Topology and Routing Algorithm Based on the Combination Gray Code with Johnson Code

Xiaoqiang Yang
School of Microelectronics, Xidian University, Xi'an , China
Email: xyangq@sina.com

Huimin Du and Jungang Han
Department of computer science, Xi'an Institute of Post & Telecommunications, Xi'an, China
Email:{ fv, hjg}@xiyou.edu.cn

Abstract—**Hypercubes utilizing Gray Code are used widely as the interconnection networks of parallel computer systems. But an n-dimensional hypercube has 2n nodes, when the size of a network have been increases added nodes must be 2i times more than its nodes and increase the degree of nodes , which affects scalable characteristic. The paper puts forward a scalable hypercube based on the Combination Gray Code with Johnson Code(GJCode(s, t)). The GJCode(s, t) has flexible scalable characteristics and the important properties of both Gray Code and Johnson Code which is there exists a link between two nodes only if their binary addresses differ in a single bit. In the GJCode(s, t), there are two adjustable parameters which offer a good tradeoff between the cost of networks and their performance in terms of the demand of application systems. The paper discusses the performance of several variations of the hypercube and proposes the routing algorithm for the GJCode(s, t), which can make packets pass along the shortest distance path and is very simple and easy to be implemented in hardware with low implementation cost. The GJCode(s, t) can be applied in parallel computer systems and networks on chip.**

Index Terms—**Network Topology, Routing Algorithm, Node Encoding, Hypercube, parallel architecture, Network on Chip**

I. Introduction

Interconnection networks provide mechanisms for data transfer among processing nodes or between processors and memory modules. They are applied in parallel computer systems, distributed computer systems, and especially networks on chip (NoC). A network is defined by the topology and the protocol. The topology of networks concerns the placement and interconnection of its nodes. Protocols specify how these nodes and links work [1].

Topology is one of the most important design issues for these networks [2]. Many topologies are orthogonal, and nodes are arranged in an orthogonal n-dimension space. The main advantages of orthogonal topologies are simple routing, support for wide application spectrum, and fault tolerance. One of the most widely used is hypercube which has been applied in the interconnection network in a wide variety of parallel systems such as Intel iPSC [3], the Ncube [4], the Connection Machine CM-2 [5] and SGI Origin 2000 [6].

The n-dimensional hypercube, however, scales too rapidly as n increases. The number of links, $n2^{n-1}$, grows more drastically than the number of processors, 2^n [7]. To exacerbate the situation, link complexity is directly related to hardware costs and affects VLSI implementation [8] [9]. Currently, practical number of links is limited to about eight per node [6]. When the size of a network have been increases, added nodes must be 2^i times more than its nodes, which affects scalable characteristic.

Several variations of the hypercube have been in the literature. Some variations focused on reduction of the hypercube diameter, for example the folded hypercube[10] and crossed cube [11]; some focused on reduction of the number of links of the hypercube, for example cube connected cycles [12], reduced hypercube[13], and exchanged hypercube [9]; and some focused on both, as in the hierarchical cubic network [14]. Reducing link complexity, however, typically compromises performance, application support and reliability. The challenge then is to achieve a good balance of affordable computational performance and support for larger scale problems [15].

In hypercube structure, the node encoding based on Gray Code has one key property which is there exists a link between two nodes only if their binary addresses differ in a single bit. This property is useful for deriving a number of parallel algorithms for the hypercube architecture and can simplify routing protocol implementation [16].

By utilizing the combination network topology with node encoding and possessing the good property of both Gray Code and Johnson Code, the paper puts forward a scalable hypercube based on the Combination Gray Code with Johnson Code (GJCode(s, t)). The GJCode(s, t) can hold balance between the cost of networks and its performance, and increases only 2t nodes which need not be 2^i times more than its nodes ,and support for larger scale problems.

The rest of the paper is organized as follows. In the next section, we introduce the definitions of binary unit-distance cyclic code, Gray Code and Johnson Code and analyze the characteristic of Gray Code and Johnson Code. In Section III, we describe topology constructing and node encoding of hypercubes utilizing Gray Code and the expanding and node coding of 2-degree torus by Johnson Code. The GJCode(s, t) is proposed in section IV. In section V, we analyze the performance of GJCode (s, t). Section VI puts forward the routing algorithm for GJCode(s,t). The conclusion and the future research are provided in section VII.

