DELAY ANALYSIS OF THE N-CUBE NETWORK

*Mokhtar A. Aboelaze**

Computer Science Dept.
York University
N. York, Ontario M3J 1P3 Canada

*Catherine E. Houstis***

Computer Science Dept.
University of Crete
Heraklion, Crete Greece

Abstract

In this paper, we analyze the delay of an average message going through an arbitrary link of the N-cube. We view each link as an M/M/1 queue and find analytic recursive relations for the arrival rate of messages at an arbitrary link. Then, we calculate the delay per link as a function of the message generation rate at the processor. We investigate two model of communication. The first, uniform communication where each processor communicate with any other processor with the same probability. The second is clustered communication, where neighboring processors communicate more than distant processors do. Finally, we investigate the effect of adding one more link at each node of the cube (*Folded Hypercube*) on the delay and the maximum number of hops.

1. INTRODUCTION

The N-cube multiprocessor is a highly parallel multiprocessor architecture consisting of 2^N identical processors. Each processor has its own memory and is connected to N neighbors in the form of a binary N-cube network. The hypercube is a message-passing multiprocessor architecture that has the ability to exploit particular topologies of problems in order to minimize communication cost [Fox85], [Ncu86].

In this paper, we report on the performance of the N-cube from the point of view of the communication delay incurred by an average message crossing an arbitrary link of the machine. The analysis is based on simple probabilistic relations and the inherent symmetry of the N-cube. Analytical recursive relations have been obtained for the message rate of an arbitrary link. Then each link is treated as an M/M/1 queue and a number of performance measures are computed. We also investigate the architecture known as *Folded Hypercube* [LaE89].

Previous work in performance of hypercube [AbP89] concentrated a synchronized system (SIMD) [KuS82], [Bat80]. However, the situation is different in MIMD systems. Where there are N processors, controlled by a single operating system, which provide interaction between processors and their programs [HwB84]. In this case there is no global clock controlling the different processors in the systems, and previous results will not be valid any more.

---

*This work was partially supported by a grant from the Engineering Council of Canada grant number NSERC-OGP0043688.

**This work was completed while the author was with the EE Department of Purdue University. It was supported by NSF grant number DMC-8508685A1.

In Section 2 some basic properties of the N-cube are summarized and a theorem is proved which is used in the delay analysis. In Section 3 the delay analysis is presented. In Section 4 the Folded Hypercube is analyzed and its performance is compared to the N-cube.

2. BASIC PROPERTIES OF THE N-CUBE

The N-cube, is a graph with 2^N nodes numbered from 0 to 2^N-1. A link connects two nodes if the binary representation of their two number differ by one and only one bit. The distance between two nodes a and b is $h(a,b)$, where h is the hamming distance.
Let us consider for a moment a link ℓ by its two incident nodes (c,d). Then
Definition : The distance d_1 between a node a and a link ℓ with incident nodes (c,d) is

$$d_1 = \min\left\{ h(a,c)\,,\, h(a,d) \right\}$$

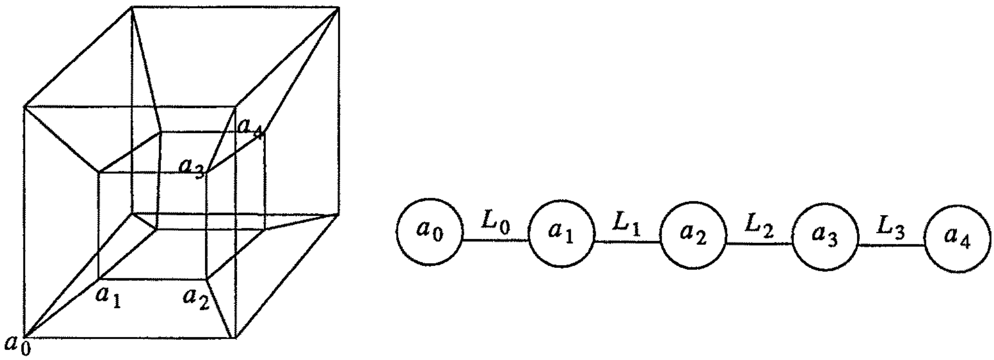
The above definition is used in the following theorem.
Theorem 1.
In an N-cube the number of nodes at distance f from a link ℓ with incident nodes (c,d), is equal to $\binom{N}{f} - \binom{N-1}{f-1}$ $\quad 0 \le f < N$.

