

VLSI layout of Benes networks

Paul Manuel *

*Department of Information Science
Kuwait University
Kuwait*

Kalim Qureshi

*Department of Mathematics and Computer Science
Kuwait University
Kuwait*

Albert William

Albert Muthumalai

*Department of Mathematics
Loyola College
Chennai 600034
India*

Abstract

The Benes network consists of back-to-back butterflies. There exist a number of topological representations that are used to describe butterfly – like architectures. We identify a new topological representation of Benes network. The significance of this representation is demonstrated by solving two problems, one related to VLSI layout and the other related to robotics. An important VLSI layout network problem is to produce the smallest possible grid area for realizing a given network. We propose an elegant VLSI layout of r -dimensional Benes networks using this representation. The area of this layout is $O(2^{2r})$ whereas the lower bound for the area of the VLSI layout of Benes networks is $O(2^{2r})$. This lower bound is estimated by applying Thompson result.

Keywords and phrases : Butterfly network, Benes network, VLSI layout problem.

*E-mails: pauldmanuel@gmail.com, p.manuel@cfw.kuniv.edu

Journal of Discrete Mathematical Sciences & Cryptography

Vol. 10 (2007), No. 4, pp. 461–472

© Taru Publications

1. Introduction

A multistage network consists of a series of switch stages and interconnection patterns, which allows N inputs to be connected to N outputs. A multistage network uses dynamic interconnection to allow communication paths to be established as needed for the transfer of information between I/O nodes. In massively parallel computing, interconnection networks remain to be one of the most critical components. *Multistage Interconnection Networks* (MINs) have long been used as the communication network for parallel computers. The main advantages of MINs are their high bandwidth $O(N)$, low diameter $O(\log N)$, and constant degree switches. Multistage networks have been used in commercial machines, such as the BBN, CM 5 and Meiko [8]. The butterfly and Benes networks are important multistage interconnection networks, which possess attractive topologies for communication networks [10]. They have been used in parallel computing systems such as IBM $\frac{SP1}{SP2}$, MIT Transit Project, and NEC Cenju-3, and used as well in the internal structures of optical couplers, e.g., star couplers [10, 13].

We represent networks as undirected graphs whose nodes represent processors and whose edges represent inter-processor communication links. An *embedding* of undirected graph, G , in another, H , comprises a one-to-one *assignment* of the nodes in G to nodes in H , plus a *routing* of each edge of G within H that is, an assignment of a path in H connecting the images of the endpoints of each edge in G . The set of nodes V of an r -dimensional butterfly correspond to pairs $[w, i]$, where i is the dimension or level of a node ($0 \leq i \leq r$) and w is an r -bit binary number that denotes the row of the node. Two nodes $\langle w, i \rangle$ and $\langle w', i' \rangle$ are linked by an edge if and only if $i' = i + 1$ and either:

1. w and w' are identical, or
2. w and w' differ in precisely the i th bit.

The edges in the network are undirected. An r -dimensional butterfly is denoted by $BF(r)$.

An r -dimensional Benes network has $2r + 1$ levels, each level with 2^r nodes. The level zero to level r vertices in the network form an r -dimensional butterfly. The middle level of the Benes network is shared by these butterflies [9]. As butterfly is known for FFT, Benes is known

for permutation routing (*rearrangeable* network). An r -dimensional Benes is denoted by $B(r)$. Figure 1 shows a $B(4)$ network. All the results in this paper are discussed only for Benes since the results corresponding to butterfly can be derived as a particular case.

One of the important parameters used to evaluate parallel architectures is symmetry. Other considerations are: bisection width, wire length (layout aspects), existence of optimal communication techniques such as routing and broadcasting, node disjoint paths (fault tolerance), and recursive decomposition (or scalability). Symmetry is helpful for solving issues related to VLSI design. A VLSI layout for a network is characterized as an embedding within a two-dimensional grid, which assigns nodes of a graph G to points in the grid together with an (incidence preserving) assignment of the edges of G to paths in the grid. The paths of the layout are restricted to follow along grid tracks and are not allowed to overlap for any distance (although a vertical path segment may cross a horizontal path segment). If they change direction at this point, it is called a *knock-knee* [5]. In addition, the paths may not cross nodes to which they are not adjacent [2].

