

Washington University in St. Louis

Washington University Open Scholarship

All Computer Science and Engineering
Research

Computer Science and Engineering

Report Number: WUCS-91-39

1991-07-01

An Access Protection Solution for Heavy Load Unfairness in DQDB

Lakshmana N. Kumar and Andreas D. Bovopoulos

This paper discusses the unfairness issue arising in a 802.6 DQDB network at high loads-- when the traffic demand to a bus exceeds the capacity of that bus. As per the 802.6 protocol, at heavy loads, the end nodes along a bus experience longer delays than the other nodes. The origin and remedy of this heavy load unfairness is discussed. An access control scheme is proposed as a solution. The comparison of the proposed scheme with 802.6 protocol is presented. The simulation results and performance characteristics are discussed under several types of loads. With symmetric load conditions under the... [Read complete abstract on page 2.](#)

Follow this and additional works at: https://openscholarship.wustl.edu/cse_research



Part of the [Computer Engineering Commons](#), and the [Computer Sciences Commons](#)

Recommended Citation

Kumar, Lakshmana N. and Bovopoulos, Andreas D., "An Access Protection Solution for Heavy Load Unfairness in DQDB" Report Number: WUCS-91-39 (1991). *All Computer Science and Engineering Research*.

https://openscholarship.wustl.edu/cse_research/657

Department of Computer Science & Engineering - Washington University in St. Louis
Campus Box 1045 - St. Louis, MO - 63130 - ph: (314) 935-6160.

An Access Protection Solution for Heavy Load Unfairness in DQDB

Lakshmana N. Kumar and Andreas D. Bovopoulos

Complete Abstract:

This paper discusses the unfairness issue arising in a 802.6 DQDB network at high loads-- when the traffic demand to a bus exceeds the capacity of that bus. As per the 802.6 protocol, at heavy loads, the end nodes along a bus experience longer delays than the other nodes. The origin and remedy of this heavy load unfairness is discussed. An access control scheme is proposed as a solution. The comparison of the proposed scheme with 802.6 protocol is presented. The simulation results and performance characteristics are discussed under several types of loads. With symmetric load conditions under the proposed scheme, all active nodes along a bus experience almost the same access delay and packet loss characteristics. Performance under several other load condition are also found to be satisfactory.

**An Access Protection Solution for
Heavy Load Unfairness in DQDB**

Lakshmana N. Kumar and Andreas D. Bovopoulos

WUCS-91-39

July 1991

**Department of Computer Science
Washington University
Campus Box 1045
One Brookings Drive
Saint Louis, MO 63130-4899**

Proceedings of INFOCOM '92.

AN ACCESS PROTECTION SOLUTION FOR HEAVY LOAD UNFAIRNESS IN DQDB

Lakshmana N. Kumar and Andreas D. Bovopoulos

Department of Computer Science

and

Computer and Communications Research Center,
Washington University, St. Louis, MO 63130-4899

July 5, 1991

ABSTRACT

This paper discusses the unfairness issue arising in a 802.6 DQDB network at high loads — when the traffic demand to a bus exceeds the capacity of that bus.

As per the 802.6 protocol, at heavy loads, the end nodes along a bus experience longer delays than the other nodes. The origin and remedy of this heavy load unfairness is discussed. An access control scheme is proposed as a solution. The comparison of the proposed scheme with 802.6 protocol is presented. The simulation results and performance characteristics are discussed under several types of loads.

With symmetric load conditions under the proposed scheme, all active nodes along a bus experience almost the same access delay and packet loss characteristics. Performance under several other load conditions are also found to be satisfactory.

1 High Load Unfairness and its Origin

Access unfairness can be defined as the inability of a subset of the nodes to gain access to a bus as quickly as the other nodes under the given access protocol mechanism. As a result some nodes experience longer delays than the other nodes before gaining access.

Under 802.6 protocol a node can file a *REQUEST* only when a packet reaches the head of its internal packet queue. Till such time it can not solicit a slot for

the other packets enqueued in its local queue. This scheme* is called *Exhaustive Scheme* of filing requests. As a consequence, under 802.6 protocol access unfairness sets in at high loads. The pattern of unfairness and the bandwidth share achieved by individual nodes can be attributed to :

- The initial state of the network *i.e.*, the number of requests enqueued by the downstream nodes when the heavy load condition sets in, as was shown in [4].
- The number of empty slots lying in between the first and last nodes along a bus at the time heavy load sets in, as was shown in [4].
- The type of load presented to the network— Under heavy loads, the nodes at the beginning or at the end of a bus (along the direction of transmission) may face unfair treatment depending on the workload type in the network. The next section discusses this in detail.

The access unfairness at high loads is also known as *stubborn unfairness* —since the unfairness effectively remains as long as the heavy load condition persists.

2 Patterns of Unfairness at Different Loads

Two different unfairness patterns are observed among the nodes depending on the workload type. This section discusses these unfairness patterns in detail. The simulation results with different network configurations show that the unfairness patterns are independent of the network configuration and depend primarily on work load type.

2.1 Symmetric Load

Symmetric load implies that an incoming packet might have any of the other remaining nodes as its destination with equal probability. In other words the traffic offered by a node is proportional to the number of nodes in the downstream of the bus. Thus the number of packets enqueued for access at individual nodes will steadily decrease along the direction of a bus.

With the growing complexity of computer networks, individual nodes in a MAN can be gateways connecting to other (LANs or MANs) networks. Hence it is more likely that the traffic generated by an individual node is bound to be proportional to the downstream node population. Thus symmetric load type is a realistic workload type. Also known as *normal load* it is widely used in the literature [2] [3] [8].

*[1] discusses another scheme of filing requests, *viz.*, *Non-Exhaustive Scheme* under which are *REQUEST* can be filed for a packet arrives for transmission almost immediately in the next available slot. Access unfairness do exist under this scheme and the paper discusses an access control scheme for such an implementation.

Under symmetric work load the nodes that are closer to the frame generating head-end suffer heavy delays. The nodes towards the end of the bus experience smaller delays. This can be explained as follows. At heavy loads the nodes at the beginning of a bus continue to receive a heavy influx of *REQUEST*s on the opposite bus. This forces them to honor a large number of pending *REQUEST*s every time they attempt for access. This happens in spite of the fact that the nodes at the beginning of a bus are more likely to generate high traffic. Thus under heavy symmetric traffic, the earlier nodes along the direction of a bus are heavily blocked for access.

2.2 Asymmetric Load

With asymmetric load type the traffic presented by a node to a bus is directly proportional to the number of nodes in the upstream of the bus. Thus the number of packets enqueued by the individual nodes steadily increases along the direction of the bus. Under this load type unfairness arises because of the following reasons :

- When the downstream nodes generate heavy traffic, the *Exhaustive Scheme* of filing *REQUEST*s of 802.6 protocol allows only a single *REQUEST* to be filed for the distributed queue. Since the end nodes generate more packets under this load type, this forces the downstream nodes to maintain longer queues. This effect combined with the latency of the propagation of a *REQUEST* force the end nodes experience longer delays.
- Though the upstream nodes generate fewer packets, they continue to have easy access at the expense of farther-away nodes. Nothing prevents them from grabbing an empty slot, unless they have received a *REQUEST*—due to their closer proximity to the frame generating head-end.

The net result is that the downstream nodes are heavily blocked. This kind of traffic may occur rarely and is discussed in [5] and [6].

3 Solution to Access Unfairness

The access unfairness implies that a subset of nodes could not grab sufficient empty slots. In order to avoid this, more empty slots need to be allowed by the other nodes. This directly implies that some sort of limits are to be applied while using the *REQUEST* counter values at individual nodes. An *access protection limit* for a node, has to be a function of the node population in the network and its position along the bus so that the bandwidth assured to individual nodes can be fixed. This can ensure fair bandwidth share for every node, with any network configuration.

However in a real scenario new nodes may join the network and few may go down or leave the network. Hence *the node population and the position of a node*

along the direction of the bus need not be static. The Dynamic Assessment of Network Topology scheme (proposed in [9]) provides the capability to dynamically update the active (The word *active* denotes the nodes that are part of the network and are potential candidates to contest for access) node population in the network and also its physical position along the bus. For the rest of the paper the node population and the position of each node along the direction of the bus is considered to be known and provided by the Dynamic Assessment of Network Topology scheme (proposed in [9]).

Filipiak in [1] suggests two such limits — one as the *upper protection limit* and the other as the *lower protection limit*. These limits are applied when the contents of the *RQ* counter were moved to that of *CD* counter when the node changes its state from *IDLE* to *COUNTDOWN* or when the count down is completed. By controlling the transfer from the *RQ* counter to the *CD* counter, bandwidth available to a node can be controlled.

The 802.6 DQDB protocol applies the following rule :

$$CD \leftarrow RQ \quad .$$

Let the upper protection limit of a node-*i* be denoted by \hat{P}_i^A and the lower protection limit be \check{P}_i^A , with respect to accessing Bus-A. Assume that a packet arrives for access and the node's $RQ > 0$.

Under upper protection scheme the above protocol description is changed to

$$CD = \min \{ RQ, \hat{P}_i^A \} \quad .$$

Here a restriction is imposed to the bandwidth claim of the node. As a consequence the stubborn unfairness can be eliminated if the limits are well chosen. In general any node that employs a upper protection limit of m gets at least $\frac{1}{m+1}$ of the unused bandwidth (*i.e.*, slots not used by the nodes in the upstream). This is true for all channel bit rates and the network configurations. The burden lies in the selection of the right limits for a given node.

If the lower protection limit is applied then

$$CD = \max \{ RQ, \check{P}_i^A \} \quad .$$

Under this scheme the restriction is such that every node contending for access always allows *at least a predefined* bandwidth, to be utilized by the other nodes. The use of lower protection limit prohibits the use (by a node) of additional bandwidth greater than the assigned limit and the node allows a fixed number of slots (equal to that of the limit assigned to it) before sending its own packet. Thus a node with m as its lower protection limit to access a particular bus can receive utmost $\frac{1}{m}$ of the overall bandwidth unused by upstream.

Thus the lower protection limit schemes imply possible loss of bandwidth, since the actual downstream traffic might be less than its share of bandwidth.

Hence, the upper protection limit scheme is a better choice which if the protection limits are properly chosen, could effectively control the unfairness issue at high loads. Hence attention is focussed on upper protection limit, in this paper.

The mode of filing a request is of paramount importance in selecting the limits. In [1] it is suggested that choosing $\hat{P}_i^A = (N - i)$ as the upper protection limit could ensure $\frac{1}{N}$ of the bandwidth for node- i in network of population N — if nodes are numbered from left to right along Bus-A and *non-exhaustive* scheme of writing a request is implemented. However this policy can work only with this non-exhaustive scheme and is not suitable for 802.6 definition of DQDB.

The *exhaustive* scheme of *REQUEST* filing used in 802.6 protocol puts a limit on the total number of *REQUEST* seen by a node. For example, a node- i in a network of N nodes may not see more than $(N - i)$ requests—one sent by each of the downstream nodes. Hence even if 802.6 protocol is modified with the introduction of upper protection limits as $\hat{P}_i^A = (N - i)$, the upper protection never really comes into play. Hence 802.6 performance is unaffected. Through simulations the 802.6 protocol performance is verified to remain unaltered with such access protection.

4 A Proposed Scheme for Access Protection

In this section an access protection scheme (also referred to as *Scheme-I*) is proposed to eliminate the high load unfairness.

4.1 Source-Destination Pair Concept

The traffic offered to the network is generally unbiased and whenever a node has packet it is more likely to be destined for any of the remaining nodes in the network with equal probability. This kind of network activity is very natural. The resulting traffic is referred to as *symmetric traffic*.

For this reason the underlying concept behind the *Access Protection Scheme* (referred to as, APS) presented in this report makes use of the upper protection limits, for which, to arrive at an estimate of the limit the source-destination pair criteria is employed. The upper protection schemes try to restrict the bandwidth claimed by a node. We argue that this restriction be based on the number of potential parties (downstream nodes) for which it can send packets along a bus and the total number of such possible pairs that can enter into a dialogue in the network.

This logic ensures that the share of bandwidth is based on potential destinations and not just equal share for every node. The dynamic update capability of the node population and the position of each node along a bus, extended by the dynamic assessment scheme [9], guarantees the expected performance even under changing network conditions. Thus the protection limits of the APS (to

be derived later) are valid even in a dynamic network environment of varying node population.

4.2 Evaluation of Individual Protection Limits

Let us consider a network which has a population of N nodes at a given instant and node- i (which is placed as the i^{th} node from the frame generating head-end along the direction of a bus). Node- i will then have $(N-i)$ possible destinations for queuing up its packets for accessing the bus. Thus the total possible Source-destination pairs along a bus is :

$$\sum_{i=1}^N (N-i) = \frac{N(N-1)}{2} .$$

Since both N and i are dynamically updated the limits can also be rightly updated and applied as said above. For some node- i , let us denote the bandwidth that is to be assured by the APS as BW_i (as a fraction of the overall channel capacity[†] at heavy loads). Under the scheme aimed at, BW_i for a node depends on its own downstream nodes and also the total possible source-destination pairs. Accordingly,

$$BW_i = \frac{\text{No. of potential destinations for node-}i}{\text{Total no. of source-destination pairs}} = \frac{(N-i)}{\frac{N(N-1)}{2}} = \frac{2(N-i)}{N(N-1)} . \quad (1)$$

Let \hat{P}_i be the upper protection limit applied by node- i which in turn implies that node- i is assured of at least $\frac{1}{\hat{P}_i+1}$ of the bandwidth that is left unused by its upstream nodes. The bandwidth that is used by all upstream nodes is the sum of the bandwidth assured to them individually by the protocol. At high loads this can be written analytically as :

$$BW_i = \frac{1}{\hat{P}_i+1} \left[1 - \sum_{k=1}^{i-1} BW_k \right] , \quad \text{for } 1 \leq i \leq N . \quad (2)$$

Using Eqn. 1 in Eqn. 2,

$$\frac{2(N-i)}{N(N-1)} = \frac{1}{\hat{P}_i+1} \left[1 - \sum_{k=1}^{i-1} \frac{2(N-k)}{N(N-1)} \right] ,$$

and

$$\hat{P}_i+1 = \frac{N(N-1)}{2(N-i)} \left[1 - \frac{2}{N(N-1)} \sum_{k=1}^{i-1} (N-k) \right] .$$

[†]Hence $\sum_{i=1}^N BW_i = 1$ is true.

Table 1: Illustration of protection limits in a small network

Protection Limits in a NW of 10 Nodes				
Node-id	For Bus-A		For Bus-B	
	\hat{P}_i^A	BW_i^A	\hat{P}_i^B	BW_i^B
i	$\frac{N-(i+1)}{2}$		$\frac{i-2}{2}$	
1	4.0	18/90	-0.5	0
2	3.5	16/90	0.0	2/90
3	3.0	14/90	0.5	4/90
4	2.5	12/90	1.0	6/90
5	2.0	10/90	1.5	8/90
6	1.5	8/90	2.0	10/90
7	1.0	6/90	2.5	12/90
8	0.5	4/90	3.0	14/90
9	0.0	2/90	3.5	16/90
10	-0.5	0	4.0	18/90

Therefore,

$$\hat{P}_i = \frac{N - (i + 1)}{2} \quad \text{for } 1 \leq i \leq N \quad (3)$$

Table-1 shows the bandwidth for every node in 10-node network along with the individual protection limits to be applied for accessing bus-A and bus-B. Node-id refers to the numbering scheme used to distinguish from one another which proceeds from left to right along Bus-A. Accordingly a Node-id of 1 refers to the first node along Bus-A—which is also the last node along Bus-B.

4.3 More Observations about Scheme-I

The same node need to use two different limits with respect to accessing each of the two buses. Node- i (placed as the i^{th} node from the frame generating head-end on bus-A in network of N nodes) will use the following protection limits to access either of the buses :

$$\hat{P}_i^A = \frac{N - (i + 1)}{2} \quad ,$$

and

$$\hat{P}_i^B = \frac{(i - 2)}{2} \quad .$$

Notice that both the limits are expressed in terms of i alone, in the above equations. The Dynamic Assessment of Network Topology scheme proposed in [9] enables each node to dynamically update its position along a given bus. The limits can be computed from using a single formula (given in Eqn-3) and the node's position along a particular bus is updated with every assessment cycle.

i.e., a node positioned 7th along Bus-A in a 10 Node network will see as—its position as 7 on Bus-A and its position on Bus-B as 4. Using Eqn-3 the different limits for accessing each of the bus can be arrived at—using the corresponding node position on a particular bus.

The negative value of protection limit for the last node of the bus (seen in Table-1) does not matter, since it has got nobody else in the downstream to correspond with. Hence the bandwidth claim of that a node is *zero*.

It may seem that the end nodes along a bus might be discriminated. However this is only partly true. As seen from Table-1 each node in that example is assured of an overall bandwidth of 18/90 along both the buses even under heavy load conditions. Actually two way conversations between two nodes (one of them being an end node along some bus) may not really suffer—without the access protection scheme that node might be completely blocked with very long delays, on the bus in which it is closer to the frame generating head-end. Most likely the impact of access protection will be positive over the conversation. With the scheme included, the previously blocked node will encounter lesser delays and this should work to the speedup of the conversation.

The protection limits shown in Table-1 are actually multiples of 0.5. In our implementation we used a flag—with the aid of which we alternately used the nearest integer on either side of the non-integer limit.

Another point of importance is that this scheme uses upper protection limits and hence chooses the minimum of RQ counter value and the protection limit. This implies that in effect the upper protection limit is operative only when the RQ counter value exceeds the protection limit value (possible at high load situations). Thus the proposed APS should retain the performance characteristics of the standard 802.6 protocol at lower loads. In other words it acts as an *extension* to 802.6 performance, meant to handle high load unfairness. Therefore, the APS is operational within the domain of high load activity only . At lower loads APS is inactive and a network using the Scheme-I modified protocol would operate as a traditional 802.6 protocol and the unfairness does exist at these low load situations as well. As far as APS is concerned the low load unfairness is a separate issue. In [10] we address this issue. The low load performance enhancement schemes do not interfere with APS and such schemes along with APS address the unfairness issue in DQDB for all possible load situations.

5 Simulation Details

The channel capacity is 44.7 Mbps and the 802.6 slot format of 53 bytes is used. However the slot overhead is overlooked. The simulations are run for a network with 100 nodes each separated by 0.5 Kms from one another. The messages are assumed to fit into a single slot. The simulations were run for 3 seconds of network activity. The performance characteristics obtained, are enclosed.

Some more results at different network configurations are also obtained with

a simulation run of 2 seconds network activity.

The configurations (all tested with Poisson arrival pattern and single packet messages) under Scheme-I are :

1. 200 nodes each node at every 0.25 Kms
2. 100 nodes each node at every 0.5 Kms
3. 25 nodes each node at every 2.0 Kms
4. 50 nodes each node at every 1.0 Kms
5. 10 nodes each node at every 5.0 Kms

Also simulations are run with messages of multiple packet sizes. Messages (Poisson arrival) of mean packet sizes of 1, 4, 8 and 20 are considered for the 100 node network with the nodes 0.5 kms apart. Messages (Poisson arrival) of mean packet sizes of 1, 4, and 8 are considered for the 25 node network with the nodes 2 kms apart.

Further the simulations are run with no restriction on the packet queue at the nodes. In other words infinite buffer length is assumed. In addition we assume that all nodes generate packets of a single priority level.

5.1 Message Generation

A list of the various arrival patterns that are used for this study is provided below.

1. Poisson Arrival
2. Graded Workload
3. Bursty Traffic

The discussion on individual patterns follows. In evaluating expressions to compute the inter-arrival time at a given particular load value specified, the following notations are used. This list is valid throughout this work.

- N = Node population in the network
- \bar{x} = Average number of packets per Message
- i = Node-id of a node, numbered left to right on Bus-A
- C = Single Channel capacity in bits per second
- S = Slot size in bits
- λ = Message arrival rate at a node
- λ_i = Message arrival rate at a particular Node- i
- λ_i^A = Message arrival rate at a node- i bound for Bus-A
- λ_i^B = Message arrival rate at a node- i bound for Bus-B
- ρ = Load value of network assigned for a simulation run

A message can be either a single packet or multiple packet. The packet corresponds to data of length 48 bytes and this equals the payload specified in the 802.6 standard protocol's slot format. A packet thus fits into a single slot sent by a head-end's frame generator. The destination policy or load type determines the bus on which a message is to be sent.

In the case of a single packet arrival \bar{x} equals *one*.

Multiple packet messages are generated according to a Geometric distribution.

$$P_r [m = i] = pq^{i-1} \quad , \quad \text{where } i \geq 1 \text{ and } q = (1 - p) \quad (4)$$

$P_r [m = i]$ indicates the probability of a message size being equal to i packets. Then the average packet size per message \bar{x} is given by

$$\bar{x} = \frac{1}{p}$$

In case of multiple packet message arrivals the average message size \bar{x} is specified when the simulation parameters are set up. The geometric distribution thus gives the number of slots that are required to transmit that message.

Multiple packet size messages are broken into several packets before queuing them up and all the packets of a single message are destined to the same node.

5.1.1 Poisson Arrival Pattern

Poisson statistics is applied to compute the next arrival time of a message. Under this traffic pattern every node has the same arrival rate λ for traffic (messages) generated. However with respect to a bus the number for packets queued up for access need not be the same. The policy adopted for destination selections *i.e.*, workload type determines it. At the beginning of the simulation the load (ρ) in the network is specified. From the value of ρ the inter arrival time between successive messages can be computed.

The load experienced by the network can be defined as:

$$\rho = \frac{\text{Input data rate}}{\text{Output data rate}} \quad .$$

Output data rate supported by the network is total bit rate offered by both the buses and the input data rate is the rate of traffic generated by every node put together. Thus,

$$\rho = \frac{NS\lambda\bar{x}}{2C} \quad , \quad \text{bps}$$

and therefore,

$$\lambda = \frac{2C\rho}{NS\bar{x}} \quad . \quad (5)$$

As mentioned before for the Poisson arrival pattern every node generates traffic with the same arrival rate for any given load and hence

$$\lambda = \lambda_1 = \lambda_2 = \lambda_3 = \dots = \lambda_{N-1} = \lambda_N \quad .$$

5.1.2 Graded Workload Arrival Pattern

This kind of traffic arrival assumes that the load offered a bus is graded (or distributed according) to a linearly decreasing function from node-1 to node- $(N-1)$. In other words the load offered by a node to a particular bus is directly proportional to the number of its downstream nodes. Thus the mean rate rate of arrival is maintained separately for every node with respect to each bus. In the actual implementation port-A of node- i will generate messages for accessing Bus-A at a rate of λ_i^A (and port-B of node- i will generate at a rate of λ_i^B for accessing Bus-B. Under these assumptions the arrival rates at individual ports of a node can be analytically derived from the network parameters defined at the beginning of a simulation run. The graded work load very closely resembles Poisson symmetric load with a variation at the implementation level.

