PAPER

HCC: Generalized Hierarchical Completely-Connected Networks

Toshinori TAKABATAKE†, *Student Member*, **Keiichi KANEKO**††, *and* **Hideo ITO**†††, *Regular Members*

SUMMARY In this paper, a new network structure called generalized Hierarchical Completely-Connected networks (HCCs) is proposed, and its properties and features are evaluated. Simple routing strategies for HCCs are also developed for shortest-paths routing algorithms. A set of HCCs constructed by the proposed method includes some conventional hierarchical networks, then it is called generalized one. The construction of an HCC starts from a basic block (a level-1 block) which consists of n nodes of constant degree. Then a level-h block for $h \geq 2$ is constructed recursively by interconnecting any pair of macro nodes (n level-$(h-1)$ blocks) completely. An HCC has a constant node-degree regardless of an increase in its size (the number of nodes). Furthermore, since an HCC has a hierarchically structured topology and the feature of uniformity, a wide variety of inter-cluster connections is possible. Evaluation results show that an HCC is suitable for very large computer systems.
key words: *interconnection networks, generalized network, hierarchical network, completely-connected network, routing strategy*

1. Introduction

In massively parallel computer systems, interconnection networks have become the center of focus because network structures and topologies as well as the processing elements greatly influence the system cost and performance [1], [2]. Among those network topologies, many hierarchical networks (i.e., tree or hypercube-based topologies) have been proposed [7]–[25]; they have the advantages of being able to deal with a huge number of nodes, as well as with reductions in the node-degree and/or the number of links.

These hierarchical networks have the characteristics of expansibility, scalability, regularity, uniformity, and symmetry [6]–[25]. Those characteristics suited for the systems are brought to the construction of the networks [1], [6]. The construction of the networks starts from a first level network topology (basic block), then a higher level is constructed recursively by inter-block connection. It should be noted that these hierarchical networks can be characterized by their basic blocks, and that can be classified into four classes according to whether the node-degree is constant at different levels or not and whether the construction-method is applicable to any level or not.

First, the basic blocks in WK-recursive networks (WK) [7]–[9], Multitriangle networks (MTR) [10], and Hypernets [11] are completely connected graphs, rings, and certain primitive topologies, respectively. The node-degree in WK, MTR, and Hypernet is constant at different levels. However, their diameters are relatively large. On the other hand, in Recursively connected networks (RCN) [15]–[17] and Block-switch networks (BSN) [18], although the basic blocks can be selected from any of the connected graphs, the construction-method of RCN and BSN makes the node-degree increase linearly and does not bring the structure of uniformity at different levels. Third, Hierarchical interconnection networks (HIN) [19], [20] and Extended hypercubes (EH) [21] have a particular network structure and/or the different node-degree at different levels. Fourth, Hierarchical hypercube (HHC) [22], Hierarchical cubic networks (HCN) [23], [24], and Cube-connected cycles (CCC) [25] are different from the first, second, and third classes. The construction-method of HHC, HCN, and CCC cannot be applied to any level so that these networks suffer from the lack of expansibility at the recursion level.

In this paper, a new network structure called generalized Hierarchical Completely-Connected networks (HCCs) is proposed, and its properties and features are evaluated. The proposed HCCs mainly possess the properties of WK and MTR mentioned above. A set of the HCCs constructed by the proposed method includes some conventional hierarchical networks, therefore it is possible to say that the proposed HCC method is a more generalized one. An HCC has a constant node-degree regardless of increase in size. Furthermore, an HCC has a hierarchically structured topology and the features of scalability, expansibility, regularity, uniformity, and symmetry. Simple routing strategies for the HCCs are also developed for shortest-paths routing algorithms.

The rest of this paper is organized as follows. Section 2 describes the definitions and topological properties of the HCCs. Section 3 shows the routing strategies for the HCCs. Section 4 evaluates an HCC and gives a comparison between the proposed network and others.

Manuscript received October 28, 1999.
†The author is with the Graduate School of Science and Technology, Chiba University, Chiba-shi, 263–8522 Japan.
††The author is with the Faculty of Technology, Tokyo University of Agri. and Tech., Koganei-shi, 184–8588 Japan.
†††The author is with the Faculty of Engineering, Chiba University, Chiba-shi, 263–8522 Japan.

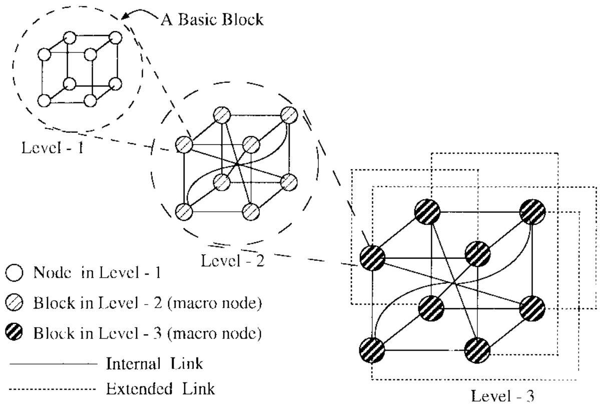

Fig. 1 The concept of an HCC.

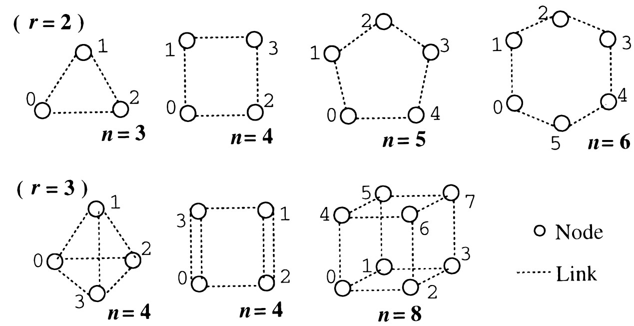

Fig. 2 Examples of basic blocks (r = 2, 3).

Section 5 summarizes and concludes this paper.

