Using Queuing theory the performance measures of cloud with infinite servers

A.Anupama
Department of Information Technology
GMR Institute of Technology
Rajam, India
anupama.a@gmrit.org

G.Satya Keerthi
Department of Information Technology
GMR Institute of Technology
Rajam, India
Satyakeerthi.g@gmrit.org

Abstract-Cloud computing has got enormous popularity as it offers dynamic, low-cost computing solutions. To get the service of cloud the user has to be in queue until he is served. Each arriving Cloud computing User (CCU) requests Cloud computing Service Provider (CCSP) to use the resources, if server is available, the arriving user will seize and hold it for a length of time, which leads to queue length and more waiting time. A new arrival leaves the queue with no service. After service completion the server is made immediately available to others. From the user's point of view he needs to be served immediately and to prevent waiting the CCSP's can use infinite servers to reduce waiting time & queue length. The arrival pattern is often Poisson in queuing theory. In this article we analysed the dynamic behaviour of the system with infinite servers by finding various effective measures like response time, average time spend in the system, utilization and throughput.

Keywords-cloud computing, stochastic process, poisson process, queueing theory, waiting time.

I. INTRODUCTION

Traditionally cloud computing is a single system serving the end users, where the service does not depend on physical locations or configuration. With abundant requests, degrades the performance of traditional approach.

With infinite number of servers we can merge several low cost computing units to one higher level system with strong computing ability and deliver some specified techniques (Infrastructure as a Service (IaaS), Platform as a Service (PaaS), and Location as a Service (LaaS)) to end users. The approach we followed in this paper is as follows. Every arrival is a random or stochastic which are independent and identical, and the process is poisson. We also interested in the time each user spends the system called as service time which is also random. The common distribution for service time is exponential. The arrival time and the service time is a collection of random variables, called stochastic process which obeys memory less or markov property in which the state of a process are independent of the past and depend only on the present. The markov property makes a process easier to analyse since we do not have to remember the complete past trajectory.

If the number of possible values is finite or countable, the process is called a discrete-state process. For example, the number of jobs in a system n(t) can only take discrete values 0,1,2…The waiting time $w(t)$, on the other hand, can take any value on the real line. Therefore, $w(t)$ is a continuous-state process. A discrete-state stochastic process is also called a stochastic chain. The discrete-space Markov processes in which the transitions are restricted to neighbouring states only are called birth-death processes. The number of jobs in a queue with a single server and individual arrivals can be represented as a birth-death process. An arrival to the queue (a birth) causes the state to change by +1 and a departure after service at the queue (a death) causes the state to change by – 1.

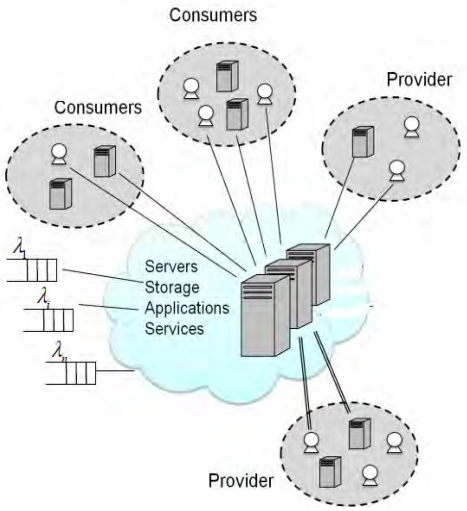
Stochastic process representation is given for the single and infinite server process of queuing in mathematical manner. Next by using Kendall's notation the queuing models are represented for single server and infinite server. Numerical results and graphs are used finally for showing the results.

II. CLOUD COMPUTING

Software as a Service (SaaS) places the separation very high in the stack where the customer is simply an end-user and requires only a web browser. For example, a web-based email client is a SaaS where the customer does not need to install or configure their email client. All the details about mail transfer, delivery, storage, and spam filtering are provided to the customer as a service with some relationship agreement in place.

Platform as a Service (PaaS) moves the separation lower in the stack where the customer is a software developer. As with SaaS, the PaaS customer does not want to pay attention to the underlying details of purchasing a computer, installing the operating system, selecting a web server, and gathering all the dependencies for running the software. Rather, the software developer wants the entire platform provided as a service to use, allowing the focus to be solely on the software being developed.

Infrastructure as a Service (IaaS) takes the separation even lower and provides the developer with the physical infrastructure needed to provide a service. Thus, the service provider controls the physical resources (networking equipment, connectivity, and physical hardware), while the developer would have control of anything above those resources. The provider typically allows the developer to create virtual machines on the physical resources, giving the developer complete control of the system from the choice of operating system to the choice of web environment.



