Parallel Processing Letters

AN OPTIMAL EMBEDDING OF HONEYCOMB NETWORKS INTO HYPERCUBES

DOINA BEIN
School of Computer Science, University of Nevada
Las Vegas, NV 89154, siona@cs.unlv.edu

WOLFGANG W. BEIN
School of Computer Science, University of Nevada
Las Vegas, NV 89154, bein@cs.unlv.edu

NATASA BRAJKOVSKA
School of Computer Science, University of Nevada
Las Vegas, NV 89154, natasa@cs.unlv.edu

SHAHRAM LATIFI
Department of Electrical Engineering, University of Nevada
Las Vegas, NV 89154, latifi@unlv.edu

Received (received date)
Revised (revised date)
Communicated by

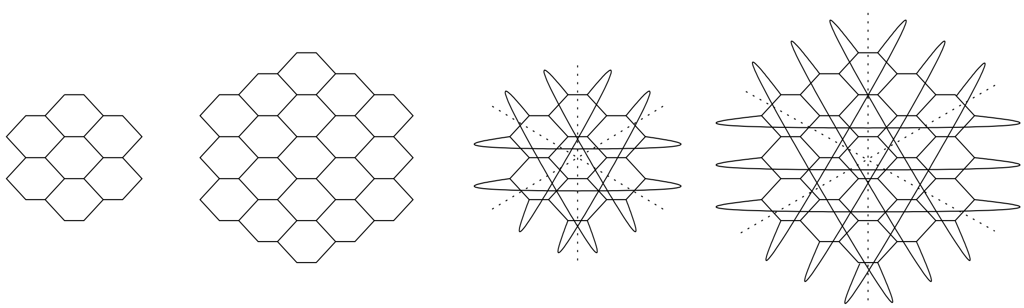
ABSTRACT

We present an optimal embedding of a honeycomb network (honeycomb mesh and honeycomb torus) of size n into a hypercube with expansion ratio of $\frac{4}{3} \approx 1.33$ when n is a power of two. When n is not a power of two, the expansion is $\frac{16}{3} \approx 5.33$, which we conjecture to be near optimal. For a honeycomb mesh, the dilation of the embedding is 1. For a honeycomb torus, the dilation can be as large as $2\lceil \log n \rceil + 3$, because of the extra links connecting symmetric opposite nodes of degree two. A honeycomb network, built recursively using hexagon tessellation, is a multiprocessor interconnection network, and also a Cayley graph, and it is better than the planar mesh with the same number of nodes in terms of degree, diameter, number of links, and bisection width.

1. Background

Built recursively using the hexagon tessellation ([1,2]), honeycomb networks are widely used in computer graphics ([3]), cellular phone base station ([4,5,6,7]), image processing ([8]), and in chemistry as the representation of benzenoid hydrocarbons ([9]). A base station controls the wireless signal into a honeycomb looks-like area, and the base stations are arranged in a honeycomb network. Honeycomb meshes ([2]) are better in terms of degree, diameter, total number of links, cost and the bisection width than mesh connected planar graphs.

An embedding consists of mapping each node of a guest graph G into a node of a host graph H, and each edge of G into an edge or a path in H. More precisely, an embedding is an injection assigning, to each vertex v of G, a single vertex $f(v)$ in H and assigning to each edge uv of G a path $f(uv)$ in H between $f(u)$ and $f(v)$, such that the internal nodes in $f(uv)$ include neither $f(z)$, for any vertex z in G, nor any point in $f(st)$ for any other edge st in G. In other words, such an embedding defines a subgraph of H which is homeomorphic to G. Appropriate embedding are very helpful in the design of parallel algorithms: Algorithms developed for one architecture (graph G) can be directly mapped to another network (graph H) without changing the code, through embedding. In this paper we present an embedding of a honeycomb mesh into a hypercube, in which the embedding is one-to-one for both nodes and links. Also, the embedding is optimal for the case when the number of nodes in the honeycomb mesh is a power of two, and we conjecture it to be near optimal in general.



