한국산학기술학회논문지
Vol. 12, No. 4 pp. 1889-1896, 2011

Biased Multistage Interconnection Network in Multiprocessor System

Chang-hoon Choi[1*]
[1]**School of Computer Information, Kyungpook National University**

다중프로세서 시스템에서 편향된 다단계 상호연결망

최창훈[1*]
[1]경북대학교 컴퓨터정보학부

Abstract There has been a lot of researches to develop techniques that provide redundant paths, there by making Multistage Interconnection Networks(MINs) fault tolerant. So far, the redundant paths in MINs have been realized by adding additional hardware such as extra stages or duplicated data links. This paper presents a new MIN topology called Hierarchical MIN. The proposed MIN is constructed with *2.5N-4* switching elements, which are much fewer than that of the classical MINs. Even though there are fewer hardware than the classical MINs, the HMIN possesses the property of full access and also provides alternative paths for the fault tolerant. Furthermore, since there is the short cut in HMIN for the localized communication, it takes advantage of exploiting the locality of reference in multiprocessor systems. Its performance under varying degrees of localized communication is analysed and simulated.

요 약 다단계 상호 연결망(MIN)에 중복 경로를 제공하여 오류를 허용하는 기술을 개발하기 위하여 많은 연구가 진행되어 왔다. 지금까지 MIN에서 중복 경로는 추가된 stage와 중복된 데이터 경로 같은 추가 적인 하드웨어를 첨가함으로써 실현되었다. 본 논문은 계층적 MIN이라는 새로운 위상의 MIN을 제안한다. 제안된 MIN은 기존의 MIN보다 적은 *2.5N-4*개의 스위칭소자를 사용하여 구성되었다. 비록 기존의 MIN 보다 적은 하드웨어를 사용할지라도 완전 접근 성질을 가지고 있으며, 또한 오류 허용을 위한 선택적 중복 경로를 제공한다. 더욱이 HMIN에서는 지역화된 통신을 위한 지름길이 제공되어 다중 프로세서 시스템에서 지역 참조성을 활용하는 장점을 갖게 된다. 지역참조성의 정도를 변화시키며 제안된 시스템의 성능을 분석하였으며 모의실험을 수행하였다.

Key Words : Interconnection Networks, Multistage Networks, Multiprocessor Systems

1. Introduction

The shared memory model, in which all N processors access a common memory in constant time, has been widely used in parallel processing system[4,7,21,23]. Crossbar switches are too expensive to construct for large N. The Multistage Interconnection Networks(MINs) among many interconnection types have been used in most of MIMD and SIMD multiprocessor systems since they have good diameter of $\log_2 N$, scalability and employ simple distributed self-routing[7,11,13,18-20]. The classical MINs - Omega[13,21], Baseline, Multistage Cube Network- have usually been designed for constructing $N\times N$ MIN using $(N/2)\log_2 N(O(N\log_2 N))$ switching elements (SEs) with size 2×2. The number of SEs used for the $N\times N$ MINs is much fewer than hardware cost of the Crossbar switch network[6,7,13]. Unfortunately since the classical MINs have only a single unique path between source-destination pairs in the network, the failure of even a single component would result in the

This work was supported by the Kyungpook National University Research Grant, 2009
*Corresponding Author : Choi, Chang-hoon(hoon@knu.ac.kr)
Received February 15, 2011 Revised March 22, 2011 Accepted April 07, 2011

failure of the MINs. This shortcoming of the unique path property(UPP) of the classical MINs have led to the design of multiple path MIN. There has been a large amount of research on multiple path MINs. So far, the multiple path MINs have been realized by adding additional hardware such as extra stages[1-3,5,8,22] or multiple links [1,10,12,14,15,24].

Also, the classical MINs have another problem in that they can not exploit the locality of reference. Interprocessor communications are determined by the algorithms used and the allocation of tasks to processors. It has been shown that optimum cluster size is application-dependent. If the communication probability distribution function for an application is available, the cluster size that will minimize interprocessor communication delay can be determined[13]. However the latency time between source and destination in classical MIN has to increase with $\log_2 N$, the number of stages in the network. Therefore, they couldn't exploit locality because all processors take the same amount of time. This loss of locality hurts its performance when compared to the *n*-cube[16], and Hypertree[10]. So far, there has been little progress for exploiting the locality of reference and short cut path in the classical MINs.

