International Academic Institute for Science and Technology

International Academic Journal of Science and Engineering Vol. 2, No. 2, 2015, pp. 109-120.
ISSN 2454-3896

International Academic Journal of Science and Engineering

www.iaiest.com

Evaluation of performance in Cloud Computing with Queue Theory

Saeed TalatianAzad[a] , Mohammad Malekzadeh[b]

[a] *Department of Computer Engineering, Faculty of Engineering, Persian Gulf University*.

[b] *Department of Computer Engineering, Faculty of Engineering, Persian Gulf University*.

Abstract

With information technology progression there is the need to perform computing tasks everywhere and all the time. It is also required that people be able to do their heavy computing tasks without having expensive hardware and software to perform the service. Cloud computing was the last answer of technology to these needs. Cloud computing is a model to provide easy access based on user demands to changeable and configured computing resources through the network (e.g., networks, servers, storage, applications and services) that the access could be provided or be released quickly with minimal need to manage resources or the need for the service provider's direct intervention. Cloud computing can be known as a new context to provide computing infrastructure, with the goal to transfer computational infrastructure to a network So that hardware resources and software maintenance costs reduced. Cloud computing systems are capable of providing access to a large pool of resources. Resources which are provided in cloud computing systems, a large size of services are hidden from the user by visualization. In this paper, we investigate the cloud service data center in form of row system with a buffer and an input buffer task which is virtually unlimited capacity.

Keywords: networking, cloud computing, performance analysis, response time, queue theory, Markov Chains

Introduction:

Cloud computing can be a form of Web-based development and use of computer knowledge. This service has become a major IT platform in recent years. Cloud computing is define as other similar terms such as software as a service (SaaS), grid computing, cluster computing, independent computing, application computing [Kim, 2009]. Saas, or software as a service, is a delivery model software in which software and its associated data are hosted centrally located on a cloud. SaaS is generally available by users who uses a Web browser from a client. Grid computing and cluster computing are also two types of computer technology, developing cloud computing. Independent computing services also includes computer system which is capable of self-management and computing applications include packaging of computing resources such as computing and storage devices [Vaquero, 2009]. Cloud centers are different based on a series of important aspects and row systems. Time of servicing the task should be the modeled generally, but do not model it as a probability function and exponential. The variance coefficient of service timing in a task may be more than one. Due to the dynamic nature of cloud environments, a variety of user's requests and time dependence in workloads has caused Cloud centers to be able to provide expected service quality for different loads [Xiong, 2009]. [Baker et al, 2011].

Developing a cloud computing platform has led to necessity of correct evaluation of the Cloud centers data efficiency. Since accurate modeling of Cloud centers can't be useful due to the nature of centers and a variety of user demands, here's a new full analytical model estimation of the probable distribution in cloud server, in order to obtain an accurate estimation of the full probable distribution of the response time to requests, and other key performance indicators. This model has allowed cloud operators to determine the server and the input buffer size on the one hand the relationship between, while performance indicators such as number of tasks in the system are likely to block, and on the other hand the likelihood of a task receive immediate service will be also evaluated. A key advantage of having multiple servers in a cloud computing environment is that the system performance increased by reducing the length of row and anticipating time reduces compared to conventional methods where there was only one server, in a way that customers do not need to wait for a long time period and rows does not need to have a large volume.

In addition to the theory of services (processes) to evaluate random arbitrary commands from are called row, queue theory can also create mathematical models to study the history of the network. During calculation system, we can reveal statements about the possibility of filing systems and information systems to decide the best plan. Assuming queue theory to estimate network traffic, this theory is a significant method of calculating network performance, evaluation and it's conclusion, in this way, we

can produce a real system, the method and interoperable, to organize, surveillance (monitoring), and to defend the network.

In this paper, the cloud center have been investigated as a row system (M / G / 1) and with a buffer request and a buffer task input and with limited capacity. Also we consider the row system performance was evaluated, using an analytical model in order to get important performance factors such as average number of tasks in a system.

