Embedding of Hypercubes into Banana trees

*,[1] Jasintha Quadras and [1] S. Sarah Surya
[1] Stella Maris College, Chennai 600 086, India
sara24solomon@gmail.com

Abstract

Graph embedding is an important technique used in the study of computational capabilities of processor interconnection networks and task distribution. In this paper, we present an algorithm for embedding the Hypercubes into Banana Trees and Extended Banana Trees and prove its correctness using the Congestion lemma and Partition lemma.

1 Introduction

In the field of interconnection networks, the study of graph embeddings is motivated by the problem of efficient simulation of interconnection networks and parallel algorithms on a different interconnection network. Several types of networks have been considered like the hypercubes, shuffle exchange networks, butterfly networks, trees and complete binary trees and a number of papers have been published in the last ten years on embedding a given network into another. Thus, graph embedding is an important technique used in the study of computational capabilities of processor interconnection networks and task distribution. They also have applications in different areas of computer science. For example, any finite graph can be considered as a model of a parallel computer, where the vertices correspond to processors and the edges represent the communication links between them. A good embedding is said to exist when the adjacent processors in the guest network are mapped to reasonably close processors in the host network and the paths between adjacent processors in the guest network are chosen in such a way that the congestion across each host edge is moderately small.

*This work is supported by UGC, Project No.F. No. 39-47 / 2010.

The quality of an embedding can be measured by certain cost criteria, namely dilation, expansion, congestion and wirelength. The dilation of an embedding is the maximum distance between the images of adjacent nodes. It is the measure for the communication time needed when simulating one network on another. The bandwidth is the dilation if the host graph is a path. The expansion of an embedding f is the ratio of the number of vertices of H to the number of vertices of G. The congestion of an embedding f of G into H is the maximum number of edges of the graph G that are embedded on any single edge of H. The congestion sum or the wirelength of a graph embedding arises from the VLSI designs, data structures, data representations, networks for parallel computer systems, biological models that deal with cloning and visual stimuli, parallel architecture, structural engineering and so on.

There are several results on the congestion problem of various architectures such as trees on cycles [2], trees on stars [3], hypercubes into grids [4], complete binary tree into grids [5], grids into grids [6], ladders and caterpillars into hypercubes [7], binary trees into hypercubes [8], complete binary trees into hypercubes [9], incomplete hypercube in books [10], m-sequencial k-ary trees into hypercubes [11], ternary tree into hypercube [12], enhanced and augmented hypercube into complete binary tree [13], circulant into arbitrary trees, cycles, certain multicyclic graphs and ladders [14], and hypercubes into cylinders, snakes and caterpillars [15].

In this paper, we present an algorithm for embedding the hypercubes into Banana Trees and Extended Banana Trees and prove its correctness using the Congestion lemma and Partition lemma.

2 Preliminaries

Definition 2.1. [4] Let $G(V, E)$ and $H(V, E)$ be finite graphs with n vertices. An *embedding* f of G into H is defined as follows:

1. f is a bijective map from $V(G) \to V(H)$
2. f is a one-to-one map from $E(G)$ to $\{P_f(f(u), f(v)) : P_f(f(u), f(v))$ is a path in H between $f(u)$ and $f(v)$ for $(u, v) \in E(G)\}$.

The graph G that is being embedded is called a *virtual graph* or a *guest graph* and H is called a *host graph.* Some authors use the name *labelling* instead of embedding.

The *edge congestion* of an embedding f of G into H is the maximum number of edges of the graph G that are embedded on any single edge of

H. Let $EC_f(G, H(e))$ denote the number of edges (u, v) of G such that e is in the path $P_f(f(u), f(v))$ between $f(u)$ and $f(v)$ in H. In other words,

$$EC_f(G, H(e)) = |\{(u,v) \in E(G) : e \in P_f(f(u), f(v))\}|$$

where $P_f(f(u), f(v))$ denotes the path between $f(u)$ and $f(v)$ in H with respect to f.

The edge congestion problem of a graph G is to find an embedding of G into H that induces $EC(G, H)$.

