Memory Size Estimation of Supercomputing Nodes of Computational Grid using Queuing Theory

Rahul Kumar
Institute of Environment & Management
Lucknow, India.

Dr. I. A. Khan
Integral University
Lucknow, India.

Dr. S. P. Tripathi
Institute of Engineering & Technology
Lucknow, India.

Dr. V. D. Gupta
Integral University
Lucknow, India.

ABSTRACT
Grid computing principles focus on large-scale resource sharing in distributed systems in a flexible, secure and coordinated fashion. The most widespread contemplation is performance, because computational grid servers must offer cost-effective and high-availability services in the elongated period, thus they have to be scaled to meet the expected load. Performance measurements can be the base for performance modeling and prediction. With the help of performance models, the performance metrics (like buffer estimation, waiting time) can be determined at the development process. This paper describes the possible queue models those can be applied in the estimation of queue length to estimate the final value of the memory size. Both simulation and experimental studies using synthesized workloads and analysis of real-world Gateway Servers demonstrate the effectiveness of the proposed system.

General Terms
Grid Computing, Computational Grid, Distributed Computing, Cloud Computing.

Keywords
Grid Computing, Queuing Model.

1. INTRODUCTION
The computer utilizes its growing ability for offices, industries, business organizations, corporations; scientific urbanization, institutions, public administrations and others are using computers for better management. Every organization has its own computing system for exchange of information. Internet has been laid down which can collect or send data globally. There are problems where huge data are required to be transferred from one computer to other computer. If we look for the locations of information data, they are physically and geographically distributed. Thus source of data is at different place. Many times data processing is required to be done in a short time. For some problems even Super Computers cannot meet the processing speed for a problem. A team has been constituted in USA in 2001 to find methodology for large data computations in shortest time. Ian Foster[1-4] carried out the study and suggested to have computational grid for USA Computational grid shall be able to provide

(a) Transfer of data in Gigabytes per second.
(b) Processing speed in Terabytes per second.
(c) Storing capacity of grid in Petabytes.

To meet the above mentioned speed and handling of very large data parallel processing of Super Computer is one of a viable solution. Foster suggested that super Computer be called a node. Many super Computers in the form of node can be part of grid. If good number of supercomputers become nodes we require having an operating system program for computing data where super computer will process in parallel. Hence a maximum parallel computation can be possible provided grid organization is established.

2. GRID ORGANIZATION
In Grid Organization the entire super Computers are to be connected using Optical Fiber System. The best grid is when Mesh Topology is used. We know that in network parallel data movement is converted to serial data movement between two computing units. This makes the working of network slow. Internet limits the processing to client computers only. Client computer can receive the data and even package programs from server or from fellow clients. Network hence provides the facility of information exchange but has a poor speed of processing.

When supercomputers form nodes then its processing speed is very high. The modular structure of program enables to process separate modules at separate nodes. After the computation of nodal programs final computation is carried out. This methodology enables computers to handle huge data with shortest time.

The data between two nodes should be transferred using Optical fibers. Optical fibers presently available 144 parallel paths through which Optical signals can be transmitted. This requires a unit in every branch of transmission to convert analytical signal to optical signal. Optical signal are constructed using laser beam. Hence if the computer word length, which is 1 to 28 bits presently, is used then entire word can be transmitted parallel by single pulse command. This makes transmission speed very fast, which is very clear that nodal system uses parallel ports of Super Computers for network while internet uses serial ports.

3. MEMORY SIZE AT GRID COMPUTING NODES
The most important resource, which the basic factor for determining the performance of a Grid, is the storage space management. Since the memory of a supercomputer is limited

and there is huge amount of data, which is to be shared among the different supercomputers in a Virtual Organization, the formation of waiting lines or queues is imminent. The ratio between the arrival rate of data at a particular node in a Grid usually denoted by λbits/sec and the service rate or departure rate of data from the node denoted by μbits/sec, gives the Grid its performance metrics known as Quality of Service (QoS).[5] The two things required for the working of node are

a) It should be ergodic[6] i.e. stable
b) Size of memory for the space complexity

Let the data present at memory is 'n' megabytes. Then the probability that there are 'n' data at time t is P_n (t). Let time is increased by small value Δt. Since the time interval is small, so there can be only one transition at a time which is either arrival or departure.

State transition can be explained as follows. Let the rate of arrival of data be λ bits/sec Therefore, after Δt secs $\lambda\Delta t$ will arrive.

