Hierarchical Interconnection Networks for Multicomputer Systems

SIVARAMA P. DANDAMUDI, MEMBER, IEEE, AND DEREK L. EAGER, MEMBER, IEEE

Abstract—**Multicomputer systems are distributed-memory MIMD systems. Communication in these systems occurs through explicit message passing. Therefore, the underlying processor interconnection network plays an important and direct role in determining their performance. Several types of interconnection networks have been proposed in the literature. Unfortunately, no network is "universally" better. Ideally, therefore, systems should use more than one such network. Furthermore, systems that have large numbers of processors should be able to exploit locality in communication in order to obtain improved performance. This paper proposes the use of hierarchical interconnection networks to meet both these requirements.**

A performance analysis of a class of hierarchical interconnection networks is presented. This analysis includes both static analysis (i.e., queueing delays are neglected) and queueing analysis. In both cases, the hierarchical networks are shown to have better cost-benefit ratios. The queueing analysis is also validated (within our model) by several simulation experiments. The impact of two performance enhancement schemes—replication of links and improved routing algorithms—on hierarchical interconnection network performance is also presented.

Index Terms—**Hypercubes, interconnection networks, multicomputer systems, parallel systems, performance.**

I. Introduction

MULTIPLE-instruction-multiple-data (MIMD) systems can be divided into two groups: shared-memory and distributed-memory. Systems in the shared-memory group are often referred to as multiprocessors and those in the other group as multicomputers. Shared-memory systems can use the memory as a communication medium; the distributed-memory systems must, however, communicate by explicitly passing messages. Both types of systems have advantages and disadvantages. The shared-memory systems are able to support code and data sharing, but they are architecturally (relatively) complex. The distributed-memory systems are architecturally more simple and economical. Seitz [25] conjectured that shared-memory organizations will be preferred for systems with tens of processors, and message-passing organizations for systems with hundreds or thousands of processors; hybrid forms may be attractive for systems having intermediate numbers of processors.

Both architectural classes are subjects of ongoing research. The Cosmic Cube [25], the Finite Element Machine [20], the NCUBE/ten [13], and the Transputer system [30] are examples of multicomputer systems. The Butterfly Parallel Processor [5], the NYU Ultracomputer [11], and the CEDAR system [9] are examples of multiprocessor systems. The IBM RP3 [23] is an example of a hybrid form. This system encompasses both shared-memory and distributed-memory paradigms (a mixture of the two can be chosen at run time).

The goal of this paper is to develop and analyze efficient static interconnection networks for multicomputer systems. Section II presents details on the hierarchical network structure that is proposed here. Some examples of these hierarchical networks are analyzed in Section III. This analysis includes both static analysis (with no contention) and queueing analysis. Locality in communication is treated in the analysis. Section IV discusses some performance enhancements to hierarchical interconnection networks. Section V concludes the paper by summarizing the results.

Manuscript received November 16, 1987; revised October 28, 1988 and March 14, 1989.

S. P. Dandamudi is with the School of Computer Science, Carleton University, Ottawa, Ont., K1S 5B6 Canada.

D. L. Eager is with the Department of Computational Science, University of Saskatchewan, Saskatoon, Sask., S7N 0W0 Canada.

IEEE Log Number 9035133.

II. Hierarchical Interconnection Networks

Hierarchical interconnection networks (HIN's) are intuitively appealing when a large number of processors are to be connected, for the reasons described in Section II-A. Several HIN's have been proposed previously in the literature. Section II-B presents details on some of these networks. Section II-C presents the structure of the HIN's considered in this paper.

A. Motivation for Hierarchical Interconnection Networks

There are many possible static interconnection networks for multicomputer systems. Some examples are the linear array, bidirectional ring (BR), star, complete connection (CC), dual-bus hypercube [33], spanning bus hypercube, tree [7], [12], [15], and cube-connected-cycle (CCC) [24]. These provide a range of choices on the cost/performance spectrum. Table I summarizes some major points on the spectrum. At one extreme are completely connected networks that provide direct communication between any pair of nodes at the cost of using a number of links that grows with the square of the number of nodes. At the other extreme are ring-like networks which require a number of links that is proportional to the number of nodes, but in which the average internode distance increases in direct proportion to the number of nodes. Bus structures exhibit performance similar to that of the ring networks. A significant intermediate point on the cost/performance spectrum is represented by hypercube networks. Hypercube networks allow the average distance between any pair of nodes to be

TABLE I
TRADEOFFS INVOLVED IN SOME INTERCONNECTION NETWORKS

Type of network	Number of links	Average internode distance
Complete connection	$O(N^2)$	$O(1)$
Hypercube	$O(N \log N)$	$O(\log N)$
Ring	$O(N)$	$O(N)$

$O(\log N)$ through a structure in which each node is directly attached to $O(\log N)$ other nodes.

There are two main motivations for HIN's. First, for very large systems, the number of links needed with a conventional network structure such as the hypercube may become prohibitively large. Future systems may be designed to minimize the number of links because most of the system space may be filled with wires [14]. HIN's exploit the locality that exists in communication patterns to allow reduction in the required number of links. It is important to note that only locality as a general phenomenon present in many parallel computations is exploited; HIN's are not tailored for specific forms of locality or particular applications. An analogy can be made with cache or memory management; in these domains as well, a general property of computations (temporal and spatial locality in this case) is exploited without tailoring for specific forms of these phenomena which might be found in particular applications.

Second, there are a number of compelling interconnection network topologies, each with advantages and disadvantages, and each most appropriate for its own set of applications. Snyder [28] argues that the problems that are of interest in the context of parallel systems are generally superlinear (i.e., their sequential time complexity is $O(n^2)$ to $O(n^4)$ for problems of size n) and thus, assuming best possible speedup, the execution time of these problems can be improved only sublinearly by using parallel systems. He concludes that one cannot afford the luxury of large overheads, and that, in order to reduce the overhead introduced by interconnection networks, it is necessary to match the structure of the problem to the communication structure of the system (i.e., network topology). As an example, in image processing, it is known that it is possible to perform two-dimensional filtering efficiently using a two-dimensionally connected grid of processing elements (by assigning one processor per pixel) [14]. Here the mismatch between the problem structure and that of the architecture is minimal: the application is two dimensional and the supporting architecture is two dimensional. Although such a perfect match is desirable, it is difficult, with current technology, to reconfigure the network structure of a parallel processing system to match each possible application. The proposed CHiP architecture has a reconfigurable network structure [26], [27], but it requires substantial advances in VLSI and packaging technologies. HIN's provide an alternate way in which several topologies can be integrated, with current technology.

B. Previous Work

Several systems have been proposed with HIN's. The Cm* is a two-level hierarchical multiprocessor system [31]. The Cm* is made up of 50 processor memory pairs called compute modules or cm's, grouped into clusters. Communication within a cluster is via a parallel bus controlled by an address mapping processor termed a *Kmap*. There are five clusters and these communicate via an intercluster bus. The Cm* is extensible, either by adding processors to each cluster (up to a maximum of 14) or by increasing the number of clusters.

The Cedar system [9] uses a bus interconnection between the processors within a cluster and the cluster memory they share, and a multistage interconnection network between all processors and a global memory shared among all clusters. The global multistage network is based on an extension of the Omega network [18].

Cluster-based multiprocessor systems using a crossbar network are also proposed by Agrawal and Mahgoub [2]. They note that the cluster-based scheme provides results closer to a fully connected crossbar-based scheme if the concept of favorite memory (i.e., a processor accesses a favorite memory with high probability) is assumed to be applicable; otherwise (i.e., when memory requests are equally distributed throughout the system), there is substantial degradation in performance. In general, they conclude, a cluster-based scheme offers great potential for future supercomputer systems.

Carlson [4] proposes a two-level mesh hierarchy scheme: the mesh with a global mesh structure. It is shown that this structure allows dramatic improvement in the efficiency of executing computations organized as a binary tree and for linear recurrences, while for other computations such as sorting, it offers no improvement.

A cluster structure using shared buses as the basic interconnection media has been proposed by Wu and Liu [34]. Multiple levels of clustering may be present in their organization. Shared buses are used to interconnect the units within a cluster. Structural complexity analysis is given under the assumption of uniform message routing. However, they take message locality into consideration for topological optimization case studies. The locality is modeled by a single parameter *LC* (similar to the parameter α discussed in Section III-A of this paper).

