1981

Cost-Performance Bounds for Multi-Microcomputer Networks

Daniel A. Reed

Herbert D. Schwetman

Report Number:

81-382

Cost-Performance Bounds for Multi-Microcomputer Networks

Daniel A. Reed
Herbert D. Schwetman

Department of Computer Sciences
Purdue University
West Lafayette, IN 47907

CSD-TR-382

ABSTRACT

Several interconnection structures for a distributed multi-microcomputer message passing system are compared on the basis of cost and performance. Among the structures analyzed are buses, double rings, D-dimensional toroids, trees, cube-connected cycles, and chordal rings. Network cost is defined in terms of the number of network nodes and the unit cost of communication links and their associated connections. Simple asymptotic performance bounds are derived based on the bottleneck analysis of a queueing network. In contrast to the usual assumption of uniform message routing, the technique permits the introduction of a reference locality notion to the message routing behavior of network nodes. Finally, the cost, performance, and performance/cost functions are examined as the number of network nodes becomes very large.

October 28, 1981

Cost-Performance Bounds for Multi-Microcomputer Networks

Daniel A. Reed

Herbert D. Schwetman

Department of Computer Sciences

Purdue University

West Lafayette, IN 47907

Introduction

In recent years, many researchers have sought ways to exploit the rapid development of LSI/VLSI technology in the construction of powerful computer systems. Proposals for multiple processor systems containing up to 10^6 VLSI chips have been made [Sull77, Witt76]. At first appearance, networks of thousands of processors may not seem justifiable. There are, however, at least two primary motivations for developing such systems. The most obvious is the need to overcome the fundamental physical limits on computation speed imposed by sequential processing. The need for performance increases of factors of 100 or even 1000 is painfully obvious to workers in such fields as speech analysis, weather modeling, and nuclear fusion research. Only by injecting parallelism into the solution of such problems can one realistically expect to obtain truly large performance increases. Second, it has been suggested that large multiple processor systems will provide appropriate architectural support for new language proposals. In particular, the functional programming languages proposed by Backus [Back78] and the communicating sequential processes of Hoare [Hoar78] seem ideally suited to multiple processor systems whose computational tasks communicate via message passing.

Many ways to interconnect multiple processors have been proposed, but no real consensus on a best proposal has yet emerged. Not only is there a paucity of knowledge concerning the effect of various interconnection structures on performance, there is also no widely accepted method for modeling such structures. This, coupled with the large number of design parameters for parallel systems, has made comparison difficult.

Overview

The context of our discussion is Wittie's *network computer* [Witt81], an MIMD (*M*ultiple *I*nstruction *M*ultiple *D*ata stream) system whose active computing nodes communicate by passing messages to one another over passive communication links. Nodes do not share any memory; all communication is performed by message passing. Each network node is assumed to consist of a processing element with some local memory, a communication processor capable of routing messages without delaying the processing element, and some (small) number of connections to communication links connecting the node to other nodes.

On such a network computer, a parallel computation may require multiple processing elements that exchange messages while executing cooperating tasks. There is no global synchronization among processing elements. Instead, computation at each processing element proceeds independently of all others except when the processing element passes a message to or receives a message from the communication processor.

The interconnection networks over which messages are passed can be broadly classified as reconfigurable multistage switching networks and passive-link interconnections. There is a considerable body of literature comparing reconfigurable multistage switching networks such as banyans [Goke73] and shuffle-exchange [Lang76]. Since these structures have generally been considered for SIMD (*S*ingle *I*nstruction *M*ultiple *D*ata stream) machines where all processing elements execute the same instruction in lock step, they are not discussed further here. Instead, passive-link structures whose nodes are embedded in the interconnection network are emphasized **(see Figure I).** For example, we compare the single bus, double ring, D-dimensional toroid, bus hypercube, cube-connected cycles, chordal ring, and tree, among others, on the basis of cost and performance.

The cost of each structure is defined as a function of the number of network nodes and the unit cost of communication links and their associated connections. Cost is significant only because it allows us to examine performance/cost ratios for various interconnection networks.

Many definitions of network performance have been proposed (e.g., average message delay, message density, and bus load). These notions are usually based on the assumption that the message routing distribution is uniform (i.e., the probability that node i sends messages to node j is the same for all i and j, $i \neq j$) and that nodes generate messages at some fixed rate. We present an alternative definition of network performance based on the asymptotic or bottleneck behavior of a queueing network that relaxes this assumption. In mapping a distributed computation onto an interconnection structure, one would hope that those tasks communicating with high frequency are placed physically close to one another in the interconnection network. Clearly this results in a message routing distribution that is significantly different from the usual assumption of uniform routing. To reflect this non-uniformity, we introduce a notion of reference locality to the message routing distribution. Furthermore, we allow the rate at which nodes generate messages to depend on the rate at which messages arrive at the nodes.

Since Wittie [Witt81] recently analyzed a subset of the structures considered here under the uniform routing assumption and provided order of magnitude values for the density of messages on links and the average number of links traversed by a message, our results can be viewed as both a refinement and an extension of his.

To simplify the presentation, we first discuss the methods used to derive cost and performance functions, and then apply these methods to several proposed networks. The notation employed throughout the remainder of the paper is summarized in **Table I.**

Cost Function

As we noted earlier, each node of the system is assumed to consist of a processing element (PE), communication processor (CP), and some number of link connections (LC) joining the node to communication links (CL). We define the following simple cost function:

$$\begin{aligned}&Cost\,(Net-type\,, Net-size\,, C_{PE}, C_{CL}, C_{LC}) = \\ &\qquad C_{PE} * Net-size\ + \\ &\qquad C_{LC} * Net-size * (number\ of\ connections\ per\ node\,) + \\ &\qquad C_{CL} * (number\ of\ links\,)\end{aligned}$$

where the following definitions apply

$Net-type$	type of interconnection structure
$Net-size$	number of nodes in the structure
C_{PE}	unit cost of a $PE-CP$ pair
C_{LC}	unit cost of a link connection
C_{CL}	unit cost of a communication link

A word of caution is in order about the unit cost of communication links. Links can be of two types, dedicated links between two nodes or buses shared by two or more nodes. In the first case, C_{CL} is simply the cost of each link. In the second case, we assume C_{CL} is the cost of the bus divided by the number of connections to it. Cost function parameters for the interconnections discussed in the remainder of the paper can be found in **Table II.**

Asymptotic Performance Function

Our performance analysis is based on asymptotic or bottleneck analysis. While its essentials are briefly reviewed here, the reader should consult Denning and Buzen [DeBu78] for complete details and a statement of the assumptions involved in the approach.

Each time a node sends a message to another node, the message must cross some number of communication links and pass through some intermediate nodes before reaching its destination processing element. At the destination, it causes some computation to take place. If we consider all possible source-destination pairs and the probability that they exchange messages, we can calculate the number of visits to each communication link and processing element made by an average message. Now consider such an average message and an arbitrary device i (either a node or a link). This average message will visit device i a certain number of times. This mean number of visits is called this visit ratio of device i and is denoted by V_i. Similarly, let S_i denote the mean time required for device i to service a message, X_i denote the mean rate of message completions at device i ($X_i \leq 1/S_i$), and U_i denote the utilization of device i. The following laws are then known to hold:

$$U_i = X_i S_i \qquad \text{Utilization Law}$$

$$X_0 = \frac{X_i}{V_i} \qquad \text{Forced Flow Law}$$

where X_0 is the message completion rate of the entire system. Simple algebra yields

$$X_0 = \frac{U_i}{V_i S_i}$$

As the number of messages in the system becomes large, the utilization of the device with the largest $V_i S_i$ product must approach one (1). Hence, the maximum value of the system message completion rate is

$$X_0 \le \frac{1}{V_b S_b}$$

where

$$V_b S_b = \max_i V_i S_i$$

In their general definition, the visit ratios are only unique up to a normalizing constant. To insure their uniqueness in our analysis, we normalize the V_i for the nodes such that their sum is one (1). The $V_i S_i$ product can then be interpreted as the total service requirement of a message at device i. Summing the $V_i S_i$ over all i gives the total service requirement of a message in the system.

To simplify analysis, we assume that all processing elements have the same mean service time S_{PE} and all links have the same mean service time S_{CL}. We also assume that each node has the same message routing distribution. By this, we mean that each node i has the same probability of sending a message to a node reachable by traversing l links for all i. Messages follow the path requiring the smallest number of link traversals to reach their destination. If there are multiple shortest paths, we assume they are visited with equal probability unless otherwise specified. Message delays due to internal routing at the communication processors of intermediate nodes are ignored. We model *only* the queueing delays and service times at the communication links and the destination processing element.

The remainder of our analysis is devoted to derivation of the maximum system message completion rate X_0 for various interconnection networks. This performance function X_0 differs in several significant ways from earlier performance metrics for distributed systems. Rather than fixing the message completion rate at the nodes and then determining the minimum message density that must be supported by the links to attain this rate, one can actually determine the message completion rate given the visit ratios and the mean service times for the processing elements and communication links. As we shall see, one can also systematically determine the effect of varying the number of network nodes and device mean service times.

Uniform Message Routing - Symmetric Structures

Messages sent by each node of a symmetric interconnection structure can reach the same number of nodes by traversing l communication links for all l. A bi-directional ring system is a simple example of a symmetric interconnection since each message can always reach two nodes by crossing l links. Under uniform message routing, the probability of node i sending a message to node j is the same for all i and j, $i \neq j$. We assume that nodes do not send message to themselves, hence $i \neq j$.

Consider such a symmetric structure with K nodes obeying the uniform routing assumption. Since each processing element is visited with equal probability by an average message, the visit ratio for the processing elements is just

$$V_{PE} = \frac{1}{K}$$

Similarly, all communication links must be visited with equal probability. Suppose we look at an arbitrary network node and the K-1 possible destinations for messages sent from that node. Define, $Reach(l, Net-type)$ as the number of nodes reachable from an arbitrary node by crossing l links in a network of type $Net-type$. The average number of links traversed by a message is $LV_{symmetric}^{uniform}$ (Uniform routing, Symmetric structure) and is given by

$$LV^{uniform}_{symmetric} = \frac{\sum_{l=1}^{lmax} l \; Reach(l, Net-type)}{K - 1}$$

where $lmax$ is the maximum number of links that must be crossed to reach any node.

