Embedding Algorithms for Bubble-Sort, Macro-star, and Transposition Graphs

HyeongOk Lee[1], Hyun Sim[1], JungHyun Seo[2], and Mihye Kim[3,*]

[1] Department of Computer Education & Engineering, Sunchon National University, 413 Jungangno Suncheon Chonnam, South Korea
{oklee,simhyun}@scnu.ac.kr

[2] Research & Development Team, Jeonnam Technopark, 315 Maegokdong Suncheon Chonnam, South Korea
jhseo@jntp.or.kr

[3] Department of Computer Science Education, Catholic University of Daegu, 330 Hayangeup Gyeonsansi Gyeongbuk, South Korea
mihyekim@cu.ac.kr

Abstract. Bubble-sort, macro-star, and transposition graphs are interconnection networks with the advantages of star graphs in terms of improving the network cost of a hypercube. These graphs contain a star graph as their sub-graph, and have node symmetry, maximum fault tolerance, and recursive partition properties. This study proposes embedding methods for these graphs based on graph definitions, and shows that a bubble-sort graph B_n can be embedded in a transposition graph T_n with dilation 1 and expansion 1. In contrast, a macro-star graph $MS(2, n)$ can be embedded in a transposition graph with dilation n, but with an average dilation of 2 or under.

Keywords: Interconnection network, Embedding, Dilation.

1 Introduction

Applications in engineering and scientific fields such as artificial intelligence, CAD/CAM (Computer-Aided Design and Computer-Aided Manufacturing), and fluid mechanics require hundreds of operations for data processing, which has led to increased interest in high-performance computers with a large number of processors. Consequently, interest in parallel processing, in which more than one processor simultaneously executes multiple tasks or a part of one program, has risen dramatically. However, it has proven challenging to design effective parallel algorithms. Parallel algorithms are usually designed for a particular parallel computer architecture, so to optimize algorithm design it is necessary to understand parallel computer architectures. There are a number of major architectures, and several methods exist for classifying them. One of the most well-known methods is Flynn's taxonomy, which categorizes architectures into four groups based on the number of instruction and data streams available in the architecture. Of the four types, MIMD (Multiple Instruction,

* Corresponding author.

C. Ding, Z. Shao, and R. Zheng (Eds.): NPC 2010, LNCS 6289, pp. 134–143, 2010.

Multiple Data streams) is the most common and it can simultaneously execute multiple instruction and data streams within one system.

MIMD parallel computers can be classified into two types based on memory architecture: multi-processor systems with shared memory and multi-computer systems with distributed memory. In a multi-computer system, each process has its own memory and is connected to other processors via an interconnection network. In a multi-computer system, each process has its own memory and is connected to other processors via an interconnection network. Inter-processor communication is achieved by sending messages among computers through the network, and a data-driven system is used for computations [1], [2]. The performance of the multi-computer system depends on the performance of each processor as well as the structure of the interconnection network and the applied algorithms. The characteristics of the interconnection network greatly influence overall performance and scalability of a multi-computer system. The most well-known topologies of interconnection networks are the mesh, hypercube, and star graph. The commonly used parameters for evaluating the performance of interconnection networks are degree, diameter, symmetry, scalability, fault tolerance, and embedding.

The embedding of interconnection networks is intended to analyze the interrelationship between graphs to observe whether a certain graph G is included in or interrelated with another graph H. The evaluation of embedding is significant: if graph G can be efficiently embedded in graph H with less cost, then the method developed in the interconnection network with graph G can be used in the interconnection network with graph H at less cost [3], [4]. In previous works [5], [6], we analyzed embedding methods for some star variations such as star, matrix-star, Rotator-Faber-Moore, and pancake graphs. In this paper we analyzed embedding methods for bubble-sort, macro-star, and transposition graphs, which are well-known as variations of the star graph. This work extended earlier work to develop a method for embedding between bubble-sort and transposition graphs [7].

2 Related Work

An interconnection network can be represented as an undirected graph $G = (V, E)$, with each processor presented as a node (vertex) v of G, and the communication channel between those processors presented as an edge (v, w). $V(G)$ and $E(G)$ represent the set of nodes and edges of graph G, respectively. That is, $V(G) = \{0, 1, 2, \ldots, n-1\}$ and $E(G)$ consists of pairs of distinct nodes from $V(G)$. There exists an edge (v, w) between two nodes v and w of G if and only if a communication channel between v and w exists [8]. If we classify the interconnection networks proposed up to now, we can divide them into the mesh variation with $n \times k$ nodes [9], the hypercube variation with 2^n nodes [2], [3], [10], the star graph variation with $n!$ nodes [8], and the odd graph variation with combination ${}_{2n}C_n$ nodes [4]. A variation of the star graph represents nodes using n distinct symbols, and the number of nodes is approximately $n!$ nodes. Star [8], [11], bubble-sort [12], pancake [11], transposition [13], macro-star [14], rotator [15], and Faber-Moore [16] graphs have been proposed as variations of the star graph. The graphs have a smaller node degree and diameter than a hypercube with a similar number of nodes.