Proof: The number of nodes at a hamming distance f from node c is equal to $\binom{N}{f}$. Not all of these nodes are at a distance f from the link (c,d). Since there is one bit in the binary representation of node c that is different from its corresponding bit in the binary representation of node d. Then, some of these nodes are at a distance $f+1$ from node d, and some are at a distance $f-1$ from it. In the first group, the nodes that are at a distance $f+1$ from node d, according to definition 3.2 are at a distance f from link (c,d). The rest of the nodes are at a distance $f-1$ from the same link. The question that remains to be answered is how many nodes are at a distance f from node c and at a distance $f-1$ from node d. Assume that the binary representation of node c is $c_0, c_1, \ldots, c_N$, and that of node d is $d_0, d_1 \ldots, d_N$. Since these two nodes differ in one and only one bit, assume without loss of generality that they differ in the first bit i.e. $c_0 \neq d_0$, and they agree in the rest of the bits. Any node at a distance f from node c and a distance $f-1$ from node d must have the same first bit as node d and differ in $f-1$ of the remaining $N-1$ bits. The number of such nodes is $\binom{N-1}{f-1}$. Therefore the number of nodes at a distance f from (c,d) is $\binom{N}{f} - \binom{N-1}{f-1}$ $\square$

3. DELAY ANALYSIS OF THE N-CUBE

In our analysis we represent the links of the N-cube as servers in a queueing system [All78], [Kle76a], [Kle76b]. The service rate of each link is its bandwidth. Communication between nodes is accomplished by exchanging messages. We assume that the time between two message generated by the same processor is exponentially distributed with mean $1/\lambda$. We also assume that the length of the message has an exponential distribution with mean $1/\mu$.

Without loss of generality let us assume that a source node is a_0 and that it sends a message to any node with equal probability. Starting at a_0 a message travels through a number of links. We pick an arbitrary link in the path from a_0 to a_N. Let L_i be this arbitrary link. The longest path is composed of a succession of $N+1$ nodes and N links, i.e., a_0, L_0, a_1, $L_1, \ldots, a_i, L_i, \ldots, a_{N-1}, L_{N-1}, a_N$. Notice that the numbering here does not correspond to the binary numbering of the nodes in the N-cube, it simply means that a_0 is the first node in the path, L_0 is the first link in the path a_1 the second node in the path, etc. Figure 1.a shows a 4-cube, and Figure 1.b shows the longest path in the 4-cube



(a) (b)

Figure 1. A 3-D view of the 4-cube

Next we proceed to define a number of terms pertinent to our analysis. Let

$p_i = prob(a_i / L_0 \cap L_1 \cap L_2 \cap, \ldots, \cap L_{i-1})$ = the probability that a message originated at a_0 is destined to node a_i given that it has traveled along the line containing the links L_0, L_1, $L_2, \ldots, L_{i-1}$.

$q_i = prob(L_i / L_0 \cap L_1 \cap L_2 \cap, \ldots, \cap L_{i-1})$ = the probability that a message originated at a_0 will travel along link L_i given it has traveled along the path containing the links L_0, $L_1, \ldots, L_{i-1}$.