2. HCC Definitions and Topological Properties

To describe the networks as a graph, the terms and the notations used in [12]–[14] are adopted here for the HCCs. Let a network G be an undirected graph $G = (V, E)$, where V is the node set and E the link set. When the degree of a node is the number of links connected to the node, a graph G is called a regular graph if and only if the degrees of all nodes are the same. Given a network G, let a sequence of nodes $P = n_1, n_2, \cdots, n_k$ ($n_i \in V$) be a path from a node n_1 to n_k where $(n_i, n_{i+1}) \in E$ for $i = 1, \ldots, k-1$. For any pair of nodes x, $y \in V$, the distance between x and y denoted by $d(x, y)$ is the length of the shortest path from x to y. The diameter of G is the maximum value among the distances of all pairs of nodes x, $y \in V$. The following subsections explain how to make the complete inter-block connection and the final networks for HCCs.

2.1 Generalization of Hierarchical Complete Inter-block Connection

There are two important points in designing a network for a massively parallel computer system. The first is to have many adjacent clusters without changing the node configuration and link connection (expansibility). The second is to maintain the low node-degree for further expansion of the network (constant node-degree). The concept of an HCC network is shown in Fig. 1. Note that some internal links between two blocks are omitted in Fig. 1 and the explanation of some extended links are described in Sect. 2.2. Let h ($1 \leq h \leq L$) denote the number of hierarchical levels for an HCC construction, where L represents the maximum level. An HCC is constructed by starting from a basic block, which is defined as follows:

Definition 1: (Basic block G) A basic block is a connected graph G which is composed of n ($n \geq 2$) nodes satisfying the following two conditions: (i) The degree of each node is the same constant value r ($2 \leq r < n$), that is, a basic block G is regular. (ii) Each node is allocated a unique address from 0 to $n-1$.

Let a basic block G be a level-1 block. By Definition 1, it is possible to select a basic block structure among a wide variety of networks. This is one of the key points for our network to offer a generalized hierarchical complete inter-block connection. Figure 2 shows some basic blocks ($r = 2, 3$). Note that other basic blocks would be possible for $r = 2$ and 3.

Generally, a level-h ($h \geq 2$) is constructed by completely connecting n level-($h-1$) ($h \geq 2$) blocks by inter-block connection for the HCCs because the number of nodes of a basic block G is n. When a level-h block is constructed, the n level-($h-1$) blocks are often called macro nodes. The macro nodes are applied for constructing higher level blocks for further expansion of the network as shown in Fig. 1. Although the construction of many hierarchical networks has been proposed (see Sect. 1), definition 2 below is used for a level-h block in the HCCs. Definition 2 is also utilized for a generalized construction of the complete inter-block connection.

Definition 2: (A level-h block) Consider macro nodes which are n level-($h-1$) blocks, $B_0, \ldots, B_{n-1}$, for $h \geq 2$. All nodes in each B_i ($0 \leq i \leq n-1$) are addressed by $x_{h-1} \cdots x_k \cdots x_1$, where $x_k \in \{0, \ldots, n-1\}$ and $1 \leq k \leq h-1$. The subscript i of B_i ($0 \leq i \leq n-1$) denotes the address of the macro node. The addresses of all nodes in a level-h block are obtained by attaching a macro node address, $x_h = i$ ($0 \leq i \leq n-1$), to the addresses, $x_{h-1} \cdots x_1$, of all nodes in B_i. Then, a new level-h block is constructed by appending links for complete inter-block connection. The links are appended to the macro nodes so that there are links between pairs of nodes, X and Y, for all nodes in the new level-h block, if and only if one node X with the address $x_h x_{h-1} \cdots x_1 = ij \cdots j$ ($i \neq j$) which satisfies $x_{h-1} = \cdots = x_1 = j$, is connected to another node Y with the address $ji \cdots i$. Here, the appended link is called a block-internal link and is denoted by B-link.

The degree, the number of nodes, and that of links of a level-h block for the HCCs are given as follows:

Property 1: Let r be the node-degree of a basic block. For a level-h block, there are n nodes in which addresses $x_h x_{h-1} \cdots x_1$ satisfy $x_h = x_{h-1} = \cdots = x_1$, and the degree of the n nodes is r, the degree of any node except the n nodes is $r+1$. The total number of nodes is n^h and the total number of links is $(n^h(r+1)-n)/2$.

By Definition 2, the diameter of a level-h block for the HCCs is given as follows:

Property 2: Let $D(h-1)$ be the diameter of a level-$(h-1)$ block for the HCCs. For a level-h block, the diameter of a level-h block, denoted by $D(h)$, is bounded by $1+2D(h-1)$ since every pair of level-$(h-1)$ blocks are connected by a B-link of level-h. Solving $D(h) \leq 1+2D(h-1)$, then, the diameter of a level-h block for the HCCs is given by $D(h) \leq 2^{h-1}(D(1)+1)-1$, where $D(1)$ is the diameter of the basic block G.

2.2 Construction of the Maximum Level Networks

In this subsection, let us consider the construction of the maximum level networks by appending some links so that the degree of all nodes is the same in the resultant network. Then the following definition is used for the HCCs.

Definition 3: (Even block and odd block) Let $B_{even}(L)$ ($B_{odd}(L)$) be the maximum level block being constructed so that the number of nodes in a basic block G is even (odd). $B_{even}(L)$ ($B_{odd}(L)$) is called an even (odd) maximum level block.

2.2.1 Construction Method When n is Even

By Property 1, the method of construction of the final network from an $B_{even}(L)$ is defined as follows:

Definition 4: ($HCC_G(L)$) In $B_{even}(L)$, let $HCC_G(L)$ denote a final network being constructed so that the n nodes with degree r addressed by $i \cdots i$ are connected to the nodes addressed by $\overline{i} \cdots \overline{i}$, where $\overline{i}$ represents n's complement of digit i. A link appended between the two nodes, $i \cdots i$ and $\overline{i} \cdots \overline{i}$, is called an extended link and is denoted by E-link.

Figure 3 shows an example of $HCC_G(3)$ constructed by starting from a basic block G ($r = 2$, $n = 4$), by appending the 6 B-links to the 4 level-2 blocks (macro nodes), and by appending the 2 E-links to $B_{even}(3)$. As a result, all nodes in the $HCC_G(3)$ have the same node-degree of three. The following properties are obvious.

Property 3: In an $HCC_G(L)$, the degree of all nodes is $r+1$, the total number of nodes is n^L and the total number of links is $n^L(r+1)/2$.

