Discrete Optimization 17 (2015) 1–13

Contents lists available at ScienceDirect

Discrete Optimization

www.elsevier.com/locate/disopt

Embedding hypercubes and folded hypercubes onto Cartesian product of certain trees

Micheal Arockiaraj[a], Jasintha Quadras[b], Indra Rajasingh[c], Arul Jeya Shalini[b,*]

[a] Department of Mathematics, Loyola College, Chennai 600034, India
[b] Department of Mathematics, Stella Maris College, Chennai 600086, India
[c] School of Advanced Sciences, VIT University, Chennai 600127, India

ARTICLE INFO

Article history:
Received 11 December 2013
Received in revised form 7 November 2014
Accepted 10 March 2015
Available online 13 April 2015

Keywords:
Embedding
Edge isoperimetric problem
Folded hypercubes
1-rooted complete binary trees
Sibling trees
Cartesian product

ABSTRACT

The hypercube network is one of the most popular interconnection networks since it has simple structure and is easy to implement. The folded hypercube is an important variation of the hypercube. Interconnection networks play a major role in the performance of distributed memory multiprocessors and the one primary concern for choosing an appropriate interconnection network is the graph embedding ability. A graph embedding of a guest graph G into a host graph H is an injective map on the vertices such that each edge of G is mapped into a path of H. The wirelength of this embedding is defined to be the sum of the lengths of the paths corresponding to the edges of G. In this paper we embed hypercube and folded hypercube onto Cartesian product of trees such as 1-rooted complete binary tree and path, sibling tree and path to minimize the wirelength.

1. Introduction

The problem of efficiently implementing parallel algorithms on parallel computers has been studied as a graph embedding problem. The computational structure of a parallel algorithm A is represented by a graph G_A and the interconnection network of a parallel computer N is represented by a graph H_N. An embedding of G_A into H_N describes the working of the parallel algorithm A when implemented on N [1]. Such a simulation problem can be mathematically formulated as follows: Given a guest graph G and a host graph H. An embedding of G into H is an ordered pair $\prec f, P_f \succ$, where f is an injective map from $V(G)$ to $V(H)$ and P_f is also an injective map from $E(G)$ to $\{P(f(u), f(v)) : P(f(u), f(v))$ is a path in H between $f(u)$ and $f(v)$ for $(u, v) \in E(G)\}$ [2–5]. See Fig. 1. An edge congestion of an embedding $\prec f, P_f \succ$ of G into H is the maximum number of edges of the graph G that are embedded on any single edge of H. Let

* Corresponding author.
E-mail address: aruljeyashalini@gmail.com (A.J. Shalini).

http://dx.doi.org/10.1016/j.disopt.2015.03.001
1572-5286/

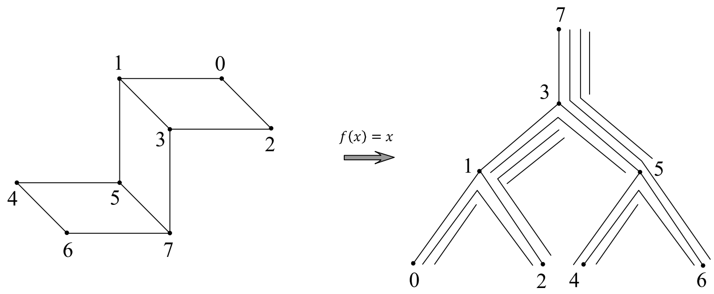

Fig. 1. Embedding the wiring of a ladder into the 1-rooted complete binary tree with $f(x) = x$ and $P_f(0,1) = (0,1)$, $P_f(0,2) = (0,1,2)$, $P_f(1,3) = (1,3)$, $P_f(1,5) = (1,3,5)$, $P_f(2,3) = (2,1,3)$, $P_f(3,7) = (3,7)$, $P_f(4,5) = (4,5)$, $P_f(4,6) = (4,5,6)$, $P_f(5,7) = (5,3,7)$, $P_f(6,7) = (6,5,3,7)$.

$EC_{\prec f,P_f \succ}(e)$ denote the number of edges (u,v) of G such that the path $P(f(u), f(v))$ contains the edge e in H [6]. In other words, $EC_{\prec f,P_f \succ}(e) = |\{(u,v) \in E(G) : e \in P(f(u), f(v))\}|$.

The performance of an embedding can be measured by dilation, expansion and edge congestion sum (wirelength). The dilation of an embedding $\prec f, P_f \succ$ is defined as $dil_{\prec f,P_f \succ}(G,H) = \max\{|P(f(u), f(v))| : (u,v) \in E(G)\}$. The smaller the dilation of an embedding is, the shorter the communication delay that the graph H simulates the graph G. The expansion of an embedding $\prec f, P_f \succ$ is defined as $Exp_{\prec f,P_f \succ}(G,H) = |V(H)| / |V(G)|$. Expansion measures the processor utilization. The smaller the expansion of an embedding is, the more efficient the processor utilization that the graph H simulates the graph G [7].

Combinatorial isoperimetric problems arise frequently in communications engineering, computer science, physical sciences and mathematics. Layout problems arise in electrical engineering when one takes the wiring diagram for some electrical circuit and lay it out on a chassis. A wiring diagram is essentially a graph, the electrical components being the vertices and the wires connecting them being the edges [8]. See Fig. 1.

The wirelength [6,8] of an embedding $\prec f, P_f \succ$ of G into H is given by

$$WL_{\prec f,P_f \succ}(G,H) = \sum_{(u,v) \in E(G)} |P(f(u), f(v))| = \sum_{e \in E(H)} EC_{\prec f,P_f \succ}(e).$$

The minimum wirelength of G into H is defined as $WL(G,H) = \min WL_{\prec f,P_f \succ}(G,H)$ where the minimum is taken over all embeddings f and P_f of G into H.

Embedding problems have been considered for binary trees into hypercubes [9–16], binomial trees into hypercubes [17,18], generalized ladders into hypercubes [19,20], hypercubes into cycles [21,22], hypercubes into grids [6,23,24], hypercubes into cylinders, snakes and caterpillars [25], hypercubes into certain trees [26], m-sequential k-ary trees into hypercubes [27], folded hypercubes into grids [28] and complete binary trees into folded hypercubes [1].

Among the interconnection networks of parallel computers, the binary hypercube has received much attention. An important property of the hypercube which makes it popular, is its ability to efficiently simulate the message routings of other interconnection networks and hence becomes the first choice of topological structure of parallel processing and computing systems. The machine based on hypercubes such as the Cosmic Cube from Caltech, the iPSC/2 from Intel and Connection Machines have been implemented commercially [1]. For $n \geq 1$, let Q_n denote the n-dimensional hypercube. The vertex set of Q_n is formed by the collection of all n-string binary representations. Two vertices $x, y \in V(Q_n)$ are adjacent if and only if the corresponding binary representations differ exactly in one bit [5]. Equivalently if $|V(Q_n)| = 2^n$ then the vertices of Q_n can also be identified with integers $0, 1, \ldots, 2^n - 1$ so that if a pair of vertices i and j are adjacent then $i - j = \pm 2^p$ for some $p \geq 0$. An incomplete hypercube on i vertices of Q_n is the graph induced by $\{0, 1, \ldots, i-1\}$ and is denoted by L_i, $1 \leq i \leq 2^n$ [29]. See Fig. 2.

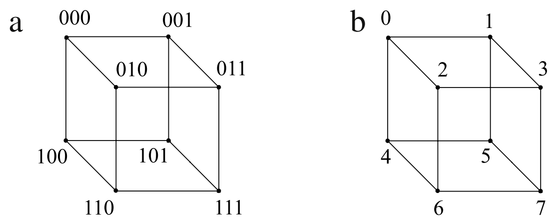

Fig. 2. The 3-dimensional hypercube with vertices labeled (a) binary and (b) decimal.