Now define $Numlinks(K, Net-type)$ as the number of communication links in a network of size K and type $Net-type$. The link visit ratio is then simply

$$V_{CL} = \frac{LV^{uniform}_{symmetric}}{Numlinks(K, Net-type)}$$

We immediately have

$$X_0 \leq \frac{1}{\max\left\{V_{PE}S_{PE} \; , \; V_{CL}S_{CL}\right\}} = \min\left\{\frac{1}{V_{PE}S_{PE}} \, , \; \frac{1}{V_{CL}S_{CL}}\right\}$$

Local Message Routing - Symmetric Structures

Now suppose the assumption of a uniform message routing distribution is relaxed. Each node of the structure is allowed to have a symmetric locality surrounding it that is visited with some high probability φ while the nodes outside the locality are visited with probability $1 - \varphi$.

Let $LocSize(L, Net-type)$ be defined as

$$LocSize(L, Net-type) = \sum_{l=1}^{L} Reach(l, Net-type)$$

Then the $LocSize(L, Net-type)$ nodes reachable in L or fewer links from a node constitute its locality and are visited with probability φ while the $K - LocSize(L, Net-type) - 1$ other nodes are visited with probability $1-\varphi$.

Since the interconnection network is symmetric, *each* node is contained in the localities of $LocSize(L, Net-type)$ other nodes and is outside the localities of $K - LocSize(L, Net-type) - 1$ nodes. Thus, each node is still visited with equal probability, and the processing element visit ratio is just

$$V_{PE} = \frac{1}{K}$$

To obtain link visit ratios, consider again an arbitrary source node and all K-1 possible message destinations. The mean number of communication links traversed by a message $LV^{local}_{symmetric}$ is

$$LV^{local}_{symmetric} = \frac{\varphi \sum_{l=1}^{L} l \; Reach(l,Net-type)}{\sum_{l=1}^{L} Reach(l,Net-type)} + \frac{(1-\varphi) \sum_{l=L+1}^{lmax} l \; Reach(l,Net-type)}{K - \sum_{l=1}^{L} Reach(l,Net-type) - 1}$$

$$= \frac{\varphi \sum_{l=1}^{L} l \; Reach(l,Net-type)}{LocSize(L,Net-type)} +$$

$$\frac{(1-\varphi)\left[LV^{uniform}_{symmetric}\,(K-1) - \sum_{l=1}^{L} l \; Reach(l,Net-type)\right]}{K - LocSize(L,Net-type) - 1}$$

The first term is simply the product of the average number of links traversed while visiting a node in the locality and the probability of visiting the locality φ. The second term has a similar interpretation for nodes outside the locality. The link visit ratio is then

$$V_{CL} = \frac{LV^{local}_{symmetric}}{Numlinks(K,Net-type)}$$

and the system message completion rate is bounded by

$$X_0 \leq \min \left\{ \frac{1}{V_{PE} S_{PE}},\ \frac{1}{V_{CL} S_{CL}} \right\}$$

Uniform Message Routing - Asymmetric Structures

In an asymmetric interconnection structure the number of nodes reachable in L links from a given node depends on the location of the source node in the network. Primary examples are b-ary trees and snowflakes [FiSo80].

Under uniform message routing, each node is visited with equal probability so the processing element visit ratio is again

$$V_{PE} = \frac{1}{K}$$

To derive the link visit ratios, consider some interval during which each node sends $K - 1$ messages (each node receives $K - 1$ messages) and the total number of messages sent is $K(K - 1)$. For each communication link j, calculate the number of messages that cross that link; call this number $Msg\,(j\,,Net\,-type\,)$. The visit ratio for link j is

$$V_{CLj} = \frac{Msg\,(j\,,Ket\,-type\,)}{K(K - 1)}$$

The maximum link visit ratio is

$$V_{CL}^{max} = \max_{j} V_{CLj}$$

and the system message completion rate is bounded by

$$X_0 \le \min\left\{\frac{1}{V_{PE}S_{PE}}\,,\ \frac{1}{V_{CL}^{max}S_{CL}}\right\}$$

Interconnection Structures

The techniques described above have been applied to eleven often cited interconnection structures: seven symmetric ones and four asymmetric ones. An example of each structure is shown in **Figure I.** Space, unfortunately, does not permit detailed derivations of the results for each interconnection; for a complete exposition see [Reed82]. To provide some insight into the technique's application, the spanning bus hypercube, a symmetric structure, and the snowflake, an asymmetric structure, are analyzed in detail. For the other structures, only a simple description of salient points is provided. The results of the cost and performance analyses are summarized in **Tables II-IV** and will be referred to frequently in the remaining discussion.

Symmetric Structures

Spanning Bus Hypercubes (SBH)

The spanning bus hypercube [Witt81] is a D-dimensional structure connecting each node to D buses in D orthogonal dimensions; w nodes share a bus in each dimension. This structure is identical to a D-dimensional w-wide lattice except the w connections in each dimension are replaced with a single bus.

Wittie [Witt81] gives a simple distributed routing algorithm for spanning bus hypercubes. Consider the routing of a message between two arbitrary nodes A and B. The node addresses of A and B can be expressed as D, base w, coordinates in a w^D lattice. Compare the ith coordinates of A and B. If they differ, route the message along the ith dimension bus to the node whose ith coordinate is equal to that of B. Repeat this process until all D coordinate positions agree. Since each move brings the message closer to its destination in one dimension, the order in which the D coordinates are checked does not matter.

Since each of the w^D nodes has D connections, there are Dw^D total connections. Each bus is shared by w nodes so there are Dw^{D-1} buses. Recalling that the cost of a bus is proportional to the number of connections to it, the cost function is

$$Cost(SBH, C_{PE}, C_{LC}, C_{CL}) = w^D(C_{PE} + D(C_{LC} + C_{CL}))$$

To derive link visit ratios for uniform message routing, consider again the base w representation of an arbitrary source-destination pair. Any two of the D coordinate positions differ with probability $\frac{w-1}{w}$. Since each of these D coordinate positions is independent, the average number of buses traversed by a message is

$$LV_{SBH}^{uniform} = \left[\frac{D(w-1)}{w}\right]\left[\frac{w^D}{w^D-1}\right] = \frac{Dw^{D-1}(w-1)}{w^D-1}$$

The correction factor $\frac{w^D}{w^D-1}$ accounts for the fact that the source and destination must differ. The $V_i S_i$ products are then

$$V_{PE}S_{PE} = \frac{S_{PE}}{w^D} \quad \text{and} \quad V_{CL}S_{CL} = \frac{S_{CL}Dw^{D-1}(w-1)}{Dw^{D-1}(w^D-1)} = \frac{S_{CL}(w-1)}{w^D-1}$$

and

$$X_0 \le \min \left\{ \frac{w^D}{S_{PE}} \text{ , } \frac{w^D - 1}{S_{CL}(w - 1)} \right\}$$

Because of fanout limitations, D must fixed at a small constant and the system size increased by increasing w. If D is fixed and w increases, the buses become the performance bottlenecks, and performance increases at approximately the rate $\frac{w^{D-1}}{S_{CL}}$.

To see the effect of locality on performance, consider the number of ways source and destination addresses can differ in l positions. Since there are w-1 ways each position can differ and each position is independent, this number is $(w - 1)^l$. There are $\binom{D}{l}$ ways to select l positions so there are

$$Reach(l, SBH) = \binom{D}{l}(w - 1)^l$$

nodes reachable using exactly l buses. The size of the reference locality is

$$LocSize(L, SBH) = \sum_{l=1}^{L} Reach(l, SBH)$$

(Recall that L is the maximum distance to any node in the reference locality.) Then the mean number of link visits by a message is

$$LV_{SBH}^{local} = \frac{\varphi \sum_{l=1}^{L} l \binom{D}{l}(w - 1)^l}{LocSize(L, SBH)} + \frac{(1 - \varphi)\left[Dw^{D-1}(w - 1) - \sum_{l=1}^{L} l \binom{D}{l}(w - 1)^l \right]}{w^D - LocSize(L, SBH) - 1}$$

The $V_i S_i$ products are

$$S_{PE} V_{VE} = \frac{S_{PE}}{w^D} \quad \text{and} \quad V_{CL} S_{CL} = \frac{S_{CL} LV_{SBH}^{local}}{Dw^{D-1}}$$

and the bound on the system message processing rate is

$$X_0 \le \min \left\{ \frac{w^D}{S_{PE}}, \frac{Dw^{D-1}}{S_{CL} LV_{SBH}^{local}} \right\}$$

As w increases, the bound for the system message completion rate, X_0, increases at the rate $\frac{w^{D-1}}{S_{CL}(1 - \varphi)}$. If one compares this with the uniform routing case, it becomes clear that this definition of locality does not change the order of the performance bound, only the constant of proportionality.

Single Global Bus

The simplest possible interconnection drops all K nodes of a system from a single global bus. One communication link traversal is required to route any message from source to destination. Because of this, no notion of a message routing distribution is relevant. Unfortunately, the single bus rapidly becomes the system bottleneck and bounds system performance by the reciprocal of its mean service time.

Complete Connection

The most expensive and best performing interconnection provides direct links between all pairs of the K system nodes. The prohibitive $O(K^2)$ interconnection cost makes this approach unsuitable for large systems, but it provides a useful point of reference. Since one link traversal suffices to reach any destination, no notion of message routing distribution is relevant here either.

Double Ring

Several proposals for cyclic or ring interconnections have been made [Liu78, Jafa78]. Typically, messages can pass in only direction around the ring. Performance improves if each node is connected to two counter-rotating rings. A node sending a message places it on the ring requiring the smallest number of link traversals to reach its destination. After traversing a link, a message queues for service on the next link in the direction of its travel until its destination is reached. Hence, no message ever needs to traverse more than $\left\lfloor \frac{K}{2} \right\rfloor$ links in a K node system.

Since messages can travel varying distances along the circumference of a ring, it is possible to define a node's reference locality. In this case, a node's locality is just all nodes lying on an arc of length $2L$ centered at the node (i.e., the nearest $2L$ nodes).

D-Dimensional Toroid

The D-dimensional toroid (D-dimensional w-wide lattice) connects each of its w^D nodes to a ring of size w in each of the D orthogonal dimensions. Because of this, no message need traverse more than $\left\lfloor \frac{w}{2} \right\rfloor$ links in any dimension.

Message routing in the D-dimensional toroid is very similar to that in spanning bus hypercubes. Instead of a single bus visit in each dimension that source and destination addresses differ, several moves along the ring in each dimension are required. As with the spanning bus hypercube, the order in which the coordinate differences are resolved does not matter.

Deriving a formula for the size of a node's reference locality requires a look at the nature of the interconnection. For the special case $w = 2$, Sullivan's CHoPP machine [Sull77], the analysis is similar to that of spanning bus hypercubes. To reduce the analysis' complexity, consider the case w odd ($w > 2$). Then without loss of generality, any node can be assumed to be at the center of the toroid. That is, the node is at the center of a D-1 dimensional hyperplane and $\left\lfloor \frac{w}{2} \right\rfloor$ hyperplanes of dimension D-1 are above it and below it. A message going up or down l links can then traverse at most $L-l$ links in the D-1 dimensional hyperplane it has reached. This leads to a fairly simple recurrence relation for the size of the reference locality. The results of its solution for the cases D=2,3 are shown in **Table IV**.

