72-83

Protocol Architecture of a Tree Network with Collision Avoidance Switches[1]

Tatsuya Suda
Department of Computer Science
Columbia University

24 September 1983

---

[1] submitted to the International Conference on Communications

Abstract

In this paper we propose a tree network with collision avoidance switches and discuss its protocol architecture and performance. In the tree network, collision avoidance switches are connected by full duplex transmission lines and form a rooted tree topology with stations at its leaves. A station transmits a packet whenever it has a new one. The collision avoidance switch allows packets to go through when it is idle, and blocks them when busy; thus collisions caused by simultaneous transmissions of packets are avoided. Blocked packets are retransmitted by the sender, while unblocked packets, after having climbed up the tree to an appropriate height, are broadcast down to a subset of stations (local broadcast). Packets are broadcast locally in order to maximize the number of concurrent transmissions. In addition, collisions among packets are completely avoided, hence, the network is highly efficient in performance. Performance analysis for simple case is shown, and the mean packet transmission delay and throughput are obtained.

Table of Contents

List of Figures

1 Introduction

Recent interests in the human interface to computer systems has spurred the development of integrated service workstations which facilitate manipulation of various kinds of information such as data, voice, text, graphics and facsimile, and the communication systems which will carry these various kinds of information. Thus the future communication systems must be able to integrate information having various requirements and constraints and support bandwidth ranging from slow data rates to a rate of a few Mbits/sec of teleconferencing application or computer graphic application. Communication systems will have to offer not only secure delivery, which is required typically in data communications, but also real time delivery for certain types of information such as voice. Synchronized delivery of multiple information streams will also be required. (For instance, in teleconferencing application, pictures or graphs might be required to be delivered synchronously with associated aural explanations.)

In order to efficiently meet the above requirements, we propose and develop in this paper a tree network with collision avoidance switches and discuss its system architecture, transmission protocols and performance. In the network, collision avoidance switches are connected by full duplex transmission lines and form a rooted tree topology with stations at its leaves. Like random access protocols such as ALOHA, a station transmits a packet whenever it has a new one. The collision avoidance switch lets packets go through when it is idle, namely, if no packet is currently passing through the switch, and blocks packets when busy, and thus avoids collisions caused by simultaneous transmissions of packets. (The reference [1] discusses implementation issues of a collision avoidance switch similar to the one assumed in this paper.) Blocked packets are retransmitted by the sender, while unblocked packets, after having climbed up the tree to the switch at an appropriate height, are broadcast down to a subset of stations (local broadcast).

In this network, there are no collisions among packets. Even if more than two stations issue requirement for the network capacity at the same time (i.e., attempt to send a packet), at least one of them is guaranteed to use the capacity. Furthermore, packets are broadcast locally, and thus multiple concurrent transmissions are possible. Communication among a subset of stations might scarcely disturb those among others. The station protocol in the tree network is quite simple: a station simply sends a packet when it has one. If the collision is detected, a station immediately retransmits a packet.

In section 2, we describe the configuration of the network. Sections 3 describes architecture of the network transmission protocol. The switch function is also discussed in this section. A performance analysis of the network is shown for a simple case, and the mean transmission delay and throughput are obtained in section 4. Numerical results are shown in section 4.

2 Network Configuration

In the tree network, collision avoidance switches are connected by full duplex transmission lines and form a rooted tree topology (see the figure 1). In the figure, each tree leaf represents a station, that is, each vertex with no lower vertex corresponds to a station. For ease of explanation, we define the following terms and notations:

- degree of a switch ; the number of lower switches and stations adjacent to a switch
- level of a switch ; the number of switches (including itself) on the longest path among all the paths from a switch to stations below
- S_j^i ; a switch j at the level i (an element of $\{S^i\}$)
- son of S_j^i ; a switch at the level i-1 adjacent to S_j^i
- father of S_j^i ; a switch at the level i+1 adjacent to S_j^i
- $\{U(S_j^i)\}$; all stations below a switch S_j^i
- ancestor and descendant ; if there is a path from S_j^i to a station k, S_j^i is an ancestor of a station k, and a station k is a descendant of S_j^i.
- proper ancestor of a packet (or a subset of stations) ; the lowest level ancestor common to both origin and destination(s) of a packet (or the lowest level ancestor common to a subset of stations)
- S_j^i subtree ; a subtree consisting of S_j^i and all of its lower switches (S_j^i is called the root of that subtree.)
- proper subtree ; a subtree whose root is the proper ancestor

As an example, in the figure 1, for instance, S_1^3 is the root of the tree, and S_2^2, which has two adjacent switches (degree 2), is a switch at the level 2 with a father S_1^3 and sons S_3^1, S_4^1. $\{U(S_2^2)\}$ consists of stations 6, 7, 8, 9, and 10. The proper ancestor of users 6 and 9 is the switch S_2^2. The tree level is 3. In the following we assume that the level of the tree, i.e., the level of the switch at the root, is H. Users and switches at the same level are numbered from left to right. Please note that length of a line between switches, consequently propagation delay, may vary line by line.

Furthermore, each edge is a full duplex transmission line, namely, it has two logical links, one is for transmitting packets upwards (uplink) and the other is for transmitting packets downwards (downlink). Uplink and downlink are assumed to have the same channel capacity.

3 Transmission Protocol Architecture

In this section we discuss a transmission protocol and switch architecture for the tree network. In this network, when a station A wants to communicate with another station B (or a subset of stations), it first transmits its packets up to the proper ancestor of stations A and B. Then, the packet is broadcast down to the destination station(s) by the proper ancestor through the proper downlink subtree network.

We define the contention period of a packet as the propagation delay time from the original sending station to the father switch of its destination through the proper ancestor. (If a packet has multiple destination, the contention period is the largest propagation delay to the fathers of its destinations.) The reason for this definition will become evident in the following subsection. The transmission time of a packet is assumed to be greater than twice the propagation delay along the path from the proper ancestor to the father switch of its destination. For instance, in the figure 1, the transmission time of a packet from the station 1 to 3 is assumed to be greater than twice the propagation delay along the path S_1^2 - S_1^1; its contention time is the propagation delay along the path 1 - S_1^2 - S_1^1.

The protocols for switches and stations are described in detail below.

3.1 Switch Protocol

Each switch consists of two components, an uplink-component and a downlink-component. All uplinks are connected via an uplink-component and all downlinks are connected via a downlink-component (see the figure 2). An uplink component receives a packet from its sons and passes it to its father. It can also pass a packet to its peer downlink component, if required. A downlink component broadcasts a packet arriving from the father or from its peer uplink component to all its sons. A switch component is called busy, if a packet is currently being transmitted through it; otherwise it is called idle. In the following section we describe the protocol for each switch component.