C. Proposed Structure of HIN's

The structure of the HIN's considered here can be informally described as follows. Let N be the total number of nodes in the network. These N nodes are divided into K_1 clusters of $n_1 = N/K_1$ nodes each. It is assumed here, for convenience of analysis, that K_1 evenly divides N, and that each cluster is of identical size, although in practice there would be no reason not to permit clusters of varying sizes. Each cluster of n_1 nodes is linked together by a level 1 interconnection network. Then one node from each cluster is selected to act as an interface node and these K_1 interface nodes are again divided into K_2 clusters of $n_2 = K_1/K_2$ nodes each. (As before, it is assumed that K_2 evenly divides K_1, and that all clusters are of the same size.) A level 2 interconnection network is used to link each of these K_2 clusters of n_2 level 1 interface nodes. Then, one node from each level 2 cluster is selected as level 2 interface node to be linked together by a level 3 network, etc. Interconnection networks used at different lev-

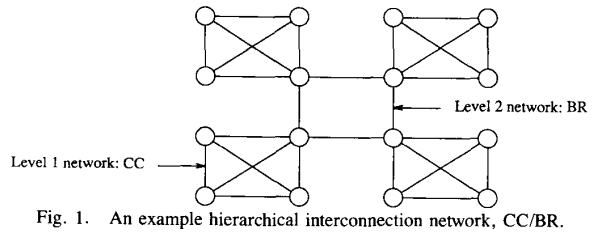


Fig. 1. An example hierarchical interconnection network, CC/BR.

els may have different topologies; furthermore, the networks used at the same level may also be different from cluster to cluster. Fig. 1 shows an example HIN with two levels. The level 1 network is a complete connection (CC) network and the level 2 network is a bidirectional ring (BR) network.

In the analysis, presented in Section III, all HIN's are restricted to two levels. It has been shown in [6] that two is a pragmatic choice for the number of levels in the hierarchy. This reference also considers the optimization of cluster sizes. Perhaps surprisingly, it was found that relatively small clusters (e.g., 8 nodes) are suitable over a very wide range of system sizes and workload parameter values. Of course, the final choice in an implementation would depend as well on packaging considerations.

It is further assumed that the same interconnection network is used in all clusters at level 1. The notation "level 1 network/level 2 network" is used to identify an HIN. For example, CC/BR identifies the HIN shown in Fig. 1.

An advantage of HIN's is that they reduce the degree (i.e., the number of links connected to a node) of the majority of nodes. The degree of the remaining nodes is the same as that of the corresponding nonhierarchical network. As an example, consider a binary hypercube (BH) network with $N = 2^D$ nodes. The degree of each node in this network is D. Now consider a two-level HIN BH/BH with a cluster size of $n = 2^d$ and the number of clusters $K = 2^{D-d}$. In this HIN, only $N/2^d$ nodes will have a degree of D (the interface nodes); the remaining nodes will have only a degree of d. If $N = 1024$ (i.e., $D = 10$) and $n = 16$ (i.e., $d = 4$), then the BH/BH HIN will have 960 nodes with degree 4 and 64 nodes with degree 10. This is in contrast to a degree of 10 for all 1024 nodes in the BH network.

This aspect of HIN's is important in system design. For example, consider the NCUBE/ten system [13], which organizes 1024 processing units (processor+memory) as a binary hypercube. Each node (i.e., processing unit) in this system has a total of 20 half-duplex link connections. Using custom-made VLSI chips, the designers could pack as many as 64 nodes nodes on a single $16'' \times 22''$ printed-circuit board (PCB). The total system consists of 16 such PCB's, and inter-PCB connections require as many as 512 wires from each PCB. (The NCUBE/ten actually uses 640 wires to allow connections to I/O devices.) The use of the BH/BH HIN (with $d = 6$ and $D = 10$) greatly reduces the number of wires needed for inter-PCB connection (from 512 wires to 8 wires). This reduction in the number of link connections has two implications on system design. First, it reduces the demand for both chip area and PCB area. Thus, more processing units can be packed in a given PCB area. Second, since inter-PCB connection is greatly simplified, it is feasible to increase the system size N substantially. This second factor is important in designing large parallel systems.

Disadvantages of HIN's include the potentially high traffic rates on intercluster links, and thus potential degradation in performance, and the potential for diminished fault tolerance due to the special role played by interface nodes. However, the performance enhancements suggested in Section IV appear to economically alleviate the problem of congestion on intercluster links, and, although not considered in detail in this paper, it would appear that standard fault-tolerance techniques can be economically applied to interface nodes.

III. Analysis of Hierarchical Interconnection Networks

This section analyzes several example HIN's, and, through this analysis, shows the advantages of HIN's in comparison to nonhierarchical structures. In the analysis, the performance of an HIN is compared to that of a nonhierarchical (reference) network; here, a BH network is used. The HIN's considered are BH/BH, BH/CCC, BH/BR, and BH/CC. It is obvious that the last two networks are not useful for large numbers of clusters, but they are included for comparison purposes only. The BH/BR uses the minimum number of links among the four hierarchical networks and the BH/CC provides the minimum average internode distance.

Section III-A presents a performance analysis of HIN's assuming that there is no contention (i.e., no queueing). Analysis with contention is presented in Section III-B. This section also presents the results of the simulation experiments that validate the queueing analysis.

A. Analysis with No Contention

Let N denote the total number of nodes in the network, n denote the number of nodes in a cluster, and K denote the number of clusters (i.e., K nodes participate in the level 2 network), where $N = Kn$. As noted previously, it is assumed that N is evenly divisible by K. It is also assumed that a node can send messages to itself. Such an assumption is not strictly necessary; however, without this assumption, the resulting mathematical equations become more unwieldy without adding any additional insight into performance.

Let α be the probability of both source and destination nodes of a message being in the same cluster. Therefore $(1 - \alpha)$ represents the probability of intercluster communication. The larger the value of α (for a fixed cluster size), the stronger the locality in communication. Locality in communication has been characterized similarly by other researchers [34]. It is further assumed that:

- Intracluster communication is uniformly random (i.e., a source node sends an intracluster message to each node within its cluster with equal probability).
- Intercluster communication is uniformly random (i.e., a source node sends an intercluster message to each other cluster with equal probability, and to each node within the destination cluster with equal probability).

It is difficult to predict what *specific* values of α one should expect in practice, but it is clear that a large number of parallel applications are in fact characterized by a communication

structure that results in significant locality. The results of the following sections suggest that HIN's yield superior performance to nonhierarchical networks over a wide range of α values, and thus should be generally useful.

1) Criteria for Comparison of Performance: The following performance measures are used to evaluate various interconnection networks.

a) *Link cost:* One goal is to minimize the link cost L as it limits the economical size of the network. The number of links is used to represent the link cost. Since bus-based structures are not treated here, the number of link connections need not be considered.

b) *Average internode distance:* Another goal is to minimize the average internode distance P as it largely determines message transmission time and, hence, the effective computing rate. The number of links in the message transmission path is used as a measure of the internode distance.

c) *LP product:* In general, trying to minimize one of the above two measures (say P) results in an increase in the other (in this case L). Therefore, a useful measure is $L \times P$, which should be minimized. The *LP product* really gives a cost–benefit ratio (with L representing the cost of the network and $1/P$ representing the benefit). Thus, benefit per unit cost can be maximized by minimizing the *LP product*. Similar performance measures are also used by other researchers [1], [3].

d) *LP ratio:* It is useful to compare the performance of an HIN to that of the reference network. The subscript "H" is used to represent the performance measures of an HIN (e.g., L_H, P_H) and "R" is used for the reference network (e.g., L_R, P_R). Then,

$$LP \text{ ratio} = \frac{L_H \times P_H}{L_R \times P_R}.$$

Smaller *LP ratios* are preferred. A value of 1 for the *LP ratio* indicates that the HIN has the same *LP product* as the reference network. An *LP ratio* < 1 is desired as it indicates an improvement in performance (at least, with respect to LP measure) associated with the HIN.

2) Analysis of BH Network: Consider a binary hypercube (BH) with $N = 2^D$ nodes, where D is the dimensionality of the hypercube. The number of links in this network is given by

$$L_{BH} = D2^{D-1}. \tag{1}$$

The average internode distance P_{BH} is now derived under the assumption of locality in communication.

Let $n = 2^d$. Then the address of a node, which is D bits long, can be divided into two groups. The d least significant bits identify a node within a cluster of nodes whose $(D-d)$ most significant bits are the same; these $(D-d)$ bits serve to identify the cluster. Then,

$$P_{BH} = \alpha P'_{BH} + (1-\alpha)P''_{BH} \tag{2}$$

where

P'_{BH} = average distance between two nodes within a cluster, and

P''_{BH} = average distance between two nodes that are in two different clusters.

