Appears in the 13th Euromicro Conference on Parallel, Distributed and Network-based Procesing. Switzerland, Feb. 2005

Practicable Layouts for Optimal Circulant Graphs

Enrique Vallejo, Ramón Beivide and Carmen Martínez
Computer Architecture Group. University of Cantabria, Spain
{enrique, mon, carmenmf}@atc.unican.es

Abstract

Circulant graphs have been deeply studied in technical literature. Midimew networks are a class of distance-related optimal circulant graphs of degree four which have applications in network engineering and coding theory. In this research, a new layout for Midimew networks which keeps the maximum link length under the value $\sqrt{5}$ is presented, considering unitary length as the subjacent mesh link's length. The most interesting Midimew sizes are studied: dense and quasi-dense cases, with bounded link length layouts in both cases, although the proposed algorithm is also valid for other network sizes. These results improve a previously known result in a maximum factor of $\sqrt{5}$.

Also, the number of planar layers necessary to implement this layout is studied. In addition, an expandability method is presented which enables network expansion without modifying existing nodes or links, except those needed to connect to new nodes. Finally, this layout is compared to the classical folded Torus. When considering both physical and topological distances and propagation delays our proposal is clearly competitive against Tori.

1. Introduction

In current computer systems, performance is being limited by interconnections. Therefore, the use of replicated hardware interconnected by a suitable network is emerging as a design solution for multiple digital systems. Currently, networks are used in the design of systems on chip, cluster microprocessors and on-chip multiprocessors. Input/output devices are nowadays being integrated in computer systems by using switched networks. Clusters of computers, multiprocessors and massively parallel processors rely on the effectiveness of their interconnection mechanisms. Finally, switch fabrics for local, metropolitan and wide area networks are beginning to be designed on the basis of direct interconnection networks.

In the last decades, hundreds of parallel computer systems have been designed. The most used network topologies have been Meshes, [1]Tori and Hypercubes. Nowadays, hypercube architectures have declined in importance but lower degree topologies such as Tori and Meshes still compete in the design of the latest parallel systems [6], [7]. Circulant graphs are also attractive topologies that have been extensively studied due to their good distance-related properties and their optimal connectivity. In fact, the Illiac IV, one of the most popular parallel computers of its time, used a degree four circulant graph of 64 nodes. Other circulant-based interconnection systems have been considered in [5], [2].

Midimew networks are a family of undirected degree four circulant graphs having optimal distance properties, that is, minimum diameter and average distance [1]. Midimew networks can be successfully applied to the design of parallel systems as they exhibit remarkable performance improvements compared to other degree four topologies such as Tori. Their richer connectivity together with this lower diameter make Midimew networks clearly superior to Tori for either individual or collective communications [8], [2]. Moreover, Midimew connectivity allows optimal embedding of multiple popular subgraphs in parallel computing such as Meshes, Rings of different sizes, Trees and Hamiltonian cycles [9]. All these gains are achieved by using layouts and wraparound connectivity patterns which are different to those employed by the Torus.

In this paper, we explore the design of optimal bi-dimensional layouts for Midimews. We provide a methodology for making the link length independent of the network size. Furthermore, we improve by a factor $\sqrt{5}$ the results presented in [3] which dealt with the same problem. We also introduce in this work a set of rules which permits graceful network expandability.

2. Related work and motivations

A degree four circulant graph with N vertices and jumps $\{a,b\}$ is an undirected graph in which each vertex $n \in \{0,...,N-1\}$ is adjacent to the four vertices $n \pm a, n \pm b$, (operations modulo N). A class of optimal degree four circulants denoted as *Midimew networks* was introduced as an alternative to Tori in [10]. Then, for any positive integer *k*, if $2k^2 \le N \le 2k^2 + 2k + 1$ a Midimew network with N nodes has jumps $\{k, k+1\}$ and if $2(k-1)^2 + 2(k-1) + 2 \le N \le 2k^2$ the jumps are $\{k-1, k\}$. In [1] it was proved that this degree four circulant graphs have minimum diameter *k* and also minimize the average distance. A minimal routing scheme was also proposed. Table 1 shows distance-related properties for both Torus and Midimew, and their asymptotic values for big networks. As can be seen, diameter is improved by a factor

[1] This work has been supported by the "Comisión Interministerial de Ciencia y Tecnología (CICYT)", proyect TIC-2001-591-C02-01.

of $\sqrt{2} = 1.41$ (41%), and average distance by $\frac{3}{2\sqrt{2}} = 1.06$.

Midimew networks, such as 2D-Tori, are mesh-like topologies with wrap-around links, whose lengths grow with the network size. Since internal links are supposed to have unitary length, most applications can be negatively affected by this unbalance. For example, wrap-around links in Tori grow as $\sqrt{N}$, N being the number of nodes. The Folded Torus as represented in Figure 1 is a solution to equalize the network links by increasing the wire length to 2. With the same aim, a new layout for rectangular midimews was proposed in [3] which results in equalized links of length 4 or 5 depending on the network size. We improve the results in [3] by means of a new layout in which the maximum wire length is bounded by $\sqrt{5}$.