Queue theory

We can find an easy row model in the communications network, sending and receiving data packets, data sequence, interpretation and distribution to the upper layer, in all this process. According to queue theory, this communication can be focused on as an example of queue theory as shown in Figure 1. According to this system, easy release of the data are consistent to the model lineup.

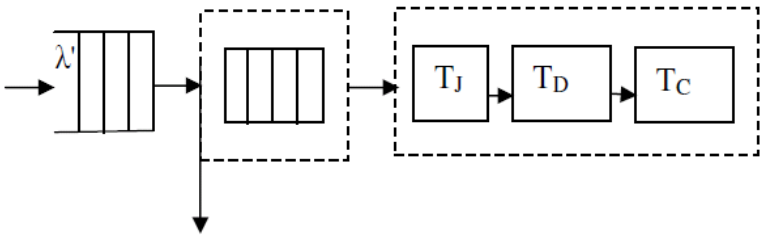

Figure 1 - queue theory model

Ts=TJ+TD+TC ..... (Equation 1)

Parameter	Description
λ'	Distributors rate transmitter
TN	Transmission delay time
Λ	Packet data reciving speed
Nq	The number of data packets stored in the buffer (temporary storage)
Y	Packet error rate in the post from receiver
Ts	Data packet service time
TJ	Decoding time
TD	Distribution time (dispatching)
TC	Calculation time
A	Time of arrival (between)
M	The rate of maintenance (services)

Arrival rate, λ, the average rate of arrival of new nodes at each time interval to be measured.

Access common units are second (access / second). The time between the arrival, a, is the average time between arrival of the nodes. This is measured at each node. A joint unit of Second will be available.

a = 1 / λ …… (Equation 2)

Row systems are usually described by three values separated by a forward slash (/).

Distribution / Distribution Services (Maintenance) / number of servers

- A = Markovian or exponential distribution (Markovian for exponentially distributed)
- D = a definite or fixed (Deterministic or constant)
- G = General or binomial distribution (General or binomial distribution)

With calculating maintenance rate, the packet data rate in heterogeneous network environment, the cloud efficiency proposal in heterogeneous environment using queue theory, through the use of differential equations is done. Parameters Table (Table 1) used in queue theory concepts described in heterogeneous environments.

Queue theory and traffic performance heterogeneous network

Network traffic is very common in heterogeneous environments. This structure will be in the form of lower (low-grade) when the heterogeneous network traffic is placed at the extreme, it leads to network congestion. Now a lot of contract studies are detected on network congestion, in addition, articles which are used queue theory to explore heterogeneous network traffic are more and more. To forecast traffic rate, we often test optional performance data that are used in network routers. Considering that λ, the rate of data flow groups to the router, and the average time that routers use to sort each group is μ / 1, router's buffer is shown with B, if a certain group (data) reaches, and the groups waiting list also identified, so it causes the loss of the group. When the arrival of groups are all finished (timeout happened), groups must re-send the data packet again. Assuming that group's average waiting time is μ / 1, we considered Pi (1) the possibility for routers row length at t time, and we assume that the length of the row is i (Prasadh and Senthilkumar, 2012).

P (t) = (P0 (l), P1 (l), . , Pi (l)), i = 0,1, . . . B+1 …… (Equation 3)

The following table describes parameters used in queue theory and heterogeneous network traffic control plan.

Table 2 – Parameters

parameters	Description
B	Buffer space in the routers
I	Row length
P(t)	Data packet receiving speed chance at t time
μ1/	Waiting time average
AC (t)	Density rate
Pn(t)	n data packet receiving speed chance at t time

Pn(t+Δt)	data packets receive speed chance at t+Δt moment
o(Δt)	Data packets arrive at Δt moment

Queue theory and congestion rates review

Network congestion rates vary all the time. Moment density rate and constant density rate are used alternatively to monitor network traffic in heterogeneous network. Moment density rates AC (t) is rate compression at t time. AC (t) can be obtain by Length probability distribution line system explanation, which is called Pn + 1 (t).