The average internode distance within a cluster (recalling that we allow message source and destination to be the same node) is given by

$$P'_{BH} = \frac{\sum_{i=0}^{d} \binom{d}{i} i}{2^d} = \frac{d}{2}. \tag{3}$$

To compute P''_{BH}, choose an arbitrary source node. Recall that the D address bits are divided into two groups. In computing P''_{BH}, it should be noted that in the most significant $(D-d)$ bits the source and destination node addresses should not be the same (the lower d bits may be the same). Noting that the number of destination nodes at a distance *dist* (source, dest) $= i + j$ from the source node, where the distance i $(i > 0)$ is contributed by the most significant $(D-d)$ bits and the distance j $(j \geq 0)$ is contributed by the least significant d bits, is equal to $\binom{D-d}{i}\binom{d}{j}$, P''_{BH} is given by

$$P''_{BH} = \frac{\sum_{i=1}^{D-d}\sum_{j=0}^{d}\left\{\binom{D-d}{i}\binom{d}{j}(i+j)\right\}}{2^D - 2^d}$$

$$= \frac{D2^{D-1} - d2^{d-1}}{2^D - 2^d}. \tag{4}$$

Equations (2), (3), and (4) then give

$$P_{BH} = \alpha\left[\frac{d}{2}\right] + (1-\alpha)\left[\frac{D2^{D-1} - d2^{d-1}}{2^D - 2^d}\right]. \tag{5}$$

It should be noted that $D = \log N$ and $d = \log n = \log(N/K)$.

3) Analysis of BH/BH Hierarchical Network: This network uses the BH at both levels. From each cluster, one node participates in the level 2 BH. This node serves as the interface node between two levels. The number of links in this structure is

$$L_{BH/BH} = d2^{D-1} + (D-d)2^{D-d-1}. \tag{6}$$

The average internode distance of an HIN is given by

$$P_H = \alpha(P_{\text{level 1}}) + (1-\alpha)[2P_{\text{level 1}} + P_{\text{level 2}}] \tag{7}$$

where

$P_{\text{level 1}}$ = average internode distance of the network at level 1; and

$P_{\text{level 2}}$ = average internode distance of the network at level 2 (when message source and destination clusters are required to be different).

For the HIN under consideration, $P_{\text{level 1}} = d/2$. $P_{\text{level 2}}$ can be computed from (3), yielding

$$P_{\text{level 2}} = \frac{\sum_{i=1}^{D-d}\binom{D-d}{i} i}{2^{D-d} - 1} = \frac{(D-d)2^{D-d-1}}{2^{D-d} - 1}.$$

Therefore, using (7)

$$P_{\text{BH/BH}} = \alpha \left\lceil \frac{d}{2} \right\rceil + (1-\alpha) \left\lceil \frac{(D+d)2^{D-1} - d2^d}{2^D - 2^d} \right\rceil. \quad (8)$$

4) Analysis of Other Hierarchical Networks: It is straightforward to derive the following for the BH/BR, BH/CC, and BH/CCC networks. Note that $N = 2^D$, $n = 2^d$, and $K = 2^{D-d}$.

For the BH/BR network:

$$L_{\text{BH/BR}} = 2^{D-d}[d2^{d-1} + 1] \quad (9)$$

$$P_{\text{BH/BR}} = \alpha \left\lceil \frac{d}{2} \right\rceil + (1-\alpha) \left[d + \frac{2^{2(D-d)}}{4(2^{D-d}-1)} \right]. \quad (10)$$

For the BH/CC network:

$$L_{\text{BH/CC}} = d2^{D-1} + 2^{D-d-1}(2^{D-d} - 1). \quad (11)$$

$$P_{\text{BH/CC}} = \alpha \left\lceil \frac{d}{2} \right\rceil + (1-\alpha)(d+1). \quad (12)$$

For the BH/CCC network: Let D_c be the dimensionality of the (level 2) CCC network. Then, the number of nodes in the level 2 network (equal to the number of clusters K in the HIN) is $D_C 2^{D_c}$ and the number of links is $3D_c 2^{D_c-1}$. The total number of nodes N is $D_c 2^{D_c+d}$, and the number of links is

$$L_{\text{BH/CCC}} = d2^{D-1} + 3D_c 2^{D_c-1}. \quad (13)$$

The average internode distance, without self-routing, in the CCC of dimensionality D_c is given by [33]

$$P_{\text{level 2}} = \frac{7}{4}D_c - 3 + \frac{D_c+1}{2^{D_c}-1}.$$

Then

$$P_{\text{BH/CCC}} = \alpha \left\lceil \frac{d}{2} \right\rceil + (1-\alpha) \left[d + \frac{7}{4}D_c - 3 + \frac{D_c+1}{2^{D_c}-1} \right]. \quad (14)$$

5) Discussion of Results: The average internode distance P and the *LP ratio*, are shown in Figs. 2–4 for some example hierarchical networks. All these networks have a cluster size $n = 16$. Figs. 2 and 3 give P and *LP ratio* values for $N = 128$ and 1024, respectively. The value of α is varied from 0 to 1. For a system with 1024 nodes and for the configurations considered, the BH/BH hierarchical network provides the best cost–performance ratio (i.e., the lowest *LP ratio*) for a wide range of α values. As α approaches 1, however, both BH/BR and BH/CCC hierarchical networks provide a better *LP ratio* because these networks allocate fewer links to the rarely used (in this case) level 2 network. Note that even for $\alpha = 0$ (reflecting a context of strong "anti-locality"), the BH/BH network has an *LP ratio* of less than one. It should be noted, however, that as α approaches 0, the required message processing capacities of level 2 links increase as will be shown in Section III-B4.

Fig. 4 gives P and *LP ratio* values as functions of the total number of nodes in the network, N, for $\alpha = 0.8$. Again, n is fixed at 16 and the number of clusters K is increased to increase N. The graph of Fig. 4 indicates that the average internode distance of the BH/BH hierarchical network is only 10% higher than that of the BH reference network. The BH/BH hierarchical network uses far fewer links than does the BH reference network (see Fig. 5). This reduction in the number of links used is directly proportional to N. This factor is important when designing large systems.

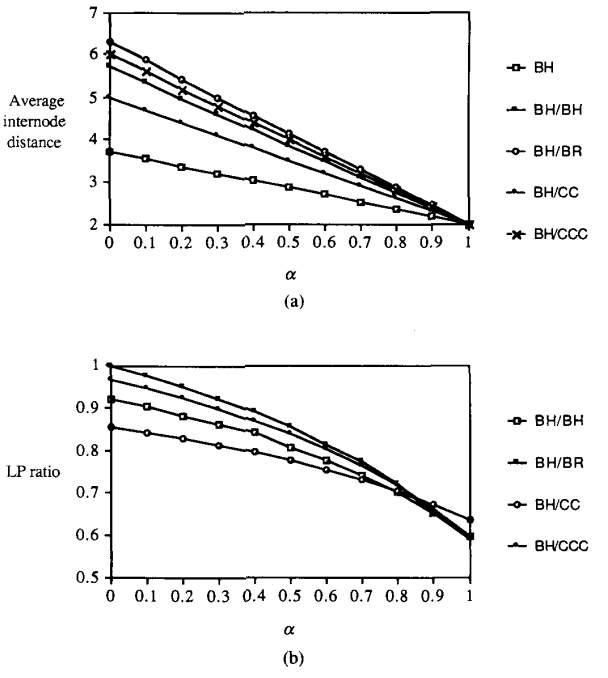


Fig. 2. Performance of hierarchical networks ($N = 128$, $K = 8$).

