International Journal of Multimedia and Ubiquitous Engineering
Vol. 13, No. 3 (2018), pp.1-6
http://dx.doi.org/10.21742/ijmue.2018.13.3.01

Embedding of Fibonacci Tree into Hyper-Star Network

Jong-Seok Kim[1], Hyung Jae Chang[2], Hyeong-Ok Lee*[3]

[1]*North California Central University*
[2]*Troy University – Montgomery*
[3]*Suncheon National University*
[1]*rockhee7@gmail.com*,[2]*hjchang@troy.edu*, [3]*oklee@scnu.ac.kr*

Abstract

As the demand for high performance computing has been increased in these days, parallel processing system becomes widely used to improve the performance of computers. In parallel processing system, all processors are connected in an interconnection network where each processor has its own memory. Recently, Hyper-Star network HS(m,k) was proposed as a new interconnection network for parallel processing. HS(m,k) has the same characteristics as Hypercube and Star graph, and it is proven to have better network cost than Hypercube with the same number of nodes. Tree plays an important role as a data structure in parallel processing system and is a useful network structure to which a divide and conquer algorithm can be easily applied. When a new algorithm is designed, embedding a given network structure in another one is a good application where the new algorithm can be used. In this paper, we show how Fibonacci tree can be embedded in HS(2n,n) and analyze its result. In conclusion, the paper presents that the embedding is possible with the dilation 1.

Keywords: *Algorithm, Hyper-star network, Fibonacci tree, Embedding.*[1]

1. Introduction

In the past, the usage of computers was limited to scientific calculations. Thanks to the advancement of computing technology, however, it has been broadened to processing various types of information, such as letters, voices and videos. As today's problems in scientific and engineering fields require much more calculations in faster time, there have been more and more demands on high performance computers for this purpose. To improve the performance, parallel processing technology became very popular for designing computer architecture. Parallel processing is a technology that can process either multiple programs or many sub-divided parts in a single program in parallel with many processors. In parallel processing system, many processors with their own memory are connected by interconnection network and the communications among the processors are realized using message passing through the interconnection network. Interconnection network can be presented in an undirected graph (G=(V, E)) with nodes being processors and edges being communication channels. Many new interconnection networks have been proposed along with the advancement of method to construct huge interconnection network as the parallel processing techniques are evolved.

One of the most efficient interconnection networks is the Hypercube [1]. Another family of regular networks, the Star graph [2], has been extensively studied. The Hyper-Star network

Article history:
Received (April 11, 2018), Review Result (May 11, 2018), Accepted (June 4, 2018)

Print ISSN: 1975-0080 IJMUE

HS(m,k) was introduced by Lee et al. [3] and Kim et al. [4] as a new type of interconnection network to overcome the drawbacks of both the Hypercube and the Star graph. The Hyper-Star network is a regular network when m=2k. A result by Lee et al. [3] also showed that Hyper-Star network gave lower network cost (measured by the product of degree and diameter) than Hypercube, Folded Hypercube, and other variants. So far, various properties of HS(m,k) have been analyzed by many researchers [3][4][5][6][7][8][9] .

Many parallel algorithms have been designed to solve various problems in different interconnection network structures. It is very important that such algorithms can be used in other different interconnection networks. Embedding is one of the widely used ways to achieve this. Embedding is a way to map processors and communication links in one network to the ones in a different network. The criteria to analyze the embedding cost are dilation, congestion, expansion and etc. [5][6].

The fact that various trees are embedded to communication networks is very important because tree plays an important role as a data structure in parallel processing system and is a useful network structure to which a divide and conquer algorithm can be easily applied. [10][11][12][13] shows that trees can be embedded to various network structures. In this paper, we analyze how embedding can be realized among Fibonacci tree and Hyper-Star network HS(2n,n). In conclusion, Fibonacci tree is shown to be embedded into HS(2n,n) with the dilation 1.

This paper is organized as follows; Section 2 presents Hyper-Star network and Section 3 shows the analysis on embedding Fibonacci tree into HS(2n,n), followed by the conclusion.

2. Hyper-Star Network

Hyper-Star HS(m,k) is an interconnection network consisting of mCk nodes, where a node is represented by the binary string of m bits, b1b2...bi...bm such that the cardinality of the set {i|1≤i≤m,|bi="1"|}=k. Let σi be an operation that exchanges b1 and bi, 2≤i≤m, where bi is a complement of b1. Then, two nodes u=b1b2...bi...bm and v=bib2...b1...bm are connected when v is obtained from the operation σi(u). The edge connecting two vertices u and v is called an i-edge. Since Hyper-Star HS(m,k) is an irregular network, this paper considers regular network HS(2n,n) and Fibonacci tree and analyzes the embedding among them. Figure 1 shows 3-dimentional Hyper-Star network HS(6,3).