1.1 Mathematical definition of a VLSI layout

Following [1], a VLSI layout $\mathbb{L}(\alpha, \rho)$ of an N -node graph $G(V, E)$ in an $m \times n$ grid M , where $N \leq mn$, is an embedding (α, ρ) of G into $M[m, n]$ where α is a one – one mapping from $V(G)$ into $V(M)$ and $\rho = \left\{ \frac{P(u,v)}{(u,v)} \right\}$ is an edge of $E(G)$ and $P(u, v)$ is a path of M joining $\alpha(u)$ and $\alpha(v)$. The routing paths of ρ collectively satisfy the following conditions:

- Distinct routing paths in the grid are edge-disjoint, so that the embedding that embodies a layout has unit congestion in the grid.
- A routing path $P(u, v)$ traverses over no image node $\alpha(w)$ where w is in $V(G)$ and $w \neq u, v$.

A *VLSI layout problem* of a graph G is to produce an area-efficient layout for G . It is shown that a VLSI layout problem of a forest of trees is NP-complete [2]. The butterfly graphs have different representations including Omega network, the flip network, the baseline, and the reverse baseline networks. Each representation exhibits different characteristics [1, 5, 9, 13]. In other words, the butterfly network is drawn in different ways to exhibit different properties. In this paper, we introduce a new

representation of Benes network, which helps solving the VLSI layout problem in Benes networks. We focus on laying out the Benes network on a square grid. Our aim is to produce an efficient VLSI layout of Benes without knock-knees. Our VLSI layout of $B(r)$ is a square area. The area of this layout is $(2^{(r+2)} - 1)(2^{(r+2)} - 1)$ which is $O(2^{2r})$. This is satisfactory since the lower bound of the area of a VLSI layout of $B(r)$ is $O(2^{2r})$.

2. Proposed VLSI representation of Benes network

In this section, we discuss a representation of Benes network, which is suitable for VLSI layout. The proposed representation is shown in Figure 2. To avoid confusion between the two representations of Figure 1 and Figure 2, the representation in Figure 1 will be called *Rearrangeable representation of Benes* and the representation in Figure 2 will be called *VLSI representation of Benes*. A similar representation for butterfly is studied by Dinitz *et al.* [5, 15].

The proposed VLSI representation of Benes network is constructed recursively as follows: Two $(r - 1)$ -dimensional Benes networks $B(r - 1)$ form mirror images with respect to an array of level 0 and level $2r$ nodes. Each 4-cycle is drawn as a diamond. Particularly, the level 0 and level $2r$ nodes are the vertices belonging to chordless 4-cycles in the diamond formation bridging the two $(r - 1)$ -dimensional Benes networks $B(r - 1)$. See Figure 2. This representation provides a structural visualization, an in-depth understanding about the cyclic properties and the organization of spanning trees of Benes networks. The description of cyclic structures is an important problem in graph theory. The following observations of Benes network, which are similar to butterfly, are straightforward from the VLSI representation given in Figure 2.

Lemma 2.1 [10].

1. *A Benes network is bipartite.*
2. *No two 4-cycles of $B(r)$ have a common edge.*
3. *The edge set E of $B(r)$ is disjoint union of 4-cycles, that is, there are $2r \times 2^{(r-2)}$ number of 4-cycles.*

Even though the Benes network consists of back-to-back butterflies, there is a subtle structural difference between Benes and butterfly. See Figure 7. The removal of level 0 nodes of $BF(r)$ leaves two disjoint copies