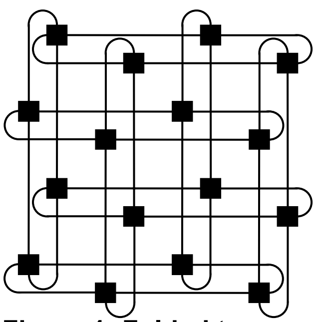

Figure 1: Folded torus

3. New Layouts for Midimew Networks

In this Section, we will first consider dense Midimew networks. They contain the maximum number of nodes for a given diameter *k,* that is, $N = 2k^2 + 2k + 1$. Dense Midimews are very attractive topologies that resemble the Twisted Torus proposed in [4]. Nevertheless, Midimew networks can be built for any number of nodes. At the end of the Section, we extend our methodology to non-dense Midimew networks containing $N = 2k^2 + 2k$ nodes. We concentrate on networks containing one node less than the dense case because they are quasi-dense. Nevertheless, other network sizes can also benefit from this procedure.

3.1. Bounded Link Layout for Dense Midimews

Our methodology starts from a base layout having $2k$ rows and $k+1$ columns plus a node in a separate row, 0 or any other by graph symmetry. Vertical links represent jumps in $k+1$ and diagonal links represent jumps in *k*. Figure 2 shows an example for $k = 3$ and $N = 25$. Every node in the network has an integer label, which identifies the node and determines its adjacencies. Apart from this label, we will refer to the position of a certain node by the number of row $\{1,...,2k+1\}$ and column $\{1,...,k+1\}$ where it is located. Next, we introduce some previous definitions.

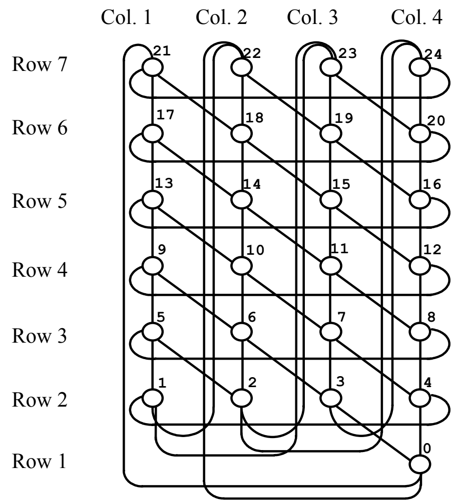

Figure 2: Dense midimew initial disposition

Definition 1: We will denote horizontal distance between two nodes as the number of columns that separate them in the mesh. In the same way, vertical distance is defined as the number of rows that separate two nodes. Using this distance concept we define the link length, which will be the physical distance between the two nodes connected by such a link, and can be divided into its horizontal and vertical components.

Definition 2: Given a certain row we define an m-order rotation (or simply, m-rotation) as a row transformation which takes m nodes on the far right and sets them on the left side without modifying their relative position. The procedure for rotating columns is identical. Obviously, in an n-node row, two rotations A and B will be equivalent if and only if $A \equiv B \bmod n$. We will say that a certain row "has m-rotation" when an m-rotation has been applied to it.

Definition 3: Given a row with nodes $\{1,2,...,n\}$, we define two shuffle transformations which map every node location onto a different one in the following way:

Table 1: Distance related properties for Torus and Midimew topologies

Topology	Nodes	Diameter	Approximate diameter	Average distance	Approximate Aver. distance
2D-Torus	$N = W^2$	$2\left\lfloor \frac{W}{2} \right\rfloor$	$\sqrt{N}$	$2\left\lfloor \frac{W}{2} \right\rfloor \left\lfloor \frac{W+1}{2} \right\rfloor \frac{W}{N-1}$	$\frac{\sqrt{N}}{2}$
Midimew	$N = 2k^2 + 2k + 1$	k	$\sqrt{\frac{N}{2}}$	$k\left[1 - \frac{2(k^2-1)}{3(N-1)}\right]$	$\frac{\sqrt{2N}}{3}$

- **Shuffle A:**

$$x' = 2x - 1 \quad if\ x \le \frac{(n+1)}{2}$$

$$x' = 2n - 2x + 2 \quad if\ x > \frac{(n+1)}{2}$$

- **Shuffle B:** (taken from [3])

$$x' = 2x \quad if\ x < \frac{(n+1)}{2}$$

$$x' = 2n - 2x + 1 \quad if\ x \ge \frac{(n+1)}{2}$$

Notice that any two consecutive nodes, after a row shuffle, have maximum horizontal distance 2. These shuffles can also be applied to columns.

In order to employ our methodology, a rotation followed by a shuffle must be applied to every row. The aim of rotations is to enable that the first and last rows get linked together by means of links with a limited length. The aim of shuffles is to eliminate peripheral links within each row. Now, we present and prove the algorithm for obtaining the new layout.

