Asian Journal of Computer Science and Technology
ISSN: 2249-0701 (P) Vol.4 No.1, 2015, pp.40-43

DOI: https://doi.org/10.51983/ajcst-2015.4.1.1743

A Comparative Study of K-Erlang Distribution and M/M/1 Model in Cloud Computing

K.Ruth Evangelin and T.Dhikhi
Saveetha University, Chennai, Tamil Nadu, India
E-mail: ruthkavati@yahoo.co.in, dhikhi@gmail.com
(Received 12 April 2015; Revised 30 April 2015; Accepted 15 May 2015; Available online 23 May 2015)

Abstract - Cloud computing is a new emerging computing paradigm in which information and computer resources can be accessed from Web browser by the users. Arriving customers who find the server busy may retry for service after a period of time in Queuing systems is a Retrial queue. If a new customer, after connected to the cloud service does not find any server free, the system automatically redirects the request towards a waiting queue. At that moment, if that waiting queue is also fully occupied by other customers, then the newly arrived customer has to retry for service after certain time period. This technique is known as Retrial queues. Priority of service request is an important issue because some requests have to be serviced earlier than others. These are requests which can't stay for a long time in the queueing system. A comparative study on the waiting time of $M/E_k/n$ (n=1) and M/M/1 queuing system with cloud computing service station. This influences the queueing system to reduce the mean waiting time.
Keywords: Cloud Computing, Erlang Distribution for K-phases, Waiting time, Queue length, priority based service class, M/M/1 model.

I.INTRODUCTION

The Cloud computing is one of the emerging modern technologies that need to meet the emerging business needs for agility, flexibility, cost reduction and time-to-value. The developments in cloud computing paradigm necessitate faster and efficient performance evaluation of cloud computing servers. The advanced modeling of Cloud servers are not feasible one, due to nature of cloud servers and diversity of user requests. Queuing systems in which arriving customers who find the server busy may retry for service after a period of time is called Retrial queues[1,2]. In our system there are priority classes, which define the order in which the service will be provided, since scheduling the queue in this manner may cause starvation, therefore we have implemented ageing technique to prevent this. If a new customer does not find any free server after connecting to the cloud service, then the system automatically redirects the request towards a waiting queue. At that moment, if the waiting queue is also fully occupied by other customers, then the newly arrived customer has to retry for service after certain time period. This issue necessitates calculating performance of cloud servers and should be capable of handling several millions of requests in few seconds. It is so possible through the specialized computing facility called Virtualization and even with it, very hard to serve plenty of requests continuously or simultaneously without delay or collision. We proposed a new mechanism to overcome these performance problems and to aim for optimize their performance; we can achieve this goal with help of queuing theory. To tackle the problem of optimizing the performance of cloud servers [3], we model the cloud servers with architecture as an M/M/1 and $M/E_k/1$[4,5] system with single task arrivals and a task buffer of finite capacity. In this model focusing on input buffer size, number of servers, mean number of request, response and etc. as metrics for performance. The proposed algorithm named approximate Analytical model, and is equivalent to the combination of Transformation based Analytical Model & an Approximate Markov Chain Model with supporting calculations works out to bring the optimized performance evaluation of cloud servers.

II.PROPOSED WORK

In this section, a priority based single service channel queuing system cloud model $M/E_K/1$ has been proposed. Consider a single server retrial queuing system in which customers arrive in a Poisson process with arrival rate λ. These customers are identified as primary calls**.** Further, assume that negative customers arrive at a rate ν which follows a Poisson process**.** Gelenbe [6] has introduced a new class of queuing processes in which customers are either Positive or Negative. Positive means a regular customer who is treated in the usual way by a server. Negative customers have the effect of deleting some customer in the queue. In the simplest version, a negative arrival removes an ordinary positive customer.

Let k be the number of phases in the service station. Assume that the service time has Erlang-k distribution [7] with service rate $k\mu$ for each phase. We assume that the services in all phases are independent and identical and only one customer at a time is in the service mechanism. If the server is free at the time of a primary call arrival, the arriving call begins to be served in Phase 1 immediately by the server then progresses through the remaining phases and must complete the last phase and leaves the system before the next customer enters the first phase. If the server is busy, then the arriving customer goes to orbit and becomes a source of repeated calls. This pool of sources of repeated calls may be viewed as a sort of queue. Every such source

produces a Poisson process of repeated calls with intensity σ.

