To appear in Queueing Systems: Theory and Applications

Queueing Systems 9 (1991) 113–132

Performance modeling and optimization of networks of bridged LANs *

Sanjay Gupta and Keith W. Ross

Department of Systems, University of Pennsylvania, Philadelphia, PA 19104, USA

Received 12 February 1991; revised 10 April 1991

An internetwork of LANs is modeled as a graph with LAN segments as edges and transparent bridges and repeaters as nodes. The graph model leads to a simple expression for the effective load on an arbitrary LAN segment, which takes into account the overhead traffic due to the learning mechanism of the transparent bridges. Simplifying assumptions for the operation of the MAC layer protocol lead to a simple expression for the average end-to-end delay in terms of the effective loads on the LAN segments.

The problem of optimally locating bridges and repeaters on the nodes in order to minimize the average delay is then studied. It is shown that this problem is equivalent to the set partitioning problem, which is NP-complete, but for which good algorithms exist to solve large problems. The related problem of minimizing cost subject to a constraint on average end-to-end delay is also discussed. Finally, the problem of locating bridges and repeaters on a linear topology, as typically found in an office building with a large number of floors, is studied. This special case gives rise to an $O(L^2)$ algorithm, where L is the number of floors.

Keywords: Bridges, LANs, data networks, network design, performance modeling.

1. Introduction

Local Area Network (LAN) installations have continued to grow at a rapid pace due to the increasing need for information processing equipment (personal computers, workstations, databases, printers, data gathering equipment, etc.) to be networked in the office and manufacturing environments. The physical layer implementation of the Medium Access Control (MAC) protocol restricts the physical extent of a LAN as well as the number of stations permitted to be attached to a LAN. For an Ethernet, which is based on the IEEE 802.3 CSMA/CD protocol, the maximum cable length permitted is 500 meters and the maximum number of stations on a cable is 100: see Tanenbaum [15] and Schwartz [14].

* Supported partially through AT&T grant 5-23690.

To increase both the physical extent and the maximum number of stations, LAN segments can be connected together with multiport repeaters. A repeater is a physical layer device that receives, and retransmits an incoming packet on every outgoing segment. In the case of multiple Ethernets connected by repeaters, the distance between any two stations should be less than 2500 meters and the number of stations should be less than 1024. Furthermore, it is required that no more than four repeaters be present between any two stations.

Bridges are MAC level store-and-forward devices that can be employed as alternatives to repeaters in order to connect multiple LANs. A transparent bridge automatically initializes and configures itself, and runs with no intervention from network management. It also performs the elementary functions of forwarding frames, learning the station locations, and building a spanning tree for the entire network: see Backes [1]. Transparent bridges have the following features:

- *Traffic filtering:* During normal operation a bridge will only forward a packet onto a segment if the destination station is in the direction of the segment. Filtering isolates traffic and potentially improves throughout and delay performance for the interconnected system.
- *Increased physical extent and maximum number of stations:* Bridges enable the interconnection of segments without the restrictions on number of stations and physical extent associated with repeaters.

However, like other store-and-forward devices, bridges add to the total delay of a packet, which can be significant at times (typically from about 200 μsec to several milliseconds: refer Benhamou [2]). This is particularly undesirable for packets which have to make more than a few hops as each bridge independently adds to the delay. Moreover, the performance of a bridge is typically limited by the rate at which it can examine the incoming packets and decide whether a packet needs to be buffered and subsequently forwarded. State-of-the-art bridges forward only a few thousand packets per second, while the arrival rate could be as much as 14,880 packets per second for a 10 Mbps Ethernet: see Boule et al. [13].

Networks consisting of Ethernet segments, repeaters, and bridges have typically been designed in an ad hoc manner. Bridges have been used mainly to increase the physical extent of the network to different floors and/or buildings. But with the ever increasing number of stations, and corresponding increase in load, there is a need to carefully engineer the networks so as to be able to extract the maximum of the resources at hand.

In this paper we develop a performance and optimization methodology for networks of interconnected LAN segments, repeaters, and transparent bridges. In order to simplify the presentation, throughout sections 2 through 6 we assume that each LAN segment is a CSMA/CD LAN; in section 7 we indicate how the methodology can be applied to interconnected token rings. In section 2 a graph theoretic model is developed which defines the interconnected system as a collection of *subnetworks*, where the nodes internal to a subnetwork consist of only repeaters. Assumptions are made concerning the MAC protocol for both the stations and the bridges. These assumptions are consistent with those normally made when analyzing the performance of LAN segments in isolation: see Lam [10], and Bertsekas and Gallager [3]. We then indicate how the *effective load on a segment* can be efficiently calculated even when overhead traffic must be taken into account.

The graph-theoretic model along with the MAC protocol assumptions enable us to analyze network throughput and end-to-end delay by separately studying the throughput and delay characteristics of each of the subnetworks. In section 3 a linear problem is given whose solution is the maximum throughput of the interconnected system. Also, for a system consisting of two segments, the increase in throughput obtained by replacing a repeater with a bridge is explicitly determined. In section 4 a formula is given for average end-to-end delay which is easy to evaluate.

The problem of optimally locating bridges on a tree topology is studied in section 5. In particular, we show that the problem of locating bridges on the network in order to minimize average end-to-end delay can be formulated as a set partitioning problem. This along with preprocessing techniques allows for the solution of networks with hundreds of potential bridge locations. The cost of the bridges can also be explicitly taken into account (either as constraints or as a term in the objective function) if the number of segments that are to be connected by a bridge is the same for each potential bridge location. We consider in section 6 the problem of minimizing delay in linear topologies, as typically found in office buildings with a large number of floors, and develop an efficient knapsack-like algorithm for this special case. Examples are given for both general and linear topologies.

2. A graph-theoretic model for interconnected Ethernets

Let $G = [X, U]$ be an undirected graph, where X is the set of nodes and U is the set of edges. Let $N := |U|$. We shall always assume that G is a tree, i.e., G has no cycles. Let X_t be the leaf nodes of G, i.e., the set of nodes in X which have a degree of 1. Let $\{X_b, X_r\}$ be a partition of $X - X_t$. We shall refer to the edges in U as *LAN segments* or simply as *segments*. Throughout sections 2 through 5 we assume that each LAN segment is a bus and employs the CSMA/CD protocol as the MAC sublayer. We shall refer to the nodes in X_b, X_r, X_t as *bridges, repeaters,* and *termination points*, respectively. An example of an interconnected system is given fig. 1. The squares represent bridges, and the circles represent repeaters and termination points.