(a) Honeycomb Meshes of size two and three (b) Honeycomb Torus of size two and three

Figure 1: The Honeycomb Mesh and Torus of Dimension Two and Three

Stojmenovic [2] has studied the topological properties of honeycomb meshes, routing in honeycomb meshes and honeycomb torus networks. The fact that the honeycomb torus is Hamiltonian has been proved in [10]. In [11], the first all-to-all broadcasting algorithms in honeycomb meshes with time complexity $O(n)$ were proposed. Parhami [12] gave a unified formulation for the honeycomb and the diamond network: He showed that the honeycomb network is a Cayley graph in order to prove that the interconnection network is node symmetric. The embedding of honeycomb meshes into trees, hypercubes and other networks was suggested as an open problem in [2].

The quality of an embedding depends on some factors, and the most important are the *expansion* and the *dilation*. The *dilation* of an edge uv under f is the length of the path $f(uv)$. The *dilation* of the embedding f is the maximum dilation of any edge of G under f. The *expansion* is the ratio between the number of nodes V_H in H and the number of nodes V_G in G, *i.e.* $\frac{V_H}{V_G}$. A good embedding has small values for dilation and expansion.

In this paper, we design embeddings of the n-dimensional honeycomb mesh and torus into hypercubes: an expansion ratio of $\frac{4}{3}$ is obtained, which is optimal when n, the dimension of the honeycomb mesh, is a power of two, otherwise the expansion is up to $\frac{16}{3}$, which is near optimal. The dilation is 1 for honeycomb mesh, and at most $2\lceil \log n \rceil + 3$ for honeycomb torus, because of the extra links connecting symmetric opposite nodes of degree two. The rest of the paper is organized as follows. Section 2 introduces basic terminology and general background. It also gives a $\frac{4}{3}$ expansion embedding of a honeycomb mesh into a rectangular planar mesh. In Section 3 we describe the method used for embedding a honeycomb mesh and torus into a hypercube. This method builds on the procedure described in 2. Concluding remarks are given in Section 4.

2. Honeycombs and Hypercubes

The honeycomb mesh was defined in [2] recursively as follows:

Definition 1 (Honeycomb mesh HM_n) *The honeycomb mesh of size 1, HM_1, is a hexagon. The honeycomb mesh of size 2, HM_2, is a hexagon surrounded by six other hexagons touching along the boundary edges of HM_1. A honeycomb mesh of size n, HM_n, is obtained by adding a layer of hexagons around the boundary of HM_{n-1}.*

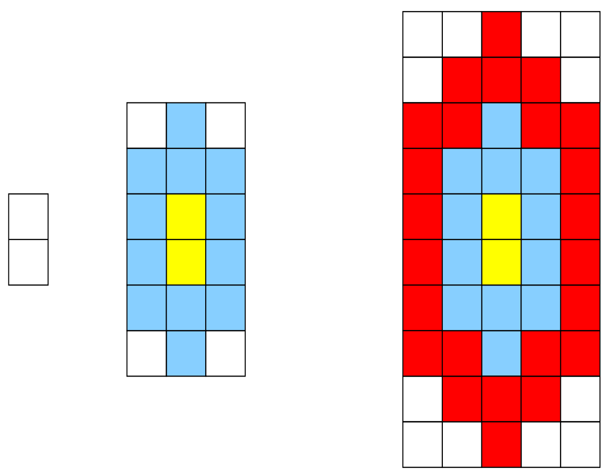


Figure 2: Embedding HM_1, HM_2 and HM_3 into Planar Grids

Figure $1(a)$ shows how HM_3 is obtained from HM_2. Properties of honeycomb meshes were studied in [2]: The total number of nodes in HM_n is $6n^2$ and the total number of links is $9n^2 - 3n$. In embeddings we use the notion of *tile*, which is a 2×3 mesh. As described in [2], the *honeycomb torus* is obtained by joining nodes of degree two such that the symmetry is kept. Figure $1(b)$ shows HT_2 and HT_3 obtained from HM_2, respectively HM_2. The lines of mirror symmetry for the wrap-around links are shown with dotted lines. The total number of nodes in HT_n is the same as for HM_n, $6n^2$, but the total number of links is $9n^2$. In embeddings we use the notion of *tile*, which is a 2×3 mesh.