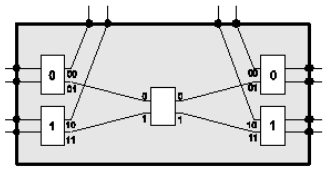
In this paper, we propose a new topology MIN, Hierarchical MIN(HMIN) which is a combination of the topology design of a tree network and the classical MINs similar to Hyperbanyan network[9] and Hypertree. Since the HMIN to be represented in this paper combines the advantages of the static type network with the dynamic type network, HMIN has the features of a binary tree as well as the classical MIN. Moreover, the HMIN is constructed with *2.5N-4(O(N))* SEs, which are much fewer than that of the classical MINs. Even though there are fewer SEs than that of the classical MINs, the HMIN possesses the property of full access and also provides alternative paths for the fault tolerant. Furthermore, since there is the short cut path for the localized communication, it takes advantage of exploiting the locality of reference. For the locality of reference, HMIN always has the distance of 3 as the short cut path regardless of increasing of the network size and has alternative paths.

2. HMIN

2.1 UNIT Module Configuration

An HMIN is a network board that has *N*-input and *N*-output terminals where *N* is a power of 2 (or $n = \log_2 N$). Since HMIN is constructed using UNIT modules, it is easily expandable in an incremental way of a UNIT module. A UNIT module is a network board with 4-input and 4-output terminals which consists of 3 stages - i.e. input stage, middle stage and output stage. The input and output stages contain 2 SEs each, and the middle stage contains one SE. We have to describe the rules of the unit module configuration. Since there are only two SEs at the input(or output stage), they can be expressed by 1-bit binary form, b_1. The binary expression of the links of the two SE at input and output stage can be numbered as b_1b_0.

Let f^i be topological describing function (i = 0 or 1). The function, $f^1(b_1) = b_1$ describes that the lower-output(input) link of a SE b_1 at input stage(output stage) is connected to the input(output) link b_1 of a SE at middle stage. The function, $f^0(b_1) = outside$ describes that the upper links of SE b_1 at input(output) stage provide a connection to another UNIT module see Fig. 1.



[Fig. 1] Unit module

2.2 Scaling MIN using UNIT module

The extension of the network is built up from BaseNet and UNIT modules. The following rules present the topological describing function.

HMIN($2^k \times 2^k$) := **BaseNet** || **UNIT**

BaseNet := **HMIN**($2^{k-1} \times 2^{k-1}$)

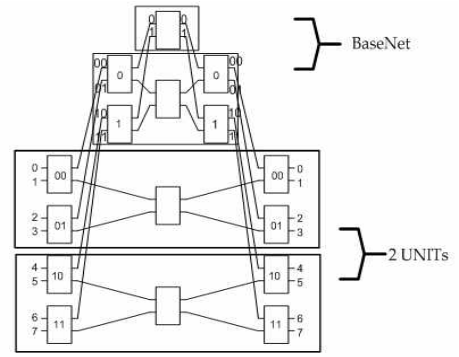
UNIT := $2^k/4$ **UNIT**, where k≥2. When k=1, HMIN($2^1 \times 2^1$) contains only 2×2 SE.

The operator, ‖ is an operation for connecting the BaseNet with $2^k/4$ UNIT modules. The SEs at the input stage(output stage) can be numbered sequentially as binary form, $b_l \cdots b_1 (l = n-1)$ starting from top.

Example :

Scaling for 8×8 HMIN(See [Fig. 2])

HMIN($2^3 \times 2^3$) := HMIN($2^2 \times 2^2$) ‖ 2 UNIT



[Fig. 2] 8×8 HMIN

2.3 Description of the HMIN

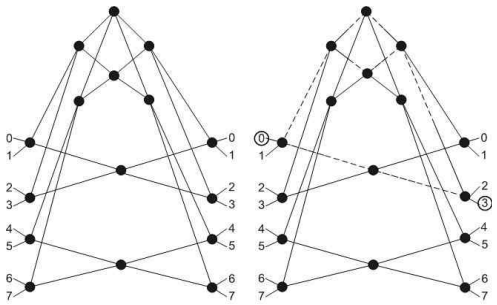
An *N*×*N* HMIN is a network that has *N*-input and *N*-output terminals where *N* is in the power of 2(or $n = \log_2 N$). The HMIN is a *2n-1* stage interconnection network having *2.5N-4* switching elements which is much fewer than the classical MINs with using $(N/2)\log_2 N$ SEs. The number of SEs in each stage *j*(and in stage *2n-2-j*) $0 \le j \le n-2$, is $N/2^{j+1}$. The center stage(stage *n-1*) has *N/2* SEs. The SEs in stage j, $0 \le j \le n-2$, are numbered from 0 to $N/2^{j+1}-1$(*n-1* in center stage) from the topmost SE. The upper and lower input(output) terminals of a SE *i* are numbered as *2i* and *2i+1*, respectively. The interconnection between the SEs in the HMIN is described below.

