Data Communications Through Large Packet Switching Networks

Leonard Kleinrock
University of California, Los Angeles, California, U.S.A.

Farouk Kamoun
Universite de Tunis, Tunis, Tunisia

ABSTRACT

The topological design and adaptive routing procedure for computer networks becomes infeasible under their present form as the number of network nodes grows. In this paper we present, optimize and evaluate hierarchical procedures to be used in the case of large networks. These procedures are an extension of present schemes and rely on a hierarchical clustering of the network nodes. Models are developed to determine optimal clustering structures which lead to a minimal routing table as well as those structures which lead to a minimal computational cost for the topological design. Both optimal structures achieve enormous savings. The effect of hierarchical routing on network throughput and delay is also studied and demonstrates the efficiency of hierarchical routing for large networks.

1. INTRODUCTION

A new store-and-forward switching technique known as "packet switching" has recently been developed for data communications in computer networks. The principles of this technique may be found in [1, 2, 3] and the bibliographies contained therein. 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, EIN in Europe, DATAPAC in Canada, 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 [2, 4, 5]. Unfortunately the cost of conducting the design is prohibitive if these same techniques are extrapolated to the case of large networks [6]. Indeed, not only does the cost of design grow exponentially with the network size, but also the cost of a straightforward adaptive routing procedures becomes prohibitive. Other design and operational procedures (routing techniques) must be found which handle the large network case and such techniques form the subject of this paper.

1.1 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.

*This research was supported by the Advanced Research Projects Agency of the Department of Defense under Contract DAHC 15-73-C-0368.

The selection of the next node P is made by a well-defined decision rule referred to as the routing policy. Routing policies may be divided into two main classes: deterministic and adaptive [5, 7, 8, 9]. 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. Such policies base their decisions on the measured values 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)[7, 8, 10].

In any case, routing information stored in tables at each node is used to identify the output line for each destination*. 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 [8] using routing information collected internally and provided from neighboring nodes. Such a scheme is used to operate the ARPANET [9].

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

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 hierarchical routing schemes which achieve this goal. Fultz [8], McQuillan [9], Gerla and others [11] proposed similar schemes but did not evaluate their performance as we do here.

*We do not consider the case where packets carry their own routing information.

1.2 THE DESIGN OF COMPUTER NETWORKS

We are interested in designing the topology of a large network under some cost and performance constraints. Several different formulations of the design problem related to the communication subnetwork can be found in the literature [4, 5]. Generally they correspond to different choices of performance measures, design variables and constraints. Here, we select the following very general formulation.

Given: node locations, channel capacity options

Minimize: total communication cost

Over: topology, channel capacities, routing policies

Subject to: delay constraint, reliability constraint, traffic requirement

In general, there are $2^{N(N-1)/2}$ possible topologies. Furthermore, capacities are available in discrete sizes. This means that an enormous integer optimization problem must be solved. The non-linearity of the time-delay functions [2, 7] and, in some cases, of the reliability measure [4] add another dimension of complexity to the problem.

There exists no efficient technique for the exact solution of this topological design problem. Several heuristic procedures have been proposed and implemented. Among them, we mention the Branch X-change method, the Cut-Saturation method [4] and the Concave Branch Elimination method [5]. Typically, they start with an initial topology over which they perform some alterations in the course of the optimization. Built into those procedures and inherent in the multicommodity nature of the flow, is the determination of the shortest path between any pair of nodes in the network. This operation requires between N^2 to N^3 operations (N = number of nodes) and may be performed many times in the course of the optimization. The overall computational complexity corresponding to those heuristics is estimated to be on the order of N^3 to N^6 [4, 11, 12].

For networks with more than a few hundred nodes, present procedures fail because of the large amount of computer time and storage needed to perform the suboptimization. As a result new approaches are needed to deal with the design of large networks. Such an approach, using a hierarchical design technique, is presented and studied in this paper.

Throughout this paper we state only results and omit all proofs. The proofs, extensions and other numerical results can be found in [6].

2. HIERARCHICAL ROUTING SCHEMES

2.1 METHODOLOGY

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 <u>set</u> of destinations for more remote nodes.

In large networks the reduction of routing information is realized through a hierarchical clustering of the network nodes. Basically, an <u>m</u>-level <u>h</u>ierarchical <u>c</u>lustering (MHC) of a set of nodes consists of grouping the nodes (which we shall define as 0-th 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 m-th level cluster. The m-th level cluster is the highest level cluster and as such it includes all the nodes of the network.

Since hierarchical routing schemes are based on an m-level heirarchical clustering, they will be denoted as MHR schemes. With the MHR schemes, only <u>one</u> 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 k-th 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.

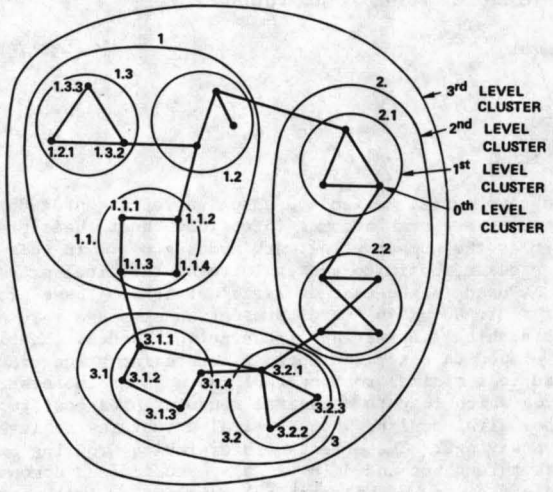

Figure 1. A 3-Level Clustered 24-Node Network.

The clustering leads to the tree representation shown in Fig. 2, where nodes are identified using the Dewey notation. To each node we now associate a reduced routing

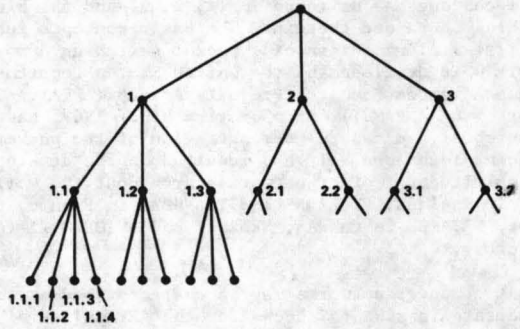

Figure 2. A Tree Representation of a 3-Level Clustered Net.