So the probability of arrival of one data = $\lambda\Delta t$

The probability that no data will arrive = $1-\lambda\Delta t$

Similarly, the probability of departure of one data = $\mu\Delta t$

The probability that no data will depart = $1-\mu\Delta t$

Therefore at any given point in time there are three possible states:

a) No data arrives at the node and no data departs from the node
b) One data arrives at the node.
c) One data departs from the node

The probability that there are 'n' data at time t is P_n (t). The time is increased by small value Δt. Therefore,

Probability (no data arrives, no data departs) = $P_n(t)(1-\lambda\Delta t)(1-\mu\Delta t)$

Probability (one data arrives) = $P_{n-1}(t)(\lambda\Delta t)$

Probability (one data departs) = $P_{n+1}(t)(\mu\Delta t)$

Hence the combined probability of all the three cases is:

$$P_n(t+\Delta t) = P_n(t)(1-\lambda\Delta t)(1-\mu\Delta t) + P_{n-1}(t)(\lambda\Delta t) + P_{n+1}(t)(\mu\Delta t)$$

$$P_n(t+\Delta t)= P_n(t)-P_n(t)(\lambda\Delta t)-P_n(t)(\mu\Delta t)+P_n(t)\lambda\mu\Delta t^2)+P_{n-1}(t)(\lambda\Delta t)+ P_{n+1}(t)(\mu\Delta t)$$

Neglecting Δt^2 term we get

$$[P_n(t+\Delta t) - P_n(t)]/\Delta t= -\lambda P_n(t) -\mu P_n(t) +\lambda P_{n-1}(t) + \mu P_{n+1}(t)$$

Now Δt is very small i.e. $\Delta t \blacktriangleright 0$ so, taking limit

$$\lim_{\Delta t\to 0} \frac{Pn(t+\Delta t) - Pn(t)}{\Delta t} =-\lambda P_n(t)-\mu P_n(t)+\lambda P_{n-1}(t)+\mu P_{n+1}(t)$$

For stability d $[P_n(t)]$/dt=0

i.e. $-\lambda Pn(t)-P_n(t) +\lambda P_{n-1}(t) +\mu P_{n+1}(t) =0$ .... (i)

Consider state transition at time t and n=0. Therefore, in this case two conditions arise

a) No data arrives in the memory buffer
b) One data departs from the memory buffer.

Probability (no data arrives) = $P_0(t)(1-\lambda\Delta t)$

Probability (one data departs) = $P_1(t)(\mu\Delta t)$

Therefore the combined probability of success

$P_0(t+\Delta t) = P_0(t)(1-\lambda\Delta t) + P_1(t)(\mu\Delta t)$

or, $P_0(t+\Delta t) = P_0(t) - P_0(t)\lambda\Delta t + P_1(t)(\mu\Delta t)$

or, $P_0(t+\Delta t) - P_0(t) = - P_0(t)\lambda\Delta t + P_1(t)(\mu\Delta t)$

or, $[P_0(t+\Delta t)- P_0(t)]/\Delta t = - \lambda P_0(t)+\mu P_1(t)$

For stability,

$$\lim_{\Delta t\to 0} \frac{P_0(t+\Delta t) - P_0(t)}{\Delta t} =0$$

Therefore,

$-\lambda P_0(t)+ \mu P_1(t)= 0$

or, $P_1(t) = (\lambda/\mu)P_0(t)$ ……………………...……………… (ii)

From equations (i) & (ii)

$P_0(t) = (\lambda/\mu)^0 P_0(t)$

$P_1(t) = (\lambda/\mu)^1 P_0(t)$

. .

$P_n(t) = (\lambda/\mu)^n P_0(t)$

But the sum of all probabilities should be one i.e.

$$\sum_{n=0}^{\infty} P_n(t) = 1$$

Therefore,

$(\lambda/\mu)^0P_0(t)+(\lambda/\mu)^1P_0(t) +(\lambda/\mu)^2P_0(t) + \ldots\ldots\ldots.+ (\lambda/\mu)^nP_0(t) = 1$

$[1+(\lambda/\mu)+(\lambda/\mu)^2+\ldots\ldots\ldots+(\lambda/\mu)^n] P_0(t) = 1$ ……….. (iii)

Now let $\lambda/\mu= x$

The sum of infinite series

$1 + x + x^2 + x^3+\ldots\ldots\ldots+x^n$

is given by

$S = 1/1-x$

Substituting $x=\lambda/\mu$ in above equation we get

$$S=1/ (1-\lambda/\mu)$$

Substituting the value of S in equation (iii) we get

$$[1/ (1-\lambda/\mu)]P_0(t) = 1$$

Now if $\lambda/\mu<1$, the system is stable

Therefore, for ergodic condition

$$P_0(t) = [1-\lambda/\mu]$$

The probability density function (pdf) of data in memory for the ergodic state is given by

$P_n(t)=(\lambda/\mu)^n[1-\lambda/\mu]$…………….……………(iv)

The above equation, which is known as the probability density function of the data in memory can be used for estimation of average value or mean value of data staying in the memory.

