Algorithmica (1996) 16: 161–180

Algorithmica

A New Combinatorial Approach to Optimal Embeddings of Rectangles[1]

Shou-Hsuan S. Huang,[2] Hongfei Liu,[2] and Rakesh M. Verma[2]

Abstract. An important problem in graph embeddings and parallel computing is to embed a rectangular grid into other graphs. We present a novel, general, combinatorial approach to (one-to-one) embedding rectangular grids into their ideal rectangular grids and optimal hypercubes. In contrast to earlier approaches of Aleliunas and Rosenberg, and Ellis, our approach is based on a special kind of doubly stochastic matrix. We prove that any rectangular grid can be embedded into its ideal rectangular grid with dilation equal to the ceiling of the compression ratio, which is both *optimal* up to a multiplicative constant and a substantial generalization of previous work. We also show that any rectangular grid can be embedded into its nearly ideal square grid with dilation at most 3. Finally, we show that any rectangular grid can be embedded into its *optimal* hypercube with *optimal* dilation 2, a result previously obtained, after much research, through an *ad hoc* approach. Our results also imply optimal simulations of two-dimensional mesh-connected parallel machines by hypercubes and mesh-connected machines, where each processor in the guest machine is simulated by exactly one processor in the host.

Key Words. Graph embedding problems, Minimum dilation, Minimum expansion, Mesh-connected processor arrays, Theory of VLSI layouts, Parallel computation.

1. Introduction. An important problem in graph embeddings and parallel computing is to embed a rectangular grid into other graphs. *Formally, an* **embedding** *of a graph* $G=\langle V, E\rangle$ *into a graph* $G' = \langle V', E'\rangle$ *is a* **one-to-one** *function* $\varphi: V \to V'$. *The quality of the embedding is measured by two parameters, the dilation and the expansion. The* **dilation** *of* φ, *denoted by* $\delta(\varphi)$, *is defined as* $\delta(\varphi) = \max\{dist(\varphi(u), \varphi(v)) \mid (u, v) \in E\}$, *where* $dist(a, b)$ *denotes the shortest path length between the nodes a and b in* G'. *The* **expansion** *of* φ, *denoted by* $\varepsilon(\varphi)$, *is defined as* $\varepsilon(\varphi) = |V'|/|V|$. In this paper the host graph G' will be either a rectangular grid or a hypercube. We need the following definitions when both G and G' are rectangles. Given two rectangular grids G and G' of sizes respectively $h \times w$ and $h' \times w'$, we assume, without loss of generality, that $h \le w$ and $w' < w$. Let h' be the smallest integer such that $h'w' \ge hw$. We call $h' \times w'$ an *ideal rectangular grid* of G and $\rho = w/w'$ the *compression ratio.* We call the square grid of side of s where $s = \lceil\sqrt{hw}\rceil$ the *ideal* square grid and the square of side $s+1$ the *nearly ideal* square grid.

Rectangular grid embedding problems have many applications; here we mention a few of them. The importance of grid algorithms to scientific applications combined with

[1] A preliminary version of this paper appeared at the IEEE International Parallel Processing Symposium, 1994. The research of the third author was supported in part by NSF Grants CCR-9010366 and CCR-9303011.

[2] Department of Computer Science, University of Houston, Houston, TX 77204, USA. s_huang/rmverma@cs.uh.edu.

Received August 16, 1993; revised September 28, 1994. Communicated by F. T. Leighton.

versatility and universality of the hypercube necessitates the study of grid embeddings into hypercubes [BCLR1], [BCLR2]. As noted by Aleliunas and Rosenberg [AR], embeddings of rectangles into squares can be used in the design of very large scale integrated circuits since the natural design of a circuit may be rectangular but the circuit may eventually be manufactured on a square chip. In this case the critical factors of area and wire length are represented by expansion and dilation. Lombardi *et al.* [LSS] have used rectangular embedding techniques for reconfiguring an array of processors which contain switching elements. Finally, a solution to the two-dimensional rectangular embedding problem may be used to solve more complex problems. For example, Chan [C1], [C2] and Chan and Chin [CC] do this in discussing the embedding of multidimensional grids into hypercubes, which has immediate application to efficient simulation of grid-like parallel processing architectures by hypercube architectures. Because of its fundamental importance, the rectangular grid embedding problem has received much attention. Previous results obtained are given below.

ONE-TO-ONE EMBEDDING OF RECTANGLES INTO RECTANGLES. Early attempts to embed rectangles into squares incurred an unbounded increase in the expansion, or an unbounded increase in the dilation (for example, the effort of Leiserson [L]). In 1982 Aleliunas and Rosenberg [AR] introduced the ideas of line-compression and folding, and demonstrated that there are pairs $\langle e, d \rangle$ (for expansion and dilation respectively) such that any n-vertex rectangular grid can be embedded into an en-vertex square grid with dilation d. However, they were unable, in general, to optimize both expansion and dilation simultaneously and conjectured that there may be an inherent expansion–dilation tradeoff. If expansion is minimized, i.e., we embed a rectangular grid into the smallest possible square grid, then the smallest dilation they obtained depends on the aspect ratio of the rectangle.

In 1988 Lombardi *et al.* [LSS] improved on the dilation results of Aleliunas and Rosenberg for some aspect ratios. In 1991 Ellis [E] showed that, *for small compression ratios*, there is no significant tradeoff between expansion and dilation. Specifically, Ellis showed that, *for compression ratios*, ρ, *of* 3 *or less*, any rectangular grid can be embedded into any of its ideal rectangular grids with dilation equal to $\lceil \rho \rceil$; and that any rectangular grid can be embedded into its nearly ideal square grid with dilation at most 3. However, Ellis's method, besides being restricted to small compression ratios of at most 3, is somewhat *ad hoc*, being based on the definitions of light and heavy 2-tiles and 3-tiles and the formation of composite tiles by "iteration" and "diagonalization."