II. BINARY UNIT-DISTANCE CYCLIC CODE

A. Binary Unit-distance Cyclic Code

Definition 1 A set of integral binary numbers is called binary unit-distance cyclic code if

(1) any two neighbouring numbers have one and only one bit different (unit distance characteristic);

(2) the first number and the last one in the sequence have one and only one bit different (cyclic characteristic).

Definition 2 Given a set of natural numbers (0, 1,…, 2^k-2, 2^k-1), Each number of them is labeled binary form such as $A_{n-1} A_{n-2} \ldots A_1 A_0$ and it can be mapped to $A_{n-1}(A_{n-1} \oplus A_{n-2})\ldots(A_3 \oplus A_2)(A_2 \oplus A_1)(A_1 \oplus A_0)$ which is called Gray Code, where " $\oplus$ " represents nonequivalence operation.

Definition 3 Given an increasing sequence (0, 1, 2… p-1, p, p+1 ,…, n-2, n-1), encode it with m=$\lceil n/2 \rceil$ bits, where n=2t, t=0, 1, 2,⋯. The encoded code satisfies the conditions of Definition 1, and if

(1) for p<m, the code of p is $Q=F_{m-1}\ldots F_k T_{k-1}\ldots T_0$, where k=p, "$F_{m-1}\ldots F_k$" is the sequence which consists of 0, while "$T_{k-1}\ldots T_0$" is the sequence which consists of 1, and if k=0, Q is the sequence which consists of 0.

(2) for p≥ m, the code of p is $Q=T_{m-1}\ldots T_k F_{k-1}\ldots F_0$, where k=p-m, "$F_{k-1}\ldots F_0$" is the sequence which consists of 0 while "$T_{m-1}\ldots T_k$" is the sequence which consists of 1, and if p=m, Q is the sequence which consists of 1. This code sequence is called Johnson code.

B. Transformation between natural binary code and Two kinds code

Algorithm 1 The transformation between natural binary code and Gray Code is such as the following.

(1) According to Definition 2, each natural binary code can be mapped to Johnson code.

(2) Each binary Gray Code $A_{n-1} A_{n-2}\ldots A_1A_0$ can also be mapped to a natural binary code $B_{n-1}B_{n-2}\ldots B_1B_0$, where $B_k=A_k \oplus B_{k+1}$, 0≤k<n-1 and $B_{n-1}= A_{n-1}$.

Algorithm 2 The transformation between natural binary code and Johnson code is such as the following.

(1) According to Definition 3, each natural binary code can be mapped to Johnson code.

(2) Accordingly, a Johnson Code sequence（A_{n-1}, A_{n-2}, …, A_p,…, A_2, A_1, A_0）has corresponding natural binary sequence which is continuous and m=$\lceil \log_2 n \rceil$ bits in bit width as follows.

① If the highest bit of Ap is 0, its natural binary code is a binary data which is equal to the sum of the number of 1in Ap;

② If the highest bit of Ap is "1", its natural binary code is binary data which equals the sum of the total number of 0 in Ap and m.

Both Gray Code and Johnson Code possess the unit distance characteristic and the cyclic characteristic, and are binary unit-distance cyclic code. In respective code set, the minimum distance between two codes is Haming Distance that is equal to the number of bits that are different in the two codes and is given by simple nonequivalence operation. But there are different characteristics and dissimilar applications between the two kinds of codes. 2^n Gray Codes can be composed of n bits while only 2n Johnson Codes can be done. The data encoded by Gray Code need smaller storage space than Johnson Code. Satisfied with cyclic and unit distance characters, Gray Code need to increase at least 2^n codes while Johnson Code 2 when their bit width increase 1bit. Therefore, Johnson Code is provided with better utilizing rate of space and its scalable characteristic is more flexible than Gray Code.

III. TOPOLOGY CONSTRUCTING AND NODE CODING OF TWO KINDS OF NETWORK

The general class of k-d meshes refers to the class of topologies consisting of d dimensions with k nodes along each dimension. Just as a linear array forms one extreme of the k-d mesh family, the other extreme is formed by an interesting topology called the hypercube.