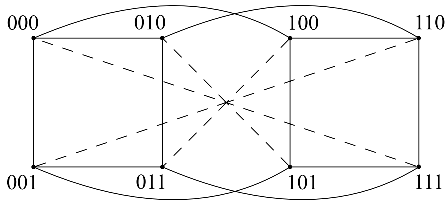

Fig. 3. The 3-dimensional folded hypercube with dotted lines represent complementary edges.

For two vertices $x = x_1x_2 \ldots x_n$ and $y = y_1y_2 \ldots y_n$, (x, y) is a complementary edge if and only if the bits of x and y are complements of each other, that is, $y_i = \overline{x}_i$ for each $i = 1, 2, \ldots, n$. The n-dimensional folded hypercube, denoted by FQ_n, is an undirected graph obtained from Q_n by adding all complementary edges [5]. It is easy to see that any n-dimensional folded hypercube FQ_n can be viewed as $G(0Q_{n-1},\ 1Q_{n-1}; C \cup \overline{C})$ where $0Q_{n-1}$ and $1Q_{n-1}$ are two $(n-1)$-dimensional hypercubes with the prefix 0 and 1 of each vertex, respectively, and $C = \{(0u, 1u) : 0u \in V(0Q_{n-1})$ and $1u \in V(1Q_{n-1})\}$, $\overline{C} = \{(0u, 1\overline{u}) : 0u \in V(0Q_{n-1})$ and $1\overline{u} \in V(1Q_{n-1})\}$ [30]. See Fig. 3.

A tree is a connected graph that contains no cycles. Trees are the most fundamental graph-theoretic models used in many fields: information theory, automatics classification, data structure and analysis, artificial intelligence, design of algorithms, operation research, combinatorial optimization, theory of electrical networks, and design of network [5]. The most common type of tree is the binary tree.

The importance of the hypercubes and folded hypercubes in the topic of interconnection networks and the importance of trees in the topic of data structures, motivated the authors to study the embedding of hypercubes and folded hypercubes into Cartesian product of certain trees. The rest of this paper is organized as follows. In the next section, some background of the edge isoperimetric problem and a technique to compute the minimum wirelength are given. Section 3 gives the minimum wirelength of embedding hypercubes and folded hypercubes into Cartesian product of 1-rooted complete binary tree and path in linear time. In Section 4, the result of Section 3 is extended to Cartesian product of sibling tree and path. The last section concludes the whole paper.

2. Background and technique

One of the first needs of edge isoperimetric problems was discovered by Harper [31]. Suppose we have to send the numbers $0, 1, \ldots, 2^n - 1$ through a binary channel and we have to assign the numbers to vertices of the hypercube Q_n. For example, we may assume that these numbers were taken from the output of an analog to digital converter. It is assumed that only single errors are likely in a transmitted word and each of the n positions may be disturbed with probability p. If the n-tuple assigned to i was transmitted and the n-tuple assigned to j was received, then $|i - j|$ is the absolute value of the error. The goal is to find an assignment so that the average absolute error in transmission is minimized under the condition that the choice of the 2^n numbers is equally probable. Thus one comes to the problem of constructing a bijective mapping $\varphi : V(Q_n) \to \{0, 1, \ldots, 2^n - 1\}$ so that the sum $\sum_{(u,v)\in E(Q_n)} |\varphi(u) - \varphi(v)|$ is minimized. Such type of problems can be formulated for an arbitrary connected graph G and the sum above may be referred to the total wirelength in a linear layout of the graph G.

The following two versions of the edge isoperimetric problems (NP-complete) of a graph $G(V, E)$ have been considered in the literature [32,33].

Version 1 (**Minimum cut problem**): 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 $\theta_G(m) = |\theta_G(A)|$.

It is interesting to note that $\theta_G(\lfloor |V| /2\rfloor)$ yields bisection width of G [33].

Version 2 (**Induced edge problem**): 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 $I_G(m) = |I_G(A)|$.

We call such a set A optimal. Clearly, if a subset of vertices is optimal with respect to Version 1, then its complement is also an optimal set. However, it is not true for Version 2 in general, although this is indeed the case if the graph is regular [32].

Theorem 1 *([8,31,34,35]). For $1 \le i \le 2^n$, $L_i = \{0, 1, \ldots, i - 1\}$ is an optimal set in the hypercube Q_n.*

Theorem 2 *([28]). For $1 \le i \le 2^n$, $L_i = \{0, 1, \ldots, i - 1\}$ is an optimal set in the folded hypercube FQ_n.*

Lemma 1 *([6,28,36]). Let $m = 2^{t_1} + 2^{t_2} + \cdots + 2^{t_l}$ be such that $n \ge t_1 > t_2 > \cdots > t_l \ge 0$. Then*

$$\text{(a) } |E(Q_n[L_m])| = [t_1 \cdot 2^{t_1-1} + t_2 \cdot 2^{t_2-1} + \cdots + t_l \cdot 2^{t_l-1}] + [2^{t_2} + 2 \cdot 2^{t_3} + \cdots + (l - 1)2^{t_l}]$$

$$\text{(b) } |E(FQ_n[L_m])| = \begin{cases} |E(Q_n[L_m])| & \text{if } m \le 2^{n-1} \\ |E(Q_n[L_m])| + m - 2^{n-1} & \text{if } m > 2^{n-1}. \end{cases}$$

Let $E^k(G)$ denote a collection of edges of a graph G with each edge in G repeated exactly k times. This type of partitioning the edges of a graph is called generalized edge partition. The following lemma provides the general method of partitioning the edges of a graph and how this method can be effectively used to solve the wirelength problem [6,37,38].

Lemma 2 (*k-Partition Wirelength Lemma*). *Let $\prec f, P_f \succ$ be an embedding of an r-regular graph G into a graph H. Let $\{S_1, S_2, \ldots, S_m\}$ be a partition of $E^k(H)$ such that each S_i is an edge cut of H. For $1 \le i \le m$, the removal of edges of S_i splits H into 2 components H_{i1} and H_{i2} and let $G_{i1} = G[f^{-1}(H_{i1})]$* and *$G_{i2} = G[f^{-1}(H_{i2})]$, where each S_i satisfies the following conditions.*

(i) *For every edge $(a, b) \in G_{ij}$, $j = 1, 2$, $P(f(a), f(b))$* has no edges in *S_i.*
(ii) *For every edge (a, b)* in *G* with *$a \in G_{i1}$* and *$b \in G_{i2}$, $P(f(a), f(b))$* has exactly one edge in *S_i.*
(iii) Either *G_{i1} or G_{i2} is an optimal set.*

Then $EC_{\prec f,P_f\succ}(S_i)$ is minimum where

$$EC_{\prec f,P_f\succ}(S_i) = \sum_{e\in S_i} EC_{\prec f,P_f\succ}(e) = r\,|V(G_{i1})| - 2\,|E(G_{i1})| = r\,|V(G_{i2})| - 2\,|E(G_{i2})|$$

and

$$WL(G,H)=\frac{1}{k}\sum_{i=1}^{m}EC_{\prec f,P_f\succ}(S_i).$$

3. Embedding hypercubes and folded hypercubes into Cartesian product of 1-rooted complete binary trees and paths

In this section we embed hypercubes and folded hypercubes into Cartesian product of 1-rooted complete binary trees and paths to minimize the wirelength.

Let T be a rooted tree. Suppose that vertex u of T adjacent to v lies in the level below v, we say that u is a child of v and v is the parent of u. Suppose that there is a path from v to w in T such that w lies below v, we say that w is a descendant of v and v is an ancestor of w. A vertex with no children is called a leaf. All other vertices are called internal vertices. A binary tree is a rooted tree in which each vertex has at most two children and each child is designated as its left child or right child. Binary trees are widely used in data structures because they are easily stored, easily manipulated, and easily retrieved. Also, many operations such as searching and storing can be easily performed on tree data structures. Furthermore, binary trees appear in communication pattern of divide-and-conquer type algorithms, functional and logic programming, and graph algorithms [5].