MANY–ONE EMBEDDINGS OF RECTANGLES INTO RECTANGLES. In 1986 and 1988 Kosaraju and Atallah [KA1], [KA2] considered the d-dimensional **many–one** simulations of one rectangle by another. Here, each processor in the host rectangle can be the image of (simulate) a constant (> 1) number of processors in the guest. They showed that, with the flexibility of a many–one embedding, the dilation is $\Theta(\max_{1 \le i < d}(g_{i+1} \cdots g_d / h_{i+1} \cdots h_d)^{1/i})$, where $g_1 \times g_2 \cdots \times g_d$ is the guest and $h_1 \times h_2 \cdots \times h_d$ is the host. Note that when $d = 2$, the expression in the Θ equals the compression ratio. Although they do not specify the constants involved, a careful reading of the paper shows that, for the upper bound, the constant is at least two even by the most conservative estimates.

ONE-TO-ONE EMBEDDINGS OF RECTANGULAR GRIDS INTO THEIR OPTIMAL HYPERCUBES (the smallest size hypercube that has at least as many nodes as the grid). This problem has also attracted much attention. After several unsuccessful attempts by many researchers (see [C3]), Chan [C1], [C3] gave a complex proof of dilation 2 embeddings.

1.1. *Overview of Our Approach and Results.* In this paper we (i) present a novel, general, combinatorial approach to the rectangle (**one-to-one**) embedding problem, and (ii) give several interesting applications of our approach. In contrast to earlier methods, our new method is elegant and totally combinatorial based on our identification of a special kind of doubly stochastic matrix with nice properties, which we call an elliptic doubly stochastic matrix. As part of the applications, we prove the following results:

- *For all compression ratios*, any rectangular grid can be embedded into any of its ideal rectangular grids with dilation equal to $\lceil w/w' \rceil$. This main result of our paper is: (a) **optimal** to within a multiplicative constant (see lower bound proofs in [DEL], [KA2]), (b) a substantial generalization of earlier work on one-to-one embeddings [AR], [LSS], [E], and (c) for two dimensions an improvement by at least a factor of two even with the one-to-one requirement over the work on many–one embeddings ([KA1], [KA2]).
- Any rectangular grid can be embedded into its nearly ideal square grid with dilation at most 3.
- Any two-dimensional grid can be embedded into its optimal hypercube with optimal dilation 2.
- Another application of our embedding algorithms is to the study of "encodings" of data structures in grids or of grids in other data structures [RS], [LED], [R].

Our results are achieved partly through the introduction of a special kind of doubly stochastic matrix, which we call an elliptic doubly-stochastic matrix, partly by proving a very general theorem relating these matrices to rectangular embeddings with dilation equal to $\lceil \rho \rceil$, and partly through an interesting "expand-and-conquer" proof idea. Our proof of the general theorem on embedding rectangular grids into ideal rectangular grids is long, so we outline below the steps involved in the proof; and then we discuss some more implications of our results.

STEPS IN DEVELOPING THE GENERAL EMBEDDING SCHEME. Earlier work on one-to-one embeddings is restricted and hard to generalize. Therefore, to develop a general embedding scheme, we first represent the embedding by a matrix (Section 3). We then define a special kind of embedding matrix, which we call an elliptic doubly stochastic matrix (Section 4). Next, we prove a general theorem relating an elliptic doubly stochastic matrix to an embedding of $h \times w$ into $w \times h$ with dilation $\lceil \rho \rceil$ (Section 5). We introduce the idea of cutting and repeating this matrix to embed $h \times w$ into general $h' \times w'$ and show that dilation does not increase (Section 6). Finally, we introduce one more restriction on the elliptic doubly stochastic matrix to get the general embedding scheme into an ideal rectangular grid (Section 7). Our definition of an elliptic doubly stochastic matrix is such that the matrix is completely determined by its first row. We show how to generate this row (and thus the matrix) so that the matrix satisfies the additional restriction desired by us. We generate this row for only relatively prime values of (matrix dimension, number of ones) and then, roughly speaking, reflect and repeat this row to get the row for arbitrary pairs of these values. The proof that the matrix generated satisfies the additional restriction uses a novel expand-and-conquer approach, where we first expand the first row and prove that the expansion has a particular form, which is then exploited in our proof.

IMPLICATIONS. First, the results of Ellis and Chan emerge as special cases of our results. Second, our approach reveals an elegant combinatorial connection between rectangle embeddings and a special kind of doubly stochastic matrix, which may be of independent interest also. These 2-valued matrices satisfy several interesting properties, namely, for any two rows or columns the partial sums are different by at most one. Our results also imply optimal simulations of two-dimensional mesh-connected processor arrays by hypercubes and mesh-connected processor arrays, where each processor in the guest machine is simulated by exactly one processor in the host (the routing problems for our embedding scheme are easy to solve). Our method could also be used to solve more complex embedding problems. For example, it appears likely that the work in [KA2] for d dimensions can be improved using our method as the basis.

2. Preliminaries. A matrix $A = [a_{ij}], i, j = 1, \ldots, n$, is said to be *doubly stochastic* if the a_{ij} are real numbers subject to the conditions $a_{ij} \geq 0$, $\sum_{j=1}^{n} a_{ij} = 1$, $i = 1, \ldots, n$, $\sum_{i=1}^{n} a_{ij} = 1$, $j = 1, \ldots, n$. Doubly stochastic matrices appear in many areas of mathematics and the physical sciences: combinatorics, probability, linear algebra, physical chemistry, etc. We use this term a bit loosely in the following (we have to divide out each element by the sum of each row). Without loss of generality, assume $h \leq w$ and $w' < w$ throughout the paper.

3. Embeddings and Embedding Matrices. Suppose we are given a grid $G = h \times w$ and an ideal rectangular grid $G' = h' \times w'$ of G.

DEFINITION 1. We define an **embedding matrix** $C = (c_{ij})_{h \times w'}$ as follows:

(C1) c_{ij} is a positive integer for $1 \leq i \leq h$ and $1 \leq j \leq w'$.
(C2) $\sum_{j=1}^{w'} c_{ij} = w$ for $1 \leq i \leq h$.
(C3) $\sum_{i=1}^{h} c_{ij} \leq h'$ for $1 \leq j \leq w'$.