Cube-Connected-Cycles (CCC)

The cube-connected cycle (CCC) interconnection was recently proposed by Preparata and Vuillemin [PrVu81] as an efficient topology for several types of parallel algorithms. A CCC with D-dimensions contains $D2^D$ nodes arranged as cycles of D nodes around each of the 2^D vertices of a binary ($w = 2$) hypercube of D dimensions (see Figure I). The ith node of a cycle is connected to the ith dimension link incident upon the vertex. Each node is connected to exactly three other nodes no matter what the dimensionality of the system. Hence, fixed fanout nodes can be used to expand the system.

Our analysis is based on the simple, non-optimal, distributed message routing algorithm given by Wittie [Witt81]. The address of any node can be expressed as a cycle position followed by the binary coordinates of the cycle in D-space:

$$Cd_{D-1} \cdots d_0 \qquad 0 \le C \le D-1 \quad 0 \le d_i \le 1$$

To route a message toward its destination, traverse cycle-links in the clockwise direction until a d_i in the destination address is found that differs from the current address. Traverse that cross-link to another vertex. Repeat this process until the correct position in D-space has been reached. Then find the shortest distance, clockwise or counterclockwise, to the correct cycle position of the destination.

Obviously, this routing algorithm is far from optimal, and it would seem that performance could be increased significantly by improving it. The average number of cross-link traversals cannot be reduced except by altering the message routing distribution so any improvement must come from reducing the number of cycle-link traversals. It can be shown that, asymptotically, the cycle-link visit ratios are only 1.25 those of the cross-links, but for all dimensions of practical interest (say, $D \le 15$) the performance increase obtainable from a better routing algorithm could be significant.

Since cross-link traversals move one to a node with the same cycle position at another vertex, finding the shortest path from any source to any destination in a cube-connected cycle is equivalent to solving the following optimization problem:

(1) Consider a ring of K nodes

(2) Distinguish a start node, end node, and k intermediate nodes $(0 \leq k \leq K - 2)$

(3) Find the shortest path from the start node to the end node that passes through all the intermediate nodes

While it is also possible to derive formulas for the cube-connected cycles under local message routing, the formulas are quite unwieldy. Details of this derivation can be found in [Reed82].

Chordal Rings

Arden and Lee [ArLe81] proposed a variation of the simple bi-directional ring called a chordal ring. Each node of a ring is augmented with an additional connection to a link joining two ring nodes via a chord.. To be precise, number the nodes 0,...,K-1 where K is even and select an odd chord length c $(1 \leq c \leq \frac{K}{2})$. Then each odd numbered node i is connected to node $(i + c) \; mod \; K$ and each even numbered node j is connected to node $(j - c) \; mod \; K$ in addition to the normal ring connections.

The distributed routing algorithm presented by Arden and Lee finds a minimum path from any source to any destination using both cycle links and chord links. It does not employ all shortest paths with equal probability but tries to evenly distribute link traversals between the two types of links. An analysis of this routing algorithm is given in **Appendix A**. Unlike the simple ring, which has a constant performance bound, the performance bound for the chordal ring can be increased by increasing the chord length as the number of nodes becomes larger.

Asymmetric Structures

All of the asymmetric structures discussed below have constant performance bounds. That is, if one fixes all parameters of the system except the number of nodes and examines the upper bound on the system message completion rate as the number of nodes approaches infinity, the upper bound approaches a constant independent of the number of nodes. This would seem to indicate the fundamental unsuitability of asymmetric interconnections for very large parallel asynchronous computations unless communication is constrained to have very high locality.

Snowflake

Finkel and Solomon [FiSo80] describe a class of asymmetric structures they call snowflakes **(see Figure I)**. A snowflake of n levels is recursively constructed as follows:

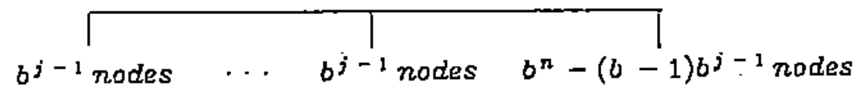
(1) A level one snowflake is composed of b nodes connected to a bus. Each of these nodes is called a corner of the snowflake.

(2) A level two snowflake connects one corner of b level one snowflakes to a new bus. Another corner of each level one snowflake is designated a corner of the level two snowflake.

(3) In general, a level n snowflake connects the corners of b level $n-1$ snowflakes to a new bus.

There are b^n nodes, $\frac{b^n - 1}{b - 1}$ buses, and $2b^n$ connections if one assumes all nodes are standard modules with a fixed number of connections. Since there is a unique path from every source to every destination, the message routing algorithm is straightforward and is detailed in [FiSo80].

To derive the link visit ratios for uniform message routing, consider the bus at level j



$b-1$ of the connections are to level $j-1$, but one connects to the bth level $j-1$ snowflake *and* the rest of the structure. Now consider some interval during which each node sends a message to each of the other $b^n - 1$ nodes. The source and destination can be in one of two places:

(1) Two level $j-1$ snowflakes

There are $2b^{2(j-1)}$ such messages. Since there are $\binom{b-1}{2}$ ways to choose a pair of level $j-1$ snowflakes,

$$2\binom{b-1}{2}b^{2(j-1)}$$

messages cross the level j bus due to messages between level $j-1$ snowflakes.

(2) Level $j-1$ snowflake and $b^n - (b-1)b^{j-1}$ group

By an argument similar to the one above, there are

$$2b^{j-1}(b-1)(b^n - (b-1)b^{j-1})$$

messages contributed by these combinations.

Then the *VS* for the level j bus is

$$V_{CLj}S_{CLj} = \frac{S_{CL}b^{j-1}(b-1)(2b^n - b^j)}{b^n(b^n-1)}$$

This clearly attains its maximum when $j=n$. Hence, the system message completion rate is bounded by

$$X_0 \le \min\left\{\frac{b^n}{S_{PE}},\ \frac{b^n-1}{S_{CL}b^{n-1}(b-1)}\right\}$$

As the number of levels becomes large, the system throughput rate approaches $\frac{b}{S_{CL}(b - 1)}$. By way of comparison, the performance asymptote for a single bus system is $\frac{1}{S_{CL}}$. Notice that $b = 2$ maximizes the performance bound. In other words, a snowflake with many levels and a small branching factor b is preferable to one with a smaller number of levels and a larger branching factor.

Dense Snowflake

The dense snowflake attempts to alleviate the communication bottleneck of the snowflake by replacing the single bus at each level with $b-1$ buses. As with the snowflake, a simple distributed routing algorithm is presented by Finkel and Solomon [FiSo80]. As shown in **Table III**, the additional message paths result in a significant performance improvement over the snowflake. Interestingly, the performance of a dense snowflake is maximized by having a larger branching factor and a smaller number of levels, the opposite of the snowflake.

Star

Instead of connecting the sublevels of a snowflake by their corners, they can be connected by their centers to form a star as follows:

(1) A level one substar has $b-1$ nodes connected to a single bus.

(2) A level two substar introduces an additional bus with $b-1$ nodes attached. Each of these nodes is attached to the empty slot on the bus of a different level one substar.

(3) In general, a level j substar introduces a new bus with $b-1$ nodes. Each of these is connected to a slot on the central bus of a different level $j-1$ substar.

(4) Finally, a new bus with b nodes is used to connect b level $n-1$ substars to form a level n star.

Finkel and Solomon [FiSo80] also present a distributed message routing algorithm for this structure. As can also be seen in **Table III**, the star has no better asymptotic performance that the snowflake.

Trees

The best known asymmetric interconnection is undoubtedly the n-level b-ary tree. Message routing is simple since there is a unique path from any source to any destination. Unfortunately, the b communication links below the root rapidly become the performance bottlenecks. Like the dense snowflake, trees with a larger branching factor and smaller number of levels give better performance than trees with a small branching factor and more levels.

Applications

There is no single "best" system; depending on the intended application, one system may be preferred over another. By specifying a subset of the system parameters (e.g., cost, number of nodes, or performance), one can determine the optimal values of the remaining parameters.

The following are but a few of the many possibilities:

(1) Given a desired performance level, determine the minimum number of nodes and type of interconnection necessary to attain it.

(2) Given a system cost, determine the maximum performance attainable using any of the systems we have discussed.

(3) Given two different systems with the same number of nodes, determine the ratio of S_{PE} to S_{CL} needed to equalize performance.

As an extended example of the power of this technique, consider the spanning bus hypercube discussed earlier. Under uniform routing we have

$$V_{PE}S_{PE} = \frac{S_{PE}}{w^D} \qquad V_{CL}S_{CL} = \frac{S_{CL}(w-1)}{w^D - 1}$$

Recall that

$$X_0 \le \min\left\{\frac{1}{V_{PE}S_{PE}}, \frac{1}{V_{CL}S_{CL}}\right\}$$

Suppose we equate $V_{PE}S_{PE}$ and $V_{CL}S_{CL}$ and solve for the ratio of processing element to link service times:

$$\frac{S_{PE}}{S_{CL}} = \frac{w^D(w-1)}{w^D - 1}$$

At this critical ratio, the communication links and the processing elements are equally the performance bottlenecks. If the ratio falls below this value, the communication links determine the upper bound on the system performance.

Now suppose the number of nodes is increased by increasing w, the width of the spanning bus hypercube. For the bound on the system message completion rate to increase linearly with the number of nodes, the ratio of processing element to communication link service times must increase linearly with w. In other words, as the number of nodes in the system becomes larger and larger, nodes must exchange messages less frequently if performance is to increase linearly with the number of nodes.

Under locality, we have

$$V_{PE}S_{PE} = \frac{S_{PE}}{w^D} \qquad V_{CL}S_{CL} = \frac{S_{CL}LV_{SBH}^{local}}{Dw^{D-1}}$$

and

$$\frac{S_{PE}}{S_{CL}} = \frac{wLV_{SBH}^{local}}{D}$$

where the mean number of link visits by a message LV_{SBH}^{local} was defined earlier when discussing the spanning bus hypercube.

If the size of the locality and the probability of visiting it remain constant, then the nodes must exchange messages with less frequency as the number of nodes becomes larger if the performance bound is to increase linearly with the number of nodes. Conversely, if the node and link service times remain constant, the probability of a message visiting a node in the locality must increase as the number of system nodes increases if the performance bound is to increase linearly.