We assume that the arrival of requests in cloud follows a Poisson process with parameter λ where 1/ λ is the mean inter arrival time and 1/ μ is the mean service time and these are assumed to be statistically independent. The postulates of the cloud are

1. Pr{an arrival occurs in an infinitesimal interval of length $\Delta t\}$ = $\lambda\Delta t+o(\Delta t)$
2. Pr{more than one arrival }= $o(\Delta t)$
3. Pr{a service completion }= $\mu\Delta t+o(\Delta t)$
4. Pr{more than one service }= $o(\Delta t)$

III. QUEUEING THEORY

We need to specify these six parameters. Queueing theorists, therefore, use a shorthand notation called the **Kendall notation** in the form A/S/*m*/*B*/*K*/SD, where the letters correspond in order to the six parameters listed above.

A is the inter arrival time distribution

S is the service time distribution

m is the number of servers

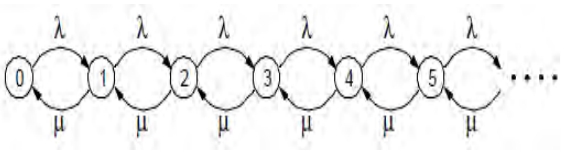
B is the number of buffers

K is the population size,

SD is the service discipline.

A. Stochastic description for M/M/1 MODEL

The simplest queueing model is one that has only one queue. The state transition diagram of a birth-death process is



{N(t), t ≥0} is the number of customers is a stochastic process, since inter arrival time and service time is exponentially distributed and the memory less property satisfied.

Probability of one arrival is $\mathbf{q_{i,i+1}= \lambda}$

Probability of one departure is $\mathbf{q_{i,i-1}=\mu}$

The steady-state probability p_n, the probability of n customers in the system at an arbitrary point of time after steady state is reached, we take the limit as t->∞.

$$\mathbf{0=- \lambda \pi_0+\mu\pi_1}$$

$$\mathbf{0= \lambda \pi_{i-1}-(\lambda +\mu)\pi_i+\mu\pi_{i+1}}$$

Since it is a homogenous equation we get πi's in terms of π0

$$\mathbf{\pi_1= \lambda /\mu \pi_0}$$

$$\mathbf{\pi_{i-1}= \lambda /\mu \pi i= \lambda^{i-1}/ \mu^{i+1} \pi_0}$$

The server utilization can be calculated by

$$\mathbf{\rho=\lambda/\mu} \qquad \mathbf{(1)}$$

The probability of zero in the system means server idle time

$$\mathbf{P0=(1- \rho)} \qquad \mathbf{(2)}$$

The probability of n customers in the queue is

$$\mathbf{Pn = (1- \rho) \rho^n} \qquad \mathbf{(3)}$$

Mean number/expectation of participants in queue system is

$$\mathbf{L=\rho/(1- \rho) =\lambda/(\mu-\lambda)} \qquad \mathbf{(4)}$$

Mean queue length is

$$\mathbf{Lq=\lambda^2/\mu(\mu-\lambda)} \qquad \mathbf{(5)}$$

Mean staying time is

$$\mathbf{W=1/(\mu-\lambda)} \qquad \mathbf{(6)}$$

Mean waiting time is

$$\mathbf{Wq=\lambda/\mu(\mu-\lambda)} \qquad \mathbf{(7)}$$

Utilization is

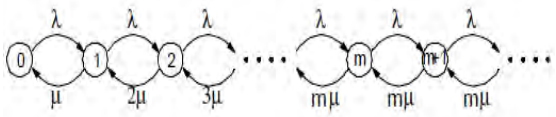
$$\mathbf{U(t)=1-p0(t)} \qquad \mathbf{(8)}$$

Throughput is

$$\mathbf{service\ time * utilization} \qquad \mathbf{(9)}$$

B. Stochastic description for M/M/∞ MODEL

The person enters in the system immediately gets the service. We make use of birth-death results with arrival rate ʎn=ʎ and μn=nμ.



The probability of zero in the system is

$$\mathbf{P0=1/\left(\frac{\lambda}{\mu} \right)^n} \qquad \mathbf{(10)}$$

$$\mathbf{n!}$$

The probability of n customers in the queue is

$$\mathbf{Pn = \frac{(\lambda/\mu)^n}{n!} e^{-(\lambda/\mu)}} \qquad \mathbf{(11)}$$

Mean number/expectation of participants in queue system is

$$\mathbf{L =\lambda/\mu} \qquad \mathbf{(12)}$$

Mean staying time is

$$\mathbf{W = 1/ \mu} \qquad \mathbf{(13)}$$

Since we have as many servers as customers in the system **Lq=0 and Wq=0.** There no length of the queue and waiting time in the queue in infinite servers.