Definition 4 The n-dimensional hypercube contains 2^n nodes and has n links per node. If unique n-bit binary Gray Code as addresses are assigned to its nodes, then a link connects two nodes if and only if their binary addresses differ in a single bit.

A. Constructing and Node Coding of hypercubes

Algorithm 3 The constructing of $n(n\geq 0, n\in z)$ dimensional hypercube is described as follows by pseudocode.

```
begin
   construct a 0-dimensional hypercube with 2^0, i.e., one
   node;
    i=0;
  while (i<n)
    {  i=i+1;
        the i-dimensional hypercube is constructed by
       connecting corresponding nodes of two (i-1)-
       dimensional hypercubes ;}
end.
```

Algorithm 4 Node coding of $n(n \geq 1, n \in z)$ dimensional hypercube by using Gray Code is described as follows by pseudocode.

```
begin
    a 1-dimensional hypercube consists of 2^1, i.e., two
    node, one code is "0", another is "1";
    i=1;
  while (i<n)
```

{ i=i+1；
node codes of a i-dimensional hypercube is constructed by adding “0” in front of one and adding “1” in front of another which are two (i-1)-dimensional hypercubes from which the i-dimensional hypercube is constructed;}
end

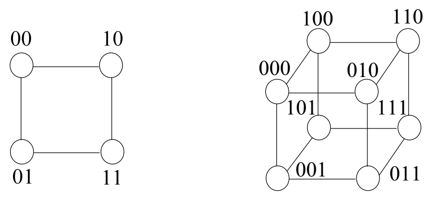
The constructing and coding of a 3-dimensional hypercube is illustrated in Figure 1. In hypercubes, the node encoding based on Gray Code make each two neighbouring codes have one and only one bit different and the minimum distance between two nodes is given by the number of bits that are different in the two codes. The calculation of the minimum distance by nonequivalence operation is simple in implementation. This property is useful for deriving a number of parallel algorithms for the hypercube architecture and can simplify routing protocol implementation. An n-dimensional hypercube has 2^n. The more nodes added in a hypercube, the more degree of nodes added and the more complexities of connection between nodes. when the size of a network increases, added nodes must be 2^i times more than its nodes, which makes scalable characteristic bad.

B. Expanding and Johnson Coding of 2-degree Torus

The 2-degree torus is called 1-dimensional torus and a simple extension of the linear array. The torus has a wrap around connection between the extremities of the linear array. In this case, each node has two neighbors. Johnson Coding of 2-degree torus is the process that the addresses of the nodes in the tours are labeled natural number sequence 0, 1, 2⋯,n-1,（n=2t, where t=0, 1, 2,…）, and the sequence is transformed to Johnson Codes according to the definition 3. The definition 3 indicates that the number of nodes in 2-degree torus using Johnson Code is even. Expanding the scale of 2-degree torus, 2t nodes (2 at the fewest nodes, where t=0, 1, 2,…) is added and the degree of nodes is invariable and still equals 2. It is obvious that the 2-degree torus using Johnson Code is good in scalable characteristic. Figure 2 shows a simple example about topology expanding and Johnson Coding of 2-degree torus. The 1-dimensional torus with two nodes has one link, but have two links such as figure 2 (a) in the paper for coherence.

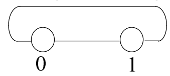
The smaller the degree of nodes is in networks, the simpler the connection and the less the cost. But networks with a few links have low performance because a few links affect routing algorithms and parallel algorithms. By contraries, networks with more degree of nodes have high performance, but are more complex in connection and more in cost. In parallel systems and networks on chip and so on, both magnitude of nodes degree and the size of networks are obtained by the demand of given systems. It is very import that the method for constructing networks with different the degree and the numbers of nodes.

