Stochastic Performance Evaluation of Hierarchical Routing for Large Networks

Farouk Kamoun
Departement Informatique, Faculté des Sciences, Tunis, Tunisia

and

Leonard Kleinrock
Computer Science Department, University of California, Los Angeles, USA

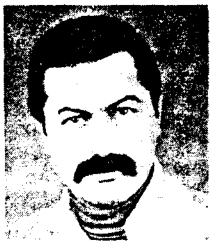
In its present form, distributed routing extracts a prohibitive price when used in large networks because of the processing time, nodal storage and line capacity required to update, store and exchange routing information among network nodes. In an earlier paper we have shown that hierarchical routing schemes with optimally selected clustering structures yield enormous reductions in routing table length and hence in routing cost, at the price of an increase in network path length. That increase was shown to be negligible in the limit of very large networks. In this paper, we evaluate the tradeoff between the reduction in routing table length and the increase in network path length in terms of the more meaningful network performance measures of *delay* and *throughput*. Extended queueing models are developed to exhibit the interrelationships which exist between network variables such as delay, throughput, channel capacity, nodal storage, network path length, routing table length, etc. These models are an extension of the classic model for networks in that they account for line overhead and storage requirements due to routing. The models demonstrate the enormous efficiency of optimized hierarchical routing for a class of large networks.

Keywords: Hierarchical routing, Routing, Computer networks, Networks, Large networks, Network overhead, Network storage, Nodal storage, Adaptive routing, Delay, Throughput

1. Introduction

In order to reduce the cost of adaptive routing in large networks, we studied hierarchical routing in a previous paper [11]. This cost is mainly composed of the nodal storage required by the routing tables, the line capacity needed for the exchange of routing information and the nodal processing capacity to update the tables. In an N-node network, present routing schemes ordinarily provide one entry per destination node in the *r*outing *t*able (RT) which

Leonard Kleinrock received the B.E.E. degree from City College of New York, NY, in 1957 and the M.S.E.E. and Ph.D.E.E. degrees from the Massachusetts Institute of Technology, Cambridge, MA, in 1959 and 1963 respectively. In 1963, he joined the faculty of the School of Engineering and Applied Science, University of California, Los Angeles, where he is now Professor of Computer Science. His research spans the fields of computer networks, computer systems modeling and analysis, queueing theory, and resource sharing and allocation in general. At UCLA, he directs a group in advanced teleprocessing systems and computer networks and is the Head of the Computer Science Department Research Laboratory. He is the author of three major books in the field of computer netwcrks: *Communication Nets: Stochastic Message Flow and Delay* (New York: McGraw-Hill, 1964; also New York: Dover, 1972); *Queueing Systems, Vol. I: Theory* (New York, Wiley-Interscience, 1975); and *Queueing Sytems, Vol. II: Computer Applications* (New York: Wiley-Interscience, 1976). He has published over 100 articles and contributed to several books. He serves as lecturer and consultant for many domestic and foreign corporations and governments and he is a referee for numerous scholarly publications and a book reviewer for several publishers. He is a Guggenheim Fellow, an IEEE Fellow, and has received various outstanding teacher and best paper awards, including the 1976 Lanchester prize for the outstanding paper in operations research, and the ICC '78 prize-winning paper award.



Farouk Kamoun was born in Sfax, Tunisia on October 20, 1946. He received the Engineering Degree from Ecole Supérieure d'Electricité Paris. France in 1970 and the M.S. and Ph. D. degrees in computer science from the University of California, Los Angeles, in 1972 and 1976, respectively. From 1973 to 1976 he was with the University of California, Los Angeles, where he participated in the ARPA Network Project as a Postgraduate Research Engineer and did research on design considerations for large computer communication networks. He is currently teaching at the Ecole Nationale d'Ingenieurs de Tunis, Tunisia.

* This research was supported by the Advanced Research Projects Agency of the Department of Defense under contract MDA 903-77-C-0272.

Computer Networks 3 (1979) 337–353

leads to a routing table length of N entries. As N becomes large (i.e., "large" networks), the cost of routing in its present form clearly becomes prohibitive. The purpose of hierarchical routing is to reduce this cost through the reduction of the RT length without impacting the network performance.

In our previous study, we capitalized on the intuitive idea of grouping nodes into "natural" clusters and consequently provided one entry per set of clustered destinations in the RT. That is, we dealt with an m-level hierarchical clustering of the set of nodes. We also observed that an optimal clustering structure could be selected so as to minimize the table length. Such an optimal structure leads to very significant savings in table length. The price we pay for this table reduction is an increase in the network path length since we have somewhat less routing information available at each node. Bounds were derived to evaluate the maximum increase in path length for a given table reduction. The bounds demonstrated a key result, namely, that in the limit of very large networks, enormous table reduction may be achieved with *no* significant increase in network path length. In other words, in the limit, hierarchical routing schemes achieve a performance as good as present schemes with very substantial savings in storage and capacity.

In this paper we evaluate the performance of the *h*ierarchical *r*outing (MHR [*]) in terms of delay and throughput, and determine those values of N where clustering becomes economical.

In what follows, we first recall the classic model for network delay analysis due to Kleinrock [8]. We observe that the model does not account for overhead in nodal storage requirements and line capacity due to routing updates, both of which become critical in large networks. In this paper, after introducing a key assumption which permits us to map hierarchical and non-hierarchical adaptive routing into *deterministic* (fixed) routing, we develop extended network queueing models. These models serve in the evaluation of the delay-throughput performance of the MHR as applied to an interesting *class of networks.* In our first model (based directly on the classic model in [8]), we relate the gains obtained by hierarchical routing simply to the relative table length l/N. In other words, we consider an ideal situation where infinite nodal storage is available and where the line capacity used by the update information is still negligible. As a result of this idealized situation and as shown in our previous paper, *N*on-*C*lustered *R*outing (NCR) schemes yields a network performance which is superior to the MHR except in the limit of very large nets where they are quite equivalent.

In the second model we account for the traffic generated by the routing updates while keeping the infinite nodal storage assumption. This model is the first to exhibit the infeasibility of the NCR techniques and the efficient behavior of the MHR's for a class of large networks.

In the third model the nodal storage is assumed to be finite whereas the update traffic is considered negligible. First, some new results are developed which show the effect of finite storage on network performance. Then, this model is used to evaluate the MHR schemes. The MHR's are found to achieve a behavior similar to that obtained with our second model.

Finally our fourth model accounts both for the routing updates and nodal storage in an approximate way. This fourth model also confirms our earlier observations which support the use of MHR in large networks.

2. A Queueing Model with no Updates and no Storage Limitation

In what follows we recall the major results for the classic delay analysis in computer networks developed in [8]. The key performance measure of a store and forward network (S/F net) is the total average delay T that a message spends in the network. T can be expressed simply in terms of the individual channel delays.

$$T = \sum_{i=1}^{M} \frac{\lambda_i}{\gamma} T_i , \tag{1}$$

where λ_i = average traffic rate on channel i [msg/sec], T_i = average nodal processing plus queueing plus transmission time on the ith channel [sec], M = number of network channels, γ = total input rate (network throughput). We have $\gamma = \Sigma_{j,k}\gamma_{jk}$ where γ_{jk} = average message rate (msg/sec) from source j to destination k. Moreover, making the assumptions of external Poisson arrivals, exponential message length distribution (identical for all messages), single packet messages, error-free channels, no nodal processing delay, independence assumption, deterministic rout-

[*] M stands for the number of levels m in the hierarchy.

ing and infinite nodal storage, the S/F net can be modelled as a network of Jackson type queues [4,10].

$$T = \frac{1}{\gamma} \sum_{i=1}^{M} \frac{\lambda_i}{\mu C_i - \lambda_i}, \tag{2}$$

where: $1/\mu$ = average message length [kilobits/message], C_i = capacity of channel i [kilobits/sec – KBPS].

A simple relationship exists between the total internal traffic $\lambda = \Sigma_{i=1}^{M} \lambda_i$, the total external traffic γ, and the average traffic-weighted path length $\overline{n}$, namely [8],

$$\overline{n} = \lambda/\gamma . \tag{3}$$

The average rate $\lambda_i (i = 1, ..., M)$ can be computed numerically, given the underlying deterministic routing. Fortunately, for some symmetrical networks (see below) a simple analytical relationship exists among these variables.

Remarks: A discussion of the above assumptions and further extensions of this model can be found in [2,3,5,8–10,14]. In what follows we systematically relax the routing and storage assumptions.

Typically, a network is designed using deterministic routing and then operated using adaptive routing. Consequently the validity of our deterministic routing assumption depends critically on the difference in behavior between deterministic and adaptive techniques. In [2], Fultz finds a close agreement between the two techniques. He shows in a 19-node application that the difference in delays obtained with a well chosen adaptive technique and with a near-optimal deterministic technique is less than 5 to 10% almost until saturation. The "adaptive" delay tends to be higher, mainly because of the line overhead utilized by the update traffic. Fultz's study is, of course, dependent on the particular adaptive policy and network considered. However, it demonstrates that adequate adaptive policies can be devised to achieve a performance very close to that of a near-optimal deterministic policy. Further refinement of the above assumption can be realized by including the line overhead due to the update traffic. This consideration will become crucial when dealing with large nets, as we show below.