A macro-star graph MS [14] is an interconnection network that improves the network cost of a star graph by generalizing the star graph. The size and degree of MS are determined by parameters l and n. A macro-star graph $MS(l, n)$ has $(nl+1)!$ nodes, $(n+l-1)$ degree, and $(nl+1)! * (n+l-1)$ edges. The address of each node is represented as a permutation of k $(=nl+1)$ distinct symbols. In other words, a node corresponds to a permutation. An edge exists between nodes u and v in $MS(l, n)$ if and only if the permutation of node v can be obtained from that of node u by applying each of, as defined below, two edge generators T_j and S_i where $2 \le j \le n+1$ and $2 \le i \le l$. A macro-star graph $MS(l, n)$ can be defined as shown in Eq. (1), where k distinct symbols $<K> = \{1, 2, .., k\}$, and a permutation of $<K>$, $U = u_{1:k} = u_1u_2...u_i...u_k$, $u_i \in <K>$.

$$V(MS(l,n))=\{U=u_{1:k} \mid u_i,u_j \in <K>,\ u_i \neq u_j,\ i \neq j,\ 1 \le i,j \le k\},$$
$$E(MS(l,n))=\{(U,V) \mid U,V \in V(MS(l,n)) \text{ satisfying } U=T_j(V) \text{ or } U=S_i(V),\ 2 \le j \le n+1,\ 2 \le i \le l\}. \quad (1)$$

Two edge generators T_j and $S_{n,i}$ are defined in the macro-star graph $MS(l, n)$ to formulate a link (edge) from a node to another node. The edge generator T_j is defined to create a permutation by interchanging the first symbol u_1 with the j^{th} symbol (u_j) of a given node. With a given node $U = u_{1:k} = u_1u_2...u_i...u_k$, the permutation of the node generated by T_j will be $T_j(U) = u_ju_{2:j-1}u_1u_{j+1:k}$. Another edge generator $S_{n,i}$ is defined to create a permutation by interchanging the sequence of symbols $u_{(i-1)n+2:in+1}$ with the sequence of symbols $u_{2:n+1}$ in a given node. With a given node $U = u_{1:k}$, the permutation of the node generated by edge $S_{n,i}$ will be $S_{n,i}(u_{1:k}) = u_1u_{(i-1)n+2:in+1}u_{n+2:(i-1)n+1}u_{2:n+1}u_{in+2:k}$. The edge generator $S_{n,i}$ is represented simply as S_i. Here, the symbol sequence $u_{(i-1)n+2:in+1}$ is referred to as a cluster [14].

Fig. 1a shows a top view of an $MS(2, 2)$ graph, while Fig. 1b presents the details of the level 2 cluster '23'. Each circle corresponds to a cluster. The smaller circles in the internal domain of the inclusive circle in Fig. 1b are nodes whose second clusters consist of '23'; that is, the full permutation of node 145 is 14523, that of 541 is 54123, and so on. The permutation generated by the edge generator T_2 is 41523 in node 14523; that is, $T_2(14523) = 41523$ and $T_3(14523) = 54123$. When the edge generators T_j and S_i are sequentially applied to the permutation of a certain node U, they are represented as $S_i(T_j(U))$, and simply $S_iT_j(U)$. For example, the sequence of generators $S_2(T_2(14523))$ will create the permutation 42315 in 14523. First, T_2 generates 41523 and then S_2 provides 42315 [5].

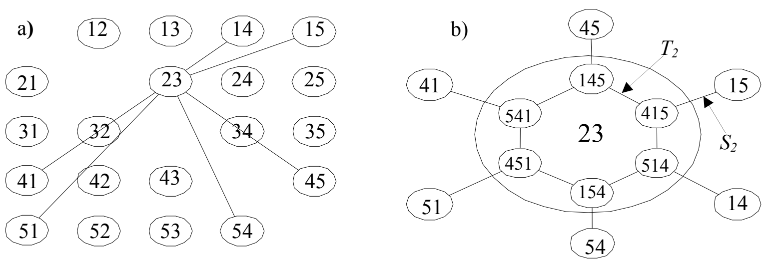

Fig. 1. Example of a macro-star graph $MS(2, 2)$

An n-dimensional bubble-sort graph B_n [12] consists of $n!$ nodes and $n(n-1)!/2$ edges. The address of each node is represented as a permutation of n symbols {1, 2, 3, ..., n}. An edge exists between two arbitrary nodes v and w if and only if the corresponding permutation of the node w can be obtained from that of v by interchanging two adjacent symbols v and w in the permutation. The bubble-sort graph Bn can thus be defined as shown in Eq. (2), where n distinct symbol sets $<N> = \{1, 2, .., n\}$, and a permutation of $<N>$, $B = b_1b_2...b_n$, $b_i \in <N>$. As the number of i- dimensional edges adjacent to B is equal to $n-1$, the bubble-sort graph B_n is a regular graph of degree $n-1$ and has a diameter of $n(n-1)/2$. It is also a hierarchical interconnection network because it can partition the graph with the edge as the center. It is node- and edge-symmetric as well as bipartite and includes Hamiltonian cycles.

