This is the published version:


Available from Deakin Research Online:

http://hdl.handle.net/10536/DRO/DU:30008054

Reproduced with the kind permissions of the copyright owner.

Copyright : 2007, Australian Computer Society
Communication Network Analysis of the Enterprise Grid Systems

Bahman Javadi\(^1\), Jemal H. Abawajy\(^2\), Mohammad K. Akbari\(^1\), Saeid Nahavandi\(^2\)

\(^1\)Computer Engineering and Information Technology Department, Amirkabir University of Technology
424 Hafez Ave., Tehran, Iran
{javadi,akbari}@ce.aut.ac.ir

\(^2\)School of Engineering and Information Technology, Deakin University
Geelong, VIC. 3217, Australia
jemal@deakin.edu.au

Abstract

This paper addresses the problem of performance analysis based on communication modelling of large-scale heterogeneous distributed systems with emphasises on enterprise grid computing systems. The study of communication layers is important because the overall performance of a distributed system is often critically hinged on the effectiveness of this part. This model considers processor as well as network heterogeneity of target system. The model is validated through comprehensive simulation, which demonstrates that the proposed model exhibits a good degree of accuracy for various system sizes and under different working conditions. The proposed model is then used to investigate the performance analysis of typical systems.

Keywords: Enterprise Grid, Performance Analysis, Analytical Modelling, Heterogeneity, Communication Model.

1 Introduction

Performance analysis and modelling of parallel and distributed systems has always been, to say the least, a contentious and problematic exercise. Recently, grid computing has been proposed as the next generation of infrastructure to support distributed applications in scientific, engineering and commercial domain (Foster 2002). The Grid is a highly heterogeneous environment that can potentially provide seamless, fast and efficient access to range of resources that are distributed over a wide area (Dongarra and Lastovetsky 2006). Since the communication in such systems has the great impact on the overall performance, a communication model can lead to understanding of key issues in system design and program development perspective.

In this paper we address the problem of analytical communication modelling for the enterprise grid which typically is heterogeneous cluster of clusters computing systems. Theses systems belong to a general class of such systems named as “Multi-Cluster” systems (Xu 2001, Abawajy and Dandamudi 2003). Examples of production-level multi-cluster systems include the DAS-2 (DAS-2 2002) and the LLNL multi-cluster system (Boas 2003).

The proposed model is based on probabilistic analysis and queuing network to analytically evaluate the performance of communication networks for cluster of clusters systems. The model takes into account processor as well as network heterogeneity among clusters. Since simplifications are often made to reduce the complexity of models, there is a need to validate the models through simulation. Validation is typically carried out for test cases, which require reasonable computation time and resources.

Several analytical performance models of multi-computer systems have been proposed in the literature for different interconnection networks and routing algorithms (e.g., Sarbazi-Azad et al. 2002; Boura and Das 1997; Drapper and Ghosh 1994). Unfortunately, little attention has been given to the grid computing systems. Most of the existing researches are based on homogenous cluster systems and the evaluations are confined to a single cluster (Du et al. 2000; Hu and Kleinrock 1995) with the exception of (Clematis and Corana 1999), which looked at processor heterogeneity. Moreover, in (Yang et al. 2005) a queuing model based on input and server distributions was proposed to analyse a special grid system, VEGA 1.1. The most of these works are based on job level modelling, but in contrast we intend to propose an analytical model in communication layer of enterprise grid systems to provide more accurate performance prediction and analysis. To our best knowledge, this work would be the first which deals with heterogeneous enterprise grid environments.

The rest of the paper is organized as follows. In Section 2, we give a brief description of the enterprise grid systems which is used in this paper. In Section 3, we give detailed description of the proposed analytical communication model. We present the model validation experiments and heterogeneity analysis as results of our
work in Section 4. Finally, we summarize our findings and conclude the paper in Section 5.

2 System Description

The computational grid architecture which is used in this paper is a typical cluster of clusters. These systems are constructed by interconnecting multiple single cluster systems thus heterogeneity may be observed in communication networks as well as processors. The system which is shown in Fig. 1, is made up of $C$ clusters, each cluster is composed of $N_i$ computing nodes (i.e., cluster size), $i \in \{0,1,...,C-1\}$, each comprising a processor with $\tau_i$ processing power (i.e., heterogeneous processors) and its associated memory module. Each cluster has two communication networks: an Intra-Communication Network (ICN1) and an interCommunication Network (ECN1). The ICN1 is used for the purpose of message passing between processors within a cluster while the ECN1 is used to transmit messages between clusters, management of the entire system. To interconnect of clusters, the ECN1 is connected through a set of Concentrators/Dispatchers (Dally and Towles 2004) to the external network, i.e., ICN2.