Pn (t) (n=0, 1, . . . ,i+1) is the probability of receiving from row time range for routers set at time t, if the time limit is n row. As a result, the router with the simple Markov row system matches the collection. Compared to process Markov, Pn (t) satisfied differential equations disputes related to future systems.

(Equation 4) {n data packages in the system at time t} possibility = Pn(t)

(Equation 5) {n current data packages in the system at time t + Δt} possibility = Pn(t+Δt)

Type 1:

For n> = 1

$$P_n(t+\Delta t) = P_n(t) * \{1-\lambda_n \Delta t + o(\Delta t)\} * \{1-\mu_n \Delta t + o(\Delta t)\} + P_{n-1}(t) * \{1-\lambda_{n-1}\Delta t + o(\Delta t)\} * \{1-\mu_{n-1}\Delta t + o(\Delta t)\} + P_{n+1}(t) * \{1-\lambda_{n+1}\Delta t + o(\Delta t)\} * \{1-\mu_{n+1}\Delta t + o(\Delta t)\} + o(\Delta t)$$

(Equation 6)

Type 2:

For n = 0

$$P_0(t+\Delta t) = P_0(t) * \{1-\lambda_0 \Delta t + o(\Delta t)\} + P_1(t) * \{1-\lambda_1 \Delta t + o(\Delta t)\}$$ …….. (Equation 7)

Type 3:

For n = C+1

$$P_B + 1(t+\Delta t) = P_B(t) * \{\lambda_B \Delta t + o(\Delta t)\} * \{1-\mu_B \Delta t + o(\Delta t)\} + P_{B+1}(t) * \{1-\mu_{B+1}\Delta t + o(\Delta t)\}$$ … ….. (Equation 8)

Momentarily Rates of heterogeneous network congestion can't be used to estimate the steady operation system, so it is necessary to obtain a fixed rate system density. The constant density rate means that when the mechanism system is in a state of constant operation at different times, it will not change. Consonant congestion rate classification is as follows

$$A_c(t) = \lim_{t\to\infty} A_c(t) \ldots$$ (Equation 9)

Considering the distribution of the constant row and C as the router protector, the constant density rate can be achieved in two ways: First, we have earned a moment density rate, and then we create the range

of it. According to its category, which can achieve distributing row. The second method, is a distribution which is done through the equation of sustainable state.

Based on the data packet receiving speed, network service rate evaluated and traffic rate is determined for testing the packet loss. If congestion occurs in the network, row length is calculated based on the compression rate and the probability of data packet-based homogeneous network traffic input and output can be calculated from heterogeneous network environments. According to the description of row system, heterogeneous network traffic controlled efficiently. (Prasadh and Senthilkumar, 2012)

Performance evaluation methods

Cloud computing services can be a complete software environment to the user. This environment can provide us compute resources such as energy, bandwidth and storage capacity. This service has been considered in large scale, but only a small portion of these activities have addressed issues related to performance and analytical approach in this regard, as only a handful offered. The response time can be known one of the major constraints in the row system, because the response time for a model distribution cloud center is provided as a M / M / m / m + r row system is in this system, time service and inter-arrival time was assumed to be distributed exponentially and system has a buffer, limited to the size of m + r. The response time is also divided a number of the waiting time, servicing and performance period, with the assumption that each interval is independent, although this independence is unrealistic, according to the theory presented in [Yang et al, 2009].

The time between arrival and / or servicing time is not exponentially and they has great complexity. Many theoretical analysis to evaluate the performance of the M / G / m row system is very reliant on research. Although the probability distribution of response time and row length of the M / G / m can't be obtained in closed form so it makes the research necessary for a good approximation of the [Yao, 2010]. In [Yao, 2010], the authors has been measured the average response time of a service request in [Martin and Nilsson, 2002], but the measurement techniques are difficult to use for predict computer service performance. In order to calculate the percentage of response times, you must first obtain the probability distribution of response times. Of course, doing so in a complex computing environment that includes a lot of computing nodes is no easy task.