With respect to accessing Bus-A, the computation of the arrival rate at port-A of any node is done as shown below. Similarly port-B arrival rates can also be computed.

Packets bound for Bus-A

The assumption implies that for any given load ρ , the traffic offered by a node is directly proportional to the number of the downstream nodes. For node-1 which is closest to frame generating head-end of Bus-A, the load offered by it is given by

$$\rho_i^A = (N - i)\rho_{(N-1)}^A \quad (6)$$

where $\rho_{(N-1)}^A$ is the load presented by the $(N-1)^{th}$ node, *i.e.*, the last but one node to its only downstream neighbor on the bus-A. Also ρ_N^A is zero since node- N being the last node does not generate traffic to the Bus-A. In the following analysis we restrict ourselves to a single bus. For any node- i ρ_i^A denotes load offered by that node to the bus-A.

$$\rho = \sum_{i=1}^N \rho_i^A = \rho_{(N-1)}^A \frac{N(N-1)}{2} \quad .$$

Therefore,

$$\rho_{(N-1)}^A = \frac{2\rho}{N(N-1)} \quad . \quad (7)$$

For any general node- i :

$$\rho_i^A = (N - i)\rho_{(N-1)}^A = \frac{2\rho(N - i)}{N(N - 1)} \quad . \quad (8)$$

Therefore,

$$\rho_i^A = \frac{2\rho(N - i)}{N(N - 1)} \quad . \quad (9)$$

The message arrival rate at any node can now be obtained with respect to Bus-A from the definition of load as follows :

$$\begin{aligned}\rho_i^A &= \frac{\text{Input data rate by node-}i}{\text{Output data rate on Bus-A}} \\ \rho_i^A &= \frac{S\bar{x}\lambda_i^A}{C}\end{aligned}$$

Therefore,

$$\lambda_i^A = \frac{C\rho_i^A}{S\bar{x}} = \lambda_i^A = \frac{2\rho C(N-i)}{SN\bar{x}}(N-i) \quad . \quad (10)$$

Retaining the same node-id (or numbering scheme for nodes is same for Bus-B also) λ_i^B can be expressed as :

$$\lambda_i^B = \frac{2\rho C(N-i)}{SN\bar{x}}(i-1) \quad . \quad (11)$$

There is no need to use a destination selection policy since the messages are bound to a particular bus and hence one of the its downstream nodes is picked as the destination. The rest of the implementation is same as that of the Poisson case and the arriving message can be just a single packet or multiple packet size.

5.1.3 Bursty Traffic Arrival Pattern

The underlying arrival pattern is Poisson (*i.e.*, the inter arrival time between successive burst messages follow Poisson distribution) except that the burst messages arrive as bursts of size 2, 4, 6, 8 or 10 packet sizes. Under our implementation, the number of distinct bursts and their corresponding values are specified when the simulation is started. In order to find the arrival rate at every node the average of all the individual bursts is used. To determine the actual size of the burst at run time, any one of the distinct burst sizes is randomly pulled with equal probability of selection. The rest of the analysis and operation is very much along the lines of Poisson arrival pattern. No separate analysis is provided here.

5.2 Performance Characteristics

The performance characteristics (considered to evaluate the merit of an access scheme) are the average access delay and the success rate of a node. The above two performance indices are plotted against the node index along the bus.

Average access delay The access delay of a packet can be defined as the period between the time a packet enters the network to the instant at which it is put on the access bus slot.

In the computation of average delay, the delays encountered by the packets that could be on the slots alone are considered. At the end of the

simulation, we computed average access delays for the packets that have been sent.

The delay of a packet is measured in terms of the number of slots that pass through the bus when the particular packet is waiting for access. Especially at high loads (like $\rho = 1.2$ or 1.8) the packet queue length tends to be high and hence the average delay also seen to be very high.

As a matter of fact, at high loads, the longer we simulate the network activity the average delay observed also goes up with the simulation time. This is because the load is excessive of the channel capacity, and we assume that the nodes have infinite buffers.

Success rate The success rate is defined as the ratio between the number of successful transmissions of packets (or slots claimed) by a node to the total number of packets that originally had arrived at a node bound for some node in its downstream along the access bus.

In short it is an indirect measure of the packet loss suffered at a given node. The larger the success rate, the lesser is the packet loss.

6 Conclusions from Simulation Results

In the following subsections we present the conclusions based on the simulation results. Each subsection is devoted to a different load type. The performance characteristics of both 802.6 protocol and the proposed Scheme-I are enclosed .

The success of the proposed access protection scheme to handle high load unfairness depends heavily on the accurate knowledge of the active node population in the network and the position of a node along the direction of the bus. However in reality it is quite possible that some active nodes may go down and new nodes may join the network. Thus every node should be able to know such changes in the network and re-evaluate its access protection limits. [9] proposes a scheme named *Dynamic Assessment of Network Topology Scheme* which can extend reliable support for access protection schemes.

6.1 Symmetric Load Traffic

Under this traffic the volume of traffic (total number packets queued up for access to a bus) decreases as we travel along the access bus (*i.e.*, the bus for which access is sought). The following observations can be made from the Scheme-I simulation results enclosed.

1. The simulations show that the low load traffic where load (ρ) offered to the bus is 0.75 and 0.9, there is no change in the performance characteristics. *The basic performance of the 802.6 protocol remains upto these load values.*

This is in accordance with the design objective that Scheme-I should come into play only at high load activity period.

2. When the load is just about the channel capacity ($\rho = 1$), the original 802.6 performance characteristics show high delay at the initial nodes close to the frame generating head-end. With Scheme-I the pattern remains but *significant* improvement in the delay at the initial nodes is observed.
3. The above observations are true for all the three arrival patterns, namely, Poisson, Graded workload[†] and Bursty Traffic (bursts of size 2, 4, 6, 8 or 10).
4. At the lower load values, because the load is less than the channel capacity, the success rate is about 100 percent.
5. Evaluation of the performance with messages of single packet size.
6. At heavy *symmetric* loads (with $\rho = 1.2$ or 1.8) the performance of Scheme-I is found to be *independent* on the arrival pattern. The results observed are *extremely convincing* and both the average access delay and the success rate are found to be *fairly uniform* for each and every node along the *entire length* of the bus.

The unfairness noted in 802.6 performance has vanished completely and the window of initial nodes (blocked for access with very high delays) along the access bus that are closer to the frame generating head-end is no longer seen. This has been the case with all the three arrival patterns.

Based on the results obtained from Scheme-I the unfairness of high symmetric load as seen in 802.6 protocol is common to every configuration and messages of multiple packet size. Scheme-I provides an effective solution to the high load symmetric traffic with *uniform delay and success rate* characteristics for every configuration and message size. More importantly the proposed scheme is independent of network configuration (or layout) and message size.

6.2 Asymmetric Load Traffic

Under this scheme the destination selection of the incoming messages is such that the probability of a node attempting access to a particular bus is proportional to the number of its upstream nodes along the same bus.

As a result the number of packets that are queued up for access along a bus (in any duration) keep increasing as we go down along the bus. The last but one

[†]Graded implies that a destination is selected in proportion to the node population in the downstream. It is similar to symmetric load except that in the implementation we generated packet arrivals at each bus-port of the node separately. The underlying arrival pattern is Poisson.

node generates maximum traffic for the bus. This pattern of traffic is a biased[§] scheme. The investigations are carried out with a network configuration having 100 nodes positioned at every 0.5 Kms.

1. The simulations show that the low load traffic where load offered to the bus (ρ) is 0.75 and 0.9, there is no change in the performance characteristics. *The basic performance of the 802.6 protocol remains upto these load values.*
2. The above observations are true for all arrival three patterns, namely, Poisson, Graded workload and Bursty Traffic.
3. At the lower load values seen so far, because the load is less than the channel capacity, the success rate is about 100 percent.
4. We evaluated the performance with single packet messages. The bursty traffic pattern alone resembles multipacket message arrivals. The same observations are still found to be valid in either case.
5. When the traffic offered is at about the channel capacity ($\rho = 1$), the 802.6 performance shows that a handful of end nodes show relatively high delays with other nodes having fairly uniform delays. Scheme-I reduces the number of such end nodes with higher delays but at the expense of the other few remaining end nodes of the bus, hiking their delays sharply.
6. At heavy *asymmetric* loads (with $\rho = 1.2$ or 1.8) the performance of Scheme-I is found to be *independent* on the arrival pattern. The characteristics are very much along the same lines mentioned in the previous item (as was with the load, $\rho = 1.0$).
7. When the load is above the channel capacity the 802.6 protocol results in higher delays towards the end of the bus. Other nodes have relatively lesser and uniform delays. Scheme-I results show that the number of such nodes experiencing heavy delays is reduced at the expense of the left out few that suffer increases in their delays. This implies the width of the window (consisting of several nodes towards the end of the bus) that was blocked with high delays previously has reduced. Also those few nodes in the window have reduced success rate.

Thus Scheme-I retains high delays towards the end of the bus but *significant* decrease in the number of nodes that suffer such high delay is observed—in addition to reduced success rate at these nodes. The results observed are *convincing*, though unfairness is not totally eliminated.

This implies that only fewer nodes towards the end of the bus experience higher delays with Scheme-I. However the other nodes (which were also blocked for access under 802.6 protocol) end up with a sharp rise in their average delay.

[§]As a matter of fact the asymmetric traffic is obtained in simulation by forcing a packet of symmetric load to change its bus of access.

From the view point of successful packets that could be transmitted, the end nodes along the bus could achieve lesser success (This is a direct outcome of the fact that under asymmetric load these end-nodes generate heavy traffic and hence has a large number packets queued up for access. However the access protection permits limited access only). This is the price extracted for ensuring *better uniform* characteristics for the more nodes than what was before. This has been the case with all three arrival patterns.

6.3 Equal Probability Load Traffic

Under this load type an incoming message attempts to access either of the two buses with equal probability independent of the number of downstream or upstream nodes.

As a result under Poisson and bursty arrivals each node *immaterial of its position along a bus* queues about the same number of packets for accessing any particular bus. Such an implementation is also biased since it heeds no attention to the number of potential users with whom a node may have a transaction.

The impact of Scheme-I on the equal probability load type under heavy load is presented below.

1. At relatively low loads like 0.2 to 0.6, we observe the performance characteristics of 802.6 protocol remains the same for all arrival patterns.
2. From a load value of 0.75 the average delay of the end nodes (*i.e.*, farther away from the frame generating head-end along the bus) encounter increased delays with the earlier nodes showing slight improvement.
3. The above effect becomes more and more pronounced with further load increase.

At a load of 0.9, Scheme-I produces larger delays for few end nodes and reduces the delays of earlier nodes along the bus. At a load value of 1.0, a handful of end nodes show a sharp rise in delays with Scheme-I, while the earlier nodes have reduced delays.

4. At heavy loads (like 1.2 or 1.8 with bursty arrivals) a window of the end nodes (that are blocked access with relatively higher delays) towards the end of the bus is observed—as a result of Scheme-I. With Poisson arrivals the end nodes that experience increased delays are widely spread out and the observed increase (in delay) keep growing as we travel down the bus. In both cases the few end nodes that experience increased delays, also end up with reduced success rate. All other nodes experience considerable reductions in their average access delays.

In conclusion Scheme-I is not effective with heavy equal probability loads. The 802.6 standard protocol performance turns out to be *fair* enough. The very first

node experiences a larger delay (because all of its traffic should go on a single bus leading to longer access delays) and all the other nodes experience relatively lesser and uniform delays and the success rate is fairly uniform but for the first node. This is found to be true with different arrival patterns as well.

It may appear that 802.6 protocol be accepted for the significantly fair performance with a load of equal probability type of load. Unfortunately this is the only kind of traffic fairly served by 802.6 protocol and this load type is often unrealistic. This load type implies that in a network of 100 nodes, say node-99 for example, generates or is expected to generate the same amount of traffic bound for node-100 alone as the total volume traffic it generates for the other 98 nodes. Unless the user-99 is biased to access his immediate neighbors only, we can not justify this bias.

If every participating nodes makes full (or best) use of the potential extended by MANs, then the traffic offered by a node is bound to be far more symmetric than being equal probability type. It is for this reason the 802.6 performance improvement need be explored for more realistic traffic load like symmetric load. Hence it is worth compromising some of the better features extended here by 802.6 protocol—exclusively for this particular load type.

A good example of a node that can produce this type of traffic may be the *gateways* that interconnect different networks. By exercising care such that *gateways* find their place some where in the middle of a MAN network, we can ensure the fair performance often with *reduced* delays from what 802.6 protocol can offer.

That is good enough reason to seek enhancements in performance with regard to different load types—more importantly to achieve a very convincing improvement with *symmetric traffic*.

7 802.6 Protocol and Scheme-I

Access protection with the source-destination pair concept leads to desirable performance changes as discussed in the previous section. The improvement achieved by Scheme-I can be summarized as follows:

- For every arrival pattern and symmetric load, Scheme-I yields *very good results* with uniform access delay and success rate characteristics all along the bus.
- For every arrival pattern and asymmetric load, Scheme-I yields also good enough results, with the last few nodes along the direction of a bus experiencing an increased delay and reduces success rate. However the number of such nodes that experience large delays are also reduced by Scheme-I.
- For every arrival pattern and equal probability load, the performance is better with the existing 802.6 protocol. Scheme-I (when ρ is 1.2) produces

a window of several end nodes that are relatively blocked to access and for other nodes the delay is significantly reduced. At still higher loads (ρ is 1.8), however, the delay increases with every node as—along the direction of the bus. There are valid reasons why this should be tolerated as explained in the previous section.

The switch between the 802.6 and Scheme-I performance characteristics is made possible by introducing access protection. Under 802.6 protocol the transfer of RQ counter value to CD counter is done as follows :

$$CD \leftarrow RQ \quad .$$

Under Scheme-I it is modified as :

$$CD = \min \left\{ RQ, \quad \hat{P}_i^A \right\} \quad ,$$

and

$$\hat{P}_i^A = \frac{N - (i + 1)}{2} \quad ,$$

where \hat{P}_i^A is the upper protection for node- i in a network of N nodes (The numbering of individual nodes is done along the direction of the bus).

7.1 Definition of 802.6 as an Access Protection Scheme

With the *exhaustive scheme* of filing a *REQUEST*, there is a limit to the value that an RQ counter can take, since requests are filed only when transmission is complete with the previous request. Thus the RQ counter of node- i can take a maximum value of $(N - i)$ only. Higher values greater than $(N - i)$ may not be realized under *exhaustive scheme* of filing requests. This implies that 802.6 protocol can safely be thought of as an access protection scheme and the upper protection limits are defined as :

$$\hat{P}_i^A = (N - i) \quad .$$

Hence 802.6 protocol definition is also an access protection scheme.

7.2 Relation between 802.6 and Scheme-I protocols

Between the 802.6 and Scheme-I protocols the improvement achieved in the performance is made possible by changing the definition of the access protection limits from $(N - i)$ to $(N - (i + 1))/2$.

Obviously, in between these two definitions there are several other possible definitions for the protection limits. Choosing one such policy, it is possible to end up with a definition that works well with equal probability load as well, without adversely affecting the improved symmetric load performance provided by Scheme-I (if so desired, when the traffic follows established pattern).

In general the access protection limits can be defined in terms of *an access weightage parameter*, as follows :

$$\hat{P}_i^A = \frac{(N - i)(1 + \alpha) - (1 - \alpha)}{2} ,$$

— subject to the condition[¶] $0 \leq \alpha \leq 1$. The above definition reduces to 802.6 protocol with $\alpha = 1$ and represents Scheme-I with $\alpha = 0$.

Thus varying the value of a such that $0 \leq a \leq 1$ we can select performance characteristics that is optimum at all types of loads. The performance limits will be that of 802.6 protocol and that of Scheme-I. In between these ranges^{||} alternate policies will have their access performance. Investigations along these lines can help to choose the right *access weightage parameter* that would yield some preferable performance under specifically known (or well established) traffic conditions. The upper protection limits along with the *access weightage parameter* provide us with a way of improving upon the delays at some affected nodes and at the same time allowing the other nodes to behave closer to the 802.6 protocol.

As evident from the analysis of the results of simulations of 802.6 protocol and that of Scheme-I, such schemes help the nodes that are closer to the frame generating head-end of a bus to exercise access protection there by reducing the large delays for accessing the bus at high load situations. This will be an significant improvement without having to sacrifice the performance of the end node with regard to heavy asymmetric or equal probability loads.

In a given environment the workload observed tends to remain the same over a given planning period. This suggests that a successful characterization of the optimal value of the *weightage parameter* as a function of the workload type would result in a family of access protection schemes, each tuned for a particular workload situation.

References

- [1] J. Filipiak, "Access Protection for Fairness in a Distributed Queue Dual Bus Metropolitan Area Network," Proceedings of the International Conference on Communications, Boston, June 11-14, 1989, pp. 635-639.
- [2] M. Conti, E.Gregori, and L.Lenzini, "DQDB Media Access Control Protocol: Performance Evaluation and Unfairness Analysis," Proceedings of the

[¶] α can not take values more than 1, for that would exceed 802.6 protocol under which a node can have utmost one pending *REQUEST* only. Thus using $\alpha > 1$ in effect will have no impact and the basic 802.6 protocol performance is retained.

Depending on the value of α , the protection limits can be non-integers. The APS discussed in this paper has protection limits in multiples of 0.5 and we switched between the two integer values on either side of it, during successive use of such non-integer limits. Other situations need special attention.

^{||}Of 802.6 protocol and Scheme-I protection limit definitions, 802.6 is the upper bound. $P_{802.6} - P_{\text{Scheme-I}} = (N - i + 1)$ and since $i \leq N$ this quantity is always positive.

- Third IEEE Workshop on MAN, Dana Point, CA, March 28-30 1989, pp. 375-408.
- [3] M. W. Garrett and S. Q. Li, "A Study of Slot Reuse in Dual Bus Multiple Access Networks," Proceedings of IEEE Infocom'90, pp. 617-629.
 - [4] J. W. Wong, "Throughput of DQDB Networks under heavy load," Proceedings of EFOC/LAN 89, Amsterdam, The Netherlands, Jun 11-16, 1989, pp. 146-151.
 - [5] K. Sauer, and W. Schödl, "Performance Aspects of the DQDB Protocol," ITC Specialist Seminar, Adelaide, 1989, Paper No. 16.3.
 - [6] T. Stock., "Influences of Multiple Priorities on DQDB Protocol Performance," Proceedings of the 13th ITC, Copenhagen, 1991, pp. 947-952.
 - [7] B. G. Kim, "Packet Delays in the IEEE 802.6 DQDB Protocol," Proceedings of the International Conference on Communications, pp. 346.4.1-346.4.5, 1990.
 - [8] H. Kaur, and G. Campbell, "DQDB - An Access Delay Analysis," Proceedings of IEEE Infocom 1990, pp. 630-635.
 - [9] L. N. Kumar, and A. D. Bovopoulos, "A Scheme for Dynamic Assessment of Network Topology in DQDB," Technical Report WUCS-91-40, Department of Computer Science, Washington University, St.Louis, 1991.
 - [10] L. N. Kumar, and A. D. Bovopoulos, "History Based Schemes for Improving Low Load Performance in DQDB," Technical Report WUCS-91-41, Department of Computer Science, Washington University, St.Louis, 1991.