With respect to the infinite nodal storage assumption, it is fairly accurate when reasonable storage is provided, however, this assumption often is unacceptable. This situation is very likely to occur in a large network environment if the routing tables are not reduced to a reasonable length. A model is presented below to deal precisely with this question. First we describe a class of networks for which we will obtain explicit results.

2.1. A Class of Symmetric Networks

The class of nets to be considered in this paper is composed of all those which belong to the family of nets presented in [11]. Briefly, the networks considered are all the connected graphs upon which it is possible to fit an m-level hierarchical clustering such that all cluster subnets are of diameter (d) which is bounded by a power law function of the number (n) of nodes in that cluster: $d \leqslant bn^{v} + c$, where b, c and v are given constants. Moreover, they must also satisfy the following properties:

i All nodes are equivalent with respect to the topology of the network; hence, for example, they are of equal degree R.
ii All channels are of equal capacity C.
iii All external input traffic rates are equal: $\gamma_{jk} = \gamma_0 \ \forall \ j, k (j \neq k)$

As an example, torus nets [7] fall into this category.

For this class of nets, the following relations exist:

Number of (simplex) channels: $M = RN$,
Total external traffic $\gamma = N(N-1)\gamma_0$.

Furthermore, it is obvious that with this particular topological structure, capacity assignment and traffic requirement, that the optimal flow assighment [3,8] is a shortest path routing. The selection of the particular shortest paths (if more than one exists) must result in perfectly balanced flows, i.e., $\lambda_i = \lambda_0 (i = 1, 2, ..., M)$. Consequently the network path length $\overline{n}$ becomes the average shortest path length h, and so $\overline{n} = h = \lambda/\gamma$. (Path length is measured in $h$ops).

Also, the total internal traffic becomes $\lambda = M\lambda_0$. Hence,

$$\lambda_0 = h\,\gamma/M . \tag{4}$$

If we let $t = T_i$ denote the average delay on any channel (the same value for all channels), then from Eq. (1) the average total delay simply becomes

$$T = ht . \tag{5}$$

Moreover as a consequence of Eqs. (2) and (4)

$$T = \frac{1}{(\mu C/h) - (\gamma/M)} . \tag{6}$$

This simply relates the delay T, the traffic γ, the channel capacity C and the network path length h.

The last equation permits the interesting observation that the net is equivalent (with respect to average delay) to a single $M/M/1$ queue with an input rate of γ/M and service rate $\mu C/h$. This leads to the definition of the network channel utilization.

$$\rho = \frac{h}{\mu C}\frac{\gamma}{M}. \tag{7}$$

This definition will be used throughout the paper, mainly for normalization purposes.

Since our main objective is to study routing in large nets, it is necessary to specify the structure of those large nets with respect to the number of nodes, N, in some continuous way. Such a specification will be referred to as a scaling scheme (or strategy).

2.1.1. A Scaling Scheme

As a network grows, a reasonable objective of a scaling strategy is to maintain the average delay T constant (say $T = T_0$) and to let the total throughput γ increase linearly with the number of nodes. Due to the uniform traffic condition ($\gamma_{jk} = \gamma_0$), the total input rate per node is, consequently, maintained constant, i.e., $\gamma/N =$ constant. Also since $M = RN$ then γ/M is constant. Thus, from Eq. (6), in order to maintain constant delay, the capacity must grow in proportion to the network average path length, i.e.,

$$C = hC_0\,. \tag{8}$$

Substituting this into Eq. (6), we arrive at

$$\gamma/M = \mu C_0 - T_0^{-1} \tag{9}$$

which is constant with respect to N. Note with this scheme that ρ is also maintained constant. Such a property will not be true when dealing with network models which take into account the updates and/or the storage limitation.

In order to evaluate the performance of the MHR schemes we must recall some of the results obtained in [7,11] and introduce the additional assumption discussed in the next section.

2.2. Modelling of the Hierarchical Adaptive Routing

In [11] we considered three hierarchical schemes: the *O*verall *B*est *R*outing (OBR), the Closest Entry Routing (CER), and the *N*on-*C*lustered *R*outing (NCR). We observed that NCR is equivalent to a degenerate one-level OBR or CER. As a result and throughout the balance of this paper we will refer to the three routing schemes (OBR, CER, NCR) as hierarchical routing schemes with the understanding that $m = 1$ refers to NCR where m is the number of levels. Moreover we observed that the underlying optimal clustering structure is completely defined once we know the table length l or the number of hierarchical levels, m. In [11] it was shown that given m, the optimal table length is $l = mN^{1/m}$ with $1 \leqslant m \leqslant \ln N$. In addition we let m take on non-integer values. In what follows, m will be referred to as the *degree of clustering*. (In a real application, we must, of course, make m discrete by raising it to its upper integer value and then choose the clustering structure which leads to the best performance.) In [7] we observed that such a choice of clustering structure in general achieves tighter bounds on the increase in path length (E), and therefore we are justified in allowing m to be a real variable. Note if $m = 1$ then $l = N$, i.e., a full table length is required which corresponds to an NCR.

The increase in network path length due to clustering can be computed numerically, given a specific network and a specific MHR scheme. Fortunately, we have derived bounds which allow us to conduct a worst and/or best case analytical performance evaluation of hierarchical routing for the class of networks considered here. Recall that those networks constitute a subset of the family studied in [11], and hence we are able to use the following explicit expression for the bound on the relative increase in path length (for a table length, $l = mN^{1/m}$):

$$0 \leqslant \frac{h_c}{h} - 1 \leqslant E \triangleq \frac{1}{a(N-1)N^{\upsilon}} \times \left[N\left(b\,\frac{N^{\upsilon} - N^{\upsilon/m}}{N^{\upsilon/m} - 1} + c(m-1)\right) - b\,\frac{N^{\upsilon+1} - N^{(\upsilon+1)/m}}{N^{(\upsilon+1)/m} - 1} - c\,\frac{N - N^{1/m}}{N^{1/m} - 1}\right]. \tag{10}$$

Here, h_c refers to the average path length with clustering.

The bound * E is valid for both the OBR and the CER schemes. Other bounds on the increase in path length on a node-pair basis have also been derived; we require one such bound in Section 4.

* In the numerical applications below a, b, c, and υ will be assigned values for torus nets, that is, $a = 1/2$, $b = 2$, $c = -2$, $\upsilon = 1/2$.

Note also that the above bound is tight for $m = 1 \Rightarrow E = 0$. As a result, the comparison between the hierarchical and non-hierarchical schemes simply reduces to the comparison of an MHR with $m > 1$ (OBR, CER) to one with $m = 1$ (NCR).

Even with the above simple specifications of the hierarchical adaptive routing scheme, the queueing analysis is still far too complicated for an exact solution. This is true for any adaptive scheme because of its dynamic nature. In the face of these difficulties we make the following assumption:

Assumption 1

a. The performance of adaptive hierarchical routing is the same as that of a deterministic routing policy whose routes satisfy Proposition 8 in [11]. That is, the length of the "fixed routing" paths are equal to the minimum estimated path lengths as obtained with an MHR.

b. With the deterministic routing specified above, and with the class of symmetrical nets considered here, there will be equal loads on all channels.

Assumption 1.a is motivated by our earlier remark on deterministic routing and becomes more accurate when we include the line and storage utilization due to the adaptive routing in the fixed routing model. Moreover, if the main objective is to compare hierarchical with non-hierarchial routing, then this assumption appears to be quite acceptable.

Assumption 1.b is motivated by the highly symmetrical structure of the networks considered here, and also by the fact that the main objective of an adaptive policy is to balance the flows over all the channels in the net.

Note that, due to the above assumption, NCR (MHR with $m = 1$) is modelled by the shortest path fixed routing which, as observed above, leads to the optimal flow assignment for this class of nets.

In summary, a hierarchical routing procedure is characterized by the table length l or equivalently by the degree of clustering m, and by the path length h_c; it can be modelled by a balanced deterministic routing procedure which results in paths of the same length in our symmetric networks.

We now proceed with the performance evaluation of the MHR schemes using our first model.

2.3. *Performance Evaluation with no Updates and no Storage Limitations*

From the above considerations, the delay analysis for our class of symmetrical nets is reduced to the one performed in Section 2.1, except that h is to be replaced by h_c. With the above scaling scheme ($T = T_0$, $C = hC_0$), the ratio of throughputs * with and without clustering is

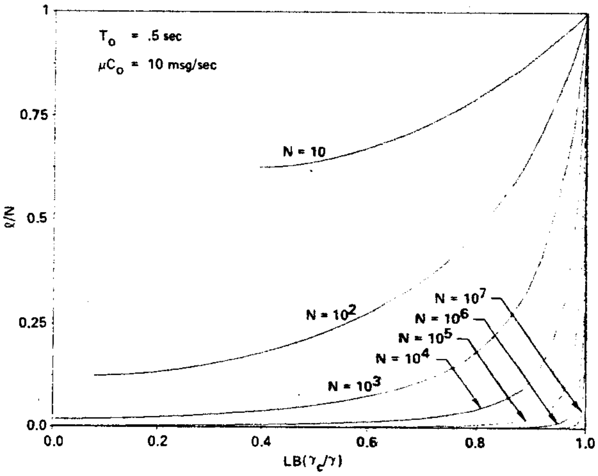


Fig. 1. The Lower Bound on Relative Throughput, LB(γ_c/γ) Versus l/N.

$$\frac{\gamma_c}{\gamma} = \frac{h\mu C_0/h_c - 1/T_0}{\mu C_0 - 1/T_0}\,. \tag{11}$$

From the above expression, the effect of clustering can be seen in the reduction of the line capacity by the fraction h/h_c.