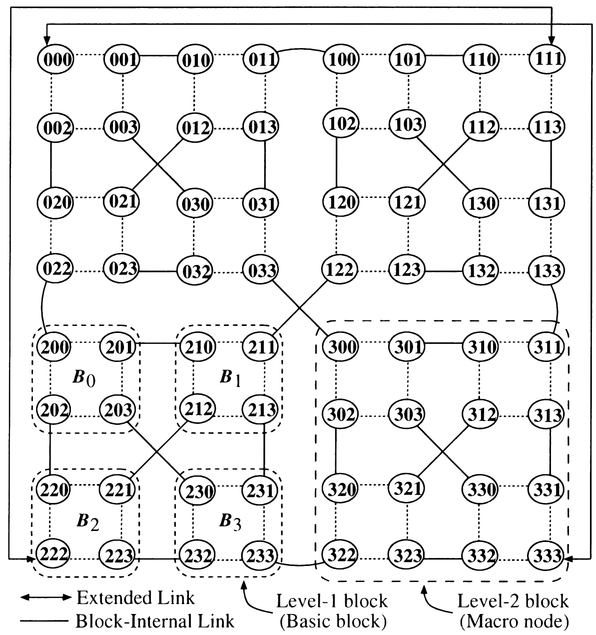

Fig. 3 An example of the final network ($HCC_G(3)$).

Property 4: An $HCC_G(L)$ is a regular graph if and only if it is constructed from $B_{even}(L)$.

2.2.2 Construction Methods When n is Odd

As shown in Fig. 4, there are four construction-methods when n is odd. Then, the final networks from an $B_{odd}(L)$ are defined as follows:

(a) : As shown in Fig. 4 (a), all nodes with degree of r (from Property 1) in $B_{odd}(L)$ have channels such as I/O ports, and the rest of the nodes do not have them.

(b) : As shown in Fig. 4 (b), the E-links are appended in the same way as in Definition 4. But in this case, there exists only the node with degree r that is not connected to any another node. Only that node has a channel such as an I/O port.

(c) : As shown in Fig. 4 (c), by adding one spare node to $B_{odd}(L)$, all nodes with degree r (from Property 1) in $B_{odd}(L)$ are connected to the spare node by E-links.

(d) : As shown in Fig. 4 (d), by adding one spare block (an arbitrary level-h block) to $B_{odd}(L)$, all nodes with degree r (from Property 1) in $B_{odd}(L)$ are connected to all nodes with degree r (from Property 1) in the spare block by E-links, where the spare block has the same basic block G structure as $B_{odd}(L)$ and there is an appended E-link between the two nodes with the same digit.

Definition 5: ($HCC_G(L)$-A, B, C, D) In $B_{odd}(L)$, let $HCC_G(L)$-A, $HCC_G(L)$-B, $HCC_G(L)$-C, and $HCC_G(L)$-D denote networks constructed based on the methods mentioned above (a), (b), (c), and (d), respectively. A set of all these networks is denoted by $HCC_G(L)$-A, B, C, D.

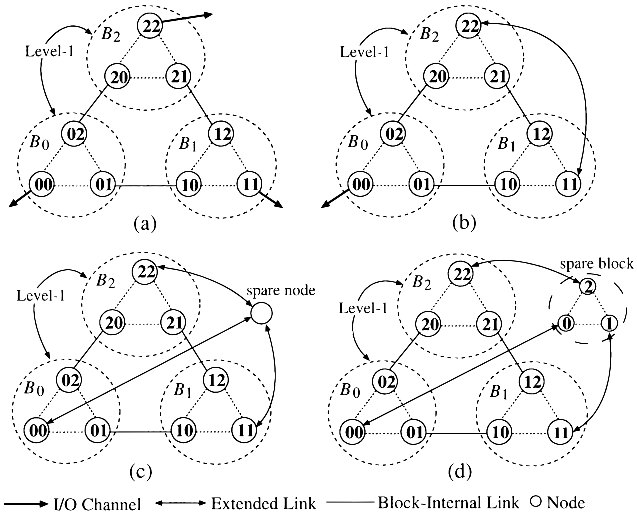

Fig. 4 Examples of the final networks ($HCC_G(2)$-A, B, C, D).

Table 1 Properties of the $HCC_G(L)$-A, B, C, D.

Network	No. of nodes	No. of links
$HCC_G(L)$-A	n^L	$(n^L(r+1)+n)/2$
$HCC_G(L)$-B	n^L	$(n^L(r+1)+n-2)/2$
$HCC_G(L)$-C	n^L+1	$(n^L+1)(r+1)/2$
$HCC_G(L)$-D	n^L+n^h	$(n^L+n^h)(r+1)/2$

Figure 4 shows examples of $HCC_G(2)$-A, B, C, D constructed by starting from a basic block G ($r=2$, $n=3$), and by the methods (a), (b), (c), and (d), respectively. $HCC_G(L)$-A and $HCC_G(L)$-B have I/O channels. However, since $HCC_G(L)$-A does not have an E-link, the reduction of the network diameter is not expected. On the other hand, since $HCC_G(L)$-C and $HCC_G(L)$-D can route a message through a spare node or a spare block, $HCC_G(L)$-D may be better in terms of fault tolerance. $HCC_G(L)$-A, B, C, D have a constant node-degree. Furthermore, they have the structure of uniformity, because any part of the block has the same structure as the other blocks such as shown in Fig. 4. The following properties are obvious.

Property 5: In $HCC_G(L)$-A, B, C, D, the degree of all nodes is $r+1$ and the number of nodes and that of links are shown in Table 1.

Property 6: $HCC_G(L)$-A, B, C, D denote the regular graph if and only if they are constructed from $B_{odd}(L)$.

Note that a reduction in diameter in the maximum level networks would be expected due to their appended E-links. However, the diameter of any HCC (the maximum level networks) cannot be derived from each class of $HCC_G(L)$ and $HCC_G(L)$-A, B, C, D because the construction of each class in HCCs has its own method of appended E-links. Therefore, we regard the diameter of any HCC as Property 2.