Performance Characteristics of Scheme-I

(Different Load Types)

Details of Simulation :

Simulation Duration = 3.0 seconds
SlotInterval = 8.59 microsec.
Configuration = 100 Nodes @ 0.5 Kms apart.
Channel Capacity = 44.7 Mbps
Simulated Load = 0.75, 0.9, 1.0, 1.2 & 1.8
Average Message Size = 1 pkt/arrival

Arrival Patterns :

Poisson Arrival
Graded Work Load
Bursty Traffic (Bursts of size - 2, 4, 6, 8, 10)

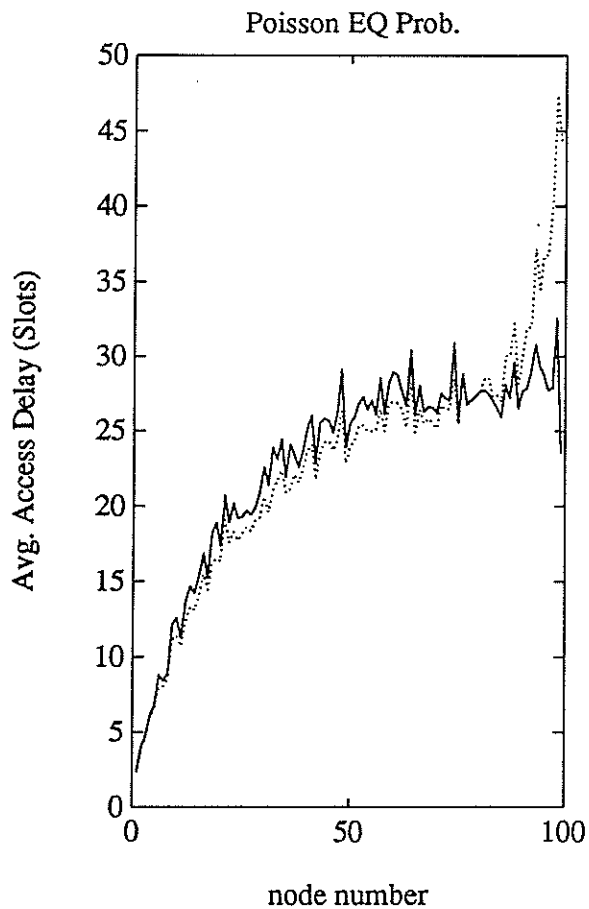
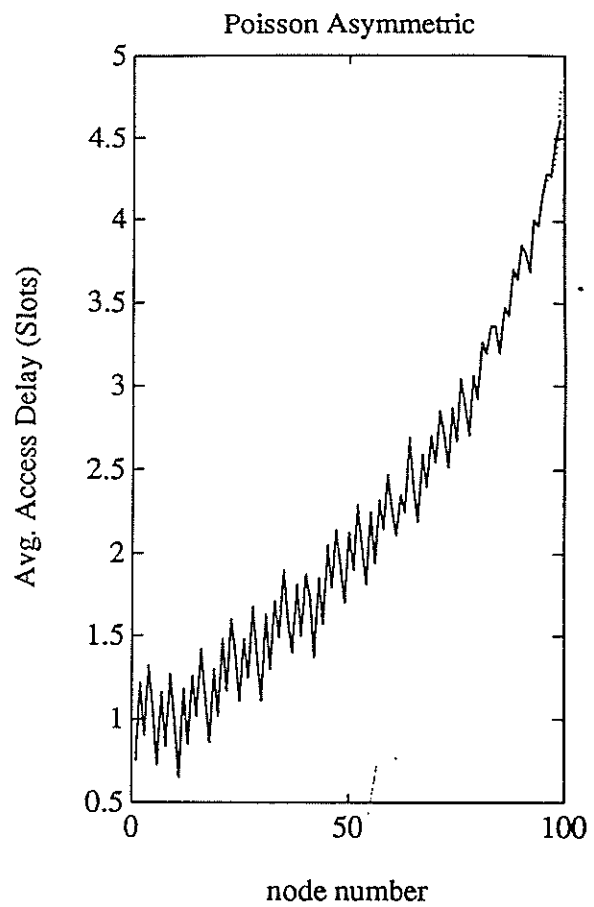
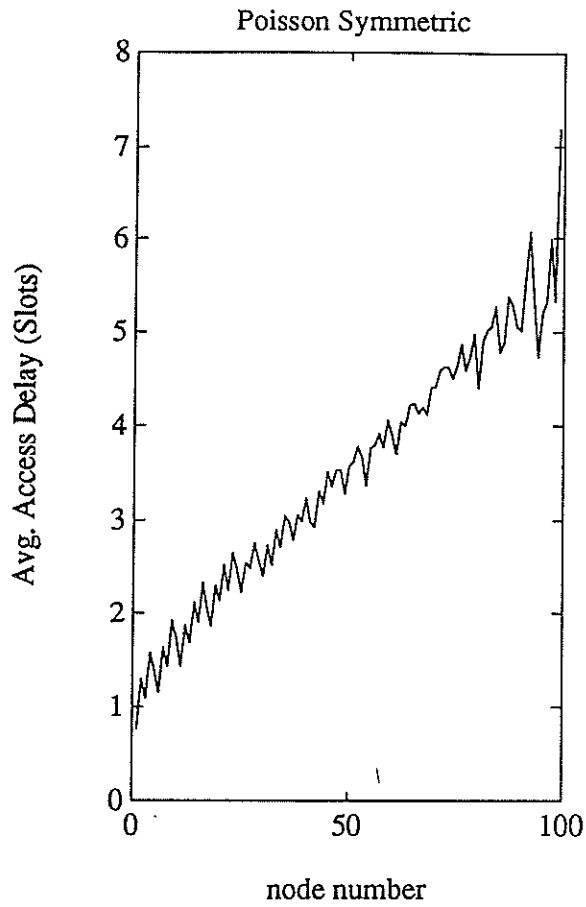
Load Types :

Symmetric Load
Asymmetric Load
Equal Probability Load

Markers Used :

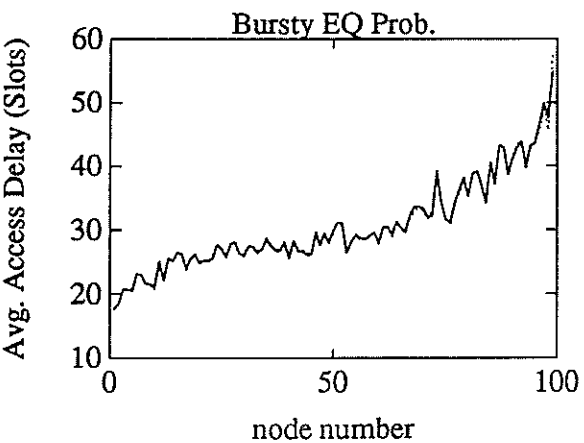
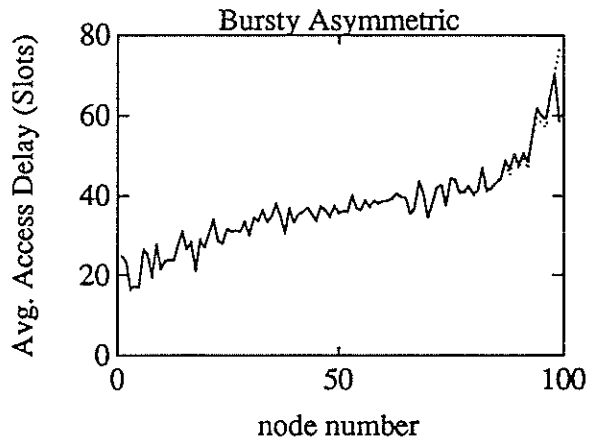
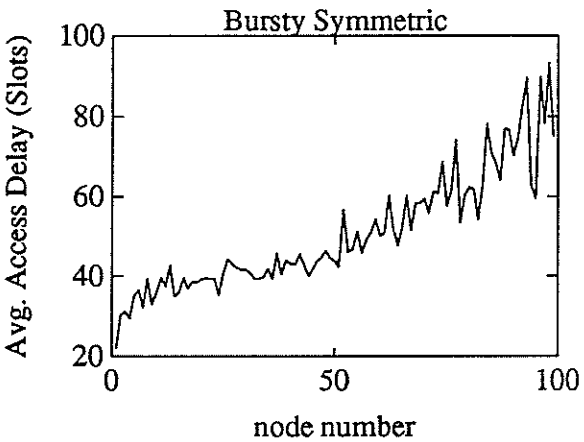
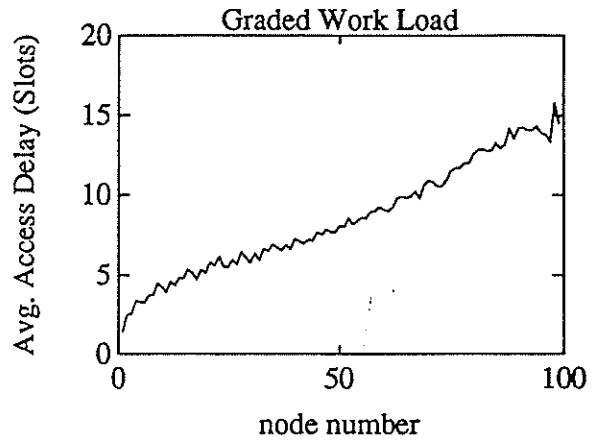
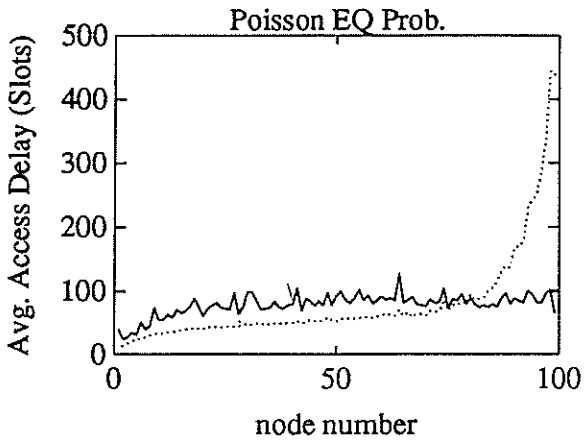
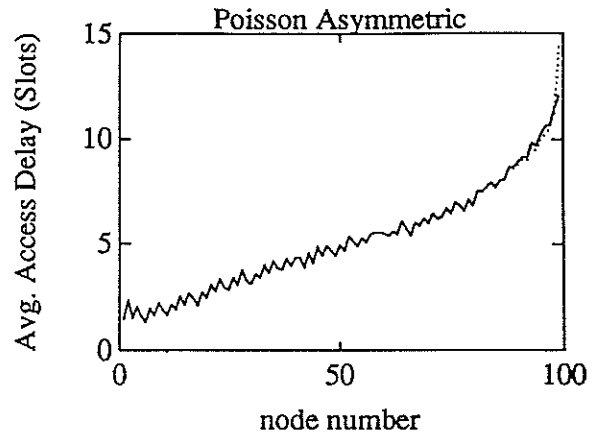
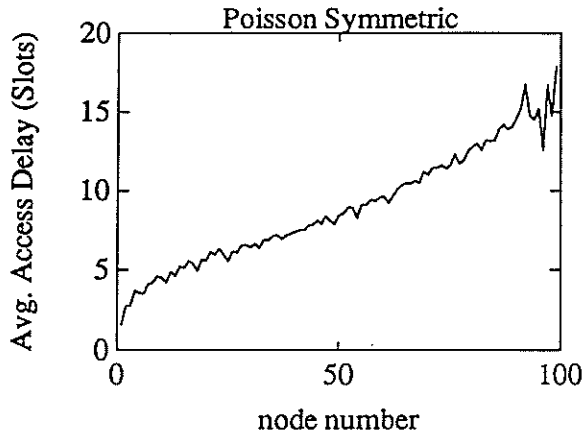
_____ 802.6 Performance Characteristic
..... Scheme-I Performance Characteristic

(The Characteristics are with respect to Bus-A access.)



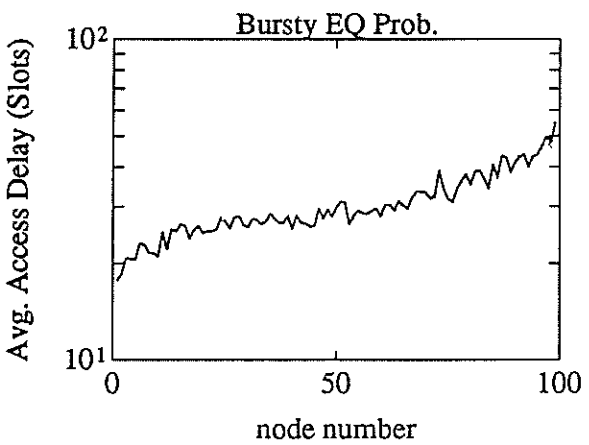
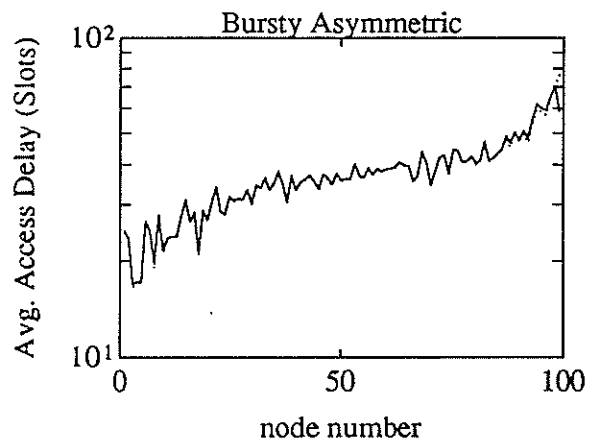
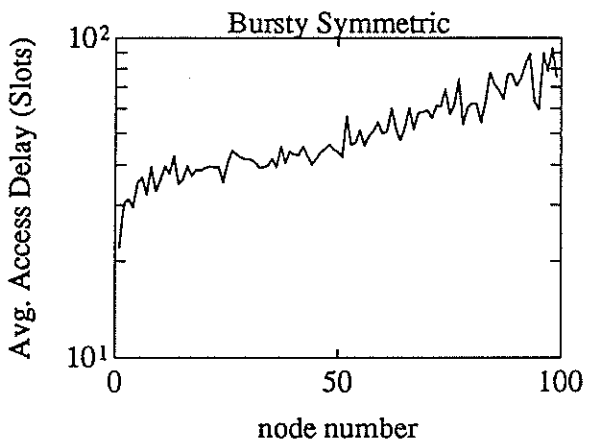
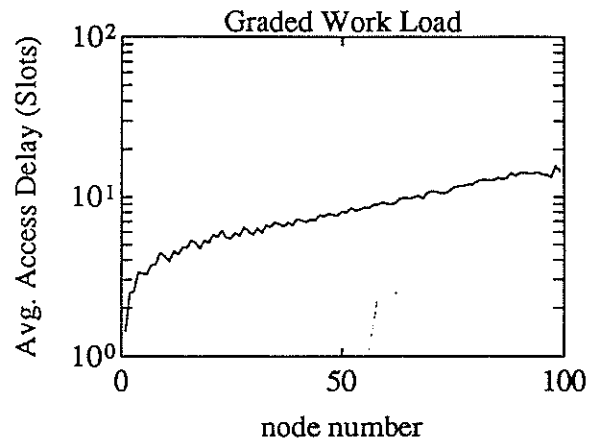
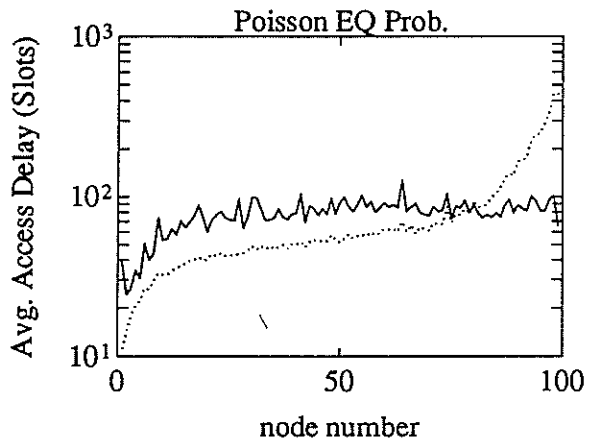
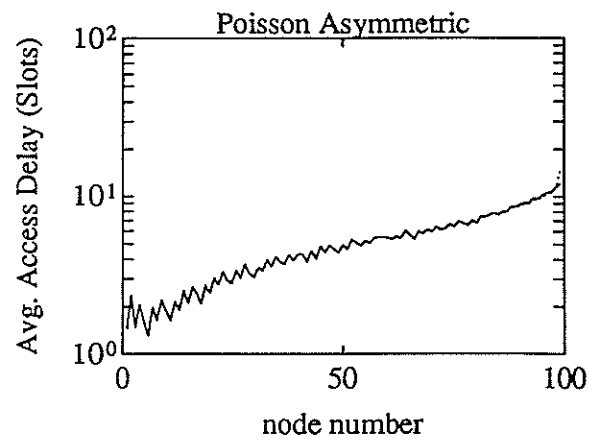
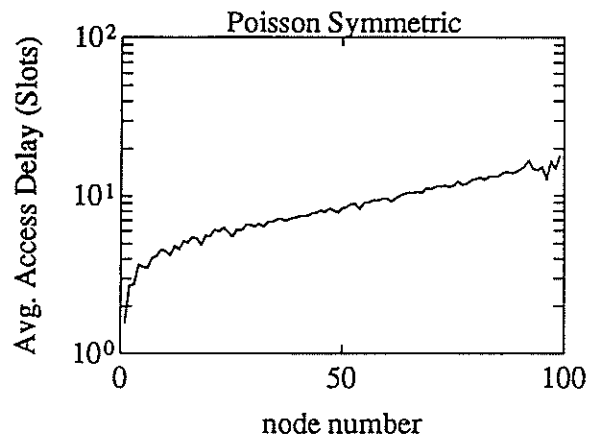
**Average Delay Characteristics
(Scheme-I)**

- * Load Value = 0.75
- * Single Packet messages
- * 100 nodes @ 0.5 Kms apart
- * (No / little variation for other loads)



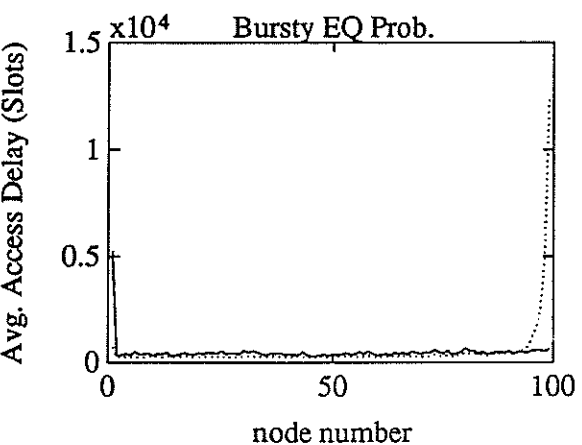
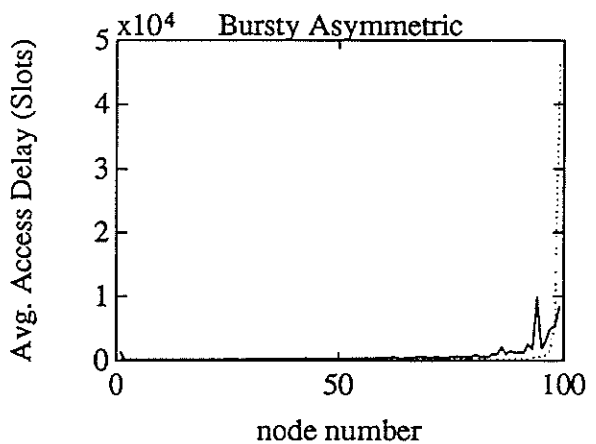
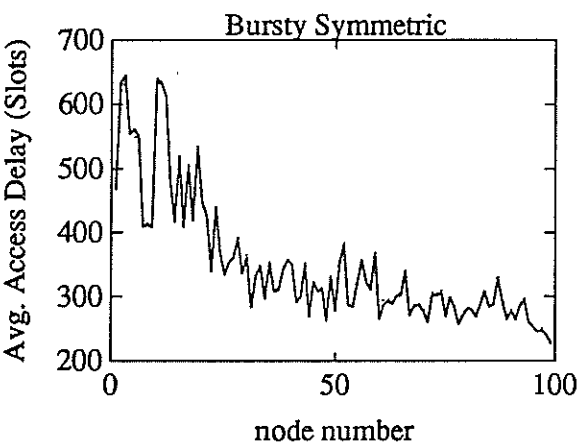
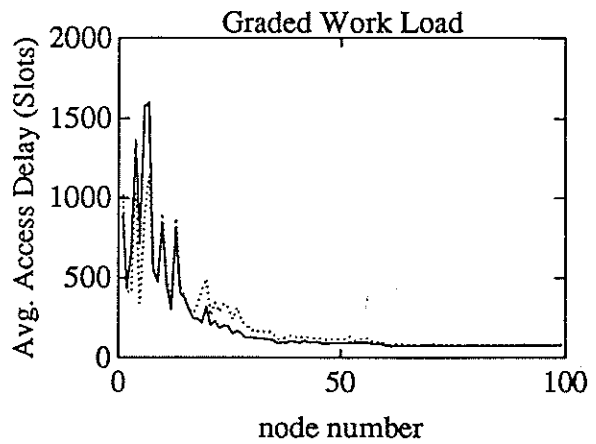
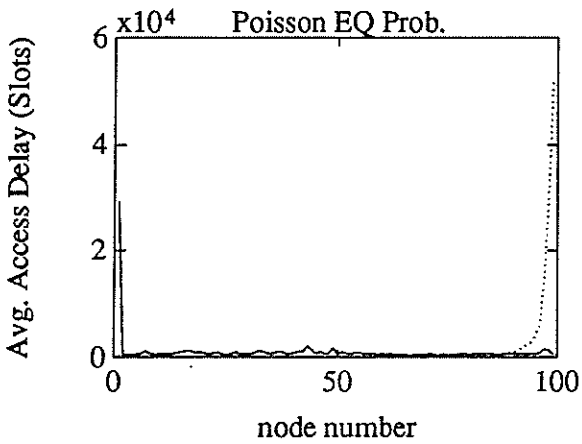
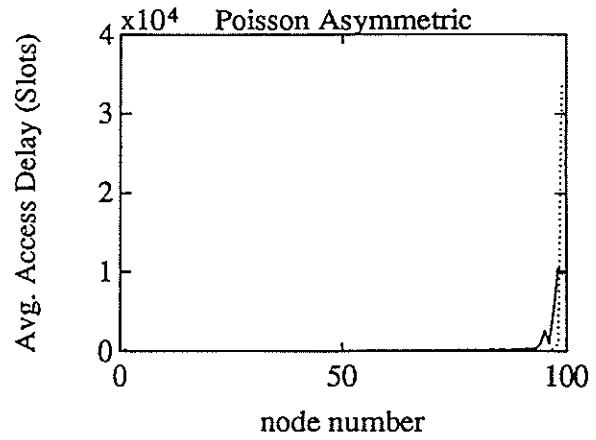
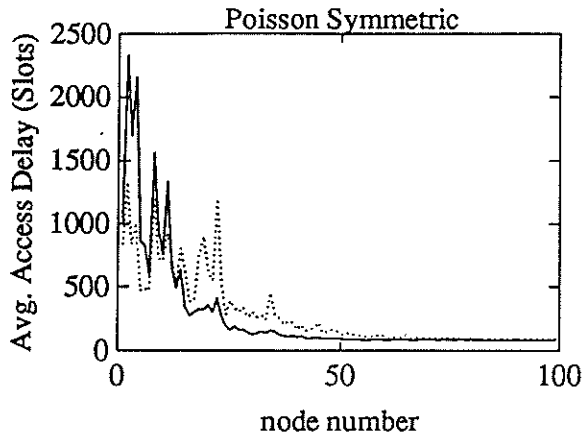
**Average Delay Characteristics
(Scheme-I)**