We may now state an asymptotic result similar to that in [11]:

As the number of nodes, N, goes to infinity, the throughput at constant delay obtained with an MHR (CER, OBR) with a fixed m, approaches that of an NCR, while the relative table length l/N, approaches zero, i.e., with significantly less nodal storage and channel capacity requirements.

This result is due to the fact that under these conditions $h_c/h \to 1$ and $l/N \to 0$. For the continuous behavior of γ_c/γ versus l/N, we may apply Eq. (10) to Eq. (11) and obtain the following lower bound:

$$\mathrm{LB}\left(\frac{\gamma_c}{\gamma}\right) \triangleq \frac{(\mu C_0/1 + E) - (1/T_0)}{\mu C_0 - (1/T_0)} \leqslant \frac{\gamma_c}{\gamma} \leqslant 1\,. \tag{12}$$

Fig. 1 illustrates the behavior of LB(γ_c/γ) with respect to l/N for several values of N. In this plane, the optimal operation is obtained in the lower right hand corner where l/N = 0 and $\gamma = \gamma_c$. Similar properties as in [11] are exhibited here. We reemphasize the

* The notation γ_c, h_c and T_c will be used throughout the paper when dealing with clustered routing.

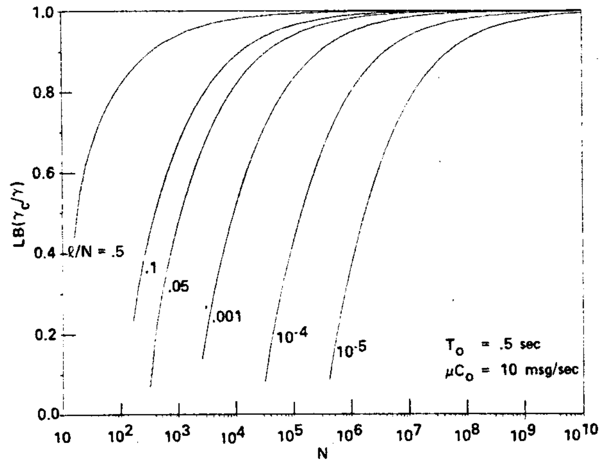


Fig. 2. Degradation in Throughput at Constant Relative Table Length.

fact that substantial table gains * can be obtained with a relatively small degradation in throughput. Larger table reduction may drive the lower bound to zero. The asymptotic property is nicely illustrated in the sharp behavior of the lower bound, for large N.

We note also that the curves in fig. 1 meet at the point $l/N = 1$, $\mathrm{LB}(\gamma_c/\gamma) = 1$. That point corresponds to $m = 1$ where the bound is tight.

Fig. 2 shows the same information in a different way, illustrating the behavior of $\mathrm{LB}(\gamma_c/\gamma)$ with respect to N for several balues of l/N. We see that as N increases, the cost incurred with a fixed relative table length l/N goes to zero.

3. A Queueing Model with Updates and no Storage Limitation

In this section we intend to account for the line traffic generated by the routing updates (in the evaluation of the λ_i's) while keeping the infinite storage assumption.

As noted earlier, the average delay in our class of symmetric nets is very simply related to the average delay at any channel; therefore, we first analyze a single channel and then generalize to a net.

3.1. Priority Model for a Channel

A simple and realistic *H*ead-*o*f-the-*L*ine (HOL) model [10] is considered here, mainly to capture the effects of updates on the average time spent by a data * message waiting to be transmitted on a channel. We assume that updates are originated at regular intervals of time (motivated by the ARPANET procedure). An optimistic approximation to the performance with aperiodic updates is to use a "no update" model (Sections 2 and 4), or a more realistic model would be to use certain distributions governing their generation times. The latter possibility can easily be included if we use the Poisson distribution. Thus, our model for a channel consists of a single queue operated with a HOL priority discipline and the following traffic characteristics.

i *Update traffic:* Deterministic arrival process at rate λ_u. Constant message length $1/\mu_u$ [kilobits/msg].

ii *Data traffic:* Poisson arrival process at rate λ_0. Exponential message length of mean $1/\mu$ [kilobits/msg].

iii *Queue discipline:* HOL preemptive resume between data and update traffic, with a higher priority for updates and FCFS (first-come-first-serve) within each priority.

iv *Channel capacity:* C [KBPS].

The "preemptive resume" assumption in (iii) is introduced to further simplify the analysis of the model.

The above model is slightly different from the usual HOL model which considers the arrival processes of all types of customers (messages) to be governed by a Poisson distribution. However, the methodology can still be used here to approximate the average time in system for a data message.

The update traffic sees a D/D/1 system; hence, as long as $\lambda_u < \mu_u C$ there is no queueing of update messages ($\lambda_u > \mu_u C$ means that more than the total capacity is required by the updates). An arriving data message will incur a delay from any message (data or update) already in service, from all data messages already in the queue and from updates arriving during its time in the system as follows.

Delay due to messages already in queue: Let $\bar{n}$ be the average number of messages in queue as seen by our arriving message. Since data arrivals are Poisson, then $\bar{n}$ is also the average number of messages in the queue at any arbitrary time [10]. Therefore, if t is

* Recall from [11] that most of the table reduction is obtained for small values of m, and the remaining reduction up to the global minimum table length is obtained with a much larger m (where $m \leqslant \ln N$).

* A data message is differentiated from an update (control) message.

the average time in system (queueing + service) for a data message, then because of our previous observation and from Little's result, $\bar{n} = \lambda_0(t - 1/\mu C)$. Because of the exponential distribution of message lengths, the average delay due to customers already in queue is equal to $\bar{n}/\mu C$.

Delay due to the message in service: Let Y be the residual life (remaining service time) of the message in service and let E denote the expectation operator. $E(Y)$ is the contribution to delay that we must evaluate. Conditioning on the type of message in service, it is clear that

$$E(Y) = E[\text{residual life of data message}]\ \frac{\lambda_0}{\mu C}$$

$$+\,E[\text{residual life of update message}]\ \frac{\lambda_u}{\mu_u C}\,.$$

Hence,

$$E(Y) = \frac{\lambda_0}{\mu C}\frac{1}{\mu C} + \frac{\lambda_u}{\mu_u C}\frac{1}{2\mu_u C}\,.$$

Delay due to updates arriving while in system: A data message spends an average time t in the system. During that time, on the average, approximately * $\lambda_u t$ update messages arrive and get serviced; hence the average delay due to these updates is approximately * $\lambda_u t(1/\mu_u C)$.

Finally, summing up all the delays incurred by our arriving message and its own service time, we arrive at the approximation

$$t \cong \frac{(1/\mu C) + (\lambda_u/2[\mu_u C]^2)}{1 - (\lambda_0/\mu C) - (\lambda_u/\mu_u C)} \qquad (13)$$

The above equation exhibits the effect of updates on the message delay over one channel.

3.2. Network Model With Updates

Eq. (13), with λ_0 replaced by λ_i, and Eq. (1) yield the expression for the average delay in the network. It is assumed that all channels receive an equal update rate λ_u.

Eqs. (4) and (5) are still valid when we replace h by h$_c$ for our class of symmetric networks operating with an MHR scheme. Using a similar notation as in Section 2 ($h, h_c, T, T_c, \gamma, \gamma_c$), we arrive at the following throughput-delay relation which characterizes the approximate performance of hierarchical routing.

$$T_c \cong h_c \frac{1 + (\mu/\mu_u)(\lambda_u/2\mu_u C)}{\mu C - h_c(\gamma_c/M) - (\mu/\mu_u)\lambda_u}\,. \qquad (14)$$

Recall that the size of an update message, $1/\mu_u$, is fixed and proportional to the length l ($l = mN^{1/m}$) of the routing table. Hence, it is of the form $1/\mu_u = \epsilon l$ where ϵ is the amount of storage (in kilobits) per table entry. As an example, in the ARPANET, an entry requires 16 bits of storage, hence $\epsilon = 0.016 = 1/62.5$. For further normalization with respect to the average data message, $1/\mu$, we choose ϵ such that $1/\mu_u = \epsilon l/\mu$. (In the numerical examples we choose $1/\mu = 1$ KB and $\epsilon = 1/64$ which closely resembles the ARPANET.)

The behavior of hierarchical routing may now be studied for networks whose growth is governed by our scaling scheme. First we must select the update rate λ_u as a function of the network size N.

3.2.1. Scaling of λ_u

The main purpose of the routing updates is to provide the routing decision algorithm with a good estimate of network congestion. Since at each update, exchange of information (not necessarily synchronized among all nodes) occurs only between neighboring nodes, then the propagation of a change occuring in a certain region of the net to another region requires a number of updates equal to the distance separating the two regions. Consequently, as the network grows and if we wish a change (conveyed through the exchange of updates) to reach remote areas within a reasonable time, then it is necessary to increase the update rate as N increases. We may also argue that the "very" remote areas would not be as concerned with that change as the closer ones; thus the update rate probably need not increase as fast as N. A realistic compromise would consists in the use of higher update rates (as N increases) but only to propagate less and less information about a region as we move away from that region. This remark is again a key motivation behind the hierarchical routing.