For any non-negative integer n, the complete binary tree of height n, denoted by T_n, is the binary tree where each internal vertex has exactly two children and all the leaves are at the same level. Clearly, a complete binary tree T_n has n levels and level i, $1\le i\le n$, contains 2^{i-1} vertices. Thus T_n has exactly 2^n-1 vertices. The 1-rooted complete binary tree BT_n is obtained from a complete binary tree T_n by attaching to its root a pendant edge. The new vertex is called the root of BT_n and is considered to be at level 0 [26]. Hence BT_n has 2^n vertices.

We first prove a few basic lemmas to attain the main result.

Lemma 3. *Let n_1 be a fixed positive integer such that $n_1\le n$. For $j=1,2,\ldots,n_1$ and $i=0,1,\ldots,2^{n_1-j}-1$, $TX_i^j=\{k\cdot 2^{n_1}+i\cdot 2^j+l : 0\le k\le 2^{n-n_1}-1, 0\le l\le 2^j-2\}$ is an optimal set on $2^{n-n_1}(2^j-1)$ vertices in Q_n and also in FQ_n.*

Proof. Define $\varphi : TX_i^j\to L_{2^{n-n_1}(2^j-1)}$ by $\varphi(k\cdot 2^{n_1}+i\cdot 2^j+l)=k+l\cdot 2^{n-n_1}$. If the binary representation of

$$\mathbf{x}=k\cdot 2^{n_1}+i\cdot 2^j+l$$

is

$$\alpha_1\alpha_2\ldots\alpha_{n-n_1}\beta_1\beta_2\ldots\beta_{n_1-j}\gamma_1\gamma_2\ldots\gamma_j$$

then we prove that the binary representation of

$$\varphi(\mathbf{x})=k+l\cdot 2^{n-n_1}$$

is

$$\underbrace{000\ldots\ldots.000}_{n_1-j\ \text{times}}\gamma_1\gamma_2\ldots\gamma_j\alpha_1\alpha_2\ldots\alpha_{n-n_1}.$$

For instance, $n = 6,\ n_1 = 4,\ j = 2$ and $i = 0$ is shown as follows:

$$TX_0^2 = \begin{bmatrix} 0 & 1 & 2 \\ 16 & 17 & 18 \\ 32 & 33 & 34 \\ 48 & 49 & 50 \end{bmatrix}$$

with binary representation

$$TX_0^2 = \begin{bmatrix} \mathbf{0000}00 & \mathbf{0000}01 & \mathbf{0000}10 \\ \mathbf{0100}00 & \mathbf{0100}01 & \mathbf{0100}10 \\ \mathbf{1000}00 & \mathbf{1000}01 & \mathbf{1000}10 \\ \mathbf{1100}00 & \mathbf{1100}01 & \mathbf{1100}10 \end{bmatrix}.$$

Then

$$\varphi(TX_0^2) = \begin{bmatrix} 0 & 4 & 8 \\ 1 & 5 & 9 \\ 2 & 6 & 10 \\ 3 & 7 & 11 \end{bmatrix}$$

with binary representation

$$\varphi(TX_0^2) = \begin{bmatrix} \mathbf{00}0000 & \mathbf{00}0100 & \mathbf{00}1000 \\ \mathbf{00}0001 & \mathbf{00}0101 & \mathbf{00}1001 \\ \mathbf{00}0010 & \mathbf{00}0110 & \mathbf{00}1010 \\ \mathbf{00}0011 & \mathbf{00}0111 & \mathbf{00}1011 \end{bmatrix}.$$

Suppose $k = 0,\ i = 0$ in $\mathbf{x}$. Then $\mathbf{x} = l$. Since $0 \le l \le 2^j - 2$, the binary representation of $\mathbf{x}$ is

$$\underbrace{000\ldots\ldots.000}_{n-n_1 \text{ times}}\underbrace{000\ldots\ldots.000}_{n_1-j \text{ times}}\gamma_1\gamma_2\ldots\gamma_j.$$

This implies that the binary representation of $\varphi(\mathbf{x}) = l \cdot 2^{n-n_1}$ is

$$\underbrace{000\ldots\ldots.000}_{n_1-j \text{ times}}\gamma_1\gamma_2\ldots\gamma_j\underbrace{000\ldots\ldots.000}_{n-n_1 \text{ times}}.$$

Suppose $k = 0$ in $\mathbf{x}$. Then $\mathbf{x} = i \cdot 2^j + l$. Since $j = 1, 2, \ldots, n_1,\ i = 0, 1, \ldots, 2^{n_1-j} - 1,\ 0 \le l \le 2^j - 2$, the binary representation of $\mathbf{x}$ is

$$\underbrace{000\ldots\ldots.000}_{n-n_1 \text{ times}}\beta_1\beta_2\ldots\beta_{n_1-j}\gamma_1\gamma_2\ldots\gamma_j.$$

This implies that the binary representation of $\varphi(\mathbf{x}) = l \cdot 2^{n-n_1}$ is

$$\underbrace{000\ldots\ldots.000}_{n_1-j \text{ times}}\gamma_1\gamma_2\ldots\gamma_j\underbrace{000\ldots\ldots.000}_{n-n_1 \text{ times}}.$$

Suppose $i = 0$ in $\mathbf{x}$. Then $\mathbf{x} = k \cdot 2^{n_1} + l$. Since $0 \le k \le 2^{n-n_1} - 1,\ 0 \le l \le 2^j - 2$, the binary representation of $\mathbf{x}$ is

$$\alpha_1\alpha_2\ldots\alpha_{n-n_1}\underbrace{000\ldots\ldots.000}_{n_1-j \text{ times}}\gamma_1\gamma_2\ldots\gamma_j.$$

This implies that the binary representation of $\varphi(\mathbf{x}) = k + l \cdot 2^{n-n_1}$ is

$$\underbrace{000\ldots\ldots.000}_{n_1-j \text{ times}}\gamma_1\gamma_2\ldots\gamma_j\alpha_1\alpha_2\ldots\alpha_{n-n_1}.$$

Similarly we can discuss all other cases.

We now show that the sets TX_i^j and $L_{2^{n-n_1}(2^j-1)}$ are isomorphic by considering into three parts.

Part A: Let the binary representations of two numbers $\mathbf{x}$ and $\mathbf{y}$ be respectively

$$\underbrace{\square\square\square\square\square\ \boxed{0(\text{resp.}1)}\ \square\square\square\square\square}_{n-n_1 \text{ bits}}\ \underbrace{\square\square\square\square\square\square\square}_{n_1-j \text{ bits}}\ \underbrace{\square\square\square\square\square\square}_{j \text{ bits}}$$

and

$$\underbrace{\square\square\square\square\square\ \boxed{1(\text{resp.}0)}\ \square\square\square\square\square}_{n-n_1 \text{ bits}}\ \underbrace{\square\square\square\square\square\square\square}_{n_1-j \text{ bits}}\ \underbrace{\square\square\square\square\square\square}_{j \text{ bits}}$$

Then the binary representations of numbers $\varphi(\mathbf{x})$ and $\varphi(\mathbf{y})$ are respectively

$$\underbrace{000\ldots\ldots000}_{n_1-j \text{ bits}}\ \underbrace{\square\square\square\square\square\square}_{j \text{ bits}}\ \underbrace{\square\square\square\square\square\ \boxed{1(\text{resp.}0)}\ \square\square\square\square\square}_{n-n_1 \text{ bits}}$$

and

$$\underbrace{000\ldots\ldots000}_{n_1-j \text{ bits}}\ \underbrace{\square\square\square\square\square\square}_{j \text{ bits}}\ \underbrace{\square\square\square\square\square\ \boxed{0(\text{resp.}1)}\ \square\square\square\square\square}_{n-n_1 \text{ bits}}$$

Hence the binary representations of two numbers $\mathbf{x}$ and $\mathbf{y}$ differ in exactly one bit if and only if the binary representations of $\varphi(\mathbf{x})$ and $\varphi(\mathbf{y})$ differ in exactly one bit. Therefore $(\mathbf{x}, \mathbf{y})$ is an edge in TX_i^j if and only if $(\varphi(\mathbf{x}), \varphi(\mathbf{y}))$ is an edge in $L_{2^{n-n_1}(2^j-1)}$. Hence TX_i^j and $L_{2^{n-n_1}(2^j-1)}$ are isomorphic. By Theorem 1, TX_i^j is an optimal set in Q_n for $j = 1, 2, \ldots, n_1$ and $i = 0, 1, \ldots, 2^{n_1-j} - 1$.