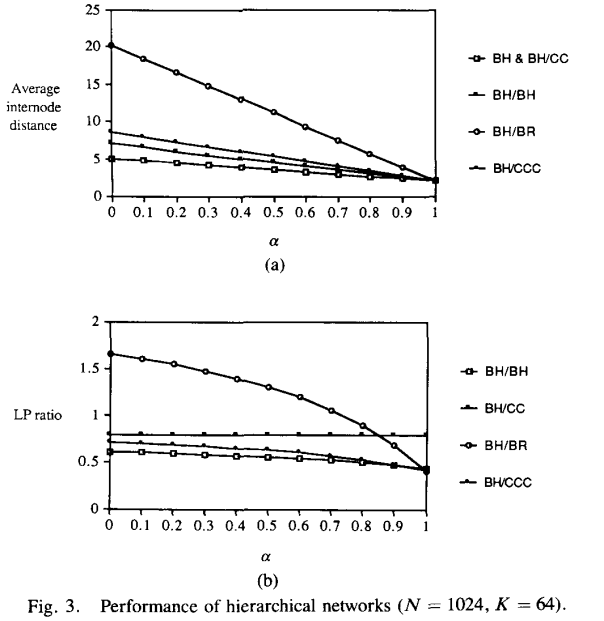
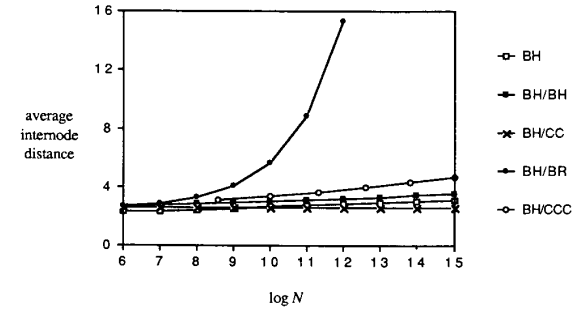
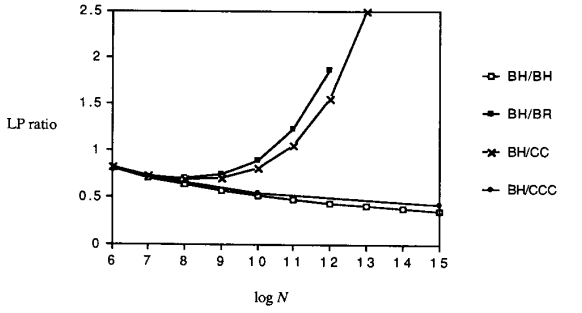


Fig. 3. Performance of hierarchical networks ($N = 1024$, $K = 64$).



(a)



(b)

Fig. 4. Performance of hierarchical networks ($\alpha = 0.8$).

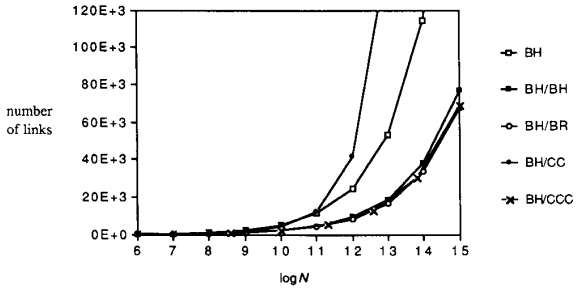


Fig. 5. Link cost of hierarchical networks.

From the sample results presented here and from other results with varying parameter values, it appears that both BH/BH and BH/CCC hierarchical networks provide substantial improvements in cost–performance ratios (as measured by the *LP ratio*) and that these improvements increase as the system size increases. The extent of these improvements depends significantly on the degree of locality in communication, as measured by α, but even for very small α some improvement is obtained.

Different applications may have greatly differing α values. It is useful to consider here, however, an example that illustrates for at least a class of applications what range of values would be likely. Consider applications that are naturally structured as two-dimensional meshes. Mesh-connected systems and correspondingly structured algorithms have been extensively studied in the literature [4], [20]–[22], [29], [32]. Such systems support efficiently nearest-neighbor communication in which each node communicates with four of its neighboring nodes. Now, suppose that a cluster consists of a square submesh of n nodes (i.e., a $\sqrt{n} \times \sqrt{n}$ mesh). It is straightforward to show that, for a mesh-structured computation on such a system,

$$\alpha = 1 - \frac{1}{\sqrt{n}}.$$

Some sample values of α are shown in Table II. We give two examples of systems that suggest that cluster sizes of 16 to 64 are implementable. The Cedar system [9] uses a cluster size of 16. The NCUBE/ten [13] packs as many as 64 nodes, organized as a BH, on a single printed-circuit board. This suggests that a cluster size of 64 is also feasible.

TABLE II
TYPICAL VALUES OF α FOR A NEAREST-NEIGHBOR COMMUNICATION PATTERN

Cluster size n	α
4	0.5
9	0.67
16	0.75
25	0.80
36	0.83
49	0.86
64	0.875

B. Analysis with Contention

A very simple model is used for the queueing analysis. The "Markovian" and other similar assumptions that we use in this analysis are not at all "realistic," in that they do not reflect the behavior of some given, particular application. Note, however, that it is not the goal here to accurately predict the performance of a particular system with some particular workload; rather, all that is desired is an "order of magnitude" evaluation of the relative performance of different types of systems. A large body of literature and experience with models of this type supports their usefulness [19].

Each network link is assumed to be full-duplex. In our analysis, we conceptually replace each full-duplex link by two half-duplex links, each of which is modeled as a queueing center. Expressions are derived for the average message delivery time R on each link. The following assumptions are made about the network and its workload.

1) Each node generates messages at rate λ and the inter-message times are exponentially distributed.

2) The node message generation processes are independent of each other.

3) Message service times are exponentially distributed; each link processes these messages at rate μ.

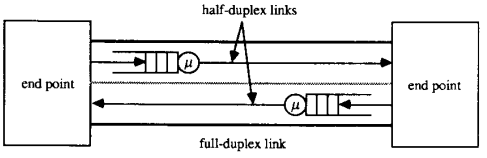
4) Each node has unbounded buffering capacity.

5) Packet-switching is used for message transmission. (Packet-switching is used in several multicomputer systems such as the Cosmic Cube [25] and the NCUBE/ten [13].)

6) All messages are routed over the shortest path between the source and destination nodes. If more than one shortest

path exists between a pair of nodes, it is assumed that random routing is used (i.e., each shortest path is selected with equal probability). Thus, the routing scheme uses no information regarding the current state of the network. (In Section IV-B, we study the impact on performance of improved routing algorithms.)

To simplify the analysis, Kleinrock's independence assumption [17] is used. This assumption states that each time a message arrives at a link, a new service time is generated for this message from the exponential service time distribution (i.e., there is no "memory" of message lengths from hop to hop). This assumption is often used in the delay analysis of communication networks [10]. The results of the simulation experiments presented in Section III-B5 indicate that this assumption is reasonable in this context. This allows each (half-duplex) link to be modeled as an M/M/1 queueing center [16]. Our link model is depicted below.



1) Analysis of BH Network: This section considers the BH network with N nodes. Let $D = \log N$ be the dimensionality of the BH. Assume that the cluster size $n = 2^d$, the number of clusters $K = 2^{D-d}$, and α is the probability of intracluster communication. The links can be divided into two groups: cluster (CL) links and noncluster (NCL) links. Cluster links are those that connect the nodes within a cluster only (i.e., nodes whose upper $(D-d)$ bits are the same) and noncluster links are those that connect two nodes that are in two different clusters (i.e., nodes whose lower d bits are the same). Let μ_{CL} and μ_{NCL} be the message processing rates of cluster and noncluster links, respectively. Furthermore, let $\lambda_{link,CL}$ and $\lambda_{link,NCL}$ be the message arrival rates at cluster and noncluster links, respectively. For cluster links, the message arrival rate $\lambda_{link,CL}$ is given by

$$\lambda_{link,CL} = \frac{2^d \lambda \left(\frac{d}{2}\right)}{2d2^{d-1}} = \frac{\lambda}{2}.$$

To compute $\lambda_{link,NCL}$, it should be noted that each intercluster message uses, on average, $(P''_{BH} - (d/2))$ noncluster links, where P''_{BH} is given by (4). Then $\lambda_{link,NCL}$ is given by

$$\lambda_{link,NCL} = \frac{(1-\alpha)\lambda 2^D \left[P''_{BH} - \frac{d}{2}\right]}{2(D-d)2^{D-1}} = (1-\alpha)\lambda \frac{2^{D-1}}{2^D - 2^d}.$$

The average delivery time of a message is given by

$$R_{BH} = \alpha \left[\frac{d}{2}\right] \Delta_{CL} + (1-\alpha) \cdot \left[\frac{d}{2}\Delta_{CL} + \left(\frac{(D-d)2^{D-1}}{2^D - 2^d}\right)\Delta_{NCL}\right] = \left[\frac{d}{2}\right]\Delta_{CL} + (1-\alpha)\left(\frac{(D-d)2^{D-1}}{2^D - 2^d}\right)\Delta_{NCL} \quad (15)$$

where

$$\Delta_{CL} = \frac{1}{\mu_{CL} - \lambda_{link,CL}}, \text{ and } \Delta_{NCL} = \frac{1}{\mu_{NCL} - \lambda_{link,NCL}}.$$

2) Analysis of BH/BH Hierarchical Network: In this network, one node from each cluster acts as an interface node. As in the analysis of the BH network, the links are divided into cluster and noncluster links, and the message arrival rate for each type of link is derived.