Mean data Q (L) = $\sum_{n=0}^{\infty} n.P_n(t)$

$$Q(L)= \sum_{n=0}^{\infty} n.(\lambda/\mu)^n[1-\lambda/\mu] \quad \ldots\ldots\ldots\ldots (v)$$

where n = Data in memory

λ = Data arrival rate

μ = Data departure rate

Let probability of success $p=\lambda/\mu$

Probability of failure $q = 1-p = \lambda/\mu$

Table 1: Calculation of Average Queue

Data(in bits)	Probability Pn(t)	$n.P_n(t)$
0	$p^0.q$	0
1	$p^1.q$	$p.q$
2	$p^2.q$	$2.p^2.q$
3	$p^3.q$	$3. p^3.q$
n	$p^n.q$	$n. p^n.q$

Mean data $Q(L) = \sum_{n=0}^{\infty} n.P_n(t)$

$$Q(L)=0 + p.q +2.p^2.q +3. p^3.q+\ldots+ n. p^n.q$$

$$Q(L)=p.q[1+2.p+3.p^2+\ldots+np^{n-1}] \quad \ldots\ldots\ldots (vi)$$

The sum of infinite series

$1+2.p+3.p^2+4.p^3+\ldots\ldots+n.p^{n-1}$

is given by

Let $S=1+2.p+3.p^2+4.p^3+\ldots+n.p^{n-1}$

$$S = \sum_{n=0}^{\infty} n.p^{n-1} \quad \ldots\ldots\ldots (vii)$$

Multiplying by p on both sides of equation (vii) we get

$$S.p=p+2.p^2+3.p^3+\ldots\ldots+\infty \quad \ldots\ldots\ldots (viii)$$

Subtracting equation (viii) from (vii) we get

$S-Sp= 1+p+p^2+\ldots\ldots$

or , $S(1-p) = 1/(1-p)$

or, $S = 1/(1-p)^2$

Substituting the value of S in equation (vi) we get

$$Q (L) = p. q. (1/ (1-p)^2)$$

$$= p / (1-p)$$

Substituting $p=\lambda/\mu$, we get

$$Q (L) = \frac{\lambda/\mu}{1-\lambda/\mu} \quad \ldots\ldots\ldots (ix)$$

The average value of the queue length can be used to estimate the Standard Deviation of the queue length, thereby the size of the memory. If the Standard Deviation is added to the average value of queue, then the size of memory will be

Size of Memory = $\overline{Q (L)}$ + SD

Standard Deviation is given by

$$\sigma^2 = \mu_2 - \mu_1^2$$

where σ = Standard Deviation

μ_2= variance

μ_1= mean or average

The probability density function (pdf) of data in memory for the ergodic state is given by

$$P_n(t)=(\lambda/\mu)^n[1-\lambda/\mu]$$

Let,

Probability of success $p=\lambda/\mu$

Probability of failure $q =1-p=1-\lambda/\mu$

$\Sigma n^2.P_n (t)=\Sigma n^2p^nq$

$= p (1+p)/ (1-p)^2$

Variance $\sigma^2 = 1/n [(\Sigma n^2P_n (t) - Q (L))^2]$

$=[p.(1+p)/(1-p)^2-p/(1-p)]$

$= [2p^2/ (1-p)^2]^2$

Therefore, $\sigma = 2p^2/ (1-p)^2 = \mu_2$

Hence,

Standard Deviation S.D. = $\sqrt{\mu_2 - \mu_1^2}$

$=\sqrt{2p^2/ (1-p)^2 - [p/ (1-p)]^2}$

$= p/ (1-p)$

Substituting $p=\lambda/\mu$

Standard Deviation S.D. = $\frac{\lambda/\mu}{1-\lambda/\mu}$

If the Standard Deviation is added to average queue length then the size of memory

Size of Memory = $\overline{Q(L)}$ + S.D.