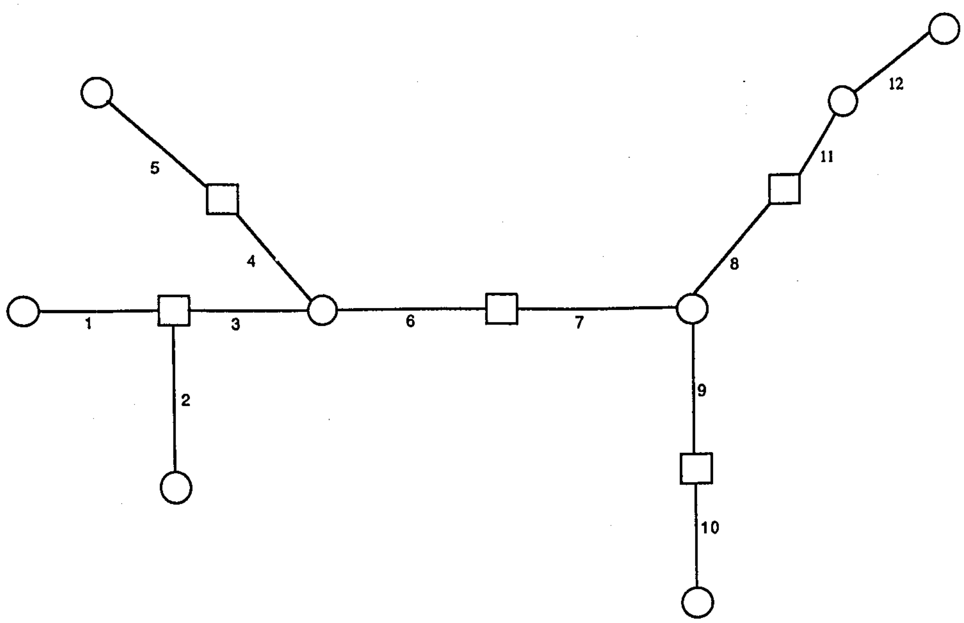


Fig. 1. Example of an interconnected network.

If we detach the edges from the nodes in X_b, then the network decomposes into, say, Q subtrees. Denote U_q, $q = 1, \dots, Q$, for the set of edges in the qth subtree. For example, the partition for the interconnected system in fig. 1 consists of the sets $U_1 = \{1\}$, $U_2 = \{2\}$, $U_3 = \{3, 4, 6\}$, $U_4 = \{5\}$, $U_5 = \{7, 8, 9\}$, $U_6 = \{10\}$, $U_7 = \{11, 12\}$. The partition can be found in $O(N)$ time with a depth-first algorithm. We shall refer to each set U_q as a *subnetwork*. Let $\pi(p, q)$ be the set of subnetworks in the path between subnetworks p and q (including p and q). For example, in the network of fig. 1 we have $\pi(3, 7) = \{3, 5, 7\}$. A standard depth-first algorithm (e.g., see Gondran and Minoux [8]) produces the sets $\pi(p, q)$, $1 \leqslant p, q \leqslant Q$, in $O(N^2)$ time.

Repeaters are physical layer relays which operate as follows: When a packet arrives from a segment, it is forwarded to all other segments connected to the repeater. Thus if a packet is transmitted on a segment of a subnetwork, then it is also transmitted on all of the other segments of the same subnetwork. The repeaters do not store packets, but simply copy bits as they arrive. We shall therefore assume that the delay introduced by a repeater to a traversing packet is negligible compared with the transmission time of a packet. An example of such a multiport repeater is the DEMPR which is a product of DEC; it allows for the connection of eight segments through eight ports, where the length of each segment can be up to 185 meters.

Attached to each segment is a finite number of stations (e.g., terminal servers, file servers, hosts, printer servers), where each station generates and receives packets. Each packet has a header which includes the address of the source and destination stations. If the source and the destination stations are on the same subnetwork, then the packet is part of the network *intratraffic*; otherwise, the packet is a part of the network *intertraffic*. We assume that packets are generated by the stations according to independent Poisson processes. If more than one station on a given subnetworks desires to transmit a packet, then the stations compete for the channel following the rules of a specific random access protocol. In order to simplify the analysis, we also make the following *station assumptions*:

(1) Each station can buffer an infinite number of packets.

(2) If a station has more than one packet to transmit, then each of these packets independently and simultaneously competes for the channel according to the random access protocol. (Note that these packets are also competing with packets held by other stations that are attached to the same subnetwork.) Thus, each packet becomes a "virtual station"; see Bertsekas and Gallager [3, p. 211] and Lam [10].

The Ethernet access protocol does not, of course, operate according to the second station assumption but instead schedules packets in a FIFO manner. However, under the second station assumption, the packet delay will be an upper bound to the delay that occurs with a FIFO scheduling policy; see Bertsekas and Gallager [3, p. 211].

Unlike a repeater, a bridge stores the entire packet before forwarding it on a segment. Once the packet is stored, the bridge must decide whether to discard it or forward it on one of the outgoing segments. This decision is made by examining a table inside the bridge, and matching the destination address of the packet with the corresponding segment. See Tanenbaum [15] for an excellent overview of transparent bridges.

If a bridge is to forward a packet onto a subnetwork, we assume it does so with the same random access protocol that is being employed by the stations in the subnetwork. In order to simplify the analysis, we also make the following *bridge assumptions:*

(1) Each bridge can buffer an infinite number of packets.

(2) If a bridge has more than one packet to transmit onto a given subnetwork, then each of these packets independently competes for the channel according to the random access protocol. (Note that these packets are also competing with the packets held by the stations that are attached to the same subnetwork.)

(3) Packets arrive to the bridges (from the stations *à la* random access) according to independent Poisson processes.

As with the second station assumption, the second bridge assumptions will lead to a packet delay that is an upper bound to the delay associated with FIFO scheduling.

