Multinode Broadcast in Hypercubes and Rings with Randomly Distributed Length of Packets

Emmanouel A. Varvarigos and Dimitri P. Bertsekas, *Fellow, IEEE*

***Abstract*— We consider a multinode broadcast (MNB) in a hypercube and in a ring network of processors. This is the communication task where we want each node of the network to broadcast a packet to all the other nodes. The communication model that we use is different than those considered in the literature so far. In particular, we assume that the lengths of the packets that are broadcast are not fixed, but are distributed according to some probabilistic rule, and we compare the optimal times required to execute the MNB for variable and for fixed packet lengths. For large hypercubes we show under very general probabilistic assumptions on the packet lengths, that the MNB is completed in essentially the same time as when the packet lengths are fixed. In particular, we show that the MNB is completed by time $(1+\delta)T_s$ with probability at least $1-\epsilon$, for any positive ϵ and δ, where T_s is the optimal time required to execute the MNB when the packet lengths are fixed at their mean, provided that the size of the hypercube is large enough. In the case of the ring we prove that the average time required to execute a MNB when the packet lengths are exponentially distributed exceeds by a factor of ln n the corresponding time for the case where the packet lengths are fixed at their mean, where n is the number of nodes of the ring.**

***Index Terms*—Hypercube, multinode broadcast, random packet lengths.**

I. Introduction

The processors of a multiprocessor system, when doing computations, often have to communicate intermediate results. The interprocessor communication time may be substantial relative to the time needed exclusively for computations, so it is important to carry out the information exchange as efficiently as possible.

One of the most frequent communication tasks is the *multinode broadcast* (MNB). In this task we want each node to broadcast a packet (the same packet) to all the other nodes. The MNB arises, for example, in interations of the form $x_{t+1} = Ax_t$, where A is a square matrix and x_t, x_{t+1} are column vectors of appropriate dimensions. In this computation, each processor computes a specific component of the vector x_{t+1} and broadcasts it to all the other processors so that it can be used during the next iteration.

Algorithms for routing messages between different processors have been studied by several authors under a variety of assumptions on the communication network connecting the processors. Saad and Shultz [14], [15] have introduced a number of generic communication problems that arise frequently in numerical and other methods. They have assumed that all packets take unit time to traverse any communication link. Johnson and Ho [9] have developed minimum and nearly minimum completion time algorithms for similar routing problems as those of Saad and Schultz but used a different communication model and a hypercube network. Their model quantifies the effects of setup time (or overhead) per packet, while it allows packets to have different lengths, and to be split and be recombined prior to transmission on any link in order to save on setup time. In the model of [9], each packet may consist of data originating at different nodes and/or destined for different nodes and the extra overhead for splitting and combining packets is considered negligible. Bertsekas *et al.* [2], and Bertsekas and Tsitsiklis [1] have used the communication model of Saad and Shultz to derive minimum completion time algorithms for several communication problems in a hypercube. In particular, they have given an algorithm for the multinode broadcast that executes in a minimum number of steps ($\lceil (n-1)/d \rceil$ for a hypercube with $n = 2^d$ processors). Several other works deal with various communication problems and network architectures related to those discussed in the present paper; see [3]–[5], [7], [8], [10]–[12], [16]–[20], and [21].

In this paper we will be dealing with a multinode broadcast in hypercubes and rings. The case where the packets broadcast by the processors have equal deterministic lengths has been studied in the literature for a variety of regular topologies. Algorithms that execute the MNB in optimal time exist for the case of the ring ([1]), the d-dimensional wraparound mesh ([12], [1]), the hypercube ([14], [15], [12], [1], [2], [9]), and other topologies. A common assumption in the communication model adopted by the previous authors is that all the packets require one unit of time in order to travel over a link (i.e., the transmission time plus the propagation delay over the link and the processing delay at the node all sum up to one unit). As a result the algorithms found were synchronous and assumed the existence of a global clock.

In this paper we will relax some of the previous assumptions. The existence of a global clock is no longer assumed. Furthermore, the lengths of the packets broadcast are not deterministic, but they are distributed according to some probabilistic rule. In order to be able to make comparisons with the fixed packet length case, the mean value of the packet lengths will be taken equal to one unit. Setup times are included in the processing time.

Manuscript received November 19, 1990; revised August 6, 1991. This work was supported by NSF under Grant NSF-DDM-8 903 385 and by the ARO under Grant DAAL03-86-K-0171.

The authors are with the Laboratory for Information and Decision Systems, Massachusetts Institute of Technology, Cambridge, MA 02139.

IEEE Log Number 9204622.

1045–9219/93$03.00 © 1993 IEEE

For both the hypercube and the ring network of processors the number of nodes will be denoted by n. For the purposes of this paper, a hypercube network (or d-cube) consists of the set of points in d-dimensional space with each coordinate equal to zero or one. There is a bidirectional communication link for every two points differing in a single coordinate. We thus obtain an undirected graph with the processors as nodes and the communication links as arcs. The binary string of length d that corresponds to the coordinates of a node of the d-cube is referred to as the *identity number* (or ID) of the node. When confusion cannot arise, we refer to a d-cube node interchangeably in terms of its identity number (a binary string of length d) and in terms of the decimal representation of its identity number. Thus, for example, the nodes $(00\cdots0)$, $(00\cdots1)$, and $(11\cdots1)$ will also be referred to as nodes 0, 1, and 2^d-1, respectively. For any two binary strings x and y we denote by $x\oplus y$ the bitwise exclusive or of x and y. We also denote by $\log n$ and $\ln n$ the base 2 and the Napier logarithm of n, respectively.

The optimal completion time of the MNB in a d-dimensional hypercube with $n=2^d$ nodes, when each packet requires one time unit (or slot) for transmission over a link, was found in [2] (see also [1]) to be $\lceil (n-1)/d\rceil$ time slots, where by $\lceil x\rceil$ we denote the smallest integer which is greater or equal to x. We evaluate the time complexity of the optimal algorithm described in [2] when the packet lengths are not constant, but are distributed according to some probabilistic rule. We consider the natural adaptation of the optimal deterministic algorithm which can deal with the probabilistic case. In particular, we assume that the same schedule of packet transmissions (i.e., order in which the packets are transmitted over the links) with that of the optimal synchronous algorithm is followed; however, the timing is not the same, since the model is no longer deterministic. Note that no synchronization is needed for the algorithm to work. Let T_{MNB} be the time required for the completion of the MNB in the asynchronous probabilistic case and let

$$T_s=\left\lceil\frac{n-1}{d}\right\rceil$$

be the corresponding optimal time for the synchronous deterministic case. For a given n, T_{MNB} is a random variable and T_s is a constant. We assume that the probability distribution of the packet lengths has unit mean and that the corresponding characteristic function $\Phi(s)$ exists for some $s>0$. We prove that given any $\delta>0$ and $\epsilon>0$, we can find $n_0=n_0(\delta,\epsilon)$ such that

$$\Pr(T_{MNB}\leq(1+\delta)T_s)\geq 1-\epsilon \qquad \text{(for the hypercube)}$$

for all $n\geq n_0$. This means that as $n\to\infty$, the MNB is completed with probability one in time less than $(1+\delta)T_s$, where δ is arbitrarily small. Thus, the probabilistic nature of the packet lengths does not deteriorate appreciably the performance of the optimal MNB algorithm for large hypercubes.

This is a rather surprising result. As shown in Section II, in the case of the ring, the average completion time of the MNB with random packet lengths increases substantially over the corresponding deterministic case. In particular, the mean time $E(T_{MNB})$ required to complete the MNB in a ring for exponentially distributed length of packets with mean one time unit is

$$E(T_{MNB})\approx(C+\ln n)T_s, \qquad \text{(for the ring with } n \text{ nodes)}$$

where $C=0.577215$ is Euler's number, and

$$T_s=\left\lceil\frac{n-1}{2}\right\rceil$$

is the time to execute the MNB in the case of unit packet lengths.