From the above considerations emerge three possible specifications for λ_u.

i. $\lambda_u = \lambda_u^0 = \text{constant}$,

ii. $\lambda_u = h\lambda_u^0 = aN^{\upsilon}\lambda_u^0$ ($\frac{1}{2}N^{1/2}\lambda_u^0$ for a torus) ,

iii. $\lambda_u = aN^{\upsilon/2}\lambda_u^0$ ($\frac{1}{2}N^{1/4}\lambda_u^0$ for a torus) .

* We are indebted to T.J. Ott [12] for pointing out that $\lambda_u t$ is not an exact expression for the average number of arrivals: however, we choose to use it as an approximation. This will yield an approximate expression for t in Eq. (13) which in turn will yield approximations in section 3.2 below.

Choice (i) represents the case whereby the update rate is insensitive to a change in network size.

Choice (ii) appears to be more intuitive since the update information needs on the order of h (average path) periods to percolate throughout the net.

Choice (iii) is a compromise between the two above; it indicates that routing information need not percolate as fast in the entire net, but only within a certain area comprising roughly $N^{1/2}$ nodes.

The behavior of hierarchical routing may now be studied for our class of symmetric networks. With our scaling scheme, $C = hC_0$, $T = T_0$, and the network throughput becomes

$$\frac{\gamma_c}{M} \cong \frac{h}{h_c}\,\mu C_0 - \frac{1}{T_0} - \frac{\epsilon l \lambda_u}{h_c} - \frac{\lambda_u}{2\mu C_0 T_0}\,\frac{\epsilon^2 l^2}{h}\,. \qquad (15)$$

For any routing to be feasible, the right hand side of the above equation must be positive. Let us assume λ_u to be of the form $\lambda_u = N^x (0 \leqslant x \leqslant 1)$.

3.2.2. Asymptotic Behavior

As the number of nodes goes to infinity and under the constraint, $0 \leqslant x \leqslant v$, for a hierarchical routing to be feasible, its number of levels m must be greater than $2/(v - x)$, For $x \geqslant v$ there are no feasible hierarchical routing schemes.

The proof follows readily when we replace l by $mN^{1/m}$ and h by $2N^v$ in Eq (15), and we use the property [see 11], that for a fixed m: $N \to \infty \Rightarrow h/h_c \to 1$. Then, if $m > 2/(v - x)$:

$$\lim_{N \to \infty} \frac{\gamma_c}{M} = \mu C_0 - \frac{1}{T_0}\,.$$

The above limiting throughput is equal to that obtained in our previous model where *no* updates were considered. It is a more realistic result because now only a feasible routing $[m > 2/(v - x)]$ can achieve that performance.

Applying the above result to our selected scaling schemes for λ_u, and using the fact that $0 < v \leqslant 1$, we arrive at a value of $2/(v - x)$ (which is greater than or equal to 2) for scheme (i), infinity for scheme (ii) and 4 for scheme (iii). As a consequence, only the first and third schemes (for λ_u) yield a feasible hierarchical routing in the large network limit. Moreover a *non-hierarchical routing (m = 1) is always infeasible* at the limit of very large networks in the sense that for an NCR to be feasible, λ_u must be a *decreasing* function of N.

3.2.3. General Behavior

As in Section 2.1, a lower and upper bound on the throughput can be derived using Eqs. (10) and (15). The expression for the approximated lower bound is

$$\mathrm{LB}\left(\frac{\gamma_c}{M}\right) \triangleq \frac{1}{1+E}\left[\mu C_0 - \frac{\epsilon l \lambda_u}{h}\right] - \frac{1}{T_0}\,\frac{\lambda_u \epsilon^2 l^2}{2\mu C_0 T_0 h}$$

Let us examine the behavior of γ_c/M with respect to N and m (or equivalently l/N) by plotting its bounds normalized by the maximum throughput per channel $\mu C_0 - T_0^{-1}$. The values selected for the different variables are: $\mu C_0 = 6$ msg/sec, $T_0 = 0.5$ sec, $\lambda_u^0 = 0.07\ \mu C_0$, $\epsilon = 1/64$. Recall that E and l are given in Eq. (10).

Figures 3, 4, and 5 illustrate the behavior of the lower bounds on throughput with respect to N for several values of the degree of clustering m. Lower and upper bound envelopes are also plotted in those figures (the upper bound curves themselves have been omitted and only their envelope is shown). The optimal m corresponding to a particular envelope as well as the envelope itself, can be determined numerically. Such an operation can easily be done by hand from the graphs presented here. Given N, choosing m on the lower bound envelope guarantees at least a throughput equal to the corresponding point on the envelope. Equivalently, given N we can determine the optimal table length which leads to the maximum lower bound throughput. This fact is illustrated in fig. 6 for $\lambda_u = \lambda_u^0$.

Note also that the lower bound envelopes for $\lambda_u = \lambda_u^0$, $N^{1/4}\lambda_u^0/2$, (figs. 3, 4) show an initially decreasing and then slowly increasing behavior with respect to N; the increase will eventually bring the curves close to their asymptote 1. However, for $\lambda_u = N^{1/2}\lambda_u^0/2$

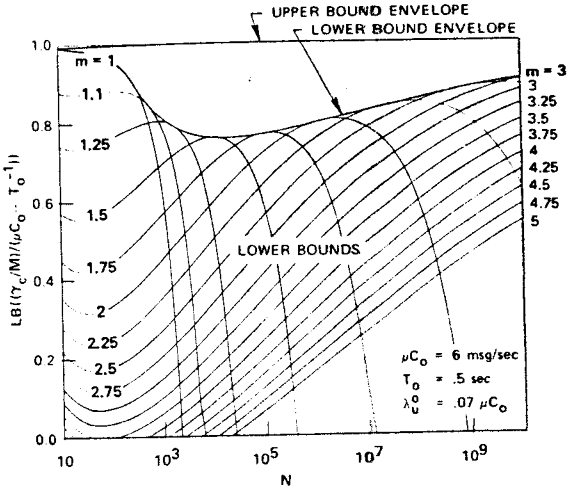


Fig. 3. Throughput at Constant Delay; Model with Updates, $\lambda_u = \lambda_u^0$.

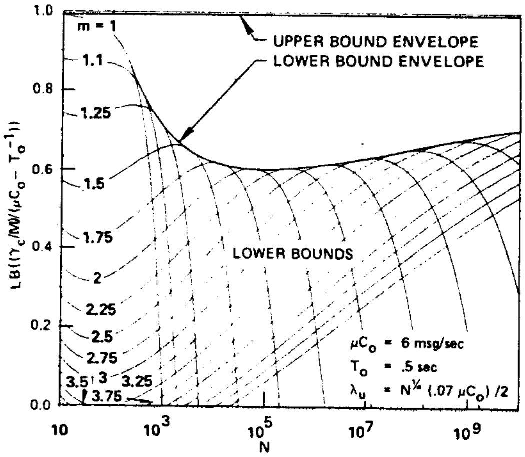


Fig. 4. Throughput at Constant Delay; Model with Updates, $\lambda_u = N^{1/4}\lambda_u^0/2$.

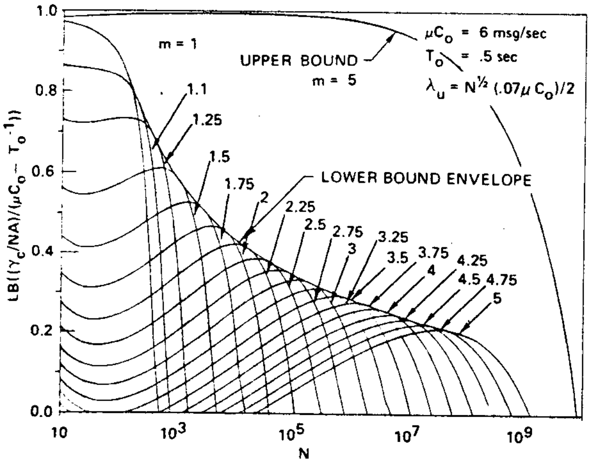


Fig. 5. Throughput at Constant Delay; Model with Updates, $\lambda_u = N^{1/2}\lambda_u^0/2$.

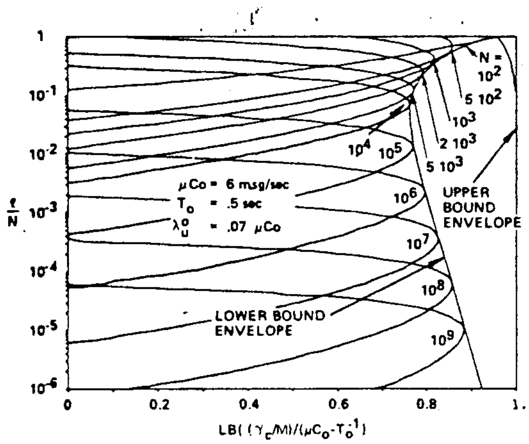


Fig. 6. Lower Bound on Throughput at Constant Delay Versus the Relative Table Length: Model With Updates $\lambda_u = \lambda_u^0$.