THE EFFECTIVE LOAD ON A SUBNETWORK

Due to our simplifying assumptions, the stations and bridges on a subnetwork generate packets according to independent Poisson processes. The sum of the rates for these Poisson processes will be referred to as the *effective load on a subnetwork*. It is important to note that the effective load on a subnetwork does not include the retransmissions due to collisions.

For each $u \in U$, denote λ_u for the rate at which packets are generated by stations on segment u in packets per packet time. (We are assuming that the length of all packets is constant; however, general packet length distributions can be taken into account: see Lam [10].) Denote

$$\lambda := \sum_{u \in U} \lambda_u$$

for the total exogenous arrival rate to the system. For a packet generated by segment u, denote P_{uv} for the probability that segment v is its destination. Thus

$$\sum_{v \in U} P_{uv} = 1, \quad u \in U.$$

We assume that λ_u, $u \in U$, and P_{uv}, $u \in U$, $v \in U$, are known to the network designer. Further denote $\lambda_{uv} := \lambda_u P_{uv}$, $u \in U$, $v \in U$, for the exogenous arrival rate of packets with source segment u and destination segment v. For $1 \leqslant p, q \leqslant Q$, denote

$$\bar{\lambda}_{pq} := \sum_{u \in U_p} \sum_{v \in U_q} \lambda_{uv}$$

for the exogenous arrival rate of packets with source subnetwork p and destination subnetwork q. If we assume that the bridges operate so that a packet being sent from subnetwork p to subnetwork q traverses subnetwork s if and only if $s \in \pi(p, q)$, then the effective load on subnetwork s, denoted by γ_s, is

$$\gamma_s = \sum_{p=1}^{Q} \sum_{q=1}^{Q} \bar{\lambda}_{pq} \mathbf{1}[s \in \pi(p, q)]. \tag{1}$$

A very important property of the effective load on a subnetwork s is that it depends on the partition $\{U_1, \dots, U_s, \dots, U_Q\}$ only through U_s. In other words, γ_s does not change if the partition is changed to $\{U_s, U_1', \dots, U_{Q'}'\}$.

In order to give an alternative and useful expression for γ_s, consider the partial graph $[X, U - U_s]$ and let $V_1, \dots, V_I$ be the edge sets for the connected components of the partial graph: cf. Gondran and Minoux [8]. (For example, with $s = 3$, the connected components for the graph of fig. 1 become $V_1 = \{1, 2\}$, $V_2 = \{5\}$, $V_3 = \{7, 8, 9, 10, 11, 12\}$.) It is easily seen that

$$\gamma_s = \lambda - \sum_{i=1}^{I} \sum_{u \in V_i} \sum_{v \in V_i} \lambda_{uv}.$$

OVERHEAD TRAFFIC

In the derivation of (1) we tacitly assumed that the table in each bridge contains a list of *all* stations in the interconnected network along with the corresponding outgoing segments. However, most transparent bridges (see Tanenbaum [15]) purge station addresses from their tables which are more than, say, T seconds old. Thus, a bridge does not always know where to forward a packet, in which case flooding is typically employed. The flooding in turn generates overhead traffic which can impede the performance of the interconnected system.

In order to estimate the additional load on a subnetwork due to the overhead traffic, we make the following *learning assumptions*:

- When the network is powered up, the tables in all the bridges are identical.
- If a bridge receives a packet with a destination address that is not in its table, then the bridge floods the packet on each of the outgoing segments.
- If a bridge receives a packet with a source address that is not in its table, then the bridge (i) updates its table; and (ii) floods the packet on the outgoing segments. (This is a simplifying assumption.) We assume the flooded packets reach the perimeter of the network relatively quickly so that the tables in the bridges essentially remain identical.

Suppose that a packet is sent from subnetwork p to subnetwork q and either the source or destination address is not in the bridge table. According to the above learning mechanisms, this packet would generate overhead on each of the subnetworks which are not part of $\pi(p, q)$. (The packet would also generate traffic on the subnetworks in $\pi(p, q)$; however, this traffic does not count as overhead.) Denote by R_{pq} the probability that a packet being sent from subnetwork p to subnetwork q has either a source or destination address that is not in the bridge table. If we denote γ_s^o for the load due to overhead traffic on subnetwork s, then it follows from the above discussion that

$$\gamma_s^o = \sum_{p=1}^{Q} \sum_{q=1}^{Q} \bar{\lambda}_{pq} R_{pq} \mathbf{1}[s \notin \pi(p, q)].$$

Thus, with the above learning mechanism, the effective load on subnetwork s becomes

$$\gamma_s = \sum_{p=1}^{Q} \sum_{q=1}^{Q} \bar{\lambda}_{pq} R_{pq}(s), \tag{2}$$

where

$$R_{pq}(s) := \begin{cases} 1 & s \in \pi(p, q), \\ R_{pq} & s \notin \pi(p, q). \end{cases}$$

We now proceed to determine R_{pq}. Assume that the rate at which packets are being generated from a given station is β, independent of the source station. (This assumption can be relaxed at the expense of a more complicated but straightforward analysis.) It follows that the fraction of time that an address (with corresponding outgoing segment) is stored in the bridge memory is $1 - e^{-\beta T}$. Since overhead is generated if either the source or destination address of a newly generated packet is not in the bridge memory, it follows that $R_{pq} \approx 1 - (1 - e^{-\beta T})^2$.

3. Throughput analysis

Consider for the moment an interconnected network $G = [X, U]$ with $X_b = \emptyset$, i.e., a network without bridges. Denote l for the maximum propagation delay (normalized by the packet time) for the network; note that l is proportional to the diameter (the maximum propagation delay between any two stations on the network) of $G = [X, U]$. The throughput and delay characteristics for such networks with random access protocols have been studied both empirically and theoretically: see Lam [10], Bertsekas and Gallager [3], and Bux [4]. We shall denote $S(l)$ for the maximum attainable throughput. For example, if nonpersistent CSMA/CD is employed then (see Bux [4])

$$S(l) \approx \frac{1}{1 + 6.44l}. \tag{3}$$

We shall denote $T(l, \lambda)$ for the average delay when the offered load is λ. We suppose that the throughput and delay functions, obtained either empirically or theoretically, are available to the network designer. For example if nonpersistent CSMA/CD with fixed packet lengths is employed then (see Bux [4])

$$T(l, \lambda) = \lambda \frac{\left[1 + (4\,e + 2)l + 5l^2 + 4\,e(2\,e - 1)l^2\right]}{2\{1 - \lambda[1 + (2\,e + 1)l]\}} + 1 + \left(2\,e + \frac{1}{2}\right)l - \frac{(1 - e^{-2l\lambda})\left(\dfrac{2}{\lambda} + \dfrac{2l}{e} - 6l\right)}{2\left[e^{-\lambda(1+l)-1} - 1 + e^{-2l\lambda}\right]}. \tag{4}$$

It is natural to assume that (i) $S(\cdot)$ is decreasing in its argument; (ii) $T(\cdot,\cdot)$ is increasing in both arguments; (iii) $\lim_{\lambda \uparrow S(l)} T(l, \lambda) = \infty$; (iv) if $\lambda \geqslant S(l)$ then the system becomes unstable and the average delay is infinite.