(a) 0-d hypercube (b) 1-d hypercube

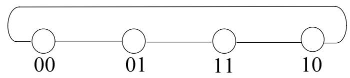


(c) 2-d hypercube (d) 3-d hypercube

Figure 1. Constructing and coding of a 3-dimensional hypercube



a) 2-degree turus with 2 nodes



b) 2-degree turus with 4 nodes

Figure 2. Expanding and Johnson Coding 2-degree torus

IV. CONSTRUCTING AND NODE CODING OF GJCODE(T, S)

A. Definition of GJCode(t, s) Code

Definition 5 The code based on the combination Gray Code with Johnson Code (GJCode(t, s) code) is a kind of two-dimensional binary unit-distance cyclic code. Let GJCode(s, t) code is $m\left(m \geq 1, m \in z\right)$ bits in width, GJCode(s, t) code = Johnson_code ++ Gray_Code and m=t+s. Where Gray_Code is Gray Code with $t\left(t \geq 0, t \in z\right)$ bits width, Johnson_code is Johnson Code with $s\left(s \geq 1, s \in z\right)$ bits width and “++” denotes join operation, and if

(1) any two neighbouring numbers in each dimension have one and only one bit different (unit distance characteristic).

(2) the first number and the last one in each dimension have one and only one bit different.(cyclic characteristic)

For t equals 0, GJCode(s, t) code is Johnson Code.

B. Constructing of GJCode(t, s)

The Constructing of GLCode(s, t) is described as follows. A 2-degree torus is an element. The degree of nodes is increased by the method for expanding dimension of hypercubes, namely, a GJCode(s, t) constructed by connecting corresponding nodes of two GJCode(s, i-1). But when degree of nodes is invariable and number of nodes only increase, what to do is to expand 2-degree torus by Johnson Code.

Algorithm 5 Constructing of GJCode(s, t) with n nodes is described as follows by pseudocode.

begin construct 2-degree torus with 2^1 nodes;
j=1;
while(j<s)
{j=j+1；2-degree torus increase 2 nodes;}
i=0;
while(i<t)
{ i=i+1;

construct the GJCode(s, i) by connecting corresponding nodes of two GJCode(s, i-1);}
end

C. Node Encoding of GJCode(s, t)

The node coding of GJcode(s, t) is described as follows. The 2-degree torus is coded by using Johnson Code with s bits width according to the definition 3, and the codes are Johnson Code part of GJCode(s, t) codes. Gray Code parts of GJcode(s, t) are constructed by adding "0" on the right side of one and adding "1" on the right side of another which are two GJCode(s, t-1) used for construction of GJCode(s, t).

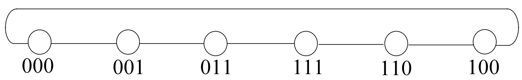
Algorithm 6 Node coding of GJCode(s, t) with n nodes is described as follows by pseudocode.

```
begin
   The 2-degree torus is encoded by using Johnson Code
   with s bits width according to the definition 3;
  Gray Code part of GJCode(s, t) is labeled empty string
   ;
  i=0;
 while(i<t)
  { i=i+1;
    Gray Code parts of GJcode(s, i) are constructed by
    adding "0" on the right side of one and adding "1"
    on the right side of another that are two GJcode(s, i-
    1) used for constructing of GJcode(s,i)  ;}
   end
```

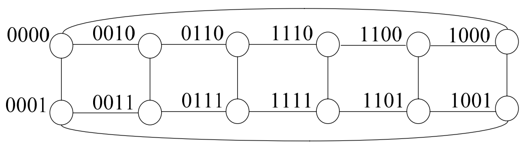
Figure 3 illustrates the construction and coding of a GJCode(3,2) network.

V. Performance Analysis

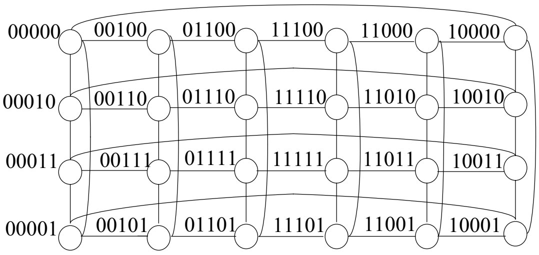
There are various criteria which characterize the performance and cost of networks. A topology is evaluated in terms of number of parameters as defined below.



(a) GLCode(3, 0)



(b) GLCode(3, 1)



(c) GLCode(3, 2)

Figure 3. Constructing and node encoding of a GJCode(3, 2)

(1) Nodes: The total number of nodes in a topology for a given dimension.

(2) Links: The total number of links in a topology for a given dimension.

(3) Degree: Number of edges per node. It is best if the number of edges per node is constant independent of the network size, because the processor organization scales more easily to systems with large number of nodes.