Algorithm 1: Bounded Layout for Dense Midimews

***Step 1 or Initial layout*:** Arrange the $2k^2 + 2k + 1$ nodes in $2k + 1$ rows, beginning with a row containing a single node (node 0, at the end of row 1) and setting the following nodes consecutively as in the initial layout of Figure 2.

Step 2 or Row rotation and shuffle:

- For rows $1 \le i \le k + 1$, apply a rotation $\left\lfloor \frac{i-1}{2} \right\rfloor$ and then apply an A shuffle to odd rows and a B shuffle to even ones.
- For rows $k + 2 \le i \le 2k + 1$, apply a rotation $\left\lfloor \frac{i}{2} \right\rfloor$ and then apply a B shuffle to odd rows and an A shuffle to even ones.

Step 3 or Column shuffle A: Shuffle all columns according to shuffle A.

Theorem 1: Algorithm 1 maps a dense Midimew of any size onto a bi-dimensional layout in which the size of any of its links is equal to or less than $\sqrt{5}$.

Proof: Changing the position of a node modifies vertical and horizontal lengths of the links attached to this node. We will classify links into four groups in order to see how previous transformations modify them:

- Group one: links that join nodes in one row with nodes in the row above, except for rows $k + 1$ and $k + 2$.
- Group two: links between the last row $2k + 1$ and rows 1 (node 0) or 2.
- Group three: links between rows $k + 1$ and $k + 2$.
- Group four: links joining extreme nodes in the same row.

We will study for each group its horizontal and vertical lengths.

<u>*Group one:*</u>

Horizontal length: According to the second step of the algorithm, there are two cases for a pair of consecutive rows:

I. The same rotation is applied to both rows. Shuffle A is applied to the lowest row and shuffle B to the highest one.

II. The lowest row has *n*-rotation and shuffle B, whereas the highest row has $(n+1)-$ rotation and shuffle A.

Case I:

If the same rotation is applied to both rows, then each node in the lower row is connected to its upper neighbors 0 and 1 positions to the left. Let's consider a node in the lower row in position *i* after rotation:

a) If $i = 1$, then the node in the lower row is connected to the first and the last ones of the upper row. Shuffle A in the lower row leaves node $i = 1$ in the first position, whereas shuffle B in the upper one takes the first and last nodes to the first positions. Thus, horizontal lengths are 0 and 1 respectively, as shown in Figure 5.

b) If $1 < i < \frac{k+1}{2}$, node *i* is connected to nodes $i - 1$ and *i* in the upper row. Shuffle A maps *i* onto $2i - 1$ in the lower row, while shuffle B maps nodes $i - 1$ and *i* into $2i - 2$ and $2i$ respectively. Horizontal lengths are 1, as shown in Figure 6.

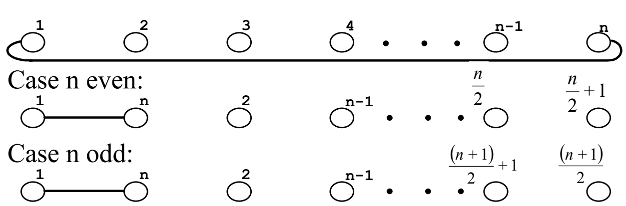

Figure 3: Shuffle A

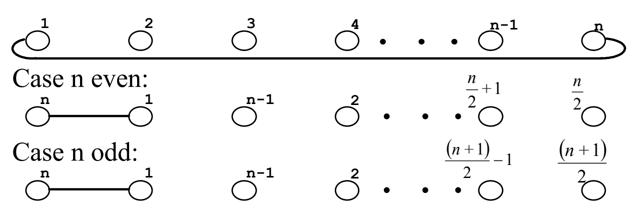

Figure 4: Shuffle B

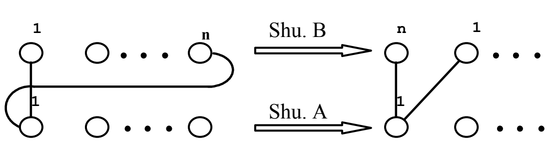

Figure 5: Group 1 case I, a)

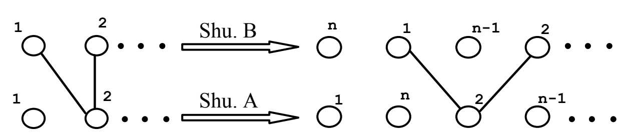

Figure 6: Group 1 case I, b)

c) If k is even and $i = \frac{k}{2} + 1$, the lower node is mapped onto the last position in the row, while the upper ones are mapped onto the last two positions. Thus, horizontal lengths are 0 and 1 respectively, as shown in Figure 7. The same happens if k is odd and $i = \frac{k+1}{2}$.

d) If $i > \frac{k+1}{2}$, shuffle A maps i onto $2k - 2i + 4$, while shuffle B in the upper one takes nodes $i-1$ and i onto $2k - 2i + 5$ and $2k - 2i + 3$, resulting in the horizontal distance 1, as shown in Figure 8.

Figure 9 illustrates the case for $k + 1 = n = 5$, with 0-rotation, representing the complete Case I.

