Hierarchical routing for large networks

Performance evaluation and optimization

Leonard Kleinrock and Farouk Kamoun
Computer Science Department, University of California, Los Angeles, CA 90024, U.S.A.

Distributed adaptive routing has proven to be useful in packet switching networks. However, the storage and updating cost of this routing procedure becomes prohibitive as the number of nodes in the network gets large. This paper deals with the specification, analysis and evaluation of some hierarchical routing procedures which are effective for large store-and-forward packet-switched computer networks. The procedures studied are an extension of present techniques and rely on a hierarchical clustering of the network nodes. In particular, optimal clustering structures are determined so as to minimize the length of the routing tables required. A price for reducing the table length is the increase in the average message path length in the network. Bounds are derived to evaluate the maximum increase in path length for a given table length. From this we obtain our key result, namely, that in the limit of a very large network, enormous table reduction may be achieved with essentially no increase in network path length.

Keywords: Packet switching, networks, computer networks, large networks, data networks, hierarchical design, routing, area routing, adaptive routing, clustering, partitioning.

Computer Networks 1 (1977) 155–174

1. Introduction

Computer networks offer large economies through resource sharing. Among such resources we include specialized hardware, specialized software and data banks. These distributed computer communication systems made their first appearance in the form of packet switching with the ARPANET [2,5,12,17, 27]. The first commercial data carrier, TELENET [29], is already operational. The basis of this demand for computer networks is the ever increasing need for computer and data communication power.

Communication among the network resources is accomplished by the communication subnetwork. This includes the hardware and software specifically dedicated to the transfer of data from node to node. Many alternative communication schemes can be implemented at the subnet level. Among these are: circuit switching [26], packet switching (a form of

Leonard Kleinrock is Professor of Computer Science at the University of California, Los Angeles. He received his B.E.E. at the City College of New York in 1957 and his M.S.E.E. and Ph.D.E.E. at the Massachusetts Institute of Technology in 1959 and 1963 respectively. In 1963 he joined the faculty of the School of Engineering and Applied Science at the University of California, Los Angeles. 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 large group in advanced teleprocessing systems and computer networks.

He serves as 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 was awarded a Guggenheim Fellowship for 1971–1972 and is an IEEE Fellow "for contributions in computer-communication networks, queueing theory, time-shared systems, and engineering education."

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 DAHC 15-73-C-0368.

store-and-forward communication) [16,18] radio broadcasting [1], satellite communication [20], or any combination of the above, etc.

The selection of the best switching scheme is a difficult problem and depends very much on the nature of the traffic to be handled by the network [3,4,24]. The bursty nature of computer traffic, as well as the continuously decreasing cost of computer hardware [28], very much favor packet switching as the technology to employ.

The basic concepts for and the first implementation of a packet switching computer network were developed by the United States Department of Defense Advanced Research Projects Agency (ARPA). This network (the ARPANET), in operation since 1969, has been an enormously successful demonstration of the packet switching technique. It has resulted in the development of a multitude of other networks throughout the world (EPSS in England, CYCLADES and TRANSPAC in France, DATAPAC in Canada, EIN in Europ, TELENET and AUTODIN II in the USA, etc.)

Present computer networks may be characterized as small to moderate in size (57 nodes for the ARPANET as of December 1975). Predictions indicate that, in fact, large networks of the order of hundreds (or even possibly thousands) of nodes are soon to come.

In the course of developing the ARPANET, a design methodology has evolved which is quite suitable for the efficient design of small and moderate sized networks [6,8,18]. Unfortunately the cost of conducting the design is prohibitive if these same techniques are extrapolated to the case of large networks [14]. Indeed, not only does the cost of design grow exponentially with the network size, but also the cost of a straightforward adaptive routing procedure becomes prohibitive. Other design and operational procedures (routing techniques) must be found which handle the large network case. Our main objective in this paper is to specify and evaluate routing policies for LARGE networks.

Routing for packet switching networks

In a packet switching network, messages are partitioned into a number of small segments called packets which then are transmitted through the network using store-and-forward switching. That is, a packet traveling from source S to destination D is received and "stored" in queue at any intermediate node K while awaiting transmission, and is then sent "forward" to node P, the next node on the route from S to D, when channel (K,P) permits.

The selection of the next node P is made by a well-defined decision rule referred to as the routing policy. Several classification schemes have been devised to characterize routing policies [16,7,9,21,22]. Generally speaking, routing policies may be divided into two main classes: deterministic and adaptive. While deterministic routing is more attractive to use at the design phase, adaptive policies are essential for the successful operation of real networks.

The major goal of an adaptive routing procedure is to sense changes in the traffic distribution and network status and then to route messages such that the congested or damaged areas of the network are avoided. It is very important for those procedures to adapt to line and node failures in order to maintain a good grade of service for the network. Such policies base their decisions on measured values, at given times, of a set of time varying quantities (number of messages enqueued, number of hops, etc.) which describe the salient features of the state of the network (traffic, topology, etc.). Such information is referred to as routing information. A central node could provide the routing information (yielding *centralized* control) and distribute it to all nodes in the network, or the nodes could collaborate in computing the routing information directly (yielding *distributed* control) [16,7,13].

In any case, routing information must be stored in tables at each node and is used to identify the output line for each destination. [1] More detailed classifications of the routing policies can be found in [7,10, 22]. In this study, we limit our considerations to the most commonly used adaptive routing policies, namely, distributed routing policies. These policies base their decisions on routing information contained in routing tables individually maintained at each node. The tables are updated periodically or asynchronously or a combination of both [7] using routing information collected internally and provided from neighboring nodes. Such a scheme is used to operate the ARPANET [22].

Typically, in a network with N nodes, each node ("IMP" in the ARPANET terminology) i ($i = 1, 2, ..., N$) has a *R*outing *T*able (to be denoted by RT) which is composed of N entries. Each entry, say k, is subdivided into three (or more) fields. The "delay"

[1] We do not consider the case where packets carry their own routing information.

field indicates the estimated minimal delay from node i to destination node k. The "next-node" field indicates the next node a message must be forwarded to on its way to node k, along the estimated minimal delay path. The "hop" field represents the minimum number of line hops to node k. The purpose of the hop-field is to allow the detection of node failures in the network.

Each node periodically (for example every 0.64 sec in the ARPANET, for a heavily loaded 50 kilobit per sec line) sends and receives update messages from neighboring nodes; these updates need not be synchronized among nodes. Upon reception of an update, a node updates its own routing table, using the delays measured on its output lines and the delay information found in the update message. An example of an updating rule is provided in Section 4.2.

To summarize, we see that, fundamental to the operation of the distributed adaptive routing schemes is the storage, maintenance, propagation and updating of routing tables. Also, it is important to note that in such schemes, the routing tables apparently must contain a number of entries equal to the number of nodes in the network.

Since the length of the routing table (which directs the traffic through each node) will grow linearly (one entry per node) with the number of nodes, we see that for large computer networks (on the order of many thousands of nodes) the storage required to contain this list in each node will be extremely costly. Also, as a direct consequence of these large table lengths, the cost of interchanging routing information among the network nodes will also grow and will represent a significant burden on the communication lines themselves. All these considerations suggest that some form of reduction of the routing table length is called for. Below we present and study some schemes which achieve this goal. Fultz [7] and McQuillan [22] proposed similar schemes but did not evaluate their performance as we do here.

2. Hierarchical routing schemes

The main idea for reducing the routing table length is to keep, at any node, complete routing information about nodes which are close to it (in terms of a hop distance or some other nearness measure), and lesser information about nodes located further away from it. This can be realized by providing one entry per destination for the closer nodes, and one entry per *set* of destinations for the remote nodes. The size of this set may increase with the distance.[2]

For routing in large networks the reduction of routing information is realized through a hierarchical clustering of the network nodes.

In what follows we first introduce and specify hierarchical routing schemes and their underlying clustering structures. We then observe that non-optimally selected clustering structures may lead to very little table reduction. As a result, it is important to find optimal structures. This we do by solving an optimization problem whose objective is to minimize the table length. The optimal solution is found to achieve significant table reductions. The ratio l/N, of the new table length l, to the one obtained with no clustering N, constitutes, in this paper, the unique performance measure by which we characterize the gains obtained from the hierarchical routing. In reality, one needs to express those gains in terms of recovered nodal storage, line capacity, CPU processing, and ultimately in terms of network throughput and delay [14]. These last we defer to a forthcoming paper [15].

Unfortunately, the gains in table length are accompanied with an increase of the message path length in the network. This results in a degradation of network performance (delay-throughput) due to the excess internal traffic caused by longer path lengths. Again we defer throughput-delay considerations [14] to a later paper, and restrict our study here to the evaluation of the increase in network path length. After further specifications and characterization of the hierarchical schemes, bounds are derived to evaluate the maximum increase in path length for a given table reduction. The bounds demonstrate 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 similar to present schemes with very substantial savings in storage and capacity. Finally, we examine the behavior of these bounds with respect to the relative table length l/N.

We now proceed with the description of the hier-

[2] A similar concept underlies the mechanisms of large information systems with pyramidal structures in which information is more and more aggregated as we move up to the higher levels in the hierarchical organization. Aggregation of information or variables is commonly introduced when dealing with large systems [23,30].

archical routing schemes. Recall that the main objective of such schemes is to operate with smaller table lengths. The reduction of routing table length is achieved through a hierarchical partitioning of the network. Basically, an m-level *H*ierachical *C*lustering (mHC) of a set of nodes consists of grouping the nodes (which we shall define as 0th level clusters) into 1st level clusters, which in turn are grouped into 2nd level clusters, etc. This operation continues in a bottom up fashion, finally grouping the $m-2$nd level clusters into $m-1$st level clusters whose union constitutes the mth level cluster. The mth level cluster is the highest level cluster and as such it includes all the nodes of the network. The mHC will be described more formally below.