3. Routing Strategies for the HCCs

A routing strategy that is simple and effective in data communication is crucial for parallel systems. In addition, alternative routing paths are also required for the systems since a congestion problem could occur if there exist a large number of messages or faulty elements in the systems [1]. In this section, shortest-path routing algorithms based on the usage of B-links and E-links for routing on the HCCs are presented. In the HCCs, there are two kinds of routing paths owing to the construction of the HCCs. One is the path which does not use E-links, the other is the path which uses both B-links and E-links [14]. For simplicity, routing algorithms in the class of the $HCC_G(L)$ are considered as follows.

3.1 The Path not Using Extended Links

Routing strategies for the HCCs not using E-links of level-1 are feasible for application to any existing routing algorithms [1] because they only depend on the basic block structure. Therefore, in the following, the routings of shortest-paths based on a level-h ($h \geq 2$) block are shown. In a level-h block, consider a routing from two nodes, S to T, where $S = s_h \cdots s_1$ is the source node address and $T = t_h \cdots t_1$ is the destination node address, and assume $s_h \neq t_h$.

For the selected paths from S to T, the paths can be classified as the number of the using B-links of level-h, denoted by k'. Here, by Definition 2, let B_i ($0 \leq i \leq n-1$) be a macro node (a level-($h-1$) block), d_{B_i} be the distance of a macro node in B_i. Then, there exist two kinds of shortest-paths, denoted by $d_{k'}$ ($k'=1,2$), between S and T belonging to distinct blocks, B_i and B_j, as follows:

- Case ($k'=1$) : $d_1 = d_{k'} = d_{B_i} + d_{B_j} + 1 = d(s_{h-1}\cdots s_1, t_h \cdots t_h) + d(s_h \cdots s_h, t_{h-1}\cdots t_1) + 1$
- Case ($k'=2$) : $d_2 = d_{k'} = d_{B_i} + d_{B_u} + d_{B_j} + 2 = \min_u\{d(s_{h-1}\cdots s_1, u\cdots u) + d(s_h \cdots s_h, t_h \cdots t_h) + d(u\cdots u, t_{h-1}\cdots t_1) + 2\}$. Hence, determine d_2 when the macro node address u of the block B_u ($0 \leq u \leq n-1$) has been determined to be the distance.

In $k' \geq 3$, the shortest-paths cannot be given by d_3 as shown in [14]. Thus, in level-h, if d_1 and d_2 have been determined by the routing, then the shortest-path can be obtained by selecting the path given by $d(S,T) = \min\{d_1, d_2\}$.

3.2 The Path Using Extended Links

Under the same assumptions made above, how to use B-links and E-links for the shortest-paths based on a maximum level network is presented as follows.

For the selected paths from S to T, the paths can

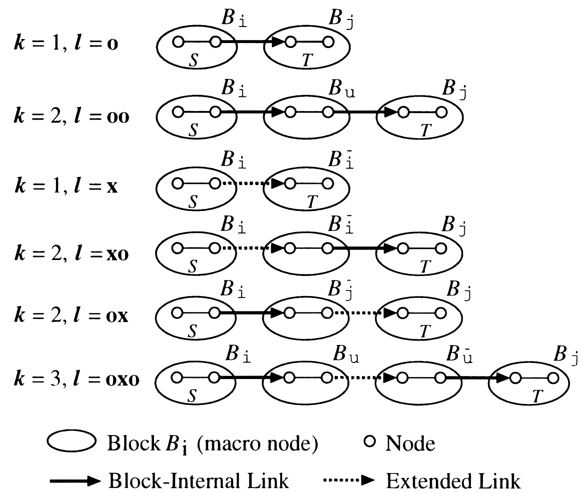

Fig. 5 Generalization of the routing paths for the HCCs.

be classified as the combination of the using the B-links and the E-links of level-L. Here, when k is the summation of the number of B-links and E-links on the using path from S to T of level-L, $\boldsymbol{l} = l_1 \cdots l_k$ denote the type and the order of use of the B-links or the E-links. Note that each l_i represents the link-type in the i-th order used. If the B-link or the E-link is used, then $l_i = \mathtt{o}$ or $l_i = \mathtt{x}$, respectively. Now, by Definition 4, the shortest-paths are classified and given as follows:

• Case ($k = 1$, $\boldsymbol{l} = \mathtt{o}$) : $d_{\mathtt{o}} = d_1$ (from Sect. 3.1)
• Case ($k = 2$, $\boldsymbol{l} = \mathtt{oo}$) : $d_{\mathtt{oo}} = d_2$ (from Sect. 3.1)
• Case ($k = 1$, $\boldsymbol{l} = \mathtt{x}$) : $d_{\mathtt{x}} = d(s_{L-1} \cdots s_1, s_L \cdots s_L) + d(t_L \cdots t_L, t_{L-1} \cdots t_1) + 1 \quad (s_L = \overline{t}_L)$
• Case ($k = 2$, $\boldsymbol{l} = \mathtt{xo}$) : $d_{\mathtt{xo}} = d(s_{L-1} \cdots s_1, s_L \cdots s_L) + d(\overline{s}_L \cdots \overline{s}_L, t_L \cdots t_L) + d(\overline{s}_L \cdots \overline{s}_L, t_{L-1} \cdots t_1) + 2 \quad (s_L \neq \overline{t}_L)$
• Case ($k = 2$, $\boldsymbol{l} = \mathtt{ox}$) : $d_{\mathtt{ox}} = d(s_{L-1} \cdots s_1, \overline{t}_L \cdots \overline{t}_L) + d(s_L \cdots s_L, \overline{t}_L \cdots \overline{t}_L) + d(t_L \cdots t_L, t_{L-1} \cdots t_1) + 2 \quad (s_L \neq \overline{t}_L)$
• Case ($k = 3$, $\boldsymbol{l} = \mathtt{oxo}$) : $d_{\mathtt{oxo}} = \min_u\{d(s_{L-1} \cdots s_1, u \cdots u) + d(s_L \cdots s_L, u \cdots u) + d(\overline{u} \cdots \overline{u}, t_L \cdots t_L) + d(\overline{u} \cdots \overline{u}, t_{L-1} \cdots t_1) + 3\} \quad (u \neq s_L, \overline{t}_L)$. Hence, determine $d_{\mathtt{oxo}}$ when the macro node address u of the block B_u $(0 \leq u \leq n-1)$ has been determined to be the distance. Especially, if $s_L \neq t_L$, then $d_{\mathtt{oxo}} > d_{\mathtt{x}}$, it is thus not necessary to consider.