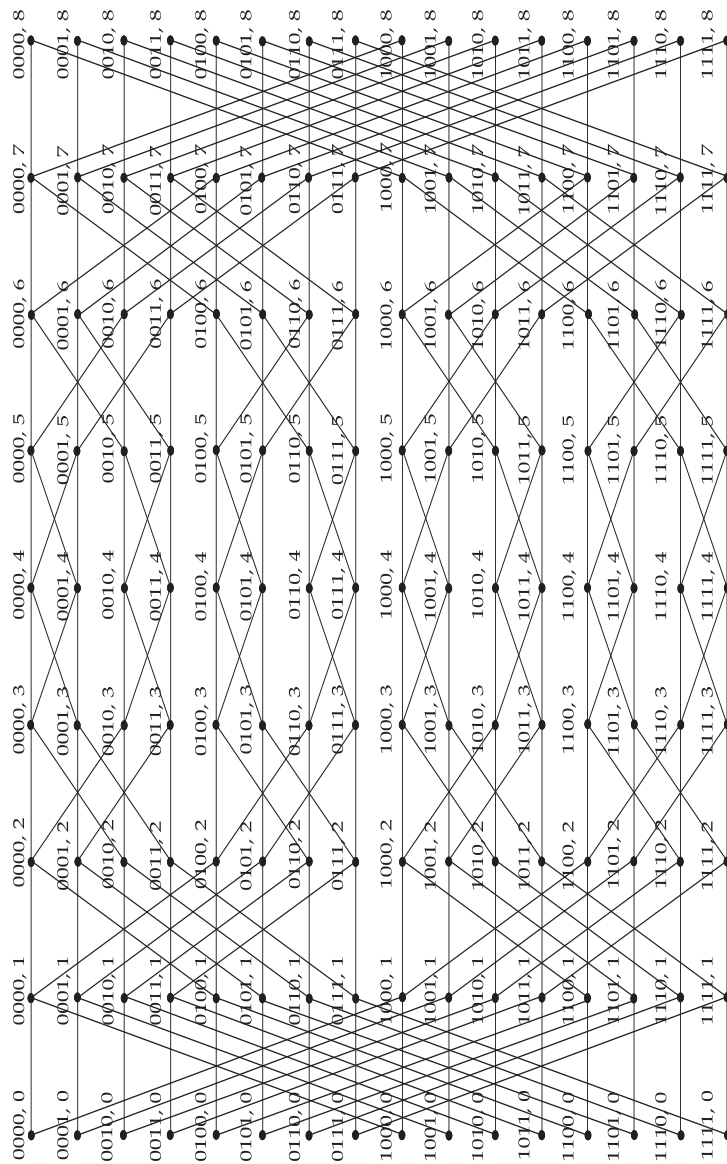


Table 1
Rearrangeable representation of Benes $B(4)$

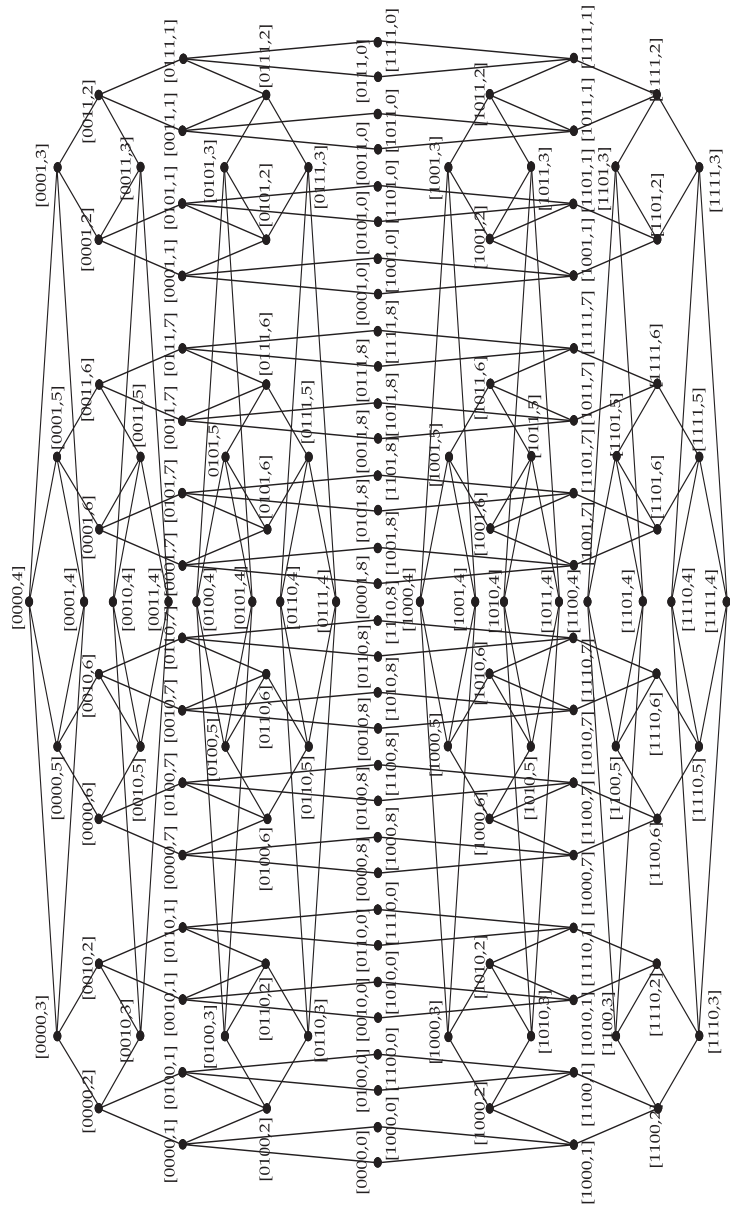


Table 2
VLSI representation of Benes $B(4)$

of $BF(r-1)$. In the same way, the removal of level r nodes of $BF(r)$ leaves two disjoint copies of $BF(r-1)$. This recursive structure can be viewed in another way. The removal of level 0 nodes and level r nodes (removal of all nodes of degree 2) of $BF(r)$ leaves 4 disjoint copies of a $BF(r-2)$. However the removal of level 0 nodes and level $2r$ nodes (removal of all nodes of degree 2) of $B(r)$ leaves 2 disjoint copies of a $B(r-1)$. In other words, the butterfly has dual symmetry, which the Benes does not have.

Lemma 2.2. *The Rearrangeable and VLSI representations of Benes are isomorphic.*

Proof. We apply induction on the dimension of Benes $B(r)$. Both 1-dimensional Rearrangeable representation and 1-dimensional VLSI representation are cycles of length 4. Let us assume that $(k-1)$ -dimensional Rearrangeable representation and $(k-1)$ -dimensional VLSI representation of Benes are isomorphic.

Now let us show that the k -dimensional Rearrangeable representation and the k -dimensional VLSI representation of Benes are isomorphic. Remove nodes of degree 2 (level 0 nodes and level $2k$ nodes) from both k -dimensional Rearrangeable representation and k -dimensional VLSI representation. By induction hypothesis, the resultant Benes networks are isomorphic. The nodes of degree 2 (level 0 nodes and level $2k$ nodes) of both k -dimensional Rearrangeable representation and k -dimensional VLSI representation are organized in the same way as follows: The level 0 nodes $[0u_2 \dots u_k, 0]$ and $[1u_2 \dots u_k, 0]$ are adjacent to level 1 nodes $[0u_2 \dots u_k, 1]$ and $[1u_2 \dots u_k, 1]$, respectively. In the same way, level $2k$ nodes $[0u_2 \dots u_k, 2k]$ and $[1u_2 \dots u_k, 2k]$ are adjacent to level $(2k-1)$ nodes $[0u_2 \dots u_k, 2k-1]$ and $[1u_2 \dots u_k, 2k-1]$, respectively. Moreover, these nodes form a chordless cycle of length 4. These 4-cycles are edge disjoint with the rest of the graph. Hence both Rearrangeable representation and VLSI representation are isomorphic. \square

3. A simple VLSI layout of Benes network

Avior *et al.* [1] have estimated that the r -dimensional butterfly network can be laid out in area $(1+o(1))2^{2r}$ while no layout of the network can have area smaller than $(1-o(1))2^{2r}$. Dinitz *et al.* [5, 15] have presented a layout whose encompassing rectangle is of area $\left(\frac{1}{2}\right)2^{2r} + o(2^{2r})$, but this

rectangle is 45° slanted with respect to the grid axes. There are different models of VLSI layouts for butterfly-like architectures [14, 15, 16]. Even though Benes network consists of back-back butterflies, these models of VLSI layouts of butterfly are not extendable to Benes. In this paper we provide a simple square VLSI layout without knock-knees for Benes which is of course applicable to butterfly networks too.

In the VLSI representation of Benes network, each 4-cycle is represented as a diamond. The 4-cycles with level r nodes of $B(r)$ are in pairs. We call them “*Nested diamonds*”. Other 4-cycles are called “*Normal diamonds*”. In Figure 2, the six vertices $[0000, 3]$, $[0000, 4]$, $[0000, 5]$, $[0001, 3]$, $[0001, 4]$ and $[0001, 5]$ form a pair of nested diamonds. Notice that a pair of nested diamonds contains 6 nodes inducing two 4-cycles. Similarly, the four vertices $[0000, 2]$, $[0000, 3]$, $[0010, 2]$ and $[0010, 3]$ form a normal diamond.