Table 1
Critical values of N

	Point at which NCR becomes infeasible ($\lambda = 0, m = 1$)	Point beyond which clustering is better
$\lambda_u = \lambda_u^0$	$N \simeq 2000$	$N \simeq 200$
$\lambda_u = \lambda_u^0 N^{\frac{1}{4}}/2$	$N \simeq 1000$	$N \simeq 150$
$\lambda_u = \lambda_u^0 N^{\frac{1}{2}}/2$	$N \simeq 300$	$N \simeq 100$

(fig. 5), the lower bound envelope is a decreasing function of N which, as predicted earlier, will eventuall reach zero. This means that in the neighborhood of a certain size N, hierarchical routing with this update function altogether becomes infeasible. For our values, that size is well beyond $N = 10^8$.

Finally Table 1 shows the approximate values of the points where a non-hierarchical routing ($m = 1$) becomes infeasible and also shows the points beyond which a 2-level hierarchical routing surely becomes more efficient (based on the lower bound). Note that the faster the update exchange rate λ_u is, the smaller are those critical values of N.

In summary, even though the above study did not account for the storage gains obtained with hierarchical routing, we were able to prove for a class of large networks, that the MHR schemes are very efficient when operated with an optimal table length and that non-hierarchical schemes are infeasible. Next we account for the gains due to nodal storage.

4. A Queueing Model with no Updates and with Storage Limitation

The purpose of this section is to develop and analyze a Kleinrock-like model which takes into consideration the limitation of nodal storage. Based on that model, we will once again study the behavior of hierarchical routing as applied to the class of symmetric networks. We choose not to account for the effect of updates on the line capacity utilization, in order both to model situations where updates can be neglected and, mainly, to isolate the consequences of finite nodal storage on the network performance.

As in the previous section, this study will also demonstrate a remarkable efficiency of hierarchical routing in large networks.

4.1. A Loss-Queueing Model for Symmetric Networks

4.1.1. The Model

Again we consider the class of symmetric networks and, in addition impose a constraint on the number of buffers reserved for the store-and-forward function. As a result of the limited storage, three issues arise: the validity of the exponential message length distribution; the fate of the rejected messages; and the sharing of the pool of S/F (store-and-forward) buffers among the outgoing channels.

With respect to the message length, it is clear that a maximum size must be imposed, as is always the case in practical situations. As an example, in the ARPANET the maximum packet size is equal to 1008 data bits. The ARPANET IMP S/F storage is divided into buffers, each of which can accommodate a maximum size packet (plus header) and cannot be utilized by more than one packet at a time. As a result, one might feel that the assumption of exponentially distributed packets should be replaced with a constant length packet assumption. However, measurements on the ARPANET [9] have shown that the average size of a data message is roughly 250 bits. The fact that the average message length is much smaller than the buffer size, and that messages * which do not fit in a single buffer occur with a very small probability (and hence can be neglected), motivates us to keep the exponential message length assumption. A better approximation perhaps would be to assume a truncated exponential message length distribution, but this makes the analysis much more complicated and no closed form solution has been obtained [1].

As for the rejected messages, we can assume that they are either retransmitted by the sending node (after a time-out, as in the ARPANET) or are lost (as with blocked telephone calls). The retransmission mode is a more realistic assumption in general S/F networks. However, this mode introduces strong dependencies in the stochastic behavior of neighboring (and even more distant) nodes [13] to the point that an analysis seems out of reach. As a result, we restrict our considerations to a loss model, in which case the dependencies between nodes due to storage limitation are eliminated.

Finally, with respect to the sharing of the pool of S/F buffers among the outgoing channels, we assume that accepted messages are first submitted to the routing policy routine and then conceptually occupy one buffer from a common pool of B buffers. The routing decision is assumed to be *fixed* (deterministic) and hence independent of buffer utilization.

Also, as with the independence assumption [8] we will assume that the queue lengths in front of channels in different nodes are stochastically independent and that the input streams at each node are governed by independent Poisson distributions.

As a result of the above considerations and assumptions, the network can be considered as a collection of independent nodes, each of which can be modelled as R M|M|1 queues sharing a waiting room of total size B. The traffic offered at each node is governed by a fixed routing decision, and the probability of blocking at a node is a function of the sharing scheme utilized. Several such schemes have been proposed and studied in [6,7] where, among others, we consider the *C*omplete *S*haring (CS) which is such that an arriving customer is accepted if any storage space is available, independent of the server to which it is directed. We assume the CS scheme in this section.

4.1.2. Analysis

Before we proceed with the analysis we make a few observations. Because of the symmetry of our class of networks, we assume that the fixed routing results in equal loads on each channel. The offered load λ_0 is defined as the input rate of traffic on any given channel before acceptance or rejection by the corresponding node. Moreover, all nodes are assumed to contain the same number of buffers B and to use the same CS sharing strategy. As a result, the probability of blocking (to be denoted by P_B) is the same at all nodes.

Because of the possibility of loss of messages, the offered external traffic γ is no longer equal to the throughput of the network, which we denote by γ_s (s for "successful" traffic). In what follows, we intend to find γ_s and the average delay T, for the successful traffic in a network of this type.

4.1.3. Throughput versus load

γ is now referred to as the traffic load. Let us define P_s as the probability that in steady state, a message transmitted over the network reaches its destination successfully. Clearly

$$P_s = \gamma_s / \gamma \,. \tag{16}$$

Since nodes are assumed independent, the probability that a message is not rejected over k hops is [1 –

* Recall that we are assuming single-packet messages.

$P_B]^k$. We assume that a message in transit is subject to rejection, whereas a message reaching its destination is always accepted.

Let γ^s_{jk} be the rate of successful traffic from node j to node k, and let h_{jk} be the path length between the two nodes, then

$$\gamma^s_{jk} = \gamma_{jk}(1 - P_B)^{h_{jk}} .$$

The sum of γ^s_{jk} is what we have defined above as γ_s. Because of the uniform traffic assumption [$\gamma_{jk} = \gamma_0$, $\gamma = N(N-1)\gamma_0$],

$$\gamma_s = \gamma \sum_j \sum_k (1 - P_B)^{h_{jk}} . \tag{17}$$

Let h be the discrete random variable which represents the distance in hops between a randomly selected pair of nodes, as derived from the fixed routing policy, given a specific network. Also, let $P_r[h = k]$ be the fraction of node-pairs at distance k and let $H(z)$ be the corresponding z-transform, i.e., $H(z) = \Sigma_k z^k \mathrm{P_r}[h = k]$. As a result $H(z)$ characterizes the lengths between pairs of nodes in a particular network. As an example, for a torus network such that $N^{1/2}$ is an odd integer (this condition is irrelevant for a large N) [7],

$$H(z) = \frac{4z(1 - z^{(N^{1/2}+1)/2})(1 - z^{(N^{1/2}-1)/2})}{(N-1)(1-z)^2} . \tag{18}$$

From the above considerations we find

$$\gamma_s/\gamma = P_s = \sum_{k \geq 1} [1 - P_B]^k P_r[h = k] = H(1 - P_B) . \tag{19}$$

4.1.4. Relation between the load γ and the total offered internal traffic λ

λ is now the sum of the *offered* input rates to each of the network channels, so that $\lambda = M\lambda_0$. Eq. (3) ($\lambda = \bar{n}\gamma$) is no longer true due to the possible loss of messages. A similar approach, as used in [8] for the derivation of Eq. (3), is considered here to derive the correct relation between λ and γ in this lossy medium.

The contribution of γ_{st}, the rate of traffic from s to t ($\gamma_{st} = \gamma_0$), to λ is simply

$$\sum_{k=0}^{h_{st}-1} \gamma_{st}(1 - P_B)^k = \frac{1 - (1 - P_B)^{h_{st}}}{P_B} \gamma_{st} .$$

The contribution of all γ_{st} yields the value of λ

$$\lambda = \frac{1 - H(1 - P_B)}{P_B} \gamma = \frac{1 - P_s}{P_B} \gamma . \tag{20}$$

The above relation and Eq. (19) are quite ***general;*** they only assume that all nodes are independent and have an equal probability of blocking P_B. Again, H(z) can be determined analytically or numerically given a particular network and the associated fixed routing policy.

Note that if $P_B = 0$, i.e., infinite nodal storage assumption, Eq. (20) becomes undefined; however, application of L'Hospital's rule results in $\lambda = H'(1)\gamma$ where $H'(z)$ is the derivative of $H(z)$ with respect to z. $H'(1)$ is, in fact, equal to the average network path length; hence we are back to the expression derived in [8] (i.e., $\lambda = \bar{n}\gamma$).

4.1.5. Average Delay of Successful Traffic

Due to the symmetry of our class of nets, a nonrejected message will incur the same delay t at each hop; therefore the average network delay is,

$$T = \bar{n}_s t , \tag{21}$$

where $\bar{n}_s$ is the average path length of the successful traffic. Note, we expect $\bar{n}_s$ to be smaller than h. Intuitively, this is due to the fact that messages which travel on longer paths are more likely to be rejected. The determination of $\bar{n}_s$ follows a derivation similar to that of P_s. From the definition of the average path length [11]

$$\bar{n}_s = \frac{\sum_j \sum_k \gamma^s_{jk} h_{jk}}{\sum_j \sum_k \gamma^s_{jk}}$$

$$= \frac{1}{\gamma_s} \sum_j \sum_k (1 - P_B)^{h_{jk}} \gamma_{jk} h_{jk}$$

Grouping together all paths of length k, and using the definition of H(z), wer arrive at

$$\bar{n}_s = \frac{1 - P_B}{P_s} H'(1 - P_B) = (1 - P_B) \frac{H'(1 - P_B)}{H(1 - P_B)} . \tag{22}$$

Note that if $P_B = 0$, then $\bar{n}_s = h$ [recall that $H'(1) =$ h and $H(1) = 1$].