$$= \frac{\lambda/\mu}{1-\lambda/\mu} + \frac{\lambda/\mu}{1-\lambda/\mu}$$

$$= \frac{2[\lambda/\mu]}{1-[\lambda/\mu]}$$

4. G/M/1 SIMULATION MODEL

In the G/M/1 Queue[7-16] simulation model, the inter-arrival times and the service times of the processes have a general distribution. For this model, an exact analysis is in general impossible. Therefore, approximations are given for the performance measures like the expected time in the queue, the expected time in the system, the expected number of clients in the queue, the expected number of clients in the system, etc. In the model, the arrival process for which the inter arrival times (A_1, A_2...) of clients are identically distributed random variables

with an arbitrary distribution function. Service times of clients (B_1, B_2...) are identically distributed random variables with arbitrary distribution function. There is a single server and the capacity of the queue is infinite. Just as in the M/M/1, M/G/1 and G/M/1 queue, the stability condition for the G/G/1 queue is that the amount of work offered per time unit to the server should be less than the amount of work the server can handle per time unit.
The arrival rate for the sake of computation conveniences are taken as λ=5000, 7500, 10000, 12500, 15000 and for μ=5001, 7501, 10001, 12501, 15001. The computed results are revealed in the subsequent table:

Table 2: Estimation of Queue Length

Rate of Arrival (λ)	Rate of Departure (μ)	Q (L) for G/M/1
5000	5001	5098
7500	7501	7647
10000	10001	10194
12500	12501	12740
15000	15001	15283

We have calculated Queue Length for 5 rates of arrivals. These 5 points are on the curve of Queue Lengths. Queue Length at very high rate will be also on the curve of Queue Length.

Let the equation of the curve is represented as:

$$y = a_0 + a_1\lambda + a_2\lambda^2 \ldots\ldots\ldots\ldots\ldots\ldots (x)$$

Now, using Non-linear regression techniques, we have to calculate the value of a_0,a_1, and a_2, for the minimal error.

These leads to develop formation equations as given below:

$$na_0 + a_1\sum\lambda_i + a_2\sum\lambda_i^{\;2} = \sum y_i \ldots\ldots\ldots\ldots\ldots (xi)$$

$$a_0\sum\lambda_i + a_1\sum\lambda_i^{\;2} + a_2\sum\lambda_i^{\;3} = \sum\lambda_i y_i \ldots\ldots\ldots (xii)$$

$$a_0\sum\lambda_i^{\;2} + a_1\sum\lambda_i^{\;3} + a^2\sum\lambda_i^{\;4} = \sum\lambda_i^{\;2} y_i \ldots\ldots\ldots (xiii)$$

There are three unknown variables i.e., a_0, a_1, and a_2 and three equations for given values of λ. These values are computed and given as:

Table 3: Regression Table for higher values of Queue Length

x	y	x^2	x^3	x^4	xy	x^2y
5000	5098	$2.50*10^7$	$1.25*10^{11}$	$6.25*10^{14}$	$2.55*10^7$	$1.27*10^{11}$
7500	7647	$5.63*10^7$	$4.22*10^{11}$	$3.16*10^{15}$	$5.74*10^7$	$4.30*10^{11}$
10000	10194	$1.00*10^8$	$1.00*10^{12}$	$1.00*10^{16}$	$1.02*10^8$	$1.02*10^{12}$
12500	12740	$1.56*10^8$	$1.95*10^{12}$	$2.44*10^{16}$	$1.59*10^8$	$1.99*10^{12}$
15000	15283	$2.25*10^8$	$3.38*10^{12}$	$5.06*10^{16}$	$2.29*10^8$	$3.44*10^{12}$

Let, x=λ & y=Q (L)

Therefore,

Σx=50000.00, Σy=50962.00, $\Sigma x^2=5.63*10^8$, $\Sigma x^3=6.88*10^{12}$, $\Sigma x^4=8.88*10^{16}$, $\Sigma xy=5.73*10^8$, $\Sigma x^2y=7.01*10^{12}$

Using the Abbreviated Doolittle Process, we get

Table 4: Abbreviated Doolittle Process for Regression Analysis

	1	2	3	y
	5	50000	$5.63*10^8$	$5.10*10^4$
		563000000	$6.88*10^{12}$	$5.73*10^8$
			$8.88*10^{16}$	$7.00*10^{12}$
A1	5	50000	$5.63*10^8$	$5.10*10^4$
B1	1	10000	$1.13*10^8$	$1.02*10^4$
A2		63000000	$-6.34*10^{16}$	$5.36*10^7$
B2		1.00	$-1.01*10^9$	$8.50*10^1$
A3			$-6.38*10^{25}$	$7.00*10^{12}$
B3			$1.00*10^0$	$-1.10*10^{13}$

With the help of above table we can find out the values of a_0, a1, a2 i.e.

a2=B3y=0.0000051

a1=B2y-B23 a2=0.850284

a0=B1y-B12a1-B13a2=1689.5

Based on the obtained value,

Q(L)= 1689.5 + 0.850284λ + 0.0000051λ2

where, λ is the rate of arrival and Q(L) is the Queue Length.