The proof for **Case II** (n- and $(n+1)$-rotation and shuffles B and A) is analogous to the previous one, but considering that every node in the lower row is connected, after rotations, at most, to its upper neighbours 0 and 1 positions to the right. The result is also maximum horizontal lengths 1.

Vertical length: Shuffle A is applied to all columns, which makes maximum vertical distance between consecutive rows become 2.

In conclusion, all links in Group 1 get vertical length 2 and horizontal 0 or 1. Applying the Pythagorean Theorem each link turns out to have maximum length $\sqrt{5}$.

Group two:

Horizontal length: In the initial layout shown in Figure 2, nodes in the highest row ($2k+1$) are connected to those nodes in the lower rows 1 and 2 positions to the left. Row $2k+1$ has k-rotation, $k \equiv -1(\text{mod}\, k+1)$, while rows 1 and 2 have 0-rotation.

The first two nodes in the upper row are connected to node 0, the only node in the lower row. After $(-1)-$rotation and shuffle B, these two nodes in the upper row maintain the same position. After 0-rotation and shuffle A in the first row, node 0 goes to the second position in the row. Then, horizontal lengths for these two links are 0 and 1.

Links between rows $2k+1$ and 2 have, originally, horizontal lengths 1 or 2. As the highest row has (-1)-rotation and row 2 has 0-rotation, these distances become 0 and 1. The same shuffle B is applied to both rows, which makes the maximum horizontal distance between connected nodes become 2 (from the two rows), as they are in the same or consecutive positions.

Vertical length: The aim of column shuffle A is to leave row $2k+1$ between 1 and 2. Then, vertical length is 1.

Then, maximum horizontal length is 2 and vertical length is 1. Again, each link turns out to have maximum length $\sqrt{5}$.

Group three:

This case includes links between rows $k+1$ and $k+2$. All these links connect nodes in the lower row to the upper node or to the node to its left. $\left\lfloor \frac{k}{2} \right\rfloor$-rotation is applied to the lower row, and $\left\lfloor \frac{k+2}{2} \right\rfloor$ to the upper one. Hence, the upper row has one more node rotation than the lower one.

Horizontal length: After rotation, links in the lower row are connected to the upper one and to the one on the right $(\text{mod}\, k+1)$. After applying the same shuffle, maximum horizontal length becomes 2.

Vertical length: Column shuffle A takes rows $k+1$ and $k+2$ into the last two positions, so vertical distance is 1.

Again, each link turns out to have maximum length $\sqrt{5}$.

Group four:

This group is composed of links between nodes in the same row, in both edges of the row.

Horizontal length: No matter which rotation is applied to the row, the first and last nodes become consecutive. Both shuffles mean consecutive nodes have maximum horizontal distance 2.

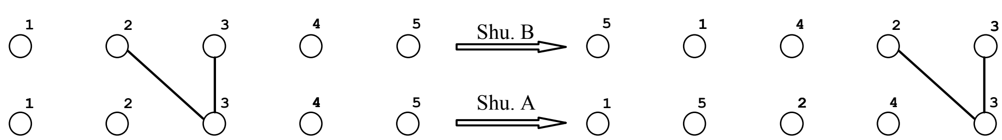

Figure 7: Group 1 case I, c)

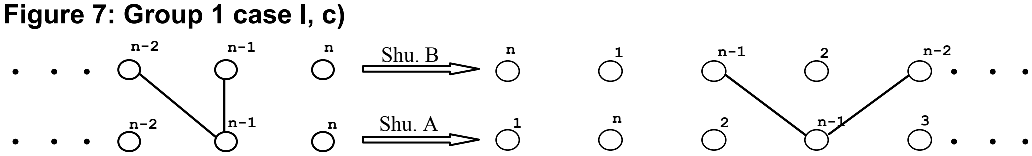

Figure 8: Group 1 case I, d)

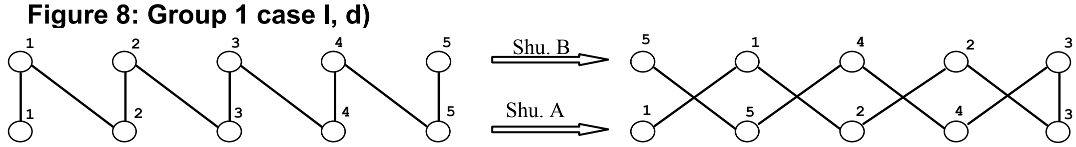

Figure 9: Example for Group 1 case I, k=4

Vertical length: Nodes in a row remain in the same row after column shuffle. Vertical distance is 0.

In this case maximum link length is 2. Figure 10 shows an example for 2-rotation and shuffle B.

QED.

To sum up, we show a final example for the case $k = 4, N = 41$ nodes. Figure 11 shows how to rotate and shuffle rows. Notice that *m*-rotation increases every two rows (except for the break in row $k+1$), which allows two things:

- Applying alternative shuffles (A, B, A, B…), which makes the links bounded, as is shown in the first group of the proof. This is why shuffles A and B can be considered complementary.
- The last row gets *k*-rotation, where $k \equiv -1(\bmod k+1)$, which is necessary for the links towards the two first rows to be bounded

Figure 12 shows the final layout for the previous example.