In the case $k = 3$ of $\boldsymbol{l} = \mathtt{ooo}, \mathtt{xoo}, \mathtt{oox}, \mathtt{xox}$, these paths are not provided with the shortest paths [14]. In case $k = 4$, there are 8 possible routing paths such as $\boldsymbol{l} = \mathtt{oooo}, \mathtt{xooo}, \mathtt{oxoo}, \mathtt{ooxo}, \mathtt{ooox}, \mathtt{xoxo}, \mathtt{xoox}, \mathtt{oxox}$, and the each path contains the combination of $\boldsymbol{l} = \mathtt{ooo}, \mathtt{xoo}, \mathtt{xoo}, \mathtt{oox}, \mathtt{ooo}, \mathtt{xox}, \mathtt{xoo}, \mathtt{xox}$ in this order, respectively. In addition, in the case $k = 3$, the shortest paths which cannot be provided have been shown. Therefore, shortest paths in the case $k = 4$ are not provided. In the same way, in case $k \geq 5$, they need not be considered.

According to the above discussion, if $a_L = \overline{b}_L$, then the shortest path is given by $d(S,T) = \min\{d_{\mathtt{o}}, d_{\mathtt{oo}}, d_{\mathtt{x}}\}$, if $a_L \neq \overline{b}_L$, then the shortest path is given by $d(S,T) = \min\{d_{\mathtt{o}}, d_{\mathtt{oo}}, d_{\mathtt{xo}}, d_{\mathtt{ox}}, d_{\mathtt{oxo}}\}$. Therefore, it is possible to obtain the shortest path in the HCCs. Figure 5 shows the shortest paths in the HCCs. Note that the routing of shortest paths in any class of the HCCs could be based on the algorithms mentioned above. Other classes of HCC routing algorithms remain subjects of future work.

3.3 Routing Algorithms for the HCCs

From the previous two subsections, the routing algorithms for the HCCs are given as follows:

Routing Algorithms for the HCCs

The function $d = (S,T)$ provide the distance between S and T, where $S = s_L \cdots s_1$ is the source node address and $T = t_L \cdots t_1$ the destination node address.

function $d(s_h \cdots s_1, t_h \cdots t_1)$
if $h = 1$ **then** *return a basic block* $d(s_1, t_1)$
else if $s_h = t_h$ **then** *return* $d(s_{h-1} \cdots s_1, t_{h-1} \cdots t_1)$
else *return* $\min\{d_{\mathtt{o}}, d_{\mathtt{oo}}\}$;

Procedure $R(s_L \cdots s_1, t_L \cdots t_1)$
  if $s_L = t_L$ **then** $R_1(s_{L-1} \cdots s_1, t_{L-1} \cdots t_1)$
  else begin
    if $s_L = \overline{t}_L$ **then** *Select a path by* $\min\{d_{\mathtt{o}}, d_{\mathtt{oo}}, d_{\mathtt{x}}\}$, *decided by* R_1.
    else *Select a path by* $\min\{d_{\mathtt{o}}, d_{\mathtt{oo}}, d_{\mathtt{xo}}, d_{\mathtt{ox}}, d_{\mathtt{oxo}}\}$, *decided by* R_1.
  end;
end;

Procedure $R_1(s_h \cdots s_1, t_h \cdots t_1)$
  if $h = 1$ **then** *Select a path on a basic block* G.
  else if $s_h = t_h$ **then** $R_1(s_{h-1} \cdots s_1, t_{h-1} \cdots t_1)$
  else begin
    Select a path by $\min\{d_{\mathtt{o}}, d_{\mathtt{oo}}\}$, *decided by* d.
  end;
end;

4. Evaluations

This section describes the properties and the features of HCCs, and the comparison with other networks.

4.1 Properties and Features of the HCCs

In this subsection, the features of the proposed HCC are described to compare it with other hierarchical networks listed in Table 2. Also listed in Table 2 are the features of the proposed HCC, WK, MTR, Hypernet, RCN, BSN, HIN, EH, HCN, HHC, and CCC about the level-1 network basic block (topology), a hierarchical structure at each level, the node-degree type, and the classification of the networks as described in Sect. 1.

• **Basic Block :** In an HCC, when choosing a basic block G (the degree and the number of nodes) for

Table 2 Features of the HCCs and other hierarchical networks.

Network	Level-1 (Basic block)	Level-2	Level-h ($h \geq 3$)	Node-degree type	Class
HCC	Regular connected graph	G_c	G_c	Constant	I
WK [7]–[9]	Completely connected graph	G_c	G_c	Constant	I
MTR [10]	Ring	G_c	G_c	Constant	I
Hypernet [11]	Hypercube (H), bus, tree	G_c	G_c	Constant	I
RCN [15]–[17]	Any connected graph	G'_c	G''_c	Linearly	II
BSN [18]	Any connected graph	G'_c	G''_c	Linearly	II
HIN [19], [20]	Any network (K_1)	K_2	K_h	Different	III
EH [21]	Hypercube-based Pyramid (P_1)	P_2	P_h	Different	III
HHC [22]	Hypercube (H_1)	Hypercube (H_2)	–	Constant	IV
HCN [23], [24]	Hypercube (H_1)	Hypercube (H_1)	–	Linearly	IV
CCC [25]	Ring	Hypercube	–	Constant	IV

G_c: Completely connected graph

the design of a parallel system, there exists a trade-off among the number of nodes, the number of links, and the diameter or the average distance [13], [14], [18]. Thus it is important that the physical constraints when choosing a basic block structure from among a wide variety of network topologies can be adjusted for implementing a system. The entries in the columns of Level-1 in Table 2 represent the network topologies. For HCC, RCN, BSN, and HIN, it is possible to select a basic block structure among a wide variety of networks. Therefore, the choice of a basic block structure is not very restricted, permitting, hypercubes, rings, or completely connected graphs.