This phenomenon is not unique to the spanning bus hypercube. In general, as the number of nodes increases, the ratio of computation time to communication time must increase or the locality of communication must increase if the performance bound is to increase linearly with the number of network nodes. The technique we have discussed permits us to quantify these relationships (i.e., determine the amount of locality needed or the minimum computation time - communication time ratio).

Comparisons

A look at **Table III** shows the following:

(1) Performance of the D-dimensional toroid is four times that of the spanning bus hypercube with the same number of nodes. The smaller number of link traversals required by a message in the spanning bus hypercube is more than offset by the additional number of links in the toroid.

(2) Neglecting the complete connection, only the spanning bus hypercube, D-dimensional toroid, and the cube-connected cycle have non-constant performance bounds if all parameters are fixed and the number of nodes is made very large.

(3) The cube-connected cycle has, asymptotically, the best performance of any interconnection. In fact, its performance bound differs from that of the binary hypercube with D dimensions by only the factor D. Unfor-

tunately, lower order terms in the performance bound prevent the cube-connected cycle's performance from exceeding that of the 3-D toroid until the number of nodes exceeds 500,000 (if the processing element and link service times are equal).

(4) Of the asymmetric structures, the dense snowflake gives the best performance.

Table IV shows that asymptotically, our definition of locality changes only the constant of proportionality not the order of the system performance bound. As long as there exists any non-zero probability of a message traversing a distance proportional to the size of the structure, this must, in the limit, bound the system performance.

Finally, **Figures II-IX** show some representative instances of these cost and performance bounds. The unit cost of nodes, connections, and links is assumed to be unity and the processing element and link service times are also assumed to be unity. These curves are but a few of an entire family of such curves obtainable by varying the cost, service times, or locality.

The ratio of performance to cost obviously depends on the values specified for $S_{PE,}$ S_{CL}, locality, and the unit cost of the nodes and their connections. For all of the interconnection networks we have discussed here, the performance bound increases at most linearly with the number of nodes. Cost, on the other hand, increases at least linearly with the number of nodes. Hence, the performance/cost ratio approaches either some constant or zero. This is evident in **Figures VI,VII,** and **IX**.

Conclusions

We have described a method for determining cost and performance bounds for a distributed message passing system. We introduced the notion of a message routing distribution and showed how it could be used to derive performance bounds under more realistic assumptions than uniform message routing. Finally, we applied the technique to several proposed interconnection structures.

Several interesting areas remain to be investigated. The most obvious is the extension of the locality results to asymmetric structures. This is likely to be more difficult since locality in asymmetric structures invalidates the assumption that all nodes are visited with equal probability. Second, the locality result for symmetric structures can easily be extended to include non-constant φ. One extended locality definition might make the probability of sending a message to a node l links away inversely proportional to l. Finally, performance and cost are not the only figures of merit for distributed systems. A weighted function of such things as cost, performance, reliability, broadcast delay, and expansion increments should provide a more precise method of selection.

References

[ArLe80] Arden, B.W.; Lee, H. "Analysis of a Chordal Ring Network", *Proc. Workshop on Interconnection Networks for Parallel and Distributed Processing*, H.J. Siegel, ed., Purdue University, April 1980

[Back78] Backus, J. "Can Programming Be Liberated from the von Neumann Style? A Functional Style and Its Algebra of Programs", *Comm. ACM*, (21,8), Aug. 1978, p. 613-641

[DeBu78] Denning, P.J.; Buzen, J.P. "The Operational Analysis of Queueing Network Models", *Comp. Surveys*, (10,3), Sept. 1978, p. 225-261

[FiSo80] Finkel, R.A.; Solomon, M.H. "Processor Interconnection Strategies", *IEEE Transactions on Computers*, Vol. C-29, May 1980, p. 360-371

[Goke73] Goke, L.R.; Lipovski, G.J. "Banyan Networks for Partitioning Multiprocessor Systems", *Proc. 1st Annual Symposium on Computer Architecture*, Dec. 1973, p. 21-28

[Hoar78] Hoare, C.A.R. "Communicating Sequential Processes", *Comm. ACM*, (21,8), Aug. 1978, p. 666-677

[Jafa78] Jafari, H.J.; Spragins, J.; Lewis, T.; "A New Modular Loop Architecture for Distributed Computer Systems", *Trends and Applications 78: Distributed Processing*, 1978, p. 72-77

[Lang76] Lang, T.; Stone H.S. "A Shuffle-exchange Network with Simplified Control", *IEEE Transactions on Computers*, Vol C-25, Jan. 1976, p. 55-66

[Liu78] Liu, M.T. "Distributed Loop Computer Networks", in *Advances in Computers*, Vol. 17, Academic Press, 1978, p. 163-221

[PrVu81] Preparata, F.P.; Vuillemin,J. "The Cube-Connected Cycles: A Versatile Network for Parallel Computation", *Comm. ACM*, (24,5), May 1981, p. 300-309

[Reed82] Reed, D. A. "Performance Models of Processor Interconnection Structures", *PhD dissertation, Department of Computer Sciences, Purdue University*, in preparation

[Sull77] Sullivan, H.; Bashkow, T.R. "A Large Scale, Homogeneous, Fully Distributed Parallel Machine", *Proc. 4th Annual Symposium on Computer Architecture*, 1977, p. 105-124

[Witt76] Wittie, L.D. "Efficient Message Routing in Mega-micro-computer Networks", *Proc. 3rd Annual Symposium on Computer Architecture*, Jan. 1976, p. 136-140

[Witt81] Wittie, L.D. "Communication Structures for Large Multimicrocomputer Systems", *IEEE Transactions on Computers*, Vol. C-30, Apr. 1981, p. 264-273

Appendix A

Chordal Ring Performance Bounds

Uniform Routing

Since the chordal ring is symmetric, one can, without loss of generality, assume that a message's source node is node 0 and the destination is some node i $(1 \le i \le K - 1)$. Arden and Lee [ArLe80] give formulas for the number of chord links $C(i)$ and ring links required to reach node i from node 0. Analysis of these formulas shows that for a fixed chord length c and increasing K, the number of ring link traversals required to reach all possible destination nodes is less than twice the number of chord link traversals needed. Because there are twice as many ring links as chord links, for large enough K, the chord links become the bottleneck.

From the formulas given by Arden and Lee, it is apparent that

$$\min\left(\left\lfloor\frac{i}{c+1}\right\rfloor, \left\lfloor\frac{K-i}{c+1}\right\rfloor\right) \le C(i) \le \min\left(\left\lceil\frac{i}{c+1}\right\rceil, \left\lceil\frac{K-i}{c+1}\right\rceil\right)$$

Furthermore, the ceiling case occurs much more frequently than the floor case.

An upper bound on the mean number of chord traversals required is then

$$UpperBound = \frac{\sum_{i=1}^{K-1} \min\left(\left\lceil\frac{i}{c+1}\right\rceil, \left\lceil\frac{K-i}{c+1}\right\rceil\right)}{K-1}$$
$$= \frac{(c+1)\left\lfloor\frac{K}{2(c+1)}\right\rfloor\left(\frac{K}{2(c+1)} - 2\left\lceil\frac{K}{2(c+1)}\right\rceil + 1\right)}{K-1} + \left\lceil\frac{K}{2(c+1)}\right\rceil$$

Since there are $\frac{K}{2}$ chord links, we have

$$V_{CL} S_{CL} \le \frac{2 S_{CL}\, UpperBound}{K}$$

Similarly, a lower bound on the mean number of chord link traversals is

$$LowerBound = \frac{\sum_{i=1}^{K-1} \min\left(\frac{i}{c+1}, \frac{K-i}{c+1}\right)}{K-1} = \frac{K^2}{4(c+1)(K-1)}$$

and

$$V_{CL}S_{CL} \geq \frac{2S_{CL}LowerBound}{K}$$

Both the lower and upper bound are asymptoticly exact and converge to a performance bound of $\frac{2(c+1)}{S_{CL}}$ as K becomes large. Using these upper and lower bounds, one can trade accuracy with computational cost on an almost continuous spectrum by calculating the exact visit ratios until the difference between them and the estimated visit ratios falls below some desired error tolerance. Thereafter, the approximation may be employed.

Locality

Unfortunately, we know of no closed form for the link visit ratios under locality. By exhaustively enumerating the K-1 message destinations from node 0, they can be calculated in $O(K)$ time.

Table I

Notation

b	Branching factor for asymmetric structures
c	Chord length
D	Dimension of mesh or hypercube
K	Number of network nodes
L	Maximum distance to a node in the locality
$lmax$	Maximum source-destination distance
n	Number of levels in an asymmetric structure
w	Lattice width of mesh or hypercube
φ	Probability of visiting locality
PE	Processing element
LC	Communication link connection
CL	Communication link
S_{PE}	Mean processing element service time
S_{CL}	Mean communication link service time
V_{PE}	Processing element visit ratio
V_{CL}	Communication link visit ratio
X_0	System message completion rate
$LocSize(L, Net-type)$	Size of locality
$LV^{uniform}_{symmetric}$	Average number of links traversed in a symmetric structure with uniform routing
$LV^{local}_{symmetric}$	Average number of links traversed in a symmetric structure with locality
$LV^{uniform}_{asymmetric}$	Average number of links traversed in an asymmetric structure with uniform routing
$NumLinks(K, Net-type)$	Number of communication links in network of size K
$Reach(l, Net-type)$	Number of nodes reachable by traversing l links

Table II

System Size			
System	**Nodes**	**Connections**	**Links**
Single Global Bus	K	K	1
Complete Connection	K	$K(K-1)$	$\frac{K(K-1)}{2}$
Double Ring	K	$4K$	$2K$
Spanning Bus Hypercube	w^D	Dw^D	Dw^{D-1}
D-dimensional Toroid	w^D	$2Dw^D$	Dw^D
Cube-Connected Cycle	$D2^D$	$3D2^D$	$3D2^{D-1}$
Chordal Ring	K	$3K$	$\frac{3K}{2}$
Snowflake	b^n	$2b^n$	$\frac{b^n-1}{b-1}$
Dense Snowflake	b^n	$2b^n$	$2b^n-1$
Star	$\frac{b((b-1)^n-1)}{b-2}$	$\frac{2b((b-1)^n-1)}{b-2}$	$\frac{b(b-1)^{n-1}-2}{b-2}$
Tree	$\frac{b^n-1}{b-1}$	$\frac{(b+1)(b^n-1)}{b-1}$	$\frac{b^n-b}{b-1}$

$$
\begin{aligned}
&Cost\,(Net-type\,, Net-size\,, C_{PE}, C_{LC}, C_{CL}) = \\
&\quad C_{PE} * Nodes + \\
&\quad P_{LC} * Connections + \\
&\quad C_{CL} * Links
\end{aligned}
$$

where the following definitions apply

$Net-type$	type of interconnection structure
C_{PE}	unit cost of a node
C_{LC}	unit cost of a link connection
C_{CL}	unit cost of a communication link