Since hierarchical routing schemes are based on an m-level hierarchical clustering, they will be denoted as mHR schemes. With the mHR schemes, only *one* entry in the routing table, at any node, say i, is provided for each node in the same 1st level cluster as i, and for each 1st level cluster (a set of nodes) in the same 2nd level cluster as i, and in general for each $k-1$st level cluster in the same kth level cluster as i $(k = 1,2, ..., m)$. The structure of this scheme can best be understood by an example. Fig. 1 shows a 3-level hierarchical clustering imposed on a 24 node network. The clustering leads to the tree representation shown in Fig. 2, where nodes are identified using the Dewey notation [19]. To each node we now associate a reduced routing table. Fig. 3 shows the layout of node 1.1.1's routing table; the number of entries is now 10 (instead of 24 without clustering). As an example, the routing of a packet from node 1.1.1 to node 3.2.2 may proceed as follows: Node 1.1.1 recognizes, from the address of the destination node *3*.2.2, that it has to use entry 3, of the 2nd level cluster entries, to decide upon the next node to which the packet must be forwarded. When the packet reaches a node, say 3.1.1, in the 2nd level cluster 3, then that node will in turn use the second entry (3.*2*.2) among the 1st level cluster entries. Finally, when the packet enters the destination cluster, 3.2, the routing will be done using 0th level cluster entry, number 2 (3.2.*2*). (Note that it was assumed that the mHC results in connected subgraphs.)

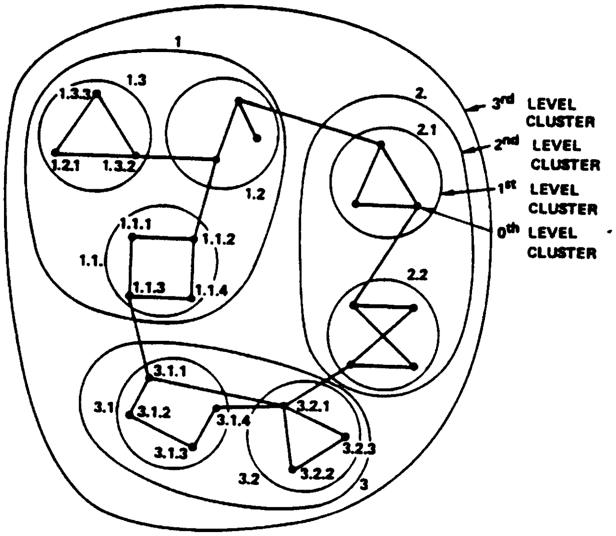

Fig. 1. A 3-level clustered 24-node network.

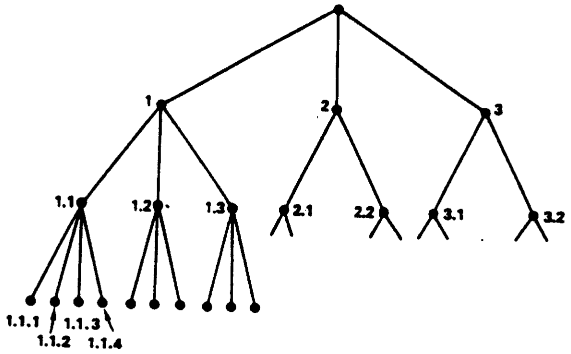

Fig. 2. A tree representation of a 3-level clustered net.

Two remarks emerge from the above considerations.

1. The length of the RT at any node is strictly a function of the clustering structure, i.e., it is a function of the number of nodes per cluster, number of clusters per supercluster, etc., and the number of levels. In what follows, in order to simplify the manipulation and implementation of the RT's in the network, we *assume that equal length tables* are provided at all nodes. Consequently, if l is that length, it must accommodate the number of entries in the RT of any node. As a result, the clustering structure of Fig. 1 leads to l = 10. If in that same example, we merge clusters 1.1 and 1.2, then l becomes equal to

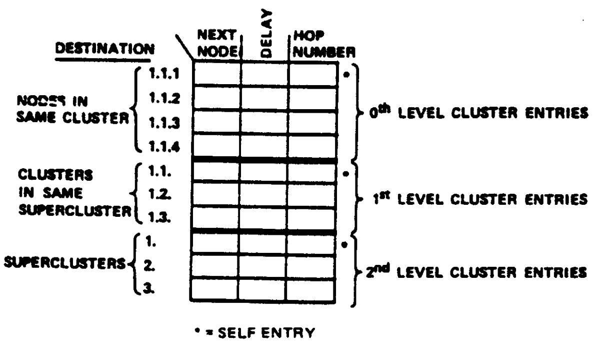

Fig. 3. Routing table of node 1.1.1.

12 (we eliminated one cluster, but increased the size of the largest cluster by 3). Moreover, it is easy to construct clustering structures which lead to values of l close to N (e.g., 2 clusters, one containing 21 nodes and the other 3, thus $l = 23$).

Since the routing cost (capacity, storage) is directly related to the table length, then it is important to determine those clustering structures which lead to a minimal table length, i.e., a minimal routing cost.

2. As we stated earlier, the reduction of routing information generally leads to an increase in network path length. To illustrate this fact, let us consider the case where messages must be sent from node 3.2.1 considers cluster 3.1 as a single node. As a result, messages destined to any node in 3.1 will enter that cluster from the *same* node (exchange node). Assume that the entry node is 3.1.1; then messages destined to 3.1.3 and 3.1.4 will incur longer path lengths (the increases are respectively by 1 and 2 hops). On the other hand, if we merge clusters 3.1 and 3.2, we eliminate the above increase in path length but this will increase the table length (only by one entry in this example). Consequently, in general, there will be a tradeoff between gains in table length and loss in path length. Moreover, given an appropriate clustering structure, the assignment of nodes to clusters, clusters to superclusters, etc., should take advantage of the natural grouping of nodes which exist in a particular application; the latter issue, however, is not examined in this paper (see [14]).

Note that the hierarchical routing procedure we propose need *not* imply a hierarchical topological structure; indeed this routing procedure provides very significant improvements when applied to a distributed network topology. On the other hand, the network topology itself could include a hierarchical structure as well.

In summary, in this paper we address the following two issues:

i. The determination of an appropriate clustering structure, i.e., the size of the clusters at all levels and the number of levels so as to minimize the length of the routing table (routing cost).

ii. The performance evaluation of the mHR schemes (in terms of path length) and their comparison with the present non-clustered policies.

3. Minimum routing information

In this section, we introduce some further notation and formally pose the problem of finding an optimal clustering structure. We then proceed with the derivation of the optimal solution and the study of its characteristics.

Any hierarchical classification scheme lends itself to a tree representation [19]. The tree structure has already been introduced in Fig. 2, to represent the 3-level hierarchical clustering of the 24-node network in Fig. 1, and it can easily be extended to represent a general m-level hierarchical partitioning.

A k^{th} level cluster, C_k, is defined recursively as a set of $k - 1^{\text{st}}$ level clusters. It corresponds to a node at level k in a tree representation.

A k^{th} level cluster is *identified*, similar to the Dewey notation, by a vector of predecessors, $i_{k+1} = (i_m, i_{m-1}, \ldots, i_{k+1})$ which can subsequently serve as an *address* of C_k. The index, i_m, indicates the $m - 1^{\text{st}}$ level cluster, say $C_{m-1}(i_m)$, to which C_k belongs; i_{m-1} indicates the $m - 2^{\text{nd}}$ level cluster in $C_{m-1}(i_m)$ to which C_k belongs, etc. The notation $C_k(i_m, i_{m-1}, \ldots, i_{k+1})$ or $C_k(i_{k+1})$, will be used when there is a need to identify C_k.

Notice that a leaf in the tree representation corresponds to a node (0^{th} level cluster) in the network, and to any node is associated an address vector i_1 which will be used for the routing of messages. As an example, node (1,3,1) is the 0^{th} level cluster $C_0(1,3,1)$; it belongs to the 1^{st} level cluster $C_1(1,3)$ which in turn belongs to the 2^{nd} level cluster $C_2(1)$, and finally all 2^{nd} level clusters belong to the unique 3^{rd} level cluster C_3.

The *degree* of a k^{th} level cluster, C_k, is defined as the number of $k - 1^{\text{st}}$ level clusters included in C_k. It also indicates the downward degree of the corresponding node in the tree. We denote by $n_k(i_{k+1})$ the degree of $C_k(i_{k+1})$, we also define $n_k = \{n_k(i_{k+1})\}\ i_{k+1}$ as the vector of degrees of all the k^{th} level clusters. Moreover, we let $n = (n_1, n_2, \ldots, n_m)$ be the degree vector. Finally, S will denote the set of nodes and N its size.

We are now ready to derive expressions for the *length of the routing table* (RT) and the *size constraint.*

The summation of the degrees of all the 1^{st} level clusters gives the total number of nodes in the network (i.e., the total number of leaves in the tree structure). Hence,

$$N = \sum_{i_m=1}^{n_m} \ldots \sum_{i_k=1}^{n_k(i_m,\ldots,i_{k+1})} \ldots \sum_{i_2=1}^{n_2(i_m,\ldots,i_3)} n_1(i_m, \ldots, i_2) . \tag{1}$$

Eq. (1) will generally serve as a constraint over the choice of the optimal degree vector n, and it will be referred to as the *size constraint.*

As an example, consider a 2-level hierarchical clustering composed of n_2 1st level clusters. Let i_2 ($i_2 = 1, 2, ..., n_2$) denote an arbitrary 1st level cluster, and $n_1(i_2)$ be the corresponding number of nodes, then

$$N = \sum_{i_2=1}^{n_2} n_1(i_2)\,. \tag{2}$$

Let $l[C_0(i_1)]$ be the length of the RT at node $C_0(i_1)$; length is defined as the number of entries in that table. Then

$$l[C_0(i_1)] = \sum_{k=1}^{m} n_k(i_m, ..., i_{k+1})\,.$$

The *assumption* is: each node of the network, $C_0(i_1)$, contains an RT with an entry for each k-1st level cluster in the same k^{th} level cluster as $C_0(i_1)$ (there are $n_k(i_m, ..., i_{k+1})$ such entries), and this for $k = 1, 2, ..., m$.