When $j = 1, 2, \ldots, n_1-1$ and $i = 0, 1, \ldots, 2^{n_1-j}-1$, the above argument holds for FQ_n and by Theorem 2, TX_i^j is an optimal set in FQ_n. Suppose $j = n_1$ then $i = 0$. In this case the binary representations of two numbers $\mathbf{x}$ and $\mathbf{y}$ differ in exactly one bit or the sum of the binary representations of $\mathbf{x}$ and $\mathbf{y}$ is $111\ldots11$ (n bits) if and only if the binary representations of $\varphi(\mathbf{x})$ and $\varphi(\mathbf{y})$ differ in exactly one bit or the sum of the binary representations of $\varphi(\mathbf{x})$ and $\varphi(\mathbf{y})$ is $111\ldots11$ (n bits). Therefore $(\mathbf{x}, \mathbf{y})$ is an edge in TX_i^j if and only if $(\varphi(\mathbf{x}), \varphi(\mathbf{y}))$ is an edge in $L_{2^{n-n_1}(2^j-1)}$. By Theorem 2, TX_i^j is an optimal set in FQ_n.

Part B: Let the binary representations of two numbers $\mathbf{x}$ and $\mathbf{y}$ be respectively

$$\underbrace{\square\square\square\square\square\square\square\square}_{n-n_1 \text{ bits}}\ \underbrace{\square\square\square\square\square\square\square}_{n_1-j \text{ bits}}\ \underbrace{\square\square\square\ \boxed{0(\text{resp.}1)}\ \square\square\square}_{j \text{ bits}}$$

and

$$\underbrace{\square\square\square\square\square\square\square\square}_{n-n_1 \text{ bits}}\ \underbrace{\square\square\square\square\square\square\square}_{n_1-j \text{ bits}}\ \underbrace{\square\square\square\ \boxed{1(\text{resp.}0)}\ \square\square\square}_{j \text{ bits}}$$

Then the binary representations of numbers $\varphi(\mathbf{x})$ and $\varphi(\mathbf{y})$ are respectively

$$\underbrace{000\ldots\ldots000}_{n_1-j \text{ bits}}\ \underbrace{\square\square\square\ \boxed{0(\text{resp.}1)}\ \square\square\square}_{j \text{ bits}}\ \underbrace{\square\square\square\square\square\square\square\square}_{n-n_1 \text{ bits}}$$

and

$$\underbrace{000\ldots\ldots000}_{n_1-j\text{ bits}}\ \underbrace{\square\square\square\ \boxed{1(\text{resp.}0)}\ \square\square\square}_{j\text{ bits}}\ \underbrace{\square\square\square\square\square\square\square\square}_{n-n_1\text{ bits}}$$

By proceeding as in Part A, we get TX_i^j is an optimal set in Q_n and FQ_n.

Part C: Let the binary representations of two numbers **x** and **y** be respectively

$$\underbrace{\square\square\square\square\square\square\square\square}_{n-n_1\text{ bits}}\ \underbrace{\square\square\square\ \boxed{0(\text{resp.}1)}\ \square\square\square\square}_{n_1-j\text{ bits}}\ \underbrace{\square\square\square\square\square\square}_{j\text{ bits}}$$

and

$$\underbrace{\square\square\square\square\square\square\square\square}_{n-n_1\text{ bits}}\ \underbrace{\square\square\square\ \boxed{1(\text{resp.}0)}\ \square\square\square\square}_{n_1-j\text{ bits}}\ \underbrace{\square\square\square\square\square\square}_{j\text{ bits}}$$

Since i and j are fixed for TX_i^j, this part does not occur. $\square$

Lemma 4. *For $j = 1, 2, \ldots, n_1$ and $i = 0, 1, \ldots, 2^{n_1-j} - 1$, $\left|E(Q_n[TX_i^j])\right| = 2^{n-n_1-1}\{2^j(n - n_1 + j) - (n - n_1 + 2j)\}$ where $n_1 \leq n$.*

Proof. By Lemma 3, TX_i^j is an optimal set on $2^{n-n_1}(2^j - 1)$ vertices in Q_n and hence $\left|E(Q_n[TX_i^j])\right| = \left|E(Q_n[L_{2^{n-n_1}(2^j-1)}])\right|$. Since $2^{n-n_1}(2^j - 1) = 2^{n-n_1} + 2^{n-n_1+1} + \cdots + 2^{n-n_1+j-1}$, by Lemma 1, $\left|E(Q_n[TX_i^j])\right| = 2^{n-n_1-1}\{2^j(n - n_1 + j) - (n - n_1 + 2j)\}$. $\square$

Lemma 5. *Let n_1 be a fixed positive integer such that $n_1 \leq n$.* (a) *For $j = 1, 2, \ldots, n_1 - 1$ and $i = 0, 1, \ldots, 2^{n_1-j}-1$, $\left|E(FQ_n[TX_i^j])\right| = \left|E(Q_n[TX_i^j])\right|$* (b) *For $j = n_1$ and $i = 0$, $\left|E(FQ_n[TX_i^j])\right| = \left|E(Q_n[TX_i^j])\right| + 2^{n-1} - 2^{n-n_1}$.*

Proof. By Lemma 3, TX_i^j is an optimal set on $2^{n-n_1}(2^j - 1)$ vertices in FQ_n and hence $\left|E(FQ_n[TX_i^j])\right| = \left|E(FQ_n[L_{2^{n-n_1}(2^j-1)}])\right|$. The proof follows from Lemmas 1 and 4. $\square$

Theorem 3. *Let n_1 be a fixed positive integer such that $n_1 \leq n$. The minimum wirelength of Q_n and FQ_n into $BT_{n_1} \times P_m$, $m = 2^{n-n_1}$ are given by* (a) $WL(Q_n, BT_{n_1} \times P_m) = 2^{n-1}\{n_1^2 - 3n_1 + 8 + n(2^{n-n_1} - 1)\} - 2^{n-n_1}(n_1+4) - 2\sum_{k=1}^{m-1} |E(Q_n[L_{k2^{n_1}}])|$ *and* (b) $WL(FQ_n, BT_{n_1} \times P_m) = 2^{n-1}\{n_1^2 - n_1 + 4 + (n+1)(2^{n-n_1} - 1)\} - 2^{n-n_1}(n_1 + 1) - 2\sum_{k=1}^{m-1} |E(FQ_n[L_{k2^{n_1}}])|$.

Proof. The proof consists of three parts namely (1) embedding Algorithm (2) proof of correctness and (3) computation of wirelength.

(1) Embedding algorithm: Label the vertex $x_1x_2\ldots x_n$ of Q_n as $\sum_{i=1}^{n} x_i \cdot 2^{n-i}$ (equivalent to lexicographic order [39] from 0 to $2^n - 1$). As $V(FQ_n) = V(Q_n)$, the label of FQ_n is same as that of Q_n. Label the vertices of $BT_{n_1} \times P_m$, $m = 2^{n-n_1}$, as follows: Let $BT_{n_1}^1, BT_{n_1}^2, \ldots, BT_{n_1}^m$ be the m vertex disjoint copies of 1-rooted complete binary tree BT_{n_1} in $BT_{n_1} \times P_m$. Label the vertices of $BT_{n_1}^i$, $1 \leq i \leq m$, by inorder traversal [40,41] from $(i-1)2^{n_1}$ to $i2^{n_1} - 1$. See Fig. 4.