Definition 2.2. [16] The *wirelength* of an embedding f of G into H is given by

$$WL_f(G, H) = \sum_{(u,v)\in E(G)} d_H(f(u), f(v)) = \sum_{e\in E(H)} EC_f(G, H(e))$$

where $d_H(f(u), f(v))$ denotes the length of the path $P_f(f(u), f(v))$ in H. Then, the *wirelength* of G into H is defined as

$$WL(G, H) = \min WL_f(G, H)$$

where the minimum is taken over all embeddings f of G into H.

The *edge isoperimetric problem* [18] is used to solve the wirelength problem when the host graph is a path and is NP-complete [17].The following two versions of the edge isoperimetric problem of a graph $G(V, E)$ have been considered in the literature [18].

Problem 1 : Find a subset of vertices of a given graph such that the edge cut separating this subset from its complement has minimal size among all subsets of the same cardinality. Mathematically, for a given m, if $\theta_G(m) = \min_{A\subseteq V,\ |A|=m} |\theta_G(A)|$ where $\theta_G(A) = \{(u,v) \in E : u \in A, v \notin A\}$, then the problem is to find $A \subseteq V$ such that $|A| = m$ and $\theta_G(m) = |\theta_G(A)|$.

Problem 2 : Find a subset of vertices of a given graph such that the number of edges in the subgraph induced by this subset is maximal among all induced subgraphs with the same number of vertices. Mathematically, for a given m, if $I_G(m) = \max_{A\subseteq V,\ |A|=m} |I_G(A)|$ where $I_G(A) = \{(u,v) \in E : u, v \in A\}$, then the problem is to find $A \subseteq V$ such that $|A| = m$ and $I_G(m) = |I_G(A)|$.

For a given m, where $m = 1, 2, ..., n$, we consider the problem of finding a subset A of vertices of G such that $|A| = m$ and $|\theta_G(A)| = \theta_G(m)$. Such subsets are called optimal. We say that optimal subsets are nested if there exists a total order $\mathcal{O}$ on the set V such that for any $m = 1, 2, ..., n$, the collection of the first m vertices in this order is an optimal subset. In this case we call the order $\mathcal{O}$ an optimal order [18]. This implies that $WL(G, P_n) = \sum_{m=0}^{n} \theta_G(m)$.

Further, if a subset of vertices is optimal with respect to Problem 1, then its complement is also an optimal set. But, it is not true for Problem 2 in general. However for regular graphs a subset of vertices S is optimal with respect to Problem 1 if and only if S is optimal for Problem 2 [18]. In the literature, Problem 2 is defined as the *maximum subgraph problem.*

Lemma 2.3. (Congestion Lemma) [16] Let G be an r-regular graph and f be an embedding of G into H. Let S be an edge cut of H such that the removal of edges of S leaves H into 2 components H_1 and H_2 and let $G_1 = f^{-1}(H_1)$ *and* $G_2 = f^{-1}(H_2)$. *Also* S satisfies the following conditions:

(i) *For every edge* $(a, b) \in G_i$, $i = 1, 2, P_f(f(a), f(b))$ *has no edges in* S.

(ii) *For every edge* (a, b) *in* G *with* $a \in G_1$ *and* $b \in G_2$, $P_f(f(a), f(b))$ *has exactly one edge in* S.

(iii) G_1 *is an optimal set.*

Then $EC_f(S)$ *is minimum and* $EC_f(S) = r\,|V(G_1)| - 2\,|E(G_1)|$. □

Lemma 2.4. (Partition Lemma) [16] Let $f : G \to H$ be an embedding. Let $\{S_1, S_2, ..., S_p\}$ be a partition of $E(H)$ such that each S_i is an edge cut of H. Then

$$WL_f(G, H) = \sum_{i=1}^{p} EC_f(S_i). \quad \square$$

The hypercube is a very popular interconnection network for parallel computation because of its regularity and the relatively small number of interprocessor connections [1].

Definition 2.5. [19] For $r \geq 1$, let Q^r denote the graph of r-dimensional hypercube. The vertex set of Q^r is formed by the collection of all r-dimensional binary representations. Two vertices $x, y \in V(Q^r)$ are adjacent if and only if the corresponding binary representations differ exactly in one bit.