VLSI layout algorithm and its proof of correctness

Our construction of VLSI layout of Benes is done in 3 steps:

- Step 1:* Draw the VLSI representation of Benes as in Figure 3.
- Step 2:* Each normal diamond is stretched to a rectangle as in Figure 4.
- Step 3:* Each pair of Nested diamonds is stretched along the grid lines as in Figure 5.

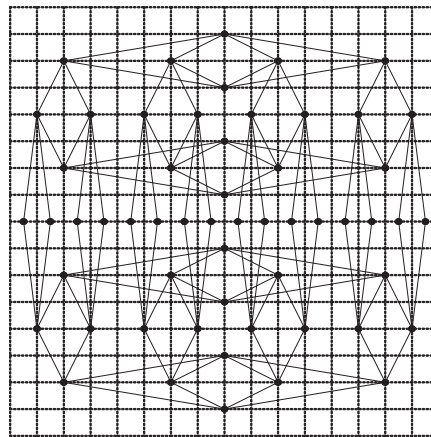


Figure 3

Step 1. Drawing Benes $B(3)$ on a grid

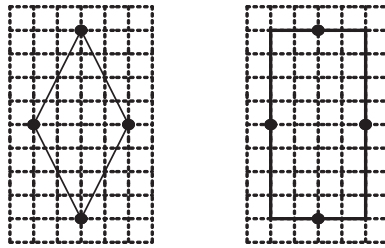


Figure 4

Step 2. A normal diamond is stretched to a rectangle of the grid

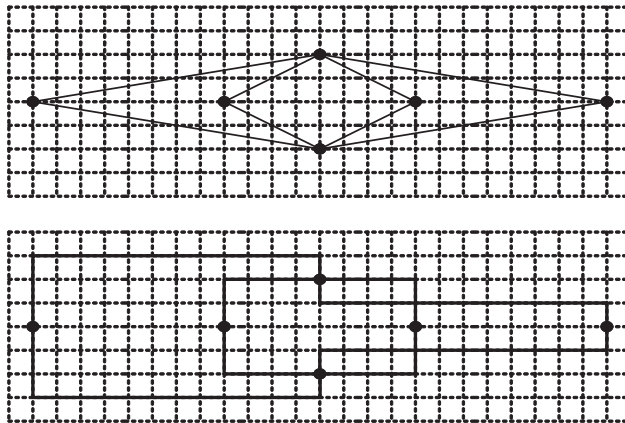


Figure 5

Step 3. A pair of nested diamonds is stretched on grid axis

The resultant layout is given in Figure 6. Now we claim that this grid embedding is indeed a VLSI layout. The proof of correctness is straightforward using induction hypothesis. Suppose it is true for $(k - 1)$ -dimensional Benes. For a k -dimensional Benes, it is enough to consider the 4-cycles (normal diamonds) at level 0 and level $2k$ nodes. By the very structure of the VLSI representation of Benes, it is easy to see that the 4-cycles (normal diamonds) at level 0 and level $2k$ nodes can be drawn as a rectangle without violating VLSI requirements.

Area of this layout and lower bound for the area of the VLSI layout of $B(r)$

Let us now estimate the area of this VLSI layout of $B(r)$. All the level 0 and level $2r$ nodes are placed in one horizontal grid axis with exactly one vertical grid axis between any two of these nodes. In the same way, all

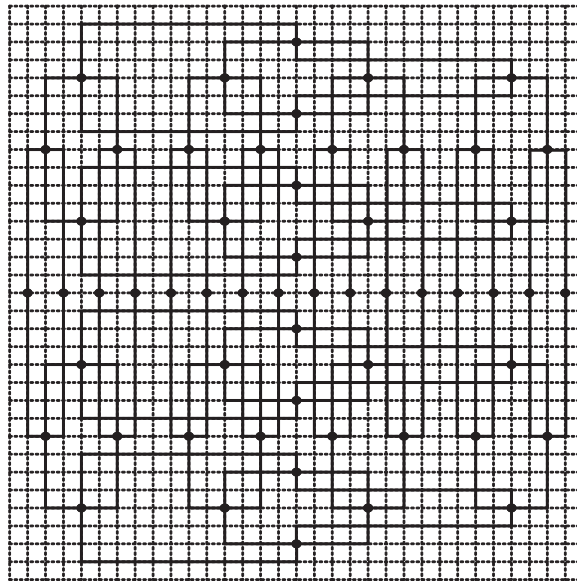


Figure 6
A VLSI layout of Benes $B(3)$

the level r nodes are placed in a vertical grid axis with exactly 3 horizontal grid axes between any two of these nodes. This observation is straightforward and it can be easily proved by induction. Thus, the area of this layout of $B(r)$ is $(2^{(r+2)} - 1)(2^{(r+2)} - 1)$ which is $O(2^{2r})$. Thompson [12] showed that, up to a constant factor, the layout area can be no less than the square of the bisection width. Since the bisection width of r -dimensional Benes network is $O(2^r)$ [8], the lower bound for the area of VLSI layout of Benes networks is $O(2^{2r})$.