They have considered a M / G / 1 related to the situation in which the interval and distribution of service time are depending on the amount of system's unfinished work. They has been used external methods to obtain estimates for the number of scales related to unfinished business and average occupation period in such an array of [9]. An important consideration is that every customer arrival time (depending on the size of the corresponding Markov chain) can be considered as entering of a high customer rating, which

the service time is carried out as the sum of the service requests from PC in [11]. A comparison has been made between all these methods that focus on obtaining reliable estimates of the expected performance of different architectures and depending on workflows. However, none economic comparison costs has been shown between the different platforms. Some of the works that has been done in this area, are based on Desktop workstations, clusters of local and shared resources of the HPC and according to scientific works times in [13], respectively.

Models and assumptions

Probabilistic Markov chain can be characterized as follow:

Markov chain is called probabilistic if it's not reduce able, does not have Null value and don't be recurring. We also define dialog as follow, conversational state i to j shown as i → j if the chain met j status with positive probability and starts from i. That, i → j, if pij (n)> 0 (for n≥0amounts). We say that i and j have internal dialog with each other, if i → j and j → i that in this case we write: i↔j. It can be seen that ↔ is an equivalent relationship and therefore the S status space can be divided into classes of equivalence of ↔; inside each equivalent class, all items are the same kind.

A C set of items are labeled as follow:

a. If pij equal to zero for all i∈C, j∈C, it is called a closed set.
b. If we have i↔j for all i, j∈C, this set is called an irreducible set.
c. Mr. Kendall Categories for row systems are available in several manipulation versions. Row models generally provide to propose a uniform status of row systems, i.e. the medium or long implementation of the system. As a result, this can be known as Stochastic models that show the probability of finding a row system in a configuration or specific situation. A common procedure for creating and analyzing such a row models are as follow:

1. System parameters diagnosis, such as time entry, time of servicing, row capacity and perhaps charting the system.
2. System position diognosis (generally involves a good number of clients, people, tasks, contacts, messages, etc. in the system that may / or may not be limited.)
3. Draw a status change diagram which shows possible status of the system and rates for the entrance and exit of every status. This graph can be considered as a representation of a Markov chain.
4. Considering the plot situation shows a similar situation between items, there is a balanced flow between situations and thus the probability that the items are adjacent can be attributed to each other mathematically and based on the rate of entry and maintenance and situation's

possibilities.

5. Showing all the situations possibilities based on the probability of empty status using the relationship between the conversions state.
6. Determine the probability of an empty position, using this principal that all the sum of the possibilities are equal to 1.

M / M / 1, shows a single-server that has a unlimited row capacity and unlimited calling population, and both entry process and servicing can be known as the Poisson processes (random), in the sense that static distribution for both times between the arrival time and servicing time, follows an exponential distribution. Due to the math nature of exponential distribution, complete simple relationships can be gained performance based on a arrival rate and service rate knowledge. M / G / 1 also shows a single server with unlimited line capacity and unlimited calling population, while the entry process may still be considered a Poisson process, in the sense that the exponential distribution of arrival times, obeys the exponential distribution and servicing time distribution, does not follow the exponential distribution. Servicing time distribution may follow a static distribution, and not only exponentially. It can still achieve a relationship for the number of (limited) scales of performance if no one knows the arrival rate and servicing rate variance average.

Although the derivatives obtained are usually very complex and difficult. Since many of these results, are to obtain a solution in the form of the package, are based on estimates, they can't be applied universally.