$$V(B_n) = \{(b_1b_2...b_n) \mid b_i \in <N>, i \neq j, b_i \neq b_j\}$$
$$E(B_n) = \{(b_1b_2...b_ib_{i+1}...b_n)(b_1b_2...b_{i+1}b_i...b_n) \mid (b_1b_2...b_i...b_n) \in V(B_n), 1 \leq i \leq n-1\}. \quad (2)$$

An n-dimensional transposition graph T_n [13] consists of $n!$ nodes and $n(n-1)n!/4$ edges. The address of each node is represented as a permutation of n distinct symbols, and an edge exists between two nodes v and w if and only if the corresponding permutation of the node w can be obtained from that of v by interchanging the positions of any two arbitrary symbols from {1, 2, .., n} in v. A transposition graph T_n can be defined by Eq. (3) with n distinct symbols $<N> = \{1, 2, .., n\}$, and a permutation of $<N>$, $P = p_1p_2...p_n$, $p_i \in <N>$. The transposition graph T_n is a regular node symmetric graph with $n(n-1)/2$ degree, because an edge exists between the permutation that consists of n symbols and that in which two arbitrary different symbols are interchanged. It has maximum fault tolerance with a diameter of $n-1$ and a fault diameter of n. It also includes Hamiltonian cycles.

$$V(T_n)=\{(p_1p_2...p_i...p_n) \mid p_i \in <N>, i \neq j, p_i \neq p_j\}$$
$$E(T_n)=\{(p_1p_2...p_i...p_j...p_n)(p_1p_2...p_j...p_i...p_n) \mid (p_1p_2...p_i...p_j...p_n) \in V(T_n), 1 \leq i,j \leq n, i \neq j\}. \quad (3)$$

3 Embedding Analysis

The embedding of one graph G into another graph H is a mapping mechanism for examining whether graph G is included in the structure of graph H, and how they are interrelated. This can be interpreted as simulating one interconnection topology using another. The embedding of graph G into a graph H is defined as a function $f = (\phi, \rho)$ where ϕ maps the set of vertices in G, $V(G)$ one-to-one into the set of vertices in H, $V(H)$, and ρ corresponds to each edge (v, w) in G to a path in H that connects nodes $\phi(v)$ and $\phi(w)$. Parameters for evaluating the efficiency of an embedding method include dilation, congestion, and expansion. The dilation of edge e in G is the length of the path $\rho(e)$ in H, and the dilation of embedding f is the maximum value of all dilations in G. The congestion of edge e' in H is the number of $\rho(e)$ included in e', and the congestion of embedding f is the maximum number of all edge congestions in H. The expansion of embedding f is the ratio of the number of vertices in H to the number in G [6].

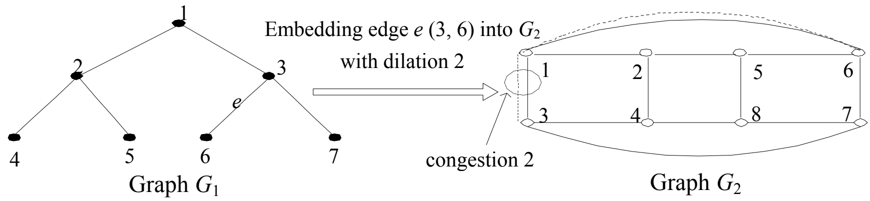

Fig. 2. Mapping example of G_1 into G_2

For instance, let each node (vertex) in the set $V(G_1)$ be mapped onto the corresponding same node number in the set $V(G_2)$ in Fig. 4. Then, edge e (3, 6) in G_1 can be mapped onto edges (3, 1) and (1, 6) or edges (3, 7) and (7, 6) in G_2 (i.e., to the path from 3 to 6 in G_2). Let us assume that edge e (3, 6) in G_1 is mapped onto edges (3, 1) and (1, 6) in G_2. Here, the dilation of this embedding is 2 because the length of the path $\rho(e)$ in G_2 is 2. As well, we can see that the congestion is 2 because edge e (1, 3) in G_2 is routed by two edges (1, 3) and (3, 6) in G_1, and the expansion is 8/7 [6].

When we assume that $R(=r_1r_2...r_i...r_j...r_n)$ is a node in an arbitrary graph and V is adjacent from R via dimensional edge J in the graph, it is represented as $V = J(R)$. And when we assume that node V is reached from node R by applying dimensional edges J and K in sequence, we denote the edge sequence as $<J, K>$ and $V = K(J(R))$. Sequentially applying edge sequence $<J, K>$ to node R means that at the first time unit, the permutation of node $J(R)$ is generated from node R via dimensional edge J, and at the second time unit, the permutation of $K(J(R))$ is created from node $J(R)$ via dimensional edge K (i.e., $V = K(J(R))$). The basic principles of embedding applied in this study are as follows. Node mapping to bubble-sort, transposition, and macro-star graphs is based on one-to-one mapping with identical node numbers. When mapping two adjacent nodes (U, V) of a source graph to a target graph, the dimensional edge sequence is defined using the edge definition of the target graph. This sequence is formulated with dimensional edges of the target graph used for the shortest path from $\phi(U)$ to $\phi(V)$. The dilation of embedding is represented as the number of dimensional edges required for the shortest path.