Network Topology Description :

in each stage j, 0 ≤j<n-1,
for each SE i,
output 2i → input i of stage j+1
output 2i+1 → input i+N/2^{j+1} of stage n-1
in stage j, j=n-1,
for a SE 0,
output 0 → input 0 of stage j+1
output 1 → input 1 of stage j+1
for each SE i, 0 = i < N/2
output 2i → input (i(mod)2^k)x4+1 of stage n+k
output 2i+1→ input (i(mod)2^k)x4+3 of stage n+k
i_k=1 (k is a index of leftmost bit which has value 1, $0 \le k \le n-2$)
in each stage j, n-1< j < 2n-2,
for each SE i, $0 \le i < 2^{j-n+1}$
output 2i → input 2ix2 of stage j+1
output 2i+1 → input (2i+1)x2 of stage j+1

Fig. 3(a) shows the graphical representation of HMIN where each node represents a switching element. The SEs of the HMIN have an interconnection pattern which is a combination of trees having a common root node in the center stage. There are *n* center nodes as a common root node in each pair of source and destination at stage *n-1*. The center nodes in center stage provide the paths between the source and destination. Therefore there are *n* paths in the HMIN. Since there are 3 root nodes in stage 3 between source 0 and destination 3, the number of path are 3. Fig. 3(b) shows 3 alternative paths(dashed line).

The structure of the HMIN allows multiple paths of varying lengths between a pair of source and destination terminals. There are *n+1* trees which have a root node at the middle stage *n-1*. Each source terminal has 3 root nodes(center stage). So, it is obvious that the structure of the HMIN allows *n* alternative paths between a pair of source and destination terminal.



(a) (b)

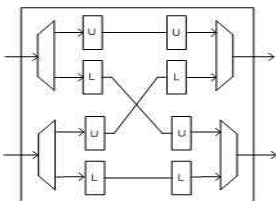
[Fig. 3] Graphical representation of HMIN

The length of *n* paths is between a minimum of 3 to a maximum of *2n-1* in the network. The shortest path among *n* alternative path is of length 3 which is a

constant value regardless of increasing of network size.

2.4 Dual Register Switch for Shortcut

To overcome the deficiency of message blockage and to provide a shortcut, we propose new SE model. The main difference from the conventional switch is that each input/output port has with dual registers marked by *Up(U)* and *Low(L)*. The detail of the new switch design is illustrated in Fig. 4. The assumption of sending a maximum of one packet from each switch output port during one clock cycle is still valid and therefore each SE input link can still receive a maximum of one packet during each clock cycle.

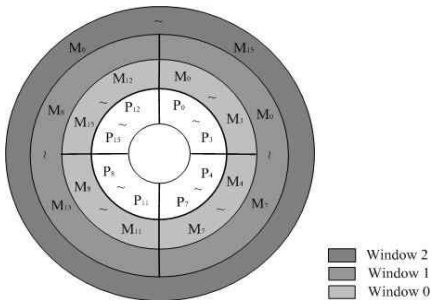


[Fig. 4] Switching Element

On the other hand, we do allow two registers(dual register, *U* and *L*) in input port to send simultaneously during a clock cycle, each packet is sent to a different output port. Up to four packets can be routed from the input registers to output registers simultaneously. This dual register output register design allows two messages to be routed to their corresponding output port registers of the same output port in one cycle. This dual output register design eliminates the conflict on output port insertion which in turn increases link utilization. The routing control inside a switch is very simple. The packet residing in the up input registers is routed to the upper output port, while the message residing in the low input registers is routed to the lower output port. Up to four packets can be routed from the input registers to output registers simultaneously. The packet in the up(or low) output register will be sent to the up(or low) register of the next stage.