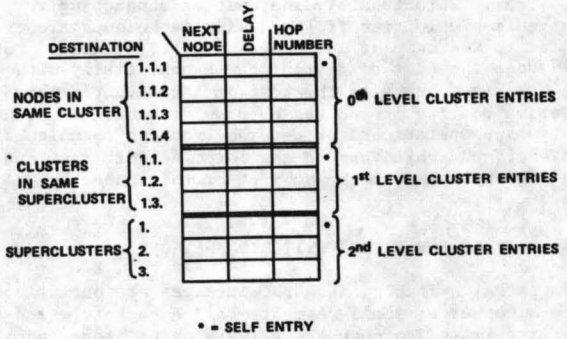

Figure 3. Routing Table of Node 1.1.1.

table. Figure 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 proceeds 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 0-th level cluster entry, number 2 (3.2.2). Note that the above construction of the routing tables implies that traffic between nodes in the same cluster (at any level) must follow paths included in that cluster's subnet. Consequently it is assumed that the clustering results in connected subgraphs.

In the rest of this section, we consider the two questions below:

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 and their comparison with the present non-clustered policies.

2.2 MINIMUM ROUTING INFORMATION

In what follows we first consider a 2-level hierarchical clustering; then we generalize to an arbitrary number of levels, m.

Consider a 2-level hierarchical clustering composed of n_2 1st level clusters. Let i_2 ($i_2 = 1, 2, \ldots, n_2$) denote an arbitrary 1st level cluster, and $n_1(i_2)$ be the corresponding number of nodes (0-th level cluster). In order to simplify the manipulation and implementation of the routing tables in the network, we assume that equal table lengths are provided at all nodes. Consequently if ℓ is that length, it must be such that

$$\ell = \max_{i_2}\{n_2 + n_1(i_2)\} \qquad (1)$$

Also, the total number of 0-th level clusters must be equal to the total number of nodes in the network, N.

$$N = \sum_{i_2=1}^{n_2} n_1(i_2) \qquad (2)$$

Let $\underset{\sim}{n} = \{n_1(i_2) \quad i_2 = 1, \ldots, n_2 \; ; n_2\}$ be the degree vector which describes our clustering structure, then the optimal structure is the solution of the following problem:

$$\begin{aligned} &\text{Given:} && N \\ &\text{Minimize:} && \ell \quad \text{[see Eq. (1)]} \\ &\text{Over:} && \underset{\sim}{n} \quad \text{positive, integer} \\ &\text{Subject to:} && \text{Eq. (2) holds} \end{aligned} \qquad (3)$$

The above formulation can be easily extended to the general case of an m-level hierarchy whose optimal structure is given below.

PROPOSITION 1

Given m, the number of levels in the hierarchy, and assuming a real valued degree vector, the optimal clustering structure is such that:

(a) All k-th level clusters are composed of an equal number of (k-1)st level clusters n_k, and this for $k = 1, \ldots, m$.

(b) All degrees n_k are equal

$$n_k = N^{1/m} \qquad k = 1, \ldots, m \qquad (4)$$

As a result, the minimum table length is

$$\bar{\ell} = mN^{1/m} \qquad (5)$$

If we now let m be an optimization variable which assumes real values, then the global optimum clustering structure is achieved when the number of levels is

$$m_* = \ln N \qquad (6)$$

and when the degree vector is

$$n_k^* = e = 2.718... \qquad k = 1, 2, \ldots, m_*$$

The corresponding minimum table length is

$$\ell_* = e \ln N \qquad (7)$$

We now proceed with some numerical examples. The ratio ℓ/N of the new table length ℓ to the one obtained with no clustering N constitutes here the performance measure by which we characterize the gains obtained from hierarchical routing. In reality, one needs to express those gains in terms of recovered nodal storage, line capacity, CPU, and ultimately in terms of network throughput and delay. This we defer until later. It is the behavior of ℓ/N at optimality that we intend to display. Fig. 4 illustrates the behavior of $\bar{\ell}/N$ (see Eq. (6)) with respect to m and for several values of N. It shows that very significant savings can be achieved.

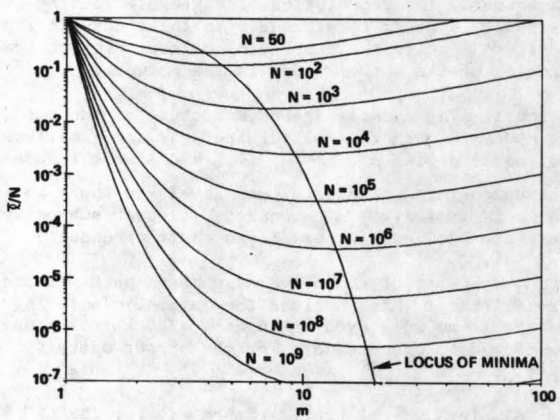

Figure 4. Minimum Relative Table Length, ℓ/N, Given m.

Note that $\bar{\ell}/N = 1$ for $m = 1$; this corresponds to the degenerate 1-level hierarchical routing which is simply our original non-clustered scheme. For m varying from 1 to ln N, $\bar{\ell}/N$ decreases to values quite a bit smaller than 1. For m greater than ln N, $\bar{\ell}/N$ 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 $\bar{\ell}/N \geq 1$ 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 $m = N$, $\bar{\ell}/N = N^{1/N}$ whose limit is 1 when N goes to infinity. The curves exhibit a very flat region around the minimum. They also show an initial fast decrease of $\bar{\ell}$ toward a value close to the minimum. 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. (6)). Such a property proves to be very valuable (see below).

In [6] the integer case is also solved. Numerical examples 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.

The price we pay for this table reduction is an increase in the network path length. Next we examine the effect of hierarchical routing on path length.

2.3 PATH CHARACTERISTICS FOR HIERARCHICAL AND NON-HIERARCHICAL ADAPTIVE ROUTING POLICIES

The purpose of this section is to characterize the routes obtained from the routing tables under certain equilibrium conditions. 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. In order to simplify the analysis, we will assume that all channels are of constant length (one hop), which allows 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 [7].

With the MHR schemes 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 "exchange" nodes in different clusters at any level. Two MHR schemes, referred to as the Closest Entry Routing (CER) and the Overall Best Routing (OBR), are presented below. They differ only in the definition and subsequently the computation of aggregate routing information. In order to proceed with their description, we must first specify the underlying m-level hierarchical partitioning of the set of nodes in 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 sub-network (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 was found to be necessary for the proper operation of the MHR's.

CER and OBR Hierarchical Routing Schemes. For the CER scheme a cluster is regarded from the outside as a single (super-) node whose distance to itself is equal to zero. For the OBR 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. The self entries in the RT at an exchange node (see Fig. 3) may be assigned to carry the aggregate routing information from one cluster to another. Note that a unique "degenerate" MHR scheme, the Non-Clustered Routing (NCR) scheme, corresponds to either the OBR or the CER schemes with only one hierarchical level (m = 1).

