac{L_2}{\lambda_1}\right) \tag{7}$$

The node PS_2 is modelled as an M/M/c queue with mean service rate μ and arrival rate $\lambda_2 = p_2\nu$ that satisfy the following condition (8):

$$\lambda_2 = p_2\gamma = p_2\left(\frac{\lambda}{1-\tau}\right) < \mu \tag{8}$$

Let L_c denote the mean queue length of the c-server queue M/M/c for which details are given in the next section under benchmark case. Thus, the mean response time T_{PS_2} of the node PS_2 is given by

$$T_{PS_2} = \left(\frac{L_c}{\lambda_2}\right) \tag{9}$$

Assume that each of the virtual servers PS_j (j = 1 and 2) is able to access the server DS with probability δ. Hence, the node DS is modelled as an M/M/1 queue with mean service rate μ and mean arrival rate

$$\delta\nu = \delta\frac{\lambda}{1-\tau} \tag{10}$$

This DS accesses any type of input/output (I/O) files from the virtual machines into its secondary memory in the cloud area. Hence, the mean response time T_{DS} is given by

$$T_{DS} = \left(\frac{1}{\mu-\delta_\gamma}\right) \tag{11}$$

The main role of the OS is to send back the response information to the client through the internet. It is noticed that this OS is modelled as an M/M/1 queue with mean arrival rate ν and service rate μ. Thus, the mean response time T_{OS} is given by

$$T_{OS} = \frac{1}{\mu - \gamma} \tag{12}$$

The server CS accesses the output responses from OS and sends requests at rate λ to the CS with probability $(1 - \tau)$ or otherwise leaves the cloud system with probability τ. Further, the node CS is modelled as an M/M/1 queue with mean arrival rate ν and service rate μ. Hence, the mean response time T_{CS} is given by

$$T_{CS} = \frac{1}{\mu - \gamma} \tag{13}$$

To obtain the expected total response time in the system (including service times) for a request, we can simply add the expected waiting times at the respective facilities, because each request visits each facility exactly once. The total of expected response times $T_{ES} + T_{PS_1} + T_{PS_2} + T_{DS} + T_{OS} + T_{CS}$ of the proposed cloud as an open Jackson network represented by Figure 1 is thus estimated as

$$T_{cloud} = \frac{1}{\mu - \lambda} + \left(\frac{L_2}{p_1\gamma}\right) + \left(\frac{1}{\mu} + \frac{C(c, p_2\nu/\mu)}{c\mu - p_2\gamma}\right) + \frac{1}{\mu - \delta\gamma} + 2\left(\frac{1}{\mu - \gamma}\right) \tag{14}$$

It is remarked that the service rate of each node/server is defined as a ratio $\mu = S / F$ where S is the average bandwidth speed in bytes per unit time and F is the average size of the file containing response data. In real cloud systems, one can use open-source cloud environment to create/shut down virtual machines dynamically depending upon the traffic loads.

Figure 1 Cloud model connecting the clients with the database server (see online version for colours)