THEOREM 2. *Given an embedding matrix C of $G = h \times w$ to $G' = h' \times w'$, there is an embedding function associated with it.*

PROOF. The embedding function φ from G to G' is described as follows. Informally, as Figure 1 depicts, this function maps c_{ij} nodes in the ith row of G to the jth column of G' in such a way that each row of G is mapped to a *chain* in G'. Formally, let (i, j) denote the node in the ith row and jth column of rectangular grid G. The embedding

$$C = \begin{pmatrix} 2&3&3&2&2&3&3&2&2 \\ 3&2&2&3&3&2&2&3&2 \\ 3&2&2&3&3&2&2&2&3 \\ 2&3&3&2&2&3&2&2&3 \end{pmatrix}$$

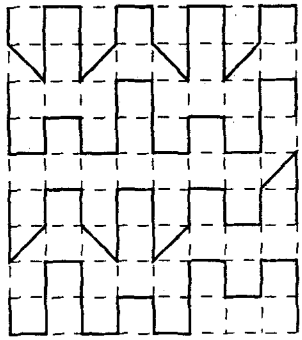


Fig. 1. Matrix C defines an embedding from grid 4×22 to grid 10×9.

function is defined as follows: $\varphi(i, j) = (k, l)$ where

$$l = \min\left\{\lambda \colon \sum_{s=1}^{\lambda} c_{is} \geq j\right\}, \qquad k = \sum_{r=1}^{i-1} c_{rl} + \begin{cases} j - \sum_{s=1}^{l-1} c_{is} & \text{if } l \text{ is odd,} \\ 1 + \sum_{s=1}^{l} c_{is} - j & \text{if } l \text{ is even.} \end{cases}$$

Using properties (C1)–(C3), it is not difficult to show that φ is one-to-one from G to G'. □

To keep the dilation small, we need to enforce additional conditions on the embedding matrix. Let $\rho = w/w'$ be the compression ratio, $\underline{\rho} = \lfloor \rho \rfloor$ and $\bar{\rho} = \lceil \rho \rceil$. The embedding matrix is called **bivalued** if $c_{ij} \in \{\underline{\rho}, \bar{\rho}\}$ for all $1 \leq i \leq h$ and $1 \leq j \leq w'$. In fact, the embedding matrix in Figure 1 is bivalued, and we only focus on such matrices in this paper.

NOTATION. Throughout the rest of this paper, we denote $\bar{\rho}$ (the bigger value) by 1 and $\underline{\rho}$ (the smaller value) by 0, except in the proofs of Theorem 5, Lemma 11, and Lemma 12 (to minimize confusion).

Our solution to the problem of embedding G into G' is to find an embedding matrix and use the above algorithm to obtain the embedding function. Thus we reduce the embedding problem to finding an embedding matrix. To find an embedding matrix (which is not necessarily square), we first define a special kind of (square) doubly stochastic matrix.

4. Elliptic Doubly Stochastic Matrices.

We first define an *even-gapped* vector, then we pair the ones in an even-gapped vector, and finally we generate the desired matrix.

DEFINITION 3. A vector of zeros and ones is said to be **even-gapped** if every maximum substring of consecutive zeros and every maximum substring of consecutive ones is of even length, except possibly at the ends.

EXAMPLE. The following two vectors are even-gapped:

$$(1\ 1\ 1\ 1\ 0\ 0\ 1\ 1\ 0\ 0\ 0),\quad (1\ 1\ 1\ 0\ 0\ 1\ 1\ 0\ 0\ 0\ 0).$$

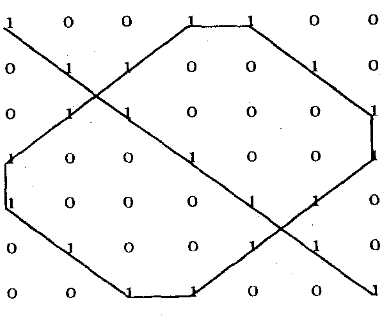
while the following two vectors are not:

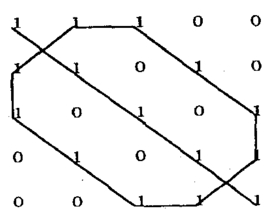
$$(1\ 0\ 0\ 1\ 0\ 1\ 0\ 1\ 1\ 0\ 0),\quad (1\ 1\ 1\ 0\ 0\ 0\ 1\ 1\ 0\ 0\ 0).$$

Now we need to pair the ones in an even-gapped vector in order to describe the generation of the corresponding embedding matrix. To avoid collision in generating the matrix, we pair the vector as follows: for each maximum substring of even length pair the ones consecutively starting from the first one in the substring and for each maximum substring of odd length (it must be at one end) take the first or last one as a pair by itself depending on whether this substring is at the left or right end, and then pair the remaining ones consecutively.

We describe the generation of a square embedding matrix based on an even-gapped vector. Take the vector as the first row of a square matrix. Move the first one in each pair of the first row of the matrix. The special pair of ones at the ends produce the diagonals. Each pair of ones produces an ellipse within the matrix as described by the following rule: the left one in a pair first moves southwest toward the west border, upon reaching the border it then starts from the next row and moves southeast toward the south border, upon reaching the border it starts from the next column and moves northeast toward the east border, finally upon reaching the border it starts from the upper row and moves toward the north border. Interestingly and obviously, it will coincide with the second one in the pair where it started in the first row.

EXAMPLE. The following examples demonstrate the creation of an ellipse inside a matrix (only the first is generated by our algorithm described in Section 7):





Notice that the even gap (including gaps of length zero) between any two pairs of ones guarantees that there is no collision between the generation of any pair of ellipses. The matrices so generated are said to be **elliptic doubly stochastic**.