If an incoming repeated call finds the server free, it is served in the same manner and leaves the system after service, while the source which produced this repeated call disappears. Otherwise, the system state does not change.There are 'k' numbers of phases in our cloud system. Basic model of single server is shown in Fig. 1. Number of phases is four, if service is not available; where as if service is available, then number of phases becomes five. Only one server is considered in this case. We have considered three priority classes. The buffer is assumed to be infinite in each class. The priority based queuing discipline is first-come-first-serve (FCFS). User's requests are being served one at a time. A new service with same priority could not be started until all the k-phases have been executed. Therefore, each arrival of users' requests increases the number of phases by k in the overall system. A mathematical basis for queuing theory [8] is provided for better understanding and for higher prediction based on the behavior of communication network. We assume that the access from the queue to the service facility follows the exponential distribution which may depend on the current number n,($n \geq 0$) the number of customers in the orbit. That is the probability of repeated attempt during a particular interval, given that there are n customers in the orbit at time t is $n\sigma\Delta t$. It is called the classical retrial rate policy. The input flow of primary calls, interval between repetitions and service time in phases are mutually independent.

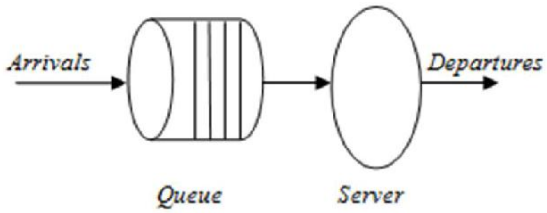


Fig. 1 Basic Single Server Queueing Model

λ=Arrival rate of customers

$\mu = Expected\ number\ of\ customers\ completing\ service\ per\ unit\ time$

III. MODEL M/E$_K$/1

$$Expected\ number\ of\ phases\ in\ the\ system = L_s(k) = \frac{k+1}{2} * \frac{\lambda}{\mu - \lambda}$$

Expected number of customers in the system = $L_s = L_q + \frac{\lambda}{\mu} = \lambda W_s$

Expected number of customers in the queue =

$$L_q = \frac{L_s(k) - average\ number\ of\ phases\ in\ the\ service}{} = \frac{1}{k} * \left[\frac{k+1}{2} * \frac{\lambda}{\mu - \lambda} - \frac{k+1}{2} * \frac{\lambda}{\mu}\right] = \frac{k+1}{2k} * \frac{\lambda^2}{\mu(\mu - \lambda)}$$

Expected waiting time of a customer in a queue = $W_q = \frac{L_q}{\lambda} = \frac{k+1}{2k} * \frac{\lambda}{\mu(\mu - \lambda)}$

Expected waiting time of a customer in the system $= W_s = W_q + \frac{1}{\mu}$

IV. (M/M/C):(∞/FIFO) QUEUING MODEL

It is assumed that , if CCU arrives at an average rate λ and server has service mean rate μ and finds the server in busy state then CCU has to wait till the server completes its job or CCU may enter into Balking or Reneging state .This results increasing in waiting time and queue length . Therefore in order to overcome this problem **(**M/M/C): (∞/FIFO). Queuing model is applied when there are multiple servers , C and each server has an independent identical exponential service time distribution n. The arrival process assumed to be poisson .and ∞ indicates CCU.

The mean service = rate will be Cμ . The steady state probabilities are:

Measures of effectiveess :

Where [L_q] : Expected number of customers in queue.

[L_s] : Expected number of customers in the system.