3.1.1 Uplink Component Protocol

When an uplink component receives a packet from one of its sons, it first reads a control field of the packet, containing the address of the packet (i.e., the proper ancestor of the packet). If the switch is older than the proper ancestor of the packet, namely, if the level of the switch is greater than that of the proper ancestor indicated in the control field of the packet, the switch passes the packet to both its peer downlink component and its father switch.

Otherwise, the switch checks its own state, and if it is busy, simply discards the packet. If it is idle, it sets itself busy, and transmits the packet only to its father (in case where it is younger than the proper ancestor of the packet), or transmits the packet both to its father and to its peer downlink component (in case where it is the proper ancestor of the packet).

When the end of packet transmission is detected, it sets itself idle and waits for a new packet. The tree top switch follows the same protocol except that it cannot transmit a packet to its father since the tree top switch does not have a father. Details of this protocol is shown in the figure 3.

3.1.2 Downlink Component Protocol

When a downlink component receives a packet from its father switch, it checks its own state and, if busy, simply discards the packet. If it is idle, it sets itself busy and broadcasts the packet to all of its sons.

When a downlink component receives a packet from its peer uplink component, it checks the proper ancestor of the packet. If the proper ancestor indicated is itself, it follows the same procedure as that in the above case. If it is older than the proper ancestor of the packet, it sets itself busy, namely it logically disconnects downlink lines to all the sons. Even if a packet is now being transmitted through the switch, it aborts the transmission. This aborted broadcast of the packet will eventually reach the origin station of that packet, and will tell it that its transmission is unsuccessful.

When the end of packet transmission is detected, it sets itself idle and wait for a new packet.

3.2 Station Protocol

Station protocol is quite simple. When a station has packets to transmit, it selects the packet in the head of the queue and attaches a control field containing the level of its proper ancestor. (We assume that every station knows the topology of the network.) A station transmits the packet and begins listening to the downlink from its father. If it is still receiving broadcast of its packet after twice a contention period has passed since the beginning of packet transmission, it knows it has acquired the network channel capacity and its packet transmission is successful. Otherwise, (i.e., if it receives either another station's packet or its own packet truncated in the middle, or if nothing is broadcast within this time interval, the station knows its transmission was unsuccessful. In this case, the station stops its transmission (,if it is still transmitting the packet,) and retransmits its packet immediately. Unlike stations in other protocols such as ALOHA and CSMA/CD, stations do not have to wait for retransmission rescheduling delay. A station retransmits its packet immediately after the packet was blocked. A station repeats this process until the packet is successfully transmitted (see the figure 5).

The following worst-case scenario explains why a station must wait for twice the contention period to be guaranteed its use of network capacity. Suppose that station 1 wants to communicate with station 3 in the figure 1, and further, the propagation delay of the path station 1 - S_1^2 - S_1^1 is τ_1 and that of the line station 2 - S_1^1 is τ_2. At time t_0 station 1 begins transmitting. At time $t_0 + \tau_1 - \tau_2 - \epsilon$, where ϵ is an infinitesimally small value, suppose station 2 also begins transmitting a packet to station 3. At time $t_0 + \tau_1 - \epsilon$, the packet from station 2 arrives at the switch S_1^1 and forces that switch to become busy. The packet from station 1 arrives at S_1^1 at time $t_0 + \tau_1$, but the switch has just become busy and, hence, the packet is blocked. Eventually the packet from station 2 will reach the switch S_1^2 and force the downlink component of the switch S_1^2 to abort current broadcast of the packet from station 1, and thus station 1 will hear its broadcast terminated at time $t_0 + 2\tau_1$. (Please note that the switch S_1^1 not only broadcasts the the packet to stations 2 and 3 but also transmits the packet to its father switch.) Thus, new arrivals at other stations in the network during a contention period may cause a blocking of current transmission, and a station cannot be sure whether it has seized the channel until twice the contention period passes. (Note that even if a station cannot seize the channel, another station can transmit a packet successfully.) However once the channel has been seized, no other stations can interfere with the transmission.

The above example also shows why a packet must be passed up to the tree top switch, even if its proper ancestor is younger than the tree top switch. In the previous example, if the switch S_1^1 did not pass the packet up to higher switches (and, hence the packet did not make the switch S_1^2 abort transmission), station 1 would not have known its transmission was blocked. By transmitting the packet up to the tree top, the whole tree is logically divided into subtrees. For instance, if station 6 is communicating with station 8 in the figure 1, the packet from station 6 sets the switches S_2^2 and S_1^3 busy, and as a result, the whole network is divided into 4 subtrees (S_3^1 subtree, S_4^1 subtree, S_2^1 subtree and S_1^2 subtree). Stations within a subtree can communicate with each other but they cannot communicate with stations in other subtrees. In other words, transmission in one subtree does not interfere transmissions in other subtrees, and thus a number of concurrent transmissions are possible.

4 Performance of the Tree Network - Broadcast Star Network

In this section we develop a performance analysis for the simple version of the tree network called a broadcast star network and obtain the mean packet transmission delay.

4.1 Assumptions

In the following we assume N number of stations, each with infinite buffer capacity for storing arriving packets. Packets are transmitted according to the First Come First Served (FCFS) discipline within a queue, that is, a packet is transmitted only after all the previously arrived packets are transmitted successfully. New packets are assumed to arrive at station i according to the Poisson distribution with the rate λ_i. The transmission time distribution of packets (i.e., packet length) arriving at station i obeys the general distribution with the distribution function $S_i(x)$, the mean m_i^1 and the second moment m_i^2.

4.2 Performance of a Broadcast Star Network

Because of the difficulties in an exact analysis of a broadcast network with the general tree topology, we develop an analytic model for the simplest case called a broadcast star network (see the figure 7). In this network, all stations are directly connected to the tree top switch, resulting in the star topology (see the figure 6). In addition to the ease of analysis, the switch function is quite simple in this network. If the idle uplink component of the switch receives a packet, it sets itself busy and passes the packet to its peer downlink component. If the uplink component is busy, it simply discards the packet. When an downlink component receives a packet (from its peer uplink component), it broadcasts the packet to all stations. When the end of transmission is detected, the uplink component is set to be idle.

Propagation delay between a station and the switch takes an important role in this network. The station with the shortest propagation delay is given priority, because it detects the end of broadcast earlier than any other stations do, and thus sends its packet before the other stations. This station will thus acquire the idle switch and remaining stations will be blocked at the switch. For our analysis, however, we assume that propagation delays of all lines are equal to constant R (unit times). Note that because there can be at most one transmission at a time in this network, the mean transmission delay obtained below gives the worst case estimation of that in the general tree network.

4.2.1 System Characteristics

In this section we examine behavior of the whole system and obtain the mean waiting time, the mean queue length and the mean busy period length of the system.