From the generation of each ellipse, we know that each pair contributes two ones in each row and each column of the matrix, and each unpaired one contributes only one to each row and each column (the diagonal line). Therefore an elliptic doubly stochastic matrix, $C = (c_{ij})_{n \times n}$, has equal number of ones in each row and each column. Further, we have the following properties:

THEOREM 4. *Any elliptic doubly stochastic matrix has the following properties*:

(P1) *It is symmetric, i.e.*, $c_{ij} = c_{ji}$ *for all* $1 \le i, j \le n$.
(P2) *Any two adjacent equal-length partial row sums differ by at most one, i.e.*, $|\sum_{j=1}^{k} c_{ij} - \sum_{j=1}^{k} c_{i+1,j}| \le 1$ *for* $1 \le i < n$ *and* $1 \le k \le n$.
(P3) *If the last elements of any two adjacent equal-length subvectors are the same, then their sums are equal, i.e.*, $\sum_{j=1}^{k} c_{ij} = \sum_{j=1}^{k} c_{i+1,j}$ *if* $c_{i,k} = c_{i+1,k}$, *for* $1 \le i < n$ *and* $1 \le k \le n$.
(P4) *All row sums and column sums are equal, and row sum equals column sum.*
(P5) *If a* 2×2 *submatrix has diagonally equal elements and the corresponding row subvectors have equal sums, then the corresponding column subvectors have equal sums, i.e., if* $c_{ik} = c_{i+1,k+1}$, $c_{i,k+1} = c_{i+1,k}$, *and* $\sum_{j=1}^{k-1} c_{ij} = \sum_{j=1}^{k-1} c_{i+1,j}$ *for all valid* i, k, *then* $\sum_{r=1}^{i-1} c_{r,k} = \sum_{r=1}^{i-1} c_{r,k+1}$.

Proof Sketch. (P1) is obvious. (P2) holds because all the elements of two adjacent partial vectors of equal length can be paired diagonally and accounted to one of the ellipses (hence, they are identical), except possibly the last elements of both partial vectors. (P3) holds due to the same reason as (P2) with the additional condition of $c_{i,k} = c_{i+1,k}$. (P4) holds from the generation process. (P5) holds basically for the same reason as for (P2). We reason as follows. If all four elements of the submatrix are equal, then (P5) follows from (P3); otherwise there are two cases. In the first case the main diagonal elements of the submatrix are $\bar{\rho}$. In this case, because the corresponding row subvectors have equal sums, these two diagonal elements must belong to the same ellipse. Now, using the reasoning in (P2) and this fact we conclude that the corresponding column subvectors have equal sums. The second case is when the main diagonal elements are $\underline{\rho}$. A similar argument applies in this case also. □

5. The Embedding Given by an Elliptic Doubly Stochastic Matrix. An elliptic doubly stochastic matrix is a square embedding matrix, it represents an embedding function as stated in the following theorem.

THEOREM 5. *Any* $h \times h$ *elliptic doubly stochastic matrix* $C = (c_{ij})$ *defines an embedding from* $G = h \times w$ *into* $G' = w \times h$ *with dilation at most* $\bar{\rho}$, *where* $w = \sum_{j=1}^{h} c_{ij}$ *is the row sum and* $\bar{\rho} = \lceil w/h \rceil$ *is the ceiling of the compression ratio.*

Proof Sketch. From the properties (P1)–(P4) we can derive the properties (C1)–(C3). Hence, an elliptic doubly stochastic matrix is also an embedding matrix. We prove the dilation conclusion as follows: consider two adjacent nodes (i, j) and (i', j') in G, they are either horizontally adjacent or vertically adjacent. We prove, by case analysis using the definition of φ and the matrix properties, that their images $\varphi(i, j)$ and $\varphi(i', j')$ are at most $\bar{\rho}$ edges away in G'. The full proof appears in the Appendix. □

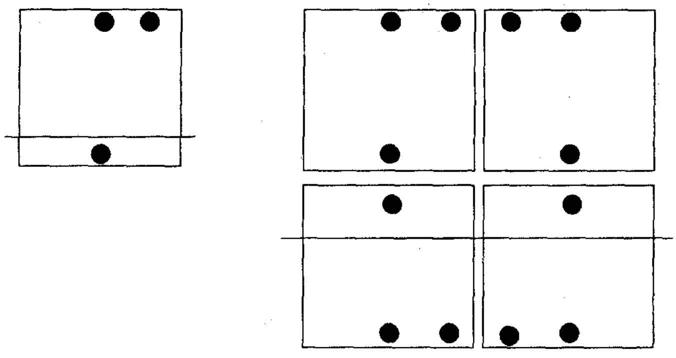


Fig. 2. Cutting and reflective repeating process.

6. Repeated Reflection and Cutting. The above theorem solves the problem of embedding grid $G = h \times w$ into $G' = w \times h$, which is one of its ideal rectangular grids. In order to come up with an embedding matrix for embedding grid $G = h \times w$ into any of its ideal rectangular grids $G' = h' \times w'$, we need to add an additional constraint on the distribution of the elements of the elliptic doubly stochastic matrix, namely, any two partial column sums of the same length differ by at most one. So far we had only imposed this restriction on adjacent partial row/column sums.

(P6) $|\sum_{i=1}^{k} c_{ix} - \sum_{i=1}^{k} c_{iy}| \leq 1$ for $1 \leq x, y \leq n$.

The reason for this restriction is that to obtain a nonsquare matrix of size $h \times w'$ from the elliptic doubly stochastic matrix C (satisfying (P6)) of size $w' \times w'$ we either choose the first h rows of C if $h \leq w'$ or repeatedly reflect (along the horizontal axis) C $\lceil h/w' \rceil$ times (each time creates a block) so that the total height is no less than h and choosing the first h rows of this "composite" matrix. Figure 2 depicts the cutting and repeating process. For example, to embed grid $G = 14 \times 23$ into $G' = 21 \times 16$, we use the top 14 rows of the 16×16 square embedding matrix (which represents an embedding from 16×23 to 23×16; the next section gives an algorithm to produce this square matrix). To embed grid $G = 19 \times 23$ into $G' = 28 \times 16$, we first obtain a 16×16 embedding matrix, then "repeat" it once to double the size to 32×16. The top 19 rows of the 32×16 embedding matrix are used. It is obvious that the cutting process does not increase the dilation, but it is less obvious for the repeating process. However, since we use C reflectively, the two vectors across the seam are identical and their corresponding chains have the same concave/convex shape. Now, because there are no unused processors in any block except the last one, the dilation between these two chains is no greater than the value of $\bar{\rho}$.