Table III

Performance Bounds - Uniform Message Routing			
System	X_0 *Asymptote*	$V_{PE}S_{PE}$	$V_{CL}^{Max}S_{CL}$
Single Global Bus	$\frac{1}{S_{CL}}$	$\frac{S_{PE}}{K}$	S_{CL}
Complete Connection	$\frac{K}{S_{PE}}$	$\frac{S_{PE}}{K}$	$\frac{2S_{CL}}{K(K-1)}$
Double Ring	$\frac{8}{S_{CL}}$	$\frac{S_{PE}}{K}$	*K even* $\frac{KS_{CL}}{8(K-1)}$
			K odd $\frac{S_{CL}(K+1)}{8K}$
Spanning Bus Hypercube	$\frac{w^{D-1}}{S_{CL}}$	$\frac{S_{PE}}{w^D}$	$\frac{S_{CL}(w-1)}{w^D-1}$
D-dimensional Toroid	$\frac{4w^{D-1}}{S_{CL}}$	$\frac{S_{PE}}{w^D}$	*w even* $\frac{S_{CL}w}{4(w^D-1)}$
			w odd $\frac{S_{CL}(w^2-1)}{4w(w^D-1)}$
Cube-Connected-Cycle	$\frac{2^{D+2}}{5S_{CL}}$	$\frac{S_{PE}}{D2^D}$	*Cross* $\frac{S_{CL}D}{D2^D-1}$
			Cycle D odd $\frac{S_{CL}2^D(5D^2-6D-1)+8D}{4D2^D(D2^D-1)}$
			Cycle D even $\frac{S_{CL}2^D(6D-8)+8}{42^D(D2^D-1)}$

$$X_0 \leq \min\left\{\frac{1}{V_{PE}S_{PE}}, \frac{1}{V_{CL}^{Max}S_{CL}}\right\}$$

Note: The X_0 asymptote is the limit on performance as the number of nodes becomes very large. For the single global bus and double ring it is the absolute upper bound on system performance as the number of nodes becomes infinite. For the other systems, it is the dominant term of the performance bound.

Table III Continued

Performance Bounds - Uniform Message Routing			
System	X_0 *Asymptote*	$V_{PE}S_{PE}$	$V_{CL}^{Max}S_{CL}$
Chordal Ring	$\frac{2(c+1)}{S_{CL}}$	$\frac{S_{PE}}{K}$	See Appendix A
Snowflake	$\frac{b}{S_{CL}(b-1)}$	$\frac{S_{PE}}{b^n}$	$\frac{S_{CL}(b-1)b^{n-1}}{b^n-1}$
Dense Snowflake	$\frac{b}{S_{CL}}$	$\frac{S_{PE}}{b^n}$	$\frac{S_{CL}b^{n-1}}{b^n-1}$
Star	$\frac{b}{S_{CL}(b-1)}$	$\frac{S_{PE}(b-2)}{b\left[(b-1)^n-1\right]}$	$\frac{S_{CL}(b-1)\left[(b-1)^n-1\right]}{b\left[(b-1)^n-1\right]-b+2}$
Tree	$\frac{b}{2S_{CL}}$	$\frac{S_{PE}(b-1)}{b^n-1}$	$\frac{2S_{CL}b^{n-2}(b-1)}{b^n-1}$

$$X_0 \leq \min\left\{\frac{1}{V_{PE}S_{PE}},\ \frac{1}{V_{CL}^{Max}S_{CL}}\right\}$$

Note: The X_0 asymptote is the limit on performance as the number of nodes becomes very large. For the chordal ring, snowflake, dense snowflake, star, and tree it is the absolute upper bound on system performance as the number of nodes becomes infinite.

Table IV

Selected Performance Bounds - Local Message Routing

Double Ring

$$V_{CL}S_{CL} = \begin{cases} S_{CL}\left[\dfrac{\varphi(L+1)}{4K} + \dfrac{(1-\varphi)(K^2-1-4L(L+1))}{8K(K-2L-1)}\right] & K \text{ odd} \\ S_{CL}\left[\dfrac{\varphi(L+1)}{4K} + \dfrac{(1-\varphi)(K^2-4L(L+1))}{8K(K-2L-1)}\right] & K \text{ even} \end{cases}$$

$$X_0 \textit{ Asymptote is } \frac{8}{S_{CL}(1-\varphi)}$$

Spanning Bus Hypercube

$$V_{CL}S_{CL} = \left[\frac{S_{CL}}{Dw^{D-1}}\right]\left(\frac{\varphi \sum_{l=1}^{L} l \binom{D}{l}(w-1)^l}{\sum_{l=1}^{L}\binom{D}{l}(w-1)^l} + \frac{(1-\varphi)\left[Dw^{D-1}(w-1) - \sum_{l=1}^{L} l \binom{D}{l}(w-1)^l\right]}{w^D - \sum_{l=1}^{L}\binom{D}{l}(w-1)^l - 1}\right)$$

$$X_0 \textit{ Asymptote is } \frac{w^{D-1}}{S_{CL}(1-\varphi)}$$

2-Dimensional Toroid (w Odd, $L \le \left\lfloor \frac{w}{2} \right\rfloor$)

$$V_{CL}S_{CL} = \left[\frac{S_{CL}}{2w^2}\right]\left[\frac{\varphi(2L+1)}{3} + \frac{(1-\varphi)\left[w(w^2-1) - 4L(L+1)\right]}{2(w^2-2L(L+1)-1)}\right]$$

$$X_0 \textit{ Asymptote is } \frac{4w}{S_{CL}(1-\varphi)}$$

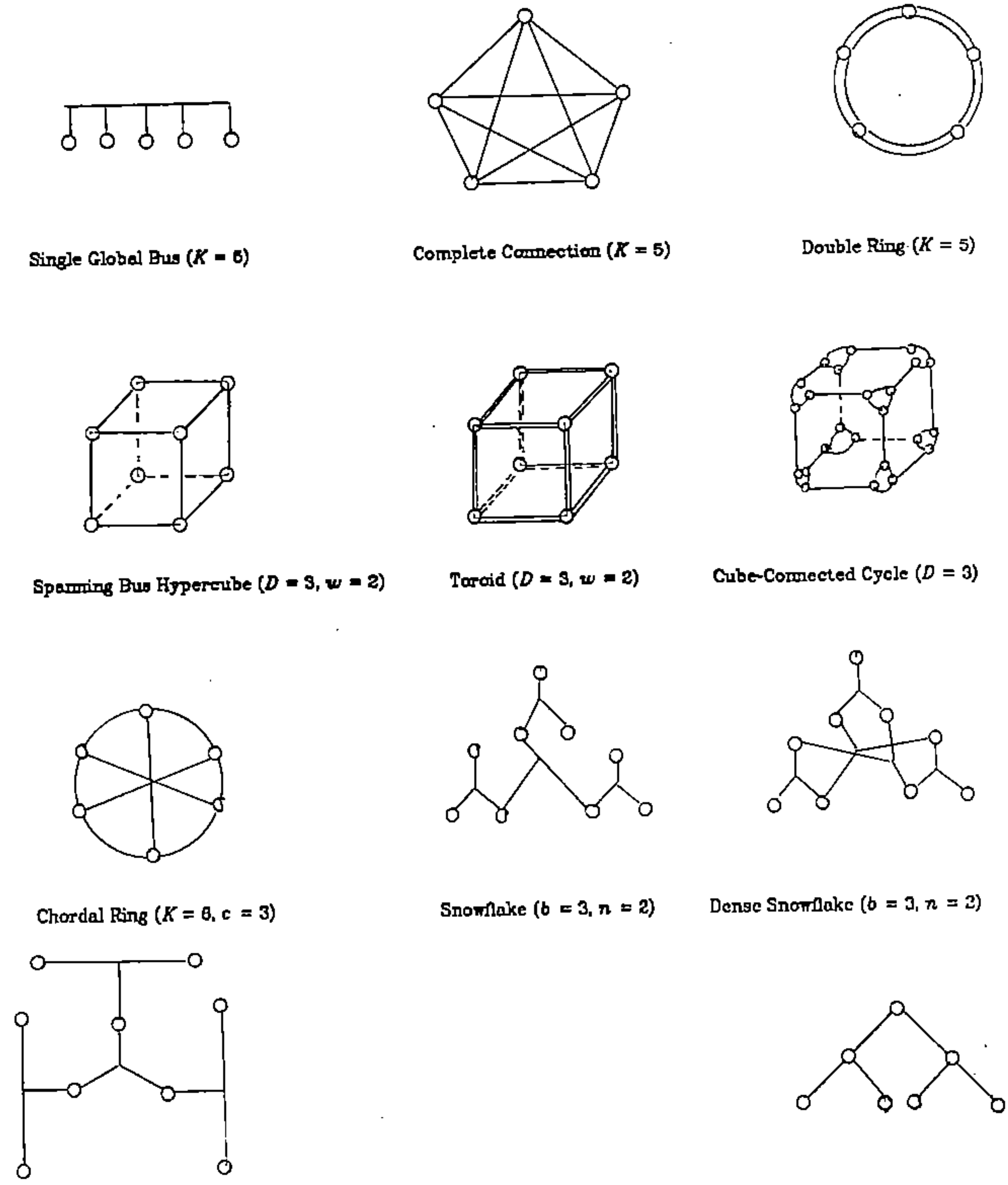
3-Dimensional Toroid (w Odd, $L \le \left\lfloor \frac{w}{2} \right\rfloor$)

$$V_{CL}S_{CL} = \left[\frac{S_{CL}}{3w^3}\right]\left[\frac{3\varphi(L+1)(L^2+L+1)}{2(L+1)(2L+1)+6} + \frac{3(1-\varphi)\left[3w^2(w^2-1) - 4L(L+1)(L^2+L+1)\right]}{4(3w^3-2L(L+1)(2L+1)-6L-3)}\right]$$

$$X_0 \textit{ Asymptote is } \frac{4w^2}{S_{CL}(1-\varphi)}$$

The values of $V_{PE}S_{PE}$ are the same as for the uniform message routing case.