So far as the mean values of system statistics are concerned, the whole network system can be considered as an M/G/1 queueing system as stated below.

Since new packets arrive at station i according to the Poisson distribution with the rate λ_i ($i = 1, \ldots, N$), aggregated arrivals to all the stations obey the Poisson process with the rate λ, where

$$\lambda = \sum_{i=1}^{N} \lambda_i \tag{1}$$

A successfully transmitted packet exclusively occupies the network channel capacity during the time interval equal to sum of its transmission time and the propagation delay R. In the example shown in the figure 8, station i detects the end of broadcast and transmits his packet at time t_1. Assuming that his transmission is successful, he begins to receive broadcast of his packet at time t_2 and ends receiving it at time t_4. During the time interval $t_4 - t_1$, any other packets are not accepted into the network, that is, other packets are blocked at the switch. Therefore, this time interval can be considered as the service time of the packet transmitted successfully. Namely, the service time Y_i of a packet from station i becomes

$$Y_i = X_i + R \tag{2}$$

where X_i represents the transmission time of a packet from station i. Because the probability of a packet to be derived from station i is λ_i/λ, the packet service time Y of the system becomes

$$Y = \sum_{i=1}^{N} (\lambda_i/\lambda) Y_i \, . \tag{3}$$

Considering that station i has its own packet transmission time distribution $S_i(x)$ and that the propagation delay is assumed to be constant, the aggregated service time distribution H(x) of the system is given by the following equation.

$$H(x) = \sum_{i=1}^{N} (\lambda_i/\lambda) S_i(x\text{-}R) \, . \tag{4}$$

Because of collision avoidance capability of the switch, the system does not waste the channel capacity at all. Whenever the system has packets to transmit, there is exactly one successful transmission.

From the above discussion the whole network system can be considered as an M/G/1 queueing system with infinite waiting space and also with the arrival rate λ and service time distribution H(x). By using results for an M/G/1 queueing system [2, 3], we have the following expressions for D, L, B and H, where

- D ; the mean packet transmission delay, i.e., the mean time interval from arrival of a packet to the time instant when it is successfully received by the destination station

- L ; the mean number of packets in the system
- B ; the mean length of a busy period of the system
- F ; the mean number of packets transmitted successfully in a busy period

In other words, D (L) is the mean of D_i (L_i) (i = 1, ..., N), where D_i (L_i) is the mean packet transmission delay (mean queue length) at station i. B is the mean of the time interval when the system has packets to send. Note that the above values do not depend on the service discipline of the system. The expressions for the above are [2, 3]:

$$D = \frac{\lambda E[H(x)^2]}{2(1 - \rho)} + E[H(x)] \tag{5}$$

$$L = \lambda D \tag{6}$$

$$B = \frac{E[H(x)]}{1 - \rho} \tag{7}$$

$$F = \frac{1}{1 - \rho} \tag{8}$$

where, $E[H(x)]$ and $E[H(x)^2]$ are the first and second moments of the service time distribution $H(x)$, respectively, and ρ is the traffic intensity of the system. That is.

$$E[H(x)] = \sum_{i=1}^{N} (\lambda_i/\lambda)(m_i^1 + R) \tag{9}$$

$$E[H(x)^2] = \sum_{i=1}^{N} (\lambda_i/\lambda)(m_i^2 + 2Rm_i^1 + R^2) \tag{10}$$

$$\rho = \lambda E[H(x)] = \sum_{i=1}^{N} \lambda_i(m_i^1 + R) \tag{11}$$

In order for this system to have steady state, the traffic intensity ρ of the system must be less than 1. Hence, we have

$$\rho = \lambda E[H(x)] < 1 \tag{12}$$

Thus, throughput S of the system defined as the mean number of successfully transmitted packets per one packet transmission time becomes

$$S = \lambda \sum_{i=1}^{N} (\lambda_i/\lambda) E[S_i(x)] < \frac{\sum_{i=1}^{N} (\lambda_i/\lambda) E[S_i(x)]}{E[H(x)]} \tag{13}$$

where $E[S_i(x)]$ is the mean of the distribution $S_i(x)$ and $\sum_{i=1}^{N} (\lambda_i/\lambda) E[S_i(x)]$ is the mean of the packet transmission times.

4.2.2 Mean Waiting Time at Each Station

In this paragraph, we obtain the mean packet transmission delay D_i at station i. The figure 9 shows the queueing model for the network. Packets arriving at station i make their own queue and are served on FCFS basis within the queue. If there is more than one station with a packet to transmit, one of them is randomly chosen and the packet in the head of its queue is served (i.e., the packet is transmitted successfully). Packets from the other stations are blocked at the switch. A successfully transmitted packet from station i occupies the network capacity during the time interval equal to $X_i + R$, where X_i is the random variable representing the transmission time of the packet at station i (see the figure 8). The distribution $H_i(x)$ of this time interval, namely the service time distribution, becomes $S_i(x\text{-}R)$.

The approximate analysis for the above queueing system has been done in the references [4] and [5]. In the references, the following equation was assumed to hold.

$$\begin{aligned}&\text{Prob(the queue length at station } i = 0 \mid \text{the queue length at station } j = k)\\ &= \text{Prob(the queue length at station } i = 0) \qquad (i \neq j \text{ and } k \neq 0)\end{aligned} \tag{14}$$

That is, the probability of the i-th queue being empty is independent of the number of packets in the other queues. Under this assumption, the mean packet transmission delay at station i is given by

$$D_i = \frac{\sum_{k=1}^{N} \rho_k E[H_k(x)^R] + \rho M \sum_{k \neq i} \alpha_k E[H_k(x)]}{1 - \rho_i - \lambda_i M \sum_{k \neq i} \alpha_k E[H_k(x)]} + E[H_i(x)] \tag{15}$$

where

$$E[H_k(x)] = m_k^{\ 1} + R \tag{16}$$

$$E[H_k(x)^2] = m_k^{\ 2} + 2Rm_k^{\ 1} + R^2 \tag{17}$$

$$E[H_k(x)^R] = \frac{E[H_k(x)^2]}{2E[H_k(x)]} \tag{18}$$

$$\rho_k = \lambda_k E[H_k(x)] \tag{19}$$

$$\rho = \sum_{k=1}^{N} \rho_k \tag{20}$$

$$\alpha_k = \frac{\lambda_k}{\sum_{k \neq i} \lambda_k} \tag{21}$$

In the above equations, the notation $\sum_{k \neq i}$ indicates that elements are summed from k = 1 to N, except k=i. M, which represents the mean number of packets served between two successive class i packets, is expressed by [4, 5]

$$M = \frac{(1 - 1/N)P}{1 - (1 - 1/N)P} \tag{22}$$

P in the above equation is the probability that at least one of the stations except station i has packets to transmit, and is obtained by solving the following set of equations.

$$P = \rho - \rho_i + \rho(\lambda - \lambda_i)E[H_i(x)]Q/(N - Q) + (\rho - P)(\lambda - \lambda_i)E[H_i(x)^R]$$

$$Q = \rho_i + \rho\lambda_i \sum_{k=1}^{N} \alpha_k E[H_k(x)]M + (\rho - Q)\lambda_i \sum_{k=1}^{N} \alpha_k E[H_k(x)^2] / \{2 \sum_{k=1}^{N} \alpha_k E[H_k(x)]\} \tag{23}$$

4.3 Numerical Results for the Broadcast Star Network

As we stated in the subsection 4.1, Poisson arrival process to each user is assumed in all of the following numerical examples.