Definition 2.6. [16] An incomplete hypercube on i vertices of Q^r is the subcube induced by $\{0, 1, , i-1\}$ and is denoted by L_i, $1 \leq i \leq 2^r$.

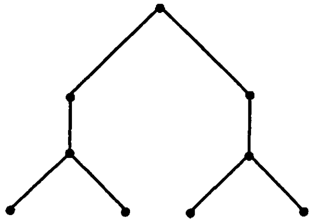

Figure 1: Banana Tree $B(2;4)$

Definition 2.7. [23] Let T be a tree having root r with sons $v_1, v_2, , v_k, k \geq 0$. In the case $k = 0$, the tree consists of a single vertex r. A postorder traversal of T is defined recursively as follows:

(i) Visit in postorder the subtrees with roots $v_1, v_2, , v_k$ in that order.

(ii) Visit the root r.

Theorem 2.8. [16] *Let Q^r be an r-dimensional hypercube. For $1 \leq i \leq 2^r$, L_i is an optimal set or a composite set.*

Lemma 2.9. *For $i = 1, 2, \cdots, r-1, NcutS_i^{2^i} = 2^i, 2^i + 1, \cdots, 2^{i+1}$ is an optimal set in Q^r.*

Proof. Define $\varphi : NcutS_i^{2^i} \rightarrow L_{2^i}$ by $\varphi(2^i + k) = k$. If the binary representation of $2^i + k$ is $\alpha_1, \alpha_2, \cdots, \alpha_r$, then the binary representation of k is $\underbrace{00\ldots00}_{r-i \text{ times}}\alpha_{r-i+1}\alpha_{r-i+2}\ldots\alpha_r$. Thus the binary representation of two numbers x and y differ exactly in one bit $\Leftrightarrow$ the binary representation of $\varphi(x)$ and $\varphi(y)$ differ in exactly one bit. Therefore, (x, y) is an edge in $NcutS_i^{2^i} \Leftrightarrow (\varphi(x), \varphi(y))$ is an edge in L_{2^i}. Hence $NcutS_i^{2^i}$ and L_{2^i} are isomorphic. By Theorem 2.8, $NcutS_i^{2^i}$ is an optimal set in Q^r. □

Lemma 2.10. [20] *For $i = 1, 2, \cdots, r-1, NcutS_i^{2^i} = 2^i, 2^i + 1, , 2^{i+1} - 1$ is an optimal set in Q^r.*

Lemma 2.11. [20] *For $i = 1, 2, \cdots, r-1, NcutS_i^{2^i} = 2^i, 2^i + 1, , 2^{i+1} - 2$ is an optimal set in Q^r.*

3 Wirelength of Hypercubes into Banana Trees

Definition 3.1. [21, 22] A $B(n, k)$ Banana Tree, is a graph obtained by

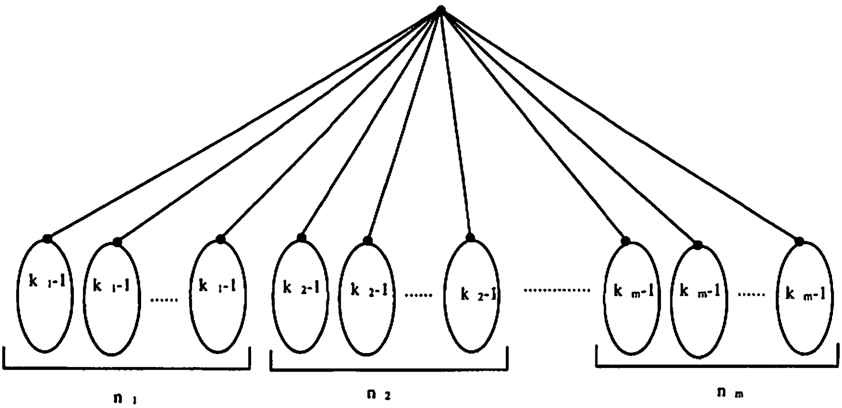

Figure 2: $B(n_1, n_2, \cdots, n_m; k_1, k_2, \cdots, k_m)$

connecting one leaf of each of n copies of an k-star graph with a single root vertex that is distinct from all the stars. See Figure 1.