We now return to the case of interest: $X_b \neq \emptyset$. Denote l_q for the maximum propagation delay (normalized by the packet time) for the qth subnetwork. Due to our assumptions in the previous section, we can view each subnetwork q as an isolated network without bridges but with an offered load of γ_q. Therefore we need

$$\gamma_q \leqslant S(l_q), \quad q = 1, \ldots, Q,$$

for each subnetwork to be stable. If each of the subnetworks is stable, the throughput of the interconnected system is $\lambda = \lambda_1 + \cdots + \lambda_Q$.

MAXIMUM THROUGHPUT

What is the maximum throughput of the interconnected system? It is straightforward to show that stability implies

$$\lambda \leqslant \sum_{q=1}^{Q} S(l_q),$$

and that this upper bound can be achieved by setting

$$\bar{\lambda}_{pq} = \begin{cases} S(l_q) & p = q, \\ 0 & p \neq q. \end{cases}$$

Thus the maximum throughput can be attained by decreasing the internetwork traffic and increasing the intranetwork traffic on each subnetwork to the corresponding maximum throughput.

What is the maximum throughput of the interconnected system if the routing probabilities are fixed? To answer this question, fix P_{uv}, $u \in U$, $v \in U$, and denote

$$c_u(p, s) := \sum_{q=1}^{Q} \sum_{v \in U_q} P_{uv} \mathbf{1}[s \in \pi(p, q)],$$

so that from (1) we have

$$\gamma_s = \sum_{p=1}^{Q} \sum_{u \in U_p} \lambda_u c_u(p, s).$$

Thus the exogenous arrival rates $\lambda^* := (\lambda_u^*: u \in U)$ that achieve maximum throughput (with routing probabilities fixed) are given by the solution to the following linear program (LP):

PROGRAM 1

$$\begin{aligned} \lambda^* := \quad & \max \quad \sum_{u \in U} \lambda_u \\ & \text{s.t.} \quad \sum_{p=1}^{Q} \sum_{u \in U_p} c_u(p, s) \lambda_u \leqslant S(l_s), \quad s = 1, \ldots, Q, \\ & \qquad\quad \lambda_u \geqslant 0, \quad u \in U. \end{aligned}$$

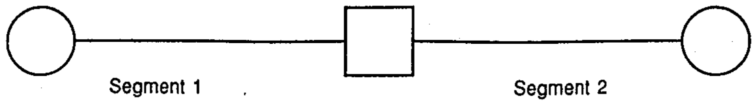


Fig. 2. An interconnected system of two segments and a bridge.

The above LP has N variables (excluding slacks) and Q constraints; it can be easily solved for problems of realistic size.

As an example, consider the interconnected system in fig. 2, which consists of two segments and a bridge. We then have $U_1 = \{1\}$ and $U_2 = \{2\}$. For this system

$$\gamma_1 = \lambda_1 + \lambda_2 P_{21},$$
$$\gamma_2 = \lambda_2 + \lambda_1 P_{12}.$$

Consider the problem of maximizing throughput with the routing probabilities fixed. Assume that $P_{12} > 0$, $P_{21} > 0$, and $P_{12}P_{21} < 1$. Also $\gamma_u < S(l_u)$, $u = 1, 2$. We now have the following:

(i) If $P_{uv}S(l_u) > S(l_v)$ or $S(l_u) > P_{vu}S(l_v)$ for $u \neq v$, $1 \leqslant u, v \leqslant 2$, then

$$\lambda_1^* = \frac{S(l_1) - P_{21}S(l_2)}{1 - P_{12}P_{21}},$$
$$\lambda_2^* = \frac{S(l_2) - P_{21}S(l_1)}{1 - P_{12}P_{21}},$$

hence,

$$\lambda^* = \frac{S(l_1) + S(l_2) - P_{12}S(l_1) - P_{21}S(l_2)}{1 - P_{12}P_{21}}.$$

(ii) Otherwise

$$\lambda^* = \min\left\{\frac{S(l_1)}{P_{12}}, \frac{S(l_2)}{P_{21}}\right\}.$$

Note that if a repeater is used instead of a bridge, then the maximum throughput of the system is $S(l_1 + l_2)$. Thus

$$\Delta(l_1, l_2) = \frac{(1 - P_{12})S(l_1) + (1 - P_{21})S(l_2)}{1 - P_{12}P_{21}} - S(l_1 + l_2) \tag{5}$$

represents the improvement that can be achieved with a bridge over a repeater under condition (i). Since $S(l)$ is decreasing in l, $\Delta(l_1, l_2) \geqslant 0$, so that it is always advantageous in terms of throughput performance to employ a bridge. To get a handle on the increase in performance, suppose that $P_{12} = P_{21} = P$ and $S(l)$ is given by (3). Then from (5) it can be shown that

$$\Delta(l_1, l_2) \approx \frac{1 - \dfrac{P}{1 + 6.44(l_1 + l_2)}}{1 + P}$$

when $l_1 + l_2 \ll 1$.