![Enterprise Grid Architecture](image)

High performance computing clusters typically utilize Constant Bisectional Bandwidth (i.e., Fat-Tree) topologies to construct large node count non-blocking switch configurations (DAS-2 2002; Force 2004). In this paper we adopted $m$-port $n$-tree (Lin 2003) as a fixed arity switches to construct the topology for each cluster system. An $m$-port $n$-tree topology consists of $2^m (m/2)^{n-1}$ communication switches. In addition, each communication switch itself has $m$ communication ports $\{0,1,2,...,m-1\}$ that are attached to other switches or processing nodes. Every switch except root switches uses ports in the range of $\{0,1,2,...,(m/2)-1\}$ to have connection with its descendants or processing node, and using ports in the range of $\{(m/2),(m/2)+1,...,m-1\}$ for connection with its ancestors.

Flow control and routing algorithms are important components of a communication network. The flow control manages the allocation of resource to messages as they progress along their route. In this paper, we used the wormhole flow control, which is commonly used in cluster network technologies, e.g., Myrinet, Infiniband and QsNet. Routing algorithms establish the path between the source and the destination of a message. Since the most of cluster network technologies adopted deterministic routing, we used a deterministic routing based on Up*/Down* routing (Schroeder et. el. 1990) which is proposed in (Javadi et al. 2006a). In this algorithm, each message experiences two phases, an ascending phase to get to a Nearest Common Ancestor (NCA), followed by a descending phase.

In our model, we express the processing power of various processors in each cluster relatively to a fixed reference processor (Clematis and Corana 1999), and not relatively to the fastest processor which is used in the most works on heterogeneous parallel systems. So the relative processing power of each node can be found as $s^{(i)} = \tau_i / \tau_f$, where $f$ is the number of reference machine. Since we consider the processor heterogeneity among clusters, the total relative processing power and the average relative processing power of the $C$ clusters in the system is as follows, respectively:

$$S = \sum_{i=0}^{C-1} s^{(i)}$$

$$\bar{s} = \frac{S}{C}$$

3 The Communication Model

In this section, we develop analytical communication model for the above mentioned enterprise grid system in which processing power of nodes and also inter- and intra-communication networks are heterogeneous.

3.1 Assumptions

The proposed model is built on the basis of the following assumptions which are widely used in similar studies (Sarbazi-Azad et al. 2002; Boura and Das 1997; Hu and Kleinrock 1995):

1. Nodes generate traffic independently of each other, and which follows a Poisson process with a mean rate of $\lambda_s^{(i)}$, where $i \in \{0,1,...,C-1\}$, messages per time unit.
2. The destination of each message would be any node in the system with uniform distribution.
3. The number of processors in all clusters are equal ($N_0 = N_1 = ... = N_{C-1}$) and the clusters’ nodes are heterogeneous in their processing power.
4. The network heterogeneity is presence between inter-cluster and intra-cluster communication networks.
5. The communication switches are input buffered and each channel is associated with a single flit buffer.
6. Message length is fixed ($M$ flits).
7. The source queue at the injection channel in the source node has infinite capacity. Moreover, messages are transferred to the node once they arrive at their destinations.

We have two types of connections in this topology, node to switch (or switch to node) and switch to switch. In the first and the last stage, we have node to switch and switch to node connection respectively. In the middle stages, the switch to switch connection is employed. Each type of connection has a service time: 

t_{rcm} = 0.5α_{m} + L_{w}β_{m} \text{ and } t_{cs} = α_{s} + L_{w}β_{s},

where \( t_{rcm} \) and \( t_{cs} \) represent times to transmit from node to switch (or switch to node) and switch to switch connection, respectively. \( α_{m} \) and \( α_{s} \) are the network and switch latency, \( β_{m} \) is the transmission time of one byte (inverse of bandwidth) and \( L_{w} \) is the length of each flit in bytes.

In the presence of network heterogeneity, we have two values for times to transmit. For intra-cluster networks the pair of \( (t_{rcm(i)}, t_{cs(i)}) \) and for inter-cluster networks the pair of \( (t_{rcm(E)}, t_{cs(E)}) \) are adopted in the model.