Note : For our discussion, we impose the condition that $nk = 2^r$.

Embedding Algorithm A

Input : The r-dimensional hypercube $Q^r, r \geq 1$ and Banana Tree $B(n,k)$.

Algorithm : Label the vertices of Q^r by using lexicographic labeling [18] and label the vertices of $B(n,k)$ using the post order labeling.

Output : An embedding f of Q^r into $B(n,k)$ given by $f(x) = x$ with minimum wirelength.

Proof of correctness : For $1 \leq i \leq n$, let $S_i = (2^r - 1, ki), S_i^{'} = (ki, ki-1)$ and $S_i^j = (ki-1, ki-1-j)$, where $1 \leq j \leq k-2$. Thus $\{S_i : 1 \leq i \leq n\} \cup \{S_i^{'} : 1 \leq i \leq n\} \cup \{S_i^j : 1 \leq i \leq n, 1 \leq j \leq k-2\}$ is a partition of $E(B(n,k))$. For each $i, 1 \leq i \leq n, E(B(n,k)) \backslash S_i$ has two components H_{i1} and H_{i2}, where $V(H_{i1}) = \{ki-1, ki-2, ..., k(i-2)\}$. Let $G_{i1} = f^{-1}(H_{i1})$ and $G_{i2} = f^{-1}(H_{i2})$. By lemma 2.9, G_{i1} is an optimal set and each S_i satisfies conditions (i), (ii) and (iii) of the Congestion Lemma. Therefore, $EC_f(S_i)$ is minimum.

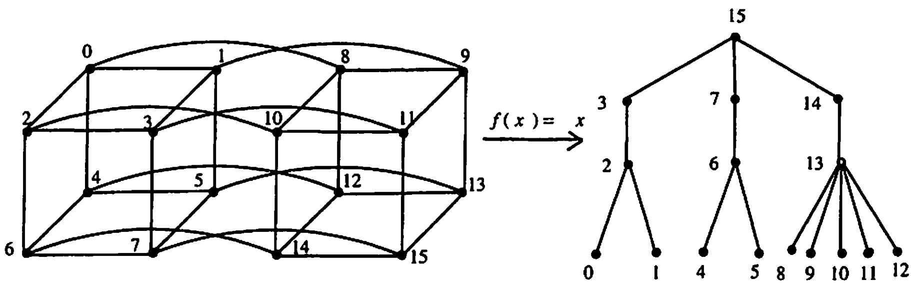

Figure 3: Embedding of Q^r into $B(2,1;4,8)$

For each $i, 1 \leq i \leq n$, we see that, $E(B(n,k))\backslash S_i^{1}$ has two components $H_{i1}^{'}$ and $H_{i2}^{'}$, where, $V(H_{i1}^{'}) = \{ki-2, ki-3, ..., k(i-1)\}$. Let $G_{i1}^{'} = f^{-1}(H_{i1}^{'})$ and $G_{i2}^{'} = f^{-1}(H_{i2}^{'})$. By lemma 2.10, $G_{i1}^{'}$ is an optimal set and each S_i^{1} satisfies conditions (i), (ii) and (iii) of the Congestion Lemma. Therefore, $EC_f(S_i^{1})$ is minimum.

For each $1 \leq i \leq n, 1 \leq j \leq k-2$, we see that, $E(B(n,k)\backslash S_i^j$ has two components H_{i1}^j and H_{i2}^j, where, $V(H_{i1}^j) = \{ki-(k-1), ki-(k-2), ..., ki-(k-j-1)\}$. Let $G_{i1}^j = f^{-1}(H_{i1}^j)$ and $G_{i2}^j = f^{-1}(H_{i2}^j)$. By lemma 2.11, G_{i1}^j is an optimal set and each S_i^j satisfies conditions (i), (ii) and (iii) of the Congestion Lemma. Therefore, $EC_f(S_i^j)$ is minimum.

The proof of the following theorem is an easy consequence of Embedding Algorithm A.