If we let $P_B \to 1$ and apply l'Hospital's rule to the above equation we arrive at $\bar{n}_s = 1$, [recall that $H(0) = 0$]. This result indicates that in the limit ($P_B \to 1$) only 1-hop traffic can ever be successful.

4.1.6. Probability of Blocking

As noted earlier, a node may be modelled by R M|M|1 queueing systems with a shared finite waiting room of size B. Each server is offered an input rate

$\lambda_0(\lambda_0 = \lambda/M)$, and has a service rate equal to μC. Among the sharing schemes studied in [6,7], Complete Sharing (CS) is optimal when $\rho = \lambda/\mu C$ is small and behaves fairly well for ρ up to a value close to 1. Below, we show that the maximum throughput γ_s is obtained for $\rho < 1$. As a result it seems reasonable for us to choose the complete sharing scheme. However under heavy traffic conditions or if a retransmission mode is used or if the channels receive very unbalanced traffic loads, then a different scheme may be necessary.

The analysis of the complete sharing scheme leads to an expression of P_B in terms of $\lambda_0/\mu C$ which, when combined with Eq. (20), results in the system of equations below.

$$\lambda_0 = \frac{1 - H[1 - P_B]}{P_B} \frac{\gamma}{M} \qquad \text{(a)}$$

$$P_B = \frac{\left(\frac{B + R - 1}{R - 1}\right)\left(\frac{\lambda_0}{\mu C}\right)^B}{\sum_{K=0}^{B} \left(\frac{K + R - 1}{R - 1}\right)\left(\frac{\lambda_0}{\mu C}\right)^K} \qquad \text{(b)} \tag{23}$$

The average delay in the network is then

$$T = \overline{n}_s \left[1/\mu C \sum_{K=0}^{B-1} \left(\frac{K + R - 1}{R - 1}\right)\left(\frac{\lambda_0}{\mu C}\right)^K - \left(\frac{B + R - 1}{R}\right) \right.$$

$$\left. \times \left(\frac{\lambda_0}{\mu C}\right)^B \quad 1 - \lambda_0/\mu C \sum_{K=0}^{B-1} \left(\frac{K + R - 1}{R - 1}\right)\left(\frac{\lambda_0}{\mu C}\right)^K \right]^{-1} \tag{24}$$

From Eq. (23, we have two relations between λ_0 and P_B. The first relation (a) shows that λ_0 is a monotonic decreasing function of P_B whereas the second relation (b) shows that P_B is a monotonic increasing function of λ_0. As a result, there exists a unique solution for λ_0 and P_B which can be determined using any converging iterative procedure..One such procedure is given in [7].

The limiting throughput of the network obtained with an infinite traffic load is

$$\lim_{\gamma \to \infty} \gamma_s = \frac{B}{B + R - 1} H'(0)\, M\mu C\, . \tag{25}$$

This limiting result has the following simple interpretation. First, $H'(0) = P_r\,[h = 1]$ is the fraction of node pairs at distance one (i.e., neighboring nodes), and $B\mu C/(B + R - 1)$ is the limiting throughput of any channel of the R M|M|1 system of queues (with a CS scheme). As a result, the limiting throughput represents the fraction of successful traffic which has to travel over a single hop. The other (finite) fraction of initially successful traffic has to travel over at least one other hop; in trying to do so, it will compete with an infinite amount of traffic generated at the next node, and thus it will be rejected. This checks with the previous result: $P_B \to 1 \Rightarrow \overline{n}_s \to 1$.

4.1.7. Application of the Loss Model to Torus Networks

Let us consider a torus (i.e., $R = 4$) operating with a fixed shortest path routing whose z-transform is given by Eq. (18). Of interest is the study of the behavior of the successful traffic λ_s with respect to the load γ, as well as the behavior of the delay T and the probability of loss $1 - P_s$.

Numerical results are shown in Figs. 7, 8 and 9. These results were obtained for $N = 121$, and $\mu C = 20$ msg/sec. More precisely, the graphs show the normalized traffic and delay and loss probability. The normalization is based on Eq. (7) which defines the utilization of the net, and on Eq. (6) for the delay.

Recall that $h = \frac{1}{2}N^{1/2}/2$ for a torus (with $N^{1/2}$ an odd integer). Fig. 7 shows that as γ increases, γ_s increases to a maximum value and then decreases to its limiting value in Eq. (25). These results are similar to that of a contention system, except that the non-retransmission (loss) of rejected messages eliminates the possibility of unstable states.

Note that if $B = \infty$ then γ_s is equal to γ for ρ varying from 0 to 1. For $\rho \geqslant 1$ a steady state solution does not exist; this is no longer true for a finite buffer size. However, with limited storage, as ρ increases beyond 1, the throughput decreases quite a bit! Another effect of finite storage is reflected in the behavior of the average delay T which asymptotically reaches a constant value as ρ goes to infinity. For $\rho \to \infty$ (i.e., $\gamma \to \infty$) only 1-hop traffic can be successful [see Eq. (25)], and then the asymptotic value of T corresponds to the delay on one hop (i.e., at one node) under the condition of an infinite input rate. From [7], that value for one node is

$$\lim_{\rho \to \infty} \mu Ct = B/(B + R - 1)\, ;$$

thus

$$\lim_{\rho \to \infty} \mu CT/h = B/[(B + R - 1)\, h]\, .$$

As the number of buffers B increases, the plots

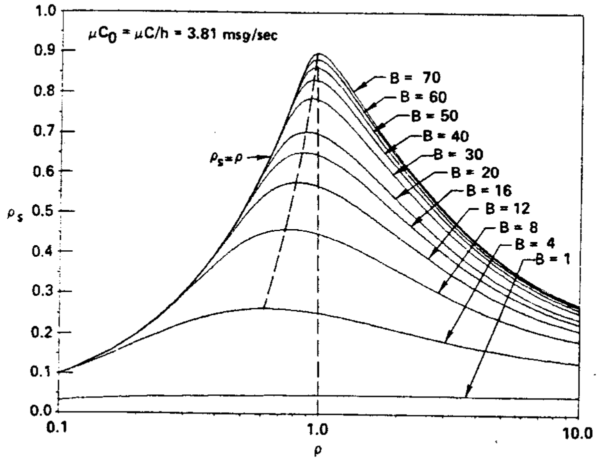


Fig. 7. Normalized Throughput ρ_s, Versus Normalized Load ρ, for a 121 Node Torus with Storage Limitation.

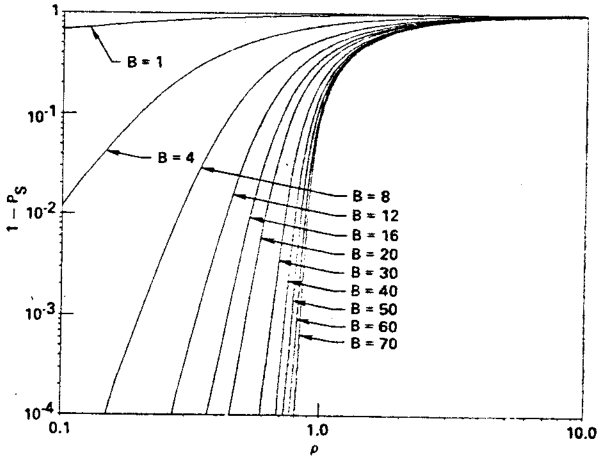


Fig. 9. Probability of Loss for a 121-Node Torus with Storage Limitation.

show substantial improvement in maximum throughput in the region of small $B (B \leqslant 30)$, whereas as B gets large the improvement becomes less significant. The maximum throughput asymptotically reaches 1 as B goes to infinity. This phenomenon is clearly shown in Fig. 9 where the probability of loss, $1 - P_s$, is plotted versus the normalized load for different values of B. Furthermore, a larger B improves the network throughput but yields larger network delays (see Fig. 8).

In summary, we are now able to evaluate the importance of buffer storage, and as expected, small values of B can degrade the network performance significantly. For the example above, at $\rho = 1$, B = 20 reduces the throughput to roughly 0.68 of that which could otherwise be obtained with an infinite number of buffers.

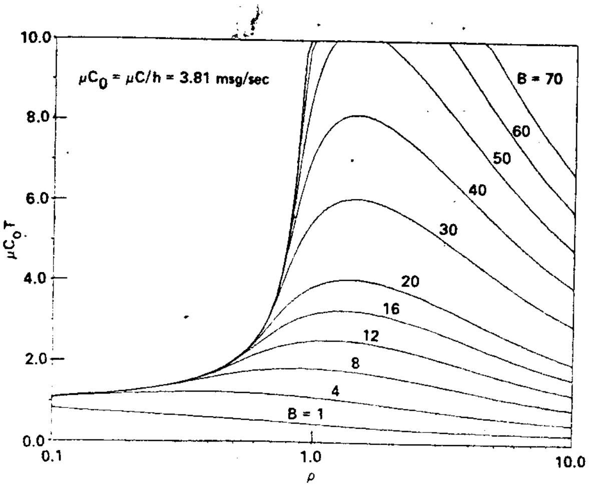


Fig. 8. Normalized Delay $\mu C_0 \bar{T}$ verus Load ρ, for a 121 Node Torus with Storage Limitation.