In deriving $\lambda_{link,NCL}$, it should be noted that each intercluster message uses $[(D-d)2^{D-d-1}/2^{D-d} - 1]$ noncluster links, on average, and each cluster generates these messages at rate $(1-\alpha)\lambda 2^d$. Therefore,

$$\lambda_{link,NCL} = \frac{2^{D-d}2^d(1-\alpha)\lambda\left[\frac{(D-d)2^{D-d-1}}{2^{D-d}-1}\right]}{2(D-d)2^{D-d-1}} = \frac{2^{D-1}(1-\alpha)\lambda}{2^{D-d}-1}.$$

The message load of intercluster messages is, however, not distributed equally among the links in a cluster. The cluster links closer to the interface node receive more intercluster messages than those that are not. Thus, $\lambda_{link,CL}$ is dependent on how close a link is to the interface node.

Let $H(s,t)$ be the distance between nodes s and t (i.e., the number of bits in which the addresses of s and t differ when expressed in binary form). Then we define a j-level link as a cluster link that connects a node at distance j from an interface node to a node at distance $j-1$ from the interface node. Then, the arrival rate of messages at a j-level link is given by

$$\lambda_{link,CL,j} = \lambda'_j + \lambda''_j$$

where

λ'_j = arrival rate (at a j-level link) due to intracluster messages, and
λ''_j = arrival rate (at a j-level link) due to intercluster messages.

Since the intracluster message load is equally distributed over the links within a cluster, λ'_j is given by

$$\lambda'_j = \frac{\alpha\lambda}{2} \qquad \text{for all } j.$$

λ''_j is not the same for all the links in a cluster. It is given by

$$\lambda''_j = n\lambda(1-\alpha)p(j) \quad (16)$$

where $p(j)$ is the probability that an intercluster message gets routed through a j-level link on its way out of (or into) the cluster.

We now derive an expression for $p(j)$. An assumption of the model is that the routing algorithm utilizes all the shortest paths with equal probability. A set of links J, all of whose element links are j-level links, act as intermediate links for all intercluster messages generated by any source node within the cluster that is at distance at least j from the interface node.

Therefore, the number of source nodes for which an element of J acts as an intermediate link is $2^d - \sum_{k=0}^{j-1} \binom{d}{k}$. Since there are $(d-j+1)\binom{d}{j-1}$ such intermediate links (i.e., the number of elements in J),

$$p(j) = \frac{2^d - \sum_{k=0}^{j-1} \binom{d}{k}}{(d-j+1)\binom{d}{j-1}2^d} = \frac{\sum_{k=j}^{d} \binom{d}{k}}{(d-j+1)\binom{d}{j-1}2^d}.$$

This completes the derivation of $\lambda_{\text{link, CL, }j}$. The mean delay of a j-level link Δ_j is given by

$$\Delta_j = \frac{1}{\mu_{\text{CL}} - \lambda_{\text{link, CL, }j}}.$$

Analogously to (7), the average message delivery time can be computed using

$$R_H = \alpha(R_{\text{level 1}}) + (1-\alpha)(2R_{\text{level 1}} + R_{\text{level 2}}) \quad (17)$$

where

$R_{\text{level 1}}$ = average message delivery time in the level 1 network, and

$R_{\text{level 2}}$ = average message delivery time in the level 2 network.

This yields

$$R_{\text{BH/BH}} = \alpha\left\lceil\frac{d}{2}\right\rceil \Delta_{\text{Avg}} + (1-\alpha) \cdot \left[2\Delta_{\text{CL-Avg}} + \left(\frac{(D-d)2^{D-d-1}}{2^{D-d}-1}\right)\Delta_{\text{NCL-Avg}}\right] \quad (18)$$

where

Δ_{Avg} = Average delay, at a link, of an intracluster message

$$= \frac{1}{d2^{d-1}} \sum_{j=1}^{d} \left\{(d-j+1)\binom{d}{j-1}\Delta_j\right\}$$

$\Delta_{\text{CL-Avg}}$ = average delay of an intercluster message in a cluster (summed over all the links traversed within the cluster)

$$= \sum_{j=1}^{d} \left\{\Delta_j(d-j+1)\binom{d}{j-1}p(j)\right\}$$

and

$\Delta_{\text{NCL-Avg}}$ = average delay, at a link, of an intercluster message in the level 2 network

$$= \frac{1}{\mu_{\text{NCL}} - \lambda_{\text{link, NCL}}}.$$

3) Analysis of Other Hierarchical Networks: This section presents the analysis of BH/BR, BH/CC, and BH/CCC hierarchical networks. The analysis is very similar to that presented in the previous section for the BH/BH network. Since only the level 2 network differs, it is obvious that $\lambda_{\text{link, CL, }j}$ and thus Δ_j, Δ_{avg}, and $\Delta_{\text{CL-avg}}$ remain the same as that derived for the BH/BH network. The following expressions can be derived for $\lambda_{\text{link, NCL}}$.

For BH/BR network:

$$\lambda_{\text{link, NCL}} = 2^{d-1}\lambda(1-\alpha)\left[\frac{2^{2(D-d)}}{4(2^{D-d}-1)}\right]$$

For BH/CC network:

$$\lambda_{\text{link, NCL}} = \frac{2^d\lambda(1-\alpha)}{2^{D-d}-1}$$

For BH/CCC network:

$$\lambda_{\text{link, NCL}} = \frac{2^d\lambda(1-\alpha)}{3}\left[\frac{7}{4}D_C - 3 + \frac{D_C+1}{2^{D_c-1}}\right].$$

The average message delivery times are given below.

$$R_{\text{BH/BR}} = \alpha\left\lceil\frac{d}{2}\right\rceil \Delta_{\text{Avg}} + (1-\alpha) \cdot \left[2\Delta_{\text{CL-Avg}} + \left(\frac{2^{2(D-d)}}{4(2^{D-d}-1)}\right)\Delta_{\text{NCL-Avg}}\right] \quad (19)$$

$$R_{\text{BH/CC}} = \alpha\left\lceil\frac{d}{2}\right\rceil \Delta_{\text{Avg}} + (1-\alpha)[2\Delta_{\text{CL-Avg}} + \Delta_{\text{NCL-Avg}}] \quad (20)$$

$$R_{\text{BH/CCC}} = \alpha\left\lceil\frac{d}{2}\right\rceil \Delta_{\text{Avg}} + (1-\alpha) \cdot \left[2\Delta_{\text{CL-Avg}} + \left(\frac{7}{4}D_c - 3 + \frac{D_c+1}{2^{D_c-1}}\right)\Delta_{\text{NCL-Avg}}\right]. \quad (21)$$

4) Discussion of Results: A large number of experiments with varying input parameter values have been performed to assess the performance of the various network architectures under the above queueing analysis. Some sample results that were thought to be representative of those obtained are presented in this section. Fig. 6(a) depicts the delays involved in BH, BH/BH, BH/CCC, and BH/CC networks for the following system parameters: $\lambda = 1$, $\mu_{\text{CL}} = \mu_{\text{NCL}} = 3$, $d = 3$, $D = 9$, and $D_c = 4$ ($N = 512$). The results for the BH/BR hierarchical network are not included in this plot as the noncluster links (i.e., links in the level 2 network) saturate at this load. Fig. 6(b) depicts the *LR ratio* which is defined similar to the *LP ratio* except that the average message delay is used instead of the average internode distance. The *LR ratio* improves as α increases. In the BH/CCC network, links in the level 2 network saturate for small values of α, and thus, R and *LR ratio* values for this network are shown for $\alpha \geq 0.8$ only.