• **Expansibility :** Realization of many adjacent clusters without changing the node configuration and link connection affects the VLSI implementation and cost [1], [2], [6]. The entries in the columns of Level-2 and Level-h in Table 2 represent the network structures of the macro node; the same (or different) notations in a row denote the same (or different) network structure. For example, the two G_cs in the row of Hypernet are the same. Although H_1 and H_2 in the row of HHC are both hypercubes, the network structures are not necessarily the same. G'_c and G''_c in the row of RCN and BSN are highly different structures so they lose their regularity and uniformity in the level of recursion. K_2 and K_h in the row of HIN used either at different levels or at each level may have different topologies (clusters). P_2 and P_h in the row of EH are pyramid-like connected structures. Thus, HCC (from Properties 4 and 6) is superior to RCN, BSN, HIN, and EH as regards expansibility, regularity, and uniformity.

• **Scalability :** A scalable architecture implies that more processors can be added in the system [1], [2], [6]. The construction of HCC, WK, MTR, Hypernet, RCN, BSN, HIN, and EH can be applied to any level of the networks. Thus, these networks are different from HCN, HHC, and CCC. However, RCN and Hypernet suffer due to their low scalability when the level increases. The scalability of HCC is better than that of RCN and Hypernet (shown in Fig. 6), where HCC, RCN, and Hypernet used for comparison all have 8 nodes for each basic block. The scalability of HCC, WK, MTR, BSN, HIN, and EH is similar, i.e., $N_{h+1} = N_1 \times N_h$, where N_1 is the basic block size and h is the level.

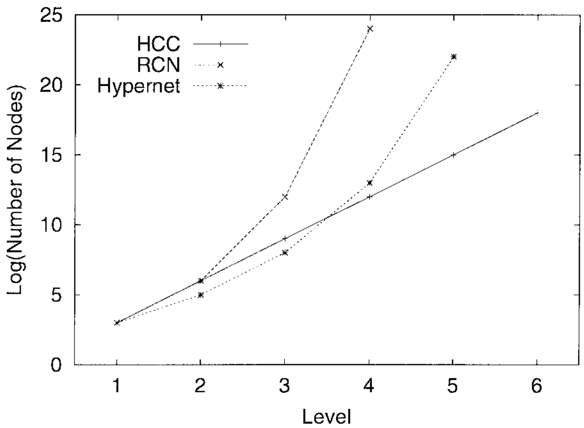

Fig. 6 Comparison with the scalability.

• **Node Degree :** A constant node-degree implies the expansibility and uniformity without any hardware change of each node for further expansion of the network [1], [2], [6]. An HCC always has the constant node-degree of $r+1$ for all nodes (from Properties 3 and 5) regardless of whether the number of nodes in a basic block G is even or odd. On the other hand, RCN and BSN, HIN and EH, or HCN are not the constant node-degree type because the construction method of these networks requires additional links to be appended for each node when the level or the network size increases. The entries in the columns of Node-degree type in Table 2 represent the type of characteristics according to the increase in network size.

According to the above discussion, HCC includes WK, MTR, and Hypernet (in the case of hypercubes in Level-1), and it is different from networks not of the constant node-degree type; furthermore, which do not apply to any level as shown in Table 2. As a result, we can conclude that HCC is different from other conventional Level-2 networks and includes conventional multi-level hierarchical networks. HCC is superior to other hierarchical networks in terms of constant node-degree, regularity, scalability, symmetry, uniformity,

Table 3 The architectural features of the HCCs and other hierarchical networks.

Network	Nodes (N)	Degree (Δ)	Diameter (D)	Links (E)
$\mathrm{HCC}_G(L)$	N_G^L	$r+1$	$2^{L-1}(D_G(1)+1)-1$	$n^L(r+1)/2$
$\mathrm{WK}(k,L)$ [7]–[9]	W^L, $(k=W)$	W	2^L-1	$W(W^L+1)/2$
MTR [10]	3^L	3	2^L-1	$3^{L+1}/2$
Hypernet [11]	$2^{2^{L-1}(d-2)+L+1}$	$d+1$	$2^{L-1}(d+1)-1$	#1
$\mathrm{RCN}(A,L)$ [15]–[17]	$N_A^{2^L}$	Δ_A+L-1	$2^{L-1}(D_A(1)+1)-1$	#2
$\mathrm{BSN}(G,L)$ [18]	N_G^L	Δ_G+L-1	$L(D_G(1)+1)-1$	$(L-1)\frac{N^L-N^{L-1}}{2}+N^{L-1}E_1$
HIN [19], [20]	2^D,$(d=\log n=\log(N/K))$	D	$(D-d)2^{D-d-1}$	$d2^{D-1}+(D-d)2^{D-d-1}$
$\mathrm{EH}(d,L)$ [21]	2^m, $(m=d\times L)$	#3	$d+2(L-1)$	$(d/2+1)[\frac{N^L-1}{1-2^{-d}}]$
$\mathrm{HHC}(m,d)$ [22]	2^m, $(m=2^d+d)$	$d+1$	2^{d+1}	$(d+1)2^{m-1}$
$\mathrm{HCN}(d,d)$ [23], [24]	2^{2d}	$d+1$	$d+\lfloor(d+1)/3\rfloor+1$	$(d+1)2^{2d-1}$
$\mathrm{CCC}(c,d)$ [25]	$c2^d$	3	$\lfloor(5c-2)/2\rfloor$	$\frac{3}{2}N$
$\mathrm{GHC}(w,d)$ [4], [5]	w^d	$(w-1)\log_w N$	$\log_w N$	$\frac{N(w-1)}{2}\log_w N$
$\mathrm{Torus}(w,d)$ [2], [3]	w^d	$2\log_w N$	$\frac{w}{2}\log_w N$	$N\log_w N$

#1 $\begin{cases} E_L = E_{L-1}S_L + S_L(S_{L-1}-1)/2 \quad (L\ge 2) \\ S_L = 2^{2^{L-2}(d-2)+1}, E_1 = d2^{d-1} \quad (L \ge 2) \end{cases}$ #2 $\begin{cases} E_L = N^{1/2}E_{L-1}+E' \quad (L\ge 2), \\ E' = N^{1/2}(N^{1/2}-1)/2 \end{cases}$ #3 $\begin{cases} \Delta_N = d+1 \text{ (for all } L) \\ \Delta_{NC} = 2^d+d+1 \text{ (for all } L) \end{cases}$

and expansibility.