4.2. Performance Evaluation of Hierarchical Routing

The above model and Assumption 1 will now allow us to study the behavior of hierarchical routing for the class of symmetric nets. We must, however, extend our scaling schemes to scale the nodal storage (B).

4.2.1. A Buffer Scaling Scheme

As seen in the conclusion above, the effect of B is naturally characterized by the maximum throughput γ_s, or, using our normalized notation, it is characterized by the maximum of ρ_s. Under the conditions of Section 2.1, the scaling scheme maintains ρ constant as N varies. It is now natural to attempt to keep the maximum ρ_s constant. With this objective in mind, let us observe the effect of storage limitation first in a single node situation and then in a network environment. From Eq. (23) we see that P_B depends only on $\lambda_0/\mu C$ and that constant B will result in a constant P_B. In a network environment, keeping P_B constant will still result in a smaller probability of success P_s as the network grows. This comes about because of the increase in network path length (aN^v) which results from a larger N. It is then necessary to increase B with N, in order to maintain max ρ_s constant.

An ad-hoc (heuristic) scaling scheme for B has been devised which, as we will see, satisfies our needs over a large range. Such a scheme is

$$B = \lceil B_0 \ln h \rceil \qquad (26)$$

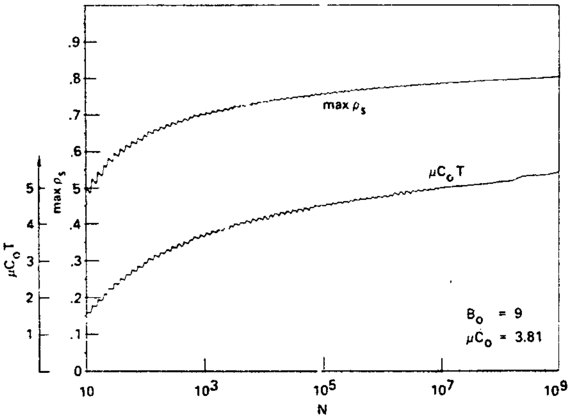


Fig. 10. Buffer Scaling: Normalized Delay and Maximum Throughput.

This scheme combined with the scaling of $C (C = hC_0)$ has been tested on the torus nets as shown in Fig. 10 where the maximum ρ_s and the corresponding delay are plotted versus the network size N. The maximum ρ_s is computed numerically using a Fibonacci search [13].

The curves show that our scaling scheme is quite acceptable, especially when $N > 1000$. For that range of N, the maxima of ρ_s lie between 0.7 and 0.8. Further graphs showing the probability of success P_s, and the average path length of successful traffic normalized with h, $\overline{n}_s/h$, at those maximum points are available in [7]; both functions increase with N to values close to 1 (for $N = 10^9, \overline{n}_s/h \simeq 1, \rho_s \simeq 0.95$).

4.2.2. Behavior of Hierarchical Routing

Recall that Assumption 1 led us to consider deterministic routing to model an adaptive hierarchical routing. If we know the distribution of the path length of the equivalent routing, we can use the loss model and scaling scheme to predict the behavior of such routing schemes as m and N vary. Once again, we use the bounds derived in [7,11] to characterize the message path lengths which result from hierarchical routing.

4.2.3. Distribution of Path Length

Since the distribution of path length deals with paths on a node-pair basis, we can no longer use the bound E, which was only valid for the average distance. Instead we use the more general bound on individual paths (to be denoted by Δ)

$$h_{st}^{\mathrm{c}} - h_{st} \leqslant \Delta = \sum_{k=1}^{m-1} d_k \;\forall\, s, t \text{ network nodes}$$

where d_k is an upper bound on the diameter of any kth level cluster [11]. However this bound, always true for CER, is only valid with OBR if s and t belong to clusters at lower levels than the mth level cluster (entire net). The fact that first, Δ is a very generous bound, and second, that we expect OBR to behave better than CER, lends credence to our use of Δ as an approximation on paths for the OBR scheme.

As a result of the above considerations, the conclusions reached below are *rigorous* for the CER scheme, which is then sufficient to prove the efficiency of hierarchical routing in large networks.

For our class of networks and for an optimal clustering of degree m, we have

$$\Delta = b\frac{N^{\nu} - N^{\nu/m}}{N^{\nu/m} - 1} + c(m - 1) .$$

Note again that $m = 1 \Rightarrow \Delta = 0$.

A "best case" and a "worst case" distribution can now be defined if we assume respectively that $h_{st}^{\mathrm{c}} = h_{st}$ and $h_{st}^{\mathrm{c}} = h_{st} + \Delta$.

The "best case" distribution corresponds to the z-transform H(z) of the shortest paths in the network.

The z-transform $H_{\mathrm{c}}(z)$ of the worst case distribution can be found by observing that the distribution of h_{st}^{c} is simply a shift of h_{st} by an amount Δ at most. Thus $H_{\mathrm{c}}(z) = z^{\Delta}H(z)$.

Since $m = 1 \Rightarrow \Delta = 0$, we also have that $m = 1 \Rightarrow H_{\mathrm{c}}(z) = H(z)$. This fortunate property again allows us to compare a lower bound performance of the MHR's with the exact performance (within our model assumptions) of a non-hierarchical scheme.

4.2.4. Buffer Assignment and Feasibility

Recall that in this study we intend to account for the storage utilized by the routing tables. The size of such storage is a linear function of the table length and counted in number of buffers it is equal to

$$\lceil \epsilon_2 l \rceil$$

where $1/\epsilon_2$ is the number of entries which fit in one buffer (in the numerical applications below ϵ_2 is chosen equal to 1/64). As a consequence, if the total number of buffers to be shared between the routing table and the S/F function is as defined in Eq. (26), then the number of buffers strictly reseved for the S/F function is

$$B = \lceil B_0 \ln h \rceil - \lceil \epsilon_2 l \rceil .$$

With an optimal clustering of degree m, and for our class of symmetrical nets ($h = aN^{\nu}$), the above equation becomes

$$B = \lceil B_0(\ln a + v \ln N)\rceil - \lceil \epsilon_2 m N^{1/m}\rceil . \qquad (27)$$

For a hierarchical routing to be feasible, B must be greater than or equal to one; we conclude that:

i. For a fixed m, the routing becomes infeasible for networks of size larger than a critical number N_c. N_c is the solution of $B = 0$ in Eq. (27) and it obviously is an increasing function of m.

ii. For very large networks, and under the condition $B_0 v \geqslant \epsilon_2 e$, only a hierarchical routing operating with a globally minimized table length is feasible, i.e., $m = m_* \triangleq \ln N$; hence $l = mN^{1/m} = e \ln N$.

In summary, as N gets larger, it becomes imperative to move toward more clustering, eventually reaching a globally minimum table length. The decision to use a higher degree of clustering m should be weighed against the degradation incurred by the corresponding increase in network path length. This phenomenon is illustrated in the numerical application below.

Numerical Application

From the above considerations, the application of the loss model to a network operated with a hierarchical routing, results in the evaluation of lower and upper bounds on the network throughput.

The lower bound performance is characterized by

$$\gamma_s = H_c(1 - P_B)\gamma . \qquad (28)$$

The probability of blocking P_B is the solution of Eq. (23) where H is replaced by H_c. With respect to the delay T, let $\overline{n_s^c}$ be the average path length of the successful traffic. Replacing $H(z)$ by $H_c(z)$ in Eq. (22) we arrive at $\overline{n_s^c} = \overline{n}_s + \Delta$; therefore, $T = (\overline{n}_s + \Delta)\ t$ instead of Eq. (24).

The upper bound performance is obtained by setting $\Delta = 0$.

In [7], we first evaluated the performance of an MHR as applied to a network of size $N = 1681$. We found that if we limit our considerations to an operational range of ρ (i.e., ρ is only allowed to vary from 0 up to a value sligthly larger than the one producing the maxima ρ_s, roughly $0 \leqslant \rho \leqslant 1$), then the value of m which leads to maximum ρ_s also leads to the best performance over that entire range. As a consequence we restrict our observations below to the behavior of max ρ_5 (lower or upper bound) with respect to N and for a set of values of m. We take max ρ_s as the measure of performance of interest to computer network design.

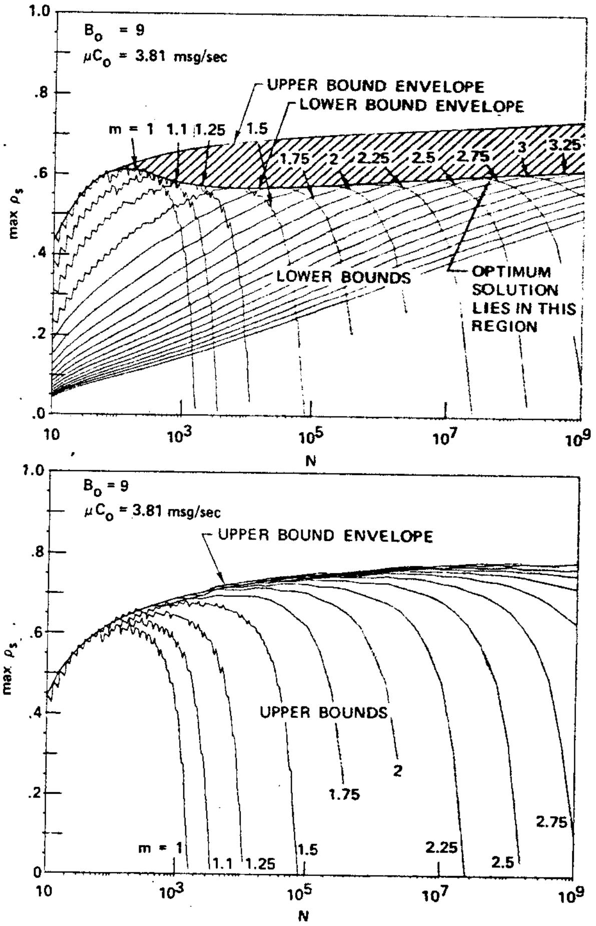


Fig. 11. Maximum Throughput Obtained with the Hierarchical Routing Model with No Updates and With Storage Limitation.