In Fig. 6, the value of α is restricted to $\alpha \geq 0.5$. For α values below this range, the noncluster links of the hierarchical networks saturate with the stated parameter values. It is clear

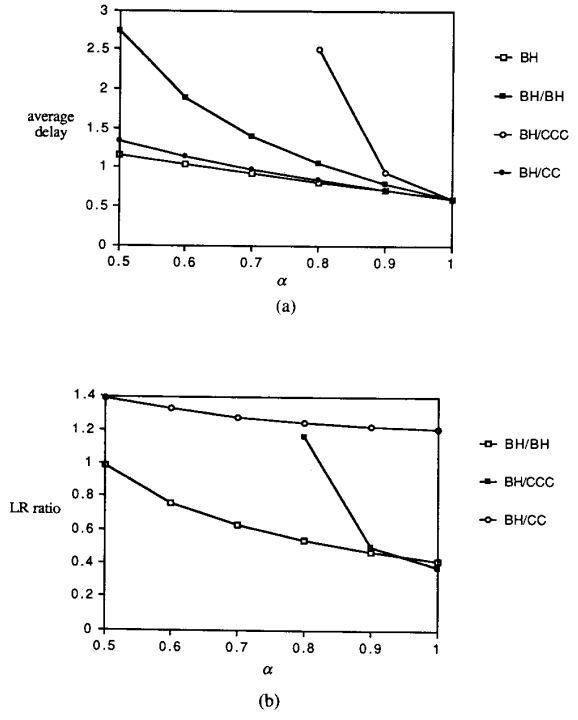


Fig. 6. Average delay and LR ratio as a function of α ($d = 3$, $D = 9$, $D_c = 4$, $\lambda = 1$, $\mu_{CL} = \mu_{NCL} = 3$).

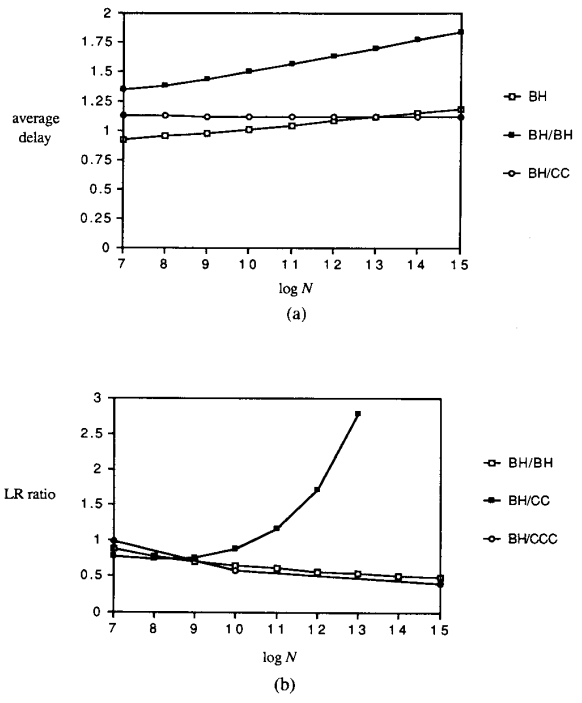


Fig. 7. Average delay and LR ratio as a function of network size.

that HIN's yield superior performance only if there is locality in communication. The value of α above which the HIN's become cost effective depends on several factors such as λ, μ_{CL}, μ_{NCL}, and the system and cluster sizes.

Fig. 7 gives R and *LR ratio* values as a function of the total number of nodes in the network N. The value of N is increased by keeping n fixed at 16 (i.e., $d = 4$) and by increasing the number of clusters. The parameters used are: $d = 4$, $\lambda = 1$, $\mu_{CL} = \mu_{NCL} = 3$, and $\alpha = 0.8$. For the BH/CCC network, μ_{NCL} is chosen as $\mu_{NCL} = i\mu_{CL}$ where i is an integer greater than 0 and may vary with N. The value of i selected is such that the resulting link utilization of the level 2 network links is less than 80%. This increase in μ_{NCL} is taken care of in the network cost by multiplying the number of level 2 links by i. Both BH/BH and BH/CCC networks offer substantial cost–performance ratio improvements for large systems. These conclusions are similar to those obtained with static analysis (Section III-A5). However, for large N, the BH/CCC network requires a much higher message service rate for the noncluster links compared to that required for the BH/BH network.

5) Validation of the Queueing Analysis: To simplify the queueing analysis, it was assumed that whenever a message arrives at a link on its path to its destination, a new service time is generated from the exponential service time distribution. In a real system, of course, the messages would retain their service times, which reflect the lengths of the messages, from the times they are generated to the times they arrive at their destinations. A simulation model was constructed that reflected this reality, and, therefore, allowed validation of the independence assumption used in the analysis. In all other respects, the simulation model matched the analytic model.

Two networks—BH and BH/BH—were selected for simulation experiments. For each network, two plots are presented. One plot gives average delay as a function of α and the other plot gives average delay as a function of the network size N. The value of α is varied from 0.5 to 1.0 in steps of 0.1 and $\log_2 N$ is varied from 5 to 8. The results for the BH network are presented in Fig. 8 and those for the BH/BH network are given in Fig. 9. It can be seen from these plots that the results obtained from the analytical formulas match closely those obtained by simulation, thus justifying the use of Kleinrock's independence assumption in this context.

IV. Performance Enhancements

The performance of HIN's can be improved in several ways. This section briefly describes two of these. Section IV-A discusses the improvement possible by replicating links in the level 2 network while the following section presents the impact of improving routing algorithms.

A. Replication of Links in the Level 2 Network

It is important to note that it is the clustering within HIN's that makes potentially reasonable the replication of some links. The potential benefits here do not merely represent a cost–benefit tradeoff, but rather indicate the advantages of (to use a telephone network analogy) a clustered organization in which a few higher capacity "trunk lines" may be profitably utilized.

The plots in Fig. 10 give link utilizations for the BH/BH and

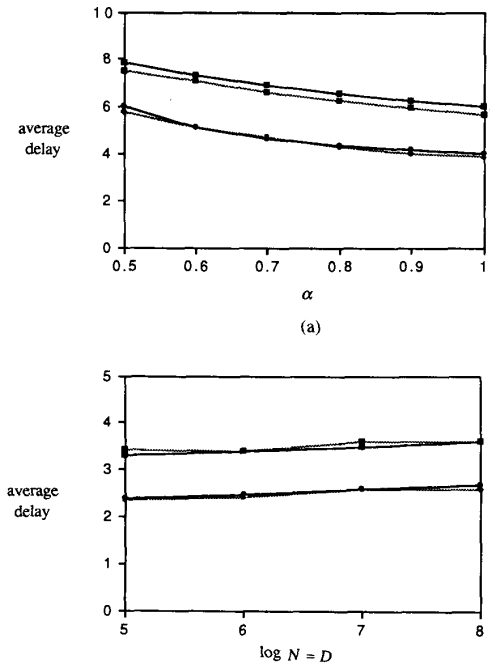


Fig. 8. Comparison of analytical and simulation results for the BH network: —— analytical ---- simulation. (a) $\lambda = 1, \mu_{CL} = \mu_{NCL} = 0.75$ for all lines. ■: $D = 6, d = 3$, ●: $D = 3, d = 2$. (b) $\lambda = 1, \mu_{CL} = \mu_{NCL} = 1, \alpha = 0.8$ for all lines. ■ $d = 3$, ●: $d = 2$.

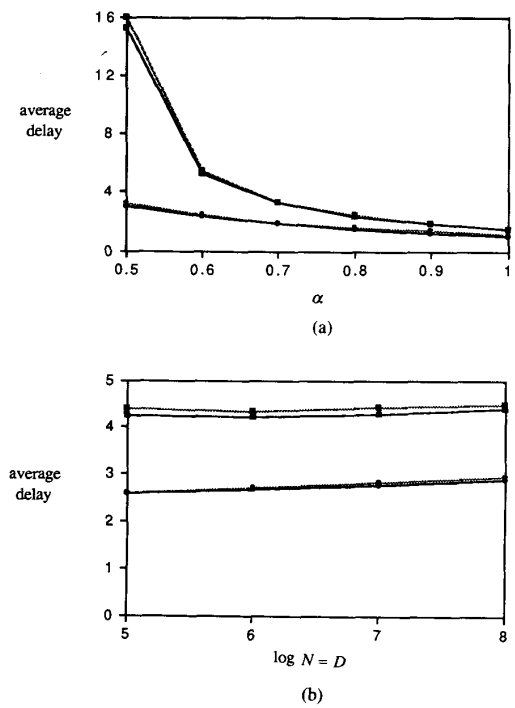


Fig. 9. Comparison of analytical and simulation results for the BH/BH network: —— analytical ---- simulation. (a) $\lambda = 1, \mu_{CL} = 1.5, \mu_{NCL} = 3$ for all lines. ■: $D = 6, d = 3$ ●: $D = 3, d = 2$. (b) $\lambda = 1, \alpha = 0.8$ for all lines. ■: $d = 3, \mu_{CL} = 1.2, \mu_{NCL} = 1.5$, ●: $d = 2, \mu_{CL} = \mu_{NCL} = 1.2$.

