Computer Science Journal of Moldova, vol.7, no.2(20), 1999

About message routing in different hypercube interconnection network types

M. Popa, M. Stratulat

Abstract

The paper treats the problem of message routing in different hypercube interconnection network types. Because the communication algorithms frequently use a few basic communication operations, the purpose was to optain relationships for the total communication time at the implementation of these basic operations in different hypercube interconnection types. The basic communication operations considered were: simple message transfer between two processors, one to all broadcast, all to all broadcast, one to all personalized communication, and all to all personalized communication. For establishing the desired relationships, the starting point were the relationships for the total communication time for the above mentioned operations implemented on three basic interconnection networks: classical hypercube, ring and mesh.

The different hypercube interconnection network types considered were: the cube connected cycles network, the extended hypercube, the hypernet network, the k array n hypercube and the composed hypercube.

The obtained relationships are useful to establish the performances of the considered networks, from the total communication time point of view, making comparisons between them and between them and the classical hypercube interconnection network with the same number of nodes. The most advantageous interconnection network from the above mentioned point of view, is the composed hypercube with the dynamic position of the nodes.

1 Introduction

The importance of parallel computing became obviuos during the last period both because the complexity of the applications and the demands of performance imposed to the processors from the serial systems. By connecting together more processors, in a parallel system, the limits of the serial systems were exceeded.

The Amdahl's law, [1]-[3], shows that there is a limit in increasing the performances of parallel systems, especially the speedup, when the number of processors increases. It means that it is useless to increase the number of processors above a limit, which is fixed by the range of applications for which the parallel system was projected. If the number of processors exceeds a certain limit, the performances decrease. The cause is the communication between the processors which has a special importance in establishing the performances of a parallel system.

The communication between the processors is ensured by interconnection networks and by specific components of parallel algorithms.

There exists a variety of interconnection networks whose performances are measured by specific parameters, [2]-[4]. One of this parameters is the diameter, defined as the maximum of distances between any two nodes of the network. The distance between two nodes was defined as the maximum number of links between the adiacent nodes founded on the way between the two nodes. Another parameter is the degree of the network, defined as the maximum of the degrees of the nodes of the network. The degree of a node is defined as the number of links (channels) incident to it. The cost is the product of the diameter and the degree and is another important parameter.

A frequently used interconnection network because both its low cost, considering the number of nodes, and other characteristics: easy message routing, flexibility, fault tolerance, is the hypercube interconnection network, [2]-[4]. The cost of this network is a $log_2 n$ function, n being the number of nodes. Although the increase of the number of nodes does not determine a patratic increase of the cost and not even a liniar one, yet the increase of the cost is important if the number of nodes increases significantly.

Many variants of the hypercube interconnection network type were developed, the purpose being that of obtaining a moderate increase of the cost when the number of nodes increases significantly. The most performant variants from the cost point of view, are described in [5]-[10]. A short presentation of other hypercube interconnection network types is given in [5] and [6], where the author describes a new hypercube interconnection network type.

The other component which ensures the communication between the processors consists of the parallel algorithms. There are a few basic communication operations frequently used as basic blocks in a variety of parallel algorithms. The efficient implementation of them on interconnection networks has a special importance in establishing the performances of the parallel system. These operations are: simple message transfer between two processors, one to all broadcast, all to all broadcast, one to all personalized communication, and all to all personalized communication. The most important parameter which defines the quality of the implementation of these operations is the time taken to communicate a message through the network, called the communication latency, [1].

This paper describes the implementation of basic communication operations on different hypercube interconnection network types, including the original one described by the author in [5] and [6]. The next paragraph presents the components of the communication latency and the following paragraphs presents the implementation of the basic communication operations in the cube connected cycles network, the extended hypercube, the hypernet network, the k array n hypercube and the composed hypercube network. The store and forward routing was used.

2 The Communication Latency

It is defined, [1],[4], as the time taken to pass a message from the source processor to the destination processor. Its components are:

a. The startup time (t_s): it is the time required to prepare the message at the source for the transfer. It means the time for adding a header to the message, for correcting possibly errors and for executing the emitting routine. It is considered only once for a single message transfer.

b. The per hop time (t_h) : it is defined as the time needed by the header of a message to pass from a processor to another, the processors being directly connected. This component is determined by the node latency.

c. The per word transfer time (t_w): it is defined with the help of the channel bandwidth. If this is t words/second, then each word needs the time $t_w = \frac{1}{t}$ for traversing the link.