The partial column sum property, (P6), on the square matrix C can be traced back to the generating vector. Figure 3 shows the relationship between the partial column sum and the number of ones distributed in the generating vector.

DEFINITION. A window is a portion of the first row (the generating vector) obtained by projecting a partial column onto the first row, which has been reflected on both sides (see Figure 3).

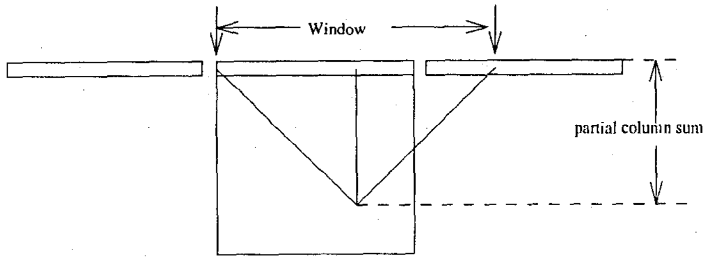


Fig. 3. Partial column sum vs. number of 1's in window.

Let $\#_1(W)$ denote the number of ones in a **window** and let $|W|$ denote its size or length. Then we have the following fact:

FACT. *Partial column sum equals* $\lceil \#_1(W)/2 \rceil$, *where* W *is the corresponding window.*

PROOF. Let W be the window obtained by projecting a partial column C_p. Each trailing one (of a pair) in W on the left-hand side of C_p and each leading one (of a pair) in W on the right-hand side of C_p contribute to the sum of C_p. If C_p contains a one (leading or trailing) in the first row, it also contributes to the sum. Therefore, we get the initial estimate $S = \lceil \#_1(W)/2 \rceil \leq$ partial column sum $\leq S + 1$ (since all the pairs completely within the window contribute exactly 1 to C_p). To get rid of the extra one in the second inequality, we reason as follows. The worst-case scenario is when W begins with a trailing one of a pair and ends on a leading one of another pair, but this implies that the length of W is even, which is impossible (projecting a partial column always gives rise to an odd-length window W). □

As long as the generating vector satisfies the **window property**, namely, if $|W_1| = |W_2|$, then $|\#_1(W_1) - \#_1(W_2)| \leq 2$, where W_1 and W_2 are two windows, the square matrix C has property (P6). In the next section we give an algorithm to generate a vector which has the window property.

7. The General Algorithm. In this section we describe our general algorithm for embedding grid $h \times w$ into its ideal rectangular grid $h' \times w'$. Based on the discussion in the previous section, we need to find a generating vector with the window property to solve the base case, i.e., the embedding of grid $s \times t$ into grid $t \times s$ ($s = w'$ and $t = w$). Let $k = t \bmod s$ (there are k bigger values in the bivalued generating vector of size s), and assume $s < t$ (compress t to s) and $2k < s$, otherwise we flip the roles of ones and zeros first, generate the vector and consequently the matrix, and flip the zeros and ones back. The reason for this flipping is to have more zeros than ones in the generation process which ensures that the rightmost element of vector U is not a 1 so that it is always possible to obtain an adjusted vector V. We first generate a vector $U = (u_1, u_2, \ldots, u_s)$ as follows:

If k is even, define $P = P_0 \cup P_1$, where $P_0 = \{\lfloor (2i-1)s/k \rfloor \mid i = 1, 2, \ldots, k/2\}$ and $P_1 = \{p + 1 \mid p \in P_0\}$. Set $u_i = 1$ if $i \in P$, and 0 otherwise.

If k is odd, define $P = P_0 \cup P_1$, where $P_0 = \{\lfloor (2i)s/k \rfloor \mid i = 0, 1, 2, \ldots, (k-1)/2\}$

and $P_1 = \{p + 1 \mid p \in P_0\}$. Set $u_i = 1$ if $i \in P$, and 0 otherwise. Note that $0 \in P_0$, but it is not used since U starts with u_1. Hence, there are exactly $2 \times (k+1)/2 - 1 = k$ ones in U; in particular $u_1 = 1$.

U is a vector of $s - k$ zeros and k ones, denote it by $U(s, k)$. For example, $U(9, 4) = (0, 1, 1, 0, 0, 1, 1, 0, 0)$ and $U(12, 5) = (1, 0, 0, 1, 1, 0, 0, 0, 1, 1, 0, 0)$.

Next we adjust the U-vector so that it becomes even-gapped. The procedure works from left to right. If the gap of zeros between two pairs of ones is odd, then push the second pair to the right by one position. Denote the adjusted vector as V. For the above examples, we have $V(9, 4) = U(9, 4)$ but $V(12, 5) = (1, 0, 0, 1, 1, 0, 0, 0, 0, 1, 1, 0)$. Finally, we use the V-vector $V(s, k)$ to generate an elliptic doubly stochastic matrix, which defines the desired embedding. V is called the *generating vector*.

Let x_i denote the index of the ith one in $U(s, k)$ (with the convention that the *special pair*—in other words the singleton 1 in index one—if it is present, is the 0th one and not the first one), then

(P7) $x_{2j-1} - x_{2i-1} \in \{\lfloor (2j - 2i)s/k \rfloor, \lceil (2j - 2i)s/k \rceil\}$,

where $1 \le i < j \le k/2$ for even k and $1 \le i < j \le \lfloor k/2 \rfloor$ for odd k. Here we used the fact

$$\left\lfloor \frac{a+b}{c} \right\rfloor - \left\lfloor \frac{b}{c} \right\rfloor \in \left\{ \left\lfloor \frac{a}{c} \right\rfloor, \left\lceil \frac{a}{c} \right\rceil \right\}.$$

Because of our construction of U it also follows that

$$x_{2j} - x_{2i} \in \left\{ \left\lfloor \frac{(2j - 2i)s}{k} \right\rfloor, \left\lceil \frac{(2j - 2i)s}{k} \right\rceil \right\},$$

where $1 \le i < j \le k/2$ for even k and $0 \le i < j \le \lfloor k/2 \rfloor$ for odd k. The proofs of these properties are straightforward for both k even and k odd.