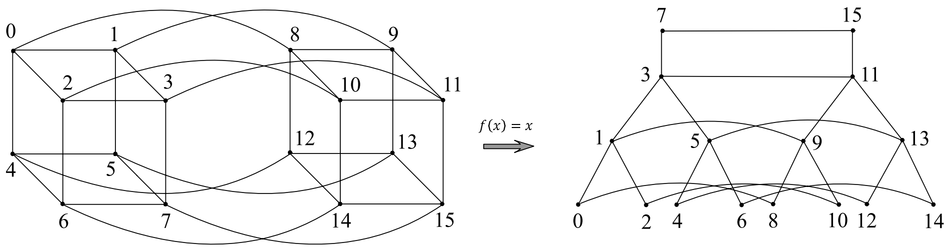

Fig. 4. Embedding of Q_4 with vertices labeled by lexicographic order into $BT_3 \times P_2$ with vertices labeled by inorder traversal.

Define Lexicographic embedding $Lex = \prec f, P_f \succ$, where

(a) f is a bijective map from $V(Q_n)$ (resp. $V(FQ_n)$) to $V(BT_{n_1} \times P_m)$ by $f(\mathbf{x}) = \mathbf{x}$ and
(b) P_f is an injective map from $E(Q_n)$ (resp. $E(FQ_n)$) to $\{P(f(u), f(v)) : P(f(u), f(v))$ is a shortest path in $BT_{n_1} \times P_m$ between $f(u)$ and $f(v)$ for $(u, v) \in E(Q_n)$ (resp. $E(FQ_n))\}$. (Note that if the shortest path is not unique, then fix any one of the paths.)

(2) Proof of correctness: For $j = 1, 2, \ldots, n_1$ and $i = 0, 1, \ldots, 2^{n_1-j} - 1$, let S_i^j be the set of edges of $BT_{n_1} \times P_m$ such that each edge has one vertex in level $n_1 - j$ and the other vertex in level $n_1 - j + 1$ of $BT_{n_1}^k$, $1 \le k \le m$. Removal of S_i^j leaves $BT_{n_1} \times P_m$ into two components X_i^j and $\overline{X}_i^j$ where $V(X_i^j)$ is TX_i^j. Let G_i^j and $\overline{G}_i^j$ be the inverse images of X_i^j and $\overline{X}_i^j$ under f respectively. By Lemma 3, $V(G_i^j)$ is an optimal set in Q_n and FQ_n. For $k = 1, 2, \ldots, m-1$, let E_k be the set of edges of $BT_{n_1} \times P_m$ such that each edge has one vertex in $BT_{n_1}^k$ and the other vertex in $BT_{n_1}^{k+1}$. Removal of E_k leaves $BT_{n_1} \times P_m$ into two components Y_k and $\overline{Y}_k$ where $V(Y_k)$ is $L_{k2^{n_1}}$. Let H_k and $\overline{H}_k$ be the inverse images of Y_k and $\overline{Y}_k$ under f respectively. By Theorems 1 and 2, $V(H_k)$ is an optimal set in Q_n and FQ_n.

Clearly $E(BT_{n_1} \times P_m) = \{S_i^j : \ j = 1, 2, \ldots, n_1, \ i = 0, 1, \ldots, 2^{n_1-j} - 1\} \cup \{E_k : \ k = 1, 2, \ldots, m-1\}$. Moreover, each edge cut S_i^j and as well as E_k satisfies conditions (i)–(iii) of the 1-partition wirelength lemma. Therefore $WL_{Lex}(Q_n, BT_{n_1} \times P_m) = WL(Q_n, BT_{n_1} \times P_m)$ and $WL_{Lex}(FQ_n, BT_{n_1} \times P_m) = WL(FQ_n, BT_{n_1} \times P_m)$.

(3) Computation of wirelength: We divide this part into two cases.

Case A (**Hypercube**): For $j = 1, 2, \ldots, n_1, \ i = 0, 1, \ldots, 2^{n_1-j} - 1, \ EC_{Lex}(S_i^j) = n \cdot 2^{n-n_1}(2^j - 1) - 2 \cdot 2^{n-n_1-1}\{2^j(n - n_1 + j) - (n - n_1 + 2j)\} = 2^{n-n_1}\{2^j(n_1 - j) + 2j - n_1\}$ and for $k = 1, 2, \ldots, m-1$, $EC_{Lex}(E_k) = n \cdot k2^{n_1} - 2\,|E(Q_n[L_{k2^{n_1}}])|$.

Therefore $WL(Q_n, BT_{n_1} \times P_m) = \sum_{j=1}^{n_1} \sum_{i=0}^{2^{n_1-j}-1} EC_{Lex}(S_i^j) + \sum_{k=1}^{m-1} EC_{Lex}(E_k) = \sum_{j=1}^{n_1} 2^{n-j}\{2^j(n_1 - j) + 2j - n_1\} + \sum_{k=1}^{m-1}\{n \cdot k2^{n_1} - 2\,|E(Q_n[L_{k2^{n_1}}])|\} = 2^{n-1}\{n_1^2 - 3n_1 + 8 + n(2^{n-n_1} - 1)\} - 2^{n-n_1}(n_1 + 4) - 2\sum_{k=1}^{m-1} |E(Q_n[L_{k2^{n_1}}])|$.

Case B (**Folded hypercube**): For $j = 1, 2, \ldots, n_1 - 1, \ i = 0, 1, \ldots, 2^{n_1-j} - 1, \ EC_{Lex}(S_i^j) = (n+1) \cdot 2^{n-n_1}(2^j - 1) - 2 \cdot 2^{n-n_1-1}\{2^j(n - n_1 + j) - (n - n_1 + 2j)\} = 2^{n-n_1}\{2^j(n_1 - j + 1) + 2j - n_1 - 1\}$ and for $j = n_1, \ i = 0, \ EC_{Lex}(S_i^j) = (n_1 + 1)2^{n-n_1}$ and $k = 1, 2, \ldots, m-1, \ EC_{Lex}(E_k) = (n+1) \cdot k2^{n_1} - 2\,|E(FQ_n[L_{k2^{n_1}}])|$.

Therefore $WL(FQ_n, BT_{n_1} \times P_m) = \sum_{j=1}^{n_1-1} \sum_{i=0}^{2^{n_1-j}-1} EC_{Lex}(S_i^j) + EC_{Lex}(S_0^{n_1}) + \sum_{k=1}^{m-1} EC_{Lex}(E_k) = \sum_{j=1}^{n_1-1} 2^{n-j}\ \{2^j(n_1 - j + 1) + 2j - n_1 - 1\} + (n_1 + 1)2^{n-n_1} + \sum_{k=1}^{m-1}\{(n+1) \cdot k2^{n_1} - 2\,|E(Q_n[L_{k2^{n_1}}])|\} = 2^{n-1}\{n_1^2 - n_1 + 4 + (n+1)(2^{n-n_1} - 1)\} - 2^{n-n_1}(n_1 + 1) - 2\sum_{k=1}^{m-1} |E(FQ_n[L_{k2^{n_1}}])|$. $\square$