Theorem 3.2. *The exact wirelength of Q^r into $B(n,k)$,is given by*

$$WL(Q^r, B(n,k)) = 3rn(k-1) - 2 \mid E(Q^r(L_{k-1})) \mid -2 \mid E(Q^r(L_k)) \mid$$

4 Wirelength of the Hypercube into the Extended Banana Tree

Definition 4.1. An Extended Banana Tree denoted by $B(n_1, n_2, \cdots, n_m; k_1, k_2, \cdots, k_m)$ is a graph obtained by connecting one leaf of each of $\{n_1, n_2, \cdots, n_m\}$ copies of $\{k_1, k_2, \cdots, k_m\}$ - star graphs with a single root vertex that is distinct from all the stars. See Figure 2.

Note : In this paper we consider the Extended Banana Tree satisfying the following conditions:

(i) For $1 \leq i \leq m, k_i$contains 2^{r_i} vertices.

(ii) The $nmth$ copy of k_m contains 2^{r_m} vertices.

(iii) $\sum_{i=1}^{m} n_i k_i = 2^r$

Remark : Let $t_i = \sum_{i=1}^{m} n_i k_i$ such that $t_0 = k_0 = 0$.

Embedding Algorithm B

Input : The r-dimensional hypercube $Q^r, r \geq 1$ and an Extended Banana Tree $B(n_1, n_2, \cdots, n_m; k_1, k_2, \cdots, k_m)$.

Algorithm : Label the vertices of Q^r by using lexicographic labeling [18] and label the vertices of $B(n_1, n_2, \cdots, n_m; k_1, k_2, \cdots, k_m)$ using the post order labeling.

Output : An embedding f of Q^r into $B(n_1, n_2, \cdots, n_m; k_1, k_2, \cdots, k_m)$ given by $f(x) = x$ with minimum wirelength. See Figure 3.

Proof of correctness : We assume that the labels represent the vertices to which they are assigned. For,

$i \neq m, S_l^i = \{(2^{r-1}, t_{i-1} + lk_i - 1)\}, 1 \leq l \leq n_i$

$i = m, S_l^i = \{(2^{r-1}, t_{i-1} + lk_i - 1)\}, 1 \leq l \leq n_i - 1$

$i \neq m, S_l^i = \{(2^{r-1}, t_{i-1} + lk_i - 2)\}, l = n_i$

$i \neq m, T_l^i = \{(t_{i-1} + lk_i - 1, t_{i-1} + lk_i - 2)\}, 1 \leq l \leq n_i$

$i = m, T_l^i = \{(t_{i-1} + lk_i - 1, t_{i-1} + lk_i - 2)\}, 1 \leq l \leq n_i - 1$

$i \neq m, T_l^i = \{(t_{i-1} + lk_i - 2, t_{i-1} + lk_i - 3)\}, l = n_i$

$i \neq m, U_l^i = \{(t_{i-1} + lk_i - 2, t_{i-1} + lk_i - 2 - j)\}, 1 \leq j \leq k_i - 2, 1 \leq l \leq n_i$

$i = m, U_l^i = \{(t_{i-1} + lk_i - 2, t_{i-1} + lk_i - 2 - j)\}, 1 \leq j \leq k_i - 2, 1 \leq l \leq n_i$

$i \neq m, U_l^i = \{(t_{i-1} + lk_i - 3, t_{i-1} + lk_i - 3 - j)\}, 1 \leq j \leq k_i - 2, 1 \leq l \leq n_i$

See Figure 3. Then $\{S_l^i, T_l^i, U_l^i\}$ is a partition of $E(B(n_1, n_2, \cdots, n_m; k_1, k_2, \cdots, k_m)$.

For $1 \leq l \leq n_i$, we see that, $E(B(n_1, n_2, \cdots, n_m; k_1, k_2, \cdots, k_m)$ S_l^i has two components H_{i1} and H_{i2}, where, for,

$i \neq m, V(S_l^i) = \{(t_{i-1} + (l-1)k_i + 1, t_{i-1} + (l-1)k_i + 2, \cdots, t_{i-1} + lk_i\}$

$i = m, V(S_l^i) = \{(t_{i-1} + (l-1)k_i + 1, t_{i-1} + (l-1)k_i + 2, \cdots, t_{i-1} + lk_i\}$

$i = m, V(S_l^i) = \{(t_{i-1} + (l-1)k_i + 1, t_{i-1} + (l-1)k_i + 2, \cdots, t_{i-1} + lk_i - 1\}$

Let $G_{i1} = f^{-1}(H_{i1})$ and $G_{i2} = f^{-1}(H_{i2})$. By lemma 2.9, G_{i1} is an optimal set and each S_l^i satisfies conditions (i), (ii) and (iii) of the Congestion Lemma. Therefore, $EC_f(S_l^i)$ is minimum.