Update Rule. Upon reception of an update from a neighboring node (s), the receiving node (t) compares its present routes with the ones computed using paths with s as the next node. The best paths are then kept and the RT's entries are updated accordingly. Because of the structure of the tables (see Fig. 3) not all entries in s and t's tables are common destination entries. The updating then, concerns only those common destination entries. For any pair of nodes s,t, the common region can be determined by inspecting the address vectors of s and t. For the degenerate NRC scheme all the nodes belong to the same unique lst level cluster; hence the RT's contain only common destination entries and as expected, the updating is performed for all entries.

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.

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. The question arises as to what is the meaning of the routing information at equilibrium, that is, what are the characteristics of the paths indicated by the routing tables. We 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, two more definitions 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.)

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.

In what follows we restrict our considerations to the CER scheme; however the bounds derived below are also valid for OBR (see [6]).

Consider two arbitrary s and t nodes which belong to the same k-th level cluster 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 and for a CER scheme satisfies the recursive equation below

$$h^c_{st} = h^i_{se_s} + h^c_{e_s t} \tag{8}$$

where e_s is an exchange node of the (k-1)st level cluster $C_{k-1}(t)$ which contains t, and which is the closest to s, i.e.,

$$h^i_{se_s} = \min_{\substack{e:\ \text{exchange nodes} \\ \text{of } C_{k-1}(t)}} \{h^i_{se}\} \tag{9}$$

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.,

$$D = (h_c/h) - 1 \tag{10}$$

where h_c and h denote the average path length in the network respectively with and without clustering. With a uniform traffic assumption, we have

$$h = \frac{1}{N(N-1)} \sum_{s,t \in S} h_{st}\ ,\quad h_c = \frac{1}{N(N-1)} \sum_{s,t \in S} h^c_{st} \tag{11}$$

Eqs. (8) and (9) provide 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 possible to numerically evaluate the relative increase D 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* of any k-th level cluster subnet (see Assumption 1) is less than or equal to a known quantity d_k, k = 1, ..., m. Note that d_m represents the diameter of the entire network and that $d_k > d_{k-1} \geq 0$ 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

*Recall that the diameter of a network is the maximum shortest path between pairs of nodes.

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 some simple bounds. The first is on individual paths,

$$h^c_{st} \le \sum_{j=1}^{k} d_j \qquad \forall\, s, t \in \text{same } k^{th} \text{ level cluster}$$
$$\forall\, k = 1, 2, \ldots, m$$

The above leads to a bound on the increase in average path length,

$$h_c - h \le \sum_{k=1}^{m-1} \left[1 - \frac{n_1 n_2 \cdots n_k - 1}{N - 1}\right] d_k \qquad (12)$$

If we relax Assumption 3 then we arrive at

$$h_c - h \le \sum_{k=1}^{m-1} d_k \qquad (13)$$

Next, we study the behavior of some of these bounds in the context of a defined class of networks.

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 have 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 than $d \le bn^v + c$, where b, c, v, are positive parameters and $0 \le v \le 1$ (see below).

If N is the size (i.e., number of nodes) of such a network, then the average path length (hop distance) of that network, h, must be a power law function of N, i.e., $h = aN^v$ where a is a positive parameter. Grid type networks, hexagonal networks, triangular networks, geodesic, etc., fall into that category when the MHC results in subnetworks of a similar structure as the original and when the path lengths are expressed in hops. We consider the special case of grid and torus networks. Expressions for their average path lengths (with a uniform traffic matrix) and for their diameters have been derived in [6]. Some of the results obtained are for a square grid of size N, $h = 2/3\, N^{1/2}$, $d = 2N^{1/2} - 2$; and for a square torus of size N, with $N^{1/2}$ an odd integer, $h = N^{1/2}/2$, $d = N^{1/2} - 1$. 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 $d \le 2\sqrt{n} - 2$. As a consequence the grid and torus networks fit the above descriptions.

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

The main characteristic of most distributed networks such as the ARPANET, AUTODIN II, CYCLADES, TRANSPAC, EPSS, EIN, DATAPAC, TELENET 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. The torus networks considered above are of connectivity 4 and with an exponent $v = 1/2$, 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. Below, 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, v as obtained for the torus net, i.e.,

$$a = 1/2 \quad b = 2 \quad c = -2 \quad v = 1/2 \qquad (14)$$

Performance Evaluation of the MHR Schemes: The family of networks considered here satisfies Assumptions 1 — 3, hence Eqs. (12) and (13) hold true. Let E be defined as the bound on the relative increase in path length D (see Eq. (10)). 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 $\underline{n}$ must satisfy Eq. (4). Then from Eqs. (4) and (12) and for this class of networks

$$0 \le \frac{h_c}{h} - 1 \le E \triangleq \frac{1}{a(N-1)N^v}\left[N\left[b\,\frac{N^v - N^{v/m}}{N^{v/m} - 1} + c(m-1)\right]\right.$$
$$\left. - b\,\frac{N^{v+1} - N^{(v+1)/m}}{N^{(v+1)/m} - 1} - c\,\frac{N - N^{1/m}}{N^{1/m} - 1}\right] \qquad (15)$$

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

$$\frac{\ell}{N} = \frac{mN^{1/m}}{N} \qquad (16)$$

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

PROPOSITION 2: ASYMPTOTIC 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.,

$$N \to \infty \Rightarrow h_c/h \to 1\,, \quad \ell/N \to 0 \qquad (17)$$

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. (15) around N^{-1}, we find

$$E = b/a\, N^{-v/m} + 0(N^{-v/m})$$

hence, since $v/m > 0$, $\lim_{N\to\infty} E = 0$. Also the second limit is obvious.

Q.E.D.

Note that the closer v is to one ($v \neq 0$), the faster is the convergence of E to zero. In other words, as could be expected, the more distributed (the less connected) the networks are the better the MHR's perform. The above result holds true if we relax Assumption 3. In this case we use the bound in Eq. (13).