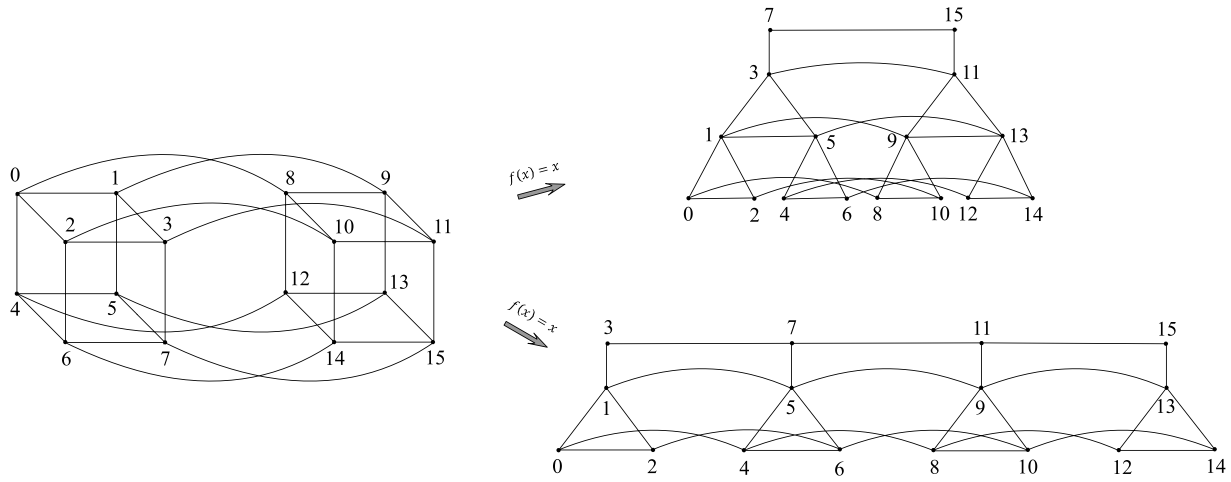

Fig. 5. Embedding of Q_4 with vertices labeled by lexicographic order into $ST_3 \times P_2$ (resp. $ST_2 \times P_4$) with vertices labeled by inorder traversal.

4. Embedding hypercubes and folded hypercubes into Cartesian product of sibling trees and paths

The sibling tree ST_n is obtained from the 1-rooted complete binary tree BT_n by adding edges (sibling edges) between left and right children of the same parent node [26]. Since $V(ST_n) = V(BT_n)$, we show that the Embedding Algorithm of Theorem 3 gives the minimum wirelength of hypercube Q_n (resp. FQ_n) onto Cartesian product tree $ST_{n_1} \times P_m$, $m = 2^{n-n_1}$. See Fig. 5. To show this result, we need the following lemma.

Lemma 6. *For $j = 1, 2, \ldots, n_1 - 1$ and $i = 0, 1, \ldots, 2^{n_1-j-1} - 1$, $STX_i^j = \{k \cdot 2^{n_1} + 2i \cdot 2^j + l,\ k \cdot 2^{n_1} + (2i + 1) \cdot 2^j + l : 0 \le k \le 2^{n-n_1} - 1, 0 \le l \le 2^j - 2\}$ is an optimal set on $2^{n-n_1+1}(2^j - 1)$ vertices in Q_n and FQ_n where $n_1 \le n$.*

Proof. We divide the proof into two parts.

Part 1 (**Hypercube**): By Lemma 3, the sets $\{k \cdot 2^{n_1} + 2i \cdot 2^j + l : 0 \le k \le 2^{n-n_1} - 1, 0 \le l \le 2^j - 2\}$ and $\{k \cdot 2^{n_1} + (2i + 1) \cdot 2^j + l : 0 \le k \le 2^{n-n_1} - 1, 0 \le l \le 2^j - 2\}$ are isomorphic to $L_{2^{n-n_1}(2^j-1)}$. Also the binary representation of $k \cdot 2^{n_1} + 2i \cdot 2^j + l$ and $k \cdot 2^{n_1} + (2i + 1) \cdot 2^j + l$ differ exactly in one bit. Therefore $\left|E(Q_n[STX_i^j])\right| = 2\left|E(Q_n[L_{2^{n-n_1}(2^j-1)}])\right| + 2^{n-n_1}(2^j - 1) = 2^{n-n_1}\{2^j(n - n_1 + j + 1) - (n - n_1 + 2j + 1)\}$. But by Lemma 1, $\left|E(Q_n[L_{2^{n-n_1+1}(2^j-1)}])\right| = 2^{n-n_1}\{2^j(n - n_1 + j + 1) - (n - n_1 + 2j + 1)\}$ and hence by Theorem 1, STX_i^j is an optimal set in Q_n.

Part 2 (**Folded hypercube**): For $j = 1, 2, \ldots, n_1 - 2$ and $i = 0, 1, \ldots, 2^{n_1-j-1} - 1$, the argument of Part 1 holds for FQ_n and by Theorem 2, STX_i^j is an optimal set in FQ_n. Suppose $j = n_1 - 1$ and $i = 0$. Then $STX_i^j = \{k \cdot 2^{n_1} + l,\ k \cdot 2^{n_1} + 2^{n_1-1} + l : 0 \le k \le 2^{n-n_1} - 1, 0 \le l \le 2^{n_1-1} - 2\}$. By Lemma 3, the sets $\{k \cdot 2^{n_1} + l : 0 \le k \le 2^{n-n_1} - 1, 0 \le l \le 2^{n_1-1} - 2\}$ and $\{k \cdot 2^{n_1} + 2^{n_1-1} + l : 0 \le k \le 2^{n-n_1} - 1, 0 \le l \le 2^{n_1-1} - 2\}$ are isomorphic to $L_{2^{n-n_1}(2^{n_1-1}-1)}$. Also for $0 \le k \le 2^{n-n_1} - 1, 0 \le l \le 2^{n_1-1} - 2$, the binary representation of $k \cdot 2^{n_1} + l$ and $k \cdot 2^{n_1} + 2^{n_1-1} + l$ differ exactly in one bit. Further for $0 \le k \le 2^{n-n_1} - 1, 1 \le l \le 2^{n_1-1} - 2$, the sum of the binary representations of $k \cdot 2^{n_1} + l$ and $(2^{n-n_1} - 1 - k) \cdot 2^{n_1} + 2^{n_1-1} + 2^{n_1-1} - 2 - (l - 1)$ is n. Therefore $\left|E(FQ_n[STX_i^j])\right| = 2\left|E(Q_n[L_{2^{n-n_1}(2^{n_1-1}-1)}])\right| + 2^{n-n_1}(2^{n_1-1} - 1) + 2^{n-n_1}(2^{n_1-1} - 2) = 2^{n-1}(n+1) - 2^{n-n_1}(n+n_1+1)$. But by Lemma 1, $\left|E(Q_n[L_{2^{n-n_1+1}(2^{n_1-1}-1)}])\right| = 2^{n-1}(n+1) - 2^{n-n_1}(n+n_1+1)$ and hence by Theorem 2, STX_i^j is an optimal set in FQ_n. □

Theorem 4. *Let n_1 be a fixed positive integer such that $n_1 \leq n$. The minimum wirelength of Q_n and FQ_n into $ST_{n_1} \times P_m$, $m = 2^{n-n_1}$ are given by* (a) $WL(Q_n, ST_{n_1} \times P_m) = 2^{n-1}\{n_1^2 - 4n_1 + 10 + n(2^{n-n_1} - 1)\} - 2^{n-n_1}(n_1 + 5) - 2\sum_{k=1}^{m-1} |E(Q_n[L_{k2^{n_1}}])|$ *and* (b) $WL(FQ_n, ST_{n_1} \times P_m) = 2^{n-1}\{n_1^2 - 2n_1 + 5 + (n + 1)(2^{n-n_1} - 1)\} - n_1 2^{n-n_1} - 2\sum_{k=1}^{m-1} |E(FQ_n[L_{k2^{n_1}}])|$.

Proof. The proof contains three parts namely (1) embedding Algorithm (2) proof of correctness and (3) computation of wirelength.

(1) Embedding algorithm: Since $V(FQ_n) = V(Q_n)$ and $V(BT_{n_1} \times P_m) = V(ST_{n_1} \times P_m)$, we call the embedding algorithm as in Theorem 3.

