ELSEVIER

Parallel Computing 27 (2001) 1273–1278

PARALLEL COMPUTING

www.elsevier.com/locate/parco

Short communication

A general broadcasting scheme for recursive networks with complete connection

Gen-Huey Chen [a,*], Shien-Ching Hwang [a], Hui-Ling Huang [b], Ming-Yang Su [c], Dyi-Rong Duh [d]

[a] *Department of Computer Science and Information Engineering, National Taiwan University, Taipei 10764, Taiwan, ROC*

[b] *Department of Information Management, Southern Taiwan University of Technology, Tainan, Taiwan, ROC*

[c] *Department of Electronic Engineering, St. John's and St. Mary's Institute of Technology, Taipei, Taiwan, ROC*

[d] *Department of Computer Science and Information Engineering, National Chi Nan University, Nantou, Taiwan, ROC*

Received 10 June 1998; received in revised form 22 January 2001; accepted 26 January 2001

Abstract

A recursive network is said to be with complete connection if its subnetworks are connected as a complete graph. In this paper, a general broadcasting scheme is proposed for recursive networks with complete connection. The scheme is simple, efficient, and easy to be implemented. Besides, there is no redundant message generated. Broadcasting algorithms for the hypercomplete, hypernet, WK-recursive, and star networks can result from this scheme. No broadcasting algorithm for the hypernet networks was proposed before.

Keywords: All-port model; Broadcasting; Hypercomplete network; Hypernet network; Recursive networks; WK-recursive network; Star network

1. Introduction

Advances in technology, especially the advent of VLSI circuit technology, have made it possible to build a large parallel and distributed system involving hundreds

* Corresponding author. Fax: +886-2-362-8167.
E-mail address: ghchen@csie.ntu.edu.tw (G.-H. Chen).

PII: S 0 1 6 7 - 8 1 9 1 (0 1) 0 0 0 8 8 - 6

or even thousands of processors. For example, the connection machine [9] contains as many as 2^{16} processors. In the design and implementation of a parallel and distributed system, one of the most fundamental issues is how to determine the interconnection network (network for short) through which the processors can communicate efficiently. There were many networks (see [2,10]) proposed for this purpose. Throughout this paper, network and graph, processor and node, and link and edge are used, interchangeably.

A network whose topology can be defined recursively is referred to as a *recursive network*. Hypercubes [20], Fibonacci cubes [11], star networks [1], arrangement graphs [5], WK-recursive networks [17], hypercompletes [3], and hypernets [12] are some examples. Recursive networks can be extended or decomposed in a recursive manner, and they have some advantages. First, they are suitable to be manufactured by VLSI technology, because they can be constructed by grouping basic building blocks. Second, they have high fault-tolerant capability. For example, a fault-free $(n-1)$-dimensional star network can be determined from an n-dimensional star network with up to $n-1$ edge and/or node faults. Third, their recursive topologies can help the derivation of useful properties and efficient algorithms (see [6]) and the embedding of other recursive networks (see [7,14,19]).

The size of a recursive network is usually specified by its expansion level. Each basic building block is assigned with level one. A recursive network of level $i > 1$ is composed of subnetworks of level $i-1$. More specifically, the node set of the former can be partitioned into a number of subsets which each constitute a subnetwork of level $i-1$. Suppose that W is a recursive network of level n, and each subnetwork of level i in W contains f_i subnetworks of level $i-1$, where $2 \leqslant i \leqslant n$. W is said to be with *complete connection* if the f_i subnetworks of level $i-1$ that constitute a subnetwork of level i are connected as a complete graph of f_i supernode, where each subnetwork of level $i-1$ is viewed as a supernode. WK-recursive networks, hypercompletes, and hypernets, for example, are all recursive networks with complete connection.

Undoubtedly, communication is one of the most important issues for networks because communication time overweighs computation time in many situations. There are several kinds of communications performed, e.g., one-to-one, one-to-many, one-to-all, many-to-many, etc. *Broadcasting*, which belongs to one-to-all communication, requires a node (i.e., the source node) to disseminate a message to all of the other nodes. Many algorithms involve broadcasting as a basic operation. For example, broadcasting has been used in a variety of linear algebra algorithms, database queries algorithms, and transitive closure algorithms (see [13]).

In an MIMD network, the processors operate asynchronously under the control of their own control units. If too many messages are present in the network, the performance degrades very quickly. Hence the amount of messages transmitted is adopted as the major metric for evaluation with the consideration of preventing the occurrence of congestion or even deadlock in the network. The maximum length of the transmission paths is considered as the second metric. This metric represents the number of transmission steps required if the all-port model [13] is adopted. The all-port model, which allows a processor to communicate on all its ports concurrently, seems reasonable, because the time required for data replication is negligible when compared with the time required for data transmission. Another reason for adopting this metric is due to fault tolerance. A shorter transmission path has a lower probability of encountering a faulty processor or faulty link. Under the all-port model, a broadcasting algorithm is *optimal* if N $(\leqslant D)$ is minimum, *worst-case optimal* if $N = D$ in the worst case, and *asymptotically optimal* if $\mathrm{O}(N) = \mathrm{O}(D)$, where N is the number of transmission steps required and D is the diameter. We adopt the all-port model in the rest of this paper.