The result of Proposition 2 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 globally optimum clustering structure. That value is, from Eq. (6), $m_* = \ln N$. Let E_* be the value of E for $m = \ln N$, of interest is the limit of E_* as N goes to infinity.

$$\lim E_* = \frac{b}{a}\left[\frac{1}{e^v - 1} - \frac{1}{e^{1+v} - 1}\right]$$

As a consequence the result of Proposition 2 is not necessarily true when m is variable. If we consider the coefficients of Eq. (14), 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 2.2 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 very small increase from the minimal table length for a tremendous improvement in performance. This fact is illustrated in Fig. 5 where we show

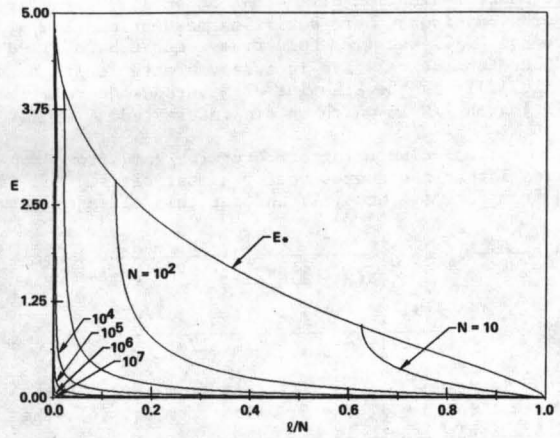

Figure 5. Bound on the Relative Increase in Path Length E, Versus the Relative Table Length ℓ/N.

the behavior of E with respect to ℓ/N and this for several values of N. We observe that 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. (7), then E increases sharply. It is the sharp region of the curve that we must avoid in order to keep the increase in path length significantly low. Fig. 5 also shows the behavior of E_* versus ℓ/N.

So far we have examined the effect of hierarchical routing on network path length. Bounds were derived to evaluate a maximum increase in path length for a given table reduction. Furthermore, the bounds demonstrate that no significant increase in path length need be incurred in the limit of a very large network.

The reduction in table length means that more channel capacity and storage are available for the transmission of data traffic in the network. However, those gains were obtained at the expense of longer paths in the network. It is natural for us next to evaluate the performance of hierarchical routing in terms of delay and throughput, and to define the region of N where clustering becomes economical. These questions are the subject of the next section.

2.4 STOCHASTIC PERFORMANCE EVALUATION OF THE HIERARCHICAL ROUTING

In this section we are interested in the trading relations among the table reduction, the nodal storage, the channel capacity, the network size, the throughput and the delay. Several queueing models are developed to capture and exhibit the interrelationships among these variables. The models demonstrate that for some reasonable cost and performance constraints and for a class of symmetrical and distributed networks, the non-hierarchical routing becomes infeasible for N (network size) beyond some "critical" value; on the other hand they show that hierarchical routing, operating with an appropriate table length, is capable of maintaining a fairly good network performance for fairly large values of N.

Our point of departure is the major result for delay analysis in networks, developed by Kleinrock [7]. An extension of Kleinrock's model is then presented to account for line overhead due to routing updates. Such a model will then be used to evaluate the performance of hierarchical routing. Other models which account first for storage, and then both for storage and capacity are developed and used to evaluate hierarchical routing in [6].

Delay Analysis in S/F Computer Networks: Kleinrock's Model: A very important performance measure of an S/F net is the total average delay T a message spends in the network. T may be expressed in terms of the individual channel delays [7],

$$T = \sum_{i=1}^{NA} \frac{\lambda_i}{\Gamma} t_i \qquad (15)$$

Γ is defined as the total input rate (throughput), $\Gamma = \sum_{j,k} \gamma_{jk}$; where γ_{jk} is the average message rate from source j to destination k. Also, λ_i is the average traffic rate on channel i; t_i is the average nodal processing plus queueing plus transmission time on the i^{th} channel; and finally, NA is the number of channels in the net. Unfortunately, we are not able in general to evaluate t_i and λ_i. However, the analysis may proceed if we make the following assumptions: external Poisson arrivals, exponential message length distribution (identical for all messages), single packet messages, error free channels, no nodal delay, independence assumption, deterministic routing, infinite nodal storage. With such assumptions, the S/F net can be modelled as a network of queues of the Jackson type. In particular, each queue behaves as an independent M/M/1 queue. As a result, the average delays t_i and T are

$$t_i = \frac{1}{\mu C_i - \lambda_i} \quad , \quad T = \frac{1}{\Gamma} \sum_{i=1}^{NA} \frac{\lambda_i}{\mu C_i - \lambda_i} \qquad (16)$$

where $1/\mu$ is the average message length [KB/msg] and C_i is the capacity of channel i [KBPS]. The average rates λ_i i = 1, ..., NA, can be numerically computed, given the underlying deterministic routing.

A simple relationship exists between the total internal traffic $\Lambda = \sum_i \lambda_i$, the total external traffic Γ, and the average weighted (with traffic) path length $\bar{n}$.

$$\bar{n} = \Lambda/\Gamma$$

This terminates the presentation of the main results in network delay analysis [7]. Further extensions and discussions can be found in [2]. A limitation of the above model is that it does not account for the nodal storage requirements and line overhead due to the routing updates These issues become critical in large networks where the line overhead and storage associated with routing becomes excessive.

A Queueing Model with Updates and No Storage Limitation: In this section we account for the traffic generated by the routing updates while keeping the infinite storage assumption. As noticed earlier, the average delay in the network is very simply related to the average delay at any channel (see Eq. (15)); therefore, we will first analyze a single channel (i) and then generalize to a net.

A simple and realistic Head of Line (HOL) model [6] 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 ARPANET). Aperiodic updates may be modelled by a "no update" model [6] or by a certain distribution governing their generation times. The latter possibility can be easily included in the model below 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 of rate λ_u, assumed to be the same on all channels. Constant message length $1/\mu_u$ KB.

ii. Data traffic: Poisson arrival process of rate λ_i (channel i). Exponential message length of mean $1/\mu$ KB.

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

iv. Channel capacity: C_i 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 the

* A data message is differentiated from an update message.

types of customers (messages) to be governed by a Poisson distribution. However, the methodology can still be used here in order to derive the average time in system for a data message.

With regard to the update traffic, it simply sees a D/D/1 system; hence as long as $\lambda_u < \mu_u C_i$ there is no queueing of update messages; whereas, an arriving data message will incur a delay from the message (data or update) already in service, from data messages already in the queue and from updates arriving during its system time. This yields

$$t_i = \frac{1/\mu C_i + \lambda_u/2(\mu_u C_i)^2}{1 - \lambda_i/\mu C_i - \lambda_u/\mu_u C_i} \qquad (17)$$

If we set $\lambda_u = 0$ in the above equation (i.e., if we neglect updates) then we arrive at the original expression for t_i Eq. (16). The difference between the two equations illustrates the effect of the updates. Furthermore, the substitution of Eq. (17) into Eq. (15) yields the average delay in the net.