3. Routing Strategy

The pair of source and destination can be classified by *n-1* windows in the HMIN. If $(S)_{n-1,i} = (D)_{n-1,i}$, where $2 \leq i \leq n-1$, the sources and destinations belong to the window *i-2*. If $(S)_{n-1} \neq (D)_{n-1}$, sources and destinations belong to the window *n-2*. The higher window includes the lower windows, i.e. the window 2 includes the window 0 and 1. As an example, Fig. 5 shows that each processor has 3 windows in 16x16 HMIN. A processor P_0 can includes the memory modules, M_0,M_1,M_2,M_3, as the window 0, M_0, ~ M_7 as the window 1 and M_0, ~ M_{15} as the window 2.



[Fig. 5] Windows in 16x16 HMIN

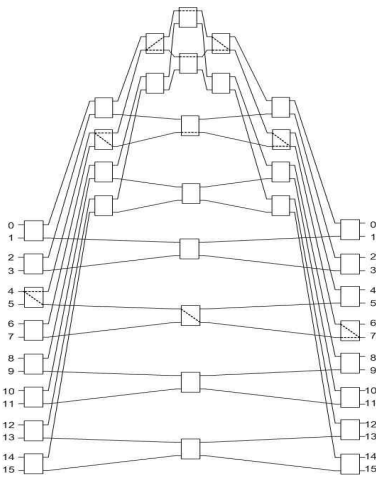
Through the windows, we can explain that the HMIN allows multiple paths of varying lengths between a pair of source and destination. The number of alternative paths to communicate with a processor and memory modules in each window are from a minimum of 2 to a maximum of *n* alternative paths. And they have the distances of a minimum of 3 to a maximum of *2n-1* for each path. The length of the shortest path l between a source S and destination D can be computed by $l = class \times 2 + 3$. For example, let S=1100, D=1111. As $(S)_{n-1,2}=(D)_{n-1,2}$, The source and destination belong to class 0. Hence the length of path l is 3.

The memories in the lower windows have the shorter distance and more alternative paths than that of higher windows. Moreover, the number of the alternative paths increases as the network size increases. As an example, processor P_0 has the memory modules, M_0,M_1,M_2,M_3 as the class 0, M_4 through M_7 as the class 1 and M_8 through M_{15} as the class 2. Since the memory modules, M_0,M_1,M_2,M_3 belong to all the windows for P_0, they have distance of 3 as the short cut. Moreover, they have 3 alternative paths in the network.

The HMIN has the capability of self-routing using the

destination tag which achieves fast routing easily. The number bits of the tag to be used for routing vary from 3 to *2n-1* bits according to the classes as mentioned above. The routing tag in HMIN is following. As a source address is $s_{n-1}s_{n-2}...s_1s_0$ and a destination address $d_{n-1}d_{n-2}...d_1d_0$, the exclusive-OR operation can be executed onto the two bit-strings, $s_{n-1}s_{n-2}...s_2$ and $d_{n-1}d_{n-2}...d_2$ i.e. $c_{n-1}c_{n-2}...c_2 = s_{n-1}s_{n-2}...s_2 \bigoplus d_{n-1}d_{n-2}...d_2$ are scanned one bit at one time from MSB to LSB. When the first bit which is set to '1', c_i=1, is found, the index is decided as *i*. So, the routing tag $R = r_{l-1}...r_0$ for shortcut can be obtained as *i-1* 0s followed by $1d_id_1d_0$. If the bit set to '1' is not found(all zeros), the index *i* will be 1. If the routing tag, r_i at stage *i* is 0, the input packet is sent to a lower register *L* of output registers, which will send to lower of the output port. Otherwise the input packets send to a higher register *U* of input registers, which will upper register of the output port.

The process is carried through the successive stages, sending the input packet at stage *i* to connect to the upper input(output) register if d_i=0 or the lower input(output) register if d_i=1. The proper connection is made using the *ith* digit of the destination tag at stage *i*. Although the shortcut path is busy, the packets are able to re-route using the paths of alternative upper link. For example, a routing tag for connection between source 0(0000) and destination 3(0011) in Fig. 6, is *R*=111 as the short cut.



[Fig. 6] Alternative paths for fault tolerant in 16×16 HMIN

4. PERFORMANCE EVALUATION

In this section, we discuss the performance of HMIN. In the classical MINs the latency is fixed from any input-output connection. However, for HMIN the latency depends on the locality of the communication. Therefore, for the performance evaluations in HMIN, we have to define window *w*[5].