IV. RESULTS

Arrival Rate(λ)=2

Depature Rate(μ)=3

Queueing Discipline=FIFO

A. Performance analysis between M/M/1 and M/M/∞

Model	W	L	Throughput
M/M/1	0.99	1.70	1.70
M/M/∞	0.45	0.89	1.89

Table 1: Measures of throughput and length and waiting time.

B. M/M/1 Model

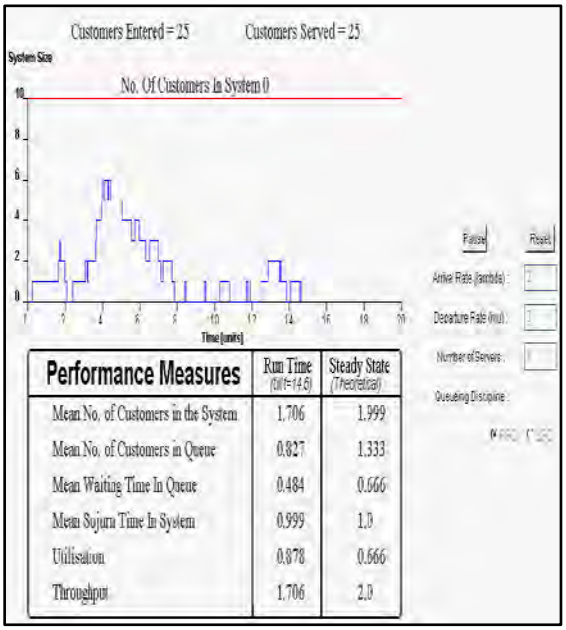
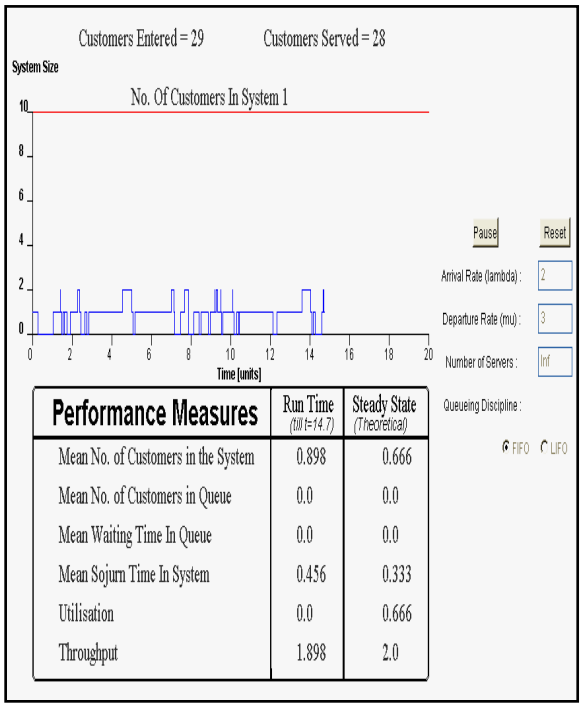


Figure 1: Performance measure of single server system

C. M/M/∞ MODEL



Performance Measures	Run Time (till t=14.7)	Steady State (Theoretical)
Mean No. of Customers in the System	0.898	0.666
Mean No. of Customers in Queue	0.0	0.0
Mean Waiting Time In Queue	0.0	0.0
Mean Sojurn Time In System	0.456	0.333
Utilisation	0.0	0.666
Throughput	1.898	2.0

Figure 2: Performance measure of infinite server system

V. CONCLUSION

In this paper we used stochastic process to analyse the dynamic behaviour of infinite severs over single server. We studied the utilization factor, throughput, length of server, waiting time of infinite server system. From the user point of view he gets service immediately there is no need to be in queue for service. With good selection of number of servers in infinite server system we can reduce queue length and increase throughput and utilization.

REFERENCES

[1] Lotfi Tadj "Waiting in line", IEEE POTENTIALS 1996.
[2] Sonam Rathore, "Efficient allocation of virtual machine in cloud computing environment", International journal of computer science and informatics, Vol.2, Issue 3, 2012, 59-62.
[3] Gross. D. and Harris (1985), "Fundamentals of queueing theory", John Wiley, New York.
[4] R.W. Wolff, "Stochastic Modelling and the Theory of Queues" (Prentice-Hall, Englewood Cliffs, NJ, 1989).
[5] Ross, S.M. 1996," Stochastic Processes." JohnWiley & Sons, Inc., New York.
[6] T.Sai Sowjanya, D.Praveen, .Satish,A. Rahmain, "The Queueing Theory in Cloud Computing to Reduce the waiting Time", IJCSET ,April-2011.
[7] Chu, J. T. and Sedaghat, M., Queuing Approaches to Stochastic Demand/Supply Systems, AIDS Proceedings, pp. 740-742, 1984.