The organization of the paper is the following. The MNB in a ring is treated in Section II. The analysis found there is rather straightforward, but it does give some insight for the case of the hypercube. Sections III–V deal with the hypercube network of processors. In Section III we describe the communication algorithm (scheduling) that will be used. Section IV derives a loose upper bound for $E(T_{MNB})$ when the packet lengths are exponentially distributed. Our main result, which holds for general distributions of the packet lengths, appears in Propositions 3 and 4 of Section V. Finally, the appendices at the end resolve some technical issues arising in our analysis.

II. Multinode Broadcast in a Ring

Consider a multinode broadcast in a ring. In the case when all the packets require one unit of time (or slot) to be transmitted over a link, the optimal time to perform the task is $\lceil (n-1)/2\rceil$, where n is the number of processors of the ring. The following algorithm, found in [1], achieves this optimal time.

At the first slot, each node sends its packet to its clockwise and counterclockwise neighbors. During slots $2,\cdots,\lceil(n-1)/2\rceil$, every node sends to its clockwise (counterclockwise) neighbor the packet received from its counterclockwise (respectively, clockwise neighbor) at the previous slot (Fig. 1).

We are interested in the case where the lengths of the packets generated by the nodes (and therefore the time required to transmit them over a link) are not constant but follow some probability distribution. We will evaluate the mean time $E(T_{MNB})$ required to complete the MNB.

The remainder of this section consists of two parts. In Section II-A we derive a general expression for $E(T_{MNB})$ and apply it to the case where the packet lengths follow a uniform distribution. In Section II-B we calculate $E(T_{MNB})$ for the case of exponentially distributed length of packets.

A. General Expression for $E(T_{MNB})$—Uniform Distribution and Bounded Distribution of Lengths

Let $x_i, i=1,\cdots,n$, be the time required to transmit the packet generated at node i over a link of the ring. Since for each processor i there is a processor which is $\lceil(n-1)/2\rceil$ links away from i, and since the packet from i has to be broadcast to all the other nodes, we have $T_{MNB}\geq\lceil(n-1)/2\rceil x_i$, for all i. Thus,

$$T_{MNB}\geq\left\lceil\frac{n-1}{2}\right\rceil\max_i x_i.$$

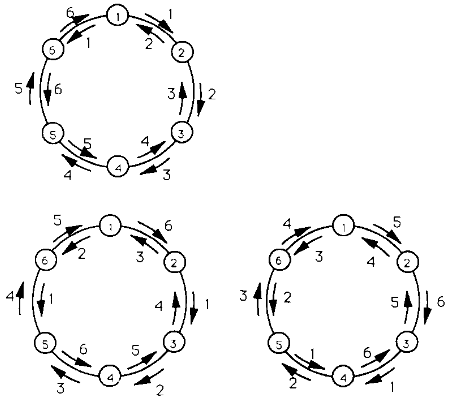

Fig. 1. Broadcast in a ring.

On the other hand, it can be seen that

$$T_{MNB} \le \left\lceil \frac{n-1}{2} \right\rceil \max_i x_i.$$

by observing that a MNB in a ring with packets of lengths (in time units) $x_i, i = 1, \cdots, n$. cannot require more time than the time required when all the packets are equal to the largest of them. An alternative way to prove this inequality, is to note that the largest packet generated by a node is the bottleneck of the algorithm, in the sense that by the time it arrives to the node opposite to its source on the ring, all the other packets of smaller length have already arrived to the node opposite to their source.

Thus $T_{MNB} = \lceil (n-1)/2 \rceil \max_i x_i$, and

$$E(T_{MNB}) = \left\lceil \frac{n-1}{2} \right\rceil E(\max_i x_i). \tag{1}$$

This equation holds for any probability distribution of the x_i's. In what follows we will examine specific distributions.

We first note that if the x_i's are bounded by a constant B, then the time required to execute the MNB is bounded by $\lceil (n-1)/2 \rceil B$. Assume now that the lengths of the packets are independent and uniformly distributed over the interval $(0, u)$. Then, straightforward calculation yields

$$E(\max_i x_i) = \frac{un}{n+1}.$$

and the desired time is

$$E(T_{MNB}) = \frac{un}{n+1} \left\lceil \frac{n-1}{2} \right\rceil.$$

If $u = 2$ so that the mean transmission time $E(x_i)$ of a packet is one time unit, the preceding formula becomes

$$E(T_{MNB}) = \frac{2n}{n+1} \left\lceil \frac{n-1}{2} \right\rceil = \frac{2n}{n+1} \cdot T_s$$

where T_s denotes the optimal completion time for the corresponding deterministic case.

B. Exponentially Distributed Length of Packets

Assume now that the transmission times x_i over a single link are exponentially distributed independent random variables with mean one time unit. Then we have

$$\Pr(x_i \le X) = 1 - e^{-X}.$$

Since $\Pr(\max_i x_i \le X) = \Pr(x_1 \le X, x_2 \le X, \cdots, x_n \le X)$ and the x_i's are independent and identically distributed random variables, we obtain

$$\Pr(\max_i x_i \le X) = (1 - e^{-X})^n.$$

On the other hand we have

$$E(\max_i x_i) = \int_0^\infty \Pr(\max_i x_i \ge X)\, dX = \int_0^\infty (1 - (1 - e^{-X})^n)\, dX.$$

By calculating this integral, we get, after some manipulation, that

$$E(\max_i x_i) = 1 + \frac{1}{2} + \cdots + \frac{1}{n}.$$

Therefore,

$$E(T_{MNB}) = \left\lceil \frac{n-1}{2} \right\rceil \left(1 + \frac{1}{2} + \cdots + \frac{1}{n}\right).$$

For large n, the sum of the n first terms of the harmonic series is approximately $\ln n$. In order to see that, we integrate $1/x$ from 1 to n, and bound the integral from above and below by discretizing it. Then we find that

$$\frac{1}{2} + \frac{1}{3} + \cdots + \frac{1}{n} \le \int_1^n \frac{1}{x} dx = \ln n \le 1 + \frac{1}{2} + \cdots + \frac{1}{n-1},$$

which gives

$$\ln n + \frac{1}{n} \le \sum_{k=1}^{n} \frac{1}{k} \le \ln n + 1. \tag{2}$$

If more accuracy is required, the following formula ([6, p. 2]) can be used:

$$\sum_{k=1}^{n} \frac{1}{k} = C + \ln n + \frac{1}{2n} - \sum_{k=2}^{\infty} \frac{A_k}{n(n+1)\cdots(n+k-1)} \tag{3}$$

where

$$A_k = \frac{1}{k} \int_0^1 x(1-x)(2-x)(3-x)\cdots(k-1-x)\, dx$$

and $C = 0.577215$ is Euler's constant. It is shown in Appendix 1 that

$$\lim_{n\to\infty} \left(\frac{1}{2n} - \sum_{k=2}^{\infty} \frac{A_k}{n(n+1)\cdots(n+k-1)} \right) = 0.$$

Therefore, for large n, we obtain the following approximation:

$$E(T_{MNB}) = \left\lceil \frac{n-1}{2} \right\rceil \left(1 + \frac{1}{2} + \cdots + \frac{1}{n}\right) \approx \left\lceil \frac{n-1}{2} \right\rceil (C + \ln n).$$

By comparing the results for the uniform and the exponential distributions, one can see an interesting difference. For the uniform distribution with unit mean packet length ($u = 2$), the expected time for the completion of the multinode broadcast is

$$\frac{2n}{n+1} \cdot T_s \approx 2T_s$$

where T_s is the completion time for the corresponding synchronous deterministic case; for the exponential distribution the corresponding mean value of the completion time is approximately $\ln\, n \cdot T_s$. Thus, the mean completion time for a MNB in a ring strongly depends upon the particular distribution of the packet lengths. It is not only the mean, but also the tail of the distribution that plays a significant role. We will see that in the hypercube, the particular distribution of the packet lengths plays a less important role. All that matters in this case is the mean of the packet lengths.

III. Modified Algorithm for the MNB in a Hypercube

In the remainder of the paper we will be dealing with a multinode broadcast in a hypercube network of processors. In order to obtain complexity results for the MNB in the random packet length case, we will analyze a slightly modified version of the optimal synchronous algorithm found in [2] and [1]. The results obtained for the modified algorithm will hold for the optimal algorithm as well. In this section, we first explain why a modified algorithm is analyzed instead of the asynchronous version of the optimal algorithm. We then give the description of the modified algorithm.