This delay analysis relies on the knowledge of the input rates (λ_i's) to the individual channels in the network. As mentioned earlier, the λ_i's can be determined once we know the deterministic routing policy and the traffic requirement. Moreover, if we know the channel capacities then the λ_i's can be computed to lead to a minimal delay T [5]. A shortcoming of the numerical procedure is that, in general, it hides the interrelationships existing among the different design variables (traffic requirement, channel capacities, network topology and average delay). Fortunately, for some symmetrical networks (see below) a simple analytical relationship exists among the above variables.

A Class of Symmetrical Networks. The class of nets to be considered in this Section is composed of all the nets which belong to the family of nets presented in Section 2.3 and which also satisfy the following properties: (i) All nodes are equivalent with respect to the topology of the network. Hence they are of equal degree, R. (ii) All channels are of equal capacity, C. (iii) All external input traffic rates are equal, i.e., $\gamma_{jk} = \gamma$ $\forall$ j,k. As an example, torus nets fall into this category. For this class of nets, the following relations exist: number of (simplex) channels: $NA = R \cdot N$; total external traffic: $\Gamma = N(N-1)\gamma$.

Furthermore, it is obvious that with this particular topological structure, capacity assignment and traffic requirement, the optimal flow assignment [5, 7] is a shortest path routing. The selection of the particular shortest paths (when more than one exists) must result in perfectly balanced flows, i.e., $\lambda_i = \lambda$ i = 1, ..., NA. Consequently the network path length $\bar{n}$ becomes the average shortest path length h, defined previously; therefore we obtain $\bar{n} = \Lambda/\Gamma = h$. Also the total internal traffic Λ becomes $\Lambda = (NA)\lambda$. Combining the last two equations, we arrive at $\lambda = h\Gamma/NA$. If we let t denote the average delay on any channel, then the total average delay becomes (Eq. (15)) $T = ht$. Finally, because of Eq. (17), we arrive at

$$T = h \frac{1 + \frac{\mu}{\mu_u}\frac{\lambda_u}{2\mu_u C}}{\mu C - h\frac{\Gamma}{NA} - \frac{\mu}{\mu_u}\lambda_u} \qquad (18)$$

This is the result we were seeking; it simply relates the delay T, the traffic Γ, the channel capacity C, the network path length h and the update function λ_u, μ_u. We now proceed with the performance evaluation of the hierarchical routing.

Performance Evaluation of Hierarchical Routing. The direct analysis of any adaptive routing scheme is far too complex because of the dynamic nature of the routing. In the face of these difficulties, we model the hierarchical adaptive routing by an "equivalent" deterministic routing; hence the assumption below:

Assumption 4: (a) The performance of an adaptive hierarchical routing is the same as that of a fixed (deterministic) routing policy which yields paths of equal length as the minimum estimated path lengths obtained with an MHR. (b) The fixed routing specified above and operating on the class of symmetrical nets considered here, results in equal loads on all channels.

Part (a) will become more accurate when we include in the fixed routing model, the line and storage utilization due to the adaptive routing. Moreover, if the main objective is to compare hierarchical with non-hierarchical routing, then this assumption appears to be quite acceptable. Part (b) is motivated by the 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, because of the above assumption, the NCR (MHR with m = 1) is modelled by the shortest path fixed routing which, as observed earlier, leads to the optimal flow assignment for this class of nets.

As a result of the above assumption and because of Eq. (18), in order to characterize the performance of the MHR we need only to know the average path length h_c. Given the relative table length $\ell/N = mN^{1/m}/N$ (or equivalently given m) we can use our previous bound E, Eq. (15), to obtain a lower and an upper bound on the network performance. Since the main objective here is to study routing in large nets, it is necessary to specify the structure of those large nets with respect to the size N, in some continuous way. Any such specification will be referred to as a scaling scheme (or strategy).

A Scaling Scheme. As the network grows, a main objective of a scaling strategy could be to maintain the same average delay T for a reasonable increase of the total traffic Γ and of the network cost (channel capacity cost). The total traffic Γ may be reasonably assumed to increase linearly with the number of nodes, which, due to the uniform traffic condition ($\gamma_{jk} = \gamma$) is equivalent to assuming that the total input rate per node is maintained constant, i.e., Γ/N = constant. Also, since $NA = R \cdot N$ this also means that Γ/NA is maintained constant. If we do not account for updates and assume an NCR scheme, then a scaling scheme which achieves that objective is

$$T = T_0 \; ; \; C = hC_0 \qquad (19)$$

Substituting the above into Eq. (18)where λ_u is set to zero we arrive at

$$\Gamma/NA = \mu C_0 - T_0^{-1} \qquad (20)$$

which is a constant.

If we account for updates then our scaling scheme no longer leads to a constant Γ/NA, but Γ/NA will be a function of λ_u and μ_u. It is the behavior of Γ/NA that we will study. Before we proceed, we need to specify λ_u in terms of network growth. We consider the three choices below.

i. $\lambda_u = \lambda_u^0$ = constant

ii. $\lambda_u = h\lambda_u^0 = aN^{\nu}\lambda_u^0$ $\qquad \frac{N^{1/2}}{2}\lambda_u^0$ for a torus

iii. $\lambda_u = a\,N^{\nu/2}\,\lambda_u^0$ $\qquad \frac{N^{1/4}}{2}\lambda_u^0$ for a torus

Choice (i) represents a worst-case condition whereby the update rate is insensitive to the increase 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.

In the numerical application below we restrict our considerations to the more conservative choice (i). In [6], the other two choices are also considered. The behavior of hierarchical routing may now be studied for our symmetrical networks whose growth is governed by the above scaling schemes. Let T_c and Γ_c be respectively the delay and throughput with an MHR. Because of Assumption 4, Eq. (18)

holds true if we replace T by T_c, Γ by Γ_c and h by h_c. The size of an update $1/\mu_u$, is fixed and proportional to the length of the routing table ℓ (Eq. (5)). Let it be of the form $1/\mu_u = \varepsilon\ell$ where ε is the inverse of the number of entries which add to 1 kbt. As an example, in the ARPANET, an entry requires 16 bits of storage, hence $\varepsilon = .016 \simeq 1/62.5$. For further normalization with respect to the average data message $1/\mu$, we choose ε such that $1/\mu_u = \varepsilon\ell/\mu$. With the scaling scheme T_c is maintained constant and $C = hC_0$, hence the throughput Γ_c over NA is given by

$$\frac{\Gamma_c}{NA} = \frac{h}{h_c}\mu C_0 - \frac{1}{T_0} - \frac{\varepsilon\ell\lambda_u}{h_c} - \frac{\lambda_u}{2\mu C_0 T_0}\frac{\varepsilon^2\ell^2}{h} \qquad (21)$$