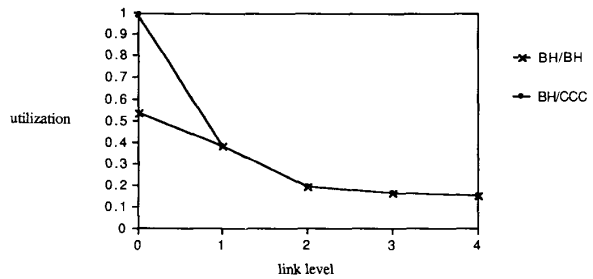


Fig. 10. Link utilizations of the BH/BH and BH/CCC networks ($d = 4$, $D = 15, D_c = 8, \lambda = 1, \mu_{CL} = 3, \alpha = 0.8$). $\mu_{NCL} = \mu_{CL}$ for the BH/BH network and $\mu_{NCL} = 4\mu_{CL}$ for the BH/CCC network.

BH/CCC networks with the following parameters: $D = 15$, $d = 4, D_c = 4, \alpha = 0.8, \lambda = 1, \mu_{CL} = 3, \mu_{CL} = \mu_{NCL}$ for the BH/BH network, and $\mu_{NCL} = 4\,\mu_{CL}$ for the BH/CCC network. Here $j = 0$ is used to represent the noncluster (level 2) links. This plot suggests that by replicating the links in the level 2 network, effectively increasing μ_{NCL}, the average delay can be reduced. If the reduction in the average delay compensates for the increase in the number of links, the overall *LR ratio* improves (decreases). This section investigates the effect of this replication on the performance of HIN's.

Fig. 11 shows the impact of replicating links in the level 2 network for the BH/BH and BH/CCC networks. The parameters used are: $d = 4$, $D = 15$, $\lambda = 1$, $\mu_{CL} = 3$, and $\alpha = 0.8$. The value of μ_{NCL} is varied to change the ratio μ_{NCL}/μ_{CL}. The μ_{NCL}/μ_{CL} ratio gives the number of parallel wires used to implement a logical link. These plots suggest that there is an optimum value ("knee") for μ_{NCL} at which the *LR ratio* is the smallest. Ideally, we would like to design the level 2 network such that the operating point is at or close to the knee. For the BH/BH network, the *LR ratio* is the smallest (allowing only integral μ_{NCL}/μ_{CL} ratios) when $\mu_{NCL} = 2\mu_{CL}$; for the BH/CCC network, it is the smallest when $\mu_{NCL} = 8\mu_{CL}$. It is important to note that the knee is quite broad indicating that there are several operating points that are nearly optimal. Thus, designing a network for near-optimum performance would apparently be straightforward. It appears from these data that the BH/CCC network can potentially achieve very similar *LR ratios* as for the BH/BH network, but that this requires a much greater replication of links in the level 2 network. In this example, the links in the level 2 network of the BH/CCC network require a replication factor that is four times that of the corresponding links in the BH/BH network.

B. Impact of Improved Routing Algorithms

The queueing analysis, and the simulation experiments described in Section III-B5, used a routing algorithm that chooses randomly among shortest paths. In this routing algorithm, if a message at a node can be routed towards its destination on any of l links attached to the node, one of these l links is selected randomly. This works comparatively well with the BH network (under the assumed workload) because both intra- and intercluster messages are uniformly distributed over the links within a cluster. However, in HIN's the inter-

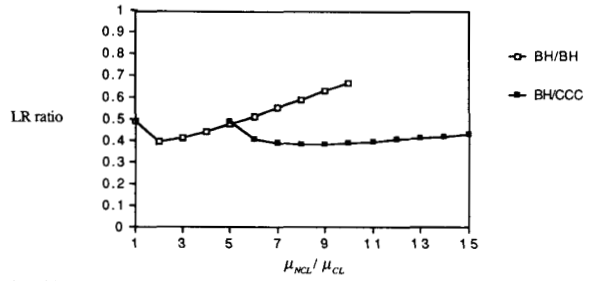


Fig. 11. Impact of replicating links within the level 2 network ($d = 4$, $D = 15$, $D_c = 8$, $\lambda = 1$, $\mu_{CL} = 3$, $\alpha = 0.8$).

cluster message density tends to be higher for the links (in a cluster) closer to the interface node. Thus, if these links can be utilized less by the intracluster messages, the overall message delivery time can be reduced. Several routing algorithms can be devised to achieve this goal. This section is intended to estimate the performance improvement possible with any algorithm that tries to equalize link utilization within clusters. Design of a specific algorithm must take into account a number of technological factors since implementation in hardware is desired (i.e., VLSI issues), and is outside the scope of this paper.

Several simulation experiments were conducted on the BH/BH network using an abstract routing algorithm that works as follows. For each node in a cluster, the cumulative number of messages that have passed through each of the d cluster links is maintained. When a message has to be routed, the link that has the lowest cumulative message count (among the links that can be used for this message) is selected. Note that real routing algorithms (such as one based on "shortest-queue" routing) could be expected to perform somewhat better. Our purpose here is to show only that the routing algorithms that attempt to equalize link utilizations within clusters are worthwhile, and to provide a lower bound on the performance improvements that they would offer.

Many experiments were conducted with different parameters. Selected results are presented in Fig. 12 that illustrate well the outputs of these experiments. The upper two curves are for $D = 6$, $d = 3$, $\lambda = 1$, $\mu_{CL} = 1.5$, and $\mu_{NCL} = 3$; the lower two curves are for $D = 3$, $d = 2$, $\lambda = 1$, $\mu_{CL} = 1.5$, and $\mu_{NCL} = 3$. In general, the improvement in performance increases with increasing utilizations of the most heavily utilized links (as may be caused by increasing the cluster size, decreasing α, and/or by increasing λ). For example, when $\alpha = 0.5$, the average delay for the larger network reduces from 15.2 to 6.3 since in this case the cluster links connected to the interface node have a utilization of 94%. For the same network, when $\alpha = 0.6$, these links have a utilization of 82% and the average delay reduces from 5.3 to 3.8. These results suggest that implementing improved routing algorithms would quite likely be worthwhile. We are presently working on divising specific routing algorithms that are both efficient and implementable in HIN's.

V. Summary

This paper has proposed the use of hierarchical interconnection networks to exploit locality in communication, and

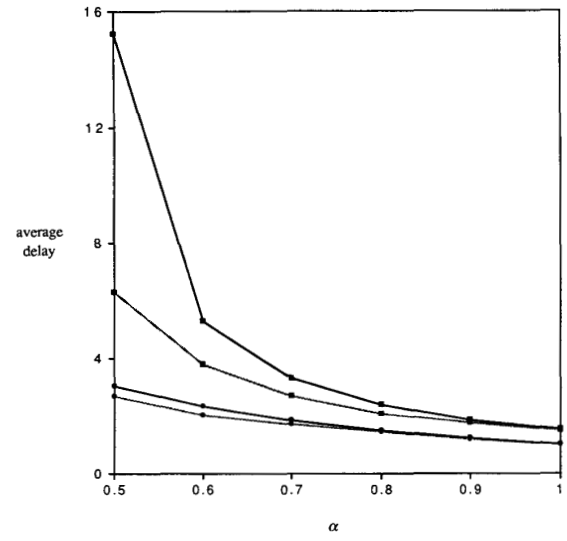


Fig. 12. Effect of improved routing algorithm on the BH/BH network. $\lambda = 1$, $\mu_{CL} = 1.5$, $\mu_{NCL} = 3$ for all lines. ■: $D = 6$, $d = 3$, ●: $D = 3$, $d = 2$. —— random routing algorithm, ---- improved routing algorithm.

also to serve as a framework for integrating different network topologies. Some example networks have been analyzed under the assumption of locality in communication. By means of static analysis and queueing analysis, evidence was presented suggesting the suitability of the hierarchical networks. In particular, the BH/BH and BH/CCC hierarchical networks reduce the link cost substantially at the expense of only moderately increasing the average internode distance and the average message delivery time. Thus, hierarchical networks allow more nodes to be included in the system when constrained by the link cost.