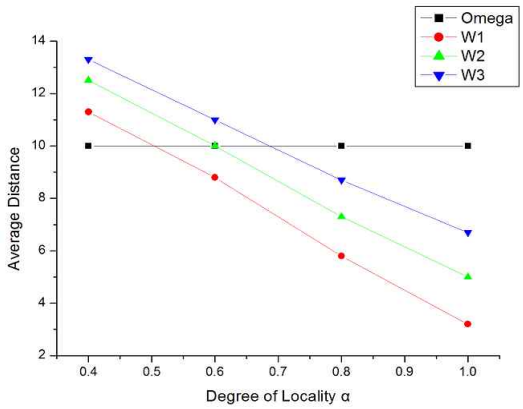
i) Window *w* ($1 \le w \le$ n-1) :

For any input-output connection, there are *n-1* windows as Fig 5.

ii) Degree of locality α:

Let α be the degree of locality to refer the window *w*.

Fig. 7 illustrates the variation of the delay time as the degree of locality α and window *w* vary in 1024×1024 network. The classical MINs take the same time to communicate between pairs. However HMIN has the less delay time as the probability of locality increase or the references come into the lower windows.



[Fig. 7] Average delay with window size and degree of locality

In a 2×2 Crossbar module, the expected number of requests that passes through each output per time unit is the same as $1-(1-m/2)^2$, where *m* is the average number of requests generated per cycle by each input [16]. However a crossbar module used in HMIN has a different request rate *m=1* for each output under uniform

communication. We assume that for an input request rate at each of the 2 input lines of a 2×2 module for all stages *i(0 ≤i ≤n-2)*, the expected number of requests, M_{out} that passes through each output link per time unit is

$$M_{out} = 2\times(2^{i+2}/N\times(N-2^{i+2})/N) + ((N-2^{i+2})/N)^2/2 + (2^{i+2}/N)^2/2 \ .$$

Consider $2^{i+2}(0<2^{i+2}\le N)$, which represents the number of destinations that can be reached from the output link of stage *i* in $N\times N$ HMIN. It follows that $0 \le 2^{i+2}/N \le 1$. Let $2^{i+2}/N = \alpha$ which is the rate of reference under uniform communication. However we consider α as the degree of the locality of reference through the output link and independent of stage *i* for the locality of reference. Then the expected numbers of requests on *U(L)* link that pass through each output per unit time, considering α, are given respectively by

$$M_i^U = 2\alpha(1-\alpha) + (1-\alpha)^2/2 \text{ and}$$
$$M_i^L = 2\alpha(1-\alpha) + \alpha^2/2.$$

The following equations determine the bandwidth, BW of an *N×N* HMIN. Since the output rate of a stage is the input rate of the next stage, one can recursively evaluate the output rate of any stage starting at stage 0. In particular, the output rate of the final stage *2n-1* determines the bandwidth of a HMIN that is the number of requests accepted per cycle. Let $m_j^{U(L)}$ be the rate of request on an *U(L)* output link of stage *j*.

$$m_j^U = (1-(1-m_{j-1}^U/2)^2)\times M_j^U$$
$$m_j^L = (1-(1-m_{j-1}^U/2)^2)\times M_j^L \quad \text{and} \quad m_{-1}^U = m,$$
for $0 \le j \le n-2$

Therefore, $BW = 2^n m_{2n-2}$

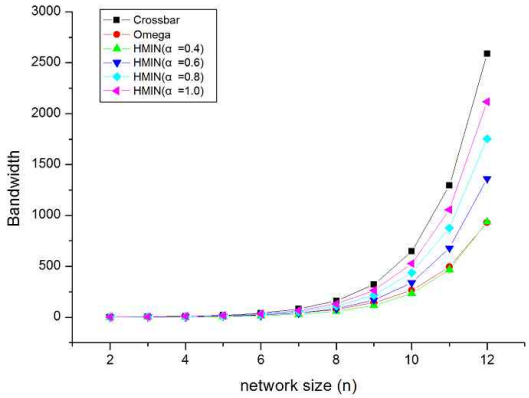
where,

$$m_i = 1-(1-\max\{m_{i-1}, m_{n-1}\}/2)^2,$$
for $n \le i \le 2n-2$
$$m_{n-1} = 1-(1-(m_{2n-2-i}^L)/2)^2 \text{ and}$$
$$m_j = m_{j-1}^U \text{ and } m_{-1}^U = m, \text{ for } 0 \le j \le n-1.$$

Fig. 8 shows the expected bandwidth according to the degree of the locality of reference through the output link as *n* grows. The bandwidth is measured in number of requests accepted per cycle.