Theorem 1. A bubble-sort graph B_n can be embedded into a transposition graph T_n with dilation 1 and expansion 1.

Proof. In the bubble-sort graph B_n, if the permutation of a node B is $b_1b_2b_3...b_{i-1}b_ib_{i+1}...b_n$, then the permutation of node B' adjacent from node B is $b_1b_2...b_{i+1}b_ib_{i-1}...b_n$. The edge that connects nodes B and B' is called the i-dimensional edge ($2 \le i \le n$). There exist (n-1) edges of dimension i in B_n. When mapping nodes B and $B'(=b_1b_2...b_{i+1}b_ib_{i-1}...b_n)$ in B_n onto nodes $T(=t_1t_2...t_it_{i+1}...t_n)$ and $T'(=t_1t_2...t_{i+1}t_i...t_n)$, respectively, in T_n, we analyze the dilation of this mapping by referring to the length of the shortest path routing from node T to node T' in T_n. In the transposition graph T_n, there exists an edge between nodes V and W if the corresponding permutation to the node W can be obtained from that of V by interchanging the positions of any two arbitrary symbols from {1, 2, .., n} in V. Here, we can see that the nodes $T(t_1t_2...t_it_{i+1}...t_n)$ and $T'(t_1t_2...t_{i+1}t_i...t_n)$ in T_n are adjacent to each other by the edge definition of transposition graph T_n. Therefore, a bubble-sort graph B_n can be embedded into a transposition graph T_n with dilation 1 and expansion 1.

Corollary 2. A bubble-sort graph B_n is a sub-graph of a transposition graph T_n.

Theorem 3. The dilation cost of embedding a transposition graph T_n into a bubble-sort graph B_n is $O(n)$.

Proof. The transposition graph T_n and bubble-sort graph B_n are both node-symmetric. When mapping the transposition graph T_n and the bubble-sort graph B_n, node $T(=t_1t_2t_3...t_i...t_j...t_n)$ in T_n maps onto node $B(=b_1b_2b_3...b_i...b_j...b_n)$ in B_n, and each node T' of $n(n-1)/2$ nodes, which are adjacent to node T via edge $T(i, j)$, maps onto B', whose address is the same as that of each corresponding node $T'(i < j)$. We prove Theorem 3 using the mapping case of two nodes $T(=t_1t_2t_3...t_i...t_j...t_n)$ and $T'(=t_nt_2t_3...t_i...t_j...t_1)$, adjacent to each other via edge $T(1, n)$, onto a bubble-sort graph. The node T', which is adjacent to node T via edge $T(1, n)$, is the permutation $t_nt_2t_3...t_i...t_j...t_1$ in which the first symbol and the n^{th} symbol are interchanged with each other from the permutation of node T. Let us map nodes $T(=t_1t_2t_3...t_i...t_j...t_n)$ and $T'(=t_nt_2t_3...t_i...t_j...t_1)$ in T_n onto nodes $B(=b_1b_2b_3...b_i...b_j...b_n)$ and $B'(=b_nb_2b_3...b_i...b_j...b_1)$ in B_n. Here, nodes T and T' are adjacent to each other, but nodes B and B' in B_n are not. Thus, we analyze dilation based on the length of the shortest path routing from B and B' in B_n. In the bubble-sort graph B_n, the dimensional edge sequence required for the shortest path routing from node B to node B' is $<1,2,3,...,n\text{-}1,n\text{-}2,n\text{-}3,...,3,2,1>$. First, the first symbol b_1 of node B can be moved to the last position (n^{th} position) using the dimensional edge sequence $<1,2,3,...,n\text{-}1>$, since only an edge exists between two nodes in which two adjacent symbols are interchanged with each other. That is, by sequentially applying the dimensional edge sequence $<1,2,3,...,n\text{-}1>$ to node $B(=b_1b_2b_3...b_i...b_j...b_n)$, we can obtain the permutation $b_2b_3...b_i...b_j...b_nb_1$. Next, the permutation $b_nb_2b_3...b_i...b_j...b_{n\text{-}1}b_1$ (i.e., the permutation of destination node B') is obtained by orderly applying the edge sequence $<n\text{-}2,n\text{-}3,...,3,2,1>$ to the node of the permutation $b_2b_3...b_i...b_j...b_nb_1$. Here, the number of dimensional edges applied for routing from B to B' in B_n is equal to $2n\text{-}3$. Therefore, we can say that the dilation cost for this embedding process is $O(n)$.

Theorem 4. A macro-star graph $MS(2, n)$ can be embedded into a transposition graph T_{2n+1} with dilation n.