3.2 Outline of the Model

The analysis was done in the top-down manner. At first, we calculated the arrival rate of messages in each communication network. Then we start the analysis at the last stage and continues backwards to the first stage. The computed average latency in each network contains three factors. First, the average message service time that takes place across the channel. Next, the service time at the last stage to message delivery. Finally, the waiting time at the source queue is added to get the overall latency.

3.2.1 Average Message Latency of Internal Network

In this section, we find the average latency of each communication network from cluster \( i \) point of view. Since each message may cross different number of links to reach its destination, we consider the network latency of an \( 2j \)-link message as \( T_{j}^{(i)} \), and averaging over all the possible nodes destined made by a message yields the average message latency as:

\[
\bar{T}^{(i)} = \sum_{j=1}^{m} (P_{j} \times T_{j}^{(i)})
\]

Where \( P_{j} \) is the probability of a message crossing \( 2j \)-link (\( j \)-link in the ascending and \( j \)-link in the descending phase) to reach its destination in a \( m \)-port \( n \)-tree topology. Different choices of \( P_{j} \) lead to different distribution for message destination, and consequently different average message distance. As it is mentioned in assumption 2, we take into account the uniform traffic pattern so, based on the \( m \)-port \( n \)-tree topology, we can define this probability as follows:

\[
P_{j} = \begin{cases} 
\frac{\left( \frac{m}{2} - 1 \right) \left( \frac{m}{2} \right)^{j-1}}{N_{0} - 1} & j = 1, 2, \ldots, n - 1 \\
\frac{\left( m - 1 \right) \left( \frac{m}{2} \right)^{n-1}}{N_{0} - 1} & j = n 
\end{cases}
\]

The message flow model of the system is shown in Fig. 2, where the path of a flit through various communication networks is illustrated. A processor, which is shown as a circle in this figure, sends its request to the ICN1 and ECN1 with probabilities \( 1 - P_{o} \) and \( P_{o} \) respectively. The message path is depicted as arrows. The request rate of a processor is \( \lambda_{x}^{(i)} \), so the input rate of ICN1 and ECN1 which are fed from the same processor will be \( \lambda_{x}^{(i)} (1 - P_{o}) \) and \( \lambda_{x}^{(i)} P_{o} \), respectively.

![Fig. 2. Message flow model in the system](image)

The probability \( P_{o} \) has been used as the probability of outgoing request within a cluster and is obtained by the following equation:

\[
P_{o} = \frac{\sum_{i=1}^{C} N_{i}}{N - 1} = \frac{(C - 1) \times N_{0}}{C \times N_{0} - 1}
\]

The external request of cluster \( i \) go through the ECN1 with probability \( P_{o} \) and then ICN2. In the return path, it again accesses the ECN1 in cluster \( v \) to get to the destination node. The concentrator/dispatcher are working as simple buffers to interface two external networks (i.e., ECN1 and ICN2) and so combine message traffic from/to one cluster to/from other cluster. Therefore, the message rate received by ICN1 and ECN1 in cluster \( i \) (to cluster \( v \)) can be calculated as follows:

\[
\lambda_{x}^{(i)} = (1 - P_{o}) \lambda_{x}^{(i)}
\]

\[
\lambda_{x}^{(i)} = P_{o} \lambda_{x}^{(i)} + P_{a} \lambda_{x}^{(i)} \quad v \neq i
\]

In the second stage, the message rate of ICN2 can be computed by following equation:

35
\[ \lambda_{i2}^{(i)} = N_o P_o \lambda_x^{(i)} \]  \hfill (8)

Given that a newly generated message in cluster \( i \) traverse \( 2j \)-link to reach its destination with probability \( P_j \), the average number of links that a message traverse to reach its destination is given by:

\[ d_{avg} = \sum_{j=1}^{K} (2j \times P_j) \]  \hfill (9)

By substituting Eq.(4) in to Eq.(9), the average message distance is obtained as follows:

\[ d_{avg} = \left( \frac{m}{2} - 1 \right) \left( \frac{m}{2} - \frac{1}{2} \right) \prod_{k=1}^{n-1} \frac{m^k}{2^k} + 1 \quad \forall n > 1 \]  \hfill (10)

For \( n=1 \) the average message distance is, \( d_{avg} = 2 \). Consequently, the rate of received messages in each channel can be driven as follows:

\[ \eta_{i1}^{(i)} = \frac{(1 - P_o^{(i)}) \lambda_x^{(i)} \times d_{avg(i1)}}{4n} \]  \hfill (11)

\[ \eta_{i2}^{(i,v)} = \frac{P_o \left( \lambda_x^{(i)} + \lambda_x^{(v)} \right) \times d_{avg(E1)}}{4n} \quad \forall \neq i \]  \hfill (12)

\[ \eta_{i2}^{(i,v)} = \frac{N_o P_o \lambda_x^{(i)} \times d_{avg(12)}}{4n} \]  \hfill (13)