If a node in HS(2*n*,*n*) is *u*=0…01…1, it is denoted as $u=0^n1^n$ as the number of 0's and 1's is *n*, respectively. Let the path be $[k_1,k_2,...,k_t]$ after we apply the operations $\sigma_{k1},\sigma_{k2},...,\sigma_{kt}$ one by one on an arbitrary node *u*. For example, the path from node 0011 to node 1100 is either "0011 – 1001 – 0101 – 1100" or "0011 – 1010 – 0110 – 1100". Therefore, the path between the 2 nodes cab be represented as [3,2,4] or [4,2,3].

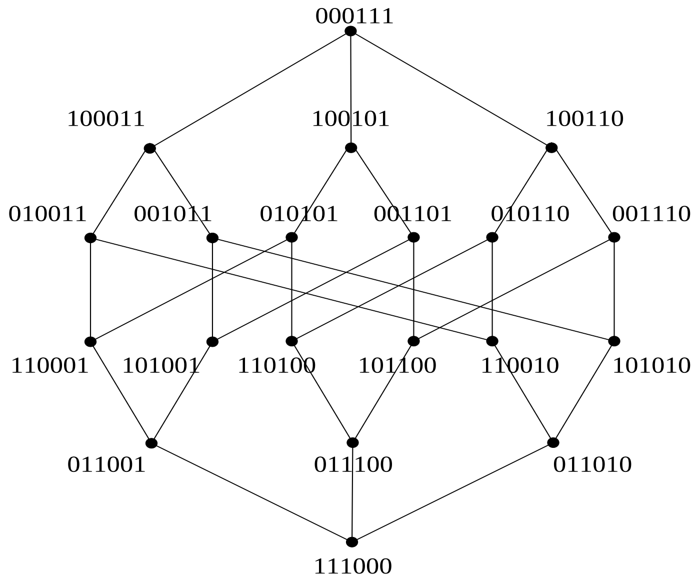

Figure 1. 3-dimentional hyper-star network HS(6,3)

Definition 1. Let 2 arbitrary nodes in HS(2*n*,*n*) be $u=0^n1^n$ and *v*. If the distance between *u* and *v* is *m*, the node *v* on the route connecting nodes *u* and *v* is located at level L_m.

Property 1. By Definition 1, if $n+1\leq i\leq 2n$, *i*-edge connects the nodes located at L_t and L_{t+1}. If $2\leq i\leq n$, *i*-edge connects the nodes located on $L_{t+1}$and L_{t+2} where *t* is an even number.

Lemma 1 [8]**.** Let's assume that an arbitrary node in HS(2*n*,*n*) is $u=0^n1^n$, and there exist 2 paths $P=[p_1,p_2,\ldots,p_s]$ and $Q=[q_1,q_2,\ldots,q_s]$ which start from node *u*. Two paths from *u* represented by $[p_1,p_2,\ldots,p_s]$ and $[q_1,q_2,\ldots,q_s]$ will end up at the same node if and only if, for every number *i*, the number of occurrences of *i* in the two sequences have the same parity.

Lemma 2 [4]**.** Let an arbitrary node in HS(2*n*,*n*) be $u=0^n1^n$. In this case, the subnetwork which consists of all nodes at level L_d $(0\leq d\leq n-1)$ and another subnetwork which consists of all nodes at level L_f $(n\leq f\leq 2n-1)$ are symmetric.

3. Fibonacci Tree Embedding

Embedding is a mapping a certain network G to another network H to check if G is included in H or how G is related to H. The criteria to evaluate embedding cost are dilation, congestion and expansion. Among these, dilation is used the most. The dilation of an edge e in network G is the length of path ρ(e) of the mapped edge e in network H. Therefore, more efficient embedding can be realized when the extension gets smaller. Since HS(2n,n) is node-symmetric, we will construct Fibonacci tree using i-edges with an internal node u=0n1n in HS(2n,n) as a root node.

The definition of Fibonacci tree is as follows;

1. Empty tree with no nodes and a tree with a single node are Fibonacci tree and represented as FT_{-1} and FT_0, respectively.

2. Fibonacci tree FT_k is a binary tree which consists of left and right sub-trees, FT_{k-1} and FT_{k-2} $(k\geq 1)$.

There are 2 properties for Fibonacci tree.

1. The number of nodes in FT_k is F_{k+3}-1. F_k is the kth Fibonacci number in recursive manner. F_0=0, F_1=1 and F_m= F_{m-1}+ F_{m-2} (m≥2).