Proof. We prove Theorem 4 by dividing it into two edges, T_i and S_j, which connect two arbitrary nodes in the macro-star graph $MS(2, n)$.

Case 1. Edge T_i, $2 \le i \le n+1$

In the macro-star graph $MS(2, n)$, the node adjacent to node $U(=u_1u_2u_3...u_i...u_{n+i}...u_{2n+1})$ via edge T_i is $T_i(U)(=u_iu_2u_3...u_1......u_{n+i}...u_{2n+1})$ and it is denoted as U'. When we map nodes $U(=u_1u_2u_3...u_i......u_{n+i}...u_{2n+1})$ and $U'(=u_iu_2u_3...u_1......u_{n+i}...u_{2n+1})$ in $MS(2, n)$ onto nodes $T(=t_1t_2t_3...t_i...t_{n+i}...t_{2n+1})$ and $T'(=t_it_2t_3...t_1...t_{n+i}...t_{2n+1})$ in T_{2n+1}, we can see that the nodes T and T' in T_{2n+1} are adjacent to each other through edge $T(1, i)$ according to the edge definition of the transposition graph. Hence, a macro-star graph $MS(2, n)$ can be embedded into a transposition graph T_{2n+1} with dilation 1.

Case 2. Edge S_j, $j = 2$

Edge S_j, which is incident on node $U(=u_1u_2u_3...u_i......u_{n+i}...u_{2n+1})$, can exist only where $j = 2$, because the macro-star graph $MS(2, n)$ consists of two modules. In $MS(2, n)$, the node adjacent from $U(u_1u_2u_3...u_i......u_{n+i}...u_{2n+1})$ via edge S_2 is $S_2(U)(=u_1u_{n+1}u_{n+2}u_{n+3}...u_{n+i}...u_{2n+1}u_2u_3u_4...u_i...u_{n+1})$, and it is denoted as U'. When mapping nodes U and $U'(=u_1u_{n+1}u_{n+2}u_{n+3}...u_{n+i}...u_{2n+1}u_2u_3u_4...u_i...u_{n+1})$ in $MS(2, n)$ onto nodes $T(=t_1t_2t_3...t_i...t_{n+i}...t_{2n+1})$ and $T'(=t_1t_{n+1}t_{n+2}t_{n+3}...t_{n+i}...t_{2n+1}t_2t_3t_4...t_i...t_{n+1})$ in T_{2n+1}, it can be seen that the nodes T and T' in T_{2n+1} are not adjacent to each other by the edge definition of the transposition graph. Thus, we analyze the dilation of this mapping using the number of edges used for the shortest path routing from node T to node T' in T_{2n+1}. The dimensional edge sequence required for routing from node $T(t_1t_2t_3...t_i...t_{n+i}...t_{2n+1})$ to node $T'(t_it_2t_3...t_1...t_{n+i}...t_{2n+1})$ in T_{2n+1} is <T(2, n+2), T(3, n+3), T(4, n+4), ..., T(i, $n+i$), ..., T(n+1, 2n+1)>. In other words, edge $T(i, n+i)$, which interchanges the symbols t_i and t_{n+i}, which are in the same position of the first and second modules in the permutation of node $T(t_1t_2t_3...t_i...t_{n+i}...t_{2n+1})$, is used n times, so the permutation identical to node $T'(t_it_2t_3...t_1...t_{n+i}...t_{2n+1})$ is generated. Therefore, its dilation is n. When we map a macro-star graph $MS(2, n)$ onto a transposition graph T_{2n+1}, the worst dilation of this embedding is n, but most edges are mapped by dilation 1. Accordingly, it may be advisable to prove that the average dilation reaches to the smallest constant.

Theorem 5. A transposition graph T_{2n+1} can be embedded into a macro-star graph $MS(2, n)$ with dilation *5*.

Proof. In this embedding, we map node $T(t_1t_2t_3...t_i...t_j...t_{2n+1})$ in the transposition graph T_{2n+1} onto node $U(u_1u_2u_3...u_i...u_j...u_{2n+1})$ in the macro-star graph $MS(2, n)$, and node T' onto node U', which has the same permutation with T' among the nodes in $MS(2,n)$. The permutation of the node adjacent to node $T(t_1t_2t_3...t_i...t_j...t_{2n+1})$ via edge T(i, j) in T_{2n+1} is $T'=t_1t_2t_3...t_j...t_i...t_{2n+1}$. Here, nodes $U(u_1u_2u_3...u_i...u_{n+i}...u_{2n+1})$ and $U'(u_1u_2u_3...u_j...u_i...u_{2n+1})$ in graph $MS(2, n)$ are not adjacent to each other, thus we analyze dilation using the number of edges used for the shortest path routing from node U to node U' in $MS(2, n)$. We prove Theorem 6 by dividing it into three cases depending on the values of i and j in T(i, j).