The figures 10, 11 and 12 are for the homogeneous star network. We assume that packet transmission time distributions $S_i(x)$ and arrival rates λ_i are equal for all i. We also assume that every user has the same propagation delay R.

For the homogeneous system, the equation (13) for the throughput becomes

$$S < \frac{m^1}{m^1 + R} = S_{max} \tag{24}$$

where m^1 is the mean of $S_i(x)$. From this, we know that

$$\lim_{R \to 0} S_{max} = 1 \tag{25}$$

$$\lim_{R \to \infty} S_{max} = 0 \tag{26}$$

The figure 10 shows S_{max} as a function of the round trip propagation delay R for various values of m^1.

In the figures 11 and 12, the mean packet transmission delay D_i at the user i is shown as a function of the traffic intensity ρ of the system. We assume exponential packet transmission times in the figure 11 and constant packet transmission times in the figure 12. The (mean) packet transmission time m^1 is 5 in both figures. The round trip propagation delay R and the number of users N are assumed to be 1 (unit time) and 10 (users), respectively, in these figures. The maximum throughput S_{max} is 5/6 for both cases.

The figure 13 shows the mean packet transmission delay of the non-homogeneous system as a function of the traffic intensity $\rho = \sum_{i=1}^{5} \rho_i$. Again, the propagation delay of each user is one. We assume one heavy traffic user (user 1) and four light traffic users (users 2, 3, 4, 5). At the heavy user, packets are assumed to arrive according to the Poisson distribution with the rate 0.007843 and have constant transmission times equal to 50. The traffic intensity ρ_1 at the heavy user is 0.4. Light users are homogeneous, that is, packets arriving at light users have constant transmission times equal to 5 and $\lambda_2 = \lambda_3 = \lambda_4 = \lambda_5$.

5 Conclusions

In this paper we have proposed a tree network with collision avoidance switches and have discussed both protocol and performance issues of the system. In this network a collision avoidance switch allows packets go through only if it is idle, so that collisions caused by simultaneous transmissions of packets are avoided. Although performance analysis for simple case is shown in this paper, the performance of the system with the general tree topology is still unknown. Developing analytic and/or simulation models will be necessary in order to examine the general system behavior, and this waits future research.

References

1. A. Albanese, "Star Network with Collision-Avoidance Circuits", Bell System Technical Journal, Vol.62, No.3, pp.631-638, March 1983.

2. M. Schwartz, "Computer Communication Network Design and Analysis", Prentice Hall, 1977.

3. L. Kleinrock, "Queueing Systems Vol.1 : Theory", John Wiley and Sons, 1975.

4. T. Suda, "Studies on Satellite Communication Systems", Ph.D thesis, Department of Applied Mathematics and Physics, Faculty of Engineering, Kyoto University, Kyoto, Japan, March 1982.

5. T. Suda, H. Miyahara and T. Hasegawa, "Approximate Analysis for a Queueing System with Probabilistic Switching", Trans. Inst. Electronics Commun. Engineers of Japan (IECEJ), Vol.J63-D, No.1, pp.1-8, Jan. 1980.