The communication latency is determined by many factors, the most important being the network topology and the switching technique. Next, only the store and forward switching technique will be used.

It is shown, [1], that the times required for transfer in three basic interconnection networks: hypercube, ring and mesh for the above mentioned operations, have the following upper bounds:

- for a hypercube interconnection network with p processors:
  - $t_s + t_w * m * logp$ for simple message transfer between two processors;
  - $(t_s + t_w * m) * logp$ for one to all broadcast;
  - $t_s * logp + t_w * m * (p-1)$ for all to all broadcast and one to all personalized communication;
  - $(t_s + \frac{1}{2}t_w * m * p) * logp$ for all to all personalized communication.

- for a ring interconnection network with p processors:
  - $t_s + t_w * m * [\frac{p}{2}]$ for simple message transfer between two processors ;

- $(t_s + t_w * m)[\frac{p}{2}]$ for one to all broadcast;
- $(t_s + t_w * m)(p - 1)$ for all to all broadcast and one to all personalized communication;
- $(t_s + \frac{1}{2} * t_w * m * p)(p - 1)$ for all to all personalized communication.

• for a mesh interconnection network with p*p processors:

- $t_s + t_w * m * 2 * [\frac{p}{2}]$ for simple message transfer between two processors;
- $2 * (t_s + t_w * m) * [\frac{p}{2}]$ for one to all broadcast;
- $2 * t_s * (p - 1) + t_w * m * (p^2 - 1)$ for all to all broadcast and one to all personalized communication;
- $(2 * t_s + t_w * m * p^2)(p - 1)$ for all to all personalized communication,

m is the number of words in the message.

The routing algorithms are presented in [1]. Each processor can communicate on only one of its ports in a step.

3 The Message Routing in the Cube-connected Cycles Network

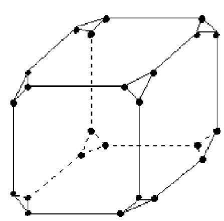

Fig.1 A cube connected cycles network

This network was presented in many papers, [7] being one of them. Every node from the hypercube network is replaced by a ring with as many nodes as the dimension of the hypercube. Fig. 1 presents such a network with an eight node hypercube and a three nodes ring.

Simple message transfer between two processes: since both the source node and the destination node are in ring interconnection networks and the message must travel, in the most unfavourable case, m rings, $m = logn$, n = the number of nodes in the hypercube network, the upper bound for the communication time is:

$$t_s + t_w * m * [\frac{p}{2}] * logn$$

p=log n being the number of nodes in the ring network.

One to all broadcast: in a hypercube network the transfer requires logn steps and each step requires the time $t_s + t_w * m$. In a ring network the transfer requires $[\frac{p}{2}]$ steps and each step requires the time $t_s + t_w * m$. The time for the one to all broadcast operation has three components:

- the time required for the one to all broadcast operation at the ring level which contains the source node;
- the time required for the transfer at the hypercube level and
- the time required for the transfer at all the other rings level except the ring which contains the source node.

The upper bound of the total time required for the one to all broadcast operation implemented on the cube connected cycles network is:

$$(t_s + t_w * m) * [\frac{p}{2}] + (t_s + t_w * m) * logn + (t_s + t_w * m) * [\frac{p}{2}] =$$

$$(t_s + t_w * m)(logn + 2 * [\frac{p}{2}])$$

All to all broadcast: in a ring interconnection network, the transfer requires $p - 1$ steps and each step needs the time $t_s + t_w * m$ and in a hypercube interconnection network the transfer requires $logn$ steps and each step needs the time $t_s + 2^{i-1} * t_w * m$, where $2^{i-1} * m$ is the size of the message in the i^{th} of the $logp$ steps. Then the total time will be maked up by:

- the time required for the transfer at the rings level; as a consequence each node from each ring will contain all the messages which are destined to it from the nodes situated in the same ring;
- the time required for the transfer at the hypetcube level; as a consequence every vertex will contain all the messages destined to the nodes from the ring situated in that vertex and
- the time required for the transfer at the rings level.