P_i = probability that a message originated at node a_0 will travel along the link L_i, for $0 \le i \le N-1$

Any message generated at node a_0 will be directed with equal probability to any node in the N-cube, by following any outgoing link of a_0. Since there are N links connected to node a_0 then the probability of this message to choose link L_0 is $1/N$. Thus

$$q_0 = prob(L_0) = 1/N \tag{1}$$

Now $p_1 = prob(a_1/L_0)$ is the probability that this message is destined to a_1 given that it has traveled along link L_0. Thus

$$p_1 = prob(a_1/L_0) = \frac{prob(a_1 \cap L_0)}{prob(L_0)} \tag{2}$$

Since there is only one route from node a_0 to its neighbor a_1 then $prob(a_1 \cap L_0) = prob(a_1) = 1/(2^N - 1)$ (all the $2^N - 1$ nodes are equally likely to receive a message from a_0)

$$p_1 = \frac{1/(2^N - 1)}{1/N} = \frac{N}{2^N - 1} \tag{3}$$

Next, we calculate q_1 which is the probability that a message originated at node a_0 will travel along the link L_1, given it has already traveled along the link L_0. This can occur only if the message destination is not a_1. The probability that the message destination is not a_1 is $1-p_1$. Notice also that a message leaving a_1 has $N-1$ nodes to choose from (it can not be forwarded to the link it just arrived from).

$$q_1 = prob(L_1/L_0) = \frac{1-p_1}{N-1} \tag{4}$$

We now need to calculate $p_2 = prob(a_2/L_0 \cap L_1)$ which is the probability that the message originated at a_0 is designated to node a_2 given that it has traveled along the line containing the link L_0 and L_1. Thus

$$p_2 = prob(a_2/L_0 \cap L_1) = \frac{prob(a_2 \cap L_0 \cap L_1)}{prob(L_0 \cap L_1)} = \frac{prob(a_2 \cap L_0 \cap L_1)}{prob(L_0)prob(L_1/L_0)} \tag{5}$$

Since node a_2 is at a hamming distance 2 from node a_0, there are 2! possible paths between the two nodes. Thus the probability to go to a_2 using one such path and in particular the path containing L_0, L_1 is $prob(a_2 \cap L_0 \cap L_1) = \frac{1}{2!(2^N-1)}$. Then

$$p_2 = \frac{\frac{1}{2!(2^N-1)}}{q_0 q_1} \tag{6}$$

In order to calculate q_2 we apply similar thinking as for q_1. Thus, $1-p_2$ is the probability that the message will not have node a_2 as its destination node and since there are $N-2$ paths outbound from a_2, then

$$q_2 = prob(L_2/L_0 \cap L_1) = \frac{1-p_2}{N-2} \tag{7}$$

In general, to calculate p_i

$$p_i = prob(a_i/L_0 \cap L_1 \cap, \ldots, \cap L_{i-1}) = \frac{prob(a_i \cap L_0 \cap L_1 \cap, \ldots, \cap L_{i-1})}{prob(L_0 \cap L_1 \cap, \ldots, \cap L_{i-1})}$$

$$= \frac{prob(a_i \cap L_0 \cap L_1 \cap, \ldots, \cap L_{i-1})}{prob(L_{i-1}/L_{i-2} \cap, \ldots, \cap L_0) prob(L_{i-2}/L_{i-3} \cap, \ldots, \cap L_0), \ldots, prob(L_0)} \tag{8}$$

Note that we apply the multiplication rule for the probability function in the denominator. Then equation (8) can be rewritten as

$$p_i = \frac{\frac{1}{i!(2^N-1)}}{q_{i-1} q_{i-2}, \ldots, q_0} \tag{9}$$

and

$$q_i = prob(L_i/L_0 \cap L_1 \cap, \ldots, \cap L_{i-1}) = \frac{1-p_i}{N-i} \tag{10}$$

This completes the first step in our analysis.

In the second step, we calculate the probability P_i which is the probability that a message originated at node a_0 will travel along link L_i. Since $q_i = prob\,(L_i / L_0 \cap L_1, \ldots, \cap L_{i-1})$ is the conditional probability that this message will travel along the specific path containing $L_0, L_1, \ldots, L_{i-1}$ one can use the total probability formula and find P_i by summing overall possible paths from a_0 to a_i through L_i. This formula is

$$P(A) = \sum_{all\ i} P(A/B_i) P(B_i) = \sum_{all\ i} P(A \cap B_i) \tag{11}$$

where the events A and B_i correspond to the events

$$A/B_i = a\ message\ traveling\ along\ L_i / has\ crossed\ all\ links\ L_0 \cap L_1 \cap, \ldots, \cap L_{i-1}$$