In this paper, a general broadcasting scheme for recursive networks with complete connection is proposed. The scheme is simple, efficient, and easy to be implemented. Besides, it generates no redundant message. Several examples are given to show the effectiveness of the scheme.

2. The scheme

Suppose that W is a recursive network of level n with complete connection. We show that by properly labeling the links of W, broadcasting on W can be performed by a stack-based method. For simplicity, we assume that the underlying graph of the basic building block is a complete graph. A link is labeled with i if it connects two subnetworks of level i that belong to the same subnetwork of level $i+1$, where $1 \leqslant i \leqslant n-1$. The links inside a basic building block are labeled with 0.

The stack-based method requires a stack of length n to be transmitted along with the message. During its execution, labels of the links are pushed (or popped) into (or from) the stack. Initially, the source node sets the stack empty and disseminates the message over all its incident links. When a node receives a copy of the message via a link of label i $(0 \leqslant i \leqslant n-1)$, it performs the following operations:

1. pop the elements of the stack until the current top element is greater than i or the stack is empty;
2. push i into the stack;
3. disseminate the message over those incident links whose labels do not appear in the stack.

By the aid of the stack, each subnetwork of level i receives exactly one copy of the message via a link of label k, where $1 \leqslant i \leqslant k < n$. When a copy of the message enters a subnetwork of level i via a link of label i, it will not be propagated out of the subnetwork via a link of label i.

Lemma 1. *Suppose that W is of level $k+1$. Any transmission path in W that starts at the source node contains at most one link of label k.*

Proof. By the stack-based method, once a label k is pushed into the stack, it will stay in the stack forever. Thus, it is impossible to transmit the message via another link of label k henceforth. □

Theorem 1. *The stack-based method can disseminate exactly one copy of the message to every node of W.*

Proof. We prove this theorem by induction. Clearly, the theorem is true if W is of level one. We thus assume that the theorem is true if W is of level $k \geqslant 1$. When W is of level $k+1$, W contains a number of node-disjoint subnetworks of level k. There is a unique link (of label k) from the source subnetwork to each of the other subnetworks, where the source subnetwork is the one which the source node belongs to. By our assumption, every node of the source subnetwork can receive exactly one copy of the message. By the stack-based method, a copy of the message will enter each of the other subnetworks via a link of label k, and by Lemma 1, it will stay there forever. Again, by our assumption, every node of these subnetworks can receive exactly one copy of the message. □

The stack-based method has the following advantages. First, it is simple and easy to be implemented in an asynchronous environment. Second, since no redundant message is generated, minimum message complexity is achieved. Third, a sequence of messages can be broadcast, in a pipeline fashion, with a maximum pipeline rate. Since the stack can be implemented as a bit array of length n, only extra n bits are required.

Although W was assumed to be with complete connection, the stack-based method remains applicable even if the f_i subnetworks of level $i-1$ that constitute a subnetwork of level i are connected as a 1-reachable graph with respect to the source subnetwork. A graph G is called 1-reachable with respect to a node $x \in G$ if every node of G can be reached from x by traversing at most one edge. We have assumed that the underlying graph of the basic building block is a complete graph. When it is not the case, the stack-based method needs a slight modification. Instead, a broadcasting algorithm for the basic building block is needed. The stack-based method, however, remains applicable outside the basic building block. Hence a hybrid one that combines the stack-based method with the broadcasting algorithm for the basic building block will result.

3. Examples and discussion

The WK-recursive networks, hypercompletes, and hypernets are three instances of recursive networks with complete connection. Each basic building block of the first two forms a complete graph. A broadcasting algorithm for the WK-recursive networks first appeared in [18] which requires extra $n \times \lceil \log_2 d \rceil$ bits to be transmitted along with the message, where n is the number of levels and d is the maximum node degree. Then, using the stack-based method, a simpler broadcasting algorithm that requires only extra $n+1$ bits was proposed in [6]. It was shown in [4] that the broadcasting algorithm is optimal. On the other hand, a worst-case optimal broadcasting algorithm for the hypercompletes that was designed using the stack-based method was proposed in [3].

There are three kinds of basic building blocks, i.e., buslet, treelet, and buslet, for constructing the hypernets. A cubelet is basically a hypercube augmented with one extra link (called external link in [12]) per node. The work of [12] was focused on cubelet-based hypernets. In [4], a broadcasting algorithm for the cubelet-based hypernets was proposed. The algorithm is a hybrid one that combines the stack-based method with a broadcasting algorithm for the hypercube. The former was applied outside the cubelets, and the latter was applied inside the cubelets. There is no redundant message generated, and the number of transmission steps required is bounded above by $2^{l-1}(d+1)-1$ (see [4]). The diameters of the cubelet-based hypernets are still unknown, and an upper bound of $2^{l-1}(d+1)-1$ was suggested in [12] for the cubelet-based hypernets of level l whose cubelets have d dimensions. No other broadcasting algorithm for the hypernets was proposed before.