We now describe the honeycomb mesh embedding into a planar grid (mesh) with *expansion* $= \frac{4}{3}$ and *dilation* $= 1$. This embedding follows directly from the brick drawing of the honeycomb network found in [2], by a simple calculation. The planar grid has $N = 2n$ nodes horizontally and $M = 4n - 1$ nodes vertically, so there are $(8n^2 - 2n)$ nodes and $(16n^2 - 10n + 1)$ links. The embedding of [2] is done recursively. A hexagon is embedded as a tile, which also represents the embedding of HM_1. HM_2 is obtained by adding 6 vertical tiles around the embedding of HM_1, with one on top, one at the bottom. The others are arranged such that one half-side covers one half-side of the previous added tile, and the other half-side covers the embedding of HM_1. Similarly, HM_n is obtained from HM_{n-1}. Figure 2 shows the embedding of HM_1, HM_2 and HM_3 into planar grids.

When going from HM_1 to HM_2, the left and right side tiles add two more nodes horizontally: $N_2 = N_1 + 2$, and the top and bottom tiles add 4 more nodes vertically: $M_2 = M_1 + 4$. For the HM_n embedding, the number of nodes in the planar grid are therefore $N_n = 2n$ and $M_n = 4n - 1$, $n \geq 1$. The expansion is $\frac{8n^2 - 2n}{6n^2}$, which approaches $\frac{4}{3}$ as n goes to infinity. We note that the dilation has value 1. Thus the next Theorem follows immediately from the work in [2].

Theorem 1 *(a)* HM_n *can be embedded into a planar rectangular grid of size* $(2n) \times (4n - 1)$, *with expansion* $\frac{4}{3}$ *and dilation=1.*
(b) Given a rectangular grid with $n \times m$ *nodes, we can embed an* HM_k *into it with dilation=1, and* $k = min(\lfloor \frac{n}{2} \rfloor, \lfloor \frac{m+1}{4} \rfloor)$.

A *n-dimensional hypercube* (or *n-cube*) has 2^n vertices. The vertices are labeled with n-bit binary numbers $b_{n-1}b_{n-2}...b_1b_0$, in such a way that for any two adjacent vertices, their labels differ in exactly one bit. The links are labeled with values from 0 to $(n-1)$ called *dimensions*, such that a link is labeled i if it joins vertices whose labels differ in the i^{th} bit, $0 \leq i < n$. Links with the same label have the same direction. (Figure 3 shows the 2-dimensional and 3-dimensional hypercubes. We note that the 0-dimensional hypercube is a simple node, and the 1-dimensional hypercube is a segment.) We note that in the binary string corresponding to the label of a node, the bits are stored from right to left, such that the rightmost bit corresponds to the dimension 0, and the leftmost bit corresponds to the dimension (n-1): $b_{n-1}b_{n-2}...b_1b_0$. Finally we note, that for defining a hypercube, is enough if we specify the label of one node (generally of the form (0,0,...0)) and the labels (dimensions) for each link. The labels of the rest of the nodes can be easily derived from this data: recursively, starting from the given node label, follow each dimension and invert the bit corresponding to the dimension.