2. The number of edges in FT_k is F_{k+1}.

Fibonacci tree can be structured using i-edges in HS(2n,n) in as follows;

1. root node is $u=0^n1^n$ and located at L_0.

2. All nodes in Fibonacci tree are located between L_0 and L_n in HS(2n,n).

3. i-edges connecting nodes located at L_t and L_{t+1} are n+t+1 and n+t+2. j-edges connecting nodes located at L_{t+1} and L_{t+2} are t+2 and t+3 where t is an even number.

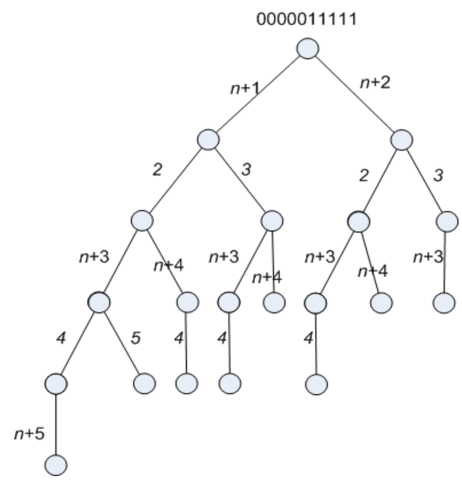

Figure 2. Fibonacci tree in 6-dimentional hyper-star network HS(12,6)

By lemma 1, it can be easily noticed that every path in Fibonacci trees constructed in HS(2n,n) in the way described above is node disjoint. In addition, the degrees of FT_1, FT_2 and FT_n (n>3) are 1, 2 and below, and 3 and below, respectively. The degree of HS(2n,n) is n. In other words, the number of neighbor nodes for all nodes in HS(2n,n) is the same as or larger than the number of neighbor nodes for all nodes in FT_n.

Theorem 1. Fibonacci tree FT_n can be embedded to HS(2n,n) with dilation 1.

Theorem 2. In HS(2n,n), there exist 2 Fibonacci trees FT_n when n≥3.

Proof. By lemma 2, there are 2 symmetric subnetworks in HS(2n,n) and thus, we can construct 2 Fibonacci trees with $u=0^n1^n$ and $w=1^n0^n$ as root nodes. By Definition 1, if the node $u=0^n1^n$ is located at level L_0, $w=1^n0^n$ is located at level L_{2n-1}. Let a Fibonacci tree with $u=0^n1^n$ as a root node be FT_u and another Fibonacci tree with $w=1^n0^n$ as a root node be FT_w. Then, all nodes in FT_u are located between L_0 and L_n and all nodes in FT_w are located between L_{2n-1} and L_{n-1}. In this scenario, we want to prove that there is no overlapped node between FT_u and FT_w. Figure 3 shows how HS(2n,n) can be structured by scalability. When all 2n-edges are removed from HS(2n,n), 2 symmetric subnetworks S_1 and S_2 are remained. Since S_1 and S_2 are symmetric, they have the same set of i-edges and the same method can be applied to structure Fibonacci tree as explained above. Additionally, since a node $u=0^n1^n$ is created from ① in S_1 and a node $w=1^n0^n$ is created from ⑧ in S_2, there are no overlapped nodes in FT_u and FT_w. Therefore, there exist 2 Fibonacci trees FT_n in HS(2n,n) when n≥3.