(4) Diameter: The diameter of a network is the largest distance between two nodes low diameter. Low diameter is better, because the diameter puts a lower bound on the complexity of parallel algorithms requiring communication between arbitrary pairs of nodes. It can used to measure the maximum communication delay.

(5) Cost: A network with a larger diameter has nodes with small degree but suffers from long delay in inter-processor communication. On the other hand, a small diameter network usually possess nodes with large degree. It is desirable to have a computer network with small diameter and degree of the node. Thus, for a symmetric network, a cost factor can be defined as product of the diameter and degree of the node.

(6) Bi-section width: The bisection width of a network is the minimum number of edges that must be removed in order to divide the network into two halves (within one). High bisection width is better, because in algorithms requiring large amounts of data movement, the size of the data set divided by the bisection width puts lower bound on the complexity of the parallel algorithms.

Table I shows the parameter values of the various interconnection networks which are GJCode(s, t), Hypercube, 2-d torus(n), Completely connected(n) (Com-con(n)), Folded Hypercube(n) [10], Crossed cube(n) [11], CCC(n) [16], HCN(n,n) [16][17] and Dual cub(n) [18]. Table1 gives the comparison performance between GJCode(s, t) with various topologies. GJCode(s, t) has two parameters which can are offer a good tradeoff of affordable computational performance and costs and support for larger scale problems. The Performance and cost of GJCode(s, t) is related to both the bit width s of Johnson Code part and the bit width t of Gray Code part in the GJCode(s, t). The greater degree of nodes is, the greater degree of nodes, the higher the performance and the cost. when the degree of nodes is invariable, GJCode(s, t) can scales up in size and increases only 2t nodes while hypercube can't. The balance between the cost of the networks and their performance is prone to adjust. By changing the parameters s , GJCode(s ,t) can scale with the invariable degree of nodes. In the GJCode(s, t), there are two adjustable parameters which offer a good tradeoff between the cost of networks and their performance in terms of the demand of application systems.

VI. Routing of GJCode(s, t)

The address of every node in the GJCode(s, t) is expressed as the form of ($N_{s+t-1}\ldots N_1N_0$), which is a

GLCode code. The shortest path between the node S and node D in the GJCode(s, t) is denoted C=Haming（S $\oplus$ D ）, where “ $\oplus$ ” represents nonequivalence operation and function Haming means Haming Distance which is the sum of the total number of 1 in binary number. The operation is simple in the implementation of hardware mechanism. In terms of the characteristic of Gray Code and Johnson Code, we achieve the routing algorithm of GJCode(s, t).is simple in the implementation of hardware mechanism.

TABLE I. PERFORMANCE ANALYSIS OF VARIOUS NETWORK TOPOLOGIES

Networks	Nodesss	Links	Degree	Diameter (d_{max})	Cost	Bisection Diameter
GLCode(s,t)	$2s2^t$	$s(t+2)2^t$	t+2	s+t	(t+2)(s+t)	2^{t+1} $s\geqslant 2$
Hypercube	2^n	$n\ 2^{n-1}$	n	n	n^2	2^{n-1}
2-d torus(n)	n	n	2	$\lfloor n/2 \rfloor$	$2\lfloor n/2 \rfloor$	2
Comp-con(n)	n	n(n-1)/2	n-1	1	n-1	$n^2/4$
Folded Hyper(n)	2^n	$(n+1)\ 2^{n-1}$	n+1	$\lceil n/2 \rceil$	$\lceil n/2 \rceil(n+1)$	2^n
Crossed cube(n)	2^n	$n\ 2^{n-1}$	n	$\lceil n+1 \rceil/2$	$n\lceil n+1 \rceil/2$	2^{n-1}
CCC(n)	$n2^n$	$3n2^{n-1}$	3	5n/2-1	15n/2-3	2^{n-1}
HCN(n,n)	2^{2n}	$2^{2n-1}(n+1)$	n+1	$n+\lfloor (n+1)/3 \rfloor+1$	$(n+1)\ d_{max}$	2^{2n-1}
Dual cub(n)	2^{2n+1}	$2^{n-1}\ (n+1)$	(n+1)/2	n+1	$(n+1)^2/2$	2^{n-2}