For any routing to be feasible, the right hand side of the above equation must be positive. As the number of nodes goes to infinity ($N \to \infty$), from among the above scaling schemes of λ_u only the 1st and 3rd may be feasible. The feasibility can be achieved with MHR schemes with a fixed number of levels m, greater than 2 for the 1st choice of λ_u and greater than 4 for the 2nd. This is due to the results obtained in Proposition 2. Consequently the non-clustered scheme (i.e., m = 1) is in the limit, not feasible for any of the above choices of λ_u (for it to be feasible λ_u must be a decreasing function of N). For the feasible schemes, the limiting throughput is $\lim_{N\to\infty} \Gamma_c/NA = \mu C_0 T_0^{-1}$ which is equal to the one obtained without updates, Eq. (20). Thus, for those MHR's in the limit, the effect of the updates on the channel utilization becomes negligible.

Let us now examine the general behavior of Γ_c/NA with respect to ℓ/N by plotting its lower bound as derived using Eqs. (15) and (21). The lower bound is normalized with respect to the maximum throughput given by Eq. (20). The values selected for the different variables are: $\mu C_o = 6$ msg/sec, $T_0 = .5$ sec, $\lambda_u = \lambda_u^0 = .07\ \mu C_0$ and $\varepsilon = 1/64$.

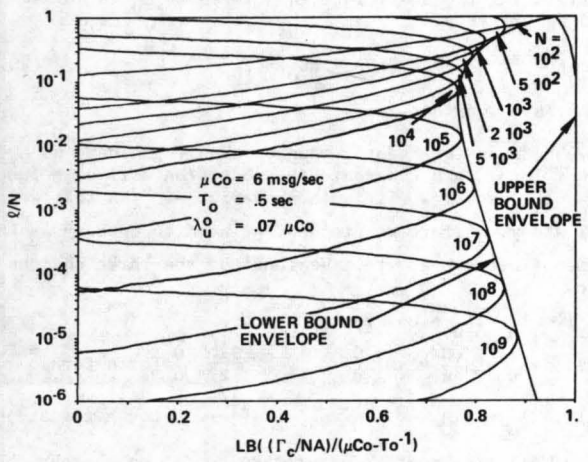

Figure 6. Lower Bound Throughput at Constant Delay Versus the Relative Table Length : Model With Updates $\lambda_u = \lambda_u^0$.

Fig. 6 shows the behavior of the normalized lower bounds with respect to ℓ/N and for a set of values of N. Lower and upper bound envelopes are also plotted. The lower bound envelope shows an initially decreasing and then slowly increasing behavior with respect to N; the increase will eventually bring the curves closer to their asymptote 1. However for $\lambda_u = N^{1/2}\lambda_u^0/2$, a case which we do not consider here (see [6]), the lower bound envelope is a decreasing function of N which, as predicted earlier, reaches zero for N beyond 10^8.

We note that for a given N there is an optimal ℓ/N which achieves a maximum lower bound throughput. We also note that for N greater than approximately 2000 (or 1000 in the case $\lambda_u = \lambda_u^0 N^{1/4}/2$) the non-hierarchical scheme becomes infeasible. Whereas, hierarchical routing with an optimal ℓ/N succeeds in maintaining a remarkably good network performance. The curves show that MHR clearly becomes superior to NCR for $N \geq 500$. In other plots [6] we note that MHR surely becomes more efficient for even smaller values of N (between 100 and 200). Moreover, a simulation study of a 64-node torus confirmed our theoretical results and showed that even for such a small size MHR achieves a (slightly) better performance than NCR. Furthermore, other models have been developed in [6]; they account first for storage and then for both storage and capacity required by the updates. Those models demonstrate similar properties as above, a remarkable efficiency of hierarchical routing for large networks. This concludes our study on routing for large networks.

3. HIERARCHICAL DESIGN OF COMPUTER NETWORKS

3.1 METHODOLOGY

Recall that a simple extrapolation of present topology design procedures becomes prohibitive in the context of large networks. A hierarchical design procedure is presented here in order to alleviate the tremendous computational design cost of large nets. The emphasis is on the determination of a clustering structure to be used in the design phase and which minimizes the computational cost of the design. The main idea behind the hierarchical design is to impose a decomposable structure on the design problem which will result in a set of smaller subproblems. In other words, we will introduce independencies among subsets of design variables. The imposed independencies will substantially reduce the set of feasible solutions and also, as a direct consequence, the computational cost. In doing so, there is the risk of eliminating the optimal solution. Therefore, it is very important to seek "natural" decompositions. Such decompositions will be realized through an m-level hierarchical clustering (MHC) of the set of nodes, based on some appropriate nearness measures. Again, because of the underlying MHC structure the m-level hierarchical topology design procedure will be denoted by MHT. Along with the hierarchical clustering of the nodes we must select the gates (exchange nodes) for all clusters at all levels. The function of the gates from a given cluster is to handle the traffic exchanged between the set of nodes in that cluster and those outside. More specifically, the assumption underlying the flow of messages is as follows.

Assumption 5: (a) Traffic between nodes in the same cluster at any level will only take paths which are internal to that cluster, i.e., paths contained in the corresponding local subnetwork. (b) Traffic between nodes in different k-th level clusters (k = 1, ..., m - 1), but which belong to the same (k+1)st level cluster, will first be channeled to a (k+1)st level gate of the originating cluster over its local subnetwork; then, it will take the (k+1)st layer subnetwork of gates to reach a (k+1)st level gate of the destination cluster, at which point it will be dispatched over the local subnetwork to finally reach the destination node. (This is the standard procedure in hierarchical networks.)

A k-th layer subnetwork is defined as a network connecting k-th level gates which belong to the same k-th level cluster. Once the hierarchical classes are defined and the gates selected, then the previously developed network design techniques for moderate sized networks may be used to design the layer subnetworks separately.

Several questions arise as to the optimal clustering structure, the decomposition of the global performance variables and requirements which then lead to a set of smaller design problems. Frank and others [4, 12] showed from a feasibility study of a 1000 mode network that indeed, hierarchical structures are desirable for the design of large networks. They also posed the same questions concerning the clustering structure, but failed to answer them for the general case of an arbitrary number

of hierarchical levels. Such questions will be addressed below.