For $1 \leq l \leq n_i$, we see that, $E(B(n_1, n_2, \cdots, n_m; k_1, k_2, \cdots, k_m)\ T_l^i$ has two components H'_{i1} and H'_{i2}, where, for,

$i \neq m, V(T_l^i) = \{(t_{i-1} + (l-1)k_i + 1, t_{i-1} + (l-1)k_i + 2, \cdots, t_{i-1} + lk_i - 1\}$

$i = m, V(T_l^i) = \{(t_{i-1} + (l-1)k_i + 1, t_{i-1} + (l-1)k_i + 2, \cdots, t_{i-1} + lk_i - 1\}$

$i = m, V(T_l^i) = \{(t_{i-1} + (l-1)k_i + 1, t_{i-1} + (l-1)k_i + 2, \cdots, t_{i-1} + lk_i - 2\}$

Let $G'_{i1} = f^{-1}(H'_{i1})$ and $G'_{i2} = f^{-1}(H'_{i2})$. By lemma 2.10, G'_{i1} is an optimal set and each T_l^i satisfies conditions (i), (ii) and (iii) of the Congestion Lemma. Therefore, $EC_f(T_l^i)$ is minimum.

For $1 \leq l \leq n_i$, we see that, $E(B(n_1, n_2, \cdots, n_m; k_1, k_2, \cdots, k_m)\ U_l^i$ has two components H''_{i1} and H''_{i2}, where, for,

$i \neq m, V(U_l^i) = \{(t_{i-1} + (l-1)k_i + 1, t_{i-1} + (l-1)k_i + 2, \cdots, t_{i-1} + lk_i - 2\}$

$i = m, V(U_l^i) = \{(t_{i-1} + (l-1)k_i + 1, t_{i-1} + (l-1)k_i + 2, \cdots, t_{i-1} + lk_i - 2\}$

$i = m, V(U_l^i) = \{(t_{i-1} + (l-1)k_i + 1, t_{i-1} + (l-1)k_i + 2, \cdots, t_{i-1} + lk_i - 3\}$

Let $G''_{i1} = f^{-1}(H''_{i1})$ and $G''_{i2} = f^{-1}(H''_{i2})$. By lemma 2.10, G''_{i1} is an optimal set and each U_l^i satisfies conditions (i), (ii) and (iii) of the Congestion Lemma. Therefore, $EC_f(U_l^i)$ is minimum.

The proof of the following theorem is an easy consequence of Embedding Algorithm B.

Theorem 4.2. *The exact wirelength of Q^r into $B(n_1, n_2, \cdots, n_m; k_1, k_2, \cdots, k_m)$, is given by*

$$WL(Q^r, B(n_1, n_2, \cdots, n_m; k_1, k_2, \cdots, k_m)) =$$

$$r[2^r - 2(n_1 + n_2 + \cdots + n_m) - 1] \qquad + r[\sum_{i=1}^{m} k_i - m]$$

$$-2[\sum_{i=1}^{m} r_i 2^{r_i - 1} - r_i] - (r_m - 1) \quad + r[\sum_{i=1}^{m} k_i] - 2[\sum_{i=1}^{m} r_i 2^{r_i - 1}] - r_m$$

5 Conclusion

In this paper, we present an algorithm for embedding the hypercubes into the Extended Banana Trees . The algorithm which generates these embedding is not only fast and efficient but also simple and elegant.

References

[1] Y.Saad, M. H.Schultz, *Topological properties of hypercubes*, IEEE, Vol. 37, 867 - 872, 1988.

[2] J. D. Chavez and R.Trapp, *The cyclic cutwidth of trees*, Discrete Appl. Math., Vol. 87, 25 - 32, 1998.