Algorithm 7 The routing algorithm of GJCode(s, t) is described as follows by pseudocode.

```
begin
  Let the addresses of  the node which will send packets
  and destination node label  S, D, and S==(S_{s+t-1}… S_1S_0),
  D=(D_{s+t-1}… D_1D_0). The routing algorithm of GJCode(s, t)
  contains two steps.
Step1: //Node S send the packet in the hypercube of
       //the GJCode(s, t) according to Gray Code.
  Node S computes P=(S_{t-1}...S_1S_0) ⊕ (D_{t-1}...D_1D_0);
  If (P= =0)
  then go  Step2;
  else
   {Node S sends the packet along dimension k, where k
    is the position of the least significant nonzero bit in P;
   go  Step1;}
Step2: //Node S send the packet in the hypercube
       //according to Gray Code.
  Node S computes P=(S_{s+t-1}...S_{t+1}S_t ⊕ (D_{s+t-1}...D_{t+1}D_t);
  If (P= =0)
  then  Stop  //Node S is destination node
  else{computes P1=( ¬S_t S_{s+t-1}...S_{t+1} ) ⊕  (D_{s+t-1}...D_{t+1}D_t);
       computes P2=(S_{s+t-2}...S_{t+1} ¬S_{s+t-1} ) ⊕ (D_{s+t-1}...D_{t+1}D_t)
       If (P1<P2)
      then  Node S send the packet to the node represents
            ( ¬S_t S_{s+t-1}...S_{t+1}S_{t-1}...S_0);
      else   Node S send the packet to the node represents
            (S_{s+t-2}...S_{t+1} ¬S_{s+t-1} S_{t-1}...S_0);
   go  Step 2;}
end
```

The routing algorithm of GJCode(s, t) adopts GJCode(s, t) code based on the Combination Gray Code with Johnson Code. The GJCode(s, t) has useful property that the minimum distance between two nodes is given by the number of different bits between their codes. The node coding implies the relation of neighbouring nodes and their links and the global information of routing. In terms of the routing algorithm of GJCode(s, t), the packet passes nodes on the shortest distance path at all time. So, the packet passes through t+s links in the worst instance.

VII. CONCLUSIONS

The paper discusses characteristics and constructing methods of Gray Code, Johnson Code, hypercube and 2-degree torus, and brings forwards GJCode(s, t). Holding important properties of both Gray Code and Johnson Code, GJCode(s, t) topology has good scalable characteristic. when the degree of nodes is invariable, GJCode(s, t) can scales up in size and increases only 2t nodes while hypercube can’t. In the GJCode(s, t), the two parameters can balance well between the cost of networks and their performance in terms of the demand of application systems. The routing algorithm of GJCode(s, t) proposed can packets passes nodes on the shortest distance path at all time and is simple in implementation. The future research on GJCode(s,t) are parallel algorithms for it.

ACKNOWLEDGMENT

It is a project supported by the National Science Foundation of China under grant No.90607008. Xi’an Applied Material Innovation Fund Project (XA-AM-200615).