Here also, the direct application of the clustering techniques may lead to various non-optimal cluster sizes which will, in general, considerably reduce the computational gains obtained from optimal size clusters. It is then important to determine those MHC structures which will minimize the computational cost incurred in the design phase of the MHT. In order to evaluate this cost, we make the following assumption.

Assumption 6: (a) The computational cost incurred in the design of a k-th layer subnet connecting a set of n k-th level gates is equal to n^{α_k} ($k = 1, 2, \ldots, m$). (b) The total computational cost involved in the design is equal to the sum of the costs induced in the design of all the layer subnets.

The polynomial form of the computational cost is the one normally used [4] to characterize the computational complexity of most of the present design algorithms. The fact that different exponents α_k's could be selected, depending on the level of the hierarchy, is provided to allow the modelling of the design of hierarchical networks where different technologies or design algorithms or both are considered at each level or group of levels.

For the assignment of gates, given an integer vector $\underset{\sim}{\beta} = (\beta_1, \beta_2, \ldots, \beta_m)$ where $\beta_1 \triangleq 1$, and a selection rule, then, starting at $k = 1$, select β_{k+1} (k+1)st level gates among the set of k-th level gates of each k-th level cluster. Repeat this step sequentially until $k = m - 1$. A network node is considered to be a 1st level gate. The choice of the vector $\underset{\sim}{\beta}$ will be mainly related to the reliability constraint. If a K-connectivity is to be imposed on the topology of the network, then the vector $\underset{\sim}{\beta}$ must be such that $\beta_i \geq K$ for $i = 2, 3, \ldots, m$. This is obvious since the set of the β_i i-th level gates of an (i-1)st level cluster represents a cut set for the other nodes in that cluster. As an example, for a centralized hierarchical network $\underset{\sim}{\beta} = (1, 1, \ldots, 1)$. We are now ready to find the expression of the total computational cost and solve for the optimal clustering structure.

3.2 MINIMUM COMPUTATIONAL COST IN THE HIERARCHICAL DESIGN OF NETWORKS

As in Section 2.2 we first consider a 2-level hierarchical clustering characterized by $\underset{\sim}{n} = \{n_1(i_2)\ i_2=1, \ldots, n_2;\ n_2\}$ and then generalize to an arbitrary number of levels m. Let G_k be the computational cost of the design of all the k-th layer subnets ($k = 1, 2$). There are n_2 1st layer subnets, hence because of the above assumptions

$$G_1 = \sum_{i_2=1}^{n_2} [n_1(i_2)]^{\alpha_1}$$

There is also a unique 2nd layer subnet connecting all 2nd level gates. From the gate assignment rule, each 1st level cluster contributes β_2 2nd level gates, hence $G_2 = [\beta_2 n_2]^{\alpha_2}$. The total cost G is then equal to $G_1 + G_2$. Also the degree vector n must satisfy Eq. (2). The optimal structure is the solution of:

Given: N

Minimize: $G = G_1 + G_2$

Over: $\underset{\sim}{n}$ positive, integer-valued vector

Subject to: Eq. (2) holds

The above formulation can be easily extended to the general case of an m-level hierarchical clustering. With different α_i's and β_i's, the expression of the optimal solution is fairly complicated and as such it is not reproduced here (see [6]). However, if we assume equal α_i's and β_i's then the following proposition holds true.

PROPOSITION 3

Given m, the number of levels in the hierarchy, and assuming that $\alpha_i = \alpha$ for $i = 1, \ldots, m$; $\alpha > 1$; and $\beta_i = \beta$ for $i = 2, \ldots, m$ (recall $\beta_1 \triangleq 1$); then the optimal solution is such that (a) all k-th level clusters are composed of an equal number of (k-1)st level clusters n_k and this for $k = 1, \ldots, m$. (b) The reduced degree vector $\underset{\sim}{n} = (n_1, n_2, \ldots, n_m)$ is given by

$$\begin{cases} n_1 = \dfrac{\alpha}{\alpha - 1}\left[\left(\dfrac{\alpha - 1}{\alpha}\right)^m \dfrac{N}{\beta}\right]^{\frac{(\alpha-1)^{m-1}}{D_{m+1}}} \\ n_k = \dfrac{\alpha}{\alpha - 1}\left[\left(\dfrac{\alpha - 1}{\alpha}\right)^m \dfrac{N}{\beta}\right]^{\frac{(\alpha-1)^{m-k}\alpha^{k-1}}{D_{m+1}}} \end{cases} \qquad k = 2, \ldots, m \tag{22}$$

where $$D_k = \alpha^{k-1} - (\alpha - 1)^{k-1} \quad \text{for} \quad k \geq 1 \tag{23}$$

With this solution, the minimum computational cost is:

$$\underline{G}(m, \alpha, \beta) = D_{m+1}\left[\left(\beta \frac{\alpha}{\alpha - 1}\right)^{\alpha(\alpha-1)D_m} \times \left(\frac{(\alpha - 1)^{(\alpha-1)^m}}{\alpha^{\alpha^m}}\right)^{m-1} N^{\alpha^m}\right]^{\frac{1}{D_{m+1}}} \tag{24}$$

We note that, with the MHT the computational cost is reduced from the order of N^α to $N^{\alpha^m/(\alpha^m-(\alpha-1)^m)}$. Also, if $\alpha < 1$ then a non-hierarchical procedure is optimal [6].

So far we have solved for the optimal clustering structure when m, the number of levels, is fixed. If we let m vary and be a real variable, then the global optimum clustering structure is achieved for a number of levels

$$m_* = \frac{\ln (N/\beta)}{\ln(\alpha/(\alpha - 1))} \tag{25}$$

and $$n_1^* = \beta \frac{\alpha}{\alpha - 1}, \quad n_k^* = \frac{\alpha}{\alpha - 1} \quad k = 2, 3, \ldots, m_* \tag{26}$$

The corresponding minimum computational cost is

$$G_*(\alpha, \beta) = \left(\frac{N}{\beta} - 1\right)\beta^\alpha \frac{\alpha^\alpha}{(\alpha - 1)^{\alpha-1}} \tag{27}$$

Below we make some remarks about the global minimum solution. We note that, at global optimality the computational cost is reduced to the order of N. Also if we let g_k denote the size of a k-th layer subnet then from our previous assumptions $g_1 = n_1$; $g_k = \beta n_k$ $k = 2, \ldots, m$.