Table 3 lists the architectural features of these hierarchical networks. The measures are the number of nodes or network size (N), degree (Δ), diameter (D), and the number of links (E). Note that the number of levels in these networks is given by L, and that d represents the number of dimensions; the base w numbers giving its coordinates. It is assumed that the proposed network is $\mathrm{HCC}_G(L)$, and that HIN is a level-two network (cluster size K) having a hypercube network as a basic block with 8 nodes [19], [20]. In Table 3, the entries in the column of Links or Degree are used in the following: in the hypernet (#1), S represents the number of subnets [11]; in RCN (#2), N the number of nodes of level-L [15]–[17]; in BSN, E_1 the number of links of level-1 [18]; in EH (#3), Δ_N and Δ_{NC} each node-degree and each controllers one as fixed d [21], respectively. For reference, Generalized Hypercube [4], [5] and Torus [2], [3] are also indicated in Table 3.

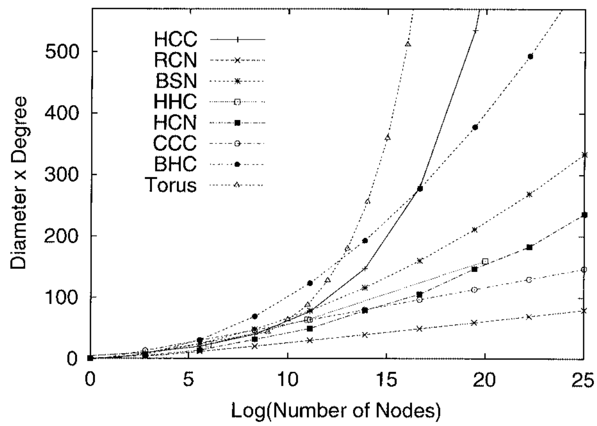

Fig. 7 Comparison with the costs (diameter × degree).

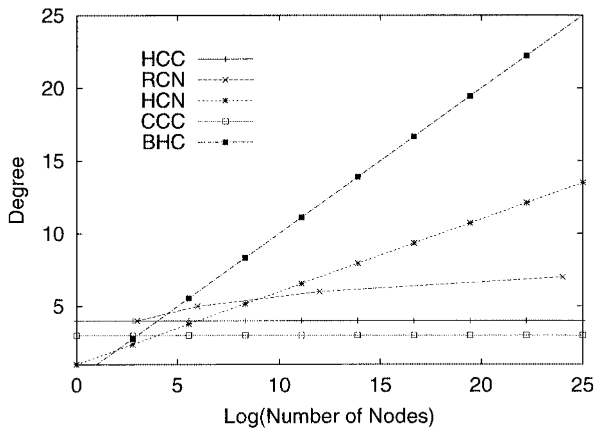

Fig. 8 Comparison with the degrees.

4.2 Comparison with Other Networks

In this subsection, an HCC is evaluated by comparing it with other hierarchical networks in terms of the standard performance measures, such as cost as the product of degree and diameter, degree, and the number of links with increasing the number of nodes (as a function of the network size) [2]. For simplicity, the HCC with a basic block G ($r=3, n=8$) is used for comparison. The other networks indicated in Table 3, such as the RCN, BSN, HHC (setting $d=3$), HCN, CCC ($c=d$), Binary Hypercube (BHC, $w=2$), and two-dimensional torus (Torus, $d=2$) are also used. Since the WK, MTR, and Hypernet are included in the HCCs and since the HIN and EH are not completely connected structures at any level, these networks are not used for comparison.

The cost (diameter×degree) is a criterion to measure the cost and performance of a computer system [1], [2], [6]. In Fig. 7, although the cost of the HCC is relatively higher than that of the other networks, the performance of the HCC is satisfactory up to about 30 thousand nodes. The node-degree reflects the number of I/O ports required per node, so it is regarded as the cost of a node [2], [6]. In Fig. 8, both the HCC and CCC have a constant node degree. Note that the degree of

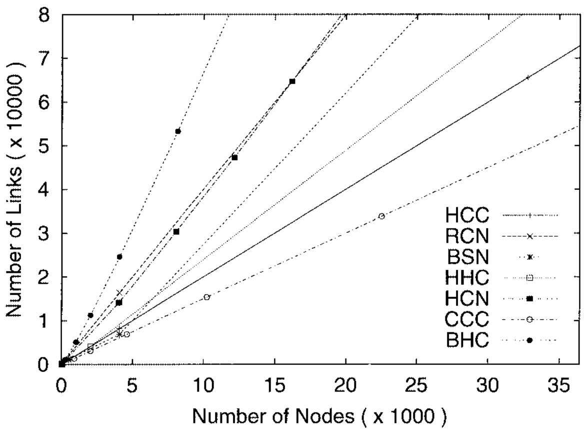

Fig. 9 Comparison with the number of links.

the HHC and Torus is the same as that of the HCC; that of the BSN is the same as that of the RCN. The total number of links gives a rough measure of the hardware cost of a network. However, this measure enables the fundamental global cost estimation of the network [2], [6]. In Fig. 9, the HCC performance is second to that of CCC. Note that the number of links in HCC is mostly equivalent to that in Torus; that in the HCC which has the basic block with $r = 2$ and $n = 4$ is the same as that in CCC.

According to the above discussions, an HCC shows relatively higher cost than some hierarchical networks. However, an HCC has the advantages of maintaining a low node-degree and a low number of links regardless of whether the number of nodes increase. Furthermore, considering current trends in network topology using mesh and torus, an HCC has satisfactory in terms of the evaluated performance measures for other networks. In view of implementing ULSI or VLSI, HCC is applicable to very large next-generation computer systems.

5. Conclusions

In this paper, a new network structure called generalized Hierarchical Completely-Connected networks (HCCs) has been proposed, and its properties and features are evaluated to compare it with other well-known hierarchical networks. Simple routing strategies for HCCs have been developed for shortest-paths routing algorithms. A set of HCCs constructed by the proposed method includes some conventional multi-hierarchical networks though it excludes two-level networks. HCC networks have the property of constant node-degree regardless of the network size, and have the structure of uniformity. Thus HCC networks are suited to very large-scale next-generation computer systems.

Further research issues on HCC networks remain to be explored; these include exploiting routing algorithms under various switching techniques and implementing a method of data communications considering the prevention of deadlock and livelock.