- * Load Value = 0.9
- * Single Packet Messages
- * 100 nodes @ 0.5 Kms apart



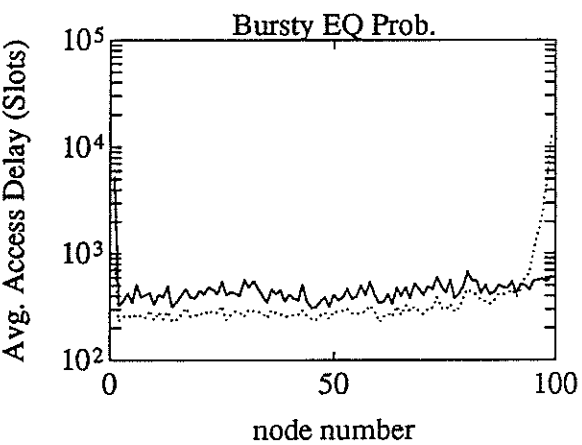
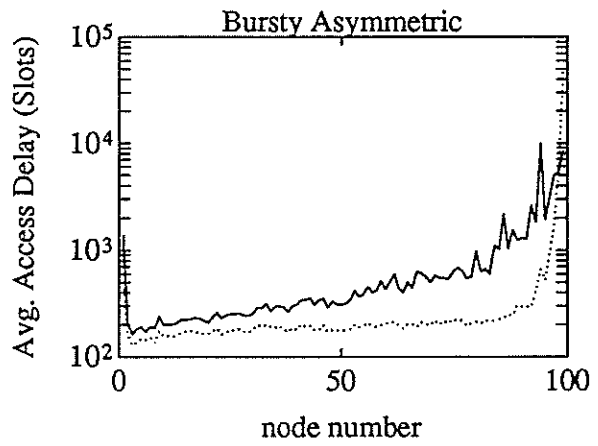
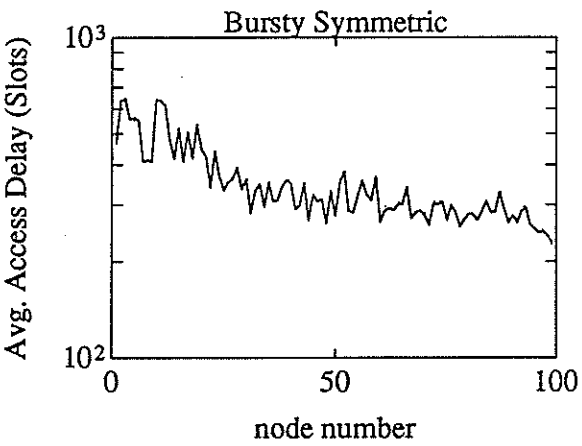
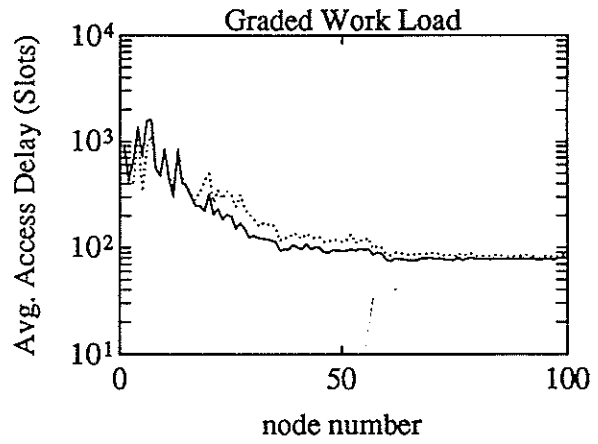
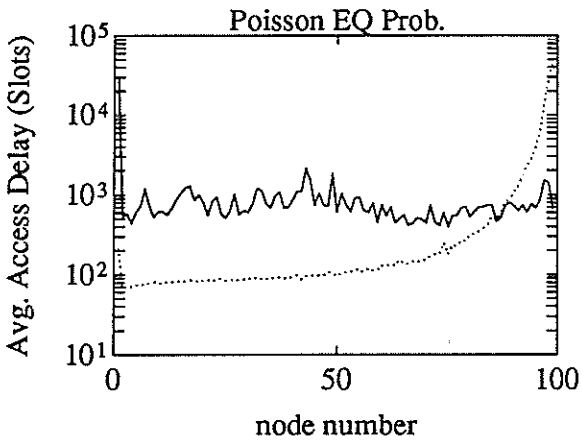
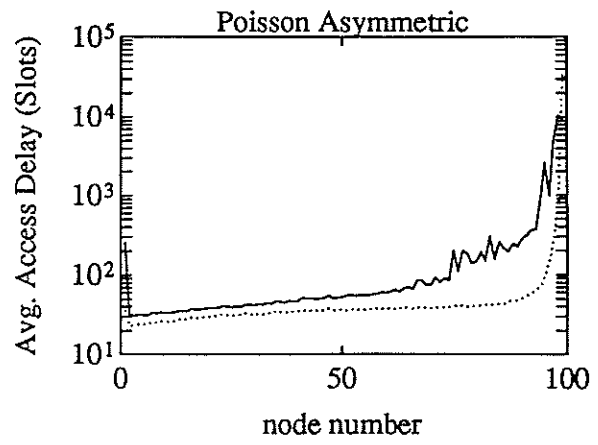
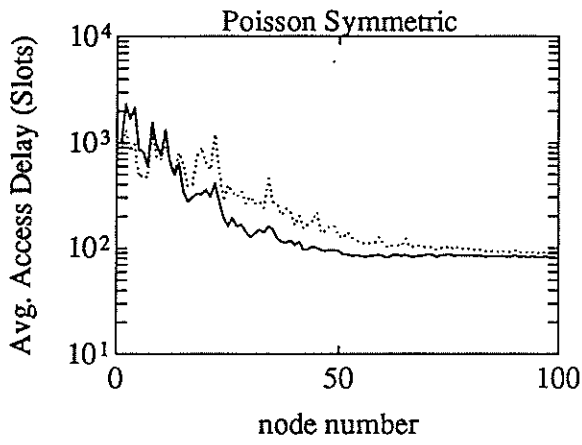
Average Delay Characteristics
(Scheme-I - Semilog plot)

- * Load Value = 0.9
- * Single Packet Messages
- * 100 nodes @ 0.5 Kms apart



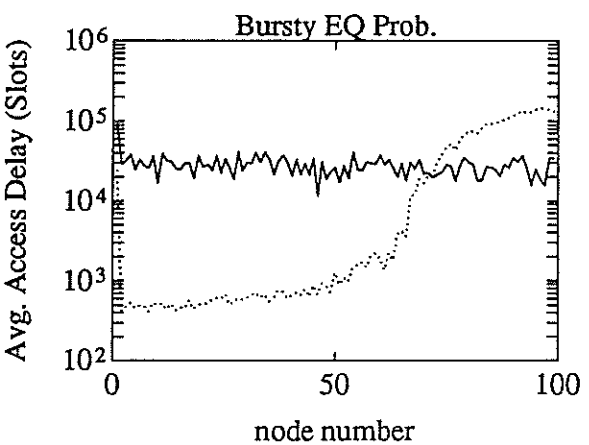
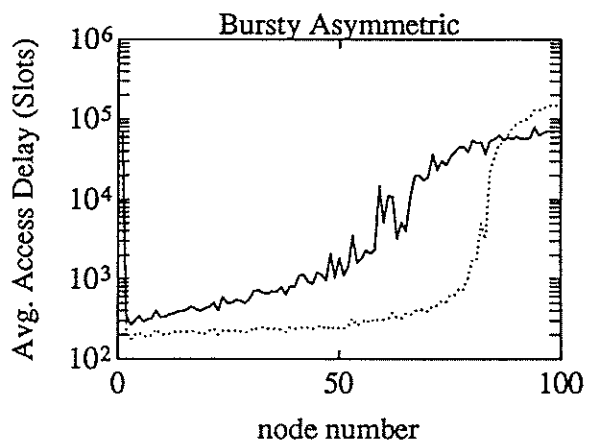
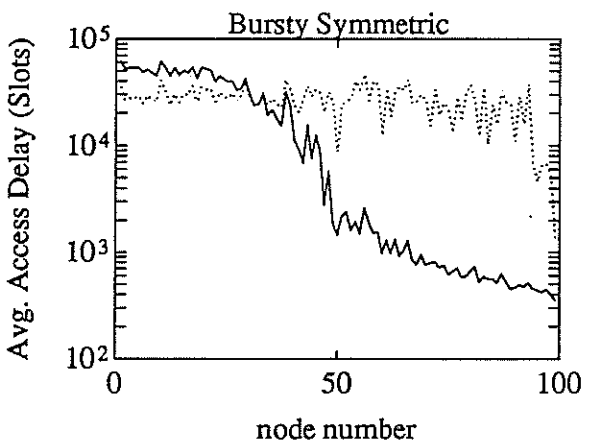
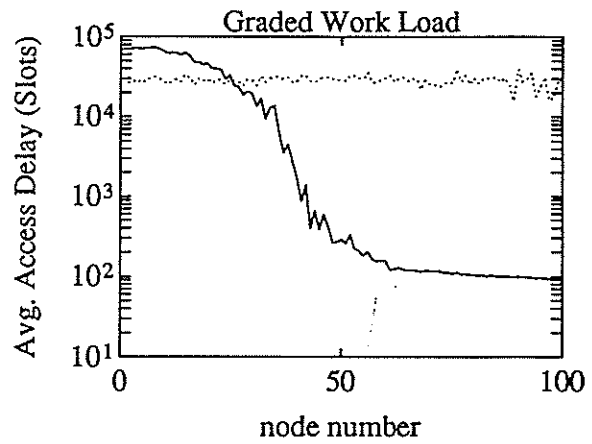
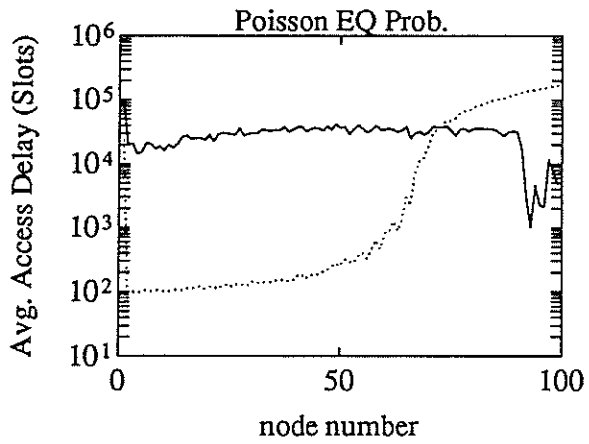
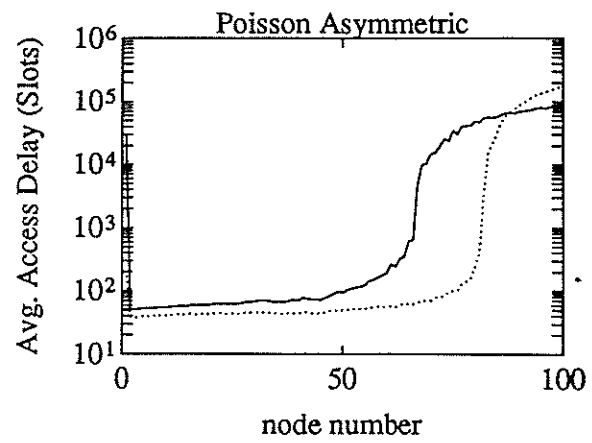
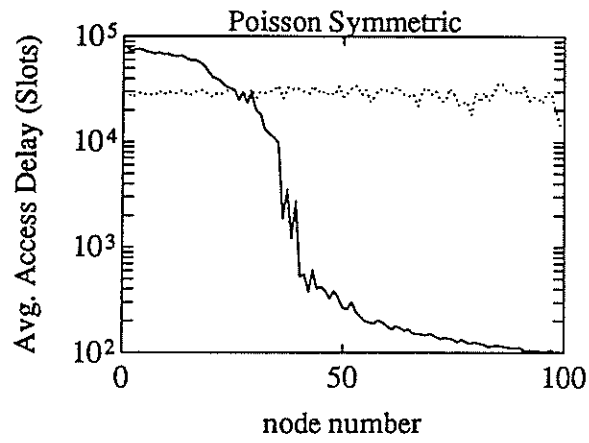
**Average Delay Characteristics
(Scheme-I)**

- * Load Value = 1.0
- * Single Packet messages
- * 100 nodes @ 0.5 Kms apart



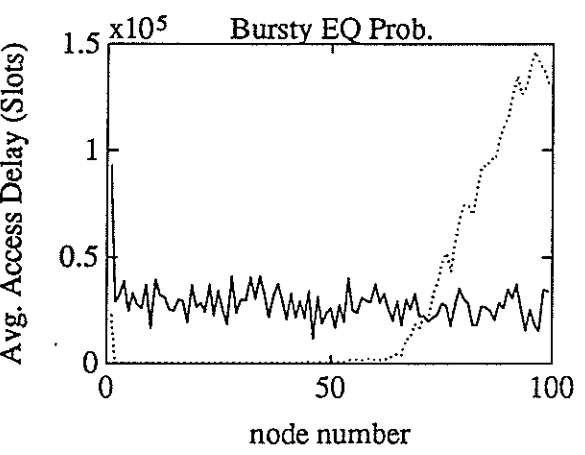
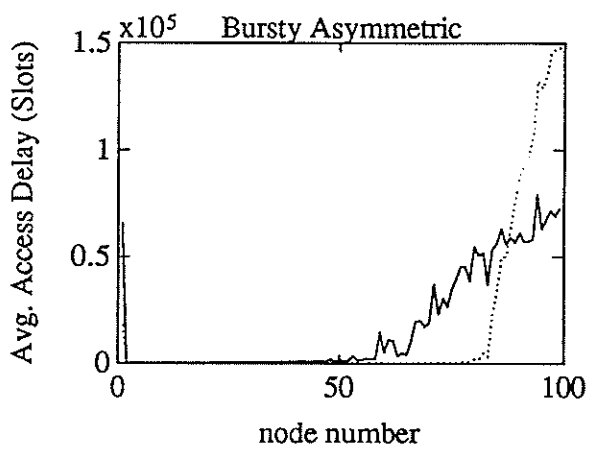
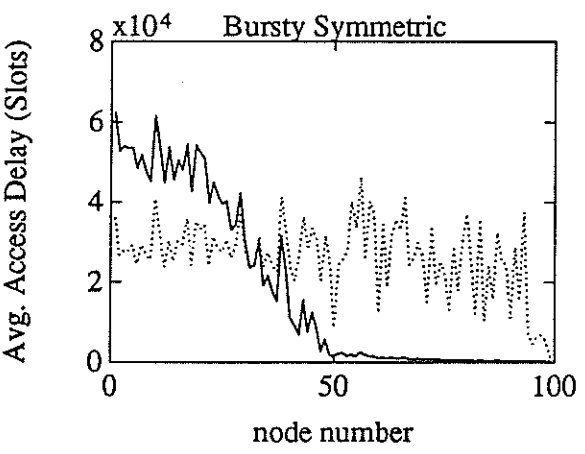
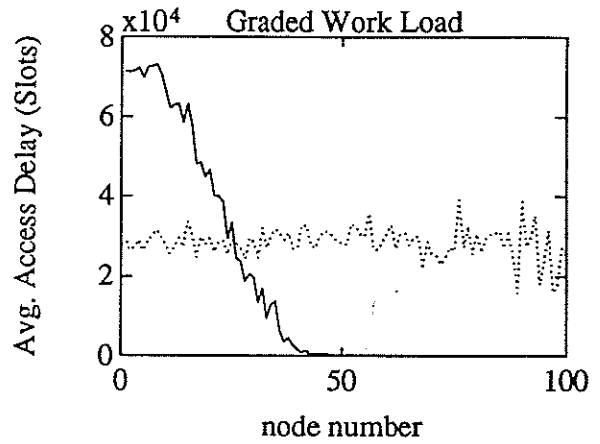
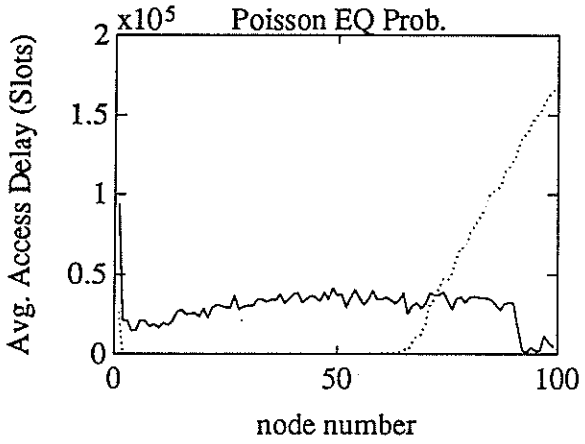
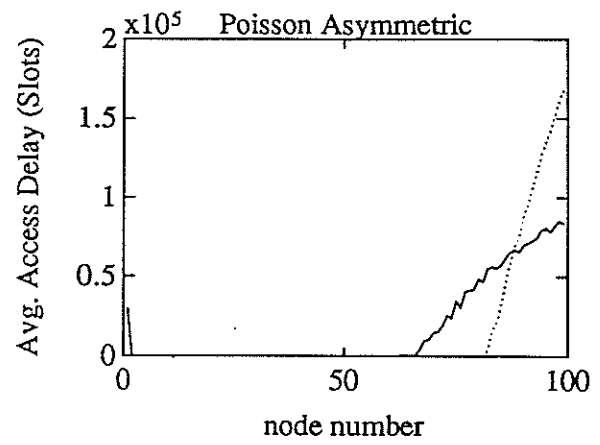
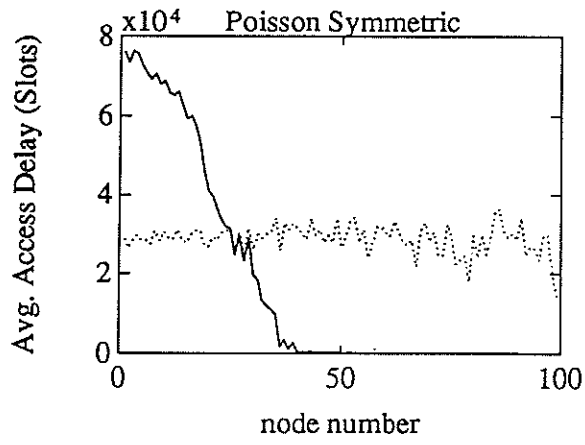
Average Delay Characteristics
(Scheme-I - Semilog Plot)

- * Load Value = 1.0
- * Single Packet messages
- * 100 nodes @ 0.5 Kms apart



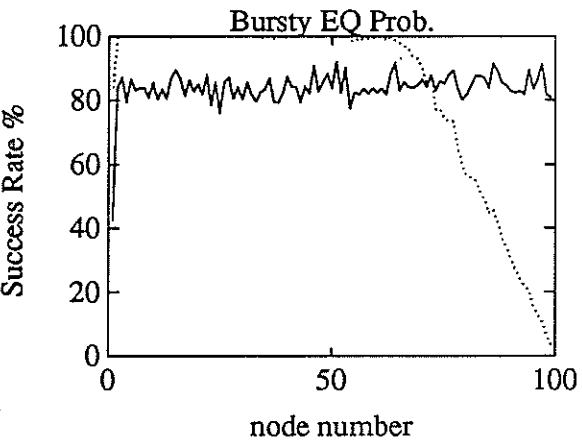
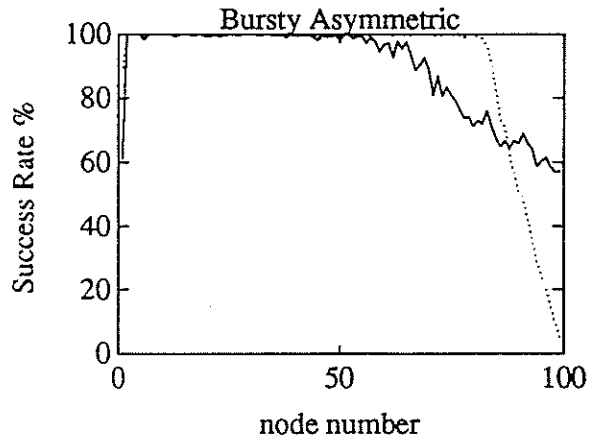
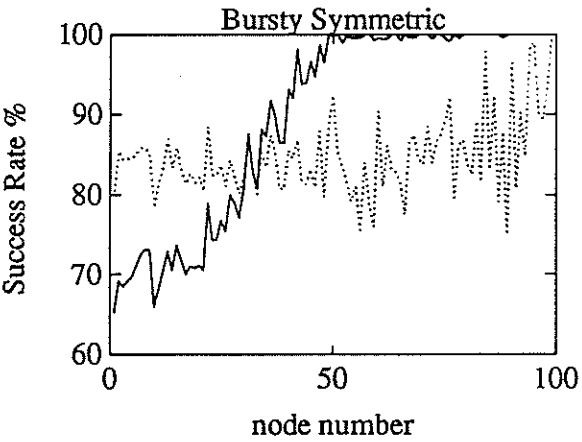
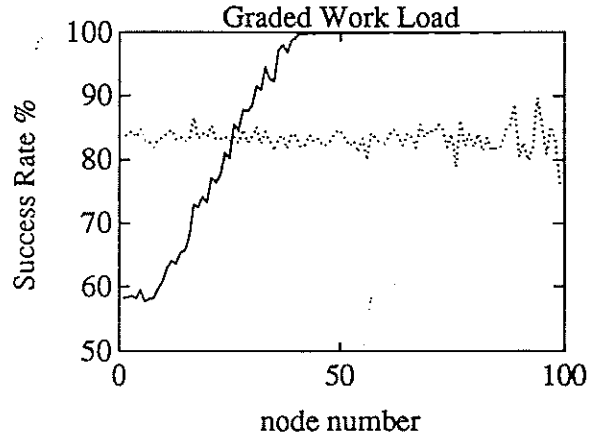
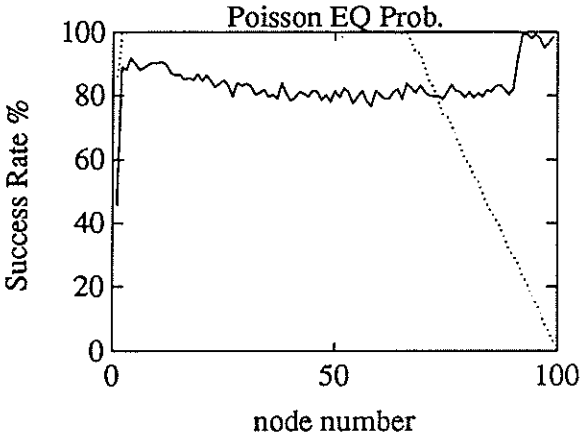
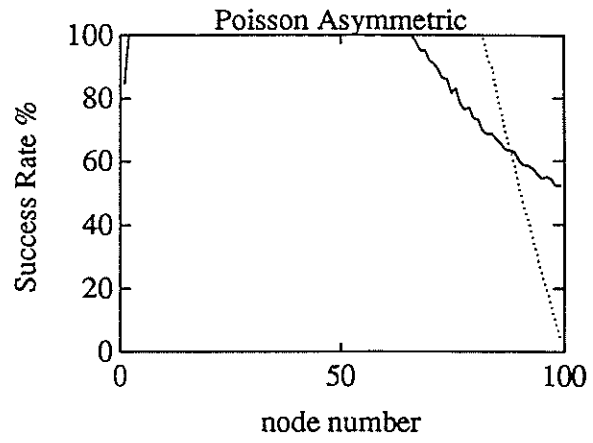
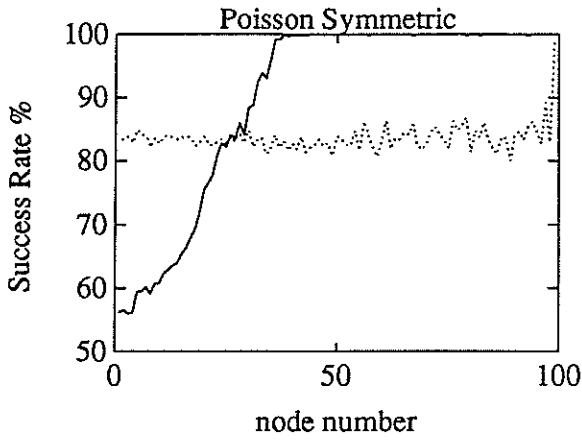
Average Delay Characteristics
(Scheme-I - Semilog plot)