Case 1. $i, j \le n-1,\ i < j$

In the edge $T(i, j)$, which connects nodes $T(t_1t_2t_3...t_i...t_j...t_{2n+1})$ and T' in the transposition graph T_{2n+1}, values of i and j smaller than (n+2) mean that the two symbols can be interchanged only from the first symbol to $(n+1)^{th}$ symbols. The occurrence of the interchange only from the first symbol to the $(n+1)^{th}$ symbol in node $U(u_1u_2u_3...\ u_i...\ u_j...\ u_{2n+1})$ of $MS(2,n)$, in which node $T(t_1t_2t_3...\ t_i...\ t_j...\ t_{2n+1})$ of T_{2n+1} is mapped, means that a symbol interchange occurs among the symbols that consist of the first cluster of node U. Therefore, the edge sequence required for the shortest path routing from node U to node U' is <T_i, T_j, T_i>, because the permutation of node U' is $u_1u_2u_3...u_j...u_i...u_{n+2}u_{n+3}...u_{2n+1}$, and nodes U and U' are not adjacent to each other. The routing process from U to U' using this edge sequence <T_i, T_j, T_i> is as follows.

First, node $T_i(U)(=u_iu_2u_3...u_1...u_j...u_{n+2}u_{n+3}...u_{2n+1})$ is reached from node $U(u_1u_2u_3...u_i...u_j...u_{n+2}u_{n+3}...u_{2n+1})$ via edge T_i, which interchanges the first symbol with the i^{th} symbol u_i in node U; that is, the node $T_i(U)$ is adjacent to node U through edge T_i. We then get to node $T_jT_i(U)(=u_ju_2u_3...u_1...u_i...u_{n+2}u_{n+3}...u_{2n+1})$ from node $T_i(U)$

via edge T_j, which interchanges the i^{th} and j^{th} symbols in node $T_i(U)$. Next, we reach node $T_iT_jT_i(U)(=u_1u_2u_3...u_j...u_i...u_{n+2}u_{n+3}...u_{2n+1})$ from $T_jT_i(U)$ via edge T_i, which interchanges symbols u_j and u_1 in node $T_jT_i(U)$. Now, we can see that the permutation of node $T_iT_jT_i(U)$, which is obtained by sequentially applying the edge sequence $<T_i, T_j, T_i>$ to node U, is the same as the permutation of node U'. Therefore, two nodes $T(t_1t_2t_3...t_i...t_j...t_{2n+1})$ and T' adjacent via edge $T(i, j)$ in the transposition graph T_{2n+1} can be embedded into a macro-star graph $MS(2, n)$ with dilation 3.

Case 2. $i \leq n+1, j \geq n+2$

In the edge $T(i, j)$ which connects nodes $T(t_1t_2t_3...t_i...t_j...t_{2n+1})$ and T' in T_{2n+1}, $i \leq n+1$ and $j \geq n+2$ mean that two symbols are interchanged with each other, and of these two, the one based on the $(n+1)^{th}$ position is positioned before it, and the other locates next to it. The occurrence of the interchange based on the $(n+1)^{th}$ position of a symbol in node $U(u_1u_2u_3...u_i...u_j...u_{2n+1})$ of $MS(2,n)$, in which node $T(t_1t_2t_3...t_i...t_j...t_{2n+1})$ of T_{2n+1} is mapped, means that an interchange occurs between one symbol of the first cluster and one symbol of the second cluster in node U. Since the permutation of node U' is $u_1u_2u_3...u_i...u_{n+2}u_{n+3}...u_j...u_{2n+1}$, and nodes U and U' are not adjacent to each other, the edge sequence required for the shortest path routing from node U to node U' is $<T_i, S_2, T_j, S_2, T_i>$. The routing process from $U(u_1u_2u_3...u_i...u_{n+1}u_{n+2}u_{n+3}...u_j...u_{2n+1})$ to $U'(u_1u_2u_3...u_i...u_{n+1}u_{n+2}u_{n+3}...u_j...u_{2n+1})$ using this edge sequence $<T_i, S_2, T_j, S_2, T_i>$ is as follows. First, node $T_i(U)(=u_iu_2u_3...u_1...u_{n+1}u_{n+2}u_{n+3}...u_j...u_{2n+1})$ is adjacent to node $U(u_1u_2u_3...u_i...u_{n+1}u_{n+2}u_{n+3}...u_j...u_{2n+1})$ via edge T_i, which interchanges the first with the i^{th} symbol u_i in node U; then node $S_2T_i(U)(=u_iu_{n+2}u_{n+3}...u_j...u_{2n+1}u_2u_3...u_1...u_{n+1})$ is adjacent to node $T_i(U)$ through edge S_2, which swaps the first and the j^{th} cluster u_j in $T_i(U)$. Next, node $T_jS_2T_i(U)(=u_ju_{n+2}u_{n+3}...u_i...u_{2n+1}u_2u_3...u_1...u_{n+1})$ is adjacent to node $S_2T_i(U)$ through edge T_j, which interchanges the first symbol u_i with the symbol u_j in the first cluster in $S_2T_i(U)$. After that, node $S_2T_jS_2T_i(U)$ $(=u_ju_2u_3...u_1...u_{n+1}u_{n+2}u_{n+3}...u_i...u_{2n+1})$ is reached from $T_jS_2T_i(U)$ via edge S_2, which exchanges the first cluster in which the symbol u_i exists with the second cluster in node $T_jS_2T_i(U)$. Then, we get to node $T_iS_2T_jS_2T_i(U)$ $(=u_1u_2u_3...u_j...u_{n+1}u_{n+2}u_{n+3}...u_i...u_{2n+1})$ from node $S_2T_jS_2T_i(U)$ via edge T_i, which interchanges the symbols u_j and u_1 in node $S_2T_jS_2T_i(U)$. Because the permutation of node $T_iS_2T_jS_2T_i(U)$, which is obtained by sequentially applying the edge sequence $<T_i, S_2, T_j, S_2, T_i>$ to node U, is identical to the permutation of node U', we can see that two nodes T and T' adjacent via edge $T(i, j)$ in the transposition graph T_{2n+1} can be embedded into a macro-star graph $MS(2, n)$ with dilation 5.