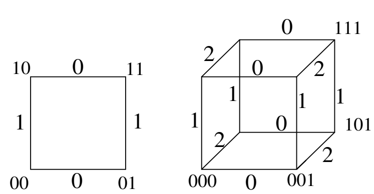


Figure 3: The 2-Dimensional and 3-Dimensional Hypercube

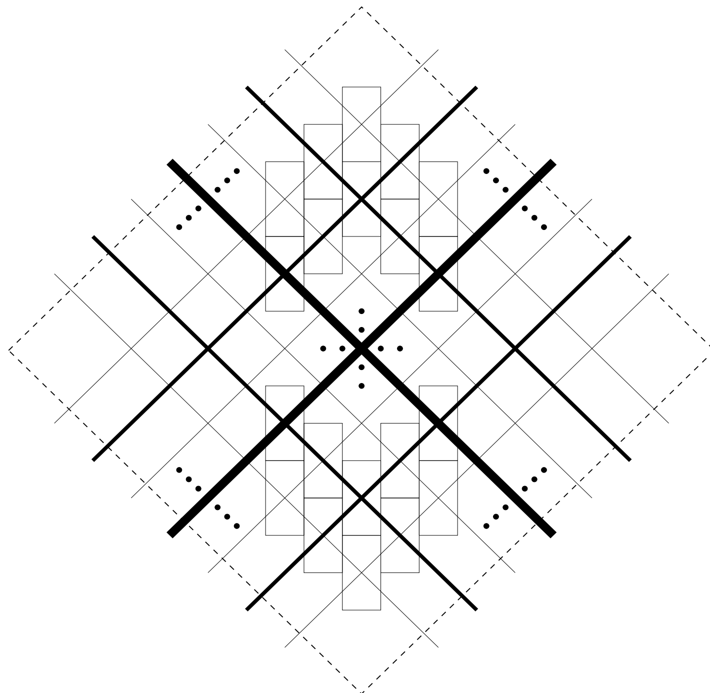


Figure 4: Orientation of Bisectors into the Brick Drawing of the Honeycomb Mesh

3. An Optimal Hypercube Embedding

In this section we define the embedding of HM_n into a $(2\lceil \log n \rceil + 3)$-dimensional hypercube, which gives an optimal expansion of $\frac{4}{3}$ when n is a power of two.* When n is not a power of two, then $\lceil \log n \rceil$ grows asymptotically to $1 + \log n$, so the expansion grows asymptotically to $\frac{16}{3}$. The dilation is 1, independent on the value of n. Applied to HT_n, the dilation of the embedding increases up to $2\lceil \log n \rceil + 3$, because of the extra links connecting symmetric opposite nodes of degree two.

We will first assume that n be a power of two. Later we describe how to adapt our method to the general case. Figure 4 contains a procedural diagram for our method. Because a hypercube can be defined in terms of its labels for each link, finding the labels for the link in the honeycomb mesh suffices to describe the hypercube embedding. Our embedding algorithm starts by creating the brick drawing of the HM_n ([2]). All horizontal links are labeled 0. For the vertical links, the algorithm then superimposes bisectors oriented (a) northeast to southwest, and (b) northwest to southeast over the brick drawing as shown in Figure 4. We call bisectors oriented northeast to southwest *upper* bisectors, and bisectors oriented northwest to southeast *lower* bisectors. In the figure, two lines – the *main* upper and the *main* lower bisector – intersect at the center of the brick drawing and are shown as very thick bold lines. Any link that is touched by the upper main bisector is given the label

*all logarithms are base 2.