The upper bound of the total time required for the all to all broadcast operation is:

$$(t_s + t_w * m)(p-1) + t_s * logn + t_w * m * (n-1) + (t_s + t_w * m)(p-1) =$$
$$2 * (t_s + t_w * m)(p-1) + t_s * logn + t_w * m * (n-1).$$

One to all personalized communication: as it is shown in [1] the complexities of the one to all personalized and all to all broadcast operations are similar and as a consequence the time for the one to all personalized operation is equal with the time for the all to all broadcast operation.

All to all personalized communication: starting from the transfer times for the basic communication operations implemented on the ring and hypercube interconnection networks one can obtain the time for the all to all persinalized communication which will have the following components:

- the time necessary for the transfer at the rings level; after that time each node will contain all the messages which correspond to it from the nodes from the same ring;
- the time necessary for the transfer at the hypercube level; after that time every vertex will contain all the messages destinated to the ring situated in that vertex but the messages will be distributed at different nodes, that is the situation is not that in which each node from each ring contains all the messages which correspond to it but only that in which all the messages which correspond to a ring have reached it and

- the time necessary for the transfer at the rings level; after that time each node will contain all the messages which correspond to it.

The upper time for the all to all personalized communication operation is:

$$(t_s+\frac{1}{2}*t_w*m*p)(p-1)+(t_s+\frac{1}{2}*t_w*m*n)logn+(t_s+\frac{1}{2}t_w*m*p)*(p-1) =$$

$$2*(t_s+\frac{1}{2}t_w*m*p)(p-1)+(t_s+\frac{1}{2}*t_w*m*n)*logn.$$

4 The Message Routing in the Extended Hypercube Network

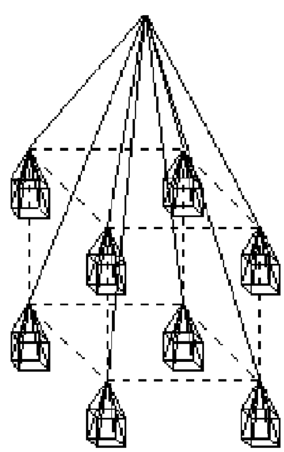

Fig. 2 A two level extended hypercube

The extended hypercube is a hierarchic interconnection network based on the hypercube network with small diameter and constant degree, no matter which the number of nodes is, [8]. Fig. 2 presents a two level extended hypercube, with 64 nodes.

The basic module of this network is a k dimensional hypercube and a control node. By connecting 2^k control nodes as a k dimensional hypercube it results an extended hypercube with 2^{k*t} nodes, where t is the number of levels and a control node. The method may continue resulting extended hypercubes with more levels.

Simple message transfer between two nodes: the time required for the transfer will have three components:

- the time necessary for the transfer between the source node and the control node of the hypercube which contains the source node;
- the time necessary for the transfer between two control nodes founded in the vertex of a hypercube;
- the time necessary for the transfer between the control node of the hypercube which contains the destination node and this one.

The upper bound of the total time needed for a simple message transfer between two nodes in a two level extended hypercube is:

$$t_s + m * t_w + m * t_w * k + m * t_w = t_s + 2 * m * t_w + m * t_w * \frac{logp}{2} =$$

$$t_s + m * t_w * (2 + \frac{logp}{2})$$

where p is the number of nodes (64 in this case).

One to all broadcast: the time needed for the transfer will have three components:

- the time necessary for the transfer at the hypercube level which contains the source node;
- the time necesary for the transfer at the basic hypercube interconnection network level and

- the time necesary for the transfer at hypercubes levels which are in the vertexes of the basic hypercube, others than the one which contains the source node.

The first two components are overlapped in time. The transfer at the hypercube level which contains the source node rerquires k steps. If the routing algorithm foresees that the message reaches at the control node in the first step and only after begins the transfer one to all in the hypercube which contains the source node, the required step for this transfer will be overlapped with the steps of the transfer in the basic hypercube. Also the last component will need a supplementary step, the one in which the message reaches a node of the hypercubes from the control node.