We generate the V-vector for only relatively prime values of (s, k). Now, suppose $d = \gcd(s, k) > 1$. In this case we first generate the V-vector corresponding to $(s/d, k/d)$, then we generate the matrix based on this vector, and then we repeatedly reflect (along the vertical and horizontal axes) d times to get the desired $s \times s$ matrix. Directly generating the V-vector based on the method described above does not lead to a matrix satisfying (P6).

To prove that the $s \times s$ matrix generated from $V(s, k)$ (for relatively prime s, k) satisfies the additional restriction (P6), we use a novel expand-and-conquer approach, where we first generate directly (as given above) a vector $V(3s, 3k)$, three times the length of the desired vector. We then prove that this vector has a particular form, which is then exploited in our proof. Once we show that the vector $V(3s, 3k)$ satisfies the window requirement (which is equivalent to the matrix satisfying (P6)) for relatively prime s, k, it will be clear that matrices obtained by repeatedly reflecting the base matrix for relatively prime values also satisfy (P6).

THEOREM 6 (Expand-and-Conquer). *If* $\gcd(s, k) = 1$, *then* $V(3s, 3k) = V(s, k) \cdot V^{\mathrm{R}}(s, k) \cdot V(s, k)$, *where* V^{R} *is the reverse of* V, *and* $\cdot$ *is the concatenation.*

PROOF. We only prove the theorem for even k. A similar argument can be given for the case when k is odd. We have to show that $v_i = v_{2s-i+1}$ in the first $2s$ subvector for

$i = 1, 2, \ldots, s$, and the last one-third of vector $V(3s, 3k)$ is the same as the first one-third. Again, let x_i and y_i, $i = 1, 2, \ldots, 3k$, denote the indexes of the ith one in $U(3s, 3k)$ and $V(3s, 3k)$, respectively. The problem reduces to showing that $y_{2i-1} + y_{2k-2i+1} = 2s$ and $y_{2k+2i-1} - y_{2i-1} = 2s$. For $i = 1, 2, \ldots, k/2$, we have

$$x_{2i-1} = \left\lfloor \frac{(2i-1)s}{k} \right\rfloor.$$

For s, k relatively prime we have

$$x_{2k-2i+1} = \left\lfloor \frac{(2k-2i+1)s}{k} \right\rfloor = 2s - \left\lfloor \frac{(2i-1)s}{k} \right\rfloor - 1.$$

The adjustment process requires that exactly one of x_{2i-1} and $x_{2k-2i+1}$ be pushed to the right one position in order to produce the even-gapped V-vector. Therefore, we have $y_{2i-1} + y_{2k-2i+1} = x_{2i-1} + x_{2k-2i+1} + 1 = 2s$. This means that the first $2s$ subvector of $V(3s, 3k)$ satisfies the symmetric property. Further, we need to show that the last one-third and the first one-third of the vector $V(3s, 3k)$ are the same. Since, for $i = 1, 2, \ldots, k/2$,

$$x_{2k+2i-1} = \left\lfloor \frac{(2k+2i-1)s}{k} \right\rfloor = 2s + \left\lfloor \frac{(2i-1)s}{k} \right\rfloor.$$

Hence, $y_{2k+2i-1} - y_{2i-1} = 2s$. □

In order to show that our general algorithm works for an arbitrary rectangular grid, we only need to consider portions of the V-vector, $V(3s, 3k)$, that are obtained by projecting partial columns, i.e., windows. Observe that all windows are odd-sized. Further, it suffices to consider only those windows whose maximum size is limited to $2s - 1$ and whose center lies in the middle one-third of the V-vector. We need to show that if s, k are relatively prime, then $V(3s, 3k)$ satisfies the window property, i.e., if $|W_1| = |W_2|$, then $|\#_1(W_1) - \#_1(W_2)| \leq 2$, where W_1 and W_2 are windows of $V(3s, 3k)$ (of size less than $2s$). Based on the property of the U-vector, we have the following lemma about the V-vector:

LEMMA 7. *If L_1 and L_2 are two windows of a V-vector both starting and ending with ones and $\#_1(L_1) = \#_1(L_2)$, then they have the same length or their lengths differ exactly by* 2.

PROOF. Let L be a window of a V-vector, let $f_u(L)$ denotes its size in vector U (i.e., trace back the first and last entries of L in V to where they were originally in U and measure the length), and let $f_v(L) = |L|$ denote its size in vector V. From (P7), we have $|f_u(L_1) - f_u(L_2)| \leq 1$. (Note that because of our construction of the generating vector V, no window can begin with a trailing one of a pair and end at a leading one. Similarly, no window can begin at a leading one and end at a trailing one. This makes application of property (P7) possible here.) From the transformation of U to V, we have $|f_v(L_1) - f_u(L_1)| \leq 1$, and $|f_v(L_2) - f_u(L_2)| \leq 1$. Hence

$$|f_v(L_1) - f_v(L_2)| \leq |f_v(L_1) - f_u(L_1)| + |f_u(L_1) - f_u(L_2)| + |f_u(L_2) - f_v(L_2)| \leq 3.$$

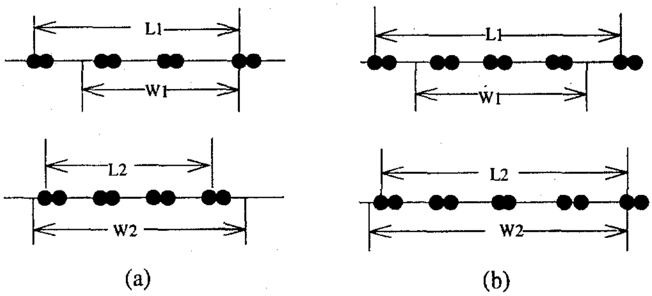


Fig. 4. Two scenarios in the proof of Lemma 8.