The queueing analysis provided a much deeper insight into the performance issues. This analysis showed that the links in the higher level network might need greater service rates than the links in the lower level network; this can be accomplished economically by replicating these links. By using a routing algorithm that uses links more effectively, the performance of hierarchical interconnection networks can be improved further.

The future may see an abundance of special-purpose systems tailored to specific applications. Technological developments may also yield reconfigurable systems much like those proposed by Snyder [26], [27], with performance superior to those of the static hierarchical networks proposed here. Hierarchical networks provide tailoring to a general phenomenon underlying many parallel computations ("locality") while remaining "general-purpose," with current technology.

Acknowledgment

We thank the referees for their critical comments on a previous version of this paper. The authors gratefully acknowledge financial support provided by the Natural Sciences and Engineering Research Council of Canada and by the University of Saskatchewan.

REFERENCES

[1] D. P. Agrawal, V. K. Janakiram, and G. C. Pathak, "Evaluating the performance of multicomputer configurations," *IEEE Comput. Mag.*, vol. 19, pp. 23-37, 1986.

[2] D. P. Agrawal and I. E. O. Mahgoub, "Performance analysis of cluster-based supersystems," in *Proc. 1st Int. Conf. Supercomput. Syst.*, IEEE Computer Society Press, 1985, pp. 593-602.

[3] L. N. Bhuyan and D. P. Agrawal, "A general class of processor interconnection strategies," in *Proc. 9th Symp. Comput. Architecture*, 1982, pp. 26-29.

[4] D. Carlson, "The mesh with a global mesh: A flexible, high-speed organization for parallel computation," in *Proc. 1st Int. Conf. Supercomput. Syst.*, IEEE Computer Society Press, 1985, pp. 618-627.

[5] W. Crowther *et al.*, "Performance measurements on a 128-node butterfly parallel processor," in *Proc. 1985 Int. Conf. Parallel Processing*, 1985, pp. 531-540.

[6] S. P. Dandamudi, "Hierarchical interconnection networks for multicomputer systems," Ph.D. dissertation, Dep. Computat. Sci., Univ. Saskatchewan, Saskatoon, 1988.

[7] A. M. Despain and D. A. Patterson, "X-tree: A tree structured multiprocessor computer architectures," in *Proc. 5th Symp. Comput. Architecture*, 1978, pp. 144-151.

[8] T-Y. Feng, "A survey of interconnection networks," *IEEE Comput. Mag.*, vol. 14, pp. 12-27, 1981.

[9] D. Gajski, D. Kuck, D. Lawrie, and A. Sameh, "CEDAR," Dep. Comput. Sci., Univ. of Illinois, Urbana, 1983. (Reprinted in *Tutorial on Supercomputers: Design and Applications*, K. Hwang, Ed., IEEE Computer Society Press, 1983, pp. 251-275.)

[10] G. Gopal and J. W. Wong, "Delay analysis of broadcast routing in packet-switching networks," *IEEE Trans. Comput.*, vol. C-30, pp. 915-922, 1981.

[11] A. Gottlieb *et al.*, "The NYU Ultracomputer—Designing a MIMD, shared memory parallel machine," *IEEE Trans. Comput.*, vol. C-32, pp. 175-189, 1983.

[12] J. A. Harris and D. R. Smith, "Hierarchical multiprocessor organizations," in *Proc. 4th Symp. Comput. Architecture*, 1977, pp. 41-48.

[13] J. P. Hayes, T. N. Mudge, Q. F. Stout, S. Colley, and J. Palmer, "Architecture of a hypercube supercomputer," in *Proc. 1986 Int. Conf. Parallel Processing*, 1986, pp. 653-660.

[14] W. D. Hillis, *The Connection Machine*. Cambridge, MA: MIT Press, 1985.

[15] E. Horowitz and A. Zorat, "The binary tree as an interconnection network: Applications to multiprocessor systems and VLSI," *IEEE Trans. Comput.*, vol. C-30, pp. 247-253, 1981.

[16] L. Kleinrock, *Queueing Systems: Vol. 1*. New York: Wiley, 1975.

[17] ——, *Queueing Systems: Vol. 2*. New York: Wiley, 1976.

[18] D. Lawrie, "Access and alignment of data in an array processor," *IEEE Trans. Comput.*, vol. C-24, pp. 1145-1155, 1975.

[19] E. D. Lazowska, J. Zahorjan, G. S. Graham, and K. C. Sevcik, *Computer System Performance*. Englewood Cliffs, NJ: Prentice-Hall, 1984.

[20] D. D. Leondorf, "Development and use of an asynchronous MIMD computer for finite element analysis," in *Algorithmically Specialized Parallel Computers*, L. Snyder, L. H. Jamieson, D. B. Gannon, and J. H. Siegel, Eds. New York: Academic, 1985, pp. 213-222.

[21] D. Nassimi and S. Sahni, "Bitonic sort on a mesh-connected parallel computer," *IEEE Trans. Comput.*, vol. C-28, pp. 2-7, 1979.

[22] ——, "Finding connected components and connected ones on a mesh-connected parallel computer," *SIAM J. Comput.*, vol. 9, pp. 744-757, 1980.

[23] G. F. Pfister, W. C. Brantley, D. A. George, S. L. Harvey, K. P. Kleinfelder, E. A. Melton, V. A. Norton, and J. Wiess, "The IBM Research Parallel Processor Prototype (RP3): Introduction and architecture," in *Proc. 1985 Int. Conf. Parallel Processing*, 1985, pp. 764-771.

[24] F. P. Preparata and J. Vuillemin, "The cube-connected-cycle: A versatile network for parallel computation," *Commun. ACM*, vol. 24, pp. 300-309, 1981.

[25] C. L. Seitz, "The Cosmic Cube," *Commun. ACM*, vol. 28, pp. 22-33, 1985.

[26] L. Snyder, "Introduction to the configurable highly parallel computer," *IEEE Comput. Mag.*, vol. 15, pp. 47-56, 1982.

[27] ——, "Supercomputers and VLSI: The effects of large-scale integration on computer architecture," in *Advances in Computers*. New York, Academic, 1984, pp. 1-33.

[28] ——, "Type architectures, shared memory, and the corollary of modest potential," *Ann. Rev. Comput. Sci.*, vol. 1, pp. 289-317, 1986.

[29] Q. F. Stout, "Mesh-connected computers with broadcasting," *IEEE Trans. Comput.*, vol. C-32, pp. 826-830, 1983.

[30] C. W. Strevens, "The Transputer," in *Proc. 12th Symp. Comput. Architecture*, 1985, pp. 292-300.

[31] R. J. Swan, S. H. Fuller, and D. Siewiorek, "Cm*—A modular, multiprocessor," in *Proc. Nat. Comput. Conf.*, 1977, pp. 39-46. (Reprinted in *Tutorial on Parallel Processing*, R. H. Kuhn and D. A. Padua, Eds. IEEE Computer Society Press, 1981, pp. 146-153.)

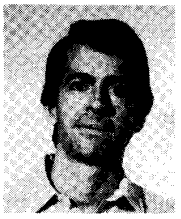
[32] C. D. Thompson and H. T. Kung, "Sorting on a mesh-connected parallel computer," *Commun. ACM*, vol. 20, pp. 263-271, 1977.

[33] L. D. Wittie, "Communication structures for large networks of microcomputers," *IEEE Trans. Comput.*, vol. C-30, pp. 264-273, 1981.

[34] S. W. Wu and M. T. Liu, "A cluster structure as an interconnection network for large multicomputer systems," *IEEE Trans. Comput.*, vol. C-30, pp. 254-264, 1981.

Sivarama P. Dandamudi (S'83-M'89) was born in Andhra Pradesh, India. He received the B.E. degree from the University of Mysore, India, the M. Tech. degree in electrical engineering from the Indian Institute of Technology, Madras, and the M.Sc. and Ph.D. degrees in computer science from the University of Saskatchewan, Saskatoon, Sask., Canada in 1984 and 1988, respectively.

Currently, he is an Assistant Professor in the School of Computer Science at Carleton University, Ottawa, Ont., Canada. His research interests include parallel and distributed systems, database systems, performance evaluation, and computer architecture.



Derek L. Eager (M'87) received the B.Sc. degree in computer science from the University of Regina in 1979, and the M.Sc. and Ph.D. degrees in computer science from the University of Toronto, Toronto, Ont., Canada, in 1981 and 1984, respectively.

Currently, he is an Associate Professor in the Department of Computational Science at the University of Saskatchewan. His research interests are in the areas of performance modeling, parallel systems, and distributed systems.