4. Delay analysis

Denote t_{pq} for the average delay of a packet with source subnetwork p and destination subnetwork q. It follows from the bridge assumptions that

$$t_{pq} = \sum_{s \in \pi(p,q)} T(l_s, \gamma_s). \tag{6}$$

Denote t for the average delay of a packet in the interconnected network. We have

$$\begin{aligned} t &= \frac{\sum_{p=1}^{Q} \sum_{q=1}^{Q} \bar{\lambda}_{pq} t_{pq}}{\lambda} \\ &= \frac{1}{\lambda} \sum_{p=1}^{Q} \sum_{q=1}^{Q} \bar{\lambda}_{pq} \sum_{s=1}^{Q} \mathbf{1}[s \in \pi(p,q)] T(l_s, \gamma_s) \\ &= \frac{1}{\lambda} \sum_{s=1}^{Q} T(l_s, \lambda_s) \sum_{p=1}^{Q} \sum_{q=1}^{Q} \bar{\lambda}_{pq} \mathbf{1}[s \in \pi(p,q)] \\ &= \frac{\sum_{s=1}^{Q} T(l_s, \gamma_s)\gamma_s}{\lambda}. \end{aligned} \tag{7}$$

Thus, once the effective load on each of the subnetworks has been determined, the average end-to-end delay can be calculated in a straightforward manner.

As an example, consider again the network in fig. 2. From (6) we have

$$t = \frac{T(l_1, \lambda_1 + \lambda_2 P_{21})[\lambda_1 + \lambda_2 P_{21}] + T(l_2, \lambda_2 + \lambda_1 P_{12})[\lambda_2 + \lambda_1 P_{12}]}{\lambda_1 + \lambda_2}. \tag{8}$$

If the two segments are instead connected with a repeater, then the delay becomes

$$t' = T(l_1 + l_2, \lambda_1 + \lambda_2). \tag{9}$$

In the case $P_{12} = P_{21} = 0$, i.e., no internetwork traffic, it is easy to show from (8), (9), and the monotonicity properties of $T(\cdot,\cdot)$ that $t < t'$. On the other hand, if either $P_{12} > 0$ or $P_{21} > 0$, then we may have $t < t'$ or $t > t'$! Under conditions of heavy loading, i.e., $\lambda_1 + \lambda_2$ close to $S(l_1 + l_2)$, the delays in the combined system will be very high and it will be advisable to include a bridge in the system. But for the case when $\lambda_1 + \lambda_2 \ll S(l_1 + l_2)$, the average delay is dominated by the fixed delay, $T(l, 0)$, which includes the packet transmission time and the propagation delay. It would then be advisable to connect them without a bridge as the delay for the internetwork packets is now almost twice as large as with a bridge due to the fact that they have to be transmitted once on each segment.

5. Optimal location of bridges

Let $Y \subseteq X - X_t$ be a fixed subset of nodes. We view each node in Y as a potential location for a bridge. A node in $X - X_t - Y$ is either a repeater or a bridge, but is assumed to be fixed due to either civil engineering or performance constraints. We consider the problem of determining which nodes in Y should be bridges in order that average delay, t, is minimized. We also consider optimization problems which take into account the cost of bridges.

At first glance, it may seem most natural to formulate these optimization problems as location problems: see Bertsekas and Gallager [3]. A more promising formulation, however, is based on set partitioning: see Gondran and Minoux [8].

Let $\xi_1, \xi_2, \ldots, \xi_M$ denote the potential subnetworks that can be generated by varying the placement of bridges in Y. An upper bound on the number of subnetworks is $2^N - 1$, although in practice it will be much smaller as it depends on the topology of $G = [X, U]$. The number of subnetworks can be reduced with the following preprocessing:

- If $l_m > 0.1$ then CSMA/CD is inappropriate and ξ_m can be removed from the list of subnetworks.
- Let γ_{fil} be the maximum rate at which any given port of a bridge can filter traffic. If $\gamma_m > \gamma_{fil}$, then ξ_m can be removed from the list of potential subnetworks.
- Let δ_i be the total amount of traffic that must be forwarded by node i. Note that δ_i for all nodes i is easy to calculate from λ_{uv}, $u \in U$, $v \in U$. Let the maximum amount of traffic that can be forwarded by a bridge be γ_{fwd}. Therefore if $\gamma_{fwd} < \delta_i$, then node i can be fixed as a repeater and removed from Y.
- If $\gamma_m \geqslant S(l_m)$, then the average delay would be infinite if the mth subnetwork were included in the design. Thus, ξ_m can be removed from the list of potential subnetworks.

Now suppose that M subnetworks remain after preprocessing. Let

$$c_m := \frac{\gamma_m}{\lambda} T(l_m, \gamma_m), \quad m = 1, \ldots, M. \tag{10}$$

(Recall that γ_m depends only on ξ_m and not on the other sets in the partition.) It follows from (7) that the problem of minimizing the average delay is equivalent to the well-known set partitioning problem with the weight c_m, $m = 1, \ldots, M$. The set partitioning problem can in turn be cast as the following $0-1$ integer program:

PROGRAM 2

$$\begin{aligned} \min \quad & \sum_{m=1}^{M} c_m z_m \\ \text{s.t.} \quad & \sum_{m=1}^{M} \mathbf{1}(u \in \xi_m) z_m = 1, \qquad u \in U, \\ & z_m = 0 \text{ or } 1, \qquad\qquad m = 1, \ldots, M. \end{aligned}$$

If $z^* = (z_1^*, \ldots, z_M^*)$ is an optimal solution to program 2, then an optimal design would consist of the subnetworks ξ_m such that $z_m^* = 1$.

Since bridges can be expensive it is natural to incorporate their cost in the optimization criterion. This can be easily done if we assume that the degree of every node in Y is the same. This assumption may not be unrealistic since most bridges currently connect only two segments. If all the nodes in Y have the same degree, say d, and if bridges are placed at K nodes, then the number of subnetworks in the partition is $K(d-1)+1$. Thus if z is a feasible solution to program 2, then the number of bridges called for by the solution z is

$$B(z) := \frac{\sum_{m=1}^{M} z_m - 1}{d-1}.$$

Therefore, if we include

$$\sum_{m=1}^{M} z_m \leqslant K(d-1)+1$$

in the constraints of program 1, then we restrict the number of bridges to be no more than K. Alternatively, we could minimize $B(z)$ and include

$$\sum_{m=1}^{M} c_m z_m \leqslant \alpha$$

in the set of constraints. This formulation would correspond to minimizing the number of bridges subject to the constraint that the average delay be smaller than α.