1. Estimates are logically correct only when comparison is small in terms of the number of servers - usually under 10 servers - and that this makes them suitable for the analysis of non-performance cloud computing data centers.
2. Estimates are very significant to the probability distribution of service time to a task and so when the coefficient of variance of service time, Cov, drives to the amount of one or more, it provides an incorrect value.
3. Finally, the estimation errors are low when p traffic density, or when the number of m servers and CoV of service time is high, are shown. As a result, results that are outlined above can't be directly used on the cloud-server performance analysis, where one of the following is true: the number of servers are high; distribution of service time is not clear and, in general, does not obey any good distributions such as exponential distribution; in the end, traffic density can vary in a great level.

Standard quantity of customers in the system

The time between arrival and the time between preparations are assumed to be they exist in other row models so that the exponential distribution can be obtained with parameter λ and μ. Imagine if the time arrival and departure times, not follow the Poisson distribution, the study of other models would be difficult. But the formula can gained from specific non-Markov models in the form of [(M / G / 1) :(∞ / GD MODEL)], where, M represents the number of entries in that t time which follows a Poisson process, and G represents the distribution of common output and ∞ indicates that the expected space capacity is unlimited, and GD is also indicate current unlimited description.

Assume that the arrival time of a project obeys Poisson arrival rate λ. Also assume that the lead time for the random variables are independent and equally distributed with a probability distribution as desired. Suppose b (T) is a probability density function of the lead time between the two outputs. Suppose N (t) is also the number of customers in system at time t≥0. Suppose t_n was the time in which, n has completed its faithful customer service and exits. Assume that X_N shows the number of customers in the system when the n faithful customers exit. Also consider the sequence of random variables $\{X_n: n = 1,2,3, ...\}$ As a Markov chain. So we have,

$$X_{n+1} = \begin{cases} X_n - 1 + A, & \text{اگر } X_n > 0 \text{ متلاً، } X_n \geq 1 \\ A & \text{اگر } X_n = 0 \end{cases}$$

In this equation, A is the number of consumers during the preparation time T of (n + 1) faithful customer which is logged.

We know that if (X_n) shows the first stage function, we can write:

$$U(X_n) = \begin{cases} 1, & \text{اگر } X_n > 0 \text{ یا } X_n \geq 1 \\ 0, & \text{اگر } X_n = 0 \end{cases}$$

So, X_ (n + 1) can be written as follow:

$$X_{n+1} = X_n - U(X_n) + A \dots\dots\dots (1)$$

Suppose that the system is in a steady state, and hence, the possibility that the number of customers in the system be independent of time, will be fixed.

In this way, the size of the average system at the time of departure is as follow:

$$E(X_{n61}) = E(X_n)$$

Factoring the sides 1, we have:

$$E(X_{n+1}) = E(X_n - U(X_n) + A)$$

$$E(X_{n+1}) = E(X_n) - E\big(U(X_n)\big) + E(A) \dots\dots. (2)$$

Since $E(X_{n61} = E(X_n)$, we have

$$E(X_n) = E(X_n) - E\big(U(X_n)\big) + E(A)$$

$$E\big(U(X_n)\big) = E(A) \dots\dots\dots. (3)$$

By drawing from Equation 1, we have:

$$X_{n+1}^2 = (X_n - U(X_n) + A)^2 = X_n^2 + U^2(X_n) + +A^2 - 2X_n U(X_n) + 2AX_n - 2AU(X_n) \dots\dots\dots (4)$$

But

$$U^2(X_n) = \begin{cases} 1 & \text{اگر } X_N^2 > 0 \\ 0 & \text{اگر } X_n^2 = 0 \end{cases} = \begin{cases} 1 & \text{اگر } X_n > 0 \\ 0 & \text{اگر } X_n = 0 \end{cases}$$

So, X_n, it shows number of customers and hence X_N can't be negative.