The three recursive networks above have a common characteristic: the complete graphs we concerned are simple graphs. It is not always this case for other recursive networks with complete connection. For example, in the star networks, the subnetworks of level $i-1$ that constitute a subnetwork of level i are connected as a complete multigraph. A multigraph allows more than one link between two nodes [8]. The execution of the stack-based method for the situation of multigraphs is almost the same as that for the situation of simple graphs except link labeling. Although there are multiple links between two subnetworks of level i that belong to the same subnetwork of level $i+1$, at most one of them is labeled with i and the others are not labeled. A deliberate labeling is thus very necessary in order to gain efficiency.

Using the stack-based method associated with a proper labeling, a broadcasting algorithm for the star networks was proposed in [4]. Besides no redundant message generated, this algorithm requires $2n-3$ transmission steps if the star network has n dimensions. Since the diameter of an n-dimensional star network is $\lfloor 3(n-1)/2 \rfloor$ (see [1]), the algorithm is asymptotically optimal. Previously, there were two broadcasting algorithms (see [15,16]) proposed for the star networks under the all-port model which require $2n-3$ [15] and $3(n-1)$ [16] transmission steps, respectively.

The stack-based method is flexible enough to be incorporated in other applications that require broadcasting. The computations at each node are simple and the same for different source nodes. Besides, the computations are irrelevant to network topologies. Broadcasting algorithms whose computations are source node-independent and topology-independent are very suited for distributed environments.

References

[1] S.B. Akers, D. Harel, B. Krishnamurthy, The star graph: an attractive alternative to the n-cube, in: Proceedings of the International Conference on Parallel Processing, 1987, pp. 393–400.

[2] J.C. Bermond (Ed.), Interconnection Networks (Ed.), Discrete Applied Mathematics 37&38 (1992).

[3] G.H. Chen, J.S. Fu, J.F. Fang, Hypercomplete: a pancyclic recursive topology for large-scale distributed multicomputer systems, Networks 35 (1) (2000) 56–69.

[4] G.H. Chen, S.C. Hwang, H.L. Huang, M.Y. Su, D.R. Duh, A general broadcasting scheme on recursive networks and its applications, Technical Report NTUCSIE 01-01, Department of Computer Science and Information Engineering, National Taiwan University, Taipei, Taiwan, 2001.
[5] K. Day, A. Tripathi, Arrangement graphs: a class of generalized star graphs, Information Processing Letters 42 (5) (1992) 235–241.
[6] D.R. Duh, On the study of WK-recursive networks, Ph.D. dissertation, Department of Computer Science and Information Engineering, National Taiwan University, Taipei, Taiwan, May 1994.
[7] K. Efe, Embedding mesh of trees in the hypercube, Journal of Parallel and Distributed Computing 11 (3) (1991) 222–230.
[8] F. Harary, Graph Theory, Addison-Wesley, Reading, MA, 1969.
[9] W.D. Hillis, The Connection Machine, MIT Press, Cambridge, MA, 1985.
[10] D.F. Hsu (Ed.), Interconnection Networks and Algorithms (Ed.), Networks 23 (4) (1993).
[11] W.J. Hsu, Fibonacci cubes: a new interconnection topology, IEEE Transactions on Parallel and Distributed Systems 4 (1) (1993) 3–12.
[12] K. Hwang, J. Ghosh, Hypernet: a communication-efficient architecture for constructing massively parallel computers, IEEE Transactions on Computers C-36 (12) (1987) 1450–1466.
[13] S.L. Johnsson, C.T. Ho, Optimum broadcasting and personalized communication in hypercubes, IEEE Transactions on Computers C-38 (9) (1989) 1249–1268.
[14] X. Shen, Q. Hu, W. Liang, Embedding k-ary complete trees into hypercubes, Journal of Parallel and Distributed Computing 24 (1995) 100–106.
[15] J.P. Sheu, W.H. Liaw, T.S. Chen, A broadcasting algorithm in star graph interconnection networks, Information Processing Letters 48 (1993) 237–241.
[16] Y.C. Tseng, J.P. Sheu, Toward optimal broadcasting in a star graph using multiple spanning trees, IEEE Transactions on Computers 46 (5) (1997) 593–599.
[17] G.D. Vecchia, C. Sanges, A recursively scalable network VLSI implementation, Future Generation Computer Systems 4 (3) (1988) 235–243.
[18] G.D. Vecchia, C. Sanges, An optimized broadcasting technique for WK-recursive topologies, Future Generation Computer Systems 5 (4) (1989/1990) 353–357.
[19] A.Y. Wu, Embedding of tree networks into hypercubes, Journal of Parallel and Distributed Computing 2 (1985) 238–249.
[20] Y. Saad, M.H. Schultz, Topological properties of hypercubes, IEEE Transactions on Computers 37 (7) (1988) 867–872.