Recall that we assume that the RT's are of equal length l, which must accommodate the number of entries at any node's RT. Hence,

$$l(m,n) \overset{\Delta}{=} \max_{\left\{\substack{\text{over all}\\ \text{nodes}}\right\}} \{\sum_{k=1}^{n} n_k(i_m, i_{m-1}, ..., i_{k+1})\}\,. \tag{3}$$

In the example above,

$$l(2,n) \overset{\Delta}{=} \max_{i_2} \{n_2 + n_1(i_2)\}\,.$$

Finally, we have the following:

Problem statement

$$\begin{array}{rll} \text{given}: & N & \\ \text{minimize}: & l(m,n) & \text{(see eq. (3))} \\ \text{over}: & m \text{ and } n & \\ \text{subject to}: & \text{size constraint} & \text{(see eq. (1))} \\ & m \text{ a positive integer variable} & \\ & n \text{ a vector of positive integer variables} & \end{array} \tag{4}$$

In Section 3.2 we give the real-valued and in Section 3.3 the integer solution to this problem.

3.2. Real-valued solution of the optimization problem

We first proceed to solve this problem with the assumption that n may be a real valued vector. We do this in order to obtain an explicit analytical expression for the optimal solution. As a consequence of this assumption, a summation as in Eq. (2) becomes meaningful only if n_2 is an integer, or if all the $n_1(i_2)$'s are equal, say to n_1, in which case the summation becomes $n_2 n_1$. In fact, the solution of the optimization problem will show that clusters at the same level must be of the same degree; hence, all the summations in Eq. (1) will become meaningful a posteriori.

Optimality for a fixed m

Proposition 1. *Given m, the number of levels in the hierarchy and assuming that n is a real valued vector, the solution of our problem is such that:*

(a) *all clusters at all levels, k = 1, ..., m, are composed of the same number of lower level clusters, that is,*

$$n_k(i_{k+1}) = n_k = N^{1/m}\,, \qquad \forall i_{k+1}\,,\ k = 1, ..., m\,; \tag{5}$$

(b) *with this optimal assignment, the minimum table length is*

$$\bar{l} = mN^{1/m}\,. \tag{6}$$

Proof. The proof proceeds by induction on the number of levels, m. First, we start by showing that Proposition 1 is true for $m = 2$. *For m = 2,* the problem becomes:

$$\begin{array}{ll} \min: l = \max\limits_{\substack{\text{over } i_1\\ 1 \leq i_2 \leq n_2}} \{n_1(i_2) + n_2\}\,, & \\ \text{over}: n_1 = \{n_1(i_2)\}_{i_2} \text{ and } n_2\,, & \\ \text{s.t.}: \sum\limits_{i_2=1}^{n_2} n_1(i_2) = N \text{ and } n_1, n_2 \text{ positive}\,. & \end{array} \tag{7}$$

From the above, we note that $l \geq n_1(i_2) + n_2$, $\forall i_2 = 1, ..., n_2$. Let n_2 be *fixed.* Then, summing this last relation over i_2, we get for a feasible vector n:

$$n_2 l \geq N + n_2^2\,.$$

This equation provides a lower bound on the optimal solution for a fixed n_2. Consequently, if a feasible solution achieves that lower bound, then it must be optimal. Such a solution is

$$n_1(i_2) = \frac{N}{n_2}\,, \qquad i_2 = 1, 2, ..., n_2\,. \tag{8}$$

If we now let n_2 be a variable, the problem reduces to minimizing $l = N/n_2 + n_2$ over n_2. The optimum is achieved for $n_2 = N^{1/2}$ which, combined with Eq. (8), proves that Proposition 1 is true for $m = 2$.

Assuming that Proposition 1 is true for up to $m - 1$ levels, let us show that this implies it is true for *m levels*. The tree structure which corresponds to this general case, is then composed of $n_m(m - 1)$ level subtrees. Each subtree, say i_m ($i_m = 1, 2, ..., n_m$), contains a certain number of network nodes (leaves) which we denote by $p(i_m)$. As a result the same constraint, Eq. (1), is equivalent to the following set of constraints:

$$\sum_{i_{m-1}=1}^{n_{m-1}(i_m)} \cdots \sum_{i_2=1}^{n_2(i_m,...,i_3)} n_1(i_m, ..., i_2) = p(i_m) ,$$

$$i_m = 1, ..., n_m , \tag{9}$$

$$\sum_{i_m=1}^{n_m} p(i_m) = N . \tag{10}$$

Let us *fix* the variables n_m and $p(i_m)$, $i_m = 1, ..., n_m$, such that Eq. (10) is satisfied. Our problem becomes decomposable into n_m subproblems, each corresponding to a given value of the index i_m. Moreover, such subproblems satisfy the induction hypothesis; hence, for a given i_m, the optimal solution is

$$n_k(i_{k+1}) = [p(i_m)]^{1/(m-1)}$$

$$\forall i_{k+1} \ (i_m \text{ fixed}) , \qquad k = 1, 2, ..., m - 1 . \tag{11}$$

With such an assignment the problem becomes

$$\min : l = \max_{i_m} \{(m - 1)[p(i_m)] + n_m\}$$

$$\text{over} : p(i_m) \qquad i_m = 1, ..., n_m \text{ and } n_m$$

$$\text{s.t.} : \text{Eq. (10) holds.}$$

The above problem can be solved similar to Problem 7 ($m = 2$). Then using Eq. (11) we arrive at Eqs. (5) and (6). A more complete proof can be found in [14].

We now intend to let m vary and solve for the global optimum.

Proposition 2. *The global optimal clustering is achieved when the number of levels is*

$$m_* = \ln N , \tag{12}$$

and when the degree vector n^* *is such that all components have equal values:*

$$n^*_k = n^* = e = 2.718 ..., \qquad k = 1, 2, ..., m_* . \tag{13}$$

The corresponding minimum table length is

$$l_* = e \ln N . \tag{14}$$

The proof follows simply from the results obtained in Proposition 1.

Duality

It is of interest to consider the dual formulation of our Problem (4). The new objective is to find the maximum number of nodes N such that there exists an mHC whose application results in a routing table of a given length. The dual propositions to 1 and 2 are respectively,

Proposition 3. *For a fixed m and l, the real valued solution of the dual problem is such that*

$$n_k = \frac{l}{m} , \qquad k = 1, 2, ..., m .$$

With this assignment

$$N = \left(\frac{l}{m}\right)^m .$$

Proposition 4. *The real valued global optimum of the dual problem is such that*

$$m_* = \frac{l}{e} ,$$

$$n^*_k = e, \qquad k = 1, ..., m_* ,$$

$$N^* = e^{l/e} .$$

We now present some numerical examples.

Examples. Recall that the ratio of table length *with* clustering to the one *without* clustering, l/N (relative table length), represents the performance measure by which we characterize the gains obtained from the hierarchical routing. It is the behavior of the optimal solution of the primal problem (4) that we display. Figures 4 and 5, respectively, illustrate the behavior of $\overline{l}/N$ and l/l_* (see Eqs. (6) and (14)) with respect to m and for several values of N. These figures show that very significant savings can be achieved.

Note that $\overline{l}/N = 1$ for $m = 1$; this corresponds to

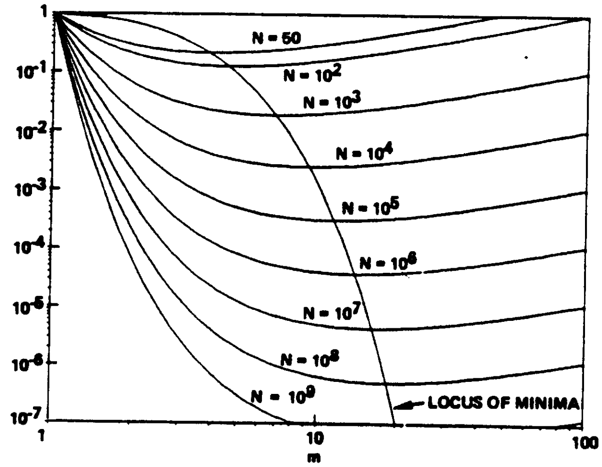

Fig. 4. Minimum relative table length, l/N, given m.

the degenerate 1-level hierarchical routing which is simply our original non-clustered scheme. For m varying from 1 to $\ln N$, l/N decreases to values quite a bit smaller than 1. For m greater than $\ln N$, l/N is an increasing function of m, and as m goes to infinity it is asymptotic to $1/N(m + \ln N)$. However, values of m which lead to $l/N \geqslant 1$ are certainly of no interest; furthermore as we will see later, it is more advantageous to operate with as small a number of levels as possible. As a result, in what follows we restrict the range of m to $\{1, \ldots, \ln N\}$. Note also that for $m = N$, $l/N = N^{1/N}$ whose limit is 1 when N goes to infinity.

The plots exhibit a very flat region around the minimum. They also show an initial fast decrease of $\bar{l}$ toward a value close to the minimum. This last property is better illustrated in Fig. 6 where $\bar{l}/N$ is plotted with respect to N for $m \in \{1,2, \ldots, \ln N\}$; this indicates that most of the table reduction can be obtained with hierarchical clustering whose number of levels is quite a bit smaller than m_* (Eq. (12)). This is an important property which proves to be very valuable below.

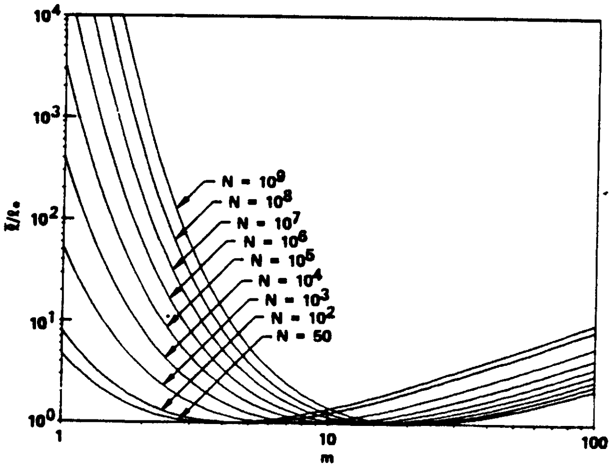

Fig. 5. Ratio of table lengths at optimality given m, and at global optimality, l/l_*.