[Wq] : Expected waiting time per customers in the queue

[W_s] : Expected waiting time per customers in the system

If the no. of servers is c = 1, then it turns out to be a single server.(i.e) an M/M/1 queuing model, where we can calculate the performance measures such as

The probability of zero customers in the system is $P_0 = \frac{1}{\frac{\lambda}{\mu})^n}{n!}}$

The probability of n customers in the system is $P_n = \frac{(\frac{\lambda}{\mu})^{n_e - (\frac{\lambda}{\mu})}}{n!}$

Average number of customers in the system is $[L_s] = (\frac{\lambda}{\mu-\lambda})$

Average waiting time at the system is $[W_s] = (\frac{1}{\mu-\lambda})$

if there are c = 2 servers or c = 3 servers, the generalized formula is given as

$$[L_q] = \frac{\rho^{c+1}}{(c-1)!(c-\rho)^2} P_0$$

$$[L_s] = [E_q] + \rho$$

$$[W_q] = \frac{[Eq]}{\lambda}$$

$$[W_s] = [Cq] + (\frac{1}{\mu})$$

V.NUMERICAL EXAMPLE

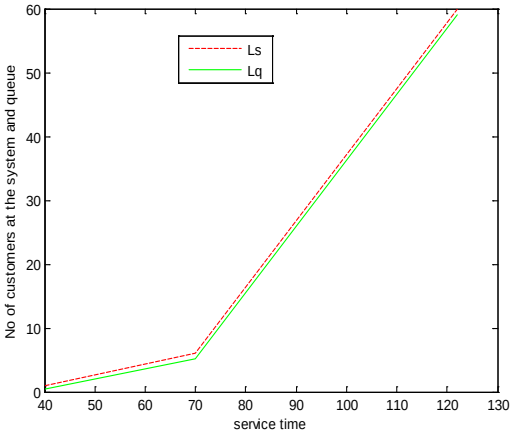
Consider two Queueing models (M/ E_k /1): (∞/FIFO), (M/M/1): (∞/FIFO) (with arrival rate $\lambda = 20, 60, 120$ *and service rate* $\mu = 40, 70, 122$). We have found out the performance measures such as the waiting time at the system $[W_s]$ and the number of customers at the system $[L_s]$, Number of customers in the queue$[L_q]$ and Waiting time of a customer in a queue $[W_q]$ numerically.

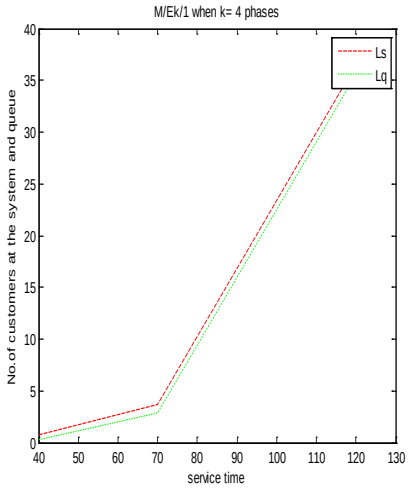
The tabular column is listed below.

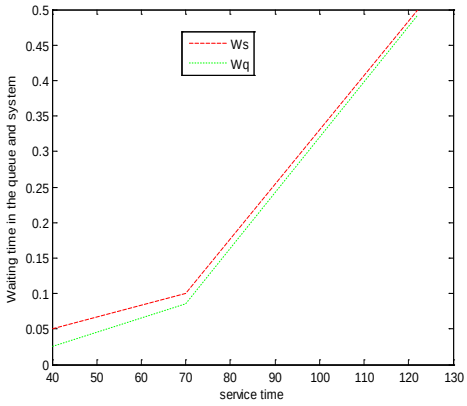
Model	λ	μ	$[L_s]$	$[L_q]$	$[W_s]$	$[W_q]$
M/M/1	20	40	1	0.5	0.05	0.025
	60	70	6	5.14	0.1	0.085
	120	122	60	59	0.5	0.492

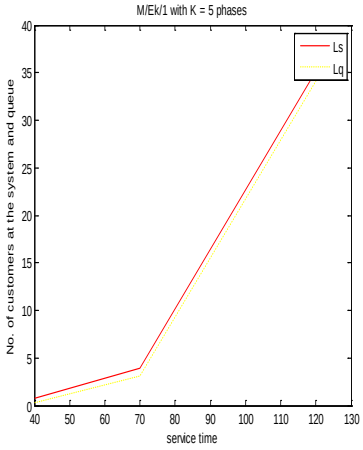
Model $M/E_k/1$	when k = 4 phases						When k = 5 phases			
	λ	μ	$[L_s]$	$[L_q]$	$[W_s]$	$[W_q]$	$[L_s]$	$[L_q]$	$[W_s]$	$[W_q]$
	20	40	0.8125	0.3125	0.0875	0.0625	0.8	0.3	0.04	0.015
	60	70	3.714	2.857	0.0683	0.054	3.942	3.085	0.066	0.0514
	120	122	37.868	36.885	0.3156	0.3074	36.393	35.409	0.3033	0.2951