The time for the one to all broadcast on a two level extended hypercube network will have the following upper bound:

$$t_s+m*t_w+(t_s+t_w*m)*k+(t_s+t_w*m)*(k+1) = (t_s+m*t_w)*(2+logp),$$

p being the total number of the nodes (64 in this case).

All to all broadcast: the time required by this transfer will have three components:

- the time necessary for the transfer all to all at the hypercubes level from the vertexes of the basic hypercube;

- the time necessary for the transfer at the basic hypercube level, and

- the time necessary for the transfer of the messages from the control node to the nodes of their hypercubes; in this time the control node will send to each node of its hypercube the messages addressed to it from all other nodes from other hypercubes.

The first component will require the number of steps needed in the case of the classic hypercube and 2^k steps more, in which all the messages from the nodes of a hypercube reach the control node.

As a consequence the time required by the all to all broadcast on the two level extended hypercube will have the following upper bound:

$$t_s * k + t_w * m * (2^k - 1) + 2^k * (t_s + m * t_w) +$$

$$t_s * k + t_w * m * (2^k - 1) + (t_s + m * t_w) =$$

$$2 * (t_s * \frac{logp}{2} + \sqrt{p} * (t_s + m * t_w) + (\sqrt{p} - 1) * t_w * m),$$

p being the number on nodes of the interconnection network.

One to all personalized communication: in accordance with [1], the value obtained in the previous case is valid in this case too.

All to all personalized communication: the time necessary for the transfer will have three components:

- the time necessary for the transfer at the hypercubes level from the vertexes of the basic hypercube; after this time each node will contain all the messages addressed to it from the nodes founded in the same hypercube;
- the time necessary for the transfer at the basic hypercube level; after that time each control node will contain all the messages which correspond to its hypercube, and
- the time necessary for transfering the messages from the control node to the nodes of its hypercube.

The first component will necessitate 2^k supplementary steps, the ones in which all the messages for the other nodes reach the control node.

The last component will necessitate 2^k supplementary steps. In each step the transfer from the control node to one of the nodes of its hypercube is performed.

The time for the all to all personalized communication in a two level extended hypercube will have the following upper bound:

$$(t_s + \frac{1}{2} * t_w * m * 2^k) * k + 2^k * (t_s + m * t_w) + (t_s + \frac{1}{2} * t_w * m * 2^k) * k +$$

$$+2^k * (t_s + m * t_w) = 2 * p(t_s + m * t_w) + (t_s + \frac{1}{2} * m * t_w * \sqrt{p}) * (\frac{logp}{2}),$$

p being the total number of nodes.

5 The Message Routing in the Hypernet Interconnection Network

It is a hierarchic interconnection network made by hypercube type modules. Each node from a module receives a supplementary link necessary for connecting the modules between them and for obtaining a network with another level. Fig. 3 presents a two level hypernet network:

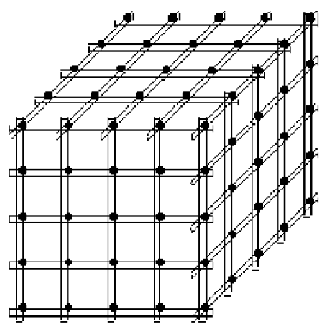

Fig. 3 A two level hypernet network

Simple message transfer between two processors: the time in which the transfer takes place has three components:

- the time necessary for the transfer at the hypercube module level which contains the source node, between this one and the node which has an extra link;

- the time necessary for the transfer between different modules, and

- the time necessary for the transfer at the module level which contains the destination node between this one and the node which establishes a connection with other modules.

Being thought that the hypercube module has q nodes and between any two modules is a direct connection, the time necessary for the transfer has the following upper bound:

$$t_s+t_w*m*logq+t_s+t_w*m+t_s+t_w*m*logq = 3*t_s+t_w*m*(1+2*log\frac{p}{4}),$$

$p = 4 * q$ being the total number of nodes.

One to all broadcast: the time necessary for the transfer has also three components:

- the time necessary for the transfer at the hypercube module level which contains the source node;
- the time necessary for the transfer at the network level which interconnects all the modules between them, and
- the time necessary for the transfer at the level of all the hypercube modules, except the one which contains the source node.