(2) Proof of correctness: Let $ST^1_{n_1}, ST^2_{n_1}, \ldots, ST^m_{n_1}$ be the m vertex disjoint copies of sibling tree ST_{n_1} in $ST_{n_1} \times P_m$. For $j = 1, 2, \ldots, n_1$ and $i = 0, 1, \ldots, 2^{n_1-j}$, let S^j_i be the set of edges of $ST_{n_1} \times P_m$ induced by the $\lceil i/2 \rceil$th parent vertex from left to right in level $n_1 - j$ with its left child if i is odd and its right child if i is even of $ST^k_{n_1}, 1 \leq k \leq m$, together with the corresponding sibling edge which is the same edge in either case. Removal of S^j_i leaves $ST_{n_1} \times P_m$ into two components X^j_i and $\overline{X}^j_i$ where $V(X^j_i)$ is TX^j_i. Let G^j_i and $\overline{G}^j_i$ be the inverse images of X^j_i and $\overline{X}^j_i$ under f respectively. By Lemma 3, $V(G^j_i)$ is an optimal set in Q_n and FQ_n.

For $j = 1, 2, \ldots, n_1 - 1$ and $i = 0, 1, \ldots, 2^{n_1-j-1} - 1$, let SS^j_i be the set of edges of $ST_{n_1} \times P_m$ induced by the ith parent vertex from left to right in level $n_1 - j$ and its two children of $ST^k_{n_1}$, $1 \leq k \leq m$. Removal of SS^j_i leaves $ST_{n_1} \times P_m$ into two components Y^j_i and $\overline{Y}^j_i$ where $V(Y^j_i)$ is STX^j_i. Let H^j_i and $\overline{H}^j_i$ be the inverse images of Y^j_i and $\overline{Y}^j_i$ under f respectively. By Lemma 6, $V(H^j_i)$ is an optimal set in Q_n and FQ_n. Let $SS^{n_1}_0 = S^{n_1}_0$. For $k = 1, 2, \ldots, m - 1$, let $EE_k = E_k$ be the set of edges of $ST_{n_1} \times P_m$ such that each edge has one vertex in $BT^k_{n_1}$ and the other vertex in $BT^{k+1}_{n_1}$. Removal of E_k leaves $ST_{n_1} \times P_m$ into two components Z_k and $\overline{Z}_k$ where $V(Z_k)$ is $L_{k2^{n_1}}$. Let I_k and $\overline{I}_k$ be the inverse images of Z_k and $\overline{Z}_k$ under f respectively. By Theorems 1 and 2, $V(I_k)$ is an optimal set in Q_n and FQ_n.

We note that the sets $\{S^j_i : j = 1, 2, \ldots, n_1, i = 0, 1, \ldots, 2^{n_1-j}\} \cup \{SS^j_i : j = 1, 2, \ldots, n_1 - 1, i = 0, 1, \ldots, 2^{n_1-j-1} - 1\} \cup \{SS^{n_1}_0\} \cup \{EE_k, E_k : k = 1, 2, \ldots, m - 1\}$ form a partition of $E^2(ST_{n_1} \times P_m)$. Moreover, each edge cut satisfies conditions (i)–(iii) of the 2-partition wirelength lemma. Hence $WL_{Lex}(Q_n, ST_{n_1} \times P_m) = WL(Q_n, ST_{n_1} \times P_m)$ and $WL_{Lex}(FQ_n, ST_{n_1} \times P_m) = WL(FQ_n, ST_{n_1} \times P_m)$.

(3) Computation of wirelength: We divide this part into two cases.

Case A (Hypercube): For $j = 1, 2, \ldots, n_1,\ i = 0, 1, \ldots, 2^{n_1-j} - 1,\ EC_{Lex}(S^j_i) = n \cdot 2^{n-n_1}(2^j - 1) - 2 \cdot 2^{n-n_1-1}\{2^j(n - n_1 + j) - (n - n_1 + 2j)\} = 2^{n-n_1}\{2^j(n_1 - j) + 2j - n_1\}$. For $j = 1, 2, \ldots, n_1 - 1,\ i = 0, 1, \ldots, 2^{n_1-j-1} - 1,\ EC_{Lex}(SS^j_i) = n \cdot 2^{n-n_1+1}(2^j - 1) - 2 \cdot 2^{n-n_1}\{2^j(n - n_1 + j + 1) - (n - n_1 + 2j + 1)\} = 2^{n-n_1+1}\{2^j(n_1 - j - 1) + (2j - n_1 + 1)\}$ and $EC_{Lex}(SS^{n_1}_0) = n_1 2^{n-n_1}$. For $k = 1, 2, \ldots, m - 1,\ EC_{Lex}(E_k) = EC_{Lex}(EE_k) = n \cdot k2^{n_1} - 2\,|E(Q_n[L_{k2^{n_1}}])|$.

Therefore $WL(Q_n, ST_{n_1} \times P_m) = \frac{1}{2}\{\sum_{j=1}^{n_1} \sum_{i=0}^{2^{n_1-j}-1} EC_{Lex}(S^j_i) + \sum_{j=1}^{n_1-1} \sum_{i=0}^{2^{n_1-j-1}-1} EC_{Lex}(SS^j_i) + EC_{Lex}\ (SS^{n_1}_0) + \sum_{k=1}^{m-1}[EC_{Lex}(E_k) + EC_{Lex}(EE_k)]\} = 2^{n-1}\{n_1^2 - 4n_1 + 10 + n(2^{n-n_1} - 1)\} - 2^{n-n_1}(n_1 + 5) - 2\sum_{k=1}^{m-1} |E(Q_n[L_{k2^{n_1}}])|$.

Case B (Folded hypercube): For $j = 1, 2, \ldots, n_1 - 1,\ i = 0, 1, \ldots, 2^{n_1-j} - 1,\ EC_{Lex}(S^j_i) = (n + 1) \cdot 2^{n-n_1}(2^j - 1) - 2 \cdot 2^{n-n_1-1}\{2^j(n - n_1 + j) - (n - n_1 + 2j)\} = 2^{n-n_1}\{2^j(n_1 - j + 1) + 2j - n_1 - 1\}$ and for $j = n_1,\ i = 0,\ EC_{Lex}(S^j_i) = (n_1 + 1)2^{n-n_1}$. For $j = 1, 2, \ldots, n_1 - 2,\ i = 0, 1, \ldots, 2^{n_1-j-1} - 1,\ EC_{Lex}(SS^j_i) = (n + 1) \cdot 2^{n-n_1+1}(2^j - 1) - 2 \cdot 2^{n-n_1}\{2^j(n - n_1 + j + 1) - (n - n_1 + 2j + 1)\}$

$= 2^{n-n_1+1}\{2^j(n_1-j)+2j-n_1\}$, $EC_{Lex}(SS_0^{n_1-1}) = n_1 2^{n-n_1+1}$ and $EC_{Lex}(SS_0^{n_1}) = (n_1+1)2^{n-n_1}$. For $k = 1,2,\ldots,m-1$, $EC_{Lex}(E_k) = EC_{Lex}(EE_k) = n \cdot k2^{n_1} - 2\,|E(Q_n[L_{k2^{n_1}}])|$.

Hence $WL(FQ_n, BT_{n_1} \times P_m) = \frac{1}{2}\{\sum_{j=1}^{n_1-1}\sum_{i=0}^{2^{n_1-j}-1} EC_{Lex}(S_i^j) + EC_{Lex}(S_0^{n_1}) + \sum_{j=1}^{n_1-2}\sum_{i=0}^{2^{n_1-j-1}-1} EC_{Lex}(SS_i^j) + EC_{Lex}(SS_0^{n_1-1}) + EC_{Lex}(SS_0^{n_1}) + \sum_{k=1}^{m-1}[EC_{Lex}(E_k) + EC_{Lex}(EE_k)]\} = 2^{n-1}\{n_1^2 - 2n_1 + 5 + (n+1)(2^{n-n_1}-1)\} - n_1 2^{n-n_1} - 2\sum_{k=1}^{m-1}|E(FQ_n[L_{k2^{n_1}}])|$. □

5. Conclusion

In this paper we have computed the minimum wirelength of hypercubes and folded hypercubes into Cartesian product of 1-rooted complete binary tree and path, sibling tree and path. The minimum wirelength of hypercubes and folded hypercubes into Cartesian product of paths have been computed in [6,28]. We would like to take up the computation of minimum wirelength of hypercubes and folded hypercubes into Cartesian product of cycles as future research.