3.2. Bounded Link Layout for Quasi-Dense Rectangular Midimews

Networks containing $N = 2k^2 + 2k$, can be laid in a rectangle of $2k \times (k+1)$, as shown in Figure 13, which is more practical. The first node may be any of them. We will begin with node 1 for resemblance to previous figures, leaving node 0 in the last row, after node $2k^2 + 2k - 1$. The next algorithm produces a new bounded layout for these networks.

Algorithm 2: Bounded Layout for Quasi-Dense Midimews

***Step 1 or Initial layout*:** Arrange the $2k^2 + 2k$ nodes in $2k$ rows $(1,2,...2k)$, with $k+1$ nodes in each row, setting nodes in consecutive order as in the initial layout of Figure 13.

Step 2 or Row rotation and shuffle:

- For rows $1 \le i \le k$, rotate $\left\lfloor \frac{i-1}{2} \right\rfloor$ each row and apply shuffle A to odd rows and shuffle B to even ones.
- For rows $k+1 \le i \le 2k$, rotate $\left\lfloor \frac{i}{2} \right\rfloor$ each row and apply shuffle B to odd rows and shuffle A to even ones.

Step 3 or Column shuffle A: Shuffle all columns according to method A. Shuffle B could also be used, as the number of rows in this case is even, and there is no need for the last row to end up between rows 1 and 2.

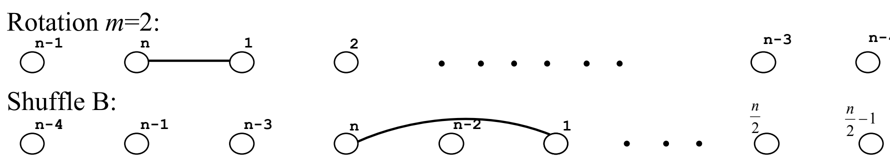

Figure 10: Rotation and shuffle

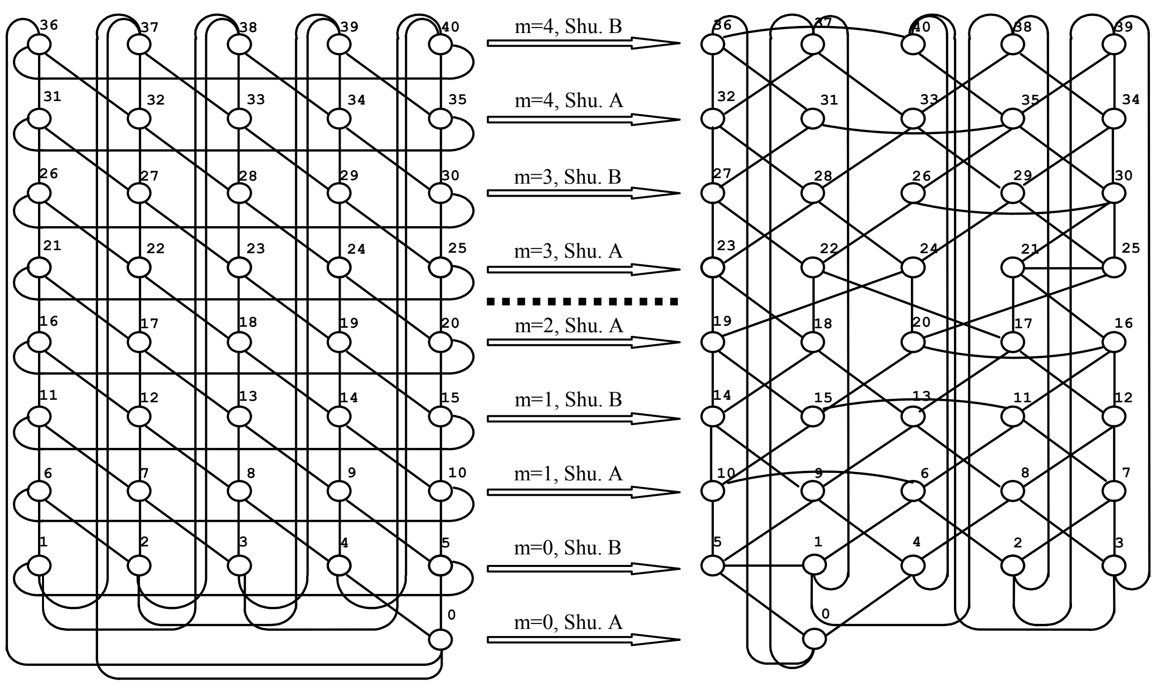

Figure 11: Initial disposition and row shuffle example for case k=4.