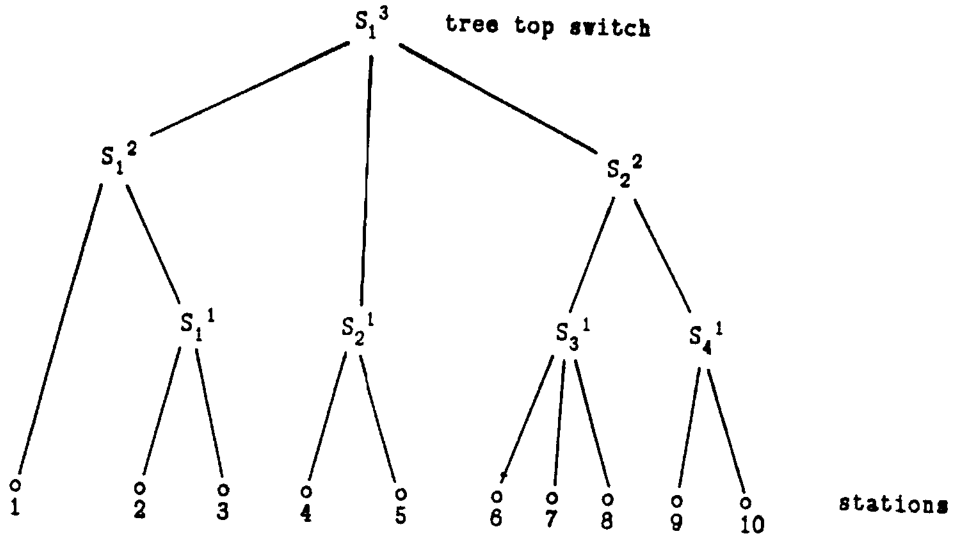


Figure 1: Topology of the Tree Network

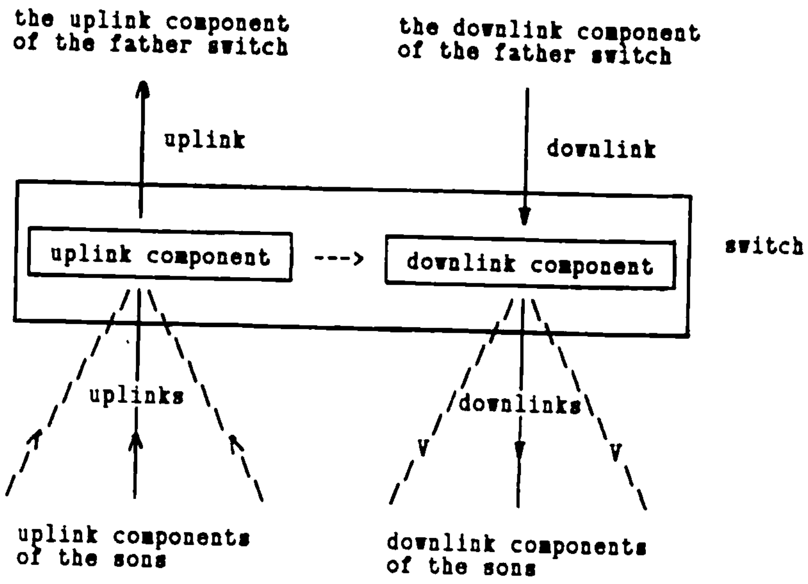


Figure 2: Switch Configuration

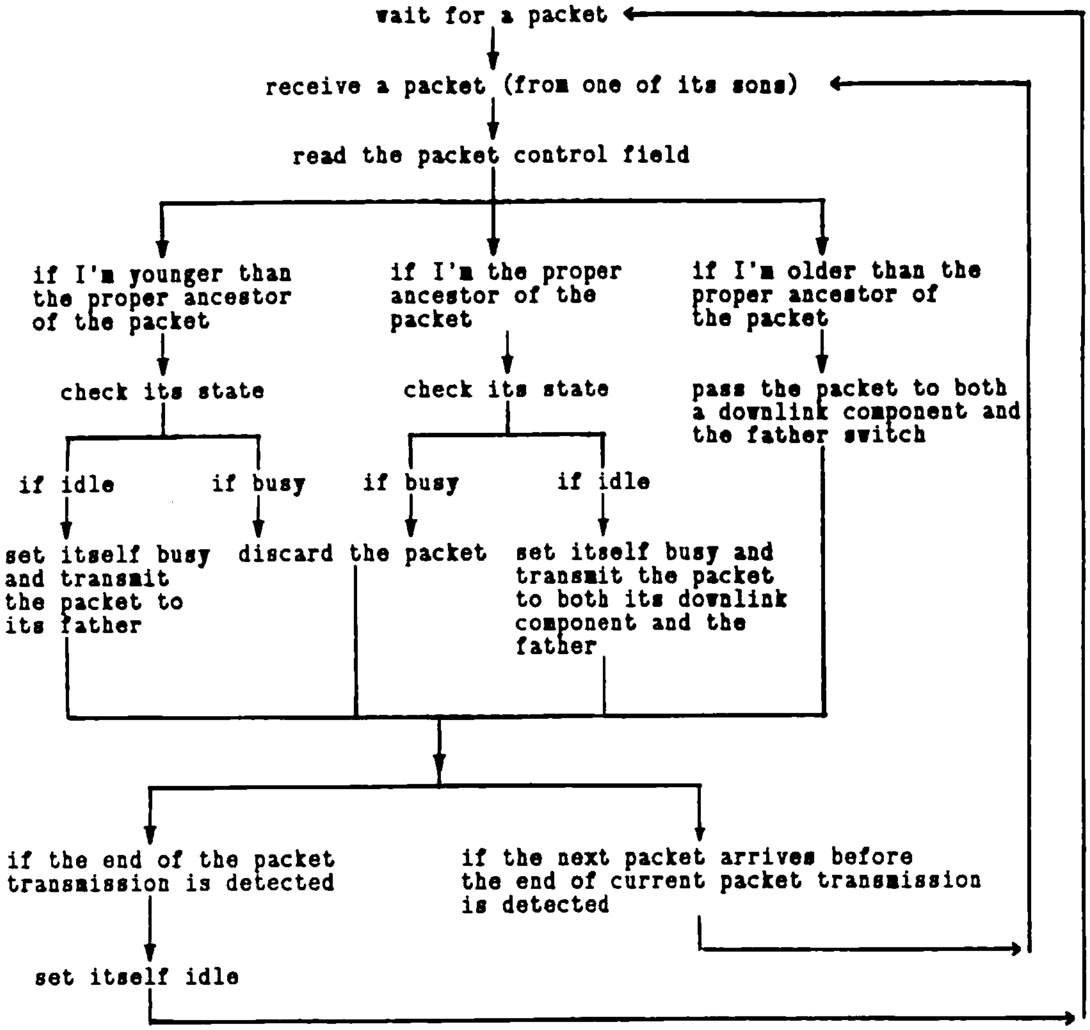


Figure 3: Uplink Switch Component Protocol

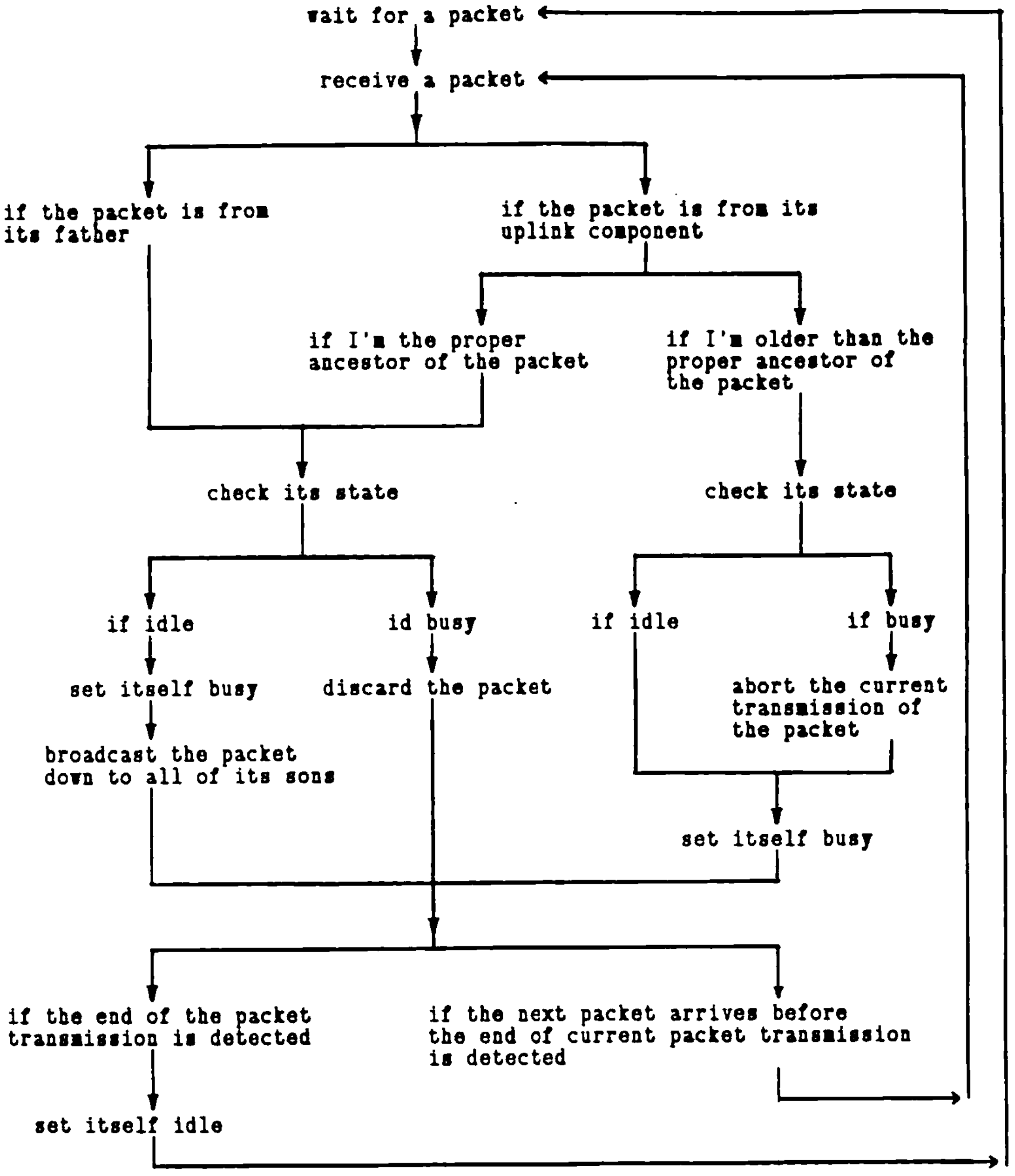