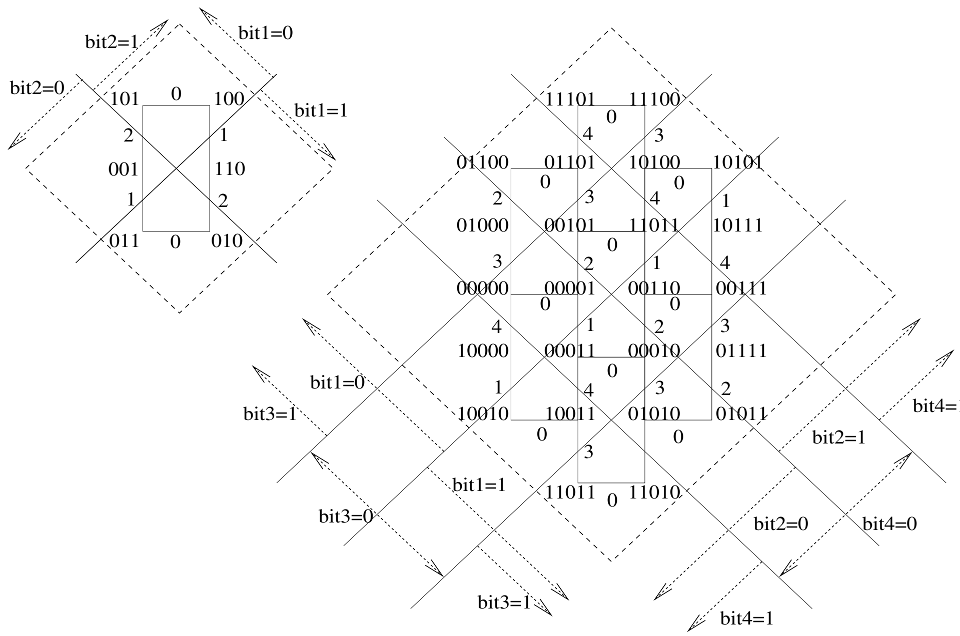


Figure 5: The Embedding of HM_1 into Q_3 and HM_2 into Q_5

1, and any link that is touched by the lower main bisector is given the label 2. We call this initial step *cut 1*.

For *cut 2* we further divide each resulting region, the region above the upper main bisector as well the region below the upper main bisector, by a secondary bisector. Those bisectors are shown in the figure as bold lines which are less thick than the main two bisectors. Any link that is touched by these two bisectors is given the label 3. We repeat this process for the main lower bisector, thus obtaining two secondary lower bisectors. Any link that is touched by those two bisectors is given the label 4. This concludes cut 2.

We continue this process iteratively. After cut $(i-1)$, we have 2^{i-1} slices oriented northeast to southwest, and 2^{i-1} northwest to southeast. Each of these slices receives a new bisector during cut i. Any link that is touched by a new upper bisector receives label $2i-1$, and any link that is touched by a new lower bisector receives label $2i$. For $n = 2^k$ we have $k+1$ cuts. Therefore if n is a power of two, there will be a total of $\lceil \log 2n - 1 \rceil = 1 + \lceil \log n \rceil$ such cuts.

For example, in Figure 5 for the HM_1 embedding, we have two bisectors, 1 and 2, which divide the rectangle into four equal sized slices: clockwise, from the left corner of the dotted diamond, the slices have $(bit_1 = bit_2 = 0)$, $(bit_1 = 0, bit_2 = 1)$,$(bit_1 = bit_2 = 1)$, and $(bit_1 = 1, bit_2 = 0)$. The nodes in each of these slices are uniquely identified by the bit_1 and bit_2 values. The same figure also shows the case of HM_2.

If n is not a power of two, we have to specify how to break ties when a slice is not

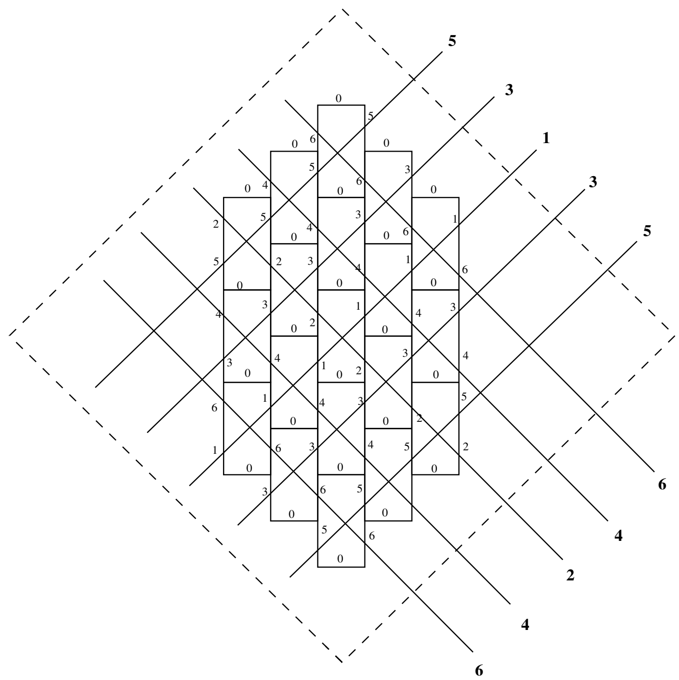


Figure 6: The HM_3 Hypercube Embedding into Q_7

even. In effect, we have in case of a tie two pairs of middle bisectors. In this case we simply choose the ones which are closer to previous labeled bisector. (Figure 6 shows a simple example.) It is clear that there are again in total $\lceil \log 2n - 1 \rceil = k+1$ cuts ($k = \lceil \log n \rceil$). Thus in general, HM_n with $6n^2$ nodes can be embedded into Q_t, $t = 2\lceil \log n \rceil + 3$, which has 2^t nodes.