Since, $f_v(L_1)$ and $f_v(L_2)$ are both of odd size, we must have either $|L_1| = |L_2|$ or $||L_1| - |L_2|| \leq 2$. $\square$

LEMMA 8. *If* $\gcd(s, k) = 1$, *and if* W_1, W_2 *are two windows of vector* $V(3s, 3k)$ *with* $|W_1| = |W_2|$, *then* $|\#_1(W_1) - \#_1(W_2)| \leq 2$.

PROOF. Without loss of generality, suppose that $\#_1(W_2) - \#_1(W_1) = 3$. Further, it suffices to consider only those windows that have length less than $2s$ and whose centers lie in the middle portion of the V-vector, i.e., positions $s+1, s+2, \ldots, 2s$. This enables the extension of window W_1 in the following.

Case 1: $\#_1(W_1)$ *is odd.* Figure 4(a) shows one such scenario. Let L_1 be a shortest window that includes W_1, starts and ends with a one, and covers two more ones than W_1. Shrink W_2 to obtain a window L_2 that starts and ends with a one and has one less one than W_2. Then we have $|L_1| \geq |W_1| + 2$, and $|L_2| \leq |W_2| - 2$ since we must lose at least one zero along with a one. Hence $|L_1| - |L_2| \geq 4$ which contradicts Lemma 7. Note that because of our assumptions, W_2 cannot have a one on both ends. Since W_2 contains an even number of ones, if it has a one on both ends, then its size would be even (because it would have an even number of zeros also), which contradicts with the fact that windows are of odd size only. Similarly, W_1 cannot have a zero on both ends.

Case 2: $\#_1(W_1)$ *is even.* Figure 4(b) shows one such scenario. Again, extend W_1 to cover one more pair of ones on one end and one more one on the other end to obtain a window L_1 starting and ending with a one and having three more ones than W_1. Shrink W_2 getting rid of the possible zeros on one end to obtain a window L_2 starting and ending with a one. Then we have $|L_2| \leq |W_2|$ and $|L_1| \geq |W_1| + 3$, hence $|L_1| - |L_2| \geq 3$, which also contradicts Lemma 7. $\square$

In summary, for the grid embedding problem, we have obtained a solution by finding the generating vector, generating the square elliptic doubly stochastic matrix, cutting or repeating/cutting into the desired rectangular embedding matrix. It is stated as:

THEOREM 9. *Any rectangular grid* $G = h \times w$ *can be embedded into its ideal rectangular grid* $G' = h' \times w'$ *with dilation no more than* $\bar{\rho} = \lceil w/w' \rceil$.

The complete algorithm for embedding a rectangle into another rectangle is given in the Appendix.

8. Applications. We show that any two-dimensional grid can be embedded into its optimal hypercube with optimal dilation 2.

THEOREM 10. *Any two-dimensional grid can be embedded into its optimal hypercube with optimal dilation* 2.

PROOF. Let the two-dimensional grid be $G = x \times y$. Without loss of generality assume $y < \frac{3}{2}\bar{y}$, where $\bar{y} = 2^{\lfloor \log y \rfloor}$ (for all cases in which neither $x < 3/2\bar{x}$, nor $y < 3/2\bar{y}$ assign the gray code directly). Embed $G = x \times y$ into $G' = x' \times \bar{y}$ using our approach and the embedding function defined in Theorem 2 (G' is one of G's ideal rectangular grids). Assign the gray code to the two dimensions of G', we have an assignment of hypercube addresses for the nodes in the original grid G, and the dilation is obviously 2 (Theorem 9). This dilation is also optimal. The dimension of the hypercube is the summation of the code lengths used in the two dimensions of grid G'. This dimension is the optimal hypercube dimension. □

8.1. *Dilation-3 Embeddings of Rectangles into Nearly Ideal Squares.* We use the folding idea of Aleliunas and Rosenberg [AR] and combine it with our matrix-based approach to show that any rectangular grid can be embedded into its nearly ideal (side $\lceil \sqrt{n} \rceil + 1$) square grid with dilation 3. The folding technique is illustrated in Figure 5. The salient features of folding are:

1. The dilation is exactly 2.
2. If a horizontal edge is dilated it maps onto a path between opposing corners of a unit square in the host grid.
3. If a vertical edge is dilated it maps onto a path comprising two adjacent vertical edges in the host grid.

As observed by Ellis, since the folding operation and compression by no more than a factor of 2 each create dilation 2, combining these operations creates dilation no more than 4. We show that if the folding is followed by a compression with compression ratio

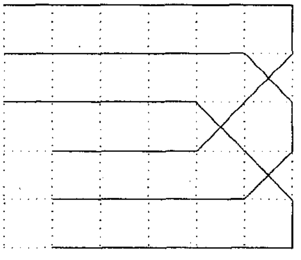


Fig. 5. Folding 3×13 into 6×7.

no more than 1.5, then the dilation is no more than 3. Let the *fold axis* of a fold operation be along the width of the guest grid, we compress along the axis that is orthogonal to the fold axis.

We show that our approach never dilates both horizontal edges of a unit square, and we also show that two adjacent horizontal edges in the guest grid are not dilated simultaneously (note that the compression axis is vertical, so vertical edges in the guest grid are horizontal from the compression point of view). Therefore, if we fold until we get a compression ratio of 1.5 or less, and then compress, the dilation will be at most 3. Let P be the guest grid, Q the grid after folding, and R the grid after compression using our approach.