Where \( n_c \), the number of trees in the ICN2, can be computed as follows:

\[ n_c = \left\lceil \log_2 \frac{c - 1}{c} \right\rceil \]  \hfill (14)

Our analysis begins at the last stage and continues backward to the first stage. The network stage numbering is based on location of switches between the source and the destination nodes. In other words, the numbering starts from the stage next to the source node (stage 0) and goes up as we get closer to the destination node. The number of stage in \( m \)-port \( n \)-tree topology is \( K = 2j - 1 \). The destination, stage \( K - 1 \), is always able to receive a message, so the service time given to a message at the final stage is \( t_{en} \). The service time at internal stages might be more because a channel would be idled when the channel of subsequent stage is busy. The average amount of time that a message waits to acquire a channel at stage \( k \) for cluster \( i \), \( W_{i,k}^{(i)} \), is given (Javadi et al. 2006b):

\[ W_{i,k}^{(i)} = \frac{1}{2} \eta_k \left( T_{i,k}^{(i)} \right)^2 \]  \hfill (15)

The average service time of a message at stage \( k \) is equal to the message transfer time and waiting time at subsequent stages to acquire a channel, so:

\[ T_{i,k}^{(i)} = \begin{cases} \sum_{j=k+1}^{K} W_{i,j}^{(i)} + M_{en} & 0 \leq k \leq K - 2 \\ M_{en} & k = K - 1 \end{cases} \]  \hfill (16)

According to this equation, the mean network latency for a message with \( 2j \)-link journey is equal to \( T_{i,k}^{(i)} = T_{j}^{(i)} \).

A message originating from a given source node in cluster \( i \) sees a network latency of \( T^{(i)} \) (given by Eq.(3)). Due to blocking situation that takes place in the network, the distribution function of message latency becomes general. Therefore, a channel at source node is modelled as an M/G/1 queue. The average waiting time for an M/G/1 queue is given by (Kleinrock 1975):

\[ W_s^{(i)} = \frac{\lambda^{(i)} \left( \sigma^{(i)} + \bar{X}^{(i)} \right)}{2 \left( 1 - \rho^{(i)} \right)} \]  \hfill (17)

\[ \rho^{(i)} = \frac{\lambda^{(i)} \bar{X}^{(i)}}{\lambda^{(i)}} \]  \hfill (18)

Where \( \lambda^{(i)} \) is the average message rate on the network, \( \bar{X}^{(i)} \) is the average service time, and \( \sigma^{(i)} \) is the variance of the service time distribution. Since the minimum service time of a message at the first stage is equal to \( M_{en} \), the variance of the service time distribution is approximated based on a method proposed in (Draper and Ghosh 1994) as follows:

\[ \sigma^{(i)} = \left( T^{(i)} - M_{en} \right)^2 \]  \hfill (19)

As a result, the average waiting time in source queue becomes

\[ W_s^{(i)} = \frac{\lambda^{(i)} \left( \left( T^{(i)} - M_{en} \right)^2 + \overline{T^{(i)}} \right)^2}{2 \left( 1 - \rho^{(i)} \right) \overline{T^{(i)}}} \]  \hfill (20)

Finally, the average message latency, \( \overline{L}^{(i)} \), seen by the message crossing from source node from cluster \( i \) to its destination, consists of three parts; the average waiting time at the source queue \( \left( \overline{W_s^{(i)}} \right) \), the average network latency \( \overline{T^{(i)}} \), and the average time for the tail flit to reach the destination \( \overline{R^{(i)}} \). Therefore,

\[ \overline{L}^{(i)} = \overline{W_s^{(i)}} + \overline{T^{(i)}} + \overline{R^{(i)}} \]  \hfill (21)

Where,
d_{av} = d_{av}(E_1) + d_{av}(I_2)

The average message latency in the ICN1 from cluster \( i \) point of view, \( \bar{T}_{E12}^{(i)} \), would be found by Eq.(21) by substitution of \( \eta_{12}^{(i)} = \eta_{11}^{(i)} \), \( \lambda^{(i)} = \lambda_{11}^{(i)} \), \( t_{rs} = t_{rs(E)} \), and \( d_{avg} = d_{avg(E)} \).