Due to its application in many fields, the set partitioning problem has been thoroughly studied by mathematical programmers. Although the set partitioning problem is NP-complete: see Garey and Johnson [7], it is now possible to solve such problems in reasonable CPU time with tens of thousands of variables: see Fisher and Kedia [6]. We refer the reader to this reference as well as the textbooks by Minoux and Gondran [11], and Nemhauser and Wolsey [12].

As an example consider the network of fig. 3. The circles represent either termination points or permanent multiport repeaters. The squares represent locations at which either bridges or repeaters can be placed. These locations are labeled a through k. Note that certain segments are always grouped together

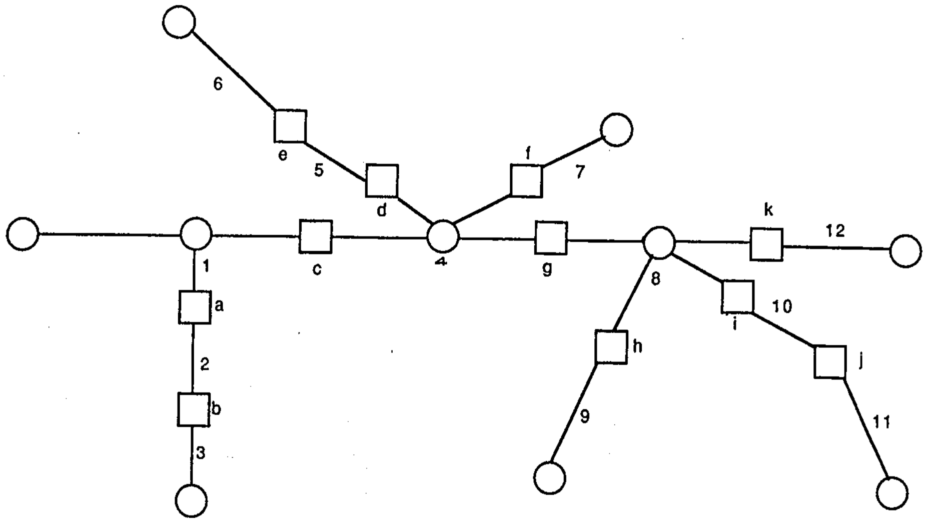


Fig. 3. Network topology for the considered example.

independent of how the bridges are located. For convenience, we shall refer to such a group of segments as a single segment. Thus, in fig. 3 we have numbered 12 segments. Suppose the length in terms of the normalized propagation delay for the 12 segments is: 0.03, 0.02, 0.02, 0.04, 0.02, 0.015, 0.02, 0.035, 0.02, 0.01, 0.01, 0.02. The exogenous arrival rates λ_{uv}, $1 \leqslant u, v \leqslant 12$, are given in table 1. A blank entry corresponds to no traffic between the corresponding LAN pairs. We suppose that $\gamma_{fil} = 0.9$ and $\gamma_{fwd} = 0.25$.

A careful look at the traffic matrix reveals that the amount of traffic that needs to be forwarded by node g is 0.27, more than the maximum filtering rate supported by a bridge. This rules out node g as a potential location for a bridge. Moreover, several subnetworks can be removed as they have lengths larger than 0.1. The list of subnetworks can be further pruned by removing those which are

Table 1
Traffic matrix for fig. 3

0.04	0.09	0.01			0.02		0.02	0.01			
0.06	0.05		0:02							0.01	
0.02		0.35	0.01	0.02		0.01					0.01
			0.03	0.02	0.04						0.01
				0.04	0.11		0.02				
	0.01			0.07	0.08		0.02		0.01		
0.01		0.01	0.01			0.04		0.01			0.02
				0.02			0.02			0.02	
0.01			0.02				0.02	0.27		0.02	
									0.03	0.06	0.02
			0.01				0.01		0.05	0.04	
0.02			0.03			0.01	0.04				0.10

unstable, i.e., $\gamma_q > S(l_q)$. The unstable subnetworks are {1, 2, 3}, {4, 5, 8}, {4, 5, 7, 8}, {4, 8, 9}, {4, 8, 10, 11}, {4, 8, 10, 12}. The subnetworks that remain to be considered are: {1}(0.18), {2}(0.133), {3}(0.352), {5}(0.175), {6}(0.222), {7}(0.06), {9}(0.234), {10}(0.094), {11}(0.124), {12}(0.139), {1, 2}(0.296), {2, 3}(1.55), {5, 6}(0.39), {4, 8}(0.338), {4, 8, 7}(0.97), {4, 8, 10}(3.187), {4, 8, 12}(2.447), and {10, 11}(0.175). The cost (calculated via (10) and (4)) associated with each subnetwork is shown in parentheses. The corresponding integer program is now:

$$\begin{aligned}
\min\quad & \{0.18z_1 + 0.113z_2 + 0.352z_3 + 0.175z_4 + 0.222z_5 \\
& +0.060z_6 + 0.234z_7 + 0.094z_8 + 0.124z_9 + 0.139z_{10} \\
& +0.296z_{11} + 1.550z_{12} + 0.390z_{13} + 0.338z_{14} \\
& +0.970z_{15} + 3.187z_{16} + 2.447z_{17} + 0.175z_{18}\} \\
\text{s.t.}\quad & z_1 + z_{11} = 1 \\
& z_2 + z_{11} + z_{12} = 1 \\
& z_3 + z_{12} = 1 \\
& z_4 + z_{13} = 1 \\
& z_5 + z_{13} = 1 \\
& z_6 + z_{15} = 1 \\
& z_7 = 1 \\
& z_8 + z_{16} + z_{18} = 1 \\
& z_9 + z_{18} = 1 \\
& z_{10} + z_{17} = 1 \\
& z_{14} + z_{15} + z_{16} + z_{17} = 1 \\
& z_m = 0 \text{ or } 1, \quad m = 1, \ldots, 18.
\end{aligned}$$

The solution consists of the following subnetworks: {1, 2}, {3}, {5, 6}, {7}, {4, 8}, {9}, {10, 11}, and {12}. Hence bridges should be placed at nodes b, c, d, f, h, i, k and repeaters at a, e, g, j to get an average delay of 1.95 packet transmission times for the network.