The time necessary for the one to all broadcast in the two level hypernet network will have the following upper bound:

$$(t_s + t_w * m) * logq + 2 * (t_s + t_w * m) + (t_s + t_w * m) * logq =$$
$$2 * (t_s + t_w * m) * (1 + log\frac{p}{4}),$$

where q is the number of nodes from the hypercube module and p is the total number of nodes from the network.

All to all broadcast: the time necessary for the transfer has three components:

- the time necessary for the all to all broadcast at the hypercube modules level; after that time every node from a hypercube will contain all the messages addressed to it but only from the nodes founded in the same hypercube;

- the time necessary for the transfer between the modules; as a consequence each module will contain in its nodes which connect it to other modules, the messages for the nodes founded in hypercube from the nodes founded in all other hypercubes, and
- the time necessary for the transfer at the hypercube modules level.

Being thought that the number of nodes from the hypercube module is q and and the total number of nodes is $p = 4 * q$, the time necessary for the transfer has the following upper bound:

$$t_s * logq + t_w * m * (q-1) + 2 * (t_s + m * t_w) + t_s * logq + t_w * m * (q-1) =$$
$$2 * (t_s * log\frac{p}{4} + t_w * m * (\frac{p}{4} - 1) + t_s + m * t_w).$$

One to all personalized communication: in accordance with [1], the relationship obtained at the all to all broadcast is valid in this case, too.

All to all personalized communication: using the same judgement as in the case of the all to all broadcast operation, the time necessary for the all to all personalized communication has the following upper bound:

$$(t_s + \frac{1}{2} * t_w * m * q) * logq + 2 * (t_s + m * t_w) + (t_s + \frac{1}{2} * t_w * m * q) * logq =$$
$$2 * ((t_s + \frac{1}{2} * t_w * m * \frac{p}{4}) * log\frac{p}{4} + t_s + m * t_w).$$

6 The Message Routing in the k-Array n-Cube Interconnection Network

This network is a combination betwen the hypercube interconnection network topology and the wraparound square mesh interconnection network topology, [10]. A k array n cube network interconnects $k * k$ wraparound square meshes as a n order hypercube. Fig. 4 presents the 5 array 3 cube network.

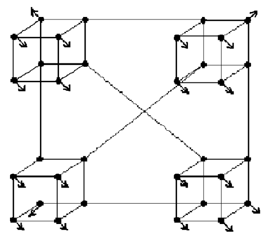

Fig. 4 The 5 array 3 cube network

Since the nodes are connected in $k*k$ wraparound square meshes, the communication times for the basic operations on this network will be used. They were presented in the second paragraph.

Simple message transfer between two processors: the transfer will be achieved in two stages: in the first stage a transfer in a $k*k$ nodes wraparound square mesh takes place and in the second stage a transfer in a k nodes ring will take place. Using the relationships which give the time for the basic communication operations in a ring interconnection network, presented in the third paragraph, it results the upper bound for the time necessary for a simple message transfer between two processors:

$$t_s + t_w * m * 2 * [\frac{k}{2}] + t_s + t_w * m * [\frac{k}{2}] = 2 * t_s + 3 * t_w * m * [\frac{k}{2}].$$

One to all broadcast: the transfer will be also in two stages: in the first one the transfer one to all at the wraparound square mesh level which contains the source node will be made and in the second one the transfer one to all at the rings which contain the nodes from the mesh which contains the source node level, will be made.

The time for the one to all broadcast operation will have the following upper bound:

$$2 * (t_s + m * t_w) * [\frac{k}{2}] + (t_s + t_w * m) * [\frac{k}{2}] =$$

$$3 * (t_s + t_w * m) * [\frac{k}{2}].$$

All to all broadcast: using the same judgement as in the case of the one to all broadcast operation , the upper bound for the time necessary for the all to all broadcast operation will be:

$$2 * t_s * (k-1) + t_w * m * (k^2 - 1) + (t_s + t_w * m) * (k-1) =$$

$$(k-1)(3 * t_s + t_w * m * k + 2 * t_w * m).$$

One to all personalized communication: in accordance with [1] the necessary time for this operation will be identically with the one for the all to all broadcast operation.