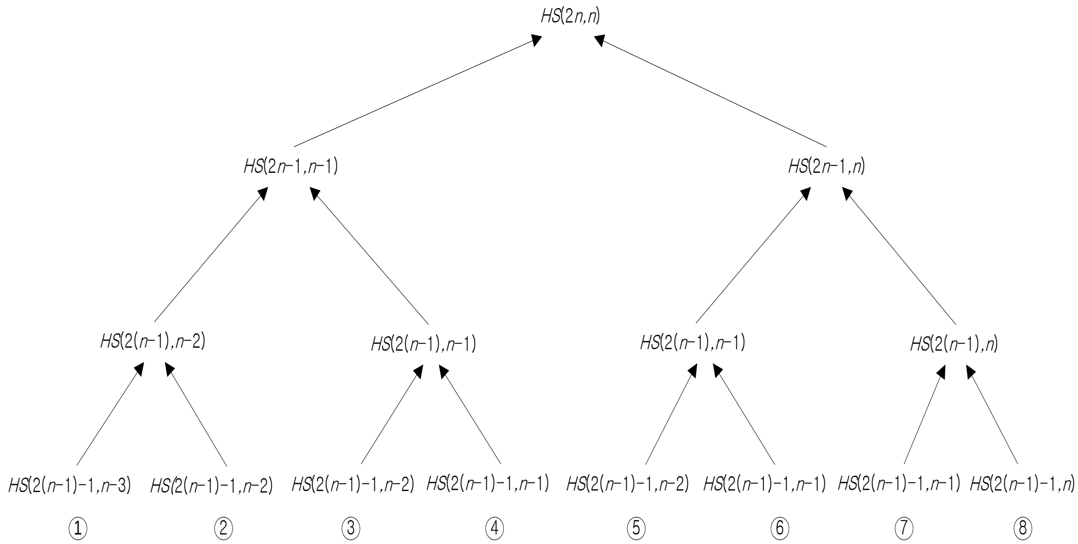

Figure 3. HS(2n,n) by scalability

4. Conclusion

This paper shows that Fibonacci tree can be embedded into Hyper-Star network HS(2n,n) with the dilation 1; network cost was reduced in Hyper-Star network compared to Hypercube which is well known as a useful interconnection network. In addition, the paper presents that 2 Fibonacci trees can be embedded into HS(2n,n). This result implies that Fibonacci tree is a subnetwork of HS(2n,n) and various properties and advantages of Fibonacci tree can be easily applied to HS(2n,n). Moreover, an edge connecting 2 nodes u=0n1n and w=1n0n is added, Fibonacci tree can be used as double rooted tree.

Acknowledgements

This research was supported by Basic Science Research Program through the National Research Foundation of Korea(NRF) funded by the Ministry of Education(2017R1D1A3B03032173).

References

[1] Y. Saad and M. H. Schultz, Topological properties of hypercubes. IEEE Transactions on Computers, Vol.37, No.7, pp.867-872 **(1988)**

[2] S. B. Akers, D. Harel, and B. Krishnamurthy, The Star graph: an attractive alternative to the N-cube, Proceedings of the International Conference on Parallel Processing, **(1987)** January 393-400.

[3] H.-O. Lee, J.-S. Kim, E. Oh, and H.-S. Lim, Hyper-star graph: a new interconnection network improving the network cost of the hypercube. Proceedings of the 1st EurAsian Conference on EurAsia-ICT 2002: Information and Communication Technology, LNCS 2510, **(2002)** October 585-865; Shiraz, Iran

[4] J.-S. Kim, E. Oh, H.-O. Lee, and Y.-N. Heo, Topological and communication aspects of hyper-star graphs. Proceedings of the 18th International Symposium on Computer and Information Sciences, LNCS 2869, **(2003)** November 51-58; Antlya, Turkey

[5] J.-S. Kim, E. Cheng, L. Liptak, and H.-O. Lee, Embedding hypercubes, rings and odd graphs into hyper-stars. International Journal of Computer Mathematics, Vol.86, No.5, pp.771-778 **(2009)**

[6] J.-S. Kim, E. Cheng, L. Liptak, and H.-O. Lee, A note on embeddings among folded hypercubes, even graphs and odd graphs. International Journal of Computer Mathematics, Vol.8, No.5, pp.882-891**(2011)**

[7] E. Cheng, P. Hu, R. Jia, and L. Lipt´ak, Matching preclusion and conditional matching preclusion for bipartite interconnection networks II: Cayley graphs generated by transposition trees and hyper-stars. Networks, Vol.59, No.4, pp.357-364 **(2012)**

[8] L. Lipt´ak, E. Cheng, J.-S. Kim, and S. W. Kim, One-to-many node-disjoint paths of hyper-star networks. Discrete Applied Mathematics, Vol.160, No.13–14, pp.2006-2014 **(2012)**

[9] A. Bossard, A set-to-set disjoint paths routing algorithm in hyper-star graphs, ISCA International Journal of Computers and Their Applications, Vol.21, No.1, pp.76-82 **(2014)**

[10] S. N. Bhatt, F. R. K. Chung, F. T. Leighton, and A. L. Rosenberg, Efficient embeddings of trees in hypercubes. SIAM Journal on Computing, Vol. 21, pp.151-162 **(1992)**

[11] A. K. Gupta, D. Nelson, and H. Wang, Efficient embeddings of ternary trees into hypercubes. Journal of Parallel and Distributed Computing, Vol.63, pp.619-629 **(2003)**

[12] V. Heun and E. W. Mayr, Efficient Dynamic Embeddings of Binary Trees into Hypercubes. Journal of Algorithms, Vol.43, pp.51-84 **(2002)**

[13] H.-S. Lim, J.-H. Park, and K.-Y. Chwa, Embedding trees in recursive circulants. Discrete Applied Mathematics, Vol.69, pp.83-99 **(1996)**