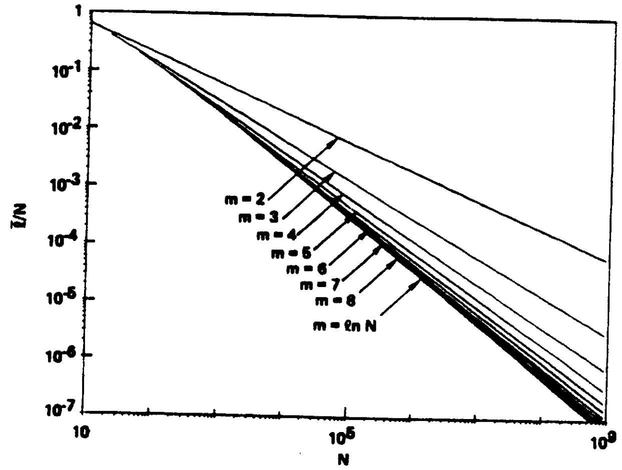

Fig. 6. Minimum relative table length l/N versus the number of nodes.

3.3. Integer solution

In this section we intend to solve the *integer* optimization problem as formulated in Eq. (4) except now we assume that all degrees at the same level are equal, and we also change the size constraint to an inequality. The problem becomes

$$\min : l = \sum_{k=1}^{m} n_k ,$$

$$\text{over} : n, m \text{ integer valued} , \tag{15}$$

$$\text{s.t.} : \prod_{k=1}^{m} n_k \geqslant N .$$

The latter modification is introduced to avoid dealing with empty feasible sets of vectors n, for some values of m and N. A solution n, such that $\Pi_{k=1}^{m} n_k > N$, practically means that there will be unused entries in some of the routing tables.

Recall that the global optimum real-valued solution is such that all the component n_k's are equal to e, and therefore since $2 < \mathrm{e} < 3$, we are not surprised in the integer case that the following proposition holds true.

Table 1

Original numbers	Transformation	Sum	Original product	New product
4	2, 2	4	4	4
5	2, 3	5	5	6
6	3, 3	6	6	9
7	2, 2, 3	7	7	12
2, 2, 2	3, 3	6	8	9

Proposition 5. *There exists a global optimum vector* $\boldsymbol{n}_*$ *which is composed of at most two components equal to* 2, *with all the others equal to* 3.

Proof. The idea [3] is that any number (component of $\boldsymbol{n}$) or set of numbers can be replaced by a set of 2's or 3's which results in the same sum but a higher product. Hence the new set is at least as good as the original. As an example, we list in Table 1 some typical transformations.

The proof consists of showing that through such transformations as listed above we can always derive from an optimal solution which does not satisfy Proposition 5, one which does. A complete proof is available in [14].

As a consequence of Proposition 5 the search for the optimal number of levels is reduced to three possibilities. From Problem 15, the optimal m must be such that

$$3^{m-x}2^x \geqslant N \qquad \text{where } x \in \{0, 1, 2\} .$$

Hence the three possible values of m are:

$$1. x = 0 \Rightarrow m_0 = \left\lceil \frac{\ln N}{\ln 3} \right\rceil ,$$

$$2. x = 1 \Rightarrow m_1 = \left\lceil \frac{\ln N/2}{\ln 3} \right\rceil + 1 ,$$

$$3. x = 2 \Rightarrow m_2 = \left\lceil \frac{\ln N/4}{\ln 3} \right\rceil + 2 .$$

Finally the optimal m, m_*, is the solution of

$$\min : l = 3m - x ,$$

$$\text{over} : (m, x) \in \{(m_0, 0), (m_1, 1), (m_2, 2)\} .$$

Note that the optimal pair (m, x) gives the composition of the optimal vector $\boldsymbol{n}_*$.

Proposition 6. *Given* m, *there exists an optimal vector* $\boldsymbol{n}$ *which is such that no two components differ by more than* 1, *and which is given by*

$$n_m = \lceil N^{1/m} \rceil ,$$

$$n_k = \lceil (N/(\Pi_{i=k+1}^{m} n_i))^{1/k} \rceil \qquad k = 2, 3, ..., m , \quad (16)$$

or any permutation of the above solution.

Proof. We can easily show [14] that, given any two numbers which differ by more than 1, we can replace them by *exactly* two numbers which do not differ by more than 1 and which result in the same sum but in a better (i.e., larger) product.

From the above property, we conclude that any n_k, $k = 1, ..., m$, is either equal to a given number, a or $a + 1$. If we let x represent the number of components equal to $a + 1$, then the problem reduces to

$$\min : l = (m - x)a + x(a + 1) = ma + x ,$$

$$\text{over} : (a, x) ,$$

$$\text{s.t.} : a^{m-x}(a - 1)^x \geqslant N ,$$

a, positive integer; $x \leqslant m$, positive integer .

Let us show that there exists at least one component, say n_m, equal to $\lceil N^{1/m} \rceil$. From the constraint above, the optimal a is such that

$$\text{i. } x = 0 \Rightarrow a^m \geqslant N \Rightarrow a = \lceil N^{1/m} \rceil$$

$$\text{ii. } x \neq 0 \Rightarrow (a + 1)^m \geqslant (a + 1)^x a^{m-x} \geqslant N \Rightarrow a + 1 = \lceil N^{1/m} \rceil .$$

Knowing that $n_m = \lceil N^{1/m} \rceil$, Problem 15 can be reduced to $m - 1$ variables, with N replaced by N/n_m. Then repeating the same procedure $m - 1$ times we arrive at Eq. (16).

Numerical examples [14] show that the integer solution exhibits properties similar to the real-valued one, namely the enormous table reduction obtained for small values of m, and that it is extremely close to the real-valued solution. Consequently we will limit our further considerations to the simple real-valued solution.

[3] Private communication with Dr. D. Cantor, Mathematics Department, UCLA.

3.4. Optimality with no "self-entries" in the routing table

In the previous model, at each routing table, one entry (called a self-entry) is reserved for the node which contains that table, and one for each of the k^{th} level clusters, $k = 1, 2, ..., m-1$, to which that node belongs. For some mHR schemes (e.g., those defined in Section 4) and/or with some extra CPU overhead, the updating algorithm can operate without those self-entries. Consequently, the new length l' of the RT's is

$$l' = l - m\,, \tag{17}$$

where l is given by Eq. (3).

The optimal clustering structure for this case is the solution of Problem 4 where l is replaced by l'.

Real-valued solution

For a fixed m, Eq. (5) still holds true. Hence the minimum length is

$$\bar{l}' = mN^{1/m} - m\,.$$

Also, the global optimum [14] is such that

$$m'_* = +\infty\,,$$

$$l'_* = \ln N\,,$$

$$n_k = 1\,, \qquad k \geqslant 1\,.$$

The above result is to be compared with Eq. (14) in which $l_* = \text{e}\; l'_*$ which indicates that, theoretically, an improvement of a fraction, l/e, of the global minimum length can be obtained. These limiting results are, however, meaningless in the integer case.

Integer-valued solution

Similar to the above, for a fixed m Proposition 6 still holds true. As for the global optimum, let us first note that the real-valued solution is such that

$$n_k = \lim_{m \to \infty} N^{1/m} = 1^+\,,$$

where we define 1^+ as the limit 1 approached from above. Therefore we are not surprised that the following proposition holds true [14].

Proposition 7. *There exists a non-degenerate (i.e., no one component is equal to 1) global optimum vector* $\boldsymbol{n}_*$ *which is such that*

$$n^*_k = 2, \qquad k = 1, 2, ..., m_*\,, \tag{18}$$

$$m_* = \left\lceil \frac{\ln N}{\ln 2} \right\rceil .$$

3.5. The catch

So far we have been primarily concerned with the introduction of the mHR schemes and their underlying hierarchical clustering structure as solutions to the reduction of the routing table and its associated overhead. Indeed, we found that enormous gains can be obtained whereby the length of the routing tables may be reduced from N entries to the order of $\text{e} \cdot \ln N$ entries. However, a shortcoming of these gains is the increase in the path length of a message in the network. This comes about from the fact that a given node must send all its traffic to a given cluster, on the same path to that cluster. This path will, in general, be optimal only for a subset of the nodes in the destination cluster. Consequently, some messages will follow longer paths than they should. This issue is addressed next.

It is also possible that less routing adaptability could result from the mHR schemes because of the aggregation of the routing information. This fact may, however, be beneficial in our context of large networks where the routing policy need not adjust to very remote and probably short lived fluctuations.

4. Path characteristics for hierarchical and non-hierarchical adaptive routing policies

The purpose of this section is to characterize the actual or virtual routes obtained from the routing tables under certain equilibrium conditions as defined below. The routing schemes are assumed to belong to the class of hierarchical or non-hierarchical adaptive policies previously introduced. Such policies basically propagate routing information describing the length of the paths to reach any destination node or a set of nodes. The path length is defined as the sum of the lengths of all the channels which constitute that path. Moreover, the length of a given channel is often taken to be a random variable which may reflect the utilization and/or the excess capacity and/or any other information which partly or entirely describes the stochastic state of that channel. The transient nature of adaptive routing renders the analysis of the above problem extremely complicated. In order to make any progress we will assume that all channels are of constant length. This is a simplifying assumption which will, however, allow us to capture the effect of clustering on the network path length; this is the main objective of this section. Moreover the above assumption is an accurate description of routing policies which are only sensitive to changes in the network topology, and of more general policies operating under light traffic conditions [16]. Furthermore if all the channels are considered to be of equal length (say 1), then the routing information is simply what we defined earlier as the hop distance. Such routing information is, in general, utilized by routing policies, at least to detect changes in the network topology.

In summary, we will restrict our considerations to hierarchical or non-hierarchical routing schemes (also referred to as clustered and non-clustered routing schemes) which use as routing information the path length only. Also we consider that all channels are of constant length. In what follows we first assume that all channels are of equal length (one hop) and then we generalize to arbitrary (constant) length channels. The arbitrary but fixed (time-invariant) channel lengths do not explicitly account for estimates of message delay, but rather they constitute a distance measure which relates to the network topology (channel layout, capacities, etc.)

4.1. Further specifications of the routing schemes

Below we show that the *N*on-*C*lustered *R*outing (NCR) scheme, to be defined here, is equivalent to a degenerate 1-level hierarchical routing. As a result the hierarchical routing schemes (mHR) specified below will also do for the NCR scheme.

Built into the mHR schemes is the reduction of the routing information whereby one entry in a routing table may be reserved for more than one destination node. Routing information is aggregated whenever it is exchanged between special nodes in different clusters at any level. Such special nodes will be referred to as *exchange* nodes. Two mHR schemes will be presented below. They differ only in the definition and subsequently the computation of the aggregate routing information. The two schemes will be referred to as the *Closest Entry Routing* (CER) and the Overall Best Routing (OBR) schemes. In order to proceed with their description, we must first specify the underlying m-level hierarchical partitioning of the set of nodes of the network.