REFERENCES

[1] W. Dally and B. Towles, “Route packets, not wires: on-chip interconnection networks,” InProc. the Design Automation Conference, Las Vegas, NV, pp. 684-689,2001.
[2] R.L. Sharma, “Network Topology Optimization:The Art and Science of Network Design,” Van Nostrand Reinhold, 1990.
[3] Vanvoorst, B., Seidel, S. & Barscz, E, “Workload of an ipsc/860,” In Proceedings of the Scalable High Performance Computing Conference, pp 221 – 228, 1994.
[4] Hayes, J. P. & Mudge, T. N, “Hypercube Super computers”. PROC. IEEE 7(12) 1829 – 1841,1989.
[5]Tucker, L. W. & Robertson, G. G., “Architectureand applications of the connection machine,” IEEE Computer 21,pp 26 – 38.1988.
[6] SGI (1998), performance tuning optimization forOrigin2000 and Onyx2,http://tec hpubs.sgi.com/Library /manuals/3000 /0351101/html/O2000Tuning
[7] Duato, J., Yalamanchili, S., Ni, L.: Interconnection Networks: an Engineering Approach. Publishing House of Electronics Industry, Beijing 2004.
[8] W.J. Hsu, “Fibonacci Cubes-A New Interconnection Topology,”IEEE Trans. Parallel and Distributed Systems, vol. 4, no. 1, pp. 3-12,Jan. 1993.
[9] Peter K.K. Loh, Wen Jing Hsu, Yi Pan, “The Exchanged Hypercube”, IEEE TRANS. Parallel and Distributed Systems, VOL. 16, NO. 9, 2005.
[10] Amawy,.& Latifi., “Properties and performance of folded hypercubes ” , IEEE Trans. on Parallel and Distributed Systems 2, pp.31 – 42,1999.
[11] Efe, K., “The crossed cube architecture for parallel computation”, IEEE Trans. Parallel and Distributed Systems 3(5),pp. 513 – 524.1992.
[12] Preparata,F. P. & Vuillemin, J., “The cubeconnected cycles: a versatile network for parallel computation”, Communications of the ACM 24,pp. 300 – 309.1981.
[13] Ziavras, S. G, “RH: a versatile family of reduced hypercube interconnection networks”, IEEE Trans. Parallel and Distributed Systems 5 (11),1994.
[14] Sang Kyun Yun , “ Comments on Hierarchical cubic networks” IEEE Trans. Parallel and Distributed systems 9(4),pp. 10-414. 1998.
[15] J. Xu, A. Kumar, and X. Yu, “On the Fundamental Tradeoffs between Routing Table Size and Network Diameter in P2P Networks,” IEEE J. Selected Areas in Comm., vol. 22, no. 1,pp. 151-163, Jan. 2004.
[16] M.Venkata Rao1, Dr.N. Chalamaiah, “Hypercube and Its Variant Networks: A Topological Evaluation”, Proceedings of the Eighth International Conference on High-Performance Computing in Asia-Pacific Region (HPCASIA’2005).
[17] M.Venkat Rao, & N.Chalamaiah, “Routing and Broadcasting algorithms for a Metacube Interconne-cton topology for large scale parallel systems” in Asia-Pacificconference on Parallel & Distributed Computing Technologies. pp.1037-1049.2004.
[18] Li, & Peng, S., “Dual-Cube :A new interconnection for high performance computing” in proceedings of the International Computer symposium, workshop on computer architecture. pp. 51-57, 2000.

Xiaoqiang Yang was born in 1971. He received B.E. degree in computer science and technology from Liaoning Technical University, Liaoning China and M.E. degree in computer and technology from Xidian University, Xi’an China in 1995 and 2003 respectively. He is currently pursuing PhD degree in Micro-Electronics from from Xidian University, Xi’an China .

He works as a lecturer at Xi'an University of Science and Technology. His current research interests include parallel algorithms, parallel architecture, SoC Design and Verification.

Huimin DU was born in 1966. She received the M.S. and PhD degree in computer science the Northwestern Polytechnical University, Xi’an, China, in 1992 and 2000 respectively. From 2000 to 2002, she got a postdoctoral position in Northwestern Polytechnical University.

She is now Director of Micro-Electronics Research and Teaching Group at computer science department at Xi'an Institute of Posts and Telecommunications, Xi'an , China. Her research interests are mainly on ASIC design, formal verification for hardware design, and computer architecture, etc.

Jungang Han was born in 1943. He received the B.E. in Mathematics Department of Jilin University, China, in 1966 and M.S. in Institute of Computing Technology, Chinese Academy of Science, Beijing, P.R. China, in 1981.

From 1967 to 1975, he worked as a Electronic Engineer, the 795th Factory at Xian Yang, Shaanxi Province, China. From 1982 to 1993, he is an associate professor, Xidian University, Xi’an China. He is a visiting scholar, Computer Science Department, University of Calgary, Canada, Oct. 1986 to Sept. 1988. Since 1994 he has been on the faculty of the Department of Computer Science at Xi'an Institute of Post and Telecommunication, where he is a professor, director, and doctor tutor of Department of Computer Science.

He is active in formal design and verification of SoC, application technology of computer etc. He has published numerous journal and conference articles. From 1999, Professor Han received Chinese Government special subsidy,. He is a senior member of member of Chinese Computer Federation, Chinese Communication Federation, special committee of CAD&CG, CCF.