Figure 4: Downlink Switch Component Protocol

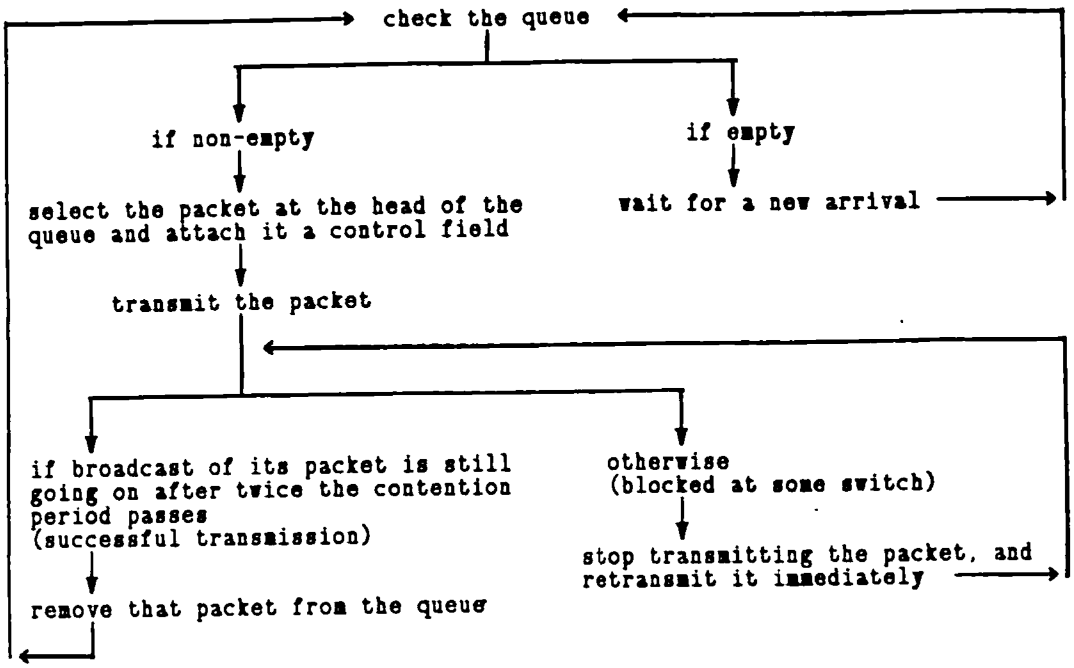


Figure 5: Station Transmission Protocol

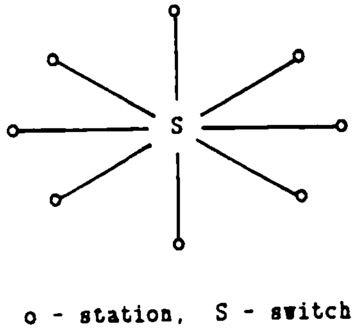


Figure 6: Broadcast Star Network

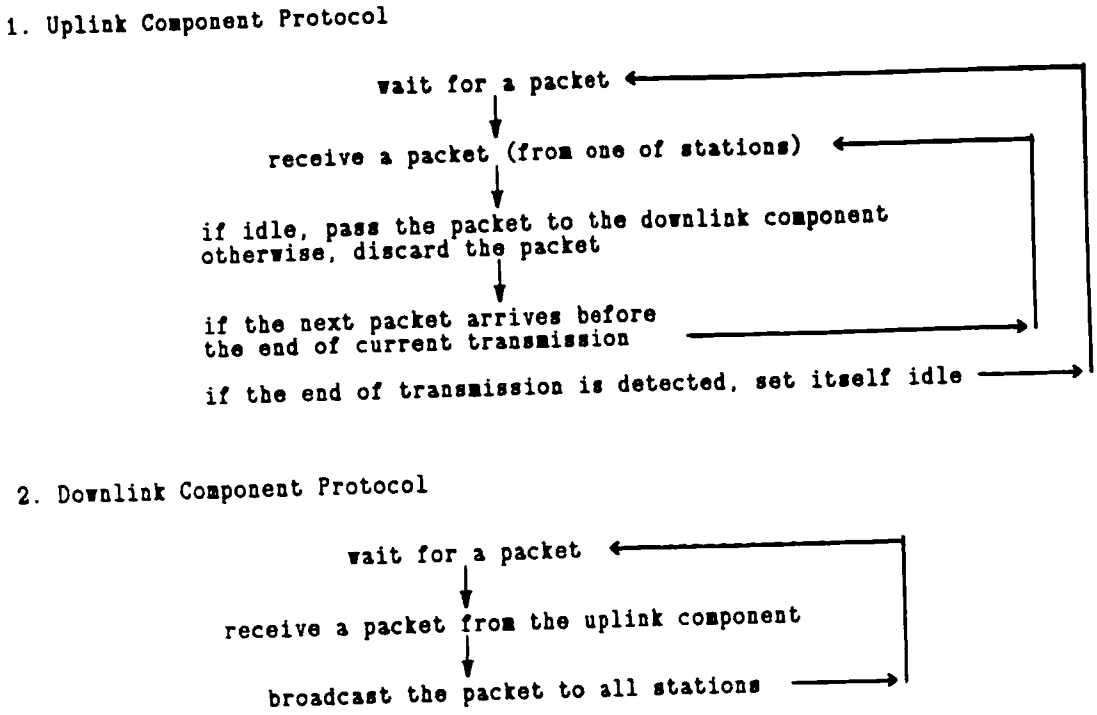
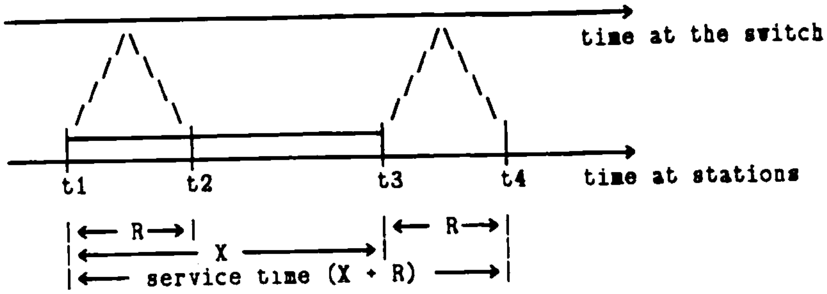