The optimal synchronous algorithm of [2] can be viewed as consisting of d phases. During phase i the packet that originates at node s, where $s = 0, \cdots, n-1$, arrives at the nodes that are located i links away from s. When d is not prime, the phases may have to overlap in order for the completion time of the MNB to be strictly optimal. This complicates the analysis of the asynchronous case, since the time required to complete a phase will be affected by previous phases that have not finished yet. The reason is that in phase $i+1$, a packet of some origin node may be scheduled to be transmitted over some link after the packet of another origin, but the latter packet may not have yet completed phase i.

To circumvent this difficulty, we modify the algorithm so that its phases are not allowed to overlap. The modified algorithm, is the same with the optimal algorithm except that there is a constraint that each packet begins phase $i+1$ only after all packets have completed phase i. The completion time of the modified algorithm is slightly larger than the actual T_{MNB} achieved by the optimal algorithm. Note, however, that the modified algorithm is introduced strictly for the purposes of analysis. In practice one would prefer to use the optimal algorithm whose running time is (slightly) superior to the one of the modified algorithm.

We now describe briefly the modified algorithm, assuming the reader is somewhat familiar with the optimal synchronous version given in [2] and [1]. We first note that if we find n synchronous single node broadcast algorithms in a hypercube, each one originating at a different node, and such that no two of them use the same link during a slot, then we have a MNB algorithm.

Let $A_l(0)$ be the set of links on which the packet originated at node $00\cdots0$ is transmitted during the lth slot. Obviously, each link in $A_l(0)$ connects two nodes with ID's that differ in a specific bit position. Our aim will be to define $A_l(0)$ in a way that no two links in $A_l(0)$ connect nodes whose ID's differ in the same bit position. If we do so, then the sets

$$A_l(s) = \{(s \oplus x, s \oplus y) | (x, y) \in A_l(0)\}, \qquad l = 1, 2, \cdots$$

can be the sets of links on which the packet generated at node s is transmitted during the lth slot, for all $s = 0, 1, \cdots n-1$. It can be seen that $A_l(s) \cap A_l(u) = \emptyset$ since $s \oplus x$ and $s \oplus y$ differ in a particular bit if and only if x and y differ in the same bit. Thus for $s = 0, 1, \cdots, n-1$, the sets $A_l(s)$ do not have common elements for a specific l, provided of course that $A_l(0)$ satisfies the condition mentioned above. In this way it is guaranteed that no two packets will claim the same link during the same slot. We now proceed to specify $A_l(0)$.

For $i = 1, \cdots, d$, we denote with N_i the set of d-bit binary numbers with exactly i ones. The cardinality of N_i is $\binom{d}{i}$. Each set N_i is in turn partitioned in disjoint subsets $R_{i1}, \cdots, R_{in_i}$ which are equivalence classes under a single bit rotation to the left. R_{i1} is selected to be the class of the element of N_i whose i rightmost bits are unity. Then each node ID is associated with a distinct number $m(t) \in \{1, 2, \cdots, n-1\}$ in the order

$$R_{11}R_{21} \cdots R_{2n_2} \cdots R_{(d-2)n_{d-2}}R_{(d-1)1}(11\cdots1). \tag{4}$$

The first element in each set R_{ij} is chosen so that its bit in position $1 + [(m(t) - 1)(\bmod d)]$ from the right, is a one. The subsequent elements of R_{ij} are found by rotating the first element to the left. We successively group together d elements of the set N_i into sets $E_{ij}, j = 1, \cdots, \lceil \binom{d}{i}/d \rceil$ in the order they appear in (4). When d is a prime integer, each one of the equivalence classes R_{ij} (except for R_{01} and R_{d1}) has d elements. This happens because when d is prime, all left shifts of less than d positions produce distinct d bit binary numbers. Then there are $\binom{d}{i}/d$ equivalence classes in N_i, each with d elements, and the sets R_{ij} and E_{ij} coincide. If d is not prime, then for each i we will have $\lceil \binom{d}{i}/d \rceil$ sets E_{ij} which we order as in (4).

Consider now the following graph consisting of nodes $0, 1, \cdots, n-1$. Every element of E_{ij} is connected to an element in N_{i-1} so that each connection corresponds to a reversal of a different bit. Every element of E_{ij} is connected to exactly one element of N_{i-1}. In this way a graph is obtained which is the single node broadcast tree for node 0. The links connecting layers $l-1$ and l of this graph constitute the links of $A_l(0)$. By construction, these links satisfy the conditions that we have set for $A_l(0)$ (see also [1, p. 60]). Then, the broadcast tree of node s can be obtained from the broadcast tree of node 0 by simply adding s to the ID of each node of the tree. The broadcast tree of node 0 for $d = 5$ is shown in Fig. 2.

IV. A Loose Upper Bound for $E(T_{MNB})$

Assume that the lengths of the packets that are broadcast by the nodes are independent exponentially distributed random variables, and consider the asynchronous version of the

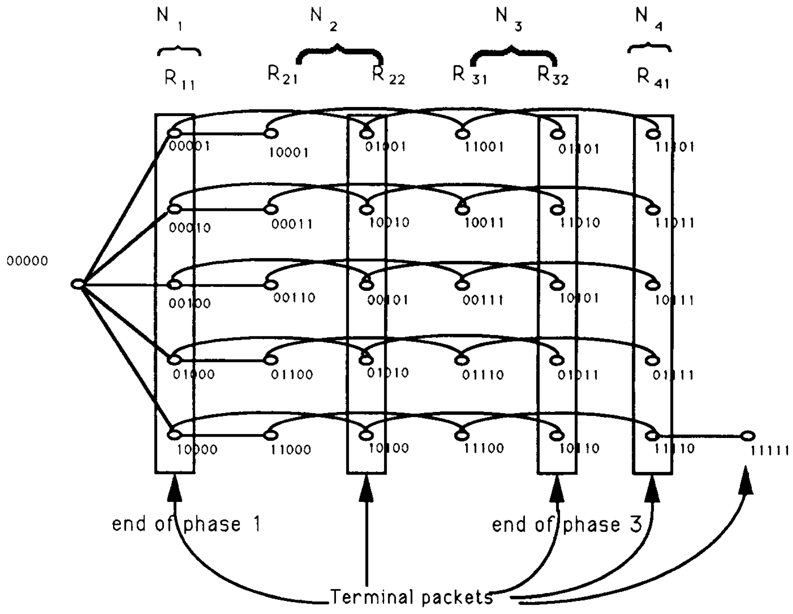

Fig. 2. The broadcast tree of node 0 for $d = 5$.

algorithm presented in Section III. This algorithm should be interpreted as specifying the order in which the packets are transmitted over the links and not the exact timing. We are interested in the completion time T_{MNB} of the algorithm.

The following upper bound for the average time required for the MNB can be found using the reasoning developed in Section II for the ring topology:

$$E(T_{MNB}) \leq (1 + 1/2 + 1/3 + \cdots + 1/n)T_s \approx (\ln n + C)T_s \tag{5}$$

where C is Euler's constant and

$$T_s = \left\lceil \frac{n-1}{d} \right\rceil$$

is the time required by the deterministic optimal algorithm.

Relation (5) can be proved by arguing that the time for the MNB when the packets which are broadcast by nodes $0, 1, \cdots, n-1$ have variable lengths $x_0, x_1, \cdots, x_{n-1}$, cannot be longer than the time for a MNB in which all packets have lengths equal to $\max_i x_i$. This gives

$$T_{MNB} \leq (\max_i x_i)T_s,$$

and by using the relation

$$E(\max_i x_i) = 1 + 1/2 + \cdots + 1/n \approx \ln n + C,$$

used in Section II, we obtain the bound (5).

However, this bound is not tight as in the case of the ring, because the packet with the greatest length is not necessarily the one that determines the completion time of the MNB in a hypercube. The reason is that, during the MNB in hypercubes, a packet is wasting most of the time waiting behind other packets that were scheduled to use a link before it. Thus, although the bound in (5) is valid, it is not tight because it corresponds to a case in which the packet having the maximum length among all the packets has to wait behind other packets which also have maximum length. This is not a typical scenario and the mean completion time is considerably overestimated.