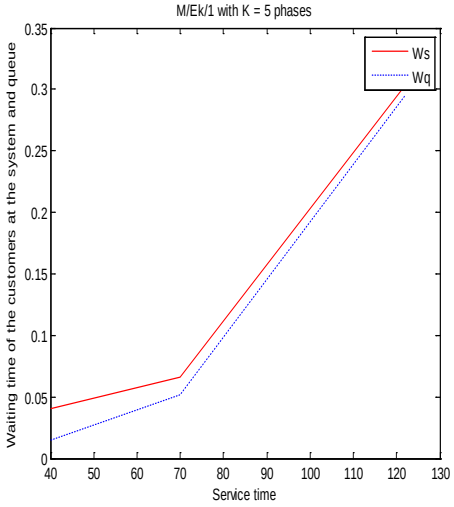
VI. Graphs

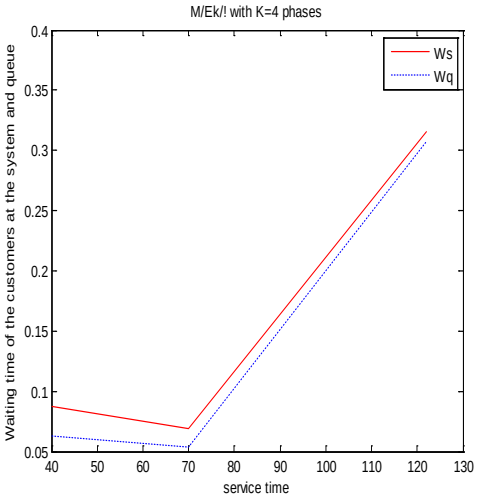












VII. CONCLUSION

From the above numerical results, we are able to analyse that the Erlang distribution gives a better result as the waiting time, the queue length is also reduced reduced, when it is compared with M/M/1 model . $M/E_k/1$ has produced better results when compared to M/M/1 interms of average queue length and average waiting time. Virtual machines[11] are modeled as server centers using $M/E_k/1$. Numerical results have demonstrated high performance which regard to $M/E_k/1$ when compared to M/M/1.

REFERENCES

[1] J. R. Artalejo, "A queueing system with returning customers and waiting line," Operations Research Letters, Vol. 17, pp. 191-199, 1995

[2] G. I. Falin, "A survey of retrial queues," Queueing Systems, Vol. 7, 127-167,1990.

[3] Anirban Kundu, Chandan Banerjee, Priya Saha, "Introducing New Services in Cloud Computing Environment," International Journal of Digital Content Technology and its Application, Vol. 4, No. 5, 2010.

[4] M. Jain, P. K. Agrawal, "$M/E_k/1$ Queueing System with Working Vacation," ICAQM, Vol. 4, No. 4, pp. 455-470,2007.

[5] Chandan Banerjee, Anirban Kundu, Ayush Agarwal, Puja Singh, Sneha Bhattacharya, Rana Dattagupta; "K-phase Erlang Distribution method in Cloud Computing"; Fourth International Conference on Advances in Communication Network and Computing – CNC 2013; LNICST pp. 53~59.

[6] E. Gelenbe "Product-Form queueing networks with negative and positive customers", Journal of Applied Probability, Vol. 28 (3): 656–663 (Sep., 1991).

[7] Donald Gross and Carl M. Harris, Fundamentals of Queueing theory, 1974.

[8] S. Sowjanya, D. Praveen, K. Satish, A. Rahiman, "The Queueing Theory in Cloud Computing to Reduce the Waiting Time," IJCSET, Vol. 1, Issue 3, pp. 110-112,2011.