[3] Y. C. Tseng, Y. S. Chen, T. Y. Juang and C. J. Chang,*Congestion - free, dilation-2 embedding of complete binary trees into star graphs*, Networks, Vol. 33, 221 - 231, 1999.

[4] S. L. Bezrukov, J. D. Chavez, L. H. Harper, M. Rottger and U. P. Schroeder, pp. *Embedding of hypercubes into grids*, MFCS, Electronic Edition Springer, Lecture Notes in Computer Science 1450, 693-701, 1998.

[5] J. Opatrny and D. Sotteau, *Embeddings of complete binary trees into grids and extended grids with total vertex congestion 1*, Discrete Appl. Math., Vol. 98, 237-254. 2000.

[6] M. Rottger and U. P. Schroeder, *Efficient embeddings of grids into grids*, Discrete Appl. Math., Vol. 108, 143-173, 2001.

[7] S. L. Bezrukov, B. Monien, W. Unger and G. Wechsung, *Embedding ladders and caterpillars into the hypercube*, Discrete Appl. Math., Vol. 83, 21-29, 1998.

[8] T. Dvořák, *Dense sets and embedding binary trees into hypercubes*, Discrete Applied Mathematics, Vol. 155, no. 4, 506 - 514, 2007.

[9] S. L. Bezrukov, *Embedding complete trees into the hypercube*, Discrete Applied Mathematics, Vol. 110, no. 2-3, 101 - 119, 2001.

[10] J.-F. Fang and K.-C. Lai, *Embedding the incomplete hypercube in books*, Information Sciences, Vol. 96, 1 - 6, 2005.

[11] I. Rajasingh, B. Rajan and R.S. Rajan, *On embedding of m-sequencial k-ary trees into hypercubes*, Applied Mathematics, Vol. 1, no. 6, 499 - 503, 2010.

[12] A.K. Gupta, D. Nelson and H. Wang, *Efficient embeddings of ternary trees into hypercubes*, Journal of Parallel and Distributed Computing, Vol. 63, no. 6, 619 - 629, 2003.

[13] P. Manuel, *Minimum average congestion of enhanced and augmented bypercube into complete binary tree*, Discrete Applied Mathematics, Vol. 159, no. 5, 360 - 366, 2010.

[14] I. Rajasingh, P. Manuel, M. Arockiaraj and B. Rajan, *Embeddings of Circulant networks*, Journal of Combinatorial Optimization, Vol. 11, no. 3, 360 - 366, 2011.

[15] P. Manuel, M. Arockiaraj, I. Rajasingh and B. Rajan, *Embedding hypercubes into cylinders, snakes and caterpillars for minimizing wirelength*, Discrete Appl. Math., Vol. 159, no. 17, 2109 - 2116, 2011.

[16] P. Manuel, I. Rajasingh, B. Rajan and H. Mercy, *Exact wirelength of hypercube on a grid*, Discrete Applied Mathematics, Vol. 157, no.7, 1486 - 1495, 2009.

[17] M.R. Garey and D.S. Johnson, *Computers and Intractability*, A Guide to the Theory of NP-Completeness, Freeman, San Francisco 1979.

[18] S.L. Bezrukov, S.K. Das and R. Elsässer, *An edge-isoperimetric problem for powers of the Petersen graph*, Annals of Combinatorics, Vol. 4, 153 - 169, 2000.

[19] J.-M. Xu, *Topological Structure and Analysis of Interconnection Networks*, Kluwer Academic Publishers, 2001.

[20] I. Rajasingh, B. Rajan and R. S. Rajan,*Embedding of Hypercubes into Necklace, Windmill and Snake Graphs, Information Processing Letters*, - Submitted.

[21] W. C. Chen, H. I. Lu, and Y. N. Yeh,*Operations of interlaced trees and graceful trees*, Southeast Asian Bull. Math., Vol. 21, 337 - 348, 1997.

[22] J. J. Jesintha, *New Classes of Graceful Trees*, Ph. D. Thesis, Anna University, Chennai, India, 2005.

[23] J. Quadras, *Embeddings and interconnection networks*, Ph. D. Thesis, University of Madras, Chennai, India, 2004.