5. Simulation Result

The simulations were conducted on the network to study various network parameters. The principal measures used to evaluate and compare the performance of packet switching networks are throughput and delay. We investigate the performance of the HMIN for the following distributions[16].



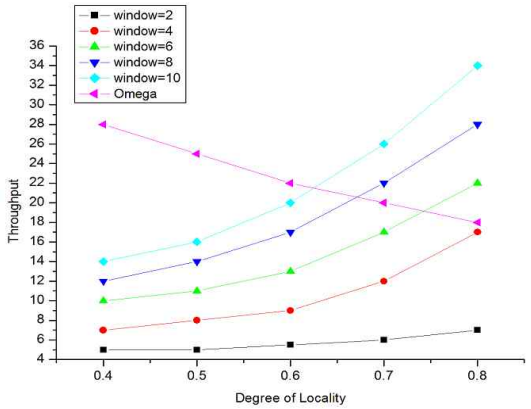
[Fig. 8] Expected bandwidth

Uniform distribution : In this distribution the probability of an input node *i* sending a message to an output node *j* is the same for all *i* and *j*.

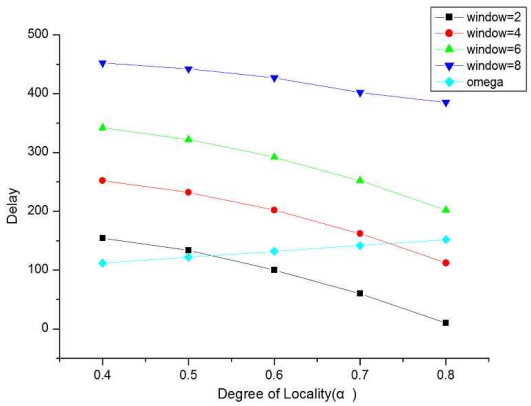
Degree of locality : It is possible for an input node *i* to exchange messages more frequently with nodes in a particular area. A node sends messages to the nodes withins window *w* with some probability α and to nodes outside the window *w* with probability 1-α.

Fig. 9 shows the variation of the throughput for a degree of locality. As expected, we find that the throughput decreases with increase in window size. Fig. 10 shows the variation of delay time. The average delay increases with increase in window size because the distances become progressively larger. For a given window size, it decreases with increase in α because destinations within the window *w* are being generated

with greater probability. From the above graphs it is clear that the HMIN performs well when the communication is localized.



[Fig. 9] Variation of throughput with window size and degree of locality



[Fig. 10] Variation of average delay with window size and degree of locality

6. Conclusions

In this paper, we proposed a new topology MIN and routing algorithm. Though the proposed HMIN has fewer switching elements and links than classical MIN of the same size, it maintains the full access property. Hence, it can form a cheap and efficient packet switching network. And the HMIN has also alternative paths and the shortcut.

In the large scale multiprocessor system where the communication is localized within small pair-sets of processors and memory modules, the HMIN performs better than the classical MINs of the same size.

References

[1] G. B. Adams, D. P. Agrawal, and H.J. Siegel, "A Survey and Comparison of Fault-Tolerant Multistage Interconnection Networks", IEEE Computer, vol. 20, pp. 14-27, June, 1987.

[2] B. D. Alleyne and I. D. Scherson, "Expanded Delta Networks for Large Parallel Computers", in Proc. Int'l Conf. on Parallel Processing, vol. I., pp. 127-131, 1992.

[3] K.V. Arya and R. K. Ghosh, "Designing a New Class of Fault Tolerant Multistage Interconnection Networks", Journal of Interconnection Networks, Vol. 6, No. 4, pp.361-382, 2005.

[4] R.J. Baron and L. Higbie, *Computer Architecture Case Studies*, Addison-Wesley Pub., 1992.

[5] C.M. Chiang, S. Bhattacharya, and L.M. Li, "Multicast in Extra-Stage Multistage Interconnection Networks", in Proc. the 6th IEEE Symp. on Parallel and Distributed Processing, pp.452-459, Oct., 1994.