V. Asymptotic Behavior of T_{MNB} as $N \to \infty$

To obtain a tighter estimate of T_{MNB}, a different approach based on Markoff's inequality will be used. We will look at the asymptotic behavior of the T_{MNB} as the dimension of the hypercube increases.

During phase i of the synchronous modified algorithm, the packets are received by all the destinations that are at a Hamming distance i from the source of the packets. Phase i consists of

$$\beta_i = \left\lceil \frac{\binom{d}{i}}{d} \right\rceil$$

steps.

As already mentioned, in order to analyze the case of random packet lengths, the scheduling of the asynchronous modified algorithm described in Section III will be used. Phase i of the algorithm is considered to have been completed only when all the packets have completed phase i. There are d copies of each packet, which upon completing transmission, mark the end of phase i of this packet. These are the $r_d(\binom{d}{i})$ copies which are transmitted during step $\Sigma_{j=1}^{i} \beta_j$ (last step of phase i), and $d - r_d(\binom{d}{i})$ copies which are transmitted during step $\Sigma_{j=1}^{i} \beta_j - 1$ (next to the last step of phase i) of the synchronous algorithm, where we denote by $r_x(y)$ the remainder of the division of y by x. We will refer to these packets as the *terminal packets* of phase i. In Fig. 2 we show the terminal packets of phase 2 for the broadcast tree rooted at node 0. Let us number the terminal packets of phase i as

$j = 1, 2, \cdots, nd$ (there are d terminal packets per origin node, so the total is nd). Let us also denote by w_{ij} the time that elapses between the beginning of phase i and the time the jth of these terminal packets completes phase i. For every terminal packet of phase i, the time required for the packet to complete phase i consists of at most β_i packet transmissions; $\beta_i - 1$ or $\beta_i - 2$ transmissions of other packets that were scheduled to use the link before it, and one transmission of the copy of the packet under consideration. Then we can write

$$w_{ij} = \sum_{l \in A_{ij}} x_l.$$

where A_{ij} is a set of distinct integers between 0 and $n - 1$ which has cardinality less or equal to β_i, and x_l is the length of the packet which originates from node l. In particular, a node belongs to set A_{ij} if a packet generated at that node is scheduled to use during the ith phase the same link that terminal packet j of phase i will use.

Let

$$\Phi(s) = E(e^{sx})$$

be the characteristic function of the distribution of the packet lengths and let

$$\mathcal{D}^+ = \{s > 0 | E(e^{sx}) \text{ exists}\}$$

be the positive portion of its domain. Since w_{ij} is the sum of at most β_i independent and identically distributed random variables, we have (since $\Phi(s) > 1$ for $s \in \mathcal{D}^+$)

$$E(e^{sw_{ij}}) \leq \Phi(s)^{\beta_i}, \qquad \forall s \in \mathcal{D}^+.$$

Also, from Markoff's inequality [13] we have

$$\Pr(e^{sw_{ij}} \geq a) \leq \frac{E(e^{sw_{ij}})}{a}, \qquad \forall a > 0, s \in \mathcal{D}^+.$$

From the last two relations we obtain

$$\Pr\left(w_{ij} \leq \frac{\ln a}{s}\right) \geq 1 - \frac{\Phi(s)^{\beta_i}}{a}, \qquad \forall a > 0, s \in \mathcal{D}^+. \quad (6)$$

The time required for the completion of the ith phase by the nd terminal packets of that phase is

$$T_i = \max_{j=1,\cdots,nd}\{w_{ij}\}.$$

Therefore, for all $s \in \mathcal{D}^+$ and $a > 0$,

$$\Pr\left(T_i \leq \frac{\ln a}{s}\right) = \Pr\left(w_{ij} \leq \frac{\ln a}{s}, j = 1, \cdots, nd\right). \quad (7)$$

The w_{ij}'s that appear at the right hand side of (7) are not independent random variables, so we cannot readily use (6) to estimate $\Pr(T_i \leq \ln a/s)$. To proceed further with the analysis, the following two propositions are needed:

Proposition 1: Let $IJ \subset N \times N$ be a subset of the set of index pairs (i, j) with $1 \leq i \leq d$, and $1 \leq j \leq nd$. Then for all $m \in \{1, \cdots, d\}, k \in \{1, \cdots, nd\}$, and all $a > 0$, we have

$$\Pr(w_{ij} \leq a, (i, j) \in IJ | w_{mk} \leq a) \geq \Pr(w_{ij} \leq a, (i, j) \in IJ).$$

Proof: We know that $w_{ij} = \Sigma_{l \in A_{ij}} x_l$. Let $\overline{x} = (x_0, x_1, \cdots, x_{n-1})$ be the random vector of packet lengths. We define the sets

$$E_{ij} = \{\overline{x} \in R^n | w_{ij} = \max_{(p,q) \in IJ} w_{pq}\}$$
$$= \left\{\overline{x} \in R^n \middle| \sum_{l \in A_{ij}} x_l = \max_{(p,q) \in IJ} \sum_{l \in A_{pq}} x_l\right\}.$$

In order for the sets $E_{ij}, (i, j) \in IJ$ to be disjoint, we say that if for some $\overline{x}$ there are two pairs (i_1, j_1) and (i_2, j_2) in IJ, with $(i_1, j_1) \prec (i_2, j_2)$ ($\prec$ is any order in $N \times N$, for example the lexicographic order), such that $w_{i_1 j_1} = w_{i_2 j_2}$, then this $\overline{x}$ will be considered to belong to $E_{i_1 j_1}$ and not to $E_{i_2 j_2}$. Then

$$\Pr(w_{ij} \leq a, (i, j) \in IJ | w_{mk} \leq a) = \sum_{(i,j) \in IJ} \Pr(\overline{x} \in E_{ij}) \Pr(w_{ij} \leq a | \overline{x} \in E_{ij}, w_{mk} \leq a). \quad (8)$$

In Appendix 2 we prove that $\Pr(w_{ij} \leq a | w_{mk} \leq b) \geq \Pr(w_{ij} \leq a)$ for all $a, b > 0$. It can be seen that the proof of the last inequality is not altered if both probabilities are conditioned on the fact $\overline{x} \in E_{ij}$. Thus,

$$\Pr(w_{ij} \leq a | w_{mk} \leq a, \overline{x} \in E_{ij}) \geq \Pr(w_{ij} a | \overline{x} \in E_{ij}).$$

The last relation together with (8) yields

$$\Pr(w_{ij} \leq a, (i, j) \in IJ | w_{mk} \leq a)$$
$$\geq \sum_{(i,j) \in IJ} \Pr(\overline{x} \in E_{ij}) \Pr(w_{ij} \leq a | \overline{x} \in E_{ij})$$
$$= \Pr(w_{ij} \leq a, (i, j) \in IJ).$$

Note that if in the preceding proof we replace w_{ij} by $w_{ij} + a - a_{ij}$ we get the more general relation

$$\Pr(w_{ij} \leq a_{ij}, (i, j) \in IJ | w_{mk} \leq a_{mk}) \geq \Pr(w_{ij} \leq a_{ij}, (i, j) \in IJ). \quad (9)$$

Q.E.D.

As noted already, the w_{ij}'s are not independent and their joint distribution is not equal to the product of the marginal distributions. However, the following proposition holds.