Because of the symmetry of the N-cube there are $i!$ such paths between a_0 and a_i, all with the same probability. Thus,

$$P_i = \sum prob\,[L_i \cap \{L_{i-1} \ldots L_0\}] \tag{12}$$

the sum is over all the possible paths $\{L_{i-1} \ldots L_0\}$. since there are $i\,!$ such paths between L_0 and L_{i-1}, then

$$P_i = prob\,[L_i \cap \{L_{i-1} \ldots L_0\}]\; i\,! \tag{13}$$

$$P_i = \left[prob\,(L_i / L_{i-1} \cap, \ldots, \cap L_0) \times prob\,(L_{i-1} \cap, \ldots, \cap L_0) \right] i\,! \tag{14}$$

Using the multiplication rule we obtain

$$P_i = \left[q_i \times prob\,(L_{i-1} / L_{i-2} \cap, \ldots, L_0) \times prob\,(L_{i-2} \cap, \ldots, \cap L_0) \right]\; i\,! \tag{15}$$

and finally

$$P_i = [q_i\; q_{i-1}, \ldots, q_0]\; i\,! \;\text{ for }\; 0 \le i < N \tag{16}$$

Thus we have determined the probability that a message originated at node a_0 will travel along link L_i, in other words we have determined the percentage of messages originated at a_0 that will travel across link L_i for $0 \le i \le N-1$.

In the third step of our analysis, we have to determine the contribution of all nodes to L_i, i.e., the total message traffic through L_i. The contribution of all nodes to link L_i will depend on their distance from it. Because of the symmetry of the N-cube all nodes at the same distance, say distance j, $1 \le j < N$, will contribute the same amount of traffic. We need to know how many such nodes exist at distance j from link L_i. According to theorem 1 the number of nodes that are at a distance j from L_i are $\binom{N}{j} - \binom{N-1}{j-1}$. then,

$$S = \sum_{j=0}^{N-1} P_j \left[\binom{N}{j} - \binom{N-1}{j-1} \right] \tag{17}$$

$\lambda_i = \lambda_p \times S$, where λ_i is the effective arrival rate at any link, and λ_p is the message generating rate at the processor.

By treating any link as an M/M/1 queue [All78], the queueing delay T_q, and the total delay T_{total} are

$$T_q = \frac{\rho}{\mu(1-\rho)}, \quad T_{total} = \frac{1}{\mu(1-\rho)} \quad , \quad \rho = \frac{\lambda_i}{\mu} = \frac{\lambda_p\, S}{\mu} \tag{18}$$

This completes our analysis under the assumption that a message is sent from any node to any other node with equal probability.

Now we consider the assumption that a node sends a message to any node with a probability which depends on the hamming distance between the binary labels of the two nodes. Let $f(i)$ be the probability that a message generated at any node will be directed to a node at a distance i from the originating node. Notice that $f(i)$ should be a decreasing function of i, i.e., the further the two processors, the less probable that they will exchange messages. We assume that $f(i)$ will take the form,. $f(i) = k/i$ for $i = 1,2, \ldots, N$ where k is a constant, such that the sum of the probabilities that a message will be directed to any node is 1. The analysis is as before, except for the calculation of p_i. Since the probability that the message will be directed to a link at a distance i is $f(i)$, then p_i is given by

$$p_i = \frac{f(i) / i!}{q_0 q_1, \ldots, q_{i-1}} \tag{19}$$

The rest of the equations remain the same. This completes our analysis.

To check the validity of our analytical work, we simulated the N-cube and compared our results with the simulation results. The difference between the analytical results and simulation results is less than 7%. Recall, λ_p represents the rate of message generation at any node. Figure 2 shows the delay per link vs λ_p for both the equal probability case and the $f(i) = k/i$ case.