Assumption 1. The underlying mHC structure of the set of network nodes is such that all clusters at the same level k are of equal degree, n_k, $k = 1, ..., m$. Also the subset of nodes composing a cluster at any level and their incident channels constitute a 1-connected cluster subnetwork (at *least* one path exists between any pair of nodes).

The former property of the above assumption partly satisfies Proposition 1 which defines the optimal clustering structure that we will eventually use. The latter property is necessary, since the traffic exchanged between nodes in the same cluster must follow paths included in that cluster's subnet.

Because of the above assumption the previous notation can be greatly simplified. In particular the degree vector is reduced to $n = (n_1, n_2, ..., n_m)$. Moreover, if there is no need to identify a cluster with its entire address vector, then the simpler notation below may be used:

$C_k(s) \triangleq k^{th}$ level cluster containing an arbitrary node s.

As a consequence of Assumption 1, the routing tables at any node will contain $l = n_1 + n_2 = ... + n_m$ entries. Note that self entries are included in the routing table. The self entries of the RT at an exchange node may be assigned to carry the aggregate routing information from one cluster to another. The content of the self entries in tables at other nodes (non-exchange nodes) need not be specified in this study. Two aggregation procedures, each for a particular mHR scheme (OBR or CER), are presented below.

CER and OBR hierarchical routing schemes. For the CER (Closest Entry Routing) scheme, no routing information describing the internal behavior of a cluster is propagated outside the cluster. With this rule, a cluster is regarded from the outside as a single (super-)node whose distance to itself is equal to zero. In other words the distance from an exchange node to the clusters at all levels to which it belongs is considered to be equal to zero.

For the OBR (Overall Best Routing) scheme, the average estimated distance from an exchange node to all the nodes in its cluster (including itself) will be propagated as the routing information for that cluster.

Update rule. Let s and t be two neighbor nodes (i.e., they are connected by a channel (s, t)) which belong to the same k^{th} level cluster C_k and not to any lower level cluster, $(k = 1, 2, ..., m)$. Let $C_{k-1}(s)$ and $C_{k-1}(t)$ respectively denote the $k-1^{st}$ level clusters to which s and t belong. As a consequence the routing tables at s and t are such that all the p-level cluster entries for $p = 0, ..., k-2$ refer to different cluster destinations; whereas all the other entries refer to the same cluster destinations.

The object of the updating procedure is to compare the estimated lengths of the paths from s or t to any common destination. Then, the routing tables are updated to show the better paths. Let

$$C_j(i) \qquad i = 1, 2, ..., n_{j+1}\ ; \qquad j = k-1, ..., m-1$$

denote a j^{th} level cluster destination which is common to s and t. To that cluster is associated an entry i (in both tables) amongst the j^{th} level cluster entries; that entry will also be denoted by $C_j(i)$. Also let $\mathrm{HF}(u, C_j(i))$ represent the content of the hop field of entry $C_j(i)$ at node u ($u = s$ or t). Finally, whenever node t receives an update message from node s, then for each common destination entry $C_j(i)$ the following updating algorithm is performed.

```
        IF $\mathrm{HF}(t, C_j(i)) > 1 + \mathrm{HF}(s, C_j(i))$
    THEN $\mathrm{HF}(t, C_j(i)) \leftarrow 1 + \mathrm{HF}(s, C_j(i))$
NEXT NODE FIELD OF $C_j(i) \leftarrow s$        END .    (19)
```

Initially all the entries are set to a large value (∞); except for the self entries. If a CER is used then all the self entries are set to zero, and if an OBR is used then only the 0^{th} level cluster self entries are set to zero, e.g., at node s

$$\mathrm{HF}(s, C_0(s)) \triangleq \mathrm{HF}((s, s) = 0$$

$$\mathrm{HF}(s, C_k(s)) = \begin{cases} 0 & \text{CER} \\ \infty & \text{OBR} \end{cases} \qquad k = 1, ..., m-1$$

all other entries $= \infty$.

Note that in the algorithm above, it is assumed that all the routing information contained in the non-common destination entries in the routing table in node s is aggregated, as specified before, to represent $\mathrm{HF}(s, C_{k-1}(s))$. When required (for OBR), the computation of the averages must proceed sequentially, starting from level 1 to level $k-2$. Moreover the content of the common self entries is not relevant.

A few more remarks can be stated about the above updating rule.

i. If s and t belong to the same 1^{st} level cluster, then their RT's contain only common destination entries. As a result, Algorithm 19 will be performed for all the entries in the table.

ii. A unique "degenerate" mHR routing scheme (NCR) corresponds to either the OBR or the CER schemes with only hierarchical level. Moreover, for such a degenerate case all the network nodes belong to the same unique 1^{st} level cluster; hence, as expected, the updating algorithm will be performed for all the entries in the RT's.

iii. For any pair of nodes s, t the common region in the routing tables can be determined by inspecting the address vectors of s and t.

With the above specifications of the *mHR* and NCR schemes, we are now ready to address the question as to what is the content of the hop fields at any RT, under some defined equilibrium conditions.

4.2. Path characteristics

If no changes occur in the topology of the network, after a certain number of updates, the contents of the hop fields in the routing table will reach "minimal" constant values. In what follows, this situation will be referred to as *equilibrium* condition. Similar to the dynamic programming approach, the above property is due to the fact that improvements are made sequentially at each update over the distance from one node to any cluster (see Algorithm 19). The question arises as to what is the meaning of the routing information at equilibrium, or in other words, what are the characteristics of the paths indicated by the routing tables. We can already note that for the degenerate one-level hierarchical clustering, i.e., when no clustering is used, those paths correspond to the shortest paths in the current topology. Before we proceed, a few more definitions and notations are necessary.

h^{c}_{st} = Length of the estimated minimum path from node s to node t as derived from the routing information at node s. (The superscript c stands for clustered routing.)

Internal path = a path is defined to be internal (included) in a cluster C_k if all the nodes in that path belong to that cluster.

h^{i}_{st} = Length of the shortest path from node s to node t *included in the lowest level cluster* to which both s and t belong (the superscript i stands for an internal path).

Exchange node = (defined previously) an exchange node (to be denoted by e or e_j) of a given cluster is a node of that cluster which is connected to one or more nodes external to that cluster.

$A_k(i_{k+1}) \triangleq$ Subset of all the exchange nodes which connect cluster $C_k(i_{k+1})$ with any k^{th} level cluster which belongs to the same k + 1st level cluster as $C_k(i_{k+1})$.

$w_{eC_k} \triangleq$ Entry in RT giving internal distance measure for C_k (an aggregate variable) as computed from the routing information contained at the exchange node e of C_k.

From the above definitions and previous specifications we note first that a network node (0th level cluster) is its own exchange node. Second we have

$$w_{eC_k} = \begin{cases} \dfrac{1}{|C_k|} \displaystyle\sum_{f \in C_k} h^{\mathrm{c}}_{ef} & \text{for the OBR scheme} \\ 0 & \text{for the CER scheme} \end{cases}$$

$$w_{eC_0} = 0 \tag{20}$$

where $|C_k|$ represents the number of nodes in cluster C_k and f is an arbitrary node of C_k. The above considerations allow us to characterize the path lengths under the mHR schemes.

Proposition 8. *Let s and t be two arbitrary nodes which belong to the same k^{th} level cluster C_k, but not to any lower level cluster; then the length of the path from node s to node t as derived at equilibrium from the routing information contained at node s, satisfies the recursive equation below,*

$$h^{\mathrm{c}}_{st} = h^{\mathrm{i}}_{se_0} + h^{\mathrm{c}}_{e_0t} \tag{21}$$

where e_0 is an exchange node of $C_{k-1}(t)$ which is such that

$$h^{\mathrm{i}}_{se_0} + w_{e_0C_{k-1}(t)} = \min_{e_j \in A_{k-1}(t)} \{h^{\mathrm{i}}_{se_j} + w_{e_jC_{k-1}(t)}\}\,, \tag{22}$$

where $C_{k-1}(t)$ is the k – 1st level cluster which contains node t, and $A_{k-1}(t)$ is its corresponding subset of exchange nodes as defined above.

Proof. The proof proceeds by induction on the level k of the lowest level common cluster. In what follows $C_j(s)$ and $C_j(t)$, $j = 1, ..., m$, will always respectively denote the j^{th} level clusters to which s and t belong.

$k = 1$ case. s, t belong to the same 1st level cluster C_1, then

$$C_0(s) = A_0(s) = s\,,$$

$$C_0(t) = A_0(t) = t\,.$$

Also, since the distance of a node to itself is zero, then

$$h^{\mathrm{c}}_{e_0t} = h^{\mathrm{c}}_{tt} = 0\,.$$

In order to prove Eq. (21) there remains to show that

$$h^{\mathrm{c}}_{st} = h^{\mathrm{i}}_{st}\,,$$

i.e., that h^{c}_{st} is the length of the shortest path from s to t included in C_1. This is true since the RT of any node in C_1 contains an entry for node t; hence at equilibrium we obtain the minimal internal path from s to t. Note that if $m = 1$, i.e., the degenerate case, all nodes belong to the same cluster C_1 which corresponds to the entire set of nodes, hence $h^{\mathrm{i}}_{st} = h_{st}$. In other words, when no clustering is used, i.e., NCR, the routing information indicates, at equilibrium, the shortest (hop) path.

Assuming that Proposition 8 is true up to $k - 1$, let us show that it is true for k.

Proof for k. Let C_k be the k^{th} level cluster common to s and t. All the nodes in C_k contain in their RT's one entry for cluster $C_{k-1}(t)$. The propagation and the subsequent updating of the RT's among the nodes of C_k, is equivalent to finding the minimum path, internal to C_k, from any node in $\{C_k - C_{k-1}(t)\}$ to the fictitious supernode $SC_{k-1}(t)$ shown in Fig. 7. In other words, seen from any node in $\{C_k - C_{k-1}(t)\}$, cluster $C_{k-1}(t)$ is equivalent, in terms of distance, to a center node $SC_{k-1}(t)$ connected to all the exchange nodes in $A_{k-1}(t)$. If $e_j \epsilon A_{k-1}(t)$ then the length of the equivalent edge, from e_j to the center node, is equal to the aggregate information representing cluster $C_{k-1}(t)$ as seen from e_j, i.e.,

$$l(e_j, SC_{k-1}(t)) = w_{e_jC_{k-1}(t)}\,, \tag{23}$$

where the distance from e_j to any other node in $C_{k-1}(t)$ is defined from the induction hypothesis.