Proposition 2: For all $a > 0$ and $k \in \{1, \cdots, nd\}$,

$$\Pr(w_{ij} \leq a, j = 1, \cdots, k) \geq \prod_{j=1}^{k} \Pr(w_{ij} \leq a). \quad (10)$$

Proof: The proof will be given by induction on k. For $k = 2$, we know from Appendix 2 that $\Pr(w_{i1} \leq a | w_{i2} \leq a) \geq \Pr(w_{i1} \leq a)$, which by using Bayes' rule yields

$$\Pr(w_{i1} \leq a, w_{i2} \leq a) \geq \Pr(w_{i1} \leq a) \Pr(w_{i2} \leq a).$$

Suppose that the proposition is true for $k - 1$ or equivalently

$$\Pr(w_{ij} \leq a, j = 1, \cdots, k - 1) \geq \prod_{j=1}^{k-1} \Pr(w_{ij} \leq a). \quad (11)$$

Then

$$\Pr(w_{ij} \le a, j = 1, \cdots, k) = \\ \Pr(w_{ij} \le a, j = 1, \cdots, k-1 | w_{ik} \le a) \Pr(w_{ik} \le a). \tag{12}$$

By using this equation together with Proposition 1 we obtain

$$\Pr(w_{ij} \le a, j = 1, \cdots, k) \\ \ge \Pr(w_{ij} \le a, j = 1, \cdots, k-1) \Pr(w_{ik} \le a)$$

which with the aid of (11) yields

$$\Pr(w_{ij} \le a, j = 1, \cdots, k) \ge \prod_{j=1}^{k} \Pr(w_{ij} \le a),$$

thus completing the induction. Q.E.D.

The next proposition is an intermediate result leading to our main result. It is a lower bound on the probability that T_{MNB} is less than some appropriate expression.

Proposition 3: Let s be a scalar in the positive portion $\mathcal{D}^+$ of the domain of $\Phi(s)$. Assume that $\mu > 0, \lambda > 0$ are any scalars such that

$$\xi \overset{\text{def}}{=} \frac{e^{\lambda s}}{\Phi(s)} > 1, \quad \zeta \overset{\text{def}}{=} \sqrt{\frac{e^{\mu s}}{\Phi(s)}} > 2.$$

Then for each $s_0 \in \mathcal{D}^+$ and $\theta > 1$, we have

$$\Pr\left(T_{MNB} \le \lambda \sum_{i=3}^{d-3} \beta_i + 2\mu\beta_2 + 3\frac{s_0}{\theta}\ln n\right) \\ \ge \left(1 - \frac{1}{\zeta^{-1} n^{\log_2 \zeta}}\right)^{2nd} \\ \cdot \left(1 - \frac{\Phi(s_0)}{n^\theta}\right)^{3n} \left(1 - \frac{1}{n^{(1/7)d \log_2 \xi}}\right)^{nd^2}$$

for sufficiently large dimension d.

Proof: By combining (7) and (10) we get

$$\Pr\left(T_i \le \frac{\ln a}{s}\right) \ge \prod_{j=1}^{nd} \Pr\left(w_{ij} \le \frac{\ln a}{s}\right),$$

which with the aid of (6) gives

$$\Pr\left(T_i \le \frac{\ln a}{s}\right) \ge \left(1 - \frac{\Phi(s)^{\beta_i}}{a}\right)^{nd}. \tag{13}$$

Inequality (13) is valid for any $a > 0$ and any $s \in \mathcal{D}^+$. We select $\lambda > 0$ so that $\xi = e^{\lambda s}/\Phi(s) > 1$ and let $a = e^{\lambda s \beta_i}$. By substituting these values of a and ξ in (13), we obtain

$$\Pr(T_i \le \lambda\beta_i) \le \left(1 - \frac{1}{\xi^{\beta_i}}\right)^{nd}. \tag{14}$$

By using the obvious inequality

$$\Pr\left(\sum_{i=p}^{q} T_i \le \sum_{i=p}^{q} c_i\right) \ge \Pr(T_i \le c_i, i = p, \cdots, q)$$

we further obtain

$$\Pr\left(\sum_{i=3}^{d-3} T_i \le \lambda \sum_{i=3}^{d-3} \beta_i\right) \ge \Pr(T_i \le \lambda\beta_i, i = 3, 4, \cdots, d-3). \tag{15}$$

Note that the T_i's are not independent. However, we prove in Appendix 3 that for all $a_i > 0$,

$$\Pr(T_i \le a_i, i = 1, 2, \cdots, m-1 | T_m \le a_m) \\ \ge \Pr(T_i \le a_i, i = 1, 2, \cdots, m-1). \tag{16}$$

(Relation (16) is intuitively clear, since the knowledge that the duration of the mth phase is less than a_m, cannot decrease the probability that the duration of phases $i = 1, 2, \cdots, m-1$ is less than $a_i, i = 1, 2, \cdots, m-1$ below its ***a priori*** value.)

By successive use of (16), we obtain from (15) that

$$\Pr\left(\sum_{i=3}^{d-3} T_i \le \lambda \sum_{i=3}^{d-3} \beta_i\right) \ge \sum_{i=3}^{d-3} \Pr(T_i \le \lambda\beta_i).$$

By combining the relations (14) for $i = 3, 4, \cdots, d-3$ we get

$$\Pr\left(\sum_{i=3}^{d-3} T_i \le \lambda \sum_{i=3}^{d-3} \beta_i\right) \ge \prod_{i=3}^{d-3} \Pr(T_i \le \lambda\beta_i) \\ \ge \prod_{i=3}^{d-3} \left(1 - \frac{1}{\xi^{\beta_i}}\right)^{nd}. \tag{17}$$

Since $\xi > 1$ and $\beta_i = \lceil \binom{d}{i}/d \rceil \ge \beta_3$ for $3 \le i \le d-3$, we get $\xi^{\beta_i} > \xi^{\beta_3}$ and (17) can be transformed to

$$\Pr\left(\sum_{i=3}^{d-3} T_i \le \lambda \sum_{i=3}^{d-3} \beta_i\right) \ge \left(1 - \frac{1}{\xi^{\beta_3}}\right)^{nd^2}. \tag{18}$$

Furthermore , we have that $\beta_3 = \lceil (d-1)(d-2)/6 \rceil > d^2/7$ for sufficiently large d. This in turn gives $\xi^{\beta_3} \ge 2^{d^2 \log_2 \xi/7} = n^{d \log_2 \xi/7}$, since $\xi > 1$. Using this, relation (18) gives

$$\Pr\left(\sum_{i=3}^{d-3} T_i \le \lambda \sum_{i=3}^{d-3} \beta_i\right) \ge \left(1 - \frac{1}{n^{(1/7)d \log_2 \xi}}\right)^{nd^2} \tag{19}$$

with $\xi > 1$.

Note that $\Sigma_{i=3}^{d-3} T_i$ is the time required for the completion of phases $3, 4, \cdots, d-4, d-3$. Phases $1, 2, d-2, d-1$ and d will be treated separately.

The time to complete phases $1, d-1$ and d is determined by the length of the longest packet. This is so because these phases consist of one step and, therefore, $T_i = \max_l x_l$, for $i = 1, d-1, d$. Then by using Markoff's inequality we obtain

$$\Pr\left(T_i \le \frac{\ln a}{s}\right) = \Pr\left(x_l \le \frac{\ln a}{s}, l = 0, 1, \cdots, n-1\right) \\ \ge \left(1 - \frac{\Phi(s)}{a}\right)^n, \quad \text{for } i \in \{1, d-1, d\}.$$

If we select $a = n^\theta$ with $\theta > 1$ and we let s be equal to any $s_0 \in \mathcal{D}^+$, the preceding inequality yields:

$$\Pr\left(T_i \le \frac{\theta}{s_0}\ln n\right) \ge \left(1 - \frac{\Phi(s_0)}{n^\theta}\right)^n, \quad \text{for } i \in \{1, d-1, d\}. \tag{20}$$

For the phases $i = 2$ and $i = d - 2$, we get from relation (13) and the fact $\beta_2 = \beta_{d-2}$, that

$$\Pr\left(T_i \le \frac{\ln a}{s}\right) \ge \left(1 - \frac{\Phi(s)^{\beta_2}}{a}\right)^{nd}, \qquad \text{for } i \in \{2, d-2\}. \tag{21}$$