Figure 7: Switch Protocol - Broadcast Star Network



t_1 ; beginning of successful transmission of a packet from station i
$t_2 - t_1$; round trip propagation delay (R)
t_2 ; station i begins to receive broadcast of its packet at t_2
t_3 ; station i ends transmitting its packet at t_3
$t_3 - t_1$; packet transmission time (X)
$t_4 - t_3$; R
t_4 ; end of broadcast of a packet from station i, and beginning of a successful transmission of a packet form station j
$t_4 - t_1$; service time of a packet

Figure 8: Service Time in a Broadcast Star Network

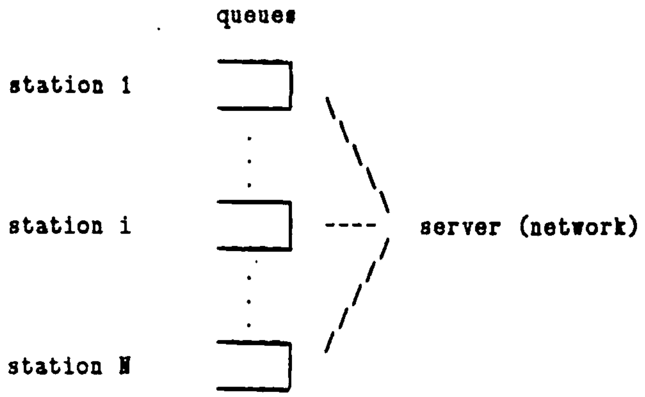


Figure 9: Queueing Model for the Broadcast Star Network

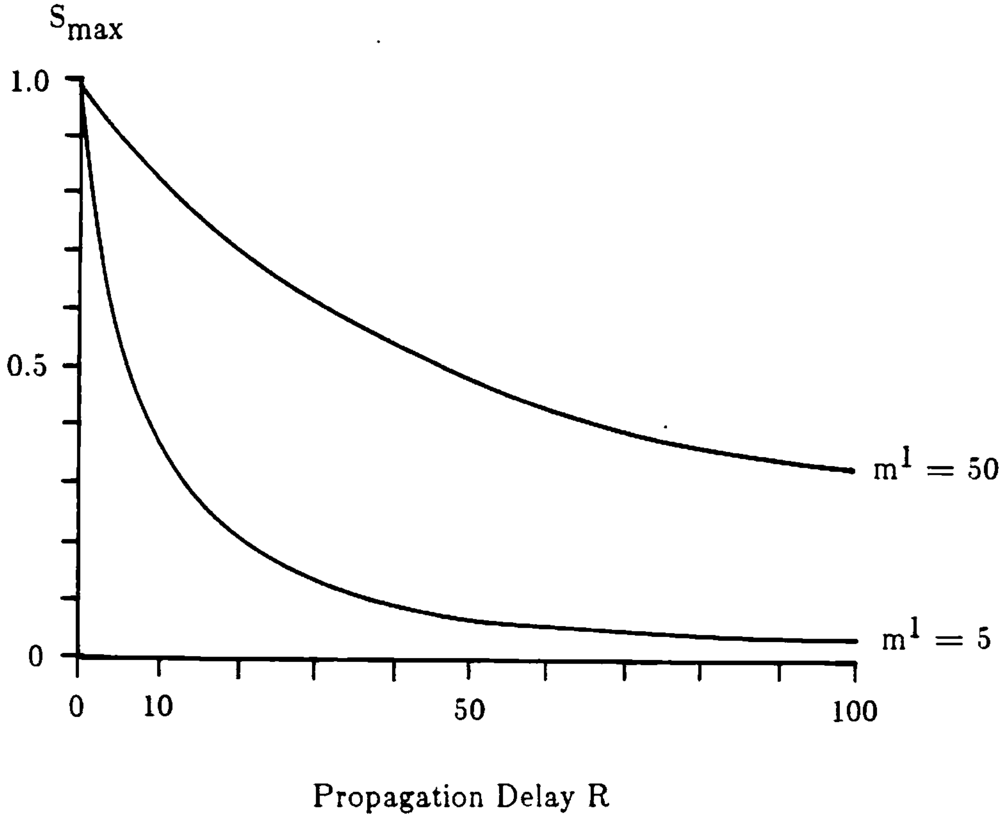


Figure 10: Maximum Throughput of the System

```
packet transmission time dis. = exp. dis. with the mean 5
N = 10
R =  1
```