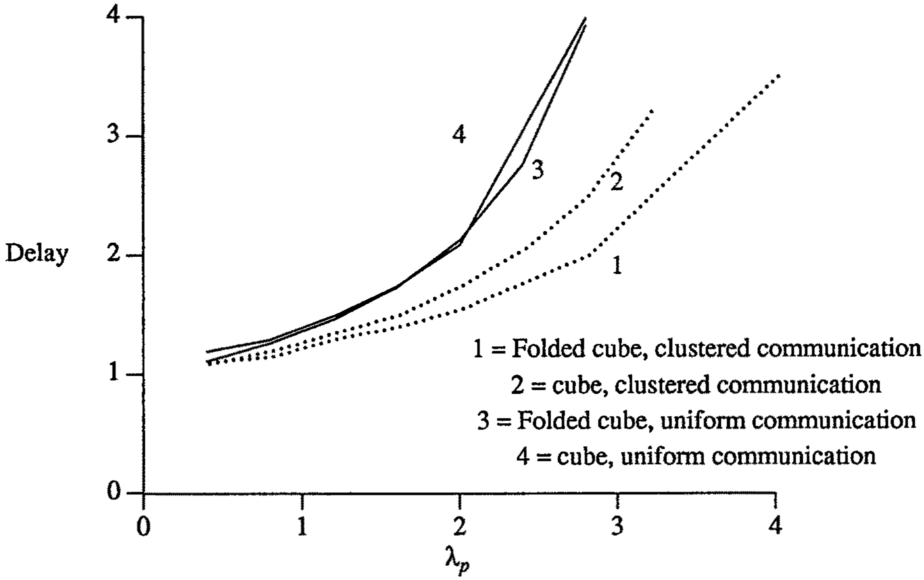


Figure 3
Delay vs λ_p for cube and folded hypercube

4. FOLDED HYPERCUBE

In N-cube the maximum distance between two nodes is N. A modified cube architecture known as *Folded Hypercube* was presented in [LaE89]. The basic differences between this modified cube and the ordinary hypercube architecture is the addition of some extra links known as the complementary links. To derive the folded hypercube from the ordinary N-cube, it is sufficient to add N bi-directional links (known as complementary links) to connect two nodes that differ in all their bits, i.e., a link is added to connect node $(a_{N-1}, a_{N-2}, \ldots, a_0)$ and node $(\bar{a}_{N-1}, \bar{a}_{N-2}, \ldots, \bar{a}_0)$. The number of the added links is N. In the rest of this section we study the effect of the added links on the average message delay.

4.1. Delay Analysis

The basic idea of routing in the modified cube is as follows. Assume that a message is directed from node a to node b, recall that the hamming distance between a and b is $h(a,b)$. Then, if $h(a,b) \leq \lceil N/2 \rceil$ then the ordinary routing is used. If $h(a,b) > \lceil N/2 \rceil$ then the message is sent to the node that differs in all the bits in its binary representation from the source node via the complementary link, and then ordinary routing is used. Thus, the maximum number of steps to send messages between any two nodes is $\lceil N/2 \rceil$.

To calculate the delay at each link, it will be very hard to follow the individual messages from node to node as we did with the ordinary cube, henceforth a simpler approach is used. Since the added link is used once at the beginning of the routing if the hamming distance is greater than $\lceil N/2 \rceil$ and never used again we can calculate the effective arrival rate for the ordinary links and the added links separately as follows. If we define the arrival rate at the ordinary links as $\lambda_{ordinary}$, then.

$$\lambda_{ordinary} = \lambda_p \times \frac{number\ of\ nodes \times Average\ number\ of\ hops\ per\ message}{Total\ number\ of\ links} \qquad (20)$$