$$U^2(X_n) = U(X_n)[U(X_n) = 1 \ \ or\ 0]$$

Also,

$$X_n U(X_n) = X_n$$

Thus the equation 4 as follow:

$$X_{n+1}^2 = X_n^2 + U(X_n) + A^2 - 2x_N + 2AX_n - 2AU(X_n)$$

For example,

$$2X_n - 2AX_n = X_N^2 - X_{n+1}^2 + U(X_n) + A^2 - 2AU(X_n)$$

$$2X_N(1 - A) = X_n^2 - X_{n+1}^2 + U(X_n) + A^2 - 2AU(X_n)$$

By a factoring of both sides, we have:

$$2[E(X_n) - E(A\ X_n)] = E(X_n^2) - E(X_{n+1}^3) + E\big(U(X_n)\big) + e(A^2) - 2E\big(AU(X_n)\big)$$

$$2[E(X_n) - E(A)E(X_n)] = E(X_n^2) - E(X_{n+1}^2) + E\big(U(X_n)\big) + E(A^2) - 2E(AU(X_n))$$

Thus, A and X_N are independent,

$$2E(X_n)[1 - E(A)] = E(A^2) - E(A^2) + E(A^2) + E(A) - 2E(A)E(A)$$

$$2E(X_n)[1 - E(A)] = E(A^2) + E(A) - 2[E(A)]^2$$

$$E(X_n) = E(A) - 2[E(A)] + \frac{E(A^2)}{2\big(1 - E(A)\big)} \dots\dots (5)$$

Since the entry period "T" is a Poisson process with rate λ,

$$E\left(\frac{A}{T}\right) = \lambda T$$

$$E\left(\frac{A^2}{T}\right) = \lambda^2 T^2 \lambda T \quad \dots\dots\dots (6)$$

This can be achieved by the mean and variance of a Poisson process, for example,

$$E[X(t)] = \lambda t$$

$$E[X^2(t)] = \lambda^2 t^2 + \lambda t$$

Also,

$$E(A) = E\left(E\left(\frac{A}{T}\right)\right) = E(\lambda T)$$

$$E(A) = \lambda E(T) \ \ldots\ldots\ldots (7)$$

Similarly,

$$E(A^2) = E(E\left(\frac{A^2}{T}\right) = E(\lambda^2 T^2 + \lambda T)$$

$$E(A^2) = \lambda^2 E(T^2) + \lambda E(T) \ \ldots\ldots\ldots (8)$$

Now, the equation 5 will be as follow:

$$E(X_n) = \lambda^2 E(t^2) + \lambda E(T) + \lambda E(T) - \frac{2[\lambda E(T)]^2}{2(1-\lambda E(T))} = \lambda^2 E(T^2) + 2\lambda E(T) - \frac{2\lambda^2[E(T)]^2}{2(1-\lambda E(T))}$$

$$= 2\lambda E(T)[1-\lambda E(T)] + \frac{\lambda^2 E(T^2)}{2(1-\lambda(ET))}$$

$$E(X_n) = \frac{2\lambda E(T)[1-\lambda E(T)]}{2(1-\lambda E(T))} + \frac{\lambda^2 E(T^2)}{2(1-\lambda E(T))} \ \ldots\ldots (9)$$

Standard quantity of customers in the system can be obtained from the given Eq. Keep in mind that a system multiple servers with multiple similar servers, is configured to respond to requests from the particular application. So, we just focused on the quality of the system and do not consider other standard consumers such as the allocation of resources, delays or source procurement, installation and development of a virtual machine and other overhead in a complex environment of cloud computing.