Case 3. $i, \ j \geq n+2$

In edge $T(i, j)$, which connects nodes $T(t_1t_2t_3...t_i...t_j...t_{2n+1})$ and T' of the transposition graph T_{2n+1}, because the values i and j are larger than $(n+1)$, the two symbols can be interchanged with each other only from the $(n+2)^{th}$ symbol to the $(2n+1)^{th}$ symbols. The interchanges from the $(n+2)^{th}$ symbol to the $(2n+1)^{th}$ symbols in node $U(u_1u_2u_3... u_i... u_j... u_{2n+1})$ of $MS(2,n)$, in which node $T(t_1t_2t_3... t_i... t_j... t_{2n+1})$ of T_{2n+1} is mapped, represent the interchanges that occur among the symbols that consist of the second cluster of U. Because the permutation of node U' is $u_1u_2u_3...u_{n+1}u_{n+2}...u_j...u_i...u_{2n+1}$, and nodes $U(u_1u_2u_3...u_{n+1}u_{n+2}...u_i...u_j...u_{2n+1})$ and U' are not adjacent to each other, the edge sequence required for the shortest path routing from node U to node U' is $<S_2, T_i, T_j, T_i, S_2>$. The routing process from node U and node U' is as follows.

We first use edge generator S_2 to exchange the first and second clusters in which symbols u_i and u_j exist, because the interchange between the symbols in node U occurs only between the symbols positioned in the first column and the symbols positioned in the first cluster. Thus, node $S_2(U)(=u_1u_{n+2}...u_i...u_j...u_{2n+1}u_2u_3...u_{n+1})$ is adjacent to node $U(u_1u_2u_3...u_{n+1}u_{n+2}...u_i...u_j...u_{2n+1})$ via edge S_2, and node $T_iS_2(U)(=u_iu_{n+2}...u_1...u_j...u_{2n+1}u_2u_3...u_{n+1})$ is adjacent to node $S_2(U)$ via edge T_i, which places symbol u_i of $S_2(U)$ at the first position. Following this, node $T_{j\text{-}}T_iS_2(U)(=u_ju_{n+2}...u_1...u_i...u_{2n+1}u_2u_3...u_{n+1})$ is reached from node $T_iS_2(U)$ via edge T_j, which interchanges the first symbol u_i with symbol u_j in the first cluster in $S_2T_i(U)$. Next, node $T_iT_jT_iS_2(U)(=u_1u_{n+2}...u_j...u_i...u_{2n+1}u_2u_3...u_{n+1})$ is connected to $T_jT_iS_2(U)$ by edge T_i, which interchanges the first symbol u_j with u_1 in node $T_jT_iS_2(U)$. Then, we use edge S_2 to swap the first and second clusters in node $T_iT_jT_iS_2(U)$, and reach node $S_2T_iT_jT_iS_2(U)(=u_1u_2u_3...u_{n+1}u_{n+2}...u_j...u_i...u_{2n+1})$ from node $T_iT_jT_iS_2(U)$ through edge S_2. Here, we can seen that the permutation of node $S_2T_iT_jT_iS_2(U)$, which is obtained by sequentially applying the edge sequence $<S_2, T_i, T_j, T_i, S_2>$ to node U, is identical to the permutation of node U'. Therefore, the edge $T(i, j)$ that connects nodes $T(t_1t_2t_3...t_i...t_j...t_{2n+1})$ and T' in the transposition graph T_{2n+1} can be embedded into a macro-star graph $MS(2, n)$ with dilation 5.

Consequently, all nodes in a transposition graph T_{2n+1} can be mapped one-to-one onto a macro-star graph $MS(2, n)$, and an edge in T_{2n+1} can be embedded into $MS(2,n)$ with dilation of 5 or under.