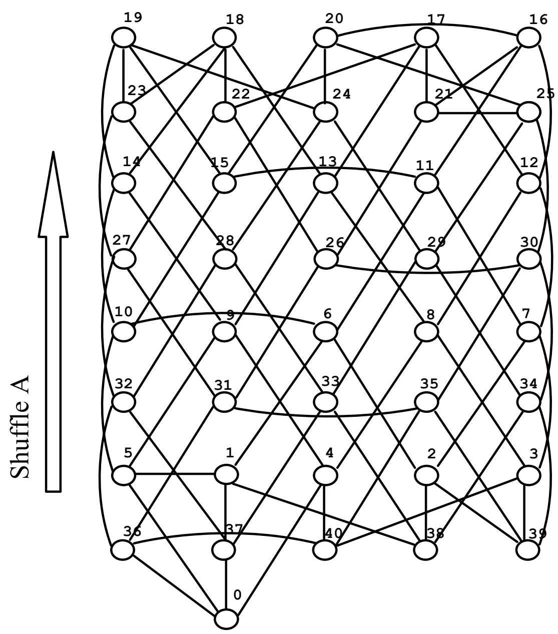

Figure 12: Column shuffle example, k=4

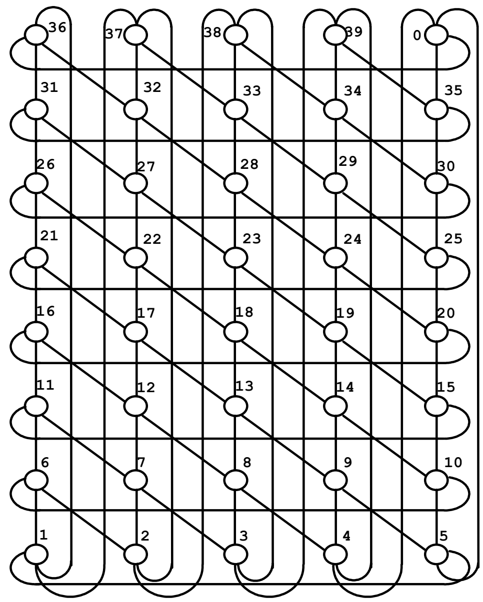

Figure 13: Non dense networks initial layout, N=2k² + 2k. Case k = 4.

Theorem 2: Given a positive integer *k,* Algorithm 2 maps a quasi-dense Midimew of diameter k onto a rectangular bi-dimensional layout in which the size of any of its links is equal to or less than $\sqrt{5}$.

Sketch of the proof:

The proof is quite similar to that of the previous case for dense Midimew networks. Group 3, in this case, is composed of links that join nodes from rows k and $k+1$, which are the rows that delimit the two sections for rotations and shuffles. The proof only differs in Group 2, that is, links between the last row $2k$ and the first row 1, but we do not include the details for the sake of clarity.

Figure 14 shows the final layout for a 40-node network with $k=4$ in which the maximum link length is $\sqrt{5}$. Notice the fact that this layout has radial symmetry.

Networks containing $2k^2$ nodes, which have been used in [5], can also be laid in a rectangle, with size $2k \times k$. In this case, optimal jumps are $\{k-1, k\}$. Due to the resemblance to Figure 13, with only one column less, Algorithm 2 can also be applied with the same result of bounded links with maximum wire length $\sqrt{5}$. The initial and resulting layouts are shown in Figure 15.

4. Wiring, scalability and performance

In this Section we analyze first the number of

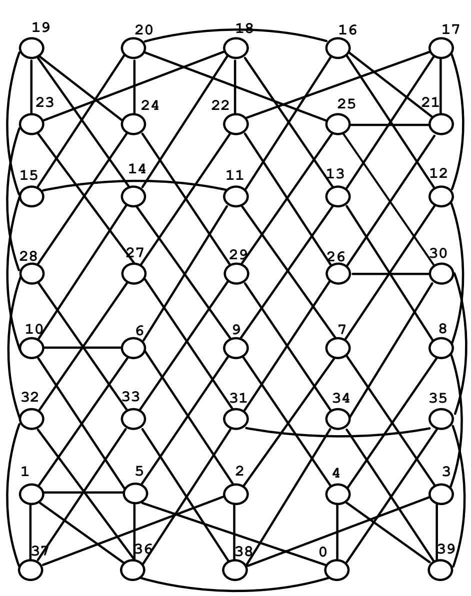

Figure 14: Final layout, N=40 nodes (k=4)