If we select $a = e^{\mu s \beta_2}$, then (21) is transformed to

$$\Pr(T_i \le \mu\beta_2) \ge \left(1 - \frac{1}{\zeta^{2\beta_2}}\right)^{nd}, \qquad \text{for } i \in \{2, d-2\} \tag{22}$$

where

$$\zeta = \sqrt{\frac{e^{\mu s}}{\Phi(s)}} > 2.$$

Since $2\beta_2 = 2\lceil (d-1)/2 \rceil \ge d - 1$, we get

$$\begin{aligned} \Pr(T_i \le \mu\beta_2) &\ge \left(1 - \frac{1}{\zeta^{d-1}}\right)^{nd} \\ &= \left(1 - \frac{1}{\zeta^{-1} n^{\log_2 \zeta}}\right)^{nd}, \qquad \text{for } i \in \{2, d-2\}. \end{aligned} \tag{23}$$

By combining (19), (20), and (23), and by using the fact $\Sigma_{i=1}^{d} T_i = T_{MNB}$, we finally obtain

$$\begin{aligned} \Pr\left(T_{MNB} \le \lambda \sum_{i=3}^{d-3} \beta_i + 3\frac{\theta}{s_0} \ln n + 2\mu\beta_2\right) &\ge \left(1 - \frac{1}{\zeta^{-1} n^{\log_2 \zeta}}\right)^{2nd} \left(1 - \frac{\Phi(s_0)}{n^\theta}\right)^{3n} \\ &\cdot \left(1 - \frac{1}{n^{(1/7)d \log_2 \xi}}\right)^{nd^2} \end{aligned} \tag{24}$$

with $\theta > 1, s_0 \in \mathcal{D}^+, \zeta > 2$, and $\xi = e^{\lambda s}/\Phi(s) > 1$. Q.E.D.

Now we are in a position to prove the main result of the paper.

Proposition 4: Let T_{MNB} be the completion time of the MNB when the lengths of the packets are distributed according to some probabilistic rule with unit mean, and let $T_s = \lceil (n-1)/d \rceil$ be the completion time of the MNB when packets have deterministic length equal to one time unit. Assume also that the positive portion $\mathcal{D}^+$ of the domain of $\Phi(s)$ is nonempty. Then given any $\delta > 0$ and $\epsilon > 0$, we can find $n_0 = n_0(\delta, \epsilon)$ such that

$$\Pr(T_{MNB} \le (1 + \delta) T_s) \ge 1 - \epsilon, \qquad \forall n \ge n_0.$$

Proof: Since $\Phi(s) < \infty$ for $s \in \mathcal{D}^+$, there exists a $\lambda > 0$ such that $e^{\lambda s} > \Phi(s)$ for some $s > 0$, and a $\mu > 0$ such that $e^{\mu s} > 4\Phi(s)$. Let

$$\xi = \frac{e^{\lambda s}}{\Phi(s)} > 1, \qquad \zeta = \sqrt{\frac{e^{\mu s}}{\Phi(s)}} > 2.$$

Consider also some $s_0 \in \mathcal{D}^+$ and $\theta > 1$. Then the conditions of Proposition 3 are satisfied and (24) holds. In Appendix 4 we prove that for

$$\xi > 1, \quad \zeta > 2, \quad \theta > 1. \tag{25}$$

the right hand side of (24) goes to one as the number of processors n goes to infinity. Thus, for all $\epsilon > 0$, we can find $n_1(\epsilon)$ such that for all $n \ge n_1(\epsilon)$

$$\Pr\left(T_{MNB} \le \lambda \sum_{i=3}^{d-3} \beta_i + 3\frac{\theta}{s_0} \ln n + 2\mu\beta_2\right) \ge 1 - \epsilon. \tag{26}$$

We denote

$$T_a(n) = \lambda \sum_{i=3}^{d-3} \beta_i + 3\frac{\theta}{s_0} \ln n + 2\mu\beta_2$$

and

$$T_s(n) = \left\lceil \frac{n-1}{d} \right\rceil.$$

$T_s(n)$ is the optimal completion time of the MNB for the synchronous (deterministic) case. Since $n - 1 = \Sigma_{i=1}^{d} \binom{d}{i}$ it can be seen that

$$\left\lceil \frac{n-1}{d} \right\rceil \le \sum_{i=1}^{d} \beta_i = \sum_{i=1}^{d} \left\lceil \frac{\binom{d}{i}}{d} \right\rceil < \frac{n-1+d}{d}.$$

From this fact, it follows with some additional calculation that

$$\lim_{n \to \infty} \frac{T_a(n)}{T_s(n)} = \lambda.$$

Therefore, given some $\delta > 0$ we can find $n_2(\delta)$ such that

$$\lambda \sum_{i=3}^{d-3} \beta_i + 3\frac{\theta}{s_0} \ln n + 2\mu\beta_2 \le \left(\lambda + \frac{\delta}{2}\right) T_s(n) \tag{27}$$

for all $n > n_2(\delta)$.

We define $n_0(\delta, \epsilon) = \max(n_1(\epsilon), n_2(\delta))$. For $n > n_0(\delta, \epsilon)$, both (26) and (27) hold. Thus for any $\delta > 0$ and any $\epsilon > 0$ there always exists a $n_0 = n_0(\delta, \epsilon)$, defined as above, such that for all $n > n_0(\delta, \epsilon)$

$$\Pr\left(T_{MNB} \le \left(\lambda + \frac{\delta}{2}\right) T_s\right) \ge 1 - \epsilon. \tag{28}$$

We will now prove that λ can always be chosen to be equal to $1 + (\delta/2)$ for any $\delta > 0$. It is enough to prove that there exists an $s \in \mathcal{D}^+$ such that

$$e^{[1+(\delta/2)]s} > \Phi(s).$$

Let

$$F(s) = \Phi(s) e^{-[1+(\delta/2)]s}.$$

Since $F(0) = \Phi(0) = 1$, it is enough to prove that $F(s)$ is strictly decreasing in a neighborhood of 0. Since $\Phi(s)$ is differentiable (because the exponential function is differentiable), we have

$$F'(s) = e^{-[1+(\delta/2)]s} \left(\Phi'(s) - \left(1 + \frac{\delta}{2}\right) \Phi(s)\right)$$

which gives $F'(0) = -(\delta/2) < 0$ (we used the fact that $\Phi'(0) = 1$, since the packet lengths have unit mean). Therefore, there exits an $s \in \mathcal{D}^+$ such that $F(s) < 1$ or equivalently $e^{[1+(\delta/2)]s} > \Phi(s)$. Thus, we can always choose $\lambda = 1 + (\delta/2)$.

By substituting this value of λ in (28) the proof is completed. Q.E.D.

The last theorem constitutes the main result of this paper. It indicates that in hypercubes of large dimension, the factor by which the completion time of the MNB increases when the packet lengths have essentially any probability distribution over the corresponding case where all the packets have constant lengths, is very close to one. This result should be compared with the case of the ring with n processors, where the average time required for the MNB when the lengths of the packets are exponentially distributed is ln n times that of the corresponding deterministic case.

An intuitive explanation of this result is the following. Loosely speaking, as d increases, the number of steps within each phase (except phases $1, 2, d-2, d-1, d$) grows faster than d. Therefore, the number of packets that are transmitted one after the other within some phase increases more rapidly than the number of phases. The sum of the lengths of the packets which are transmitted over some link during phase i is then forced by the central limit theorem to come close to its mean value b_i. As a result the total time required for the MNB of the asynchronous probabilistic case approaches the time complexity of the synchronous deterministic case. Therefore, although the result may seem unexpected, it is not counterintuitive. We also remark that the MNB algorithm of [9] and [14] have different properties than those of the algorithms presented here (the phases are of different durations). Thus a result analogous to the one proved here may not hold for these algorithms.

We finally note that the existence of a global clock was not assumed at any point throughout the analysis. The only exception is the initialization of the algorithm, which was assumed to take place synchronously for all the processors. If this assumption is relaxed the additional overhead for the initialization using a naive scheme (single node broadcast of a start signal) will be $O(d)$ which is small, so the result still holds.