If e_0 is the exchange node in $A_{k-1}(t)$ which belongs to the minimal path from s to $SC_{k-1}(t)$

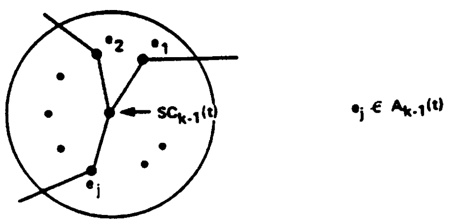

Fig. 7. Equivalent representation of cluster $C_{k-1}(t)$.

obtained at equilibrium, then e_0 satisfies Eq. (22) which represents the length of that minimal path. Due to the routing function previously specified all messages to be forwarded or sent from node s to node t will follow the same minimal path up to the exchange node e_0. At that point e_0 and t belong to the same $k-1^{\text{st}}$ level cluster, hence, $h^c_{e_0 t}$ is known from the induction hypothesis. Consequently Eq. (21) holds true.

Remarks. (1) If CER is used, e_0 represents the closest exchange node of $C_{k-1}(t)$ to node s (for paths included in C_k), which explains the nomenclature: Closest Entry Routing.

(2) If we let the channels have variable lengths and change the previous definitions of path lengths accordingly, we can show [14] that Proposition 8 still holds true.

4.3. Bounds on the increase in path length

The effect of the clustering (reduction of routing information) is an increase in the path length between any pair of nodes, s, t, of an amount $h^c_{st} - h_{st}$. A measure of performance of the mHR schemes is the relative increase of the average path length, i.e.,

$$D = \frac{h_c}{h} - 1\,, \tag{24}$$

where h_c and h denote the average path length in the network respectively with and without clustering (with a uniform traffic assumption)

$$h = \frac{1}{N(N-1)} \sum\sum_{s,t\in S} h_{st}\,,$$
$$h_c = \frac{1}{N(N-1)} \sum\sum_{s,t\in S} h^c_{st}\,. \tag{25}$$

Proposition 8 provides a means for *computing* the values of h^c_{st} for any pair of nodes s, t for a given outcome of the m-level hierarchical clustering of the set of nodes S. Consequently, for that particular situation, it is *numerically* possible to evaluate the relative increase D from Eq. (24) and then compare the clustered with the non-clustered schemes. Moreover, with further assumptions on the structure of the hierarchical partitioning of the nodes, we can obtain analytic bounds on the increase in the path length.

Assumption 2. The diameter [4] of any k^{th} level cluster subnet (see assumption 1) is less than or equal to a quantity d_k, $k = 1, ..., m$.

Note that d_m represents the diameter of the entire network and that $d_k > d_{k-1} > 0$ for all k.

Assumption 3. Any cluster at any level $k = 1, 2, ..., m$ contains the shortest path (if it is not unique, then at least one is contained) between any given pair of nodes which belong to that cluster.

Assumption 2 is simply the specification of the outcome of the clustering of the nodes, since the d_k's can be of any value, whereas Assumption 3 is a natural property that any clustering scheme should seek. The reason for this is that traffic between nodes in the same cluster must (because of the routing function above) follow paths internal to that cluster.

The above assumptions lead to the derivation of some simple bounds. These bounds on the increase in path length apply to the routing schemes (OBR, CER) described above. All the properties listed below rely on Assumptions 1 and 2. If Assumption 3 is used, it will be so specified.

Lemma 1. *Under the above conditions, the value of h^c_{st} for any pair of nodes s, t which belong to the same k^{th} level cluster is such that*

$$h^c_{st} \leq \sum_{j=1}^{k} d_j$$
$$\forall s, t \in \textit{same } k^{\text{th}} \textit{ level cluster}, \forall k = 1, 2, ..., m\,. \tag{26}$$

A very simple proof can be found in [14].

Lemma 1 leads to the following bound on the increase in the average path length.

Proposition 9. *Under the conditions above and Assumption 3, the increase in the average path length in the network due to the reduction of routing information is such that*

$$h_c - h \leq \sum_{k=1}^{m-1} \left[1 - \frac{n_1 n_2 \ldots n_k - 1}{N-1}\right] d_k\,. \tag{27}$$

Proof. Let $C_k(s)$ denote the k^{th} level cluster ($k = 0$, .., ..., m) to which s belongs. Then from Eq. (25)

$$h_c - h = \frac{1}{N(N-1)} \sum_{s\in S} \sum_{k=1}^{m} \sum_{\substack{t\in C_k(s)\\ t\notin C_{k-1}(s)}} (h^c_{st} - h_{st})\,. \tag{28}$$

Let $C_{k-1}(j)$ be a $k-1^{\text{st}}$ level cluster included in

[4] Recall that the diameter of a network is the maximum shortest path between pairs of nodes [11].

$C_k(s)$; there are n_k such clusters, then

$$\sum_{\substack{t\in C_k(s)\\ t\notin C_{k-1}(s)}} (h^{\mathrm{c}}_{st} - h_{st}) = \sum_{\substack{j=1\\ C_{k-1}(j)\cap C_{k-1}(s)=\phi}}^{n_k} \sum_{t\in C_{k-1}(j)} (h^{\mathrm{c}}_{st} - h_{st}) \,. \qquad (29)$$

Since $C_{k-1}(j) \cap C_{k-1}(s) = \phi$ and since both are included in C_k's, Eq. (21) holds true for s and any node t in $C_{k-1}(j)$; after some algebra using Eqs. (20), (21) and (22) we arrive at

$$\sum_{t\in C_{k-1}(j)} h^{\mathrm{c}}_{st} = |C_{k-1}(j)| \min_{e_j\in A_{k-1}(j)} \{h^{\mathrm{i}}_{se_j} + w_{e_jC_{k-1}(j)}\} \,. \qquad (30)$$

Let us define e_s to be the closest (inside $C_k(s)$) exchange node of $A_{k-1}(j)$ to node s, i.e.,

$$h^{\mathrm{i}}_{se_s} = \min_{e_j\in A_{k-1}(j)} \{h^{\mathrm{i}}_{se_j}\} \,. \qquad (31)$$

From Eq. (30) and for any exchange node e_j, particularly e_s, the relation below is true.

$$\sum_{t\in C_{k-1}(j)} h^{\mathrm{c}}_{st} \leqslant |C_{k-1}(j)| h^{\mathrm{i}}_{se_s} + \sum_{t\in C_{k-1}(j)} h^{\mathrm{c}}_{e_st} \,. \qquad (32)$$

Note that in the equation above w was replaced by its value as defined by Eq. (20).

Moreover from Assumption 3 and the definition of e_s,

$$h_{st} = h^{\mathrm{i}}_{st} \geqslant h^{\mathrm{i}}_{se_s} \,, \qquad \forall t\in C_{k-1}(j) \,, \qquad (33)$$

thus

$$\sum_{t\in C_{k-1}(j)} h_{st} \geqslant |C_{k-1}(j)| h^{\mathrm{i}}_{se_s} \,. \qquad (34)$$

Substituting Eq. (34) into Eq. (32), we arrive at

$$\sum_{t\in C_{k-1}(j)} (h^{\mathrm{c}}_{st} - h_{st}) \leqslant \sum_{t\in C_{k-1}(j)} h^{\mathrm{c}}_{e_st} \,. \qquad (35)$$

Note that $e_s, t \in C_{k-1}(j)$, then from Lemma 1,

$$h^{\mathrm{c}}_{e_st} \leqslant \sum_{j=1}^{k-1} d_j \,, \qquad \forall t\in C_{k-1}(j) \,. \qquad (36)$$

From Assumption 1

$$|C_{k-1}(j)| = n_1 n_2 \ldots n_{k-1} \qquad \forall k, j \,. \qquad (37)$$

Substituting Eq. (35), (36) and (36) into Eq. (29), we find

$$\sum_{\substack{t\in C_k(s)\\ t\notin C_{k-1}(s)}} (h^{\mathrm{c}}_{st} - h_{st}) \leqslant (n_k - 1) n_1 n_2 \ldots \ldots n_{k-1} \sum_{j=1}^{k-1} d_j \,. \qquad (38)$$

Note that this last equation is true for any level k, and for any node s, hence by substituting it into Eq. (28), we obtain Eq. (27), after some algebra.

Remark. For a CER scheme the relation in Eq. (32) is tight (i.e., the equality holds true). This indicates that the summation of path lengths obtained with the OBR scheme is smaller than or equal to the one obtained with the CER scheme. Hence the *average path with an OBR is smaller than or equal to the average path length with a CER.*

The above proposition deals with averages; we now place a bound on the increase of the path length between an arbitrary pair of nodes s, t.

Lemma 2. *Under the previous conditions and Assumption 3, and for the CER scheme*

$$h^{\mathrm{c}}_{st} - h_{st} \leqslant \sum_{j=1}^{k-1} d_j$$

$$\forall s, t \in \textit{same } k^{\text{th}} \textit{ level cluster } C_k, \qquad \forall k = 1, 2, \ldots, m \,. \qquad (39)$$

This is due to the fact that with CER the closest exchange node is used to enter a cluster (see [14]) which is not always true with OBR.

We observed previously that Assumption 3 is a realistic one, but if it is not specifically built into the clustering algorithm, there is no guarantee that the outcome of the clustering always satisfies that assumption. This remark leads us to the following proposition.

Proposition 10. *Under the conditions of Proposition 9 and with Assumption 3 removed,*

$$h_{\mathrm{c}} - h \leqslant \sum_{k=1}^{m-1} d_k \,. \qquad (40)$$

The proof [14] relies on the fact that Assumption 3 is always true for the highest level cluster C_m (i.e., for the entire network $h^{i}_{st} = h_{st}$). Hence Eq. (38) is true for $k = m$, and the proof follows from there.