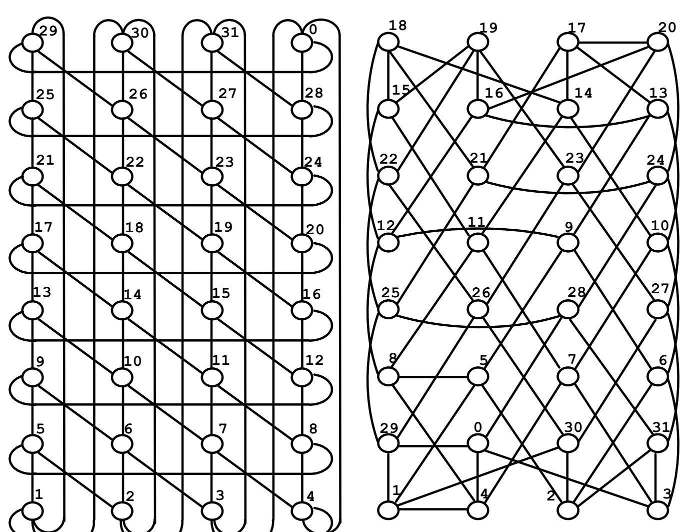

Figure 15: Initial and final layouts, N = 32 nodes (k = 4)

independent metal layers to implement the proposed layout. Second, we address the problem of how the original layout can be expanded in order to build a bigger network while preserving as much as possible the original arrangement. Finally, some coarse performance metrics are considered.

4.1. Number of necessary planes for wire laying

Four planes, and no less than four, are enough to lay all the network links without any link in the same plane cutting any other. We group the network links in the following categories:

- Links in the inner part of the figure that rise and go to the left in a certain plane (1); another plane for links that rise to the right (2).
- Peripheral links that are in the sides can be grouped alternately into any of the two previous planes, (1) and (2).
- Links that join two nodes in the same row must be grouped in a separate plane (3), as they cut links from both planes (1) and (2)
- Links between the first two rows need to be grouped into two planes, different to (1) and (2), as they cut links from both of them simultaneously. It is not enough to include them in (3), as such links cut each other.

As there are four links that cross each other, no less than 4 planes are necessary. Though it is tedious to prove this formally, the example in Figure 16 illustrates four such links among nodes $13-18$, $15-20$, $17-22$ and $19-24$ for a network with 41 nodes.

4.2. Scalability issues

An interesting property for any layout is the capability of increasing the number of nodes with the minimum impact on the existing layout. The proposed layouts can be easily scaled, adding the necessary nodes to go from a certain network size to the corresponding following one.

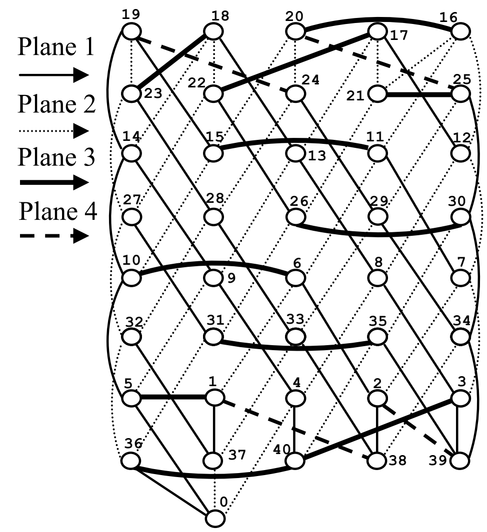

Figure 16: Necessary planes for wire laying

The number of nodes to add in an expansion is the number needed to increase the network diameter by one unit. That is, to go from a diameter k to $k' = k+1$, the number of additional nodes is $4k+4$. For example, a dense Midimew with $k=3$ contains $N=25$ nodes. Adding 16 more nodes, we can build a dense Midimew with $k'=4$ and $N=41$ nodes. Quasi-dense Midimews behave in the same way. For expanding the network, existing nodes must not be reordered. It is necessary to divide them in two parts and modify the connectivity of the border links. New nodes will lie in two rows in the middle of the network and on one side in each half, as shown in Figure 17. All nodes are renamed after the node addition.

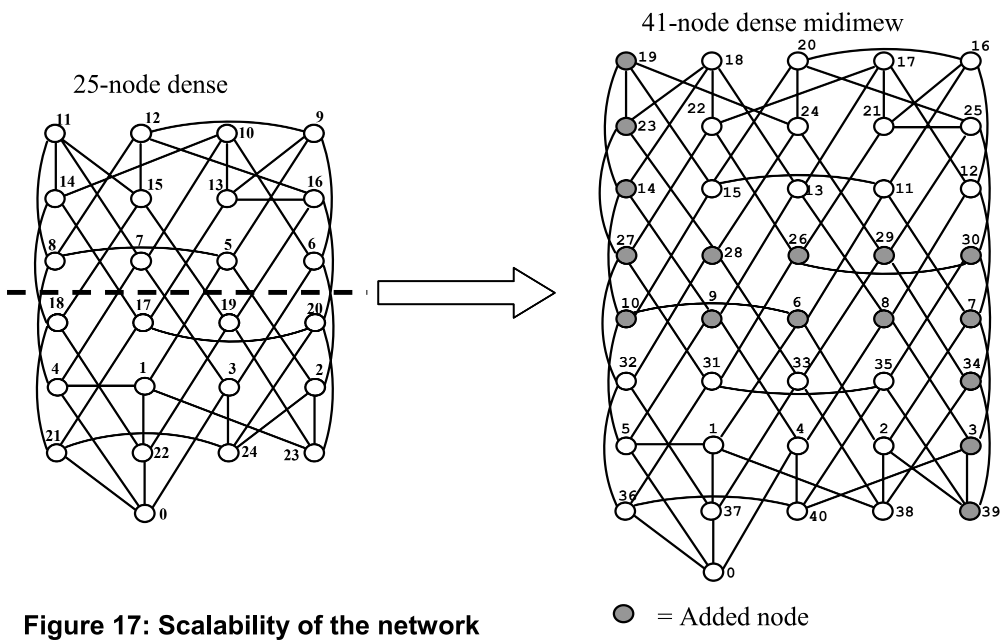

Figure 17: Scalability of the network

As can be seen, links remain in the same position, except those few that should connect old and new nodes, which might change. See, for example, link $24-2$ in the left layout, and $40-3$ in the right one.