[6] T. Y. Chung and D. P. Agrawal, "Cost - Performance Trade - off in Manhattan Street Network versus 2-D Torus", in Proc. IEEE Int'l Conf. on Parallel Processing", pp. 169-172, Aug., 1990.

[7] A.L. Decegama, *The Technology of Parallel Processing : Parallel Processing Architectures and VLSI hardware volume I*, Prentice-Hall International Editions, 1989.

[8] C.C. Fan and J. Bruck, "Tolerating Fault Interconnection Networks with Minimal Extra states", IEEE Transactions on Computers, 49(9), pp. 998-1004, 2000.

[9] C.S. Ferner and K.Y. Lee, "Hyperbanyan Networks: A New Class of Networks for Distributed-Memory Multiprocessor", in Proc. the Fourth Symp. on the Frontiers of Massively Parallel Computation, pp.254-261, Oct., 1992.

[10] J.R. Goodman and C.H. Sequin, "Hypertree: A Multiprocessor Interconnection Topology", IEEE Trans. on Compt., vol. C-30, pp.923-93., Dec., 1981.

[11] A. Gottlieb, R. Grishman, et al., "The NYU Ultacomputer Designing a MIMD Shared Memory Parallel Computer", IEEE Trans. on Computers, Vol. C-32, No.2, pp.175-189, Feb. 1983.

[12] T. Hanawa, H. Amano, and Y. Fusikawa, "Multistage

Interconnection Networks with Multiple Outlets", in Proc. Int'l Conf. on Parallel Processing, vol. I, pp. 1-8, 1994.

[13] K. Hwang, *Advanced Computer Architecture : Parallelism Scalability Programmability,* McGraw-Hill International Editions, 1993.

[14] Nitin, S. Garhwal and N. Srivastava, “Designing a Fault-tolerant Fully-Chained Combining Switches Multi-stage Interconnection Network with Disjoint Paths”, The Journal of Supercomputing, Vol. 55, No. 3, pp. 400-431, 2011.

[15] K. Padmanabhan and D.H. Lawrie, "A Class of Redundant Path Multistage Interconnection Networks", IEEE Trans. Compt., vol. C-32, pp. 1099-1108, Dec., 1983.

[16] J.H. Patel, "Performance of Processor-Memory Interconnections for Multiprocessors" IEEE Trans. on Compt., vol. C-30, pp.771-780, Oct., 1981.

[17] M. C. Pease, "The Indirect binary n-cube microprocessor array", IEEE Trans. Comt., vol. 26, pp. 458-473, May, 1977.

[18] G. F. Pfister and W. C. Brantley, et al., "The IBM Research Parallel Processor Prototype(RP3): Introduction and Architecture", Proc. Int'l. Conf. on Parallel Processing, pp.767-771, Aug., 1985.

[19] G. F. Pfister and W. C. "Butterfly GP1000 Overview", BBN Advanced Computer Inc., Nov., 1988.

[20] G. F. Pfister and W. C. "TC2000 Technical Product Summary", BBN Advanced Computer Inc., Jul., 1989.

[21] C. Schäck, W. Heenes and R. Hoffmann, “A Multiprocessor Architecture with an Omega Network for the Massively Parallel Model GCA”, Lecture Notes in Computer Science, Vol. 5657, pp. 98-107, 2009.

[22] S.R. Shankar and L. Jenkins, "The extra stage fault tolerant technique for self-routing permutation networks", Proc. FTS 1st Int'l. Conf. on Fault Tolerant Systems, 1995.

[23] H.J. Siegel, *Interconnection Networks for Large-scale Parallel Processing*, Lexington books, 1985.

[24] T.H. Szymanski, "On the universality of Multipath Multistage Interconnection Networks", J. Parallel and Distributed Computing, vol.7, pp.541-569, 1989.

Chang-hoon Choi [Regular member]

- Feb. 1990 : Sogang Univ., Computer Science, MS
- Feb. 1997 : Sogang Univ., Compter Science, PhD
- Oct. 1995 ~ Feb. 1996 : AT&T(NCR), San Diego Visiting Researcher
- Sept. 2007 ~ Feb. 2009 California State Univ., Dept. of Computer Science, Visiting Professor
- Sept. 1997 ~ Feb. 2009 : Sangju National Univ., Dept. of Computer Eng., Professor
- Mar. 2009 ~ current : Kyungpook National Univ., School. of Computer Information., Professor

<Research Interests>
Parallel Processing System, Embedded System.