- * Load Value = 1.2
- * Single Packet Messages
- * 100 nodes @ 0.5 Kms apart



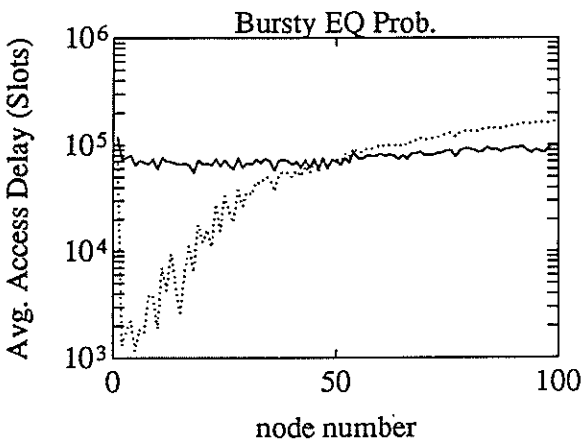
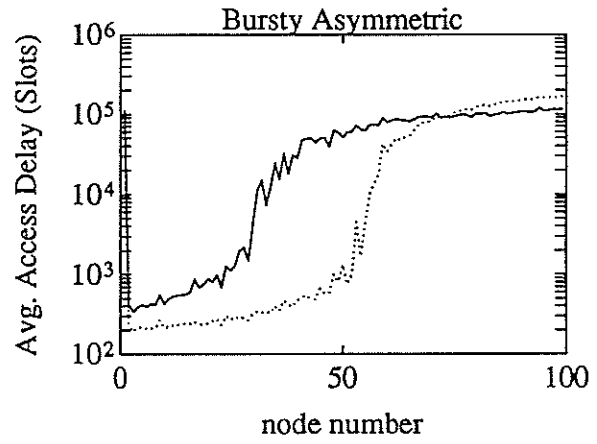
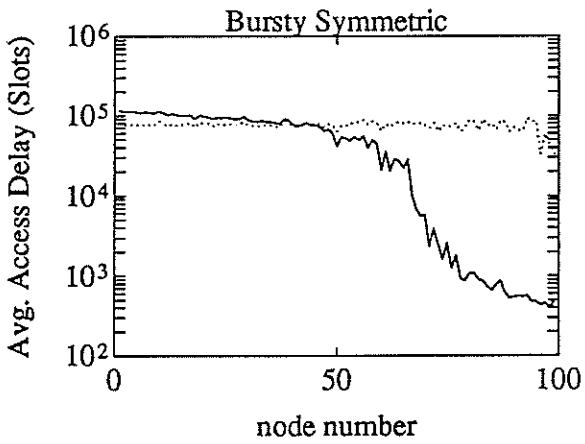
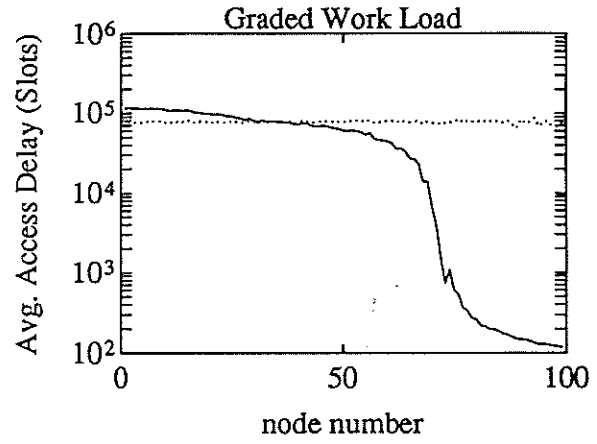
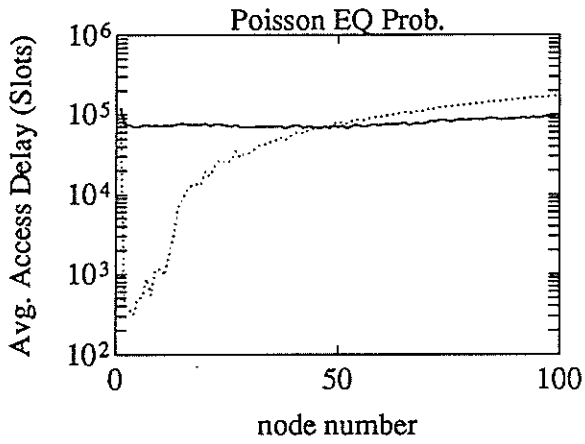
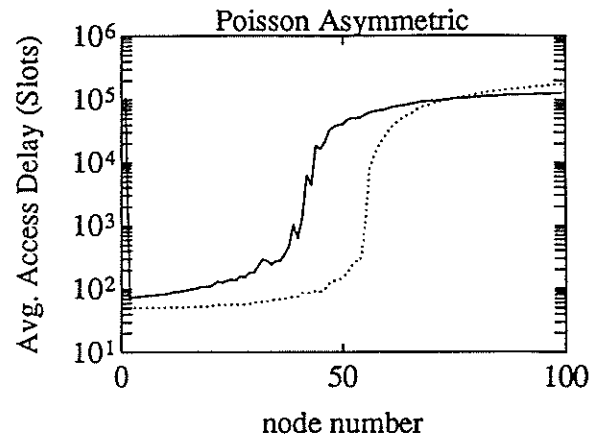
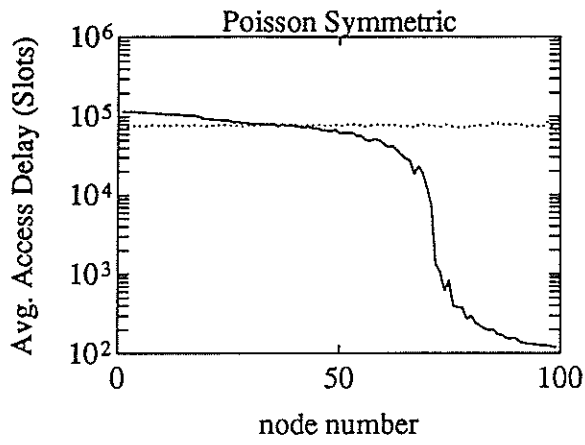
**Average Delay Characteristics
(Scheme-I)**

- * Load Value = 1.2
- * Single Packet Messages
- * 100 nodes @ 0.5 Kms apart



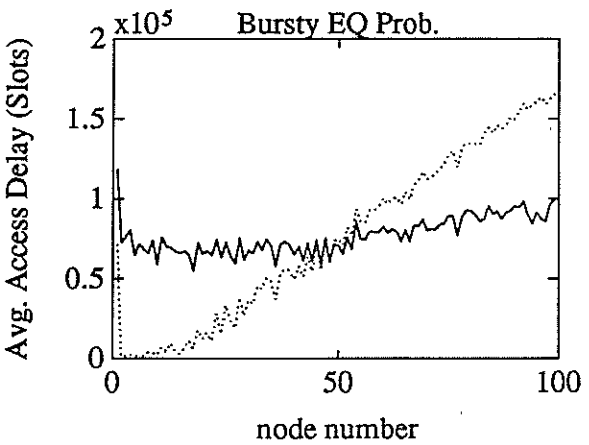
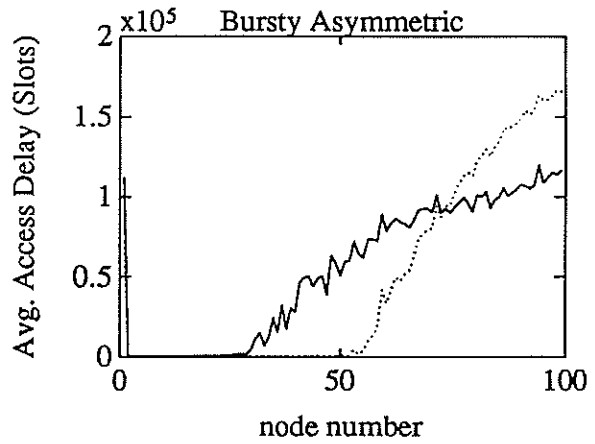
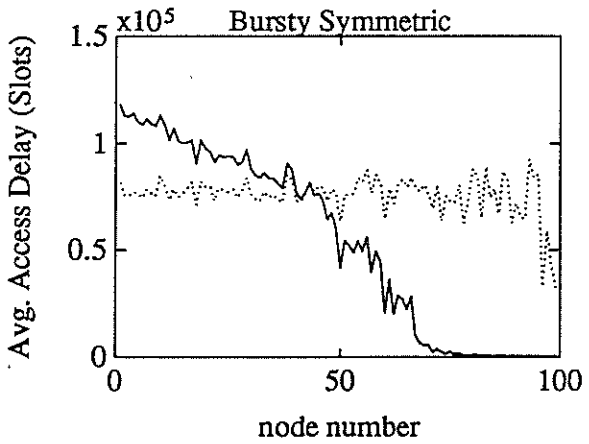
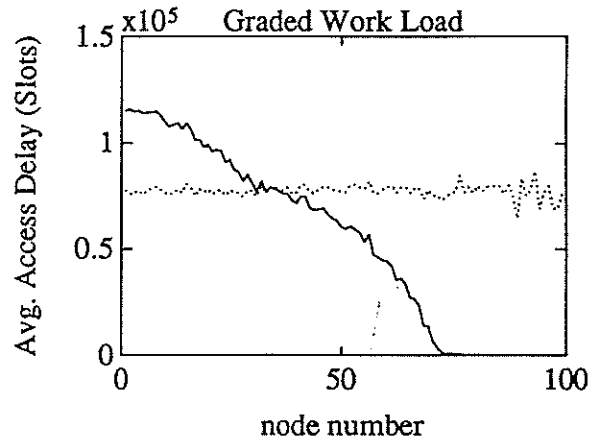
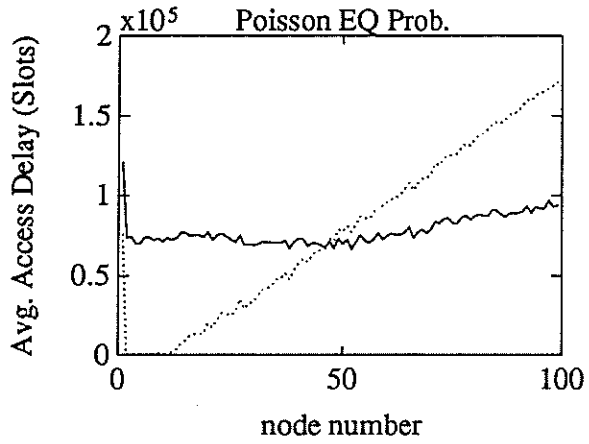
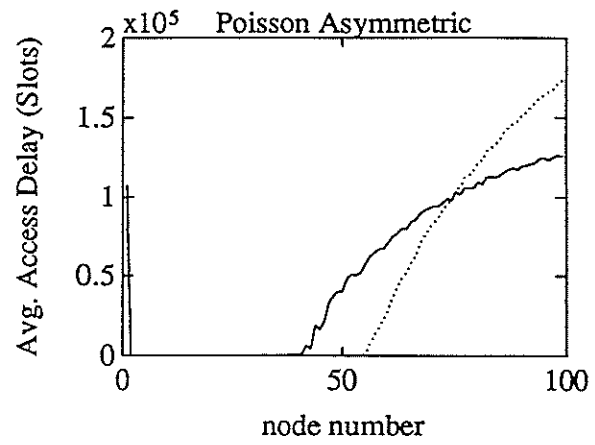
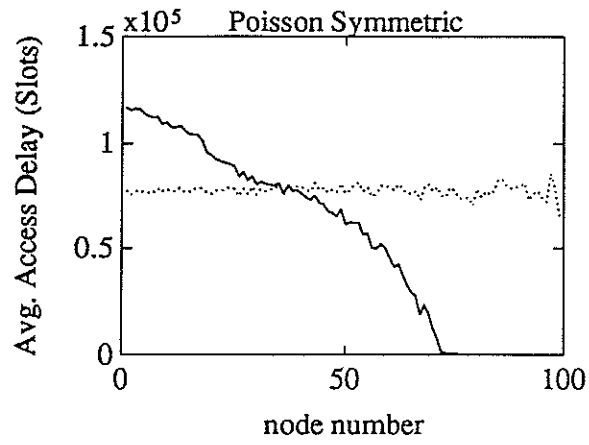
**Success Rate Characteristics
(Scheme-I)**

- * Load Value = 1.2
- * Single Packet Messages
- * 100 nodes @ 0.5 Kms apart



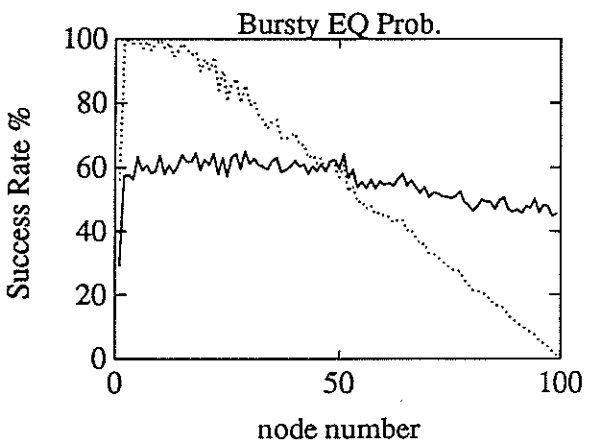
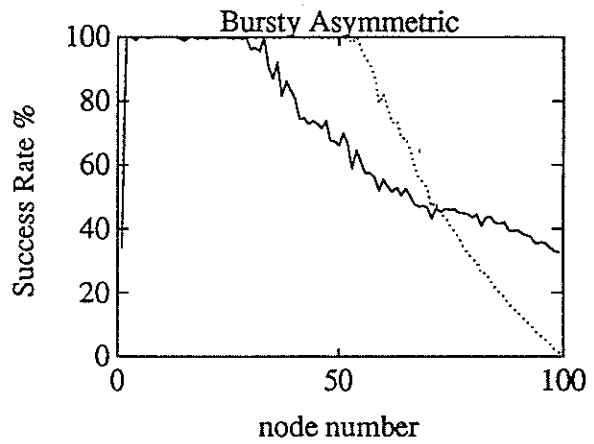
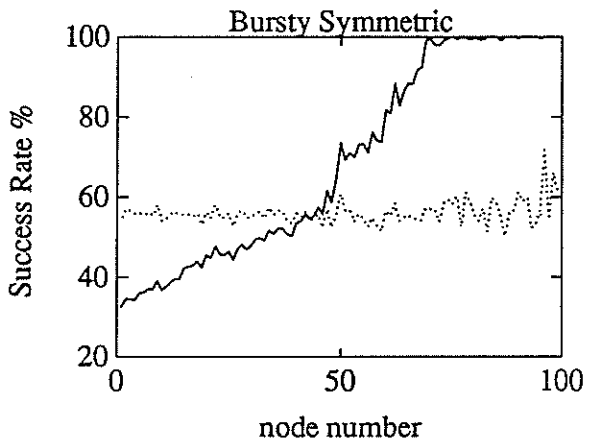
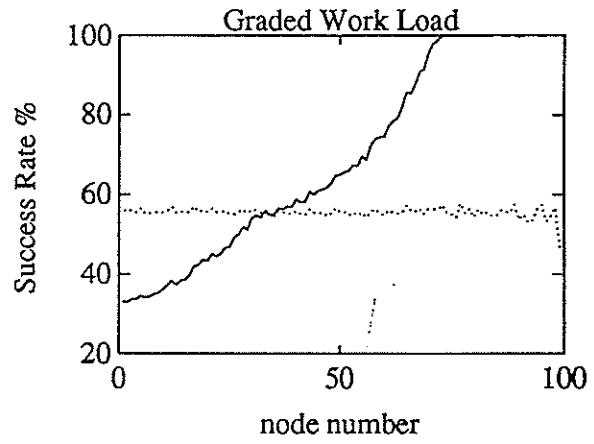
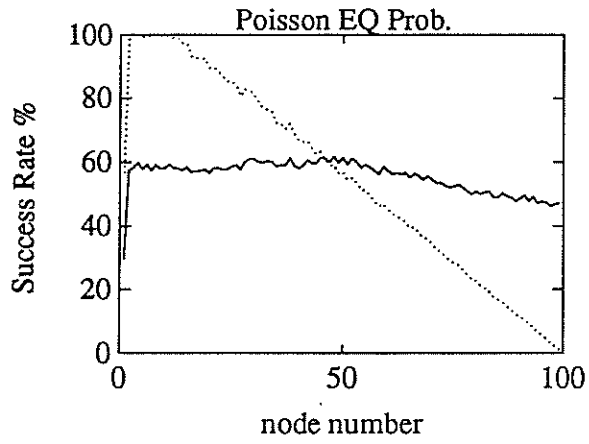
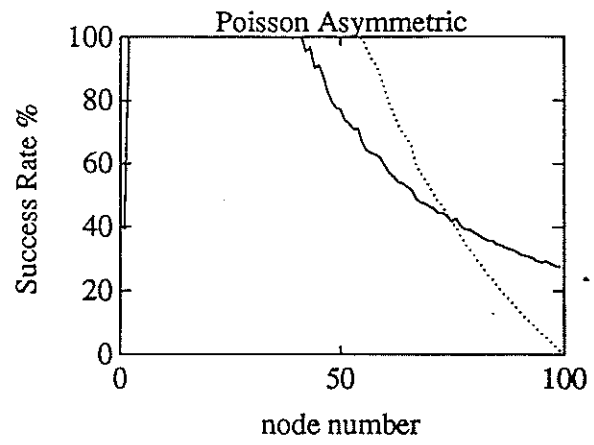
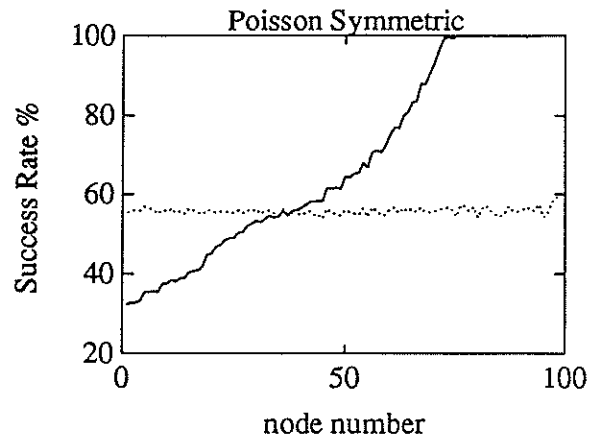
Average Delay Characteristics
(Scheme-I - Semilog Plot)

- * Load Value = 1.8
- * Single Packet messages
- * 100 nodes @ 0.5 Kms apart



**Average Delay Characteristics
(Scheme-I)**

- * Load Value = 1.8
- * Single Packet messages
- * 100 nodes @ 0.5 Kms apart



Success Rate Characteristics
(Scheme-I)

- * Load Value = 1.8
- * Single Packet Messages
- * 100 nodes @ 0.5 Kms apart

Performance Characteristics of Scheme-I

(Different Network Configurations)