All to all personalized communication: using the same judgement as in the case of the one to all broadcast operation, the upper bound for the time necessary for the all to all personalized communication operation will be:

$$(2 * t_s + t_w * m * k^2) * (k-1) + (t_s + m * t_w) * (k-1) =$$

$$(k-1) * (3 * t_s + t_w * m + t_w * m * k^2).$$

7 The Message Routing in the Composed Hypercube Interconnection Network

The composed hypercube was described in several papers,[5] and [6] being the most significant ones.

The composed hypercube, labeled HC(n,p), is a hierarchic network and consists of a n dimensional hypercube, named superior hypercube, to which each node is replaced by another p dimensional hypercube named inferior hypercube. p and n may or may not be equal. The composed network may have more than two levels. Fig. 5 presents the composed hypercube HC(3,3).

In [5] and [6] it is shown that by adding extra hardware, which ensures that the position of the nodes from the inferior hypercubes become dynamic, one obtains a constant and and small diameter, practically the diameter of the superior hypercube no metter the number

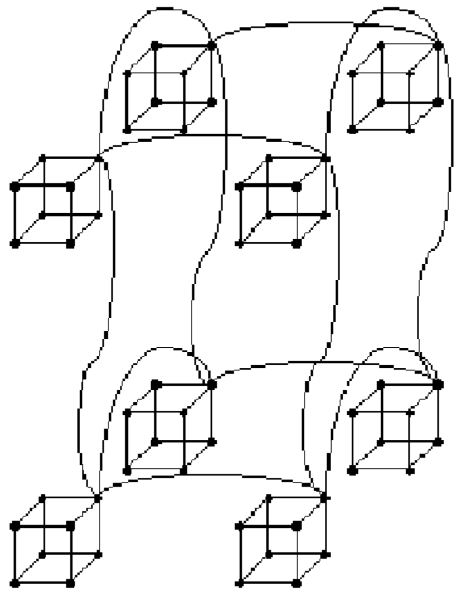

Fig. 5 The composed hypercube HC(3,3)

of nodes is. In this case, all the relationships which give the transfer time for the basic communication operations implemented on the classic hypercube network, remain valid in this case too with the observation that the number of processors considered in this case is equal only with the number of the vertexes of the superior hypercube, no metter the number of levels (that is the number of processors) is. At a rigorous calculation of the communication times, it must be also considered the time necessary for modifying the position of the nodes by the extra hardware. From [5] and [6] it results that this time consists of the delay on a combinational module so that its value is low. Even more, that time must be considered only once no metter the length of the message or the number of the messages are. In conclusion,this time can be neglected.

Next, the case in which the position of the nodes from the inferior hypercubes is static will be treated. This variant does not need extra hardware but does not ensure the performances (diameter, cost) offered by the other variant.

Simple message transfer between two processors: the time in which the transfer is performed has three components:

- the time necessary for the transfer from the source node to the common node between the inferior hypercube which contains the source node and the superior hypercube;
- the time necessary for the transfer at the superior hypercube level, between the two vertexes in which there are two inferior hypercubes which contain the source and the destination nodes, and
- the time necessary for the transfer from the common node between the superior hypercube and the inferior hypercube which contains the destination node and this one.

The time for the transfer will have the following upper bound:

$$t_s+t_w*m*logp+t_w*m*logn+t_w*m*logp = t_s+t_w*m*(2*logp+logn)$$

and if p=n then the value will be

$$t_s + 3 * t_w * m * logp.$$

One to all broadcast: the time necessary for the transfer will consist of:

- the time necessary for the one to all broadcast in the inferior hypercube which contains the source node;
- the time necessary for the transfer in the superior hypercube, and
- the time necessary for the transfer in all the inferior hypercubes, others than the one which contains the source node.

The time necessary for the one to all broadcast in a two level composed hypercube will have the following upper bound:

$$(t_s + t_w * m) * logp + (t_s + t_w * m) * logn + (t_s + t_w * m) * logp =$$

$$(t_s + t_w * m) * (2 * logp + logn)$$

and if n=p than the value will be

$$3 * logp * (t_s + t_w * m).$$

All to all broadcast: the time necessary for the transfer will have the following components:

- the time necessary for an all to all broadcast operation at the inferior hypercubes level; at the end of this time each node will contain the messages addressed to it from all the nodes from the same inferior hypercube;
- the time necessary for an all to all broadcast operation at the superior hypercube level; at the end of this time each common node will contain all the messages which correspond to the respective inferior hypercube, and
- the time for an one to all broadcast operation at the inferior hypercubes level; as a consequence each node will receive for the common node all the messages which correspond to it form the nodes founded in all other hypercubes.