Note that all the bounds derived above are tight for the degenerate case of 1-level hierarchical routing (NCR). To prove this fact, for $m = 1$ Eqs. (27) and (40) lead to $h_c - h \leqslant 0$; but since $h_c - h \geqslant 0$ then $h_c = h$. Similarly for $m = 1$, Eq. (39) gives $h^c_{st} = h_{st}$.

In summary, several fairly general bounds have been derived, depending on the assumptions and/or the routing schemes selected. In the next section we will study the behavior of some of those bounds for a class of networks.

5. Static performance evaluation of the *m*HR schemes for a family of networks

Recall from Section 2 that in this paper we do not explicitly account for the very significant gains obtained in reducing the CPU, storage and line utilization required by the routing procedures from the reduction in *l/N;* as a result the application of the *m*HR schemes will appear to result in a degradation of the performance of the network, as compared to the utilization of a non-clustered scheme. This loss in performance (delay, throughput) is closely related to the average path length a message follows in the network. The evaluation of the increase in path length provides us with a first cut modeling of the loss in network performance. Moreover, the study of the bounds, derived previously, represents a worst case evaluation of the *m*HR schemes. Since the evaluation is in terms of path length, we will refer to it as *static performance evaluation.* On the other hand, the gains we obtain are still modeled by the single variable *l/N* which represents the reduction of routing information. We defer the throughput-delay evaluation to a later paper [15]. In that paper we find that the table reduction provides savings in capacity, storage, throughput and delay which more than compensate for the vanishing increase in path length.

The static performance evaluation is performed over a class of computer networks.

5.1. A family of large distributed networks

The networks to be considered are all the connected graphs upon which it is possible to fit an *m*-level hierarchical clustering whose outcome satisfies Assumptions 1–3. Also the resulting cluster subnets at any level are of diameters bounded by a power law function of the number of nodes in that cluster; i.e., if n is the size of a cluster and d the diameter of that cluster's subnet then

$$d \leqslant bn^{\upsilon} + c\,, \tag{41}$$

where b, c, υ are positive parameters and $0 \leqslant \upsilon \leqslant 1$ (see below).

If N is the size of such a network, then the average path length (hop distance) of that network h must be a power law function of N,

$$h = aN^{\upsilon}\,, \tag{42}$$

where a is a positive parameter.

Grid type networks, hexagonal networks, etc., fall into that category when the *m*HC results in subnetworks of a similar structure as the original and when the path lengths are expressed in hops. Expressions for the average path length (with a uniform traffic matrix) and for the diameter of the grid and the torus networks have been derived in [14]. Some of the results obtained are:

$$\text{square grid of size } N \begin{cases} h = \tfrac{2}{3}\sqrt{N}\,, \\ d = 2\sqrt{N} - 2\,, \end{cases} \tag{43}$$

$$\begin{array}{l}\text{square torus of size } N \\ \text{(with } \sqrt{N} \text{ an odd integer)}\end{array} \begin{cases} h = \tfrac{1}{2}\sqrt{N}\,, \\ d = \sqrt{N} - 1\,, \end{cases} \tag{44}$$

Furthermore, if the partitioning of either the square grid or torus networks results in grid cluster subnets at all levels, then for any cluster subnet of size n its diameter d is such that

$$d \leqslant 2\sqrt{n} - 2\,. \tag{45}$$

As a consequence the grid and torus networks fit the above descriptions. Note also that for those networks the exponent υ (Eqs. (41) and (42)) is equal to $\tfrac{1}{2}$.

In general, the exponent υ reflects the connectivity of the network considered. For very highly connected networks υ is in the neighborhood of zero; e.g., for a fully connected network $\upsilon = 0$ ($h = 1, d = 1$). Whereas for very low connected networks υ is in the neighborhood of one; e.g., for loop or chain type networks, $\upsilon = 1$.

Computer communication networks fall into the class of distributed networks. This class includes networks such as the ARPANET, AUTODIN II, CYCLADES, TRANSPAC, EPSS, EIN, DATAPAC, TELENET, etc. The main characteristic of those distributed networks is their low connectivity. In general, a connectivity 2 (or 3) is imposed on their design. For large distributed networks a connectivity of 3 to 4 seems more appropriate [25]. The torus networks considered above are of connectivity 4 and with an exponent $\upsilon = \frac{1}{2}$, hence they appear to be good representatives of large distributed networks. Moreover, their topological structure leads to a simple partition such as square subgrid clusters. In the sequel, we will first derive a limiting result valid for the entire class of networks, then we will restrict our numerical applications to values of a, b, c, υ as obtained for the torus net, i.e.,

$$a = \tfrac{1}{2}\,, \quad b = 2\,, \quad c = -2\,, \quad \upsilon = \tfrac{1}{2}\,. \tag{50}$$

5.2. *Asymptotic performance evaluation of the mHR schemes*

The family of networks considered here satisfies Assumptions 1–3, hence Proposition 9 holds true. Let E be defined as the bound on the relative increase in path length D (see Eq. (24)). It is the behavior of E versus the relative table length l/N in which we are interested.

For an optimal clustering structure we know from Proposition 1 that the degree vector $\boldsymbol{n}$ must satisfy Eq. (5). Then from Eqs. (27), (41) and (42) and after some algebra we obtain

$$0 \leqslant \frac{h_c}{h} - 1 \leqslant E \triangleq \frac{1}{a(N-1)N^{\upsilon}} \left[N \left[b\,\frac{N^{\upsilon} - N^{\upsilon/m}}{N^{\upsilon/m} - 1} + c(m-1) \right] - b\,\frac{N^{\upsilon+1} - N^{(\upsilon+1)/m}}{N^{(\upsilon+1)/m} - 1} - c\,\frac{N - N^{1/m}}{N^{1/m} - 1} \right], \tag{51}$$

where υ is assumed to be different from zero. Note again that for $m = 1$, $E = 0$. Also from Eq. (6) the relative table length is

$$l/N = \frac{mN^{1/m}}{N}\,. \tag{52}$$

The above considerations lead to the general limiting result below, which is a key theorem.

Proposition 11. (Limiting Performance). *Consider the above family of networks and the above mHR schemes (OBR, CER) with a fixed number of levels m and an optimal clustering structure. Then as N, the number of nodes, goes to infinity, the "static" performance of the mHR schemes approaches that of a non-clustered routing scheme, while the relative table length approaches zero; i.e.,*

$$N \to \infty \Rightarrow \begin{cases} h_c/h \to 1\,, \\ l/N \to 0\,. \end{cases}$$

Thus we claim that in the limit, hierarchical routing leads to enormous table reduction with relatively no significant increase in path length. In other words, hierarchical routing will achieve similar throughput-delay performance as the NCR, while requiring significantly less nodal storage and channel capacity. This is a fundamental result which greatly satisfies our initial objective of reducing the operating cost of adaptive routing in large networks. This cost vanishes in the limit!

Proof. It is enough to prove that the limit of E is zero. Expanding Eq. (51) around N^{-1}, we find

$$E = \frac{b}{a} N^{-\upsilon/m} + 0(N^{-\upsilon/m})\,, \tag{53}$$

hence

$$\underset{N\to\infty}{\text{limit}}\, E = 0\,.$$

Also, the second limit is obvious. Q.E.D.

Note that the closer υ is to one ($\upsilon \neq 0$), the faster is the convergence of E to zero. In other words, as could be expected, the more distributed (and less connected) the networks are, the better the mHR's perform.

The above results hold true if we *relax* Assumption 3; in this case we use the bound derived in Proposition 10 (Eq. (40)) [14].

The result of Proposition 11 was derived for a fixed m; let us now examine the situation where m is *variable.* Of interest is the value of m which corresponds to the global optimum clustering structure. That value is, from Eq. (12), $m_* = \ln N$.

Substituting Eq. (12) into Eq. (51), we arrive at E_* whose limit is

$$\underset{N\to\infty}{\text{limit}}\, E_* = \frac{b}{a}\left[\frac{1}{e^{\upsilon} - 1} - \frac{1}{e^{1+\upsilon} - 1}\right]. \tag{54}$$

As a consequence the result of Proposition 11 is not necessarily true anymore when m is variable. If we consider the coefficients of Eq. (50) then the above limit is equal to 5.01. This shows that the cost of operating at the (global) minimum table length may be quite high (up to 6 times the increase in path length). Fortunately, as noticed in Section 3.2, most of the table reduction, for practical purposes, may be obtained with m quite a bit smaller than the global number of levels m_*, and the cost at a small m is quite minimal. In other words, choosing m smaller than m_* results in giving back very little gains in table length for a tremendous improvement in performance. This fact is illustrated in Fig. 8, where we note a very sharp increase of E as l/N gets close to its global minimum value. It is that sharp region of the curve that we need to avoid in order to keep the increase in path length significantly low. Fig. 8 also shows the behavior of E_* versus l/N.

5.3. Static performance evaluation of the mHR's: numerical applications

In the previous section we observed that at the limit ($N \to \infty$) considerable table reduction can be achieved with no loss in performance. Now, we intend to look at the more general case of a finite N. The purpose is again to correlate the degradation in performance with the table reduction. We evaluate a maximum performance degradation in terms of the gains in table length. Also this evaluation will be carried out with an mHC which results in a *minimal* table length.

Recall that the numerical study below is restricted to values of a, b, c, v, as obtained for torus networks (Eq. 50), although such a study could easily be repeated for other networks which belong to the family considered here.

Eqs. (51) and (52) provide us with a parametric representation of E as a function of l/N. m acts as the coupling variable in that representation. By letting m vary from 1 to $\ln N$ we obtain all the possible values of l/N; and subsequently for each value of l/N we obtain the corresponding value of E. The above range of m is chosen in accordance with the results obtained in Section 3.2 (refer to Proposition 2 and Fig. 5); and also in accordance with the fact that E is an increasing function of m (this fact is obvious from the proof of Proposition 9).