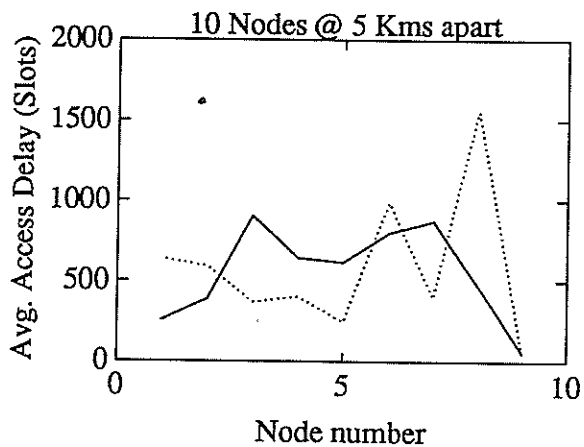
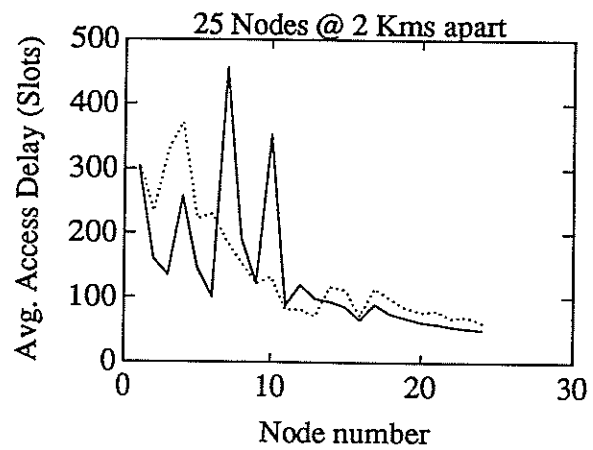
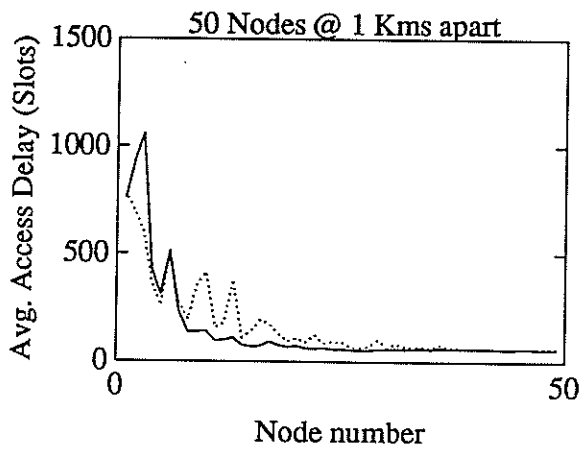
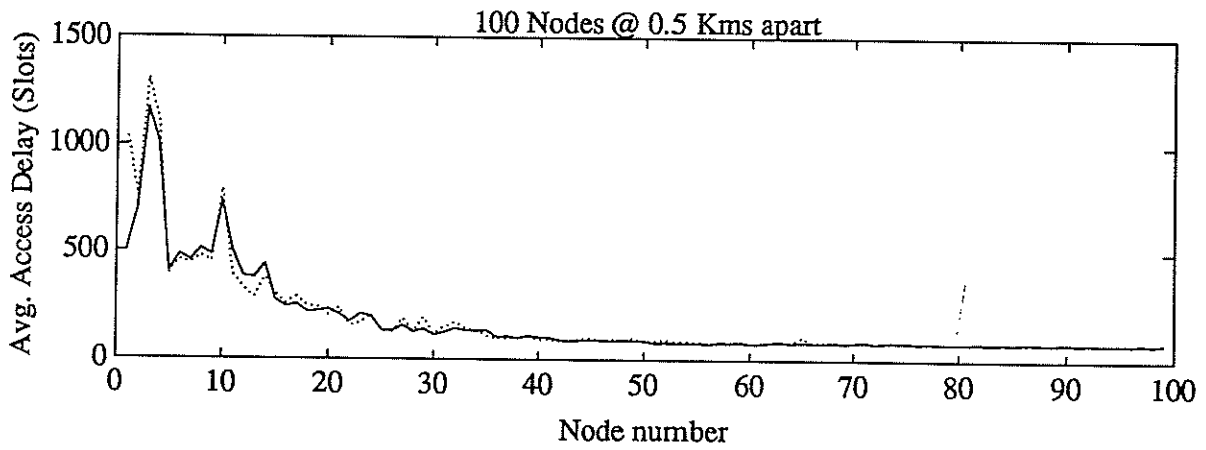
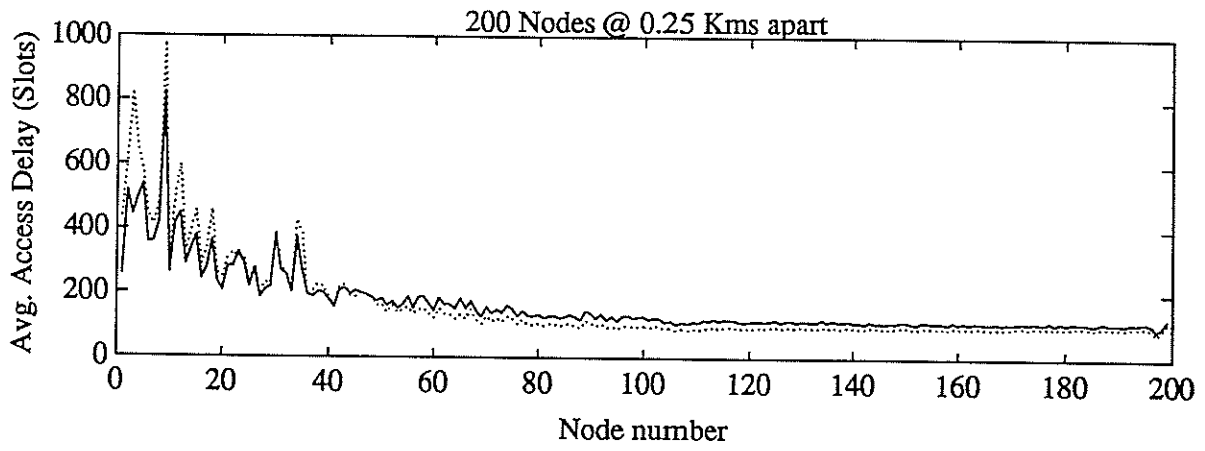
Details of Simulation :

Simulation Duration	= 2.0 seconds
Slot Interval	= 8.59 microsec.
Configuration	= 200 Nodes @ 0.25 Kms apart. 100 Nodes @ 0.5 Kms apart. 50 Nodes @ 1 Kms apart 25 Nodes @ 2 Kms apart 10 Nodes @ 5 Kms apart
Channel Capacity	= 44.7 Mbps
Simulated Load	= 1.0 & 1.2
Average Message Size	= 1 pkt/arrival
Arrival Pattern	: Poisson Arrival
Load Type	: Symmetric Load

Markers Used :

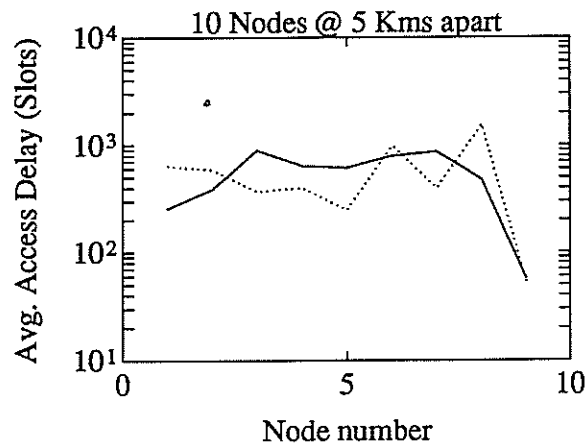
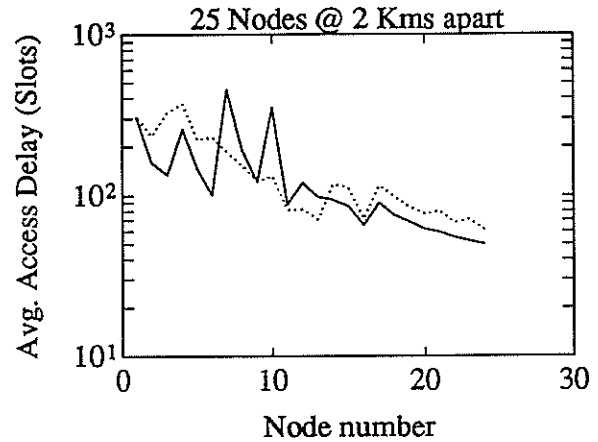
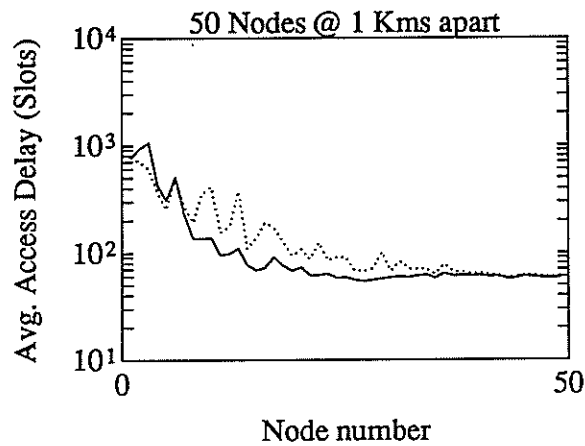
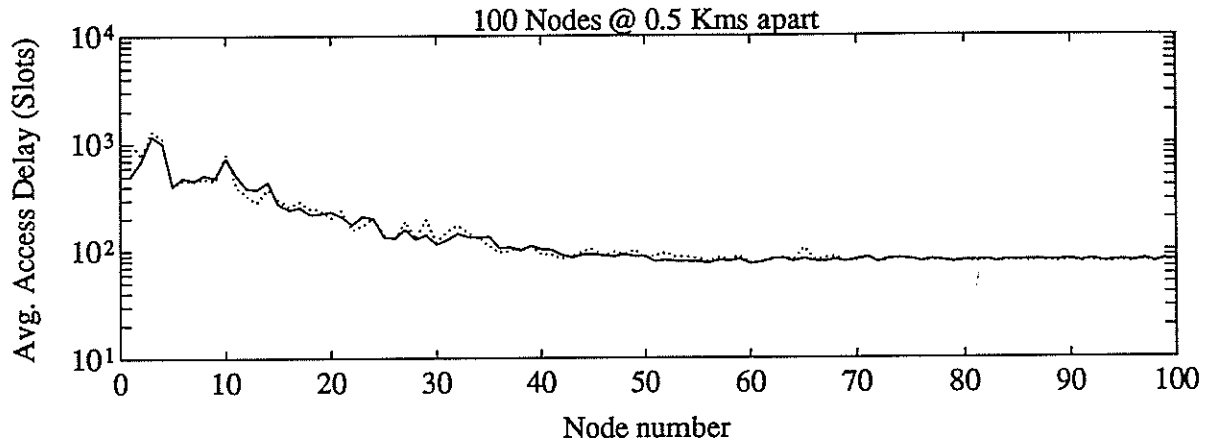
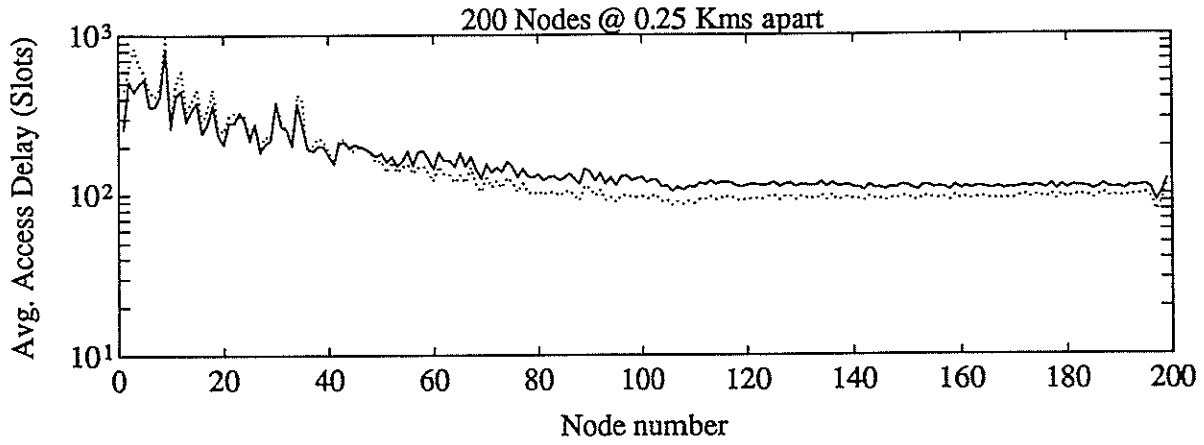
_____	802.6 Performance Characteristic
.....	Scheme-I Performance Characteristic

(The Characteristics are with respect to Bus-A access.)



Average Delay Characteristics

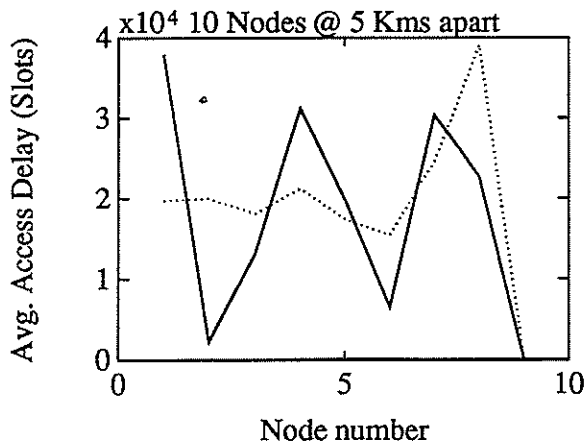
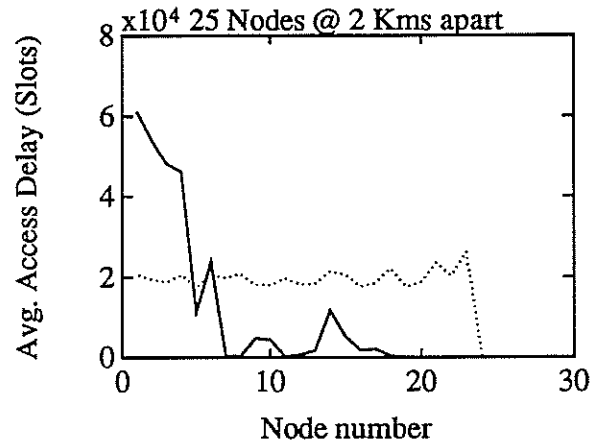
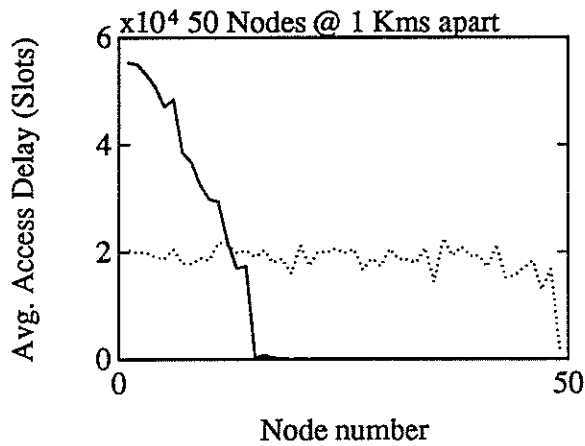
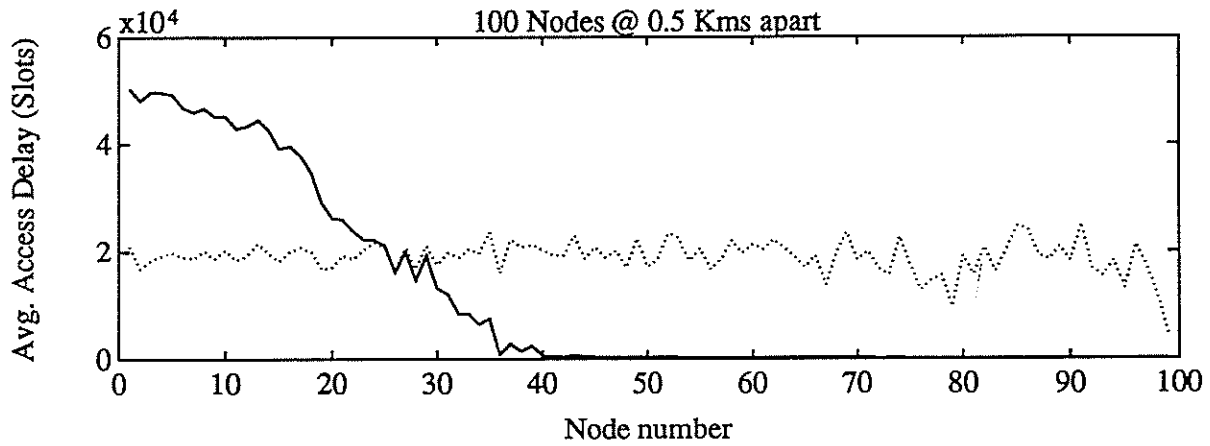
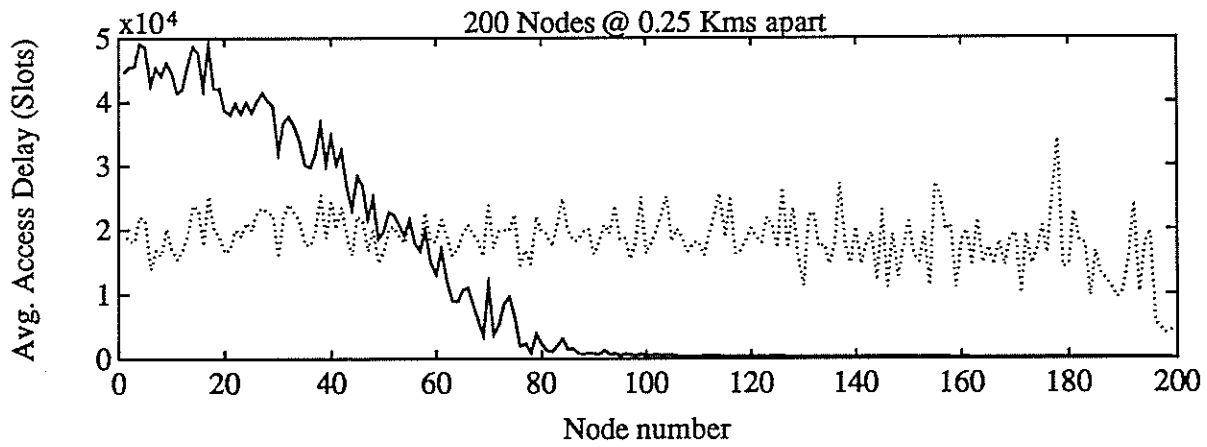
- * Load Value = 1.0
- * Single Packet Messages
- * Poisson Arrival
- * Symmetric Load Type



Average Delay Characteristics

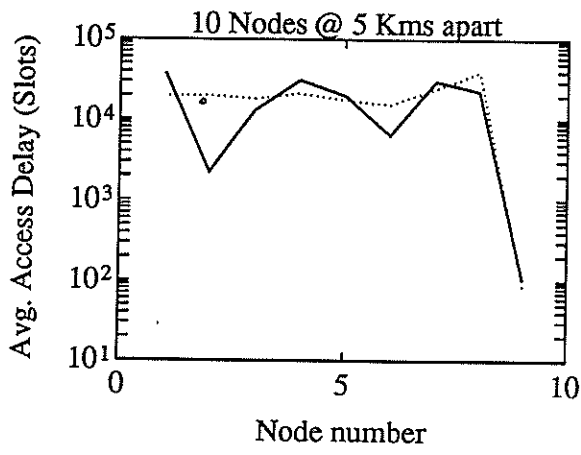
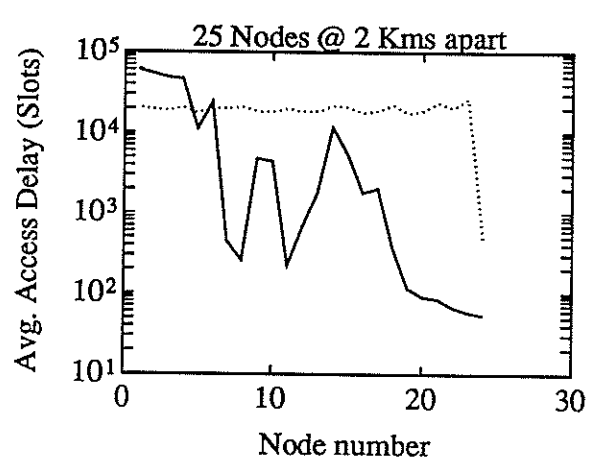
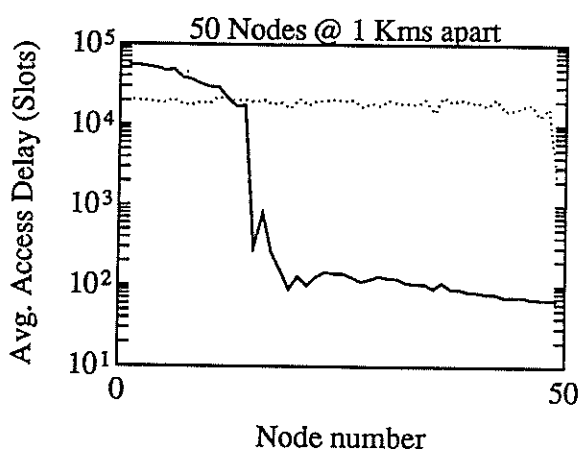
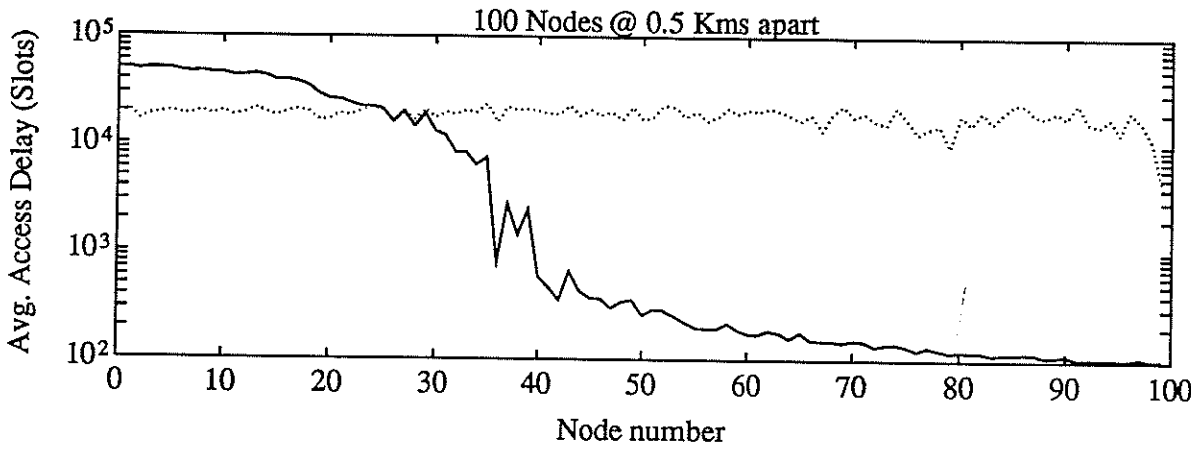
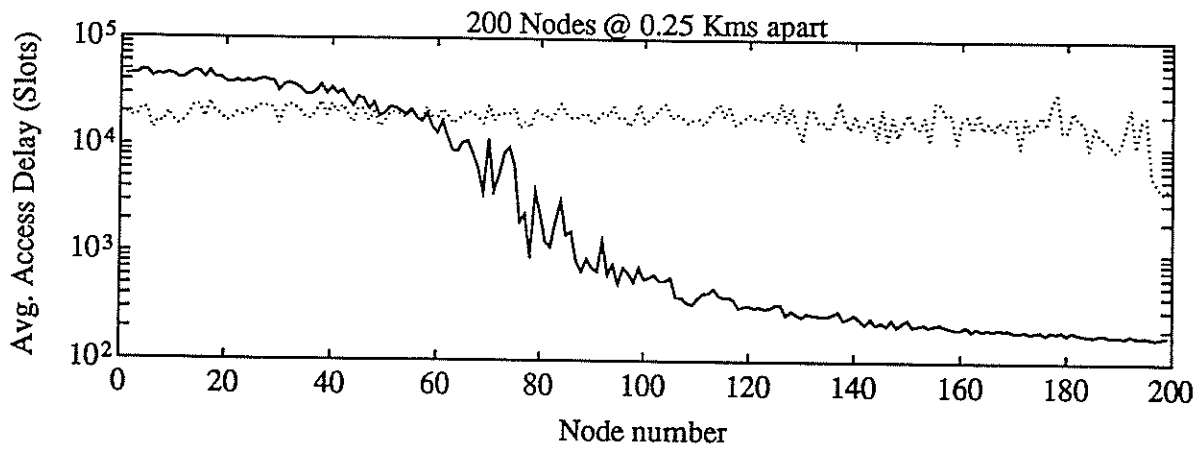
(Semilog Plot)