Figure I



Single Global Bus ($K = 6$)

Complete Connection ($K = 5$)

Double Ring ($K = 5$)

Spanning Bus Hypercube ($D = 3, w = 2$)

Toroid ($D = 3, w = 2$)

Cube-Connected Cycle ($D = 3$)

Chordal Ring ($K = 6, c = 3$)

Snowflake ($b = 3, n = 2$)

Dense Snowflake ($b = 3, n = 2$)

Star ($b = 3, n = 2$)

Tree ($b = 2, n = 3$)

FIGURE II - UNIFORM ROUTING
(UNIT SERVICE TIMES)

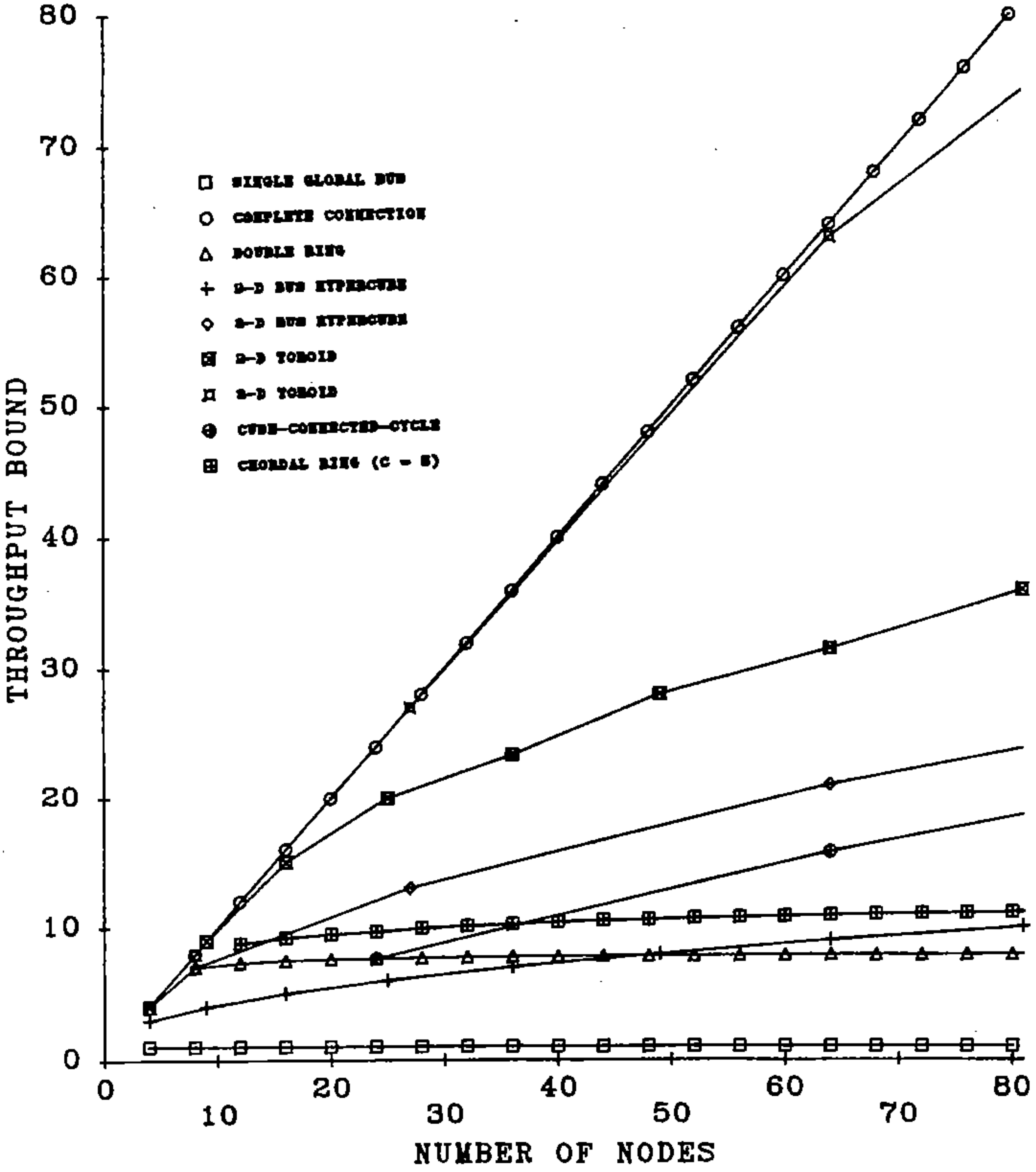


FIGURE III - UNIFORM ROUTING (UNIT SERVICE TIMES)

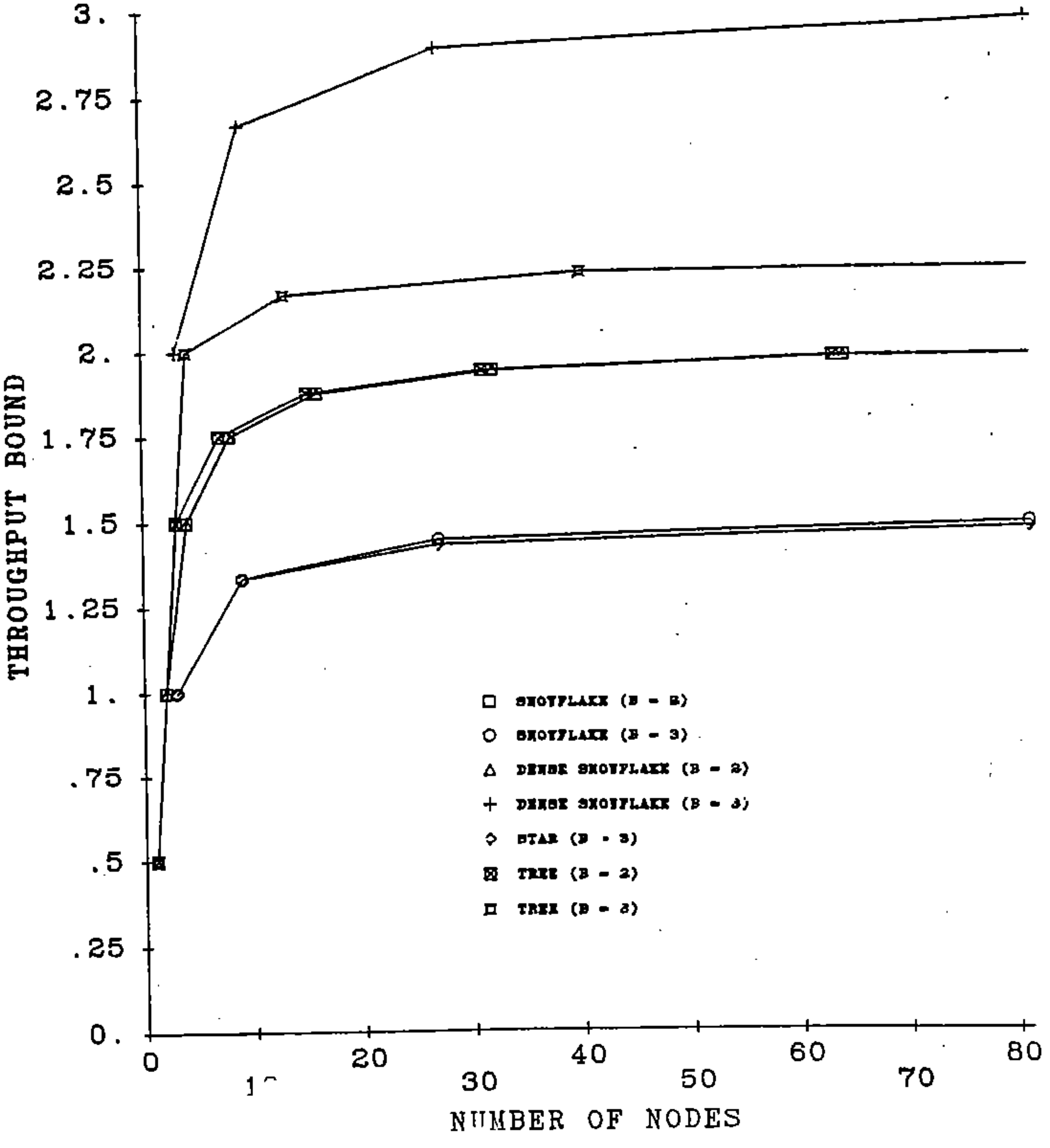


FIGURE IV - SYSTEM COST
(UNIT COMPONENT COSTS)

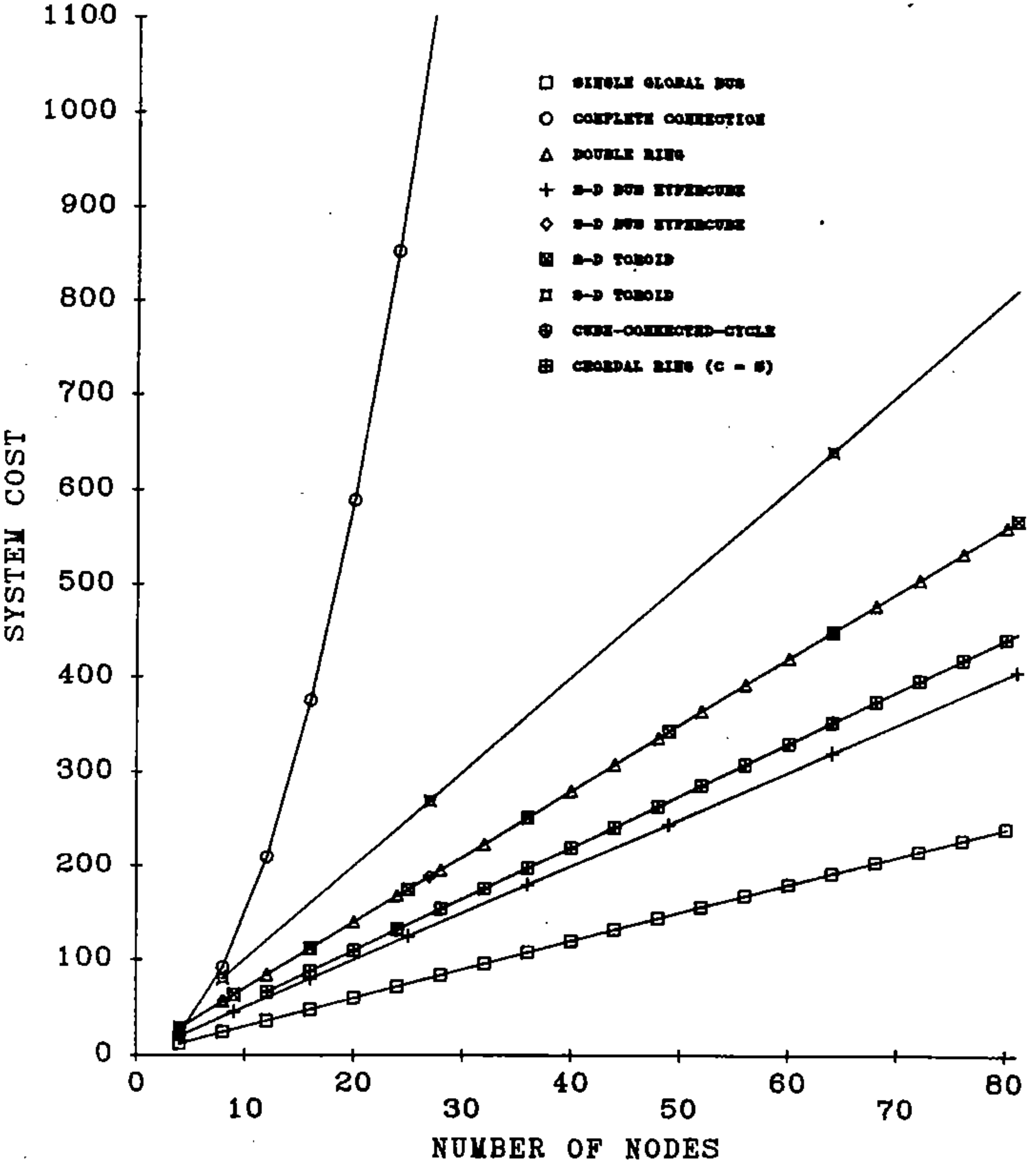


FIGURE V - SYSTEM COST (UNIT COMPONENT COSTS)