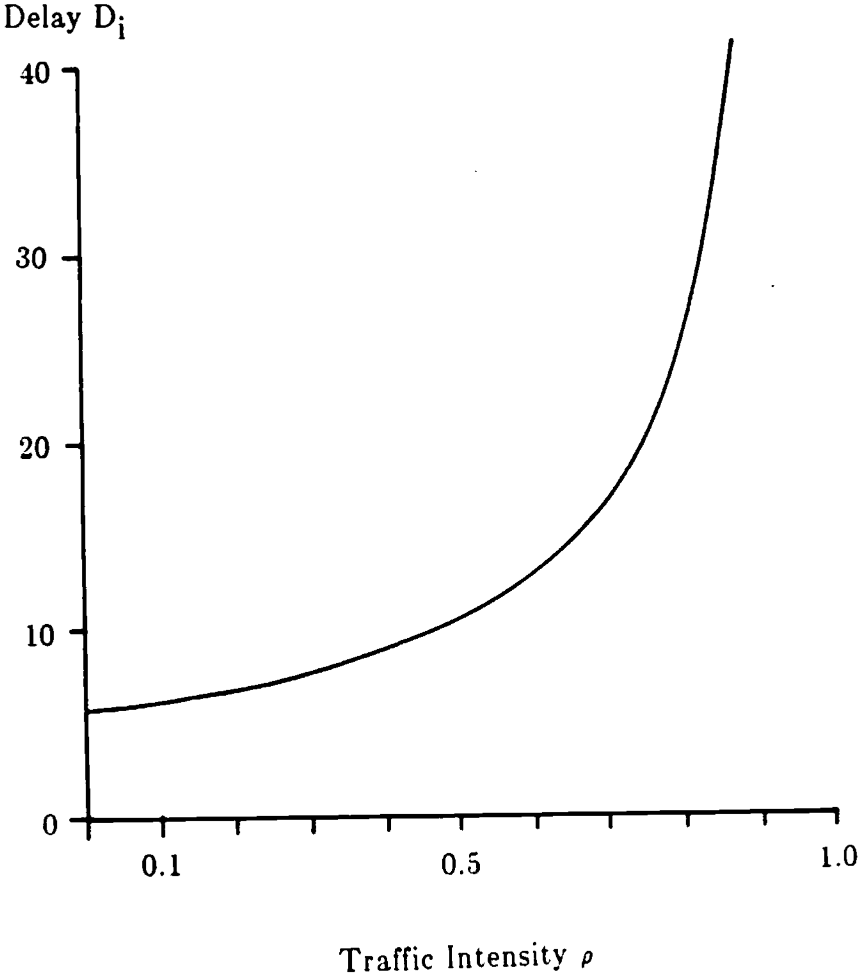


Figure 11: Mean Transmission Delay - Exponential Packet Transmission Time

```
packet transmission time dis. =  5 (constant)
N = 10
R =  1
```

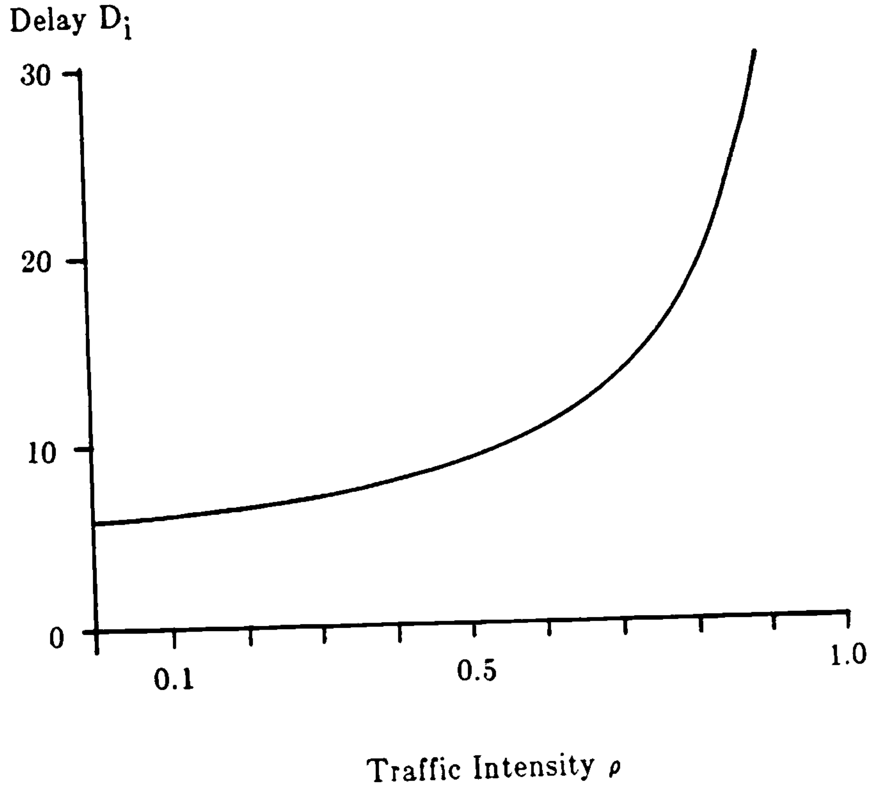


Figure 12: Mean Packet Transmission Delay - Constant Packet Transmission Time

```
one heavy station and 4 light stations
heavy station ; packet transmission time = 50
                traffic intensity = 0.4
light station ; packet transmission time = 5
R = 1
```

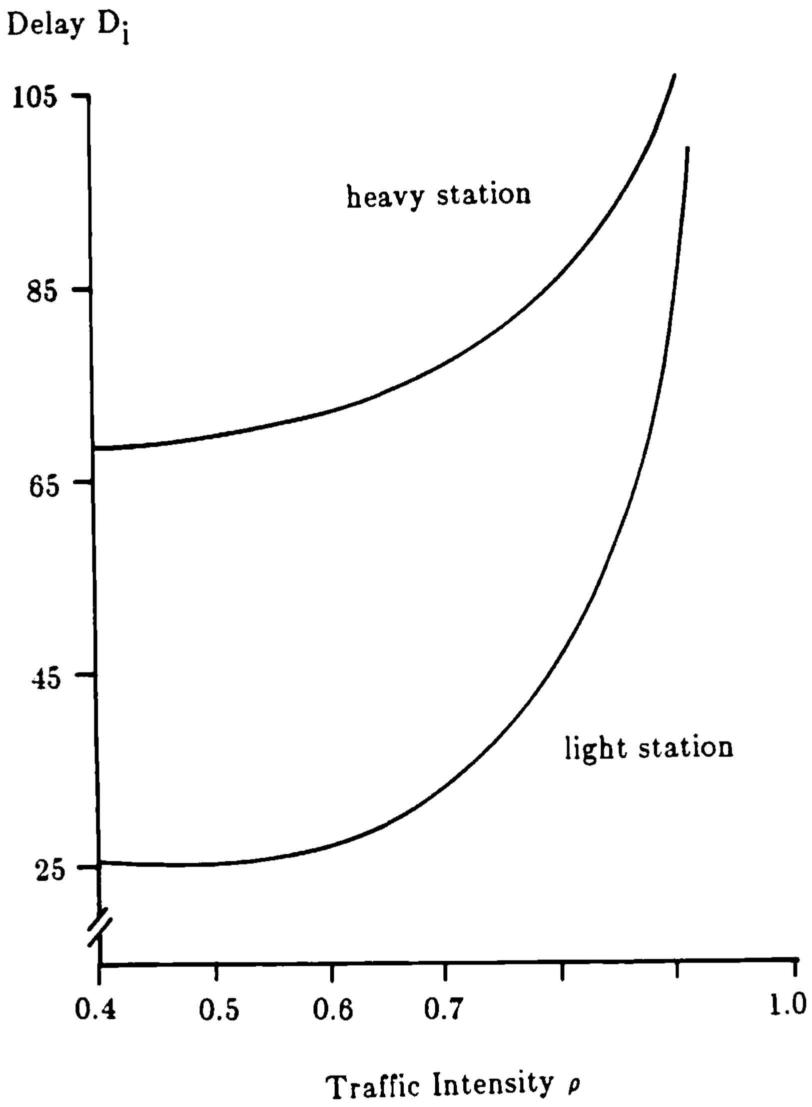


Figure 13: Mean Packet Transmission Delay