Thus, at global optimality $g_k^* = \beta \frac{\alpha}{\alpha - 1}$ $k = 1, 2, \ldots, m_*$

This indicates that at optimality, all the layer subnets are of an equal size which depends only on α and β. An intuitive explanation of this very simple and interesting property is given in [6]. Moreover, in the search for optimal clustering structures we purposely omitted an additional constraint on the degree vector. This constraint results from the gate assignment rule and is such that $g_k \geq \beta \Rightarrow n_1 \geq \beta$, $n_k \geq 1$ $k = 2, \ldots, m$. From Eqs. (22) and (25) we see that the above constraint is fortunately always satisfied at optimality, given that $m \leq m_*$, which is the region of interest.

For practical purposes $N/\beta >> 1$; hence, from Eq. (27)

$$G_* \simeq N\beta^{\alpha-1} \frac{\alpha^\alpha}{(\alpha - 1)^{\alpha-1}} = \alpha \frac{N}{n_1^*}\left(\beta \frac{\alpha}{\alpha - 1}\right)^\alpha$$

Since N/n_1^* is the number of all the 1st layer subnets, and $\beta\alpha/(\alpha - 1)$ is the size of any such subnet, then $G_* \simeq \alpha G_1^*$

This says that the design of the 1st layer subnets represents approximately $1/\alpha$ of the total computational cost.

Finally, there exists a one to one correspondence between the set of regular trees of degree K (K ≥ 2 integer) whose number of levels is m_* (integer ≥ 2) and the global optimal solutions of our optimization problem, where $\alpha = K/(K-1)$ and $N = K^{m_*}$ for k = 2, 3, ..., ∞. Notice that the above set of α's is contained in the interval of (1, 2] of real values. i.e., $1 < \alpha \leq 2$. Also, $\alpha = 2$ corresponds to a binary tree representation.

Let us also note that from Eq. (24)

$$G_\infty \triangleq \lim_{m \to \infty} \underline{G}(m, \alpha, \beta) = \beta^{\alpha-1} \frac{\alpha^\alpha}{(\alpha-1)^{\alpha-1}} N$$

Note also that the difference, $G_\infty - G_*$ is independent of N, and that the relative difference $(G_\infty - G_*)/G_*$ goes to zero as N goes to infinity. The curves below illustrate these properties. Fig. 7 shows the initially decreasing,

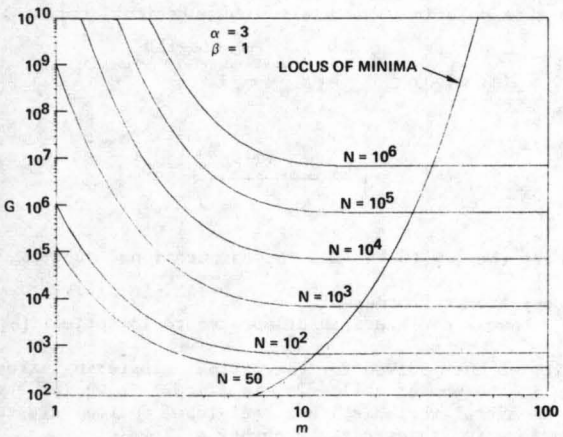

Figure 7. Minimum Computational Cost G(m, α,β), Given m; α = 3, β = 1.

then-slightly increasing and asymptotic behavior of $\underline{G}(m, \alpha, \beta)$ with respect to m and for several values of N. It also shows a "clamping" effect whereby, once G(m) reaches its minimum value G_*, it appears as if it remains indefinitely at that value. They also illustrate the fairly fast convergence of $\underline{G}(m)$ toward a value close to the minimum, for a value of m relatively smaller than m_*. This indicates that we may actually obtain most of the computational gains with hierarchical structures whose number of levels (m) is much smaller than the optimal ones (m_*). Such a property was also observed in the context of hierarchical routing (see Section 2.2)!

So far we have determined optimal structures for the hierarchical design of computer networks. A few questions remain as to the actual assignment of nodes to clusters, the decomposition of global variables in terms of secondary variables related to the different levels in the hierarchy, etc. The decomposition of the average delay is performed in [6].

4. SUMMARY

Faced with the prohibitive cost of a simple extrapolation of present design and routing procedures for large networks, the goal of this paper was to present and evaluate some new techniques to be used for large networks. The techniques studied here represent an extension of present schemes and rely mainly on the natural hierarchical clustering of the network nodes. More specifically we specified, evaluated and discussed the adaptive m-level Hierarchical Routing (MHR) schemes as well as some issues related to the m-level Hierarchical Topology (MHT) design of large networks.

With respect to the MHR, models were developed to determine clustering structures which lead to a minimal routing cost (storage, capacity) and their effect on the network path length. More importantly, we determined the effect of table reduction in terms of network throughput and delay. As a result we were able to demonstrate that under some reasonable cost and performance constraints for a class of large distributed networks, present routing schemes become infeasible, whereas hierarchical routing schemes with optimally chosen table lengths maintain remarkably good network performance for a phenomenal range of network sizes.

With respect to the MHT, a general methodology was specified for the hierarchical design of large networks. A model was developed to determine clustering structures which minimize the computational cost involved in the design phase. Such optimal structures lead to very significant savings (i.e., proportional to N).

REFERENCES

[1] Davies, D.W. and D. Barber. Communication Networks for Computers, John Wiley, New York, 1973.

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

[3] Roberts, L.G. "Multiple Computer Networks and Intercomputer Communications," ACM Symposium on Operating Systems Principles, Gatlinburg, Tennessee, October 1967.

[4] Frank, H. and W. Chou, "Topological Optimization of Computer Networks," Proceedings of the IEEE, 60(11):1385-1397, November 1972.

[5] Gerla, M. "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.

[6] Kamoun, F. Design Considerations for Large Computer Communication Networks, Ph.d. Dissertation, School of Engineering and Applied Science, University of California, Los Angeles, March 1976.

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

[8] Fultz, G.L. "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.

[9] McQuillan, J.M. "Adaptive Routing Algorithms for Distributed Computer Networks," Bolt Beranek and Newman, Inc., Cambridge, Massachusetts, Report No. 2831, May 1974.

[10] Kahn, R.E. "Resource-Sharing Computer Communication Networks," Proceedings of IEEE, 60(11):1396-1407, November 1972.

[11] Gerla, M., W. Chou, and H. Frank. "Computational Considerations and Routing Problems for Large Computer Communication Networks," Proceedings of the National Telecommunication Conference, 2:2B-1 to 2B-11, Atlanta, Georgia, November 1973.

[12] Frank, H., M. Gerla, and W. Chou. "Issues in the Design of Large Distributed Computer Communication Networks," Proceedings of the National Telecommunication Conference, 37A-1 to 37A-8, Atlanta, Georgia, November 1973.