The waiting time distribution

The customer waiting time can obtain with the help of the equation as the standard quantity of customers calculated in the system.

$$E(X_n) = \frac{2\lambda E(T)[1-\lambda E(T)]}{2(1-\lambda E(T))} + \frac{\lambda^2 E(T^2)}{2(1-\lambda E(T))} = \frac{2\lambda E(T)[1-\lambda E(T)]}{2(1-\lambda E(T))} + \frac{\lambda^2 E(T^2)}{2(1-\lambda E(T))}$$

$$E(Xi_i) = 2\frac{pE(T)[1-\lambda E(T)]}{2(1-\lambda E(T))} + p\frac{\lambda E(T^2)}{2(1-\lambda E(T))} \ldots\ldots (10)$$

$$E(Xi\,i_n) = \left[\frac{2\lambda E(T)[1-\lambda E(T)]}{2(1-\lambda E(T))} + \frac{\lambda^2 E(T^2)}{2(1-\lambda E(T))}\right] P \ldots\ldots (11)$$

$$E(Xii\,i_n) = \left[\frac{2E(T)[1-\lambda E(T)]}{2(1-\lambda E(T))} + \frac{\lambda E(T^2)}{2(1-\lambda E(T))}\right] - \left(\frac{1}{\mu}\right) \ldots\ldots\ldots (12)$$

With the help of the distribution of waiting times, delays and row values can be provided by customers who are waiting in row for resources and placed it in the hands of service providers.

Conclusion and Summary

Evaluating server centers performance can be considered one of the essential aspects of cloud computing, which can be important for cloud serving and these servers client. In this paper, an analytical study to evaluate the efficiency of a data cloud computing center is provided. In the future, these methods can be used to improve provider's profit (with the help of pricing techniques). Sensitive pricing point can be a very important technique to improve provider's profit as well as customer satisfaction.

References:

B. Yang, F. Tan, Y. Dai, and S. Guo, "Performance Evaluation of Cloud Service Considering Fault Recovery," Proc. First Int'l Conf. Cloud Computing (CloudCom '09), pp. 571-576, Dec. 2009.

Brown Mary, Saravanan k., Performance Factors of Cloud Computing Data Centers, International Journal of Grid Computing & Applications (IJGCA) Vol.4, pp., 2013.

C. Knessl, B. Matkowsky, Z. Schuss and C. Tier,"Asymptotic analysis of a state-dependent M/G/1 queueing system," SIAM J. Appl. Math. 46 (1986) 483–505.

D. D. Yao, "Refining the diffusion approximation for the M/G/m queue," Operations Research, vol. 33, pp. 1266–1277.

J. Baker, C. Bond, J. Corbett, J.J. Furman, A. Khorlin, J. Larsonand, J.M. Leon, Y. Li, A. Lloyd, and V. Yushprakh, "Megastore: Providing Scalable, Highly Available Storage for Interactive Services," Proc. Conf. Innovative Data Systems Research (CIDR), Jan , pp. 223-234, 2011.

J. Martin, and A. Nilsson, "On service level agreements for IP networks," In Proceedings of the IEEE INFOCOM, June 2002.

K. Xiong and H. Perros, "Service Performance and Analysis in Cloud Computing," Proc. IEEE World Conf. Services, pp. 693-700, 2009.

L. Vaquero , L. Rodero-Merino , J. Caceres and m. Lindner, "A break in the clouds: towards a cloud definition," ACM SIGCOMM Computer Communication Review, vol. 39, no.1, 2009.

M.B. Comb´e and O.J. Boxma , "BMAP modelling of a correlated queue. In: Network performance modeling and simulation", J. Walrand, K. Bagchi andG.W. Zobrist (eds.) 177–196, 1998.

Prasdh k., Senthilkumar R., Nonhomogeneous Network Traffic Control System Using Queuing Theory, IJCET, Volume 3, Issue 3, October , pp. 394-405, 2012.

W. Kim, "Cloud computing: Today and Tomorrow," Journal of Object Technology, 8, 2009.

Wikipedia, "Cloud Computing," In http://en. wikipedia.org /wiki/Cloud_computing.

Y. Simmhan and L. Ramakrishnan, "Comparison of resource platform selection approaches for scientific workflows," 19th ACM Intl. Symp. on HighPerformance Distributed Computing, HPDC, Chicago, IL, pp. 445–450, 2010.