4.3. Some Performance Indicators

To end the Section, we are going to consider some indicators which can roughly predict the behavior of the proposed layout when compared with a folded Torus. The next table shows maximum and average physical path lengths for both topologies. L stands for the original unitary link length.

Table 2: Physical distances comparative

Topology	Link length	Approximate diameter	Approx. Aver. dist.
2D torus	$2L$	$L\sqrt{4N}$	$L\sqrt{N}$
Midimew	$\sqrt{5}L$	$L\sqrt{\frac{5N}{2}}$	$L\frac{\sqrt{10N}}{3}$

As can be seen, the diameter physical path length is improved by a factor $\sqrt{\frac{8}{5}}=1.26$ (26%), whereas average distance physical path length increases by $\sqrt{\frac{10}{9}}=1.05$ (5%). For example, for a 256-node network, diameter and average path lengths are $32L$ and $16L$ in a folded Torus, while they are $25.3L$ and $16.7L$ in a Midimew. Thus the proposed layout increases maximum length traffic performance while mean length traffic is hardly degraded. Nevertheless, the most important outcome of this paper is the proposal of a new layout in which network links are kept under a bounded value. Furthermore, path physical length is not the only limitation in traffic latency and throughput. Every node has to route packets, introducing a delay that depends on the number of nodes that the packet has to cross. Midimews reduce the average number of nodes a packet has to cross, thus decreasing the processing latency. In addition, Midimew networks clearly outperform Tori when executing collective communications [9].

5. Conclusions

This paper has introduced a new layout for Midimew networks which keep the maximum link length under the value $\sqrt{5}$. In consequence, we improve the previously known result by a factor $\sqrt{5}$ because the algorithm presented in [3] needs, in the worst case, links of length 5. In addition, we have explored network sizes that have not been considered before. The low number of planar layers and the graceful expandability make the proposed layout very suitable to be implemented at any scale. In this way, these topologies are clearly competitive against classical Folded Torus.

6. References

[1] R. Beivide, E. Herrada, J.L. Balcázar, Agustín Arruabarrena, "Optimal Distance Networks of Low Degree for Parallel Computers". IEEE Transactions on Computers, Vol. 40, No 10, Oct 1991, pp. 1109-1124

[2] Z. Cvetanovic. "Performance Analysis of the Alpha 21364-based HP GS1280 Multiprocessor". Proceedings of the 30th Annual International Symposium on Computer Architecture. pp. 218-228, 2003.

[3] Francis C. M. Lau, Guihai Chen, "Optimal Layouts of Midimew Networks". IEEE Transactions on Parallel and Distributed Systems, Vol 7, No 9, pp 954-961

[4] C. Sequin. "Doubly Twisted Torus Networks for VLSI Processor Arrays". Proceedings of the 8th Annual International Symposium on Computer Architecture, pp. 471-480, 1981.

[5] Y. Yang, A. Funashashi, A. Jouraku, H Nishi, H Amano and T. Sueyoshi, "Recursive Diagonal Torus: An Interconnection Network for Massively Parallel Computers". IEEE Transactions on Parallel and Distributed Systems, Vol. 12, No. 7, pp. 701-715, 2001.

[6] N.R. Adiga et al. "An overview of the BlueGene/L Supercomputer". Supercomputing 2002 Technical Papers, November 2002. Also available at: http://sc-2002.org/paperpdfs/pap.pap207.pdf

[7] B. Camp, J. Tomkins "The Red Storm Computer Architecture and its Implementation". Conference on High-Speed Computing. April 21 -24, 2003. Also available at: http://www.lanl.gov/orgs/ccn/salishan2003/program.htm

[8] V. Puente, C. Izu, J.A. Gregorio, R. Beivide, J.M. Prellezo and F. Vallejo. "Rearranging Links to Improve the Performance of Parallel Computers: The Case of Midimew Networks". Proc. of International Conference on Supercomputing. ICS'2000. pp. 44-53. 2000.

[9] R. Beivide, C. Martinez, J. Gutierrez and C. Izu. "Gaussian Networks" Submitted for publication. 2004.

[10] R. Beivide, E. Herrada, J. L. Balcazar and J. Labarta. "Optimized Mesh-Connected Networks for SIMD and MIMD Architectures". Proceedings of the14th Annual International Symposium on Computer Architecture, pp. 163-169, 1987.