The time from an all to all broadcast operation on a two level composed hypercube will have the following upper bound:

$$t_s * logp + t_w * m * (p-1) + t_s * logn + t_w * m * (n-1) + (t_s + t_w * m) * logp$$

and if p=n the value will be

$$2 * (t_s * logp + t_w * m * (p-1)) + (t_s + t_w * m) * logp.$$

One to all personalized communication: in accordance with [1], the relationship obtained above is valid in this case too.

All to all personalized communication: using the same judgement as in the all to all broadcast operation it results the upper bound for the transfer time as being:

$$(t_s+\frac{1}{2}*t_w*m*p)*logp+(t_s+\frac{1}{2}*t_w*m*n)*logn+t_s*logp+t_w*m*(p-1)$$

and if p=n, the value will be

$$2*(t_s+\frac{1}{2}*t_w*m*p)*logp+t_s*logp+t_w*m*(p-1).$$

8 Conclusions

The present paper approached some problems about message routing in some hypercube interconnection network types. A few variants of the classic hypercube interconnection network were considered which were described in other papers and which were introduced for obtaining a more moderate increase of the cost with the increase of the number of nodes than in the case of the classic hypercube. The networks considered were: the cube connected cycles network, the extended hypercube, the hypernet network, the k array n cube network and the composed hypercube.

Because of the importance of basic communication operations implemented on the above mentioned interconnection networks, relationships for the following basic operations were developped: simple message transfer between two processors, one to all broadcast, all to all broadcast, one to all personalized communication and all to all personalized communication. The relationships were developped only for the approached examples and not for the general case. The relationships which give the communication times for three classic interconnection networks: the hypercube, the ring and the mesh were used.

The relationships developped are useful in establishing the performances of the considered interconnection networks, from the communication latency point of view, both through comparisons between them and between them and the classic hypercube interconnection network with the same number of nodes. The most advantageous, from the mentioned point of view, is the composed hypercube network with the dynamic position of the nodes at which the increase of the number of nodes does not affect the communication time.

References

[1] .Kumar, A.Grama, A.Gupta, G.Karypis. Introduction to Parallel Computing. The Benjamin/Cummings Publishing Company Inc., 1994.

[2] .T.Leighton. Introduction to Parallel Algorithms and Architectures. Arrays∗ Trees∗ Hypercubes. Morgan Kaufmann Publishers Inc., San Mateo, California, 1992.

[3] .Cosnard, D.Trystram. Algorithms et architectures parelleles. Intereditions, Paris, 1993.

[4] . de Rummeur. Communications dans les reseaux de processeurs. Masson, Paris, 1994.

[5] .Popa. The Increase of the Number of Nodes Connected in a Hypercube Network. International Conference on Technical Informatics CONTI'96, Timisoara, 1996.

[6] .Popa, C.Hora. The Minimization of the Increase of the Diameter in the Hypercube Network. Buletinul Stiintific al U.P.T., Seria Automatica si Calculatoare, ISSN 1224-600X, 1997.

[7] .F.Tzeng. A Cube Connected Cycles Architecture with High Reliability and Improved Performance. IEEE Transactions on Computers, February, 1994.

[8] .M.Kumar, L.M.Patnaik. Extended Hypercube: A Hierarchical Interconnection Network of Hypercubes. IEEE Transactions on Parallel and Distributed Systems, v. 3, No. 1, January, 1992.

[9] .Hwang, J.Gosh. Hypernet: A Communication-Efficient Architecture for Constructing Massively Parralel Computers. IEEE Transactions on Computers, December, 1987.

[10] .J.Dally. Performance Analysis of k-Array n-Cube Interconnection Networks. IEEE Transaction on Computers, June, 1990.

M.Popa, M.Stratulat
Computer Science Department
"Politehnica" University of Timisoara
2 V. Parvan blvd., 1900,
Timisoara, ROMANIA
e-mail: mpopa@utt.ro
smircea@utt.ro

Received September 10, 1999