- * Load Value = 1.0
- * Single Packet Messages
- * Poisson Arrival
- * Symmetric Load Type



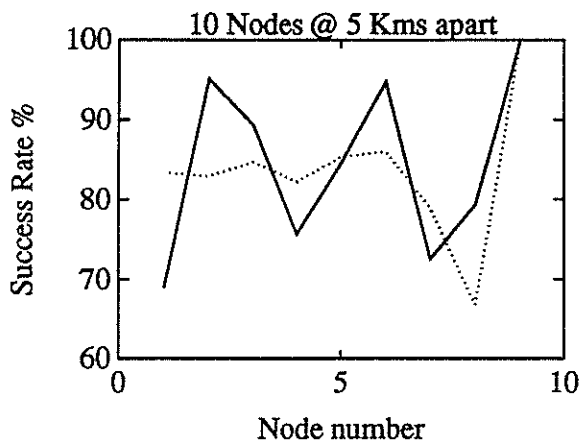
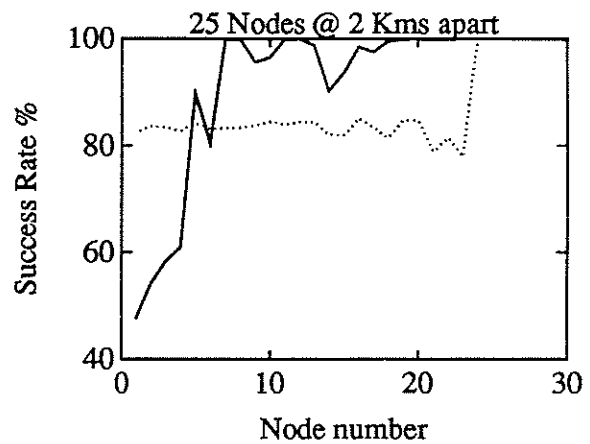
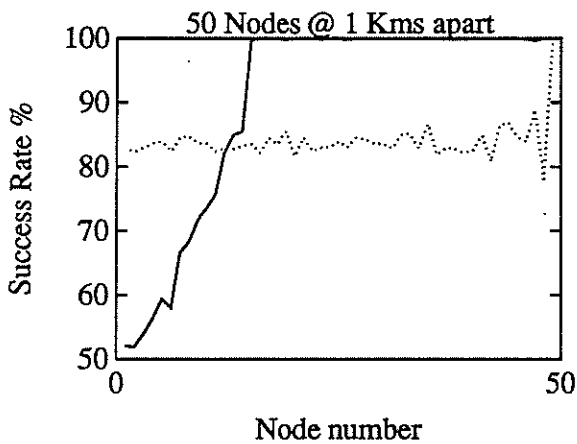
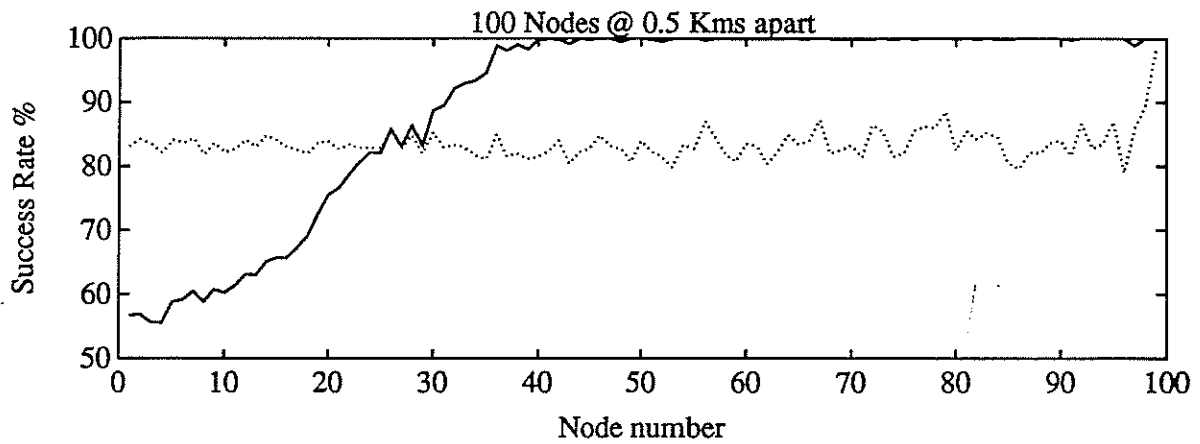
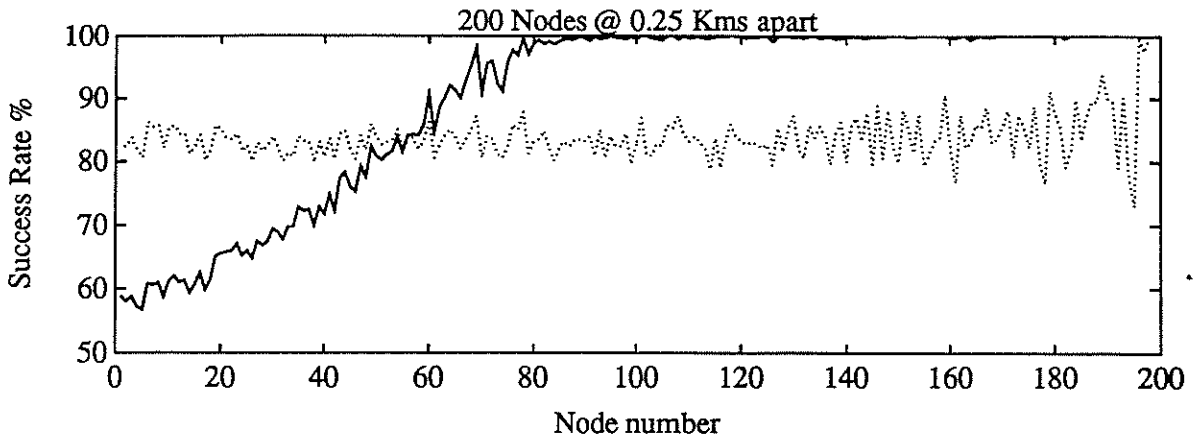
Average Delay Characteristics

- * Load Value = 1.2
- * Single Packet messages
- * Poisson Arrival
- * Symmetric Load Type



Average Delay Characteristics
(Semilog Plot)

- * Load Value = 1.2
- * Single Packet messages
- * Poisson Arrival
- * Symmetric Load Type



Success Rate Characteristics

- * Load Value = 1.2
- * Single Packet Messages
- * Poisson Arrival
- * Symmetric Load Type

Performance Characteristics of Scheme-I

(Multiple packet messages in Different networks)

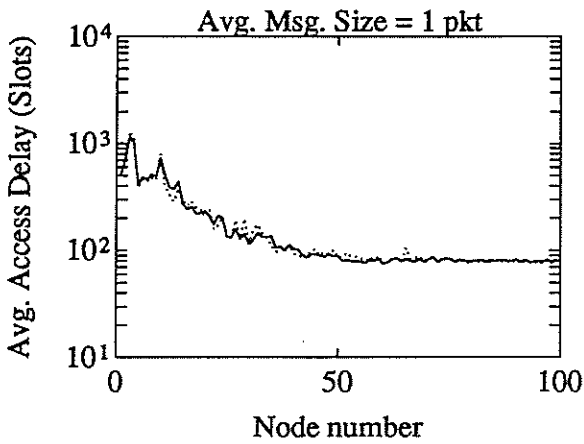
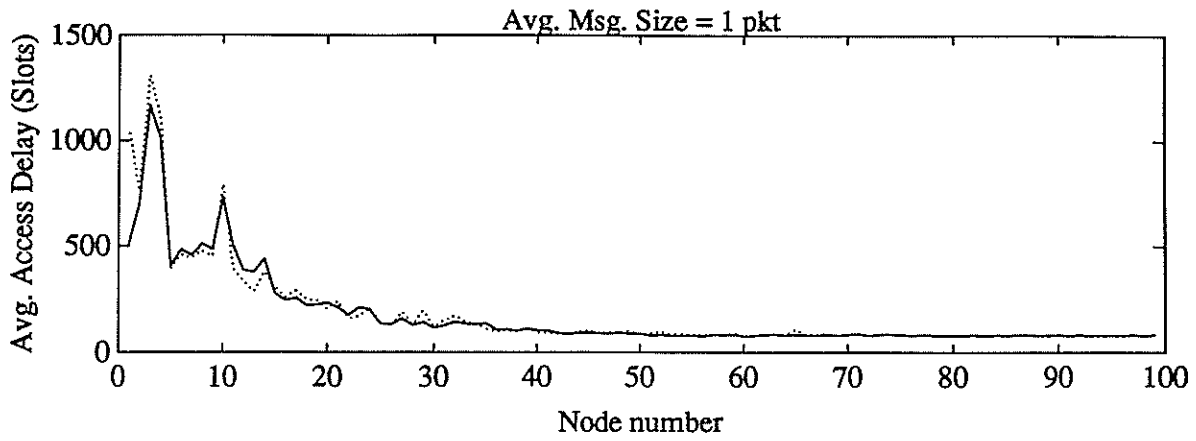
Details of Simulation :

- Simulation Duration = 2.0 seconds
- Slot Interval = 8.59 microsec.
- Configuration = 100 Nodes @ 0.5 Kms apart.
25 Nodes @ 2 Kms apart
- Channel Capacity = 44.7 Mbps
- Simulated Load = 1.0 & 1.2
- Average Message Size = 1, 4, 8 & 20 pkts / arrival
- Arrival Pattern : Poisson Arrival
- Load Type : Symmetric Load

Markers Used :

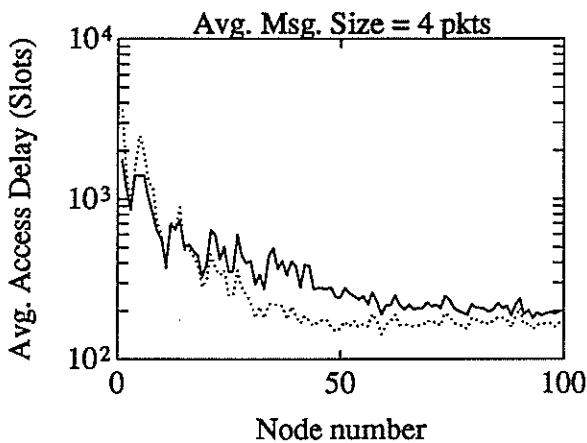
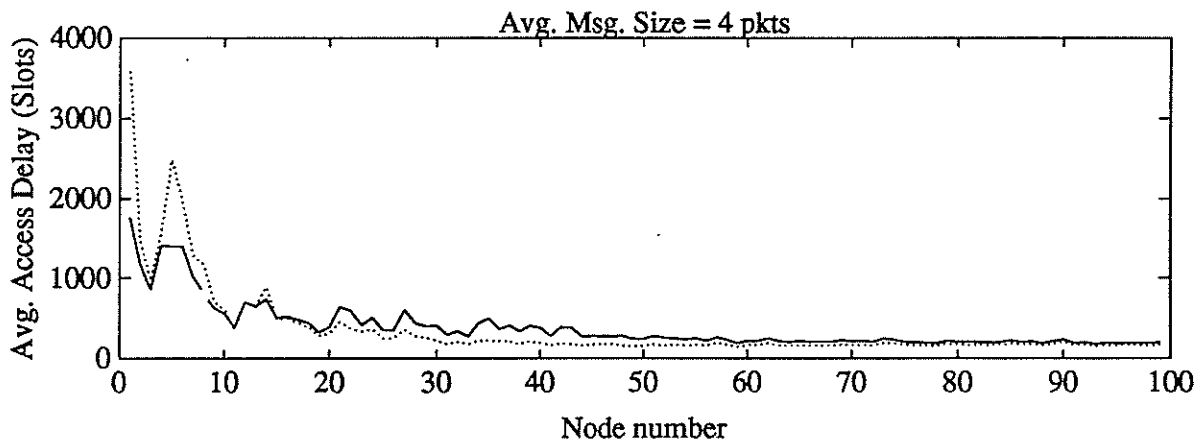
- _____ 802.6 Performance Characteristic
- Scheme-I Performance Characteristic

(The Characteristics are with respect to Bus-A access.)



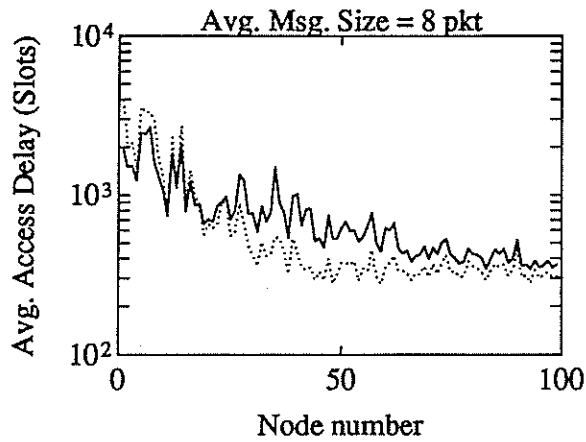
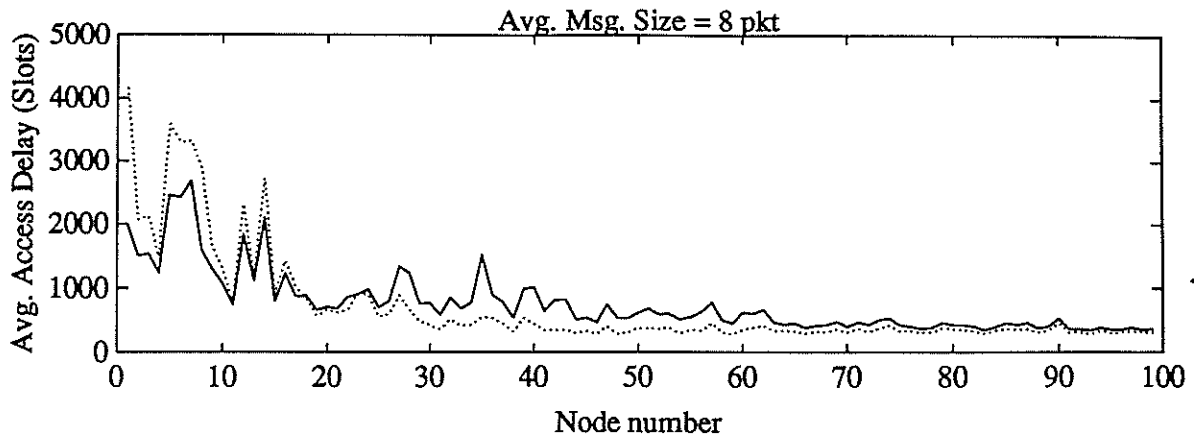
**Average Delay Characteristics
(Exact & Semilog Plots)**

- * Load Value = 1.0
- * 100 nodes @ 0.5 Kms apart
- * Poisson Arrival
- * Symmetric Load Type



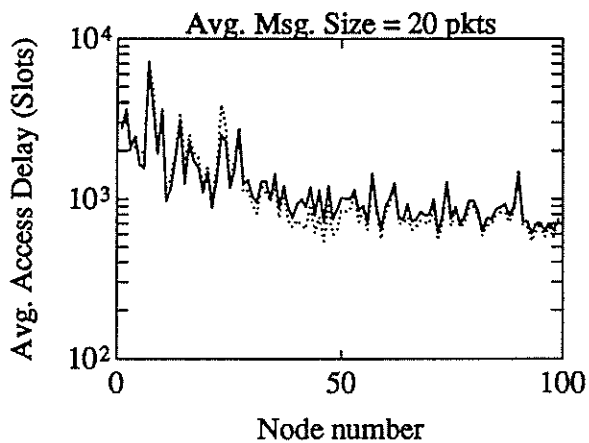
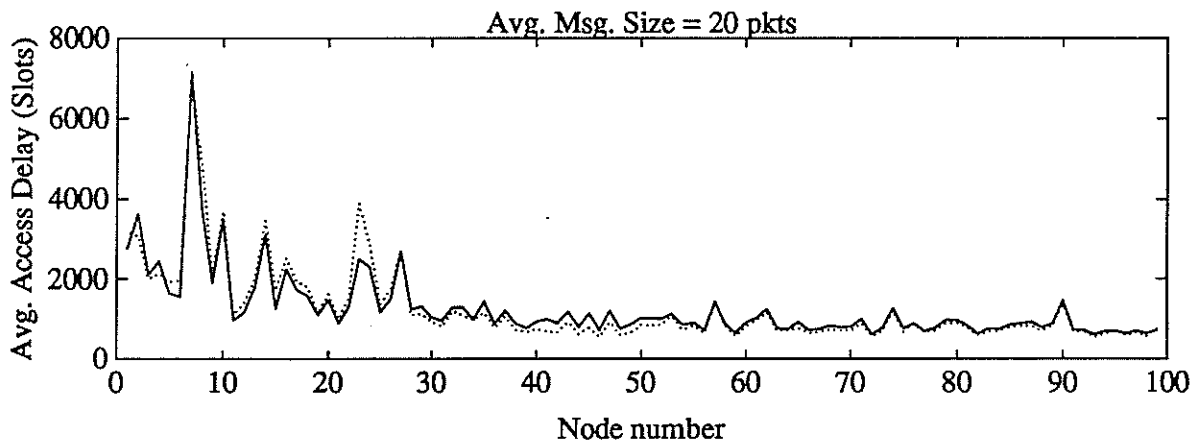
**Average Delay Characteristics
(Exact & Semilog Plots)**

- * Load Value = 1.0
- * 100 nodes @ 0.5 Kms apart
- * Poisson Arrival
- * Symmetric Load Type



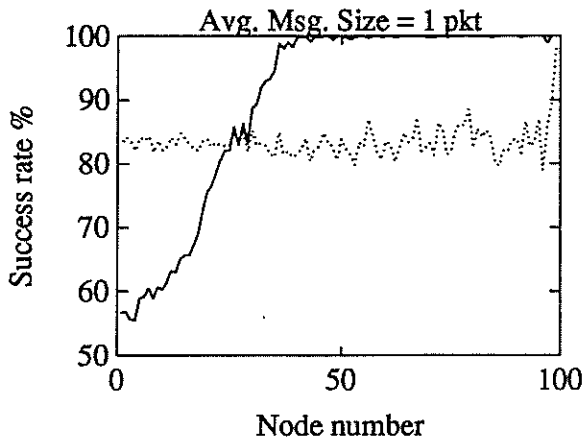
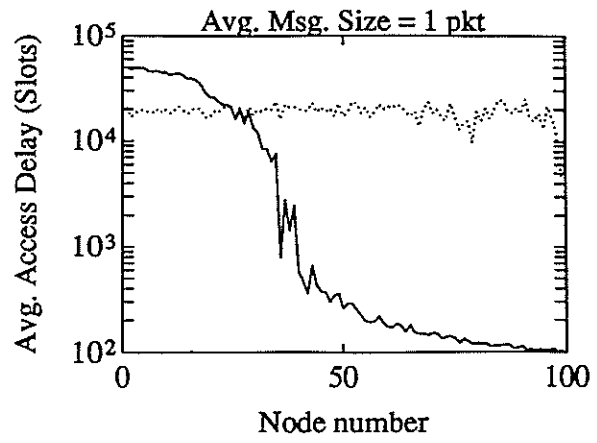
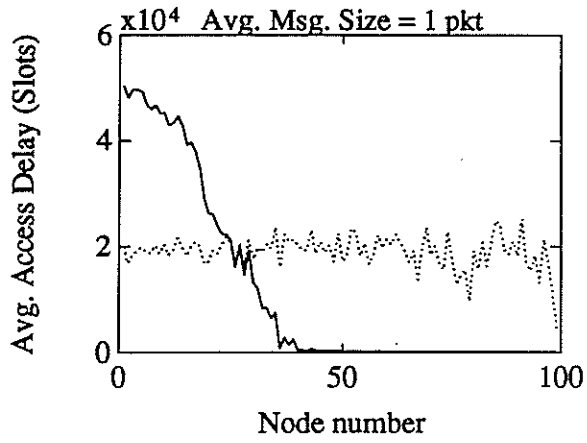
**Average Delay Characteristics
(Exact & Semilog Plots)**

- * Load Value = 1.0
- * 100 nodes @ 0.5 Kms apart
- * Poisson Arrival
- * Symmetric Load Type