The results for the Queue Length and hence estimated memory size is as given in the following table

Table 5: Computation of Queue Length w.r.t. G/M/1 model

Rate of Arrival	Q (L) for G/M/1	Memory size [G/M/1]
10^{10}	$6.00008*10^{14}$	$2[6.00008*10^{14}]$
10^{11}	$6.00008*10^{16}$	$2[6.00008*10^{16}]$
10^{12}	$6.00008*10^{18}$	$2[6.00008*10^{18}]$

5. CONCLUSION

We proposed that for supercomputing nodes, the buffer estimation is contemporary need in crafting the blueprint of future computational grid. The study delineates that for stable functioning of supercomputer, each node of the computational grid should work in ergodic environment. The other issue that damages the implementation of computational grid is freezing of data in some node because of inadequate size of memory. In circumstances of multi-channel arrivals and single channel departure, akin to Queue Model, G/M/1 has been studied in facet by simulating queues at lower rate of arrivals. The consequence so attained is extrapolated to compute the size of memory at very towering rate. The outcome demonstrates that we require few hundred terra bytes for arrival rate of 1012bytes. Extensive simulation study illustrates pioneering method that can provide smooth performance control and better track in computational grid systems.

6. REFERENCES

[1] Foster I. and Kesselman C., "The GRID: Blueprint for a new Computing Infrastructure," Morgan Kauffman Publishers, 151 Edition 1999, 2nd Edition 2003.

[2] Foster I., Kesselman C., Tuecke S., "The Anatomy of the Grid: Enabling Scalable Virtual Organizations," Int. Journal of Supercomputer Applications, 15(3), 200l.

[3] Gentzsch W., Enterprise Resource Management: Applications in Research and Industry, In: Ian Foster and Carl Kesselman, The Grid: Blueprint for a new computing infrastructure, 2nd Edition, Morgan Kaufmann Publisher 2003.

[4] Karl Czajkowski, Ian Foster & Carl Kesselman , "Resource Co-Allocation in Computational Grids" ,Proceedings of the 8th IEEE International Symposium on High Performance Distributed Computing,1999.

[5] Foster I., Kesselman C., Nick J.M., Tuecke S., **"**Grid Services for Distributed System Integration**",** Computer, June 2002.

[6] Piera, F.J. Mazumdar, R.R.: "An ergodic result for queue length processes of state-dependent queueing networks in the heavy-traffic diffusion limit", Communication,Control And Computing, pp 504-507,2008.

[7] Barford,P., Crovella,M.: "Generating representative web workloads for network and server performance evaluation",in Measurement and Modeling of Computer Systems,pp.151-160,1998.

[8] Menasce,D., Almeida,V.: "Capacity Planning for Web Services Metrics,Models,and Methods",Prentice Hall PTR,2001.

[9] Lazowska, E.D.: "Quantitative system performance: computer system analysis using queuing network models",Prentice-Hall, Inc,1984.

[10] Anderson & Darrell: "A Case for Buffer Servers", p.82, IEEE Seventh Workshop on Hot Topics in Operating Systems, 1999.

[11] Roberts & Jim W.: "Traffic Theory and the Internet", IEEE Transactions on Communications, 2001.

[12] Zari & Mazen: "Understanding and Reducing Web Delays", IEEE Journal for Electronics and Computer Science, pp.30-37, Vol.34, No.12, 2001.

[13] Steven H. & Srikant R.: "A Mathematical Framework for Designing a Low-Loss, Low-Delay Internet", IEEE Transactions, 2002.

[14] Chen X. & Mohopatra P.: "Performance Evaluation of Service Differentiating Internet Servers", pp. 1368-1375, Vol. 51, No.11, 2002.

[15] Ying Lei: "Global Stability of Internet Congestion Controllers with Heterogeneous Delays", IEEE Transactions on Communications, 2003.

[16] Kleinrock L.: "Queueing Systems", Vol. 2, Applications. John Wiley Publications, NY, 1976.