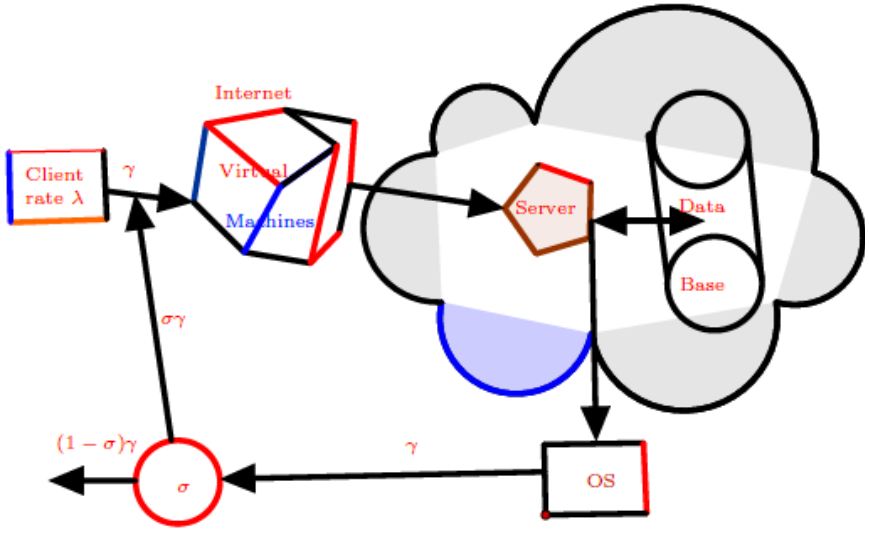
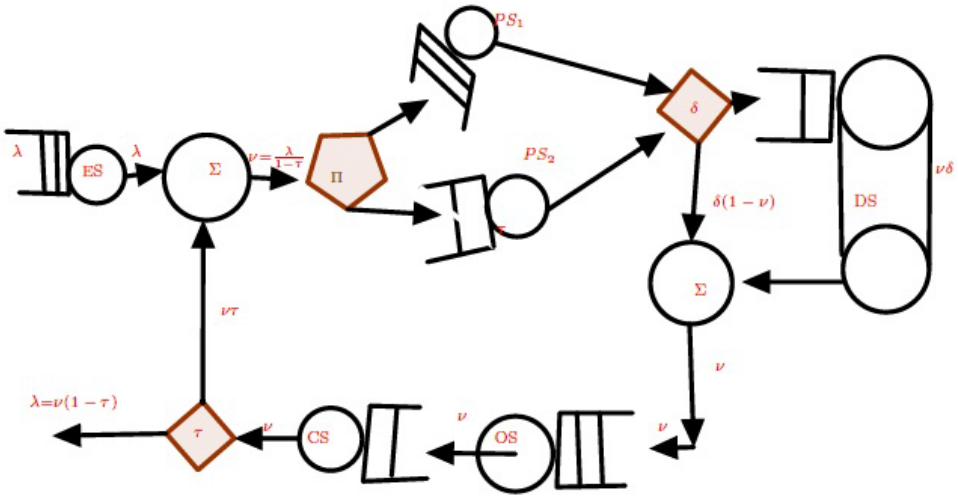


Figure 2 Proposed cloud computing model as open Jackson network with two virtual nodes at PS_1 and PS_2 (see online version for colours)



3 BMCM with PS_1 node as an M/M/1 service facility

Let us have a BMCM by replacing the processor PS_1 from the cloud platform described in Figure 2 under study by an M/M/1 model. The node PS_2 is retained as an M/M/c model.

M/M/c queue: Consider an M/M/c queue with mean arrival rate λ and service rate μ for each of the c identical servers. The probability of forcing an arriving customer to join waiting line when all c servers are occupied is known as Erlang's $C = C(c, \lambda/\mu)$ formula.

$$C(c, \lambda/\mu) = \frac{\left(\frac{(c\rho)^c}{c!}\right)\left(\frac{1}{1-\rho}\right)}{\sum_{j=0}^{c-1}\frac{(c\rho)^j}{j!} + \frac{(c\rho)^c}{c!}\left(\frac{1}{1-\rho}\right)}, \rho = \frac{\lambda}{c\mu} < 1 \tag{15}$$

The mean number L_c of customers in the system (queue + service) is

$$L_c = c\rho + \left(\frac{\rho}{1-\rho}\right)C(c, \lambda/\mu) \tag{16}$$

Thus, the mean response time $T_{PS_2}(c)$ of requests processed by the virtual machine M/M/c with arrival rate $(p_2\nu)$ and service rate μ, using the fact $\lambda = p_2\nu$ in equation (16), is

$$T_{PS_2}(c) = \frac{L_c}{p_2\gamma} = \frac{1}{\mu} + \frac{C(c, p_2\nu/\mu)}{c\mu - p_2\gamma} \tag{17}$$

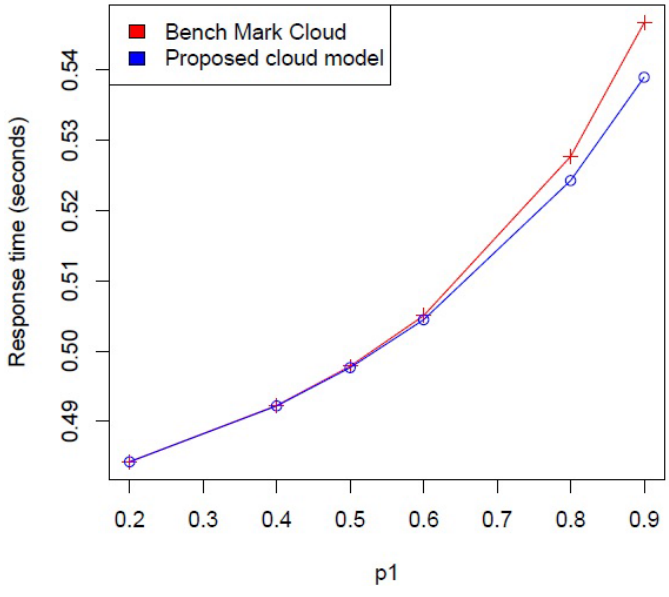
The total expected response time $T_{BMCM} = T_{ES} + T_{PS_1} + T_{PS_2}(c) + T_{DS} + T_{OS} + T_{CS}$ of the proposed cloud as an open Jackson network represented by Figure 2 is thus estimated:

$$T_{BMCM} = \frac{1}{\mu-\lambda} + \frac{1}{\mu-p_1\nu} + \left(\frac{1}{\mu} + \frac{C(c,\, p_2\gamma/\mu)}{c\mu - p_2\gamma}\right) + \left(\frac{1}{\mu-\delta\gamma}\right) + 2\left(\frac{1}{\mu-\gamma}\right). \tag{18}$$

4 Numerical illustrations

To support the results relating to the calculation of the expected total response times T_{cloud} of equation (14) and T_{BMCM} of equation (18), numerical illustrations are carried out by feeding a given set of input values (λ, μ, p_1, δ, τ). A comparative study is also conducted for a given set as $\lambda = 4.8$, $\mu = 28.4$ and $\tau = 0.8$ while the probability p_j of choosing the PS_j node (j = 1 and 2) by a request varies between '0' and '1' such that $p_1 + p_2 = 1$. The corresponding total response times T_{cloud} for the proposed cloud and T_{BMCM} for the benchmark cloud settings are computed and a graph showing the relationship between the probability p_1 and the response time is drawn in Figure 3.

Figure 3 Graph for p_1 vs. response times (see online version for colours)



Comparing the paths of the two options T_{cloud} of equation (14) and T_{BMCM} of equation (18) drawn in Figure 3, we see that each of response times increases with increasing values of the probability p_1. Further values of T_{cloud} are smaller than T_{BMCM} against each value of p_1. Thus, the cloud architecture producing smaller response time of computing requests is T_{cloud}. For achieving the best QoS from among the two options T_{cloud} and T_{BMCM}, T_{cloud} should be preferred over the benchmark cloud option T_{BMCM}.

5 Summary

By organising c identical servers with service rate μ at the node PS_2 and installing one more virtual machine with a state dependent service rate at the node PS_1, novel cloud architecture is designed in Figure 2 as a Jackson networking of queuing facilities. Using the properties of the Jackson network theory, response times for two types T_{cloud} and T_{BMCM} of cloud systems are computed and compared through a graphical illustration for a given set of input parameters. This comparative study establishes that the system of T_{cloud} produces lower responsive times as compared with T_{BMCM} system. The methodology applied here can be easily implemented by all types of industries who wish to undertake cloud computing with less expenditure.

References

El Aattar, M.B. and Haqiq, A. (2014) 'Performance modeling for a cloud computing center using GE/G/m/k queuing system', *International Journal of Science and Research (IJSR)*, Vol. 3, No. 5, pp.783–789.

Jaiganesh, M., Ramadoss, B., Kumar, A.V.A. and Mercy, S. (2015) 'Performance evaluation of cloud services with profit optimization', *Eleventh International Multi-conference on Information Processing (IMCIP), Procedia Computer*, Vol. 54, pp.24–30.

Khazaei, H., Misic, J. and Misic, V.B. (2012) 'Performance analysis of cloud computing centers using M/G/m/m+r queuing systems', *IEEE Transactions on Parallel and Distributed Systems*, Vol. 23, pp.936–943.

Oumellal, F., Hanini, M. and Haqiq, A. (2014) 'MMPP/G/m/m+r queuing system model to analytically evaluate cloud computing center performances', *British Journal of Mathematics and Computer Science*, Vol. 4, No. 10, pp.1301–1317.

Sivasamy, R., Thillaigovindan, N., Paulraj, G. and Paranjothi, N. (2019) 'Quasi-birth and death processes of two-server queues with stalling', *Opsearch* [online] http://doi.org/10.1007/s12597-019-00376-1.

Vilaplana, J., Solsona, F., Teixido, I., Mateo, J., Abella, F. and Rius, J. (2014) 'A queuing theory model for cloud computing', *J. Supercomputer*, Springer Science+Business Media©, New York, DOI: 10.1007/s11227-014-1177-y.

Xiong, K. and Perros, H. (2009) 'Service performance and analysis in cloud computing', *IEEE Computer Society*, pp.693–700, DOI: 10.1109/SERVICES-I.2009.121.