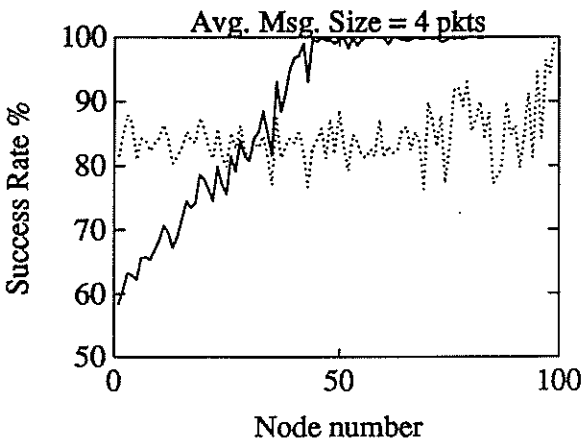
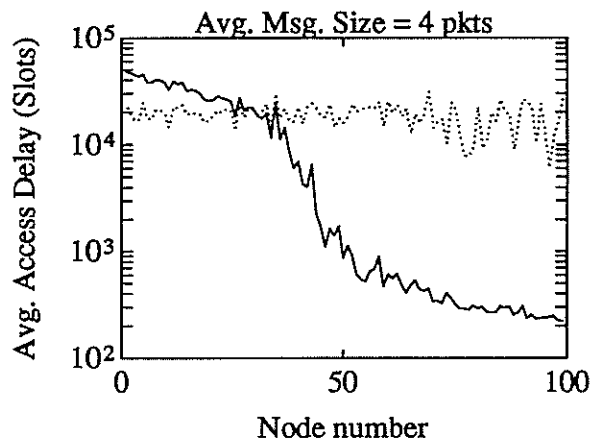
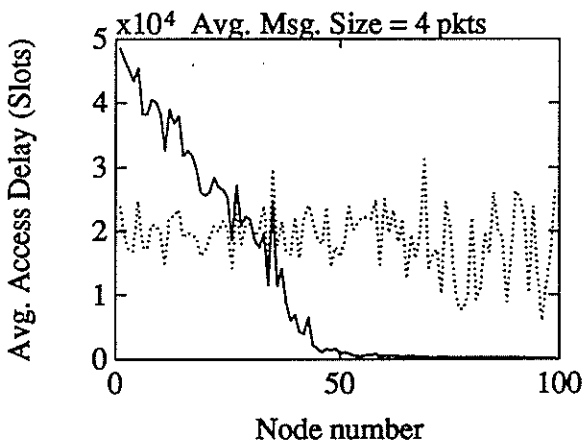
**Average Delay Characteristics
(Exact & Semilog Plots)**

- * Load Value = 1.0
- * 100 nodes @ 0.5 Kms apart
- * Poisson Arrival
- * Symmetric Load Type



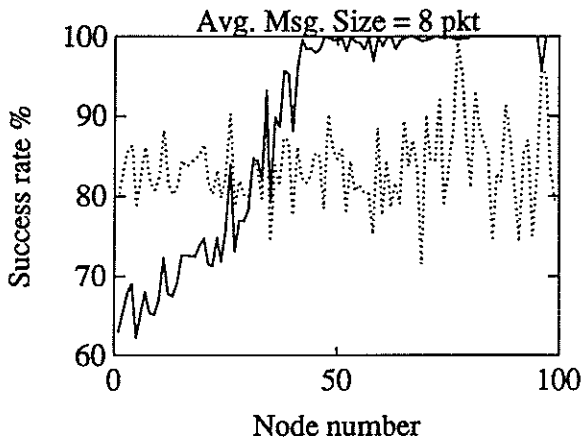
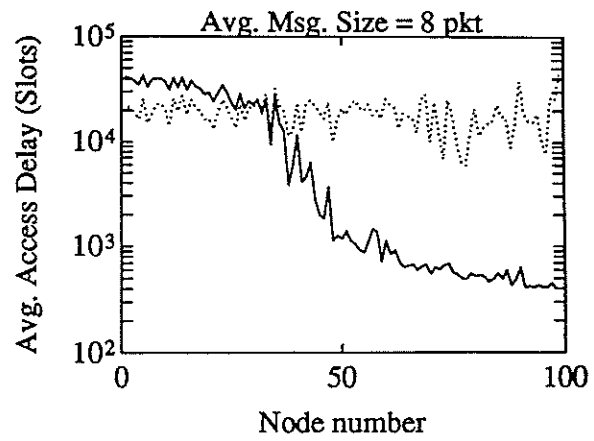
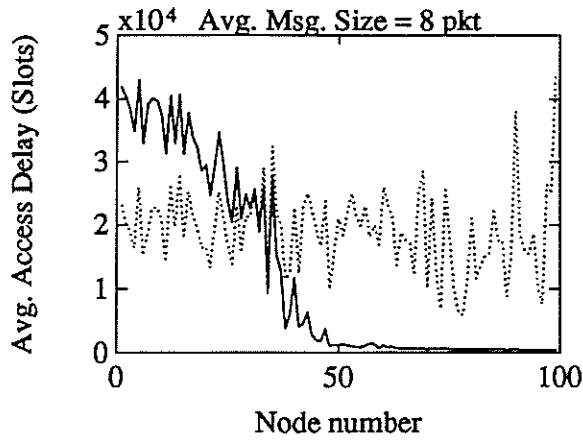
**Average Delay (Exact & Semilog),
Success Rate Characteristics**

- * Load Value = 1.2
- * 100 nodes @ 0.5 Kms apart
- * Poisson Arrival
- * Symmetric Load Type



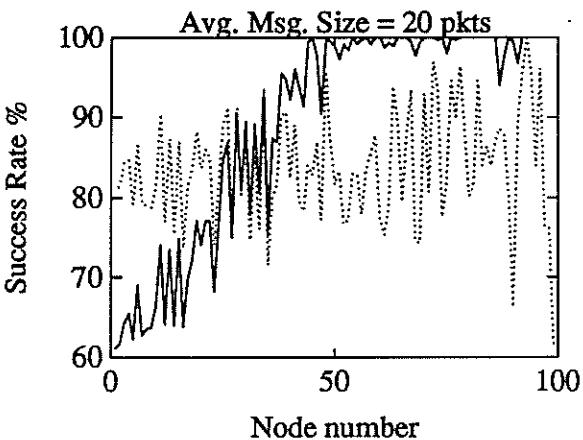
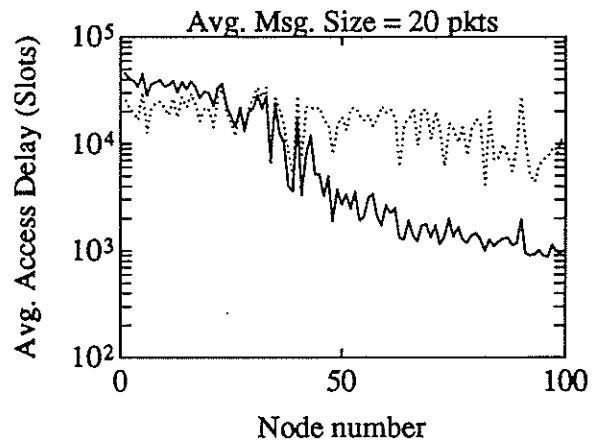
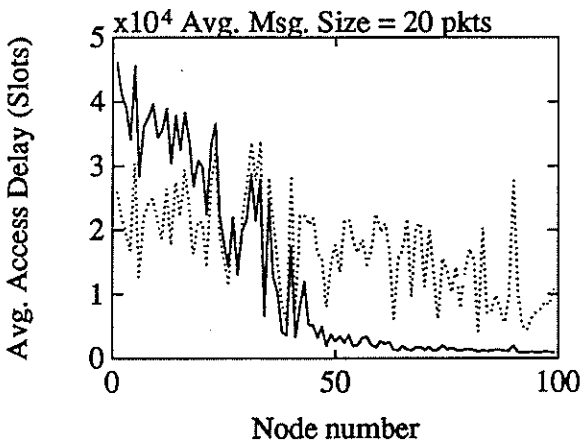
**Average Delay (Exact & Semilog),
Success Rate Characteristics**

- * Load Value = 1.2
- * 100 nodes @ 0.5 Kms apart
- * Poisson Arrival
- * Symmetric Load Type



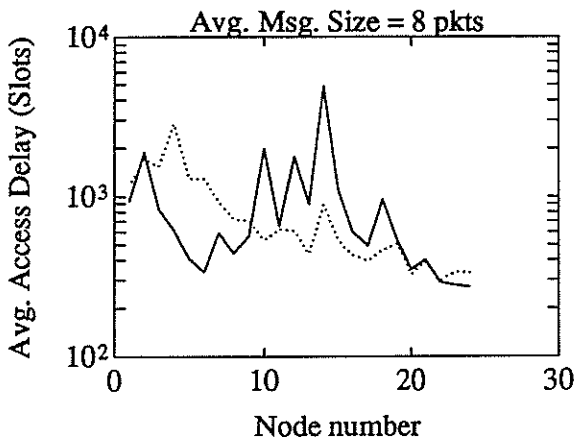
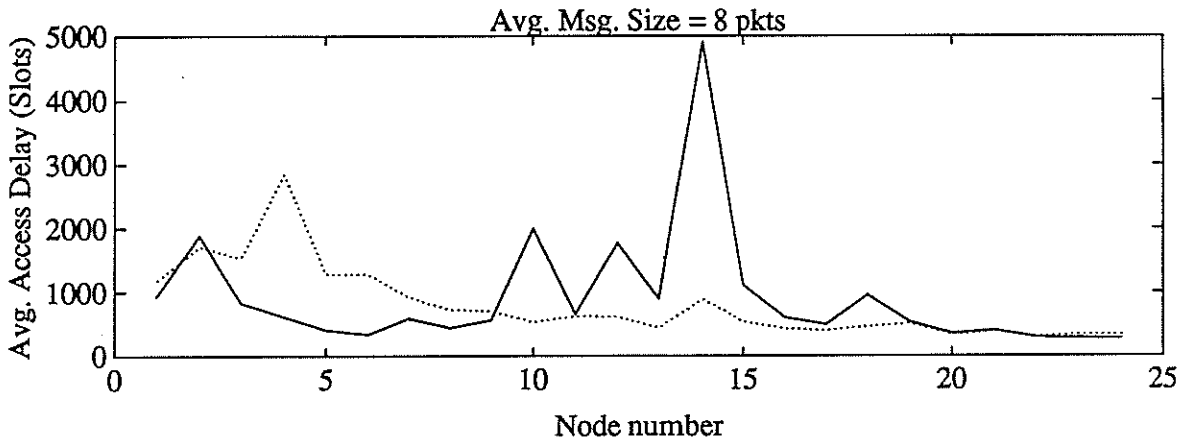
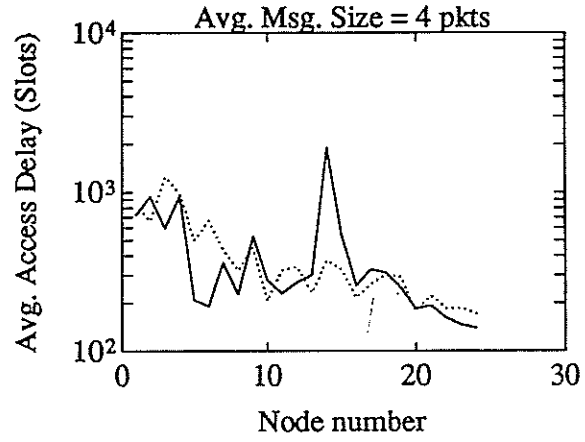
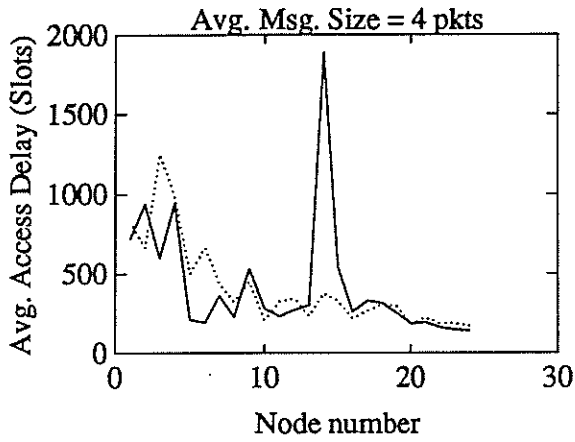
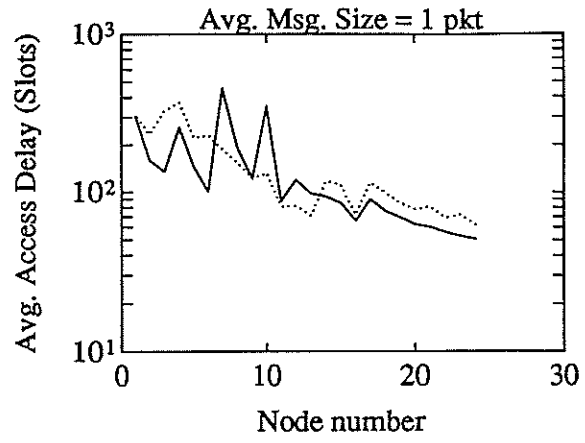
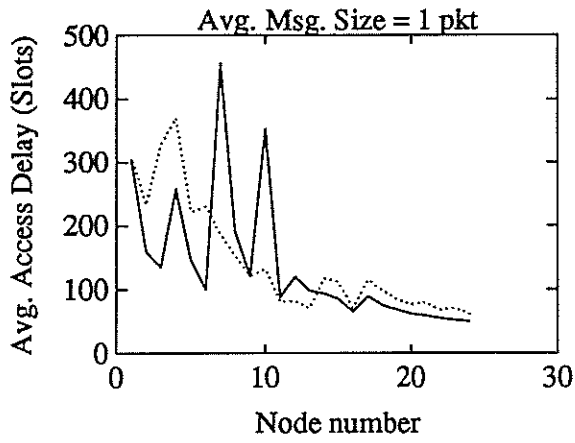
**Average Delay (Exact & Semilog),
Success Rate Characteristics**

- * Load Value = 1.2
- * 100 nodes @ 0.5 Kms apart
- * Poisson Arrival
- * Symmetric Load Type



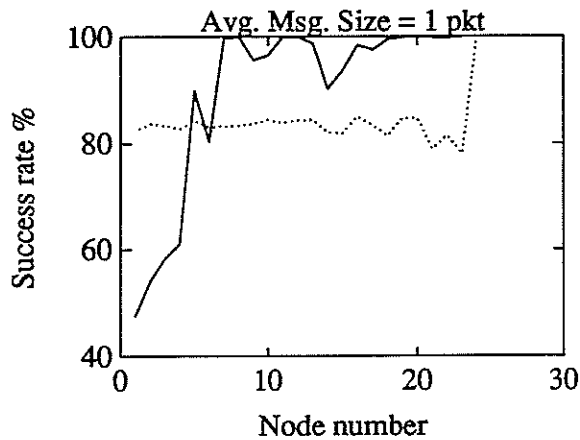
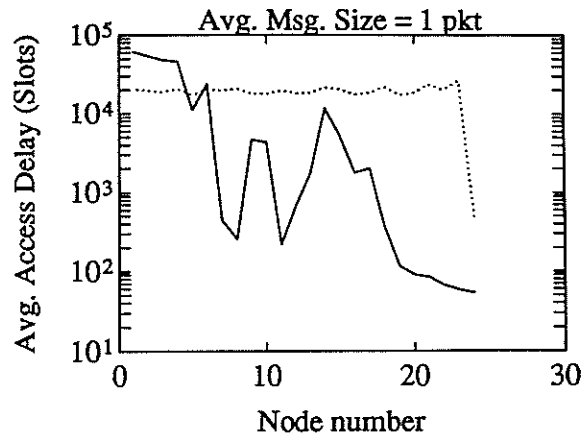
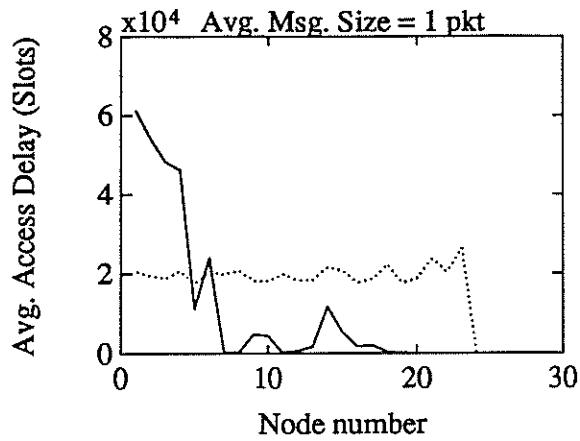
**Average Delay (Exact & Semilog),
Success Rate Characteristics**

- * Load Value = 1.2
- * 100 nodes @ 0.5 Kms apart
- * Poisson Arrival
- * Symmetric Load Type



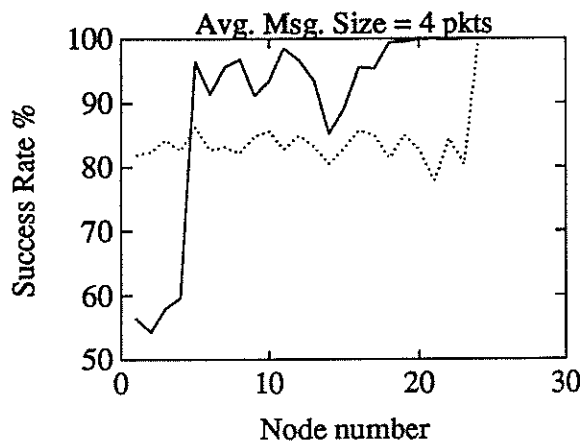
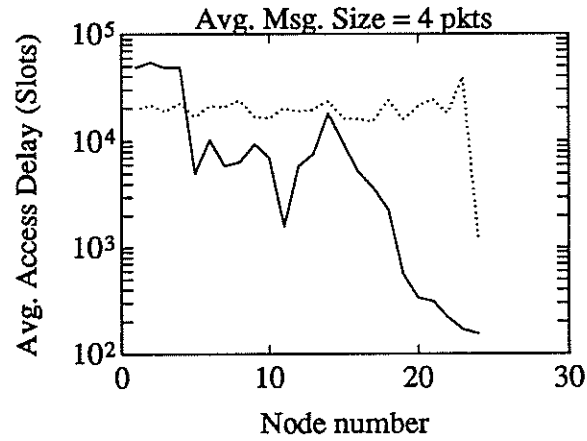
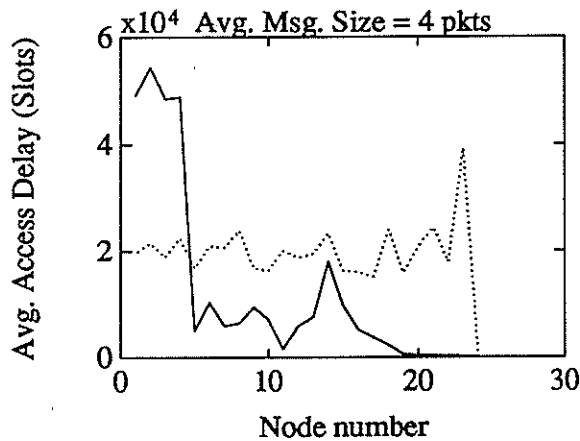
**Average Delay Characteristics
(Exact & Semilog Plots)**

- * Load Value = 1.0
- * 25 nodes @ 2 Kms apart
- * Poisson Arrival
- * Symmetric Load Type



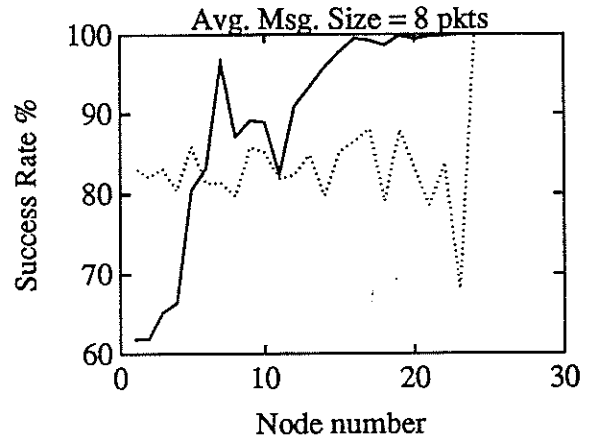
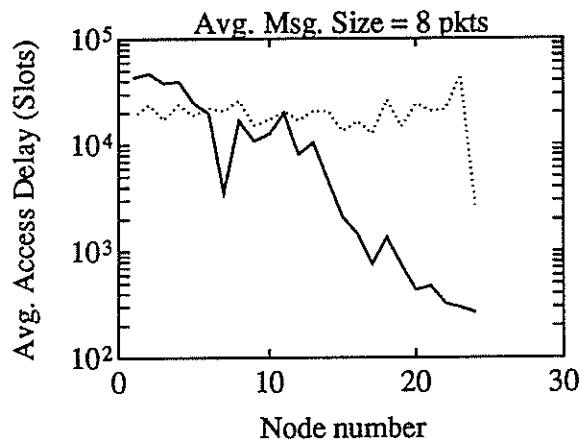
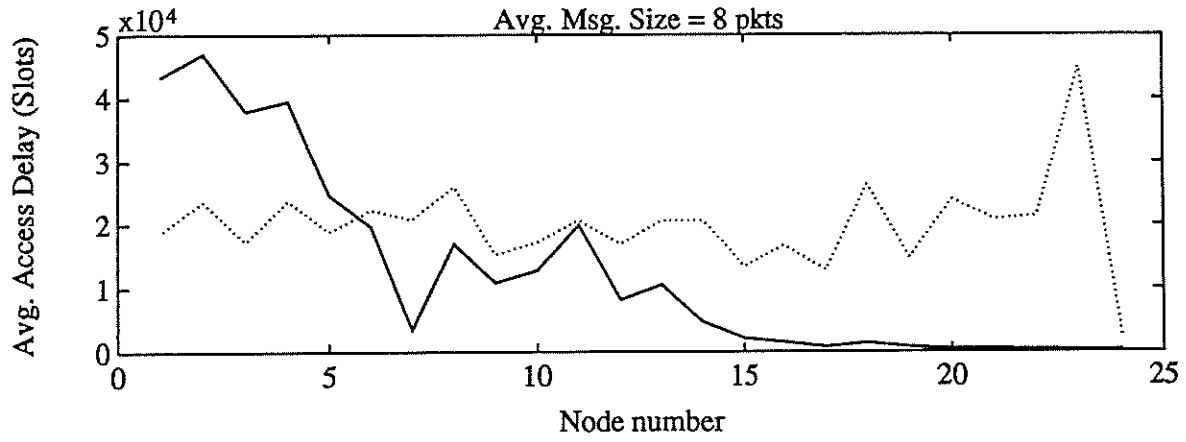
**Average Delay (Exact & Semilog),
Success Rate Characteristics**

- * Load Value = 1.2
- * 25 nodes @ 2 Kms apart
- * Poisson Arrival
- * Symmetric Load Type



**Average Delay (Exact & Semilog),
Success Rate Characteristics**

- * Load Value = 1.2
- * 25 nodes @ 2 Kms apart
- * Poisson Arrival
- * Symmetric Load Type



Average Delay (Exact & Semilog), Success Rate Characteristics

- * Load Value = 1.2
- * 25 nodes @ 2 Kms apart
- * Poisson Arrival
- * Symmetric Load Type