References

[1] S.A. Choudum, U. Nahdini, Complete binary trees in folded and enhanced cubes, Networks 43 (4) (2004) 266–272.
[2] Y.L. Lai, K. Williams, A survey of solved problems and applications on bandwidth, edgesum, and profile of graphs, J. Graph Theory 31 (1999) 75–94.
[3] J. Opatrny, D. Sotteau, Embeddings of complete binary trees into grids and extended grids with total vertex-congestion 1, Discrete Appl. Math. 98 (2000) 237–254.
[4] A.L. Rosenberg, Graph embeddings 1988: recent breakthroughs, new directions, in: VLSI Algorithms and Architectures (Corfu, 1988), in: Lecture Notes in Computer Science, vol. 319, Springer, New York, 1988, pp. 160–169.
[5] J.-M. Xu, Topological Structure and Analysis of Interconnection Networks, Kluwer Academic Publishers, 2001.
[6] P. Manuel, I. Rajasingh, B. Rajan, H. Mercy, Exact wirelength of hypercube on a grid, Discrete Appl. Math. 157 (7) (2009) 1486–1495.
[7] C.-H. Tsai, Embedding of meshes in Möbius cubes, Theoret. Comput. Sci. 401 (2008) 181–190.
[8] L.H. Harper, Global Methods for Combinatorial Isoperimetric Problems, Cambridge University Press, 2004.
[9] S.L. Bezrukov, Embedding complete trees into the hypercube, Discrete Appl. Math. 110 (2001) 101–119.
[10] S.N. Bhatt, F.R.K. Chung, F.T. Leighton, A.L. Rosenberg, Efficient embeddings of trees in hypercubes, SIAM J. Comput. 21 (1) (1992) 151–162.
[11] S.A. Choudum, I. Raman, On embedding subclasses of height-balanced trees in hypercubes, J. Appl. Math. 179 (2009) 1333–1347.
[12] W.-K.K. Chen, M.F.M. Stallmann, On embedding binary trees into hypercubes, J. Parallel Distrib. Comput. 24 (1995) 132–138.
[13] T. Dvořák, Dense sets and embedding binary trees into hypercubes, Discrete Appl. Math. 155 (2007) 506–514.
[14] I. Havel, On certain trees in hypercubes, in: Topics in Combinatorics and Graph Theory, Physica-Verlag, Heidelberg, 1990, pp. 353–358.
[15] V. Heun, E.W. Mayr, Optimal dynamic embeddings of complete binary trees into hypercubes, J. Parallel Distrib. Comput. 61 (2001) 1110–1125.
[16] B. Monien, H. Sudborough, Simulating binary trees on hypercubes, in: VLSI, Algorithms and Architectures, Proceedings of the 3rd Aegean Workshop on Computing, in: Lecture Notes in Computer Science, vol. 319, 1988, pp. 170–180.
[17] S.-C. Wang, Y.-R. Leu, S.-Y. Kuo, Distributed fault-tolerant embedding of several topologies in hypercubes, J. Inf. Sci. Eng. 20 (2004) 707–732.
[18] J. Wu, E.B. Fernandez, Y. Luo, Embedding of binomial trees in hypercubes with link faults, J. Parallel Distrib. Comput. 16 (1998) 59–74.
[19] S. Bezrukov, B. Monien, W. Unger, G. Wechsung, Embedding ladders and caterpillars into the hypercube, Discrete Appl. Math. 83 (1998) 21–29.
[20] R. Caha, V. Koubek, Optimal embeddings of generalized ladders into hypercubes, Discrete Math. 233 (2001) 65–83.
[21] J.D. Chavez, R. Trapp, The cyclic cutwidth of trees, Discrete Appl. Math. 87 (1998) 25–32.
[22] C.-J. Guu, The circular wirelength problem for hypercubes (Ph.D. dissertation), University of California, Riverside, 1997.
[23] S.L. Bezrukov, J.D. Chavez, L.H. Harper, M. Röttger, U.P. Schroeder, The congestion of n-cube layout on a rectangular grid, Discrete Math. 213 (2000) 13–19.
[24] I. Rajasingh, M. Arockiaraj, B. Rajan, P. Manuel, Minimum wirelength of hypercubes into n-dimensional grid networks, Inform. Process. Lett. 112 (2012) 583–586.
[25] P. Manuel, M. Arockiaraj, I. Rajasingh, B. Rajan, Embedding hypercubes into cylinders, snakes and caterpillars for minimizing wirelength, Discrete Appl. Math. 159 (2011) 2109–2116.
[26] I. Rajasingh, P. Manuel, B. Rajan, M. Arockiaraj, Wirelength of hypercubes into certain trees, Discrete Appl. Math. 160 (2012) 2778–2786.

[27] I. Rajasingh, B. Rajan, R.S. Rajan, On embedding of m-sequential k-ary trees into hypercubes, Appl. Math. 1 (6) (2010) 499–503.
[28] I. Rajasingh, M. Arockiaraj, Linear wirelength of folded hypercubes, Math. Comput. Sci. 5 (2011) 101–111.
[29] H. Katseff, Incomplete hypercubes, IEEE Trans. Comput. 37 (1988) 604–608.
[30] Q. Zhu, J.-M. Xu, X. Hou, M. Xu, On reliability of the folded hypercubes, Inform. Sci. 177 (2007) 1782–1788.
[31] L.H. Harper, Optimal assignment of numbers to vertices, J. Soc. Ind. Appl. Math. 12 (1964) 131–135.
[32] S.L. Bezrukov, S.K. Das, R. Elsässer, An edge-isoperimetric problem for powers of the Petersen graph, Ann. Comb. 4 (2000) 153–169.
[33] M.R. Garey, D.S. Johnson, Computers and Intractability, A Guide to the Theory of NP-Completeness, Freeman, San Francisco, 1979.
[34] A.J. Boals, A.K. Gupta, N.A. Sherwani, Incomplete hypercubes: Algorithms and embeddings, J. Supercomput. 8 (1994) 263–294.
[35] H.-L. Chen, N.-F. Tzeng, A boolean expression-based approach for maximum incomplete subcube identification in faulty hypercubes, IEEE Trans. Parallel Distrib. Syst. 8 (1997) 1171–1183.
[36] S.L. Bezrukov, Edge isoperimetric problems on graphs, in: L. Lovasz, A. Gyarfas, G.O.H. Katona, A. Recski, L. Szekely (Eds.), Graph Theory and Combinatorial Biology, Bolyai Soc. Math. Stud., Vol. 7, Budapest, 1999, pp. 157–197.
[37] M. Arockiaraj, P. Manuel, I. Rajasingh, B. Rajan, Wirelength of 1-fault Hamiltonian graphs into wheels and fans, Inform. Process. Lett. 111 (18) (2011) 921–925.
[38] I. Rajasingh, P. Manuel, M. Arockiaraj, B. Rajan, Embedding of circulant networks, J. Comb. Optim. 26 (1) (2013) 135–151.
[39] S.L. Bezrukov, J.D. Chavez, L.H. Harper, M. Röttger, U.P. Schroeder, Embedding of hypercubes into grids, in: MFCS, 1998, pp. 693–701.
[40] T.H. Cormen, C.E. Leiserson, R.L. Rivest, C. Stein, Introduction to Algorithms, MIT Press and McGraw-Hill, New York, 2001.
[41] I. Rajasingh, J. Quadras, P. Manuel, A. William, Embedding of cycles and wheels into arbitrary trees, Networks 44 (2004) 173–178.