### 3.2.2 Average Message Latency of External Networks

As mentioned before, external messages cross through both networks, ECN1 and ICN2, to get to their destination in other cluster. Since the flow control mechanism is wormhole, the latency of these networks should be calculated as a merge one. Of this and based on the Eq.(3) we can write,

\[
\bar{T}_{E12}\left( i,v \right) = \sum_{j=1}^{n} \sum_{k=1}^{g} \left( P_{jkh} \times T_{jkh}^{(i,v)} \right)
\]

\[P_{jkh} = P_j \times P_h\]  

(24)

Where \( P_j \) and \( P_h \) can be calculated from Eq.(4). It means each external message cross 2j-link through the ECN1 (j-link in the source cluster \( i \) and j-link in the destination cluster \( v \)) and 2h-link in the ICN2 to reach to its destination. So the analysis would be done for \( K = 2(j+h)-1 \) stages. Moreover, the average channel rate in Eq.(15) must be substituted with the following equation:

\[
\eta_k^{(i,v)} = \begin{cases} 
\eta_{12}^{(i)} & j \leq k < j+2h-1 \\
\eta_{11}^{(i,v)} & \text{otherwise}
\end{cases}
\]

(25)

The average message latency of inter-cluster networks from cluster \( i \) point of view can be found as the arithmetic average of all latencies which the message from cluster \( i \) to all other clusters, namely cluster \( v \), might be seen as follows:

\[
\bar{T}_{E12}\left( i,v \right) = \sum_{j=1}^{n} \sum_{k=1}^{g} \left( P_{jkh} \times T_{jkh}^{(i,v)} \right)
\]

(26)

Where \( \bar{T}_{E12}\left( i,v \right) \) would be determined with Eq.(21) by the following substitutions:

\[
\lambda^{(i,v)} = \lambda_{E1}^{(i,v)}
\]

\[t_{rs} = t_{rs(E)} \quad \text{and} \quad t_{rv} = t_{rs(E)}
\]

\[d_{avg} = d_{avg(E)} + d_{avg(I2)}\]

(29)

The average waiting time at the concentrator/dispatcher is calculated in a similar manner to that for the source queue (Eq.(17)). By modelling the injection channel in the concentrator/dispatcher as an M/G/1 queue, the average arrival rate and average waiting time are given by following equations:

\[
\bar{W}_d^{(i)} = \frac{\lambda_{12}^{(i)} \left( M_{t(E1)} \right)^2}{2(1 - \lambda_{12}^{(i)} M_{t(E1)})}
\]

(30)

Also, we model the ejection channel in the concentrator/dispatcher as an M/G/1 queue, with the same rate of injection channel. So, the average waiting time at the dispatcher would be that same which is given by Eq.(30).

Based on Fig. 2, we could find the average message latency of cluster \( i \) with the following equation:

\[
\bar{t}^{(i)} = \left(1 - P_o\right) \left( 1 - P_{E1} \right) + P_o \left( \bar{L}_{E12}^{(i)} + 2\bar{W}_d^{(i)} \right)
\]

(31)

To calculate the total average of message latency, we use a weighted arithmetic average as follows:

\[
\bar{t} = \sum_{v=0}^{C-1} \left( s_v \times \bar{t}^{(i)} \right)
\]

(32)

At last, to perform our analysis we chose to express the degree of processor heterogeneity of the system through a single parameter, i.e., the standard deviation of relative processing power as follows:

\[
H = \left( \frac{1}{C} \sum_{v=0}^{C-1} \left( s_v - \bar{s} \right)^2 \right)^{1/2}
\]

(33)

### 4 Validation of the Model

In order to validate the proposed model and justify the applied approximations, the model was simulated with a discrete event-driven simulator. Requests are generated randomly by each processor with an exponential distribution of inter-arrival time with a rate of \( \lambda^{(i)} \). The destination node is determined by using a uniform random number generator. Each packet is time-stamped after its generation. The request completion time is checked in every “sink” module in each processor to compute the message latency. For each simulation experiment, statistics were gathered for a total number of 100,000 messages. Statistic gathering was inhibited for the first 10,000 messages to avoid distortions due to the warm-up phase. Also, there is a drain phase at the end of simulation in which 10,000 generated messages were not in the statistic gathering to provide enough time for all packets to reach their destination.