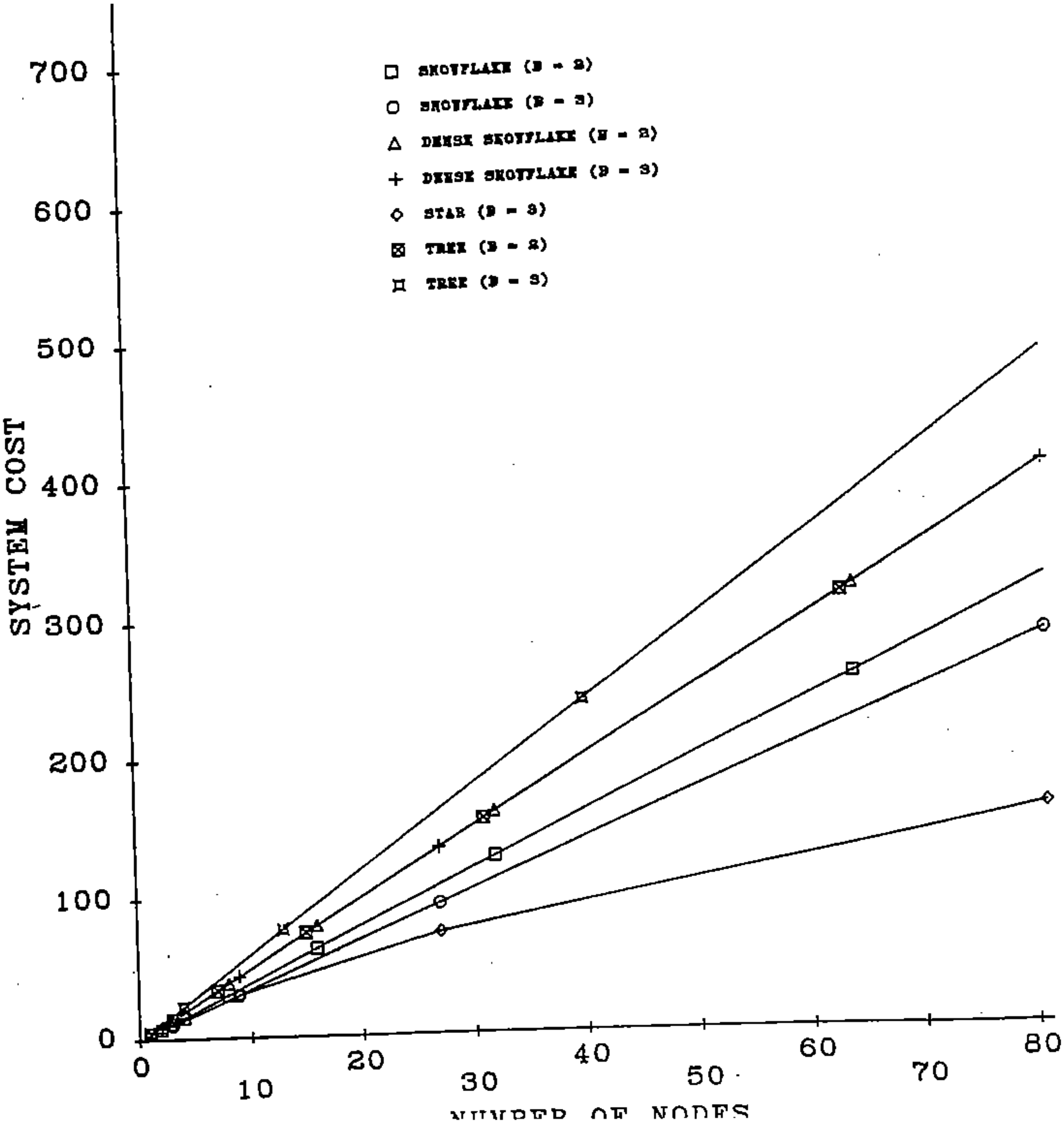


FIGURE VI - UNIFORM ROUTING (UNIT SERVICE TIMES)

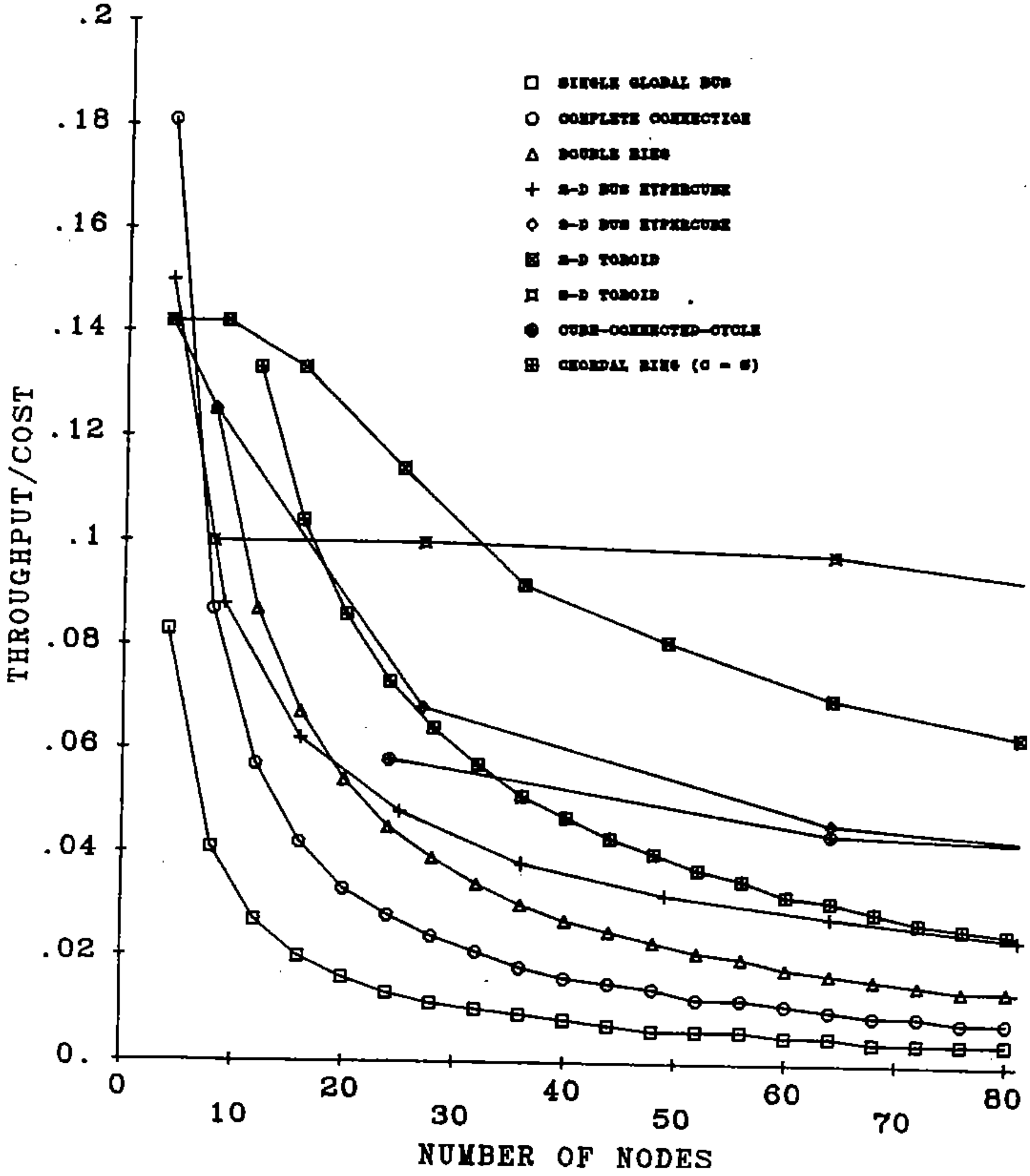


FIGURE VII - UNIFORM ROUTING (UNIT SERVICE TIMES)