Now consider the problem of minimizing the number of bridges subject to the constraint of 3 packet transmission times for the average delay. We must then modify the above integer program as discussed above. The optimal solution consists of the following subnetworks: {1, 2}, {3}, {5, 6}, {4, 8, 7}, {9}, {10, 11}, {12}. The average delay in the network is now 2.52 packet transmission times. The number of bridges called for by the optimal solution has been reduced from 7 to 6.

6. Linear topologies

In a office building there is typically a network of segments on each floor, and each of these networks is connected by a spine that runs up the height of the

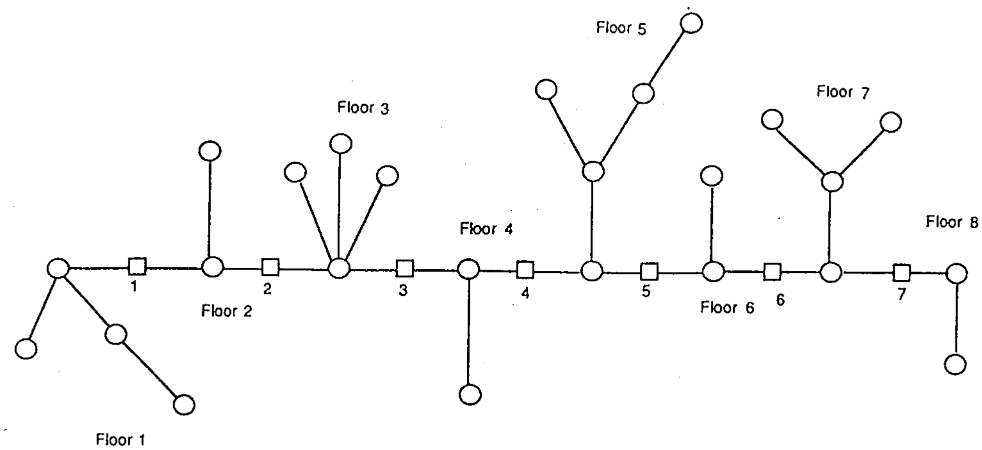


Fig. 4. Example of a linear topology.

building. If the network is heavily utilized, then it may become necessary to place bridges on the spine between the various floors in order to localize traffic. We are therefore led to consider "linear topologies" as in fig. 4. (As before, the squares represent potential bridge locations.)

It turns out that the problem of placing bridges at the L locations in order to minimize average delay can be solved efficiently for the important special case of a linear topology. To demonstrate this claim, let $\xi_{k,l}$ denote the subnetwork between the potential bridge locations k and l, where $0 \leqslant k \leqslant l \leqslant L+1$. Thus, $\xi_{0,L+1}$ is the subnetwork consisting of the entire network, and $\xi_{l-1,l}$ is the subnetwork on the lth floor. Let $c_{k,l}$ be the cost, as defined through (10), of the subnetwork $\xi_{k,l}$. Now for $l = 1, \dots, L+1$, consider the subproblem of placing bridges in order to minimize the average delay of a network consisting of only floors 1 through l. Let g_l be the minimum delay for the lth subproblem. Note that g_{L+1} is the minimum delay for the original problem. Clearly, g_l, $l = 1, \dots, L+1$, can be calculated recursively as follows:

$$g_0 = 0,$$

$$g_l = \min\{g_k + c_{k,l} \colon 0 \leqslant k \leqslant l-1\}, \quad l = 2, \dots, L+1.$$

The arguments at which the minima occur determine the subnetworks, and hence the bridges, that are to be included in the optimal design.

Thus the problem of locating bridges on a linear topology in order to minimize average delay can be solved in $O(L^2)$ time. Note the close resemblance of this problem to the knapsack problem: see Denardo [5]. We can also incorporate bridge costs into the optimization criterion by employing a single Lagrange multiplier.

As an example we consider the network in fig. 4 (where $L = 7$). The length of the segments (as before in terms of maximum propagation delay) are: 0.005,

Table 2
Traffic matrix for fig. 4

0.07	0.10			0.01		0.01	
0.01	0.06	0.05			0.01		0.01
0.01	0.01	0.15	0.01	0.01		0.02	0.01
	0.01	0.01	0.03	0.01	0.01	0.01	
0.01	0.01	0.02	0.02	0.02	0.01		0.02
0.01		0.01	0.03	0.02	0.01	0.01	
	0.01		0.02	0.02		0.28	0.15
0.01	0.01			0.02	0.01	0.19	0.30

0.005, 0.008, 0.002, 0.003, 0.001, 0.003, 0.004. The associated traffic matrix is given in table 2. As before blank entries should be interpreted as there being no traffic between the corresponding LANs. After preprocessing the valid subnetworks are {1}, {1, 2}, {2}, {1, 2, 3}, {2, 3}, {3}, {1, 2, 3, 4}, {2, 3, 4}, {3, 4}, {4}, {2, 3, 4, 5}, {3, 4, 5}, {4, 5}, {5}, {2, 3, 4, 5, 6}, {3, 4, 5, 6}, {4, 5, 6}, {5, 6}, {6}, {6, 7}, {7}, and {8}. The corresponding costs are 0.151, 0.319, 0.19, 0.0878, 0.501, 0.208, 2.346, 0.882, 0.355, 0.095, 2.38, 0.607, 0.224, 0.122, 4.575, 0.757, 0.283, 0.198, 0.076, 1.891, 0.930, 0.989. From (11) and (12) we have that $g_1 = 0.151$, $g_2 = 0.319$, $g_3 = 0.527$, $g_4 = 0.622$, $g_5 = 0.744$, $g_6 = 0.81$, $g_7 = 1.74$, $g_8 = 2.73$. The subnetworks that figure in the optimal solution are {1, 2}, {3}, {4, 5, 6}, {7}, and {8}. The optimal locations for the bridges are at the nodes 2, 3, 6, and 7. The minimum average delay for the network is 2.73 packet transmission times.

7. Extension to interconnected token rings

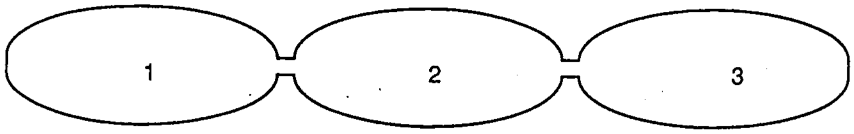
Consider a case where each subnetwork is a token ring. It is convenient to represent a token ring by a bus even though the topology is that of a ring. The length of a subnetwork is now the round trip propagation delay on the ring including the delay introduced by each station. Since each station acts as a repeater, two token rings can be connected to each other by simply breaking and connecting them (there is no need of a special device); see fig. 5. Bridges for token rings operate in a manner similar to those for Ethernets.