Extensive validation experiments have been performed for several combinations of clusters sizes, network sizes, message length, and degree of heterogeneity. The general conclusions have been found to be consistent across all the cases considered. After all, to illustrate the result of some specific cases to show the validity of our model, the items which were examined carefully are as follows:

- System size: \( N=2^9 \) and \( N=2^{10} \)
- Cluster size: \( C=2^4 \) and \( C=2^5 \)
- Switch size: \( m=4 \) and \( m=8 \) ports
- Message length: \( M=32 \) and \( M=64 \) flits
- Flit length: \( L_m=256 \) and 512 bytes
- Total relative processing power: \( S=C \)

Note that we changed the degree of processor heterogeneity while the total relative processing power is fixed and equal to the number of clusters, i.e., \( S=C \). Moreover, two different network assignments are used in the validation experiments which are listed in Table 1. In the first configuration, all communication networks (i.e., ICN1, ECN1, and ICN2) are same with Net.1 and in the second the ICN1 is Net.1 and ECN1 and ICN2 are Net.2.

Table 1. Network configurations for the model validation

<table>
<thead>
<tr>
<th>Network Parameter</th>
<th>Net.1</th>
<th>Net.2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Network technology bandwidth</td>
<td>500</td>
<td>300</td>
</tr>
<tr>
<td>Network latency (time unit)</td>
<td>0.02</td>
<td>0.05</td>
</tr>
<tr>
<td>Switch latency (time unit)</td>
<td>0.01</td>
<td>0.02</td>
</tr>
</tbody>
</table>

The results of simulation and analysis for a system with above mentioned parameters are depicted in Fig. 3 to Fig. 6 in which the average message latencies are plotted against the offered traffic with degree of processor heterogeneity equals to 0.2. The figures reveal that the analytical model predicts the mean message latency with a good degree of accuracy when the system is in the steady state region, that is, when it has not reached the saturation point. However, there are discrepancies in the results provided by the model and the simulation when the system is under heavy traffic and approaches the saturation point. This is due to the approximations that have been made in the analysis to ease the model development. One of the most significant terms in the model under heavily loaded system, is the average waiting time at the source queue. The approximation which is made to compute the variance of the service time received by a message at a given channel (Eq. (19)) is a factor of the model inaccuracy. Also, in this region the traffic on the links is not completely independent, as we assume in our analytical model. Since, the most evaluation studies focus on network performance in the steady state regions, so we can conclude that the proposed model can be a practical evaluation tool that can help system designer to explore the design space and examine various design parameters.
To have a performance analysis with the proposed model, Fig. 7 depicts the maximum network throughput as a function of processor heterogeneity in the system which is used in the validation section with $C=32$, message lengths $M=64, 128, L_m=256$, and two different values for the network bandwidths: first, the ICN1 is Net.1 and ECN1 and ICN2 are Net.2 and second, 20 percent increasing in bandwidth of all networks. It is observed that the homogenous system ($H=0.0$) posse’s highest maximum throughput, so it can better handle heavy traffic compared to heterogeneous ones. Our result is in accordance with the work of (Clematis and Corana 1999), where it is affirmed that an increase in heterogeneity worsens performance. So, the optimal configuration is the homogenous system because it yields the most uniform distribution of the total communications among the available links. Also, the figure shows that the maximum throughput of communication network will be degraded as the processor heterogeneity increases but the performance degradation is more considerable for the processor heterogeneity less than 50 percent. As it can be seen in this figure, the system performance is fewer for longer messages, because of increasing the service time received by a message at a channel, so lead to longer blocking time, and thus to a higher latency. To consider of effect of network bandwidth on the system performance, the second system configuration (i.e., more network bandwidth) has been analyzed. This upgradation yields the similar behaviors, however about 19 percent improvements in all points of the system performance. Our results confirmed by the work of (Lee et al. 2000), which based on the measurement in a grid test bed found the communication bandwidth and latencies have much more effective on the performance of the system rather than the processor speed.

5 Conclusions

Analytical models play a crucial role in evaluation of a system under various design issues. In this paper, an analytical model of communication layer for enterprise grid computing systems is discussed. Both processor and network heterogeneity have been involved to outline the model. The proposed communication model has been validated with versatile configurations and design parameters. Simulation experiments have proved that the model predicts message latency with a high degree of accuracy. The results of performance analysis have revealed that bandwidth of communication networks are most effective factors on such systems and the processor heterogeneity has marginal impact to degrade the overall system performance. For future work, we intent to take the non-uniform traffic pattern into account, which is closer to the real traffic in such systems.

6 References


http://www.cs.vu.nl/das2