4. Conclusion

Even though this paper focuses on Benes networks, all the results are applicable to butterfly too. We provide a simple VLSI layout without knock-knees for Benes network. This VLSI layout of $B(r)$ is laid in a square area. The area matches with the lower bound up to a constant factor.

Though wrapped butterfly is a butterfly-like architecture, it is not straightforward to extend these results to wrapped butterfly. The NP-complete problems such as achromatic number problem and minimum crossing number problem [2, 6] are open for Benes and butterfly networks.

It is interesting to see whether these problems can be solved using this representation.

References

- [1] A. Avior, T. Calamoneri, S. Even, A. Litman and A. L. Rosenberg, A tight layout of the butterfly network, *Theory of Computing Systems*, Vol. 31 (4) (1998), pp. 474–488.
- [2] S. N. Bhatt and F. T. Leighton, A framework for solving VLSI graph layout problems, *MIT/LCS/TR*, Vol. 305 (44) (1983).
- [3] H. L. Bodlaender, Achromatic number is NP-complete for co-graphs and interval graphs, *Information Processing Letters*, Vol. 32 (3) (1989), pp. 135–138.
- [4] C. Bornstein, A. Litman, B. M. Maggs, R. K. Sitaraman and T. Yatzkary, On the bisection width and expansion of butterfly networks, *Theory of Computing Systems*, Vol. 34 (6) (2001), pp. 491–518.
- [5] Y. Dinitz, S. Even, R. Kupershtok and M. Zapolotsky, Some compact layouts of the butterfly, in *Proceedings of the 11th Annual ACM Symposium on Parallel Algorithms and Architectures*, Saint Malo, France (June 27-30, 1999), pp. 54–63.
- [6] M. R. Garey and D. S. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness*, W. H. Freeman and Company, 1979.
- [7] L. T. Q. Hung, M. M. Syslo, M. L. Weaver and D. B. West, Bandwidth and density for block graphs, *Discrete Mathematics*, Vol. 189 (1998), pp. 163–176.
- [8] S. Konstantinidou, The selective extra stage butterfly, *IEEE Transactions on Very Large Scale Integration Systems*, Vol. 1 (2) (June 1993).
- [9] T. F. Leighton, *Introduction to Parallel Algorithms and Architecture: Arrays, Trees, Hypercubes*, Morgan Kaufmann Publishers, 1992.
- [10] X. Liu and Q. P. Gu, Multicasts on WDM all-optical butterfly networks, *Journal of Information Science and Engineering*, Vol. 18 (2002), pp. 1049–1058.
- [11] D. Manlove and C. McDiarmid, The complexity of harmonious coloring for trees, *Discrete Applied Mathematics*, Vol. 57 (1995), pp. 133–144.
- [12] C. D. Thompson, Area-time complexity for VLSI, in *11th Annual ACM Symposium on Theory of Computing*, (1979).
- [13] J. Xu, *Topological Structure and Analysis of Interconnection Networks*, Kluwer Academic Publishers, 2001.

- [14] C. H. Yeh, B. Parhami, E. A. Varvarigos and H. Lee, VLSI layout and packaging of butterfly networks, in *Proceedings of the 12 Annual ACM Symposium on Parallel Algorithms and Architectures*, Bar Harbor, Maine, United States (July 09-13, 2000), pp. 196–205.
- [15] Y. Dinitz, S. Even and M. Zapolotsky, A compact layout of the butterfly, *Journal of Interconnection Networks*, Vol. 4 (1) (2003), pp. 53–75.
- [16] S. Muthukrishnan, M. S. Paterson, S. C. Sahinalp and T. Suel, Compact grid layouts of some multi-level networks, in *31st Symposium of Theory of Computing*, Georgia, Atlanta, (May 1999), pp. 455–463.

Received July, 2006