To aid in the analysis of token ring networks we make the following commonly employed assumptions (see Bertsekas and Gallager [3]):

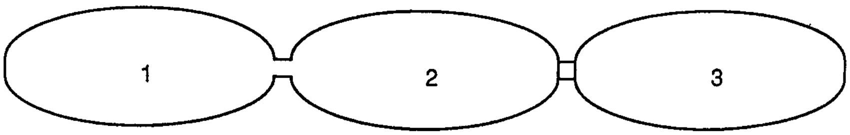
(1) Each station can buffer an infinite number of packets.

(2) Packets at any given station are served in First Come First Served (FCFS) order.

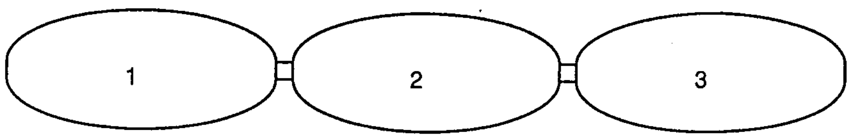
(3) All stations generate the same amount of traffic. Hence the number of stations on a segment is assumed to be proportional to the exogenous traffic on a segment.



(a)



(b)



(c)

Fig. 5. Bridging three token ring segments: (a) two "repeaters"; (b) one "repeater" and one bridge; (c) two bridges.

In addition the following bridge assumptions are made:
(1) Each bridge can buffer an infinite number of packets.
(2) Packets residing in the bridge buffer are served according to FCFS discipline.
(3) Packets arrive to the bridge according to a Poisson process.

Under the assumptions made above the maximum attainable throughput of a token ring of length l is (see Bertsekas and Gallager [3]):

$$S(l) = \frac{1}{1+l}. \tag{11}$$

It should be emphasized that (11) is true under the assumptions (consistent with the IEEE 802.5 standard): (i) each station transmits a single packet each time it captures the token and; (ii) a station relinguishes the token only after its outstanding packet has returned to the station. The average delay for an IEEE 802.5 token ring with offered traffic λ and length l is approximated by the well known formula (see Bertsekas and Gallager [3], and Schwartz [14]):

$$T(l, \lambda) = \frac{\lambda(1+l)^2 + l + \lambda l(1+l)/m}{2(1-\lambda(1+l))}, \tag{12}$$

where m is the number of stations on the ring. It is easy to see that most of the ideas in sections 2 through 5 can now be applied to networks of interconnected token rings with (3) and (4) replaced by (11) and (12) respectively, and the stations and bridge assumptions of section 2 replaced by those made earlier in this section. Also the preprocessing rules can be suitably modified to take into account the limitations of the MAC protocol under consideration.

Expressions for the maximum throughput and average delay can be found in the literature for other LAN standards: see Bertsekas and Gallager [3], and Schwartz [14]. Therefore, the methodology can be applied to a variety of other LAN protocols as well.

8. Concluding remarks

In this paper we have developed a performance and optimization methodology for internetworks of LANs, bridges, and repeaters. For IEEE 802.3 Ethernets the methodology hinges on the station and bridge assumptions, and in particular on the assumption that packets waiting in the same station or in the same bridge compete with each other for access to the channel. As stressed in section 2, this assumption will give an upper bound to average delay. It is of interest to determine whether this upper bound is tight for realistic loads and whether the truly optimal solution for the location of the bridges differs significantly when the upper bound is loose. We have also shown how the methodology applies to networks of bridged token rings. By making suitable station and bridge assumptions the methodology can be applied to other LAN standards as well.

A related problem involves the location of file servers (either heterogeneous or homogeneous) on the internetwork. This would involve determining (i) on which subnetworks the file servers should be placed; (ii) how traffic should be directed in the case of multiple homogeneous file servers. The problem of simultaneously locating bridges and file servers on an internetwork is also of interest. Some initial results for these problems are given in Gupta [9].

References

[1] F. Backes, Transparent bridges for interconection of IEEE 802 LANs, IEEE Network 2 (1988) 5–9.

[2] E. Benhamou, Integerating bridges and routers in a large internetwork, IEEE Network 2 (1988) 65–71.
[3] D. Bertsekas and R. Gallager, *Data Networks* (Prentice–Hall, Englewood Cliffs, NJ, 1987).
[4] W. Bux, Local Area Networks: A performance comparison, IEEE Trans. Comm., COM-29 (1981) 1465–1473.
[5] E.V. Denardo, *Dynamic Programming: Models and Applications* (Prentice–Hall, Englewood Cliffs, NJ, 1982).
[6] M.L. Fisher and P. Kedia, Optimal solution of set covering/partitioning problems with dual heuristics, to appear in Manag. Sci. (1988).
[7] M.R. Garey and D.S. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness* (Freeman, 1979).
[8] M. Gondran and M. Minoux, *Graphs and Algorithms* (Wiley, Chichester, 1984).
[9] S. Gupta, Performance modeling and optimization of interconnected Ethernets, Master's thesis, University of Pennsylvania, Dept. of Electrical Engineering (1989).
[10] S.S. Lam, A carrier sense multiple access protocol for local networks, Computer Networks 4 (1980) 21–32.
[11] M. Minoux, A class of combinatorial problems with polynomially solvable large scale set covering/partitioning relaxations, RAIRO Oper. Res. 21 (1988) 105–134.
[12] G.L. Nemhauser and L.A. Wolsey, *Integer and Combinatorial Optimization* (Wiley, New York, 1988).
[13] H. Salwen, R. Boule and J.N. Chiappa, Examination of the applicability of router and bridging techniques, IEEE Network 2 (1988) 77–80.
[14] M. Schwartz, *Telecommunication Networks: Protocols, Modeling and Analysis* (Addison–Wesley, Reading, MA, 1987).
[15] A.S. Tanenbaum, *Computer Networks* (Wiley, Chichester, 1988).