References

[1] J. Duato, S. Yalamanchili, and L. Ni, Interconnection networks — An engineering approach, IEEE Computer Society Press, 1997.
[2] D.A. Reed and R.M. Fujimoto, Multicomputer networks: Message-based parallel processing, Cambridge, MA: MIT Press, 1987.
[3] W.J. Dally, "Performance analysis of k-ary n-cube interconnection networks," IEEE Trans. Comput., vol.39, no.6, pp.775–785, June 1990.
[4] L.N. Bhuyan and D.P. Agrawal, "Generalized hypercube and hyperbus structures for a computer network," IEEE Trans. Comput., vol.C-33, no.4, pp.323–333, April 1984.
[5] Y. Saad and M.H. Schultz, "Topological properties of hypercubes," IEEE Trans. Comput., vol.37, no.7, pp.863–872, July 1988.
[6] V. Cantoni, M. Ferretti, and L. Lombardi, "A comparison of homogeneous hierarchical interconnection structures," Proc. IEEE, vol.79, no.4, pp.416–428, April 1991.
[7] G.-H. Chen and D.-R. Duh, "Topological properties, communication, and computation on WK-recursive networks," Networks, vol.24, no.6, pp.303–317, 1994.
[8] R. Fernandes, "Recursive interconnection networks for multicomputer networks," Proc. 21st Int. Conf. on Parallel Processing, pp.76–79, 1992.
[9] G.V. Vecchia and C. Sanges, "A recursively scalable network VLSI implementation," Future Generation Computer Systems, vol.4, pp.235–243, 1988.
[10] J. Wu and H. Qian, "Multitriangle: A constant node degree interconnection network," Int. J. Foundations of Computer Science, vol.8, no.2, pp.187–209, 1997.
[11] K. Hwang and J. Ghosh, "Hypernet: A communication-efficient architecture for constructing massively parallel computers," IEEE Trans. Comput., vol.C-36, no.12, pp.1450–1466, Dec. 1987.
[12] T. Takabatake, K. Kaneko, and H. Ito, "Generalized hierarchical completely-connected networks," Proc. 4th Int. Symp. on Parallel Architectures, Algorithms, and Networks, pp.68–73, June 1999.
[13] T. Takabatake, K. Kaneko, and H. Ito, "Hierarchical completely-connected networks," IEICE Technical Report, FIIS98-41, Sept. 1998.
[14] T. Takabatake, K. Kaneko, and H. Ito, "Hierarchical completely-connected cubic networks," IEICE Technical Report, FTS98-62, Aug. 1998.
[15] M. Hamdi, "A class of recursive interconnection networks: Architectural characteristics and hardware cost," IEEE Trans. Circuits & Syst. –I: Fundamental Theory and Applications, vol.41, no.12, pp.805–816, Dec. 1994.
[16] M. Hamdi and R.W. Hall, "An efficient class of interconnection networks for parallel computations," The Computer Journal, vol.37, no.3, pp.207–218, 1994.
[17] M. Hamdi and R.W. Hall, "RCC-Full: An effective network for parallel computations," J. Parallel and Distributed Computing, vol.41, pp.139–155, 1997.
[18] W.K. Chiang and R.J. Chen, "Block-switch networks: A cost-effective class of interconnection networks," Computer Systems Science and Engineering, vol.12, no.3, pp.175–185, May 1997.
[19] S.P. Dandamudi and D.L. Eager, "Hierarchical interconnection networks for multicomputer systems," IEEE Trans. Comput., vol.39, no.6, pp.786–797, June 1990.
[20] S.P. Dandamudi and D.L. Eager, "On hypercube based hierarchical interconnection network design," J. Parallel and

Distributed Computing, vol.12, pp.283–289, 1991.

[21] J.M. Kumar and L.M. Patnaik, "Extended hypercubes: A hierarchical interconnection network of hypercubes," IEEE Trans. Parallel and Distributed Systems, vol.3, no.1, pp.45–57, Jan. 1992.

[22] Q.M. Malluhi and M.A. Bayoumi, "The hierarchical hypercube: A new interconnection topology for massively parallel systems," IEEE Trans. Parallel and Distributed Systems, vol.5, no.1, pp.17–30, Jan. 1994.

[23] K. Ghose and K.R. Desai, "Hierarchical cubic networks," IEEE Trans. Parallel and Distributed Systems, vol.6, no.4, pp.427–435, April 1995.

[24] S.-K. Yun and K.H. Park, "The optimal routing algorithm in hierarchical cubic network and its properties," IEICE Trans. Inf. & Syst., vol.E78-D, no.4, pp.436–443, April 1995.

[25] F.P. Preparata and J. Vuillemin, "The cube-connected cycles: A versatile network for parallel computation," Commun. ACM, vol.24, no.5, pp.300–309, May 1981.

Toshinori Takabatake received the B.BAISc. and M.BAISc. degrees from Tokyo University of Information Sciences in 1995 and 1997, respectively. He is currently a student of doctor course in Chiba University, Japan. His research interests include massively parallel computer systems, interconnection networks, and fault tolerant routing systems. He is a member of IPSJ, IEEE, and ACM.

Keiichi Kaneko received the B.E., M.E., and Ph.D. degrees from the University of Tokyo in 1985, 1987, and 1994, respectively. In 1987, he joined the University of Tokyo as a research associate. In 1996, he joined Chiba University as a lecturer. Since 1999, he has been an Associate Professor in University of Tokyo Agriculture and technology. His main research areas are functional programming language, software engineering, parallel and distributed computation, and fault tolerant systems and other related areas. He is a member of ACM, IPSJ, and JSSST.

Hideo Ito received the B.E. degree from Chiba University in 1969, and the D.E. degree from Tokyo Institute of Technology in 1984. He joined Nippon Electric Co. Ltd. in 1969, and Kisarazu Technical College in 1971. Since 1973, he has been a member of the Chiba University, where he is currently a Professor in the Department of Information and Image Sciences. His research interests include easily testable design, test generation, VLSI architecture, fault-tolerant VLSI design of parallel & distributed systems, and defect-tolerant VLSI design and other related areas. He is a member of IPSJ and the IEEE Computer Society.