Notice that, the nodes that are at a distance of $i \leq \lceil N/2 \rceil$ from the source node, require i hops. However, the nodes that are at a distance of $i \geq \lceil N/2 \rceil$ from the source node require one hop via the complementary link and $N-i$ hops via the ordinary links to reach its destination

$$Average\ number\ of\ hops = \frac{\sum_{i=1}^{\lceil N/2 \rceil} i \times \binom{N}{i} + \sum_{i=\lceil N/2 \rceil +1}^{N} (N-i) \times \binom{N}{i}}{\sum_{i=1}^{\lceil N/2 \rceil} \binom{N}{i} + \sum_{i=\lceil N/2 \rceil +1}^{N} \binom{N}{i}} \qquad (21)$$

$$\lambda_{ordinary} = \lambda_p \frac{\left[\sum_{i=1}^{\lceil N/2 \rceil} i \times \binom{N}{i} + \sum_{i=\lceil N/2 \rceil +1}^{N} (N-i) \times \binom{N}{i} \right] \times 2^N}{\left[\sum_{i=1}^{\lceil N/2 \rceil} \binom{N}{i} + \sum_{i=\lceil N/2 \rceil +1}^{N} \binom{N}{i} \right] \times N 2^N} \qquad (22)$$

For the complementary link, the traffic that crosses it is the part of the traffic directed to a node at distance greater that $\lceil N/2 \rceil$

$$\lambda_c = \frac{\lambda_p}{2^N - 1} \times \sum_{i=\lceil N/2 \rceil +1}^{N} \binom{N}{i} \qquad (23)$$

Table 1 shows the relation between $\lambda_{ordinary}$ (the arrival rate at the ordinary links in the Folded hypercube), λ_c (arrival rate at the complementary link for the folded hypercube), and λ_i (arrival rate for links in the hypercube) for different values of N. Figure 2 shows the delay vs λ_p for the folded hypercube.

N	$\lambda_{ordinary}$	λ_c	λ_i
2	0.500	0.333	0.667
3	0.500	0.143	0.571
4	0.357	0.333	0.533
5	0.400	0.194	0.516
6	0.355	0.349	0.508
7	0.389	0.228	0.504
8	0.366	0.365	0.501
9	0.392	0.254	0.501
10	0.378	0.377	0.500

Table 1. $\lambda_{ordinary}$, λ_c, and λ_i ($\lambda_p = 1$)

5. CONCLUSION

In this paper, we studied the average message delay per link in an N-cube. We assumed that the system is running asynchronously, the rate of message generation in each processor is Poisson and the length of the message is exponentially distributed. We established recursive relations for the rate of message arrival at each link. Knowing that, we calculated some performance measures such as the average message delay per link and the average queue length at each link. We also studied the delay in the Folded Hypercube, and we showed that the average message delay in the Folded Hypercube is much less than the average message delay in the ordinary cube. Simulation results were shown to validate our analysis, the difference between the simulation results and our analysis is less than 7%.

6. REFERENCES

[AbP89] S. Abraham and K. Padmanabhan, "Performance of the Direct Binary n-Cube Network for Multiprocessors," *IEEE Transactions on Computers,* Vol. 38, No. 7, July 1989, pp. 1000-1011.

[All78] A. O. Allen, *Probability, Statistics, and Queueing Theory,* Academic Press, 1978.

[Bat80] K. E. Batcher, "Design of a Massively Parallel Processor," *Trans. on Computers,* Vol. C-29,N 9, Sept. 1980, pp. 836-840.

[Fox85] G. Fox, *The Performance of the Caltech Hypercube in Scientific Calculations,* Caltech Report CALT-68-1298, Caltech, 1985.

[HwB84] K. Hwang and F. A. Briggs, *Computer Architecture and Parallel Processing,* McGraw-Hill, 1984.

[Kle76a] L. Kleinrock, *Queueing Systems, Volume 1: Theory,* John Wiley and Sons, 1976.

[Kle76b] L. Kleinrock, *Queueing Systems, Volume 2: Computer Applications,* John Wiley and Sons, 1976.

[KuS82] D. J. Kuck and R. A. Stokes, "The Burroughs Scientific Processor (BSP)," *IEEE Transactions on Computers,* Vol. C-31, May 1982, pp. 363-376.

[LaE89] S. Latif and A. El-Amawy, "On Folded Hypercubes," *Proc. of International Conference on Parallel Processing,* 1989.

[Ncu86] NCUBE Corp., *NCUBE Handbook, version 1.0,* NCUBE Corp., Beaverton, Oregon, 1986.