Appendix 1

In this Appendix we will prove that

$$\lim_{n\to\infty}\left(\frac{1}{2n}-\sum_{k=2}^{\infty}\frac{A_k}{n(n+1)\cdots(n+k-1)}\right)=0.$$

Denote

$$F(n)=\sum_{k=2}^{\infty}\frac{A_k}{n(n+1)\cdots(n+k-1)}.$$

Since $A_k > 0$, obviously $F(n) > 0$ for all $n > 0$. From (2) and (3) we get that $C - F(n) \geq 1/2n \geq 0$, which gives $F(n) \leq C$, for all $n > 0$. Therefore, for $n = 1, F(1) \leq C$. It can also be seen from the definition of $F(n)$ that $F(n) \leq F(1)/n$. This gives

$$0 < F(n) \leq \frac{C}{n}.$$

Therefore, $\lim_{n\to\infty} F(n) = 0$ and

$$\lim_{n\to\infty}\left(\frac{1}{2n}-F(n)\right)=0.$$

Q.E.D.

Appendix 2

In this Appendix we prove the inequality

$$\Pr(w_{ij} \leq a | w_{mk} \leq b) \geq \Pr(w_{ij} \leq a) \tag{29}$$

which was used to prove Proposition 1.

We have

$$w_{ij} = \sum_{l\in A_{ij}} x_l, \quad w_{mk} = \sum_{l\in A_{mk}} x_l.$$

The dependence between w_{ij} and w_{mk} comes from the fact that some indexes l are common in A_{mk} and A_{ij}. Let $C = A_{mk} \cap A_{ij}, A = A_{ij} - C$ and $B = A_{mk} - C$. Then if we denote $y = \Sigma_{l\in C} x_l, z = \Sigma_{l\in A} x_l$, and $r = \Sigma_{l\in B} x_l$, the random variables y, z, r are independent, since for distinct l's, the random variables x_l are independent.

By using the above notation, it is enough to prove that

$$\Pr(y+z \leq a | y+r \leq b) \geq \Pr(y+z \leq a). \tag{30}$$

Bayes' rule gives

$$\Pr(y+z \leq a | y+r \leq b) = \frac{\Pr(y+z \leq a, y+r \leq b)}{\Pr(y+r \leq b)}.$$

We have that

$$\begin{aligned}&\Pr(y+z \leq a | y+r \leq b, a-z \leq b-r)\\ &\quad= \frac{\Pr(y+z \leq a | a-z \leq b-r)}{\Pr(y+r \leq b | a-z \leq b-r)}\\ &\quad\geq \Pr(y+z \leq a | a-z \leq b-r)\end{aligned}$$

and

$$\begin{aligned}&\Pr(y+z \leq a | y+r \leq b, a-z \geq b-r)\\ &\quad= \frac{\Pr(y+r \leq b | a-z \geq b-r)}{\Pr(y+r \leq b | a-z \geq b-r)} = 1\\ &\quad\geq \Pr(y+z \leq a | a-z \geq b-r).\end{aligned}$$

These relations give

$$\begin{aligned}&\Pr(y+z \leq a | y+r \leq b)\\ &\quad= \Pr(y+z \leq a | y+r \leq b, a-z \leq b-r)\\ &\qquad\cdot \Pr(a-z \leq b-r)\\ &\qquad+ \Pr(y+z \leq a | y+r \leq b, a-z \geq b-r)\\ &\qquad\cdot \Pr(a-z \geq b-r)\\ &\quad\geq \Pr(y+z \leq a | a-z \leq b-r)\Pr(a-z \leq b-r)\\ &\qquad+ \Pr(y+z \leq a | a-z \geq b-r)\Pr(a-z \geq b-r)\\ &\quad= \Pr(y+z \leq a).\end{aligned}$$

As a result, relation (30), which is equivalent to relation (29) holds. Q.E.D.

Appendix 3

In this Appendix, we prove (16), repeated below for convenience,

$$\begin{aligned}\Pr(T_i \le a_i, i = 1, 2, \cdots, m-1 | T_m \le a_m) \\ \ge \Pr(T_i \le a_i, i = 1, 2, \cdots, m-1).\end{aligned} \qquad (31)$$

By definition we have

$$T_i = \max_{k=1,\cdots,nd} w_{ik}.$$

Let $k_i, i = 1, 2, \cdots, m$, be the arguments that attain the maximum above for $i = 1, 2, \cdots, m$, and let $\Pr(k_i, i = 1, \cdots, m)$ be the corresponding probability. By substituting these equations in (31), we have to prove the equivalent relation

$$\begin{aligned}&\sum_{k_i} \Pr(k_i, i = 1, \cdots, m) \\ &\quad \cdot \Pr(w_{ik_i} \le a_i, i = 1, 2, \cdots, m-1 | k_i, i = 1, 2, \cdots, m, \\ &\qquad w_{mk_m} \le a_m) \\ &\quad \ge \sum_{k_i} \Pr(k_i, i = 1, \cdots, m) \Pr(w_{ik_i} \le a_i, \\ &\qquad\quad i = 1, 2, \cdots m-1 | k_i, i = 1, 2, \cdots, m).\end{aligned} \qquad (32)$$

In Proposition 1 we proved [cf. (16)] that

$$\begin{aligned}\Pr(w_{ik_i} \le a_i, i = 1, 2, \cdots, m-1 | w_{mk_m} \le a_m) \\ \ge \Pr(w_{ik_i} \le a_i, i = 1, 2, \cdots, m-1).\end{aligned} \qquad (33)$$

It can be seen from the proof of this proposition that the same result holds even if all the probabilities in (33) are conditioned on the event $\{k_i, i = 1, 2, \cdots, m\}$ and therefore

$$\begin{aligned}&\Pr(w_{ik_i} \le a_i, i = 1, 2, \cdots, m-1 | k_i, i = 1, 2, \cdots, m, \\ &\quad w_{mk_m} \le a_m) \\ &\quad \ge \Pr(w_{ik_i} \le a_i, i = 1, 2, \cdots, m-1 | k_i, \\ &\qquad i = 1, 2, \cdots, m).\end{aligned}$$

Using this, (32) is proved and from there (31) follows immediately. Q.E.D.

Appendix 4

In this Appendix we will show that the right hand side of inequality (24) goes to 1 as $n \to \infty$, i.e.,

$$\begin{aligned}\lim_{n\to\infty} \left(1 - \frac{1}{\zeta^{-1} n^{\log_2 \zeta}}\right)^{2nd} \left(1 - \frac{\Phi(s_0)}{n^\theta}\right)^{3n} \\ \cdot \left(1 - \frac{1}{n^{(1/7)d \log_2 \xi}}\right)^{nd^2} = 1\end{aligned} \qquad (34)$$

when $\xi > 1, \zeta > 2$, and $\theta > 1$. We recall that $d = \log_2 n$.

The proof consists of two steps. At first we show that all the terms of the product in (34) are of the form

$$\left(1 - \frac{1}{\Omega(x)}\right)^x$$

where $x = x(n)$ goes to infinity as n goes to infinity and $\Omega(x)$ is a function of x such that

$$\lim_{x\to\infty} \frac{x}{\Omega(x)} = 0. \qquad (35)$$

As a second step we will prove that for a function $\Omega(x)$ satisfying (35), we have that

$$\lim_{x\to\infty} \left(1 - \frac{1}{\Omega(x)}\right)^x = 1.$$

Step 1: In order to prove that all the terms in (34) are of the desired form, we will use successive applications of L'Hospital rule. This rule states that if $f(x), g(x)$ are differentiable functions in the neighborhood of ∞ (i.e., for sufficiently large x) with the property that $\lim_{x\to\infty} f(x) = \lim_{x\to\infty} g(x) = \infty$ then $\lim_{x\to\infty} f(x)/g(x) = \lim_{x\to\infty} f'(x)/g'(x)$. Although the number of processors n is an integer, we will treat it as a continuous variable here and allow differentiation with respect to it. This is permitted since we are interested in the limit $n \to \infty$ and we are dealing with continuous functions of n. Thus

1) Since $\zeta > 2$,

$$\log_2 \zeta \overset{\text{def}}{=} \alpha > 1.$$

Thus

$$\begin{aligned}\lim_{n\to\infty} \frac{2nd}{\zeta^{-1} n^a} &= \lim_{n\to\infty} \frac{2(1 + \ln n)\zeta}{(a \ln 2) n^{\alpha - 1}} \\ &= \lim_{n\to\infty} \frac{2\zeta}{(a \ln 2) n^{\alpha-1}} = 0\end{aligned}$$

as $n \to \infty$ and $\alpha > 1$. Thus

$$\zeta^{-1} n^{\log_2 \zeta} = \Omega(2nd).$$

2) Obviously

$$\lim_{n\to\infty} \frac{3n\Phi(s_0)}{n^\theta} = 0,$$

since $\theta > 1$ and $\Phi(s_0)$ is a constant.