Fig. 11 illustrates the behavior of the maximum normalized throughput as obtained from the lower and upper bound considerations [Eq. (28)], with respect to N for several values of m. Lower and upper bound envelopes are also plotted.

A few remarks emerge from the observation of Fig. 11. These remarks are, in general, quite similar to the ones stated at the end of Section 3, namely with regard to the optimal degree of clustering (m) for a given N, and the feasibility and efficiency of hierarchical routing. Before we proceed, let us note that the jagged nature of the curves is due to the discrete changes of B [see Eq. (27)]. This fact is more accentuated for smaller values of N where B is small, and consequently a change of one unit is relatively noticeable. Round-off errors, as well as errors due to our numerical algorithms for finding P_B and especially max ρ_s (Fibonacci search), also contribute to this jagged behavior.

Hierarchical routing with an appropriate degree of clustering, m (equivalently with an appropriate table length), guarantees that the max ρ_s (with respect to N) will lie between the upper and lower bound envelopes. It is quite remarkable that the lower bound envelope remains relatively flat (around 0.6), for N beyond one hundred. Moreover, the upper bound envelope is very close to the curves obtain in Fig. 10 where we assume no storage was required by the updates. This means that at the point (N, m) corresponding to those envelopes, the storage required by the updates is relatively negligible. As a consequence, the gap existing between the lower and upper bound envelopes is mainly caused by the increase in path length Δ.

Finally, let us note that the performance of a non-hierarchical routing (this is the $m = 1$ curve in both parts of Fig. 11; note these are the same curves in both parts since the $m = 1$ analysis is exact) deteriorates very rapidly for values of N around 1000, and that for N greater than roughly 250, hierarchical routing clearly becomes superior.

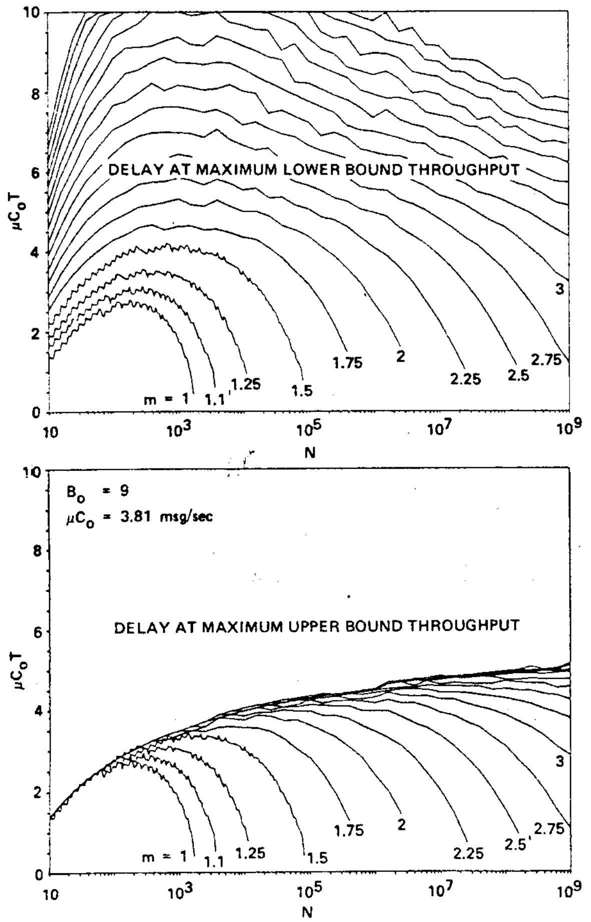


Fig. 12. Network Delay at Maximum Throughput with the MHR.

Fig. 12 illustrates the behavior of the normalized delay at the maximum points of Fig. 11 with respect to N. The curves in the upper half of the figure represent the delay at the maximum throughput as obtained from the lower bound considerations, while the curves in the lower half correspond to the upper bound considerations. Equivalently, the curves show upper (top curves) and lower (bottom curves) bounds on delay at maximum throughput for a given network size N and degree of clustering m (i.e., for a given table length $l = mN^{1/m}$).

5. A Queueing Model with Updates and Storage Limitation

In this section, we apply our previous results in order to devise a model whereby we account *both* for line capacity and nodal storage used for routing.

The R M|M|1 single node model with a finite number of buffers B must now be modified in order to account for the updates. Because of the results in Section 3.1, a channel can be modeled by an HOL priority queue (M|M|1, D|D|1). This is, however a major obstacle in an analytical solution. A more careful observation of the approximate analysis of the HOL system [Eq. (13)] shows that the effect of the updates is primarily to reduce the line capacity available for data traffic from C to $(1 - \rho_u)\ C$. A secondary effect is the added term $\lambda_u/2(\mu_u C)^2$ in the numerator of Eq. (13); we neglect this term in our simplified model here.

Moreover, we will assume that the handling of updates utilizes some storage (working storage) other than the S/F area.

As a result of the above considerations, our study here is now reduced to the one performed in Section 4, where C is to be replaced by $(1 - \rho_u)\ C$. A numerical application in [7] assumed the same environment as the one used to obtain Fig. 11 and $\lambda_u = N^{1/4}\lambda_u^0/2$, where $\lambda_u^0 = 0.14\mu C_0$. The results were found to be quite similar to those in Figs. 11 and 12. The effect of updates was seen in the drop of the maximum normalized throughput (lower bound) by roughly 0.05, except for very large N's where the drop became very small. In addition, hierarchical routing was superior to non-hierarchical routing for N at around 180 instead of the previous 250.

6. Summary

In this paper we demonstrated the following for a class of symmetric distributed networks:

i. In an ideal situation of sufficient storage and line capacity, it is no surprise that the performance of non-hierarchical routing is, in general, better than that obtained with hierarchical routing. However, in the limit of very large nets they become quite comparable! Moreover the hierarchical system gives enormous table reductions (and this provides very significant savings in nodal storage and line capacity).

ii. With a more realistic situation and with reasonable assumptions on network growth, hierarchical routing becomes not only a necessity for large nets, but also it preseves a remarkably good network performance for a phenomenal range of network sizes.

The particular numerical examples studied in this paper show that the transition point where hierarchical routing surely becomes better than a non-hierarchial one, occurs for relatively small N (between 100 and 200).

Indeed, for a variety of performance and economic reasons, we observe that the new public and private packet switched networks (e.g., TELENET) are hierarchical in structure even for 50 to 100 nodes.

References

[1] W. Chu, A Study of ansychronous time division multiplexing for time-sharing computer systems, AFIPS Conference Proceedings, (FJCC, Las Vegas, Nevada, 1969) 35, 669–678.

[2] G. Fultz, Adaptive routing techniques for message switching computer-communication networks, UCLA-ENG-7252, School of Engineering and Applid Science, University of California, Los Angeles (July 1972).

[3] M. Gerla, The design of store-and-forward (S/F) networks for computer communications, UCLA-ENG-7319, School of Engineering and Applied Science, University of California, Los Angeles (January 1973).

[4] J. Jackson, Networks of waiting lines, Operations Research 5 (1957) 518–521.

[5] R. Kahn and W. Crowther, A study of the ARPA computer network design and performance, Report No. 2161, Bolt Beranek and Newman Inc., Cambridge, Mass. (1971).

[6] F. Kamoun and L. Kleinrock, Analysis of shared storage in a computer network environment, Proc. of the 9th Hawaii Int'l. Conf. on System Sciences, Honolulu (1976).

[7] F. Kamoun, Design considerations for large computer communication networks, UCLA-ENG-7642, School of Engineering and Applied Science, University of California, Los Angeles (April 1976).

[8] L. Kleinrock, Communication Nets; Stochastic Message Flow and Delay, (McGraw-Hill, New York 1964, out of print; Reprinted by Dover Publications, New York, 1972).

[9] L. Kleinrock and W. Naylor, On measured behavior of the ARPA network, AFIPS Conference Proceedings (NCC, Chicago, Illinois, 1974) 43, 767–780.

[10] L. Kleinrock, Queueing Systems, Vol. II: Computer Applications (Wiley Interscience, New York, 1976).

[11] L. Kleinrock and F. Kamoun, Hierarchical routing for for large networks; performance evaluation and optimization, Computer Networks 1 (1977) 155–174.

[12] T.J. Ott, private communication, December 12, 1979.

[13] W. Zangwill, Nonlinear Programming, A Unified Approach, (Prentice-Hall, Englewood Cliffs, N.J., 1969).

[14] J. Ziegler, Nodal blocking in large networks, School of Engineering and Applied Science, UCLA-ENG-7167, University of California, Los Angeles (1971).