4 Conclusion

The star graph, a well-known topology of MIMD multi-computer systems with distributed memory, is a small diameter, node-symmetric, hierarchical, and maximum fault-tolerant interconnection network. The transposition graph not only improves the fault tolerance of multi-computer systems, but also shares the advantages of a star graph. It also contains a star graph as its sub-graph. The bubble-sort and macro-star graphs also share the advantages of the star graph. In this paper, we proposed methods for embedding bubble-sort, transposition, and macro-star graphs into one another, which have been introduced as variations of the star graph. These graphs have the same number of nodes and also the same number of symbols in a node.

The proposed embedding methods are based on one-to-one mapping of two arbitrary nodes U and U' of a source graph G onto two nodes in a target graph G'. We assumed that two mapped nodes in G' are connected with a minimum of edges based on the edge definition of target graph G'. Then, we analyzed dilation by the number of edges used for the shortest path routing between two mapped nodes in G'. Embedding analysis using the edge definition of graphs is possible because the bubble-sort, transposition, and macro-star graphs are all node-symmetric. The results of this study indicate that bubble-sort graph B_n can be embedded into transposition T_n with dilation 1 and expansion 1. In addition, macro-star graph $MS(2, n)$ can be embedded into transposition graph T_{2n+1} with dilation n, but with an average dilation of 2 or less.

Acknowledgement. This research was supported by Basic Science research program through the National research Foundation of KOREA(NRF) funded by the Ministry of Education, Science and Technology(2009-0086676).

References

1. Feng Athas, W.C., Seitz, C.L.: Multicomputers: Message-Passing Concurrent Computers. IEEE Computer, 9–24 (August 1988)
2. Saad, Y., Schultz, M.H.: Topological Properties of Hypercubes. IEEE Trans. Comput. 37(7), 867–872 (1988)
3. Azevedo, M.M., Bagherzaeh, N., Latifi, S.: Low Expansion Packing and Embeddings of Hypercubes into Star Graphs: A Performance-Oriented Approach. IEEE Parallel and Distributed Systems 9(3), 261–274 (1998)
4. Ghafoor, A., Bashkow, T.R.: A Study of Odd Graphs as Fault-Tolerant Interconnection Networks. IEEE Trans. Computers 40(2), 225–232 (1991)
5. Kim, J.S., Seo, J.H., Sim, H., Lee, H.O.: Embedding Algorithm Between the Macro-star Graph and the Matrix-star Graph. In: Proceedings of the International Conference on Parallel and Distributed Processing Techniques and Application. PDPTA, pp. 418–426 (2006)
6. Kim, M.H., Kim, D.W., Lee, H.O.: Embedding Algorithms for Star, Bubble-Sort, Rotator-Faber-Moore, and Pancake Graphs. In: Hsu, C.-H., Yang, L.T., Park, J.H., Yeo, S.-S. (eds.) ICA3PP 2010. LNCS, vol. 6082, pp. 348–357. Springer, Heidelberg (2010)
7. Seo, J.H., Lee, H.O., Jang, M.S.: Node Mapping Algorithm Between Macro-star Networks and Transposition Networks. In: Proceedings of the Korean Information Science Society Conference 2008, vol. 35(1B), pp. 584–587 (2008)
8. Akers, S.B., Krishnamurthy, B.: A Group-Theoretic Model for Symmertric Interconnection Network. IEEE Trans. Comput. 38(4), 555–565 (1989)
9. Yang, X., Tang, Y.Y., Cao, J.: Embedding Torus in Hexagonal Honeycomb Torus. IET Computers & Digital Techniques 2, 86–93 (2008)
10. Park, J.H., Lim, H.S., Kim, H.C.: Embedding Starlike Trees into Hypercube-Like Interconnection Networks. In: Min, G., Di Martino, B., Yang, L.T., Guo, M., Rünger, G. (eds.) ISPA Workshops 2006. LNCS, vol. 4331, pp. 301–310. Springer, Heidelberg (2006)
11. Berthome, P., Ferreira, A., Perennes, S.: Optimal Information Dissemination in Star and Pancake Networks. IEEE Trans. on Parallel and Distributed Syst. 7(12), 1292–1300 (1996)
12. Chou, Z.T., Hsu, C.C., Sheu, J.P.: Bubblesort Star graphs: A New Interconnection Network. In: 9th International Parallel Processing Symposium, pp. 41–48 (1996)
13. Latifi, S., Srimani, P.K.: Transposition Networks as a Class of Fault-Tolerant Robust Networks. IEEE. Trans. Comput. 45(2), 230–238 (1996)
14. Yeh, C.H., Varvarigos, E.A.: Macro-Star Networks: Efficient Low-Degree Alternatives to Star Graphs. IEEE Trans. Parallel and Distributed Systems 9(10), 987–1003 (1998)
15. Corbett, P.F.: Rotator Graphs: An Efficient Topology for Point-to-Point Multiprocessor Networks. IEEE Trans. Parallel Distributed Syst. 3(5), 622–626 (1992)
16. Faber, V., Moore, J. W.: High-degree, Low-diameter Interconnection Networks with Vertex Symmetry: The Directed Case. LA-UR-88-1051, LosAlmos National Lab., Los Almos (1988)