3) We denote $\alpha = \log_2 \xi / 7 \ln 2 > 0$ for $\xi > 1$ and take into account that $d = \log_2 n = \ln n / \ln 2$. Then for the third term of (34) to be of the desired form it is enough to prove that

$$\lim_{n\to\infty} \frac{n(\ln n)^2}{(\ln 2)^2 n^{\alpha \ln n}} = 0.$$

By successive applications of L'Hospital rule we find that

$$\begin{aligned}\lim_{n\to\infty} \frac{n(\ln n)^2}{n^{\alpha \ln n}} &= \lim_{n\to\infty} \frac{n(\ln n + 2)}{2\alpha n^{\alpha \ln n}} \\ &= \lim_{n\to\infty} \frac{n(\ln n + 3)}{4\alpha^2 (\ln n) n^{\alpha \ln n}} = 0.\end{aligned}$$

Therefore

$$n^{(1/7)d \log_2 \xi} = \Omega(nd^2).$$

Step 2: For the second step we note that

$$\lim_{n\to\infty}\left(1-\frac{1}{\Omega(x)}\right)^{x}=\lim_{x\to\infty}\left(\left(1-\frac{1}{\Omega(x)}\right)^{\Omega(x)}\right)^{x/\Omega(x)}.$$

Since $\lim_{x\to\infty}(1-1/\Omega(x))^{\Omega(x)}=e^{-1}$ and $\lim_{x\to\infty} x/\Omega(x) = 0$, we finally obtain that

$$\lim_{x\to\infty}\left(1-\frac{1}{\Omega(x)}\right)^{x}=1.$$

Steps 1 and 2 together give (34). Q.E.D.

REFERENCES

[1] D. P. Bertsekas and J. N. Tsitsiklis, *Parallel and Distributed Computation: Numerical Methods.* Englewood Cliffs, NJ: Prentice-Hall, 1989.

[2] D. P. Bertsekas, C. Ozveren, G. D. Stamoulis, P. Tseng, and J. N. Tsitsiklis, "Optimal communication algorithms for hypercubes," *J. Parallel Distributed Comput.,* vol. 11, pp. 263–275.

[3] S. N. Bhatt and I. C. F. Ipsen, "How to embed trees in hypercubes," Yale Univ., Dep. Comput. Sci., Res. Rep. YALEU/DCS/RR-443, 1985.

[4] W. J. Dally and C. L. Seitz, "Deadlock-free message routing in multiprocessor interconnection networks," *IEEE Trans. Comput.,* vol. C-36, pp. 547–553, 1987.

[5] E. Dekel, D. Nassimi, and S. Sahni, "Parallel matrix and graph algorithms," *SIAM J. Comput.,* vol. 10, pp. 657–673, 1981.

[6] I. S. Gradshteyn and I. M. Ryzhik, *Table of Integrals, Series, and Products.* Orlando, FL, Academic, 1980.

[7] S. M. Hedetniemi, S. T. Hedetniemi, and A. L. Liestman, "A survey of gossiping and broadcasting in communication networks," *Networks,* vol. 18, pp. 319–349, 1988.

[8] S. L. Johnsson, "Communication efficient basic linear algebra computations on hypercube architectures," *J. Parallel Distributed Comput.,* vol. 4, pp. 133–172, 1987.

[9] S. L. Johnsson and C. T. Ho, "Optimum broadcasting and personalized communication in hypercubes," *IEEE Trans. Comput.,* vol. 38, pp. 1249–1268, 1989.

[10] D. W. Krumme, K. N. Venkataraman, and G. Cybenko, "The token exchange problem," Tufts Univ., Tech. Rep. 88-2, 1988.

[11] O. A. McBryan and E. F. Van de Velde, "Hypercube algorithms and their implementations," *SIAM J. Sci. Statist. Comput.,* vol. 8, pp. 227–287, 1987.

[12] C. Ozveren, "Communication aspects of parallel processing," Lab. for Information and Decision Systems Rep. LIDS-P-1721, M.I.T., Cambridge, MA, 1987.

[13] S. M. Ross, *Stochastic Processes.* New York: Wiley, 1983.

[14] Y. Saad and M. H. Schultz, "Data communication in hypercubes," *J. Parallel Distributed Comput.,* vol. 6, pp. 115–135, 1989.

[15] ——, "Data communication in parallel architectures," Yale Univ., Report, Mar. 1986.

[16] ——, "Topological properties of hypercubes," *IEEE Trans. Comput.,* vol. 37, pp. 867–872, 1988.

[17] G. Stamoulis and J. Tsitsiklis, "Efficient routing schemes for multiple broadcasts in hypercubes," Lab. for Information and Decision Systems, Rep. LIDS-P-1948, Feb. 1990.

[18] Q. F. Stout and B. Wagar, "Passing messages in link-bound hypercubes," in *Proc. 1986 Hypercube Conf.,* SIAM, Philadelphia, PA, 1987, pp. 251–257.

[19] D. M. Topkis, "Concurrent broadcast for information dissemination," *IEEE Trans. Software Eng.,* vol. 13, pp. 207–231, 1983.

[20] E. A. Varvarigos, "Optimal communication algorithms for multiprocessor computers," M.S. thesis, Dept. of Elec. Eng. and Comput. Sci., M.I.T.; also Center for Intelligent Control Systems Rep., CICS-TH-192, Jan. 1990.

[21] E. A. Varvarigos and D. P. Bertsekas, "Optimal communication algorithms for isotropic tasks in hypercubes and wraparound meshes," Lab. for Information and Decision Systems, Rep. LIDS-P-1972, May 1990.

Emmanouel A. Varvarigos was born in Athens, Greece, in 1965. He received a Diploma in electrical engineering (with highest honors) from the National Technical University of Athens, Greece, in 1988 and the M.S. and Electrical Engineer degrees in electrical engineering from the Massachusetts Institute of Technology in 1990 and 1991, respectively. He is currently completing his Ph.D. degree in electrical engineering at M.I.T.

In 1990 he conducted research on optical fiber communications at Bell Communications Research, Morristown, NJ. His research interests are in the areas of parallel and distributed computation and data networks.

Mr. Varvarigos received the first panhellenic prize in the Greek Mathematic Olympiad in 1982, and four times (1984–1988) the Technical Chamber of Greece Award. He is a member of the Technical Chamber of Greece.

Dimitri P. Bertsekas (S'70–SM'77–F'84) received a combined B.S.E.E. and B.S.M.E. from the National Technical University of Athens, Greece, in 1965, the M.S.E.E. degree from George Washington University in 1969 and the Ph.D. degree in system science from the Massachusetts Institute of Technology in 1971.

He has held faculty positions with the Engineering–Economic Systems Department, Stanford University (1971–1974) and the Electrical Engineering Department of the University of Illinois, Urbana (1974–1979). He is currently Professor of Electrical Engineering and Computer Science at the Massachusetts Institute of Technology. He consults regularly with private industry and has held editorial positions in several journals. He has done research in the areas of estimation and control of stochastic systems, linear, nonlinear and dynamic programming, data communication networks, and parallel and distributed computation, and has written numerous papers in each of these areas. He is the author of *Dynamic Programming and Stochastic Control* (New York: Academic, 1976), *Constrained Optimization and Lagrange Multiplier Methods* (New York: Academic, 1982), *Dynamic Programming: Deterministic and Stochastic Models* (Englewood Cliffs, NJ: Prentice-Hall, 1987), *Linear Network Optimization: Algorithms and Codes* (Cambridge, MA: M.I.T. Press, 1991); and co-author of *Stochastic Optimal Control: The Discrete-Time Case* (New York: Academic, 1978), *Data Networks,* 1987, and *Parallel and Distributed Computation: Numerical Methods* (Englewood Cliffs, NJ: Prentice-Hall), 1989.