Theorem 2 *(a) HM_n can be embedded into hypercube of dimension t, Q_t, $t = 2\lceil \log n \rceil + 3$, with dilation of 1 and expansion in between $\frac{4}{3}$ and $\frac{16}{3}$. When n is a power of two, this is an optimal embedding.*
(b) Given a hypercube of dimension n, we can embed an HM_k into it, with $k = 2^{n-3}$, dilation=1 and expansion of $\frac{4}{3}$.

Proof. We have $2^{2 \log n + 3} \leq 2^{2\lceil \log n \rceil + 3} < 2^{2 \log n + 5}$, and therefore $8n^2 \leq 2^t < 32n^2$. The expansion of the embedding is the ratio of number of nodes in the hypercube (2^t) over the number of nodes in the honeycomb mesh ($6n^2$), yielding:

$$\frac{4}{3} \leq expansion < \frac{16}{3}$$

When n is a power of two, then $t = 2\log n + 3$, so we obtain the expansion of $\frac{4}{3}$. When n is not a power of two, then $\lceil \log n \rceil$ grows asymptotically tight to $\log n + 1$, so the expansion grows asymptotically tight to $\frac{2^{2(\log n+1)+3}}{6n^2} = \frac{32n^2}{6n^2}$.
Our result is optimal in the case where n is a power of two. To see this we simply note that, by taking a dimension for the hypercube less than $t = 2\log n + 3$ bits, we

do not have a correct embedding: If $k = \log n$, then we need $2k+3$ bits in the node representation in the hypercube. This is obvious because if we have only $2k+2$ bits, then the corresponding hypercube will have $2^{2k+2} = 4n^2$ nodes.
Statement (b) follows directly from statement (a) $\Box$.

n	V_{HM}	V_{hyp}	e
100	$6*100^2 = 60000$	$2^{2*7+3} = 131072$	$\frac{131072}{60000} = 2.18$
128	$6*128^2 = 98304$	$2^{2*7+3} = 131072$	$\frac{131072}{98304} = 1.33$
500	$6*500^2 = 1500000$	$2^{2*9+3} = 2097152$	$\frac{2097152}{1500000} = 1.39$
512	$6*512^2 = 1572864$	$2^{2*9+3} = 2097152$	$\frac{2097152}{1572864} = 1.33$
1000	$6*1000^2 = 6000000$	$2^{2*10+3} = 8388608$	$\frac{8388608}{6000000} = 1.39$
1024	$6*1024^2 = 6291456$	$2^{2*10+3} = 8388608$	$\frac{8388608}{6291456} = 1.33$
5000	$6*5000^2 = 150000000$	$2^{2*13+3} = 536870912$	$\frac{536870912}{150000000} = 3.57$
8192	$6*8192^2 = 402653184$	$2^{2*13+3} = 536870912$	$\frac{536870912}{402653184} = 1.33$
10000	$6*10000^2 = 600000000$	$2^{2*14+3} = 2147483648$	$\frac{2147483648}{600000000} = 3.57$

Table 1: Expansion Ratios

For given values of n, the size of the honeycomb mesh or torus, the number of nodes in HM_n or HT_n (V_{HM}) versus the number of nodes in the hypercube using our embedding (V_{hyp}), and the expansion ratio ($e = V_{HM}/V_{hyp}$) are shown in Table . For honeycomb torus HT_n, the only difference in embedding is given by the labels of the extra links, that connect opposite nodes. The nodes in each such a pair are symmetric towards the main upper and main lower bisectors, which gives them a distance among their binary codings of two. So they cannot be connect by a link in the hypercube (from the definition of the hypercube, only nodes whose binary codings differ in one position are connected through a link.) In fact, depending on the pair of the nodes, the difference in binary coding can be as much as $2\lceil \log n \rceil + 3$, so the link will be embedded into a path of length $2\lceil \log n \rceil + 3$.