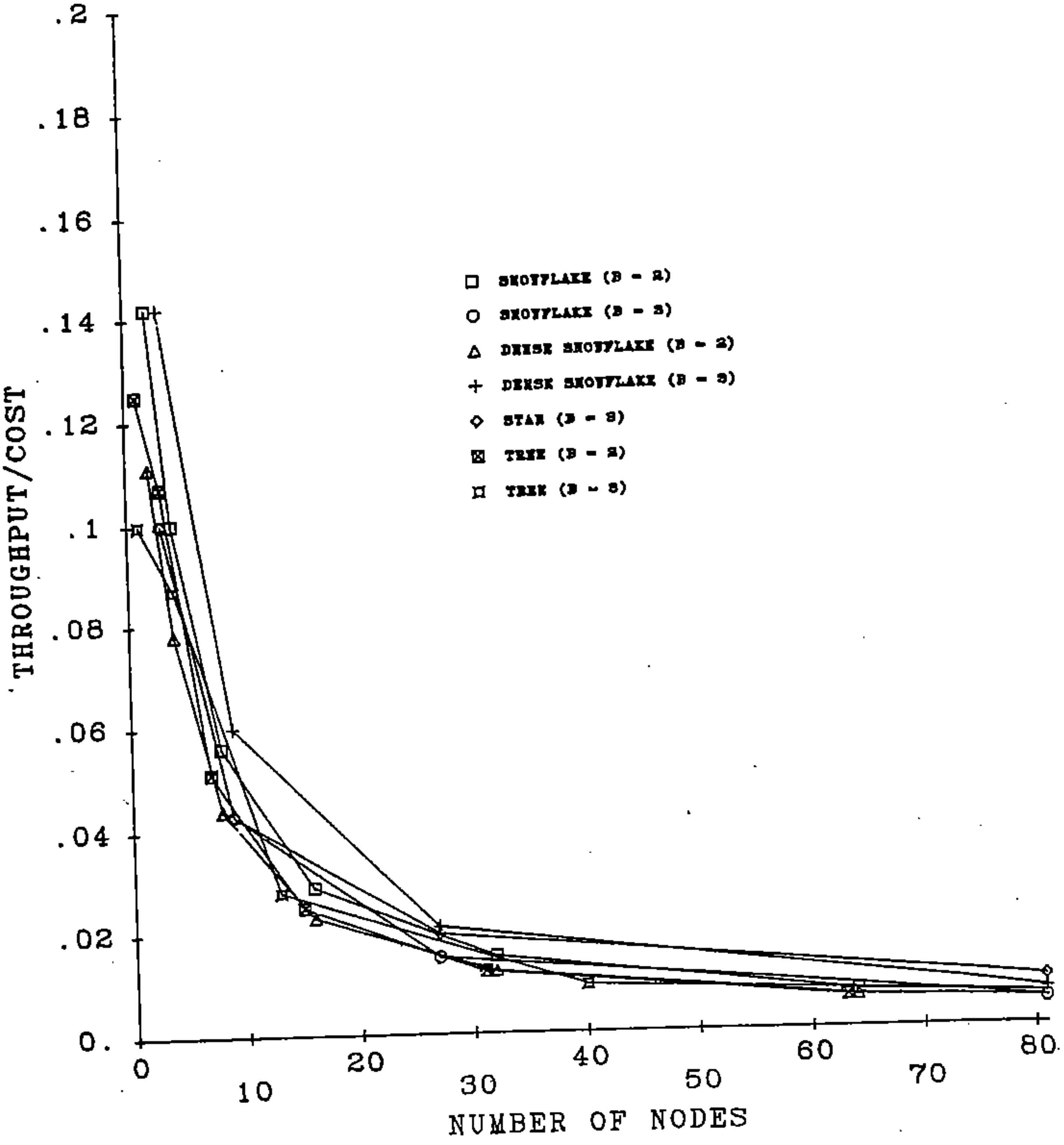


FIGURE VIII - LOCALITY (PHI = 0.5, L = 1)

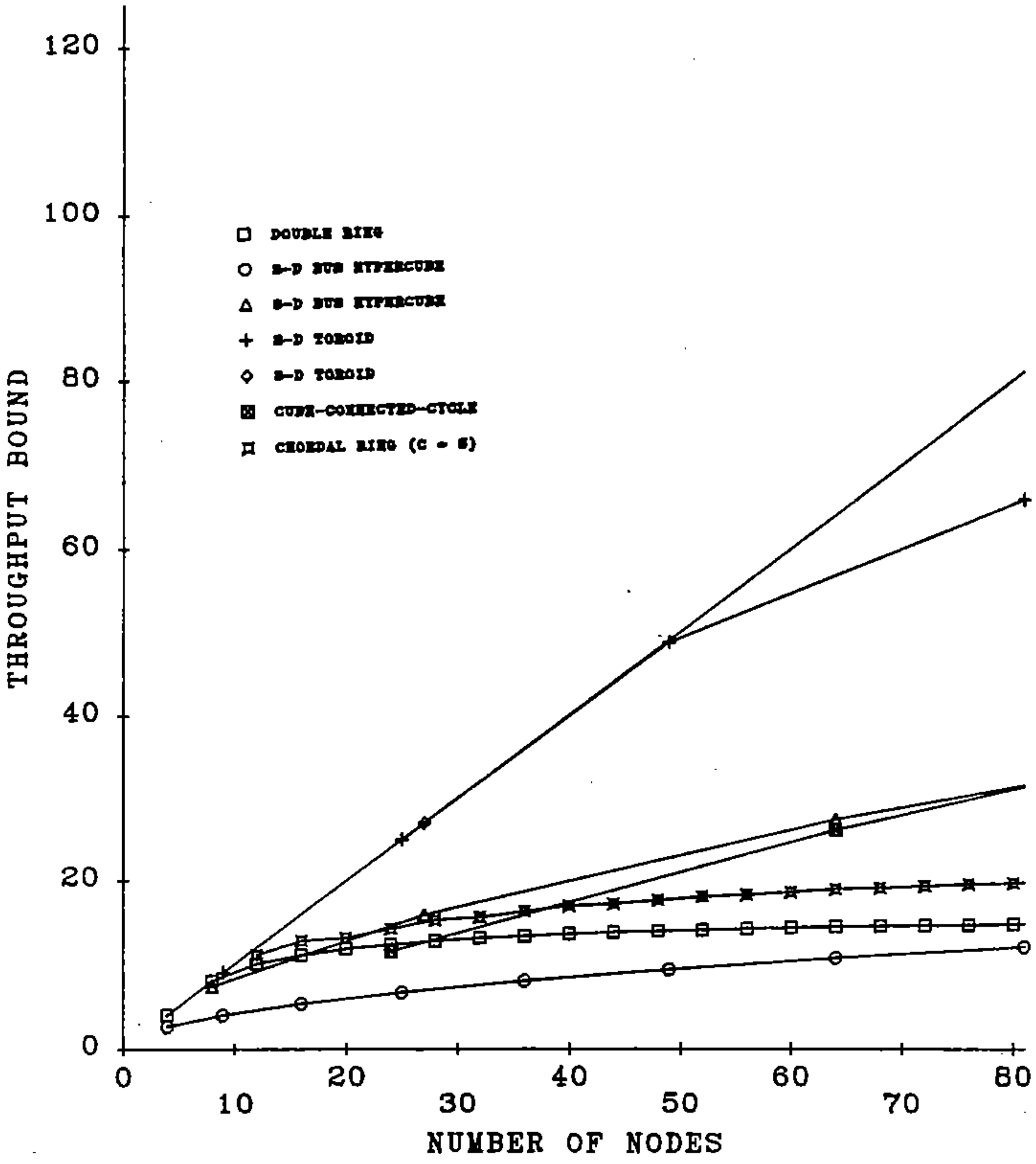
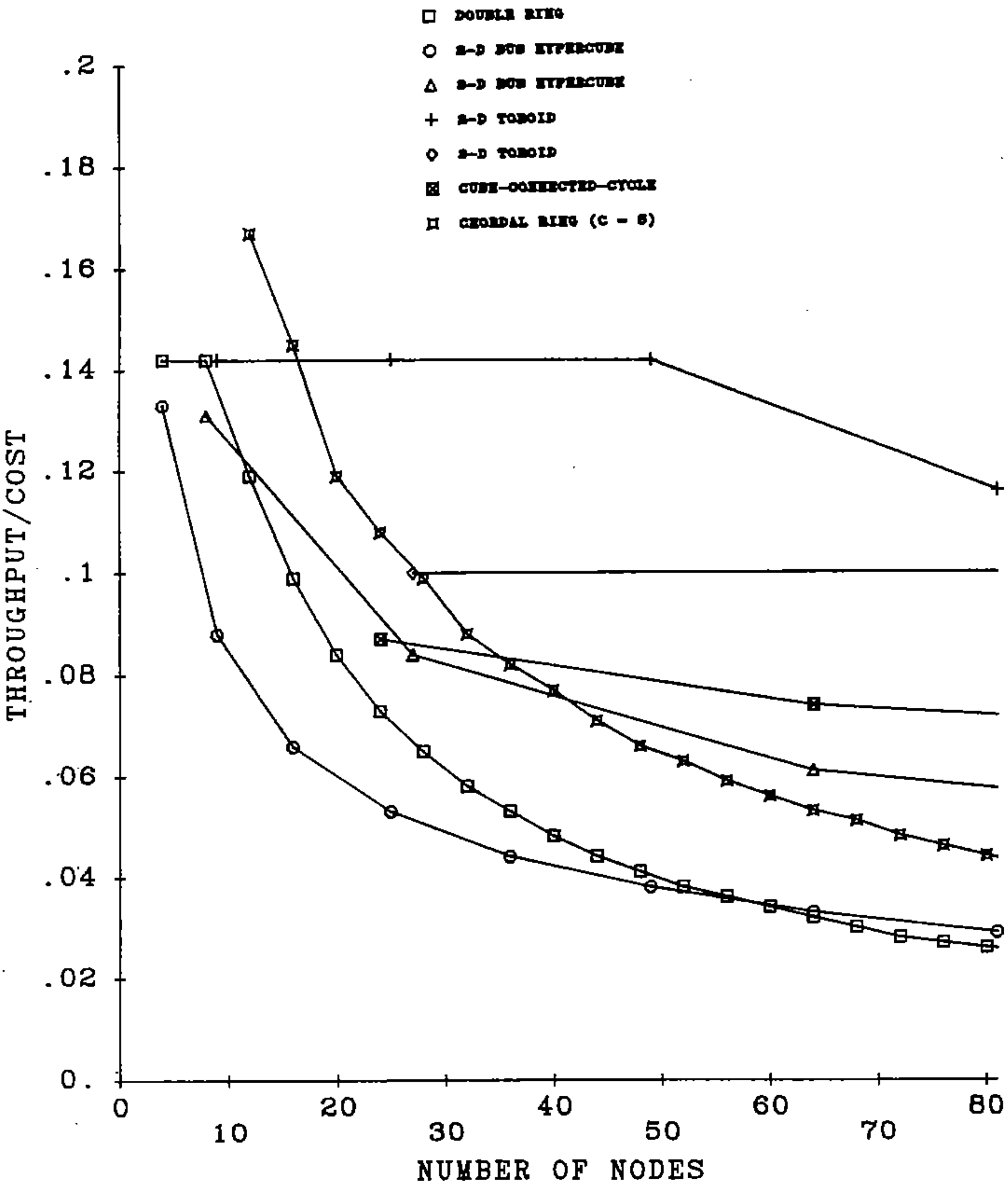


FIGURE IX - LOCALITY (PHI = 0.5, L = 1)


□ DOUBLE RING
○ 2-D BUS HYPERCUBE
△ 3-D BUS HYPERCUBE
+ 2-D TOROID
◇ 3-D TOROID
⊠ CUBE-CONNECTED-CYCLE
⌑ CHORDAL RING (C = 6)
THROUGHPUT/COST
NUMBER OF NODES