Numerical results are presented in a set of figures

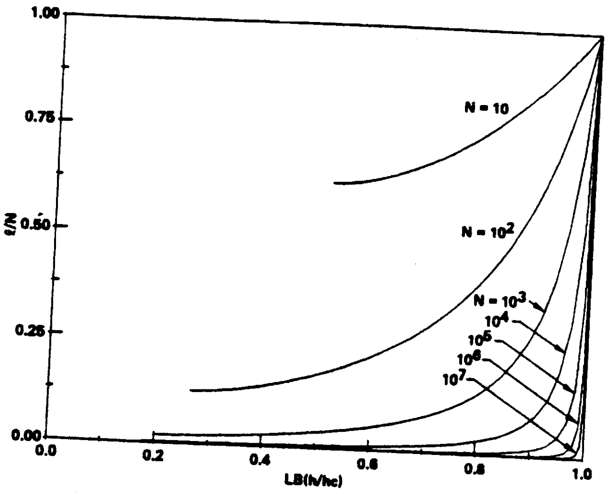

Fig. 9. Lower bound on the ratio of path length without and with clustering versus l/N.

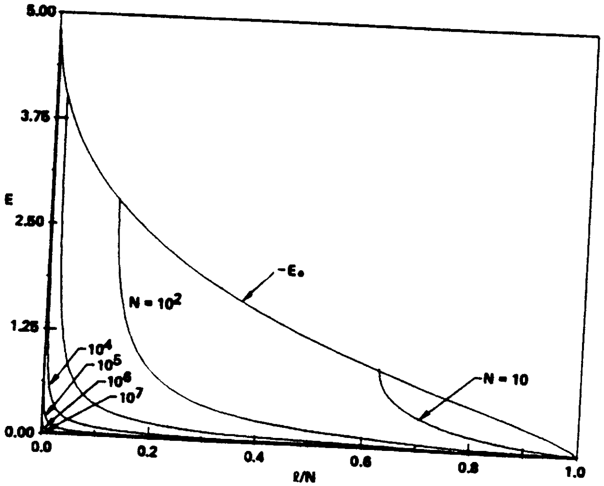

Fig. 8. Bound on the relative increase in path length E, versus the relative table length l/N.

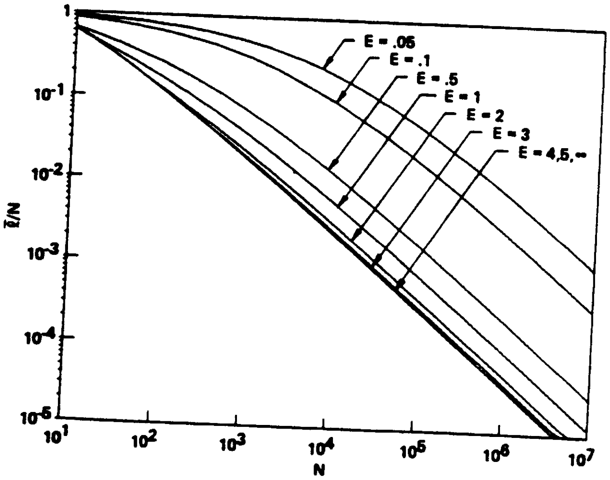

Fig. 10. Decrease in table length for a given maximum increase in path length.

as follows: Fig. 8 illustrates the behavior of E with respect to l/N and for several values of N. We observe that an original substantial table reduction can be achieved for small values of E, i.e., for a small drop in performance. However if we try to reduce l/N to values close to its global minimum, Eq. (14), then E increases sharply. Fig. 8 also illustrates the limiting behavior of the mHR schemes (see Proposition 11) whereby as N becomes larger, more reduction in l/N can be obtained for a lesser loss in performance. This property is shown by the fact that the curves for E versus l/N remain flat on the l/N axis for larger intervals.

Fig. 9 shows the behavior of $1/(1 + E)$ with respect to l/N. That is, from Eq. (51) we see that $h/h_c \geqslant 1/(1 + E) \triangleq LB(h/h_c)$. These figures exhibit properties similar to the previous ones.

Finally, Fig. 10 shows how much table reduction can be obtained for a given "tolerance" E as a function of the size N. The concentration of the curves for $1 \leqslant E \leqslant 5$ (recall from Eq. (54) that $E = 5.01$ is the maximum error) again shows that beyond a certain point the gains in table length can only be achieved at the expense of large losses (large E). However in the range 0 to 1 for E considerable gains can yet be obtained. For that range of E the corresponding range of the number of levels m is limited to fairly small values, $m \leqslant 4$ [14]. Moreover, in Section 3.2, as noticed earlier, most of the table reduction is obtained for small values of m. We conclude that the mHR schemes operating with a small number of levels $2 \leqslant m \leqslant 4$ yield substantial table reduction for a relatively small increase in path length.

6. Summary

In this paper, we have examined the tradeoffs which come about due to hierarchical routing in large networks. The obvious gain is that the length of the routing tables in each node can be reduced significantly. With smaller routing tables, we require less storage and processing in the nodes as well as less communications overhead. The loss is that smaller (i.e., clustered) routing tables give less precise routing information which then results in longer path lengths for the message traffic.

The investigations in this paper have led to an evaluation of these two opposing variables, i.e., the routing table length and network path length. We have shown that hierarchical routing schemes and their underlying hierarchical clustering structure lead to significant reductions of the routing table length. The optimal hierarchical clustering structure was found which minimized the length of the routing table and consequently resulted in a minimum cost routing scheme. Enormous gains were achieved whereby the table length was reduced from N (N = number of nodes) to $e \ln N$.

As regards the network path length, we were able to place an upper bound on its increase due to the introduction of hierarchical routing as a function of the routing table reduction. These bounds allowed us to establish our major result, namely, that in the limit of very large networks, enormous table reductions may be achieved with essentially no increase in network path lengths (an intuitively pleasing, and possibly obvious, result).

However, routing table length and network path length are not the qualities by which one ordinarily evaluates network performance. Rather, we are usually interested in the throughput-delay tradeoff. Clearly, these four quantities are related through the storage, processing and updating requirements they create. In a forthcoming paper [15] we evaluate the performance of hierarchical routing directly in terms of delay and throughput. Indeed, we show that for large distributed networks, present (full table length) routing procedures very quickly become infeasible. More importantly, we establish that hierarchical routing procedures are capable of operating very efficiently in the environment of large networks.

References

[1] N. Abramson, The ALOHA system – another alternative for computer communications, AFIPS Conference Proceedings, (FJCC, Las Vegas, Nevada, 1970) 37, 281–285.

[2] S. Carr, S. Crocker and V. Cerf, HOST-HOST communication protocol in the ARPA network, AFIPS Conference Proceedings, (SJCC, Atlantic City, New Jersey, 1970) 36, 589–597.

[3] F. Closs, Message delays and trunk utilization in line-switched and message-switched data networks, Proceedings of the First USA-Japan Computer Conference, (1972) 524–530.

[4] F. Closs, Time delays and trunk capacity requirements in line-switched and message-switched networks, International Switching Symposium Record (Boston, Massachusetts, 1972) 428–433.

[5] H. Frank, J.T. Frisch and W. Chou, Topological considerations in the design of the ARPA network, AFIPS Conference Proceedings, (SCJJ, Atlantic City, New Jersey, 1970) 36, 581–587.

[6] H. Frank and W. Chou, Topological optimization of computer networks, Proc. IEEE, 60 (11) (1972) 1385–1397.

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

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

[9] M. Gerla, W. Chou and H. Frank, Computational considerations and routing problems for large computer communication networks, Proc. National Telecommunication Conference, 2: 2B-1 to 2B-11, Atlanta, Georgia, November 1973.

[10] M. Gerla, Deterministic and adaptive routing policies in packet-switched computer networks, Proc. Third Data Communication Symposium, St. Petersburg, Florida, November 1973, 23–28.

[11] F. Harary, Graph Theory (Addison-Wesley, Reading, MA, 1972).

[12] F. Heart, R. Kahn, S. Ornstein, W. Crowther and D. Walden, The interface message processor for the ARPA computer network, AFIPS Conference Proceedings, (SJCC, Atlantic City, New Jersey, 1970) 36, 551–567.

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

[14] F. Kamoun, Design considerations for large computer communication networks, Ph.D. Dissertation, Computer Science Department, University of California, Los Angeles, March 1976.

[15] F. Kamoun and L. Kleinrock, Stochastic performance evaluation of hierarchical routing for large networks, submitted to Computer Networks.

[16] L. Kleinrock, Communication Nets; Stochastic Message Flow and Delay, (McGraw-Hill, New York, 1964, reprinted by Dover Publications, 1972).

[17] L. Kleinrock, Analytic and simulation methods in computer network design, AFIPS Conf. Procs, (SJCC, Atlantic City, New Jersey, 1970) 36, 569–579.

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

[19] D.E. Knuth, The Art of Computer Programming, Vol. 1, (Addison-Wesley, Reading, MA, 1969).

[20] S. Lam, Packet switching in a multi-access broadcast channel with applications to satellite communication in a computer network, School of Engineering and Applied Science, University of California, Los Angeles, UCLA-ENG-7249, March 1974.

[21] C. McCoy, Jr., Improvements in routing for packet-switched networks, Naval Research Laboratory; Washington, D.C., NRL Report 7848, 1975.

[22] J.M. McQuillan, Adaptive routing algorithms for distributed computer networks, Bolt Beranek and Newman Inc., Cambridge, MA, Report No. 2831, May 1974.

[23] M. Mesavoric, D. Macko and Y. Takahara, Theory of Hierarchical Multilevel Systems (Academic Press, New York, 1970).

[24] H. Miyahara, T. Hasegawa and Y. Teshigawara, A comparative analysis of switching methods in computer communication networks, Proc. ICC, 6-6 to 6-10, San Francisco, California, June 1975.

[25] Network Analysis Corporation, The practical impact of recent computer advances on the analysis and design of large scale networks, First Semiannual Technical Report, Glen Cove, New York, May 1973.

[26] E. Port and F. Closs, Comparison of switched data networks on the basis of waiting times, IBM Zurich, Report RZ405, January 1971.

[27] L.G. Roberts and B.D. Wessler, Computer network development to achieve resource sharing, AFIPS Conf. Proc, (SJCC, Atlantic City, New Jersey), 1970) 36, 543–549.

[28] L.G. Roberts, Data by the packet, IEEE Spectrum, 11 (2) (1974) 46–51.

[29] L.G. Roberts, Telenet principles and practice, Communications Networks (On Line), September (1975) 315–329.

[30] P.P. Schoderbeck, Management Systems (John Wiley, New York, 1971).