4. Conclusions

We note that an embedding where we first embed the honeycomb mesh into the grid as described in Section 2 and then embed the grid into a hypercube using Grey codes does not give a better result. To see this we note that the rectangular grid in which the HM_N is embedded has $2n$ nodes in one dimension and $4n-1$ nodes in the other dimension. Using now Gray codes, the mesh of dimension $2n(4n-1)$ can be embedded into a hypercube of dimension t_i, $t_i = \lceil \log(2n) \rceil + \lceil \log(4n-1) \rceil$. It is easy to show that, for any value of $n \geq 1$, $\lceil \log(4n-1) \rceil = \lceil \log(4n) \rceil$. Therefore, by this indirect embedding we obtain an hypercube of dimension $t_i = \lceil \log(2n) \rceil + \lceil \log(4n) \rceil = 3 + 2\lceil \log n \rceil$ which is of the same dimension as the one obtained from our direct embedding.

There are a number of problems on honeycomb meshes that remain open for further investigation. Initial investigations suggest, and we conjecture, that the HM_n embedding presented here is near optimal even when n is not a power of 2. However we have not been able to formally prove this. Other interesting problems concern the embedding of honeycomb meshes into a star graph and other Cayley graphs. Such problems are are still open. We feel that the method of embedding the honeycomb mesh into hypercube given in this paper can be a starting point for those embeddings.

Acknowledgements

Wolfgang W. Bein acknowledges NSF support under CCR-0312093 for Information Technology Research.

References

[1] I. Stojmenovic. Honeycomb networks. In *Proceedings of Mathematical Foundations of Computer Science MFCS 1995, Lecture Notes in Computer Science*, volume 969, pages 267–276. Springer-Verlag, 1995.

[2] I. Stojmenovic. Honeycomb networks: Topological properties and communication algorithms. *IEEE Transactions on Parallel and Distributed Systems*, 8(10):1036–1042, 1997.

[3] L. N. Lester and J. Sandor. Computer graphics on hexagonal grid. *Computer Graphics*, 8:401–409, 1984.

[4] Yuh-Shyan Cen, Yau-Wen Nian, and Jang-Ping Sheu. An energy-efficient diagonal-based directed diffusion for wireless sensor networks. In *Proceedings of the 9th International Conference and Parallel and Distributed Systems, Taiwan*, pages 445–450. IEEE, 2002.

[5] A. Gamst. Homogeneous distribution of frequencies in a regular hexagonal cell system. *IEEE Transactions on Vehicular Technologies*, 31(3):132–144, 1982.

[6] U. Black. Mobile and wireless networks. *Prentice Hall PTR, Upper Saddle River, NJ*, 1996.

[7] V. K. Garg and J. E. Wilkes. Wireless and personal communication systems. *Prentice Hall PTR, Upper Saddle River, NJ*, 1996.

[8] S.B.M. Bell, F.C.Holroyd, and D.C.Mason. A digital geometry for hexagonal pixels. *Image and Vision Computing*, 7:194–204, 1989.

[9] R. Tosic, D. Masulovic, I. Stojmenovic, J. Brunvoll, B.N. Cyvin, and S.J. Cyvin. Enumeration of polyhex hydrocarbons to h=17. *Journal Of Chemical Information and Computer Sciences*, 35:181–187, 1995.

[10] G.M.Megson, Xiaofan Yang, and Xiaoping Liu. Honeycomb tori are Hamiltonian. *Information Processing Letters*, 72:99–103, 1999.

[11] Jean Carle, Jean-Frederic Myoupo, and David Seme. All-to-all broadcasting algorithms on honeycomb networks and applications. *Parallel Processing Letters*, 9(4):539–550, 1999.

[12] Behrooz Parhami and Ding-Ming Kwai. A unified formulation of honeycomb and diamond networks. *IEEE Transactions on Parallel and Distributed Systems*, 12(1):74–79, 2001.