LEMMA 11. *The compression process from Q to R does not dilate both horizontal edges of any unit square box in Q.*

PROOF. Let the square be $ABDC$, where $A = (i, j)$, $B = (i, j+1)$, $C = (i+1, j)$, and $D = (i+1, j+1)$. Suppose that both AB and CD are dilated by the compression. Let $\varphi(i, j) = (k, l)$, then $\varphi(i, j+1) = (k-1, l+1)$ or $\varphi(i, j+1) = (k+1, l+1)$. If $\varphi(i, j+1) = (k-1, l+1)$, then by using the embedding function we can obtain $\sum_{s=1}^{l} c_{is} = j$ and

$$\begin{cases} \displaystyle\sum_{r=1}^{i-1} c_{rl} - \sum_{r=1}^{i-1} c_{r,l+1} = 1 & \text{if } l \text{ is even,} \\ \displaystyle\sum_{r=1}^{i} c_{rl} - \sum_{r=1}^{i} c_{r,l+1} = 1 & \text{if } l \text{ is odd.} \end{cases}$$

The only possible case that results in the dilation of CD is $\varphi(i+1, j) = (k+1, l)$ and $\varphi(i+1, j+1) = (k, l+1)$. We can obtain $\sum_{s=1}^{l} c_{i+1,s} = j$ and

$$\begin{cases} \displaystyle\sum_{r=1}^{i} c_{rl} - \sum_{r=1}^{i} c_{r,l+1} = 1 & \text{if } l \text{ is even,} \\ \displaystyle\sum_{r=1}^{i+1} c_{rl} - \sum_{r=1}^{i+1} c_{r,l+1} = 1 & \text{if } l \text{ is odd.} \end{cases}$$

We have a contradiction with property (P3) for both even and odd value of l. Case $\varphi(i, j+1) = (k+1, l+1)$ can be verified similarly. □

LEMMA 12. *If the compression ratio is at most 1.5, then the compression process from Q to R does not dilate both edges AB and BC, where $A = (i, j-1)$, $B = (i, j)$ and $C = (i, j+1)$.*

PROOF. See the Appendix. □

THEOREM 13. *Any rectangular grid can be embedded into its nearly ideal square grid with dilation 3.*

9. Conclusion. We have presented a general approach for embedding rectangular grids into rectangular grids and hypercubes. We have also given several applications of our approach. An important issue is whether we can embed rectangular grids into nearly ideal or ideal square grids with dilation 2. Recently, we have shown that any rectangular grid can be embedded into its optimal square grid with dilation 6 [HLV]. It appears possible that our method can be generalized to (one-to-one) embedding higher-dimensional grids into grids or hypercubes. It may be possible to improve the work of [KA2] on many–one d-dimensional simulations of grids into grids by using our better result for two dimensions.

Acknowledgments. The authors would like to thank the referee for detailed comments and suggestions, which have significantly improved and clarified the paper.

Appendix A.

THEOREM 5. *Any $h \times h$ elliptic doubly stochastic matrix $C = (c_{ij})$ defines an embedding from $G = h \times w$ into $G' = w \times h$ with dilation at most $\bar{\rho}$, where $w = \sum_{j=1}^{h} c_{ij}$ is the row sum and $\bar{\rho} = \lceil w/h \rceil$ is the ceiling of the compression ratio.*

PROOF. From the properties (P1)–(P4) we can derive the properties (C1)–(C3). Hence, an elliptic doubly stochastic matrix is also an embedding matrix. We prove the dilation conclusion as follows: Consider two adjacent nodes (i, j) and (i', j') in G, they are either horizontally adjacent or vertically adjacent. We prove, by case analysis using the definition of φ and matrix properties, that their images $\varphi(i, j)$ and $\varphi(i', j')$ are at most $\bar{\rho}$ edges away in G'. Let $\delta = |k - k'| + |l - l'|$.

Case A. Nodes are horizontally adjacent, i.e., $(i', j') = (i, j + 1)$. Let $\varphi(i, j) = (k, l)$ and $\varphi(i, j + 1) = (k', l')$. It is easy to see that $0 \le l' - l \le 1$.

(a) If $l = l'$, then $\delta = 1 \le \bar{\rho}$. Because of the way φ is defined, if two horizontally adjacent nodes are mapped to the same column, then they are vertically adjacent in the host grid.

(b) If $l' = l + 1$, then $\sum_{s=1}^{l} c_{is} = j$ (otherwise, if $\sum_{s=1}^{l} c_{is} > j$, then $\sum_{s=1}^{l} c_{is} \ge j + 1$, hence $l' \le l$) and

$$k = \sum_{r=1}^{i-1} c_{rl} + \begin{cases} c_{il} & \text{for } l \text{ odd,} \\ 1 & \text{for } l \text{ even,} \end{cases}$$

$$\begin{aligned} k' &= \sum_{r=1}^{i-1} c_{r,l+1} + \begin{cases} j + 1 - \displaystyle\sum_{s=1}^{l} c_{is} & \text{for } l' \text{ odd, i.e., } l \text{ even,} \\ 1 + \displaystyle\sum_{s=1}^{l+1} c_{is} - j - 1 & \text{for } l' \text{ even, i.e., } l \text{ odd,} \end{cases} \\ &= \sum_{r=1}^{i-1} c_{r,l+1} + \begin{cases} 1 & \text{for } l \text{ even,} \\ c_{i,l+1} & \text{for } l \text{ odd.} \end{cases} \end{aligned}$$

By (P1) and (P2), $|k - k'| \le 1$, hence $\delta \le 2 \le \bar{\rho}$.

Case B. Nodes are vertically adjacent, i.e., $(i', j') = (i + 1, j)$. Let $\varphi(i, j) = (k, l)$ and $\varphi(i + 1, j) = (k', l')$. It is easy to see that $-1 \le l' - l \le 1$ (P2).

(a) If $l' = l$, then

$$k' = \sum_{r=1}^{i} c_{rl} + \begin{cases} j - \sum_{s=1}^{l-1} c_{i+1,s} & \text{for } l \text{ odd,} \\ 1 + \sum_{s=1}^{l} c_{i+1,s} - j & \text{for } l \text{ even.} \end{cases}$$

$$\begin{aligned} k' - k &= c_{il} + \begin{cases} \sum_{s=1}^{l-1} c_{is} - \sum_{s=1}^{l-1} c_{i+1,s} & \text{for } l \text{ odd,} \\ \sum_{s=1}^{l} c_{i+1,s} - \sum_{s=1}^{l} c_{is} & \text{for } l \text{ even,} \end{cases} \\ &\le \bar{\rho}. \end{aligned}$$

The reasoning for the last step above is as follows. If $c_{il} = \underline{\rho}$, then using (P2) we have $|k' - k| \le \underline{\rho} + 1 = \bar{\rho}$. If $c_{il} = \bar{\rho}$, then both of the difference terms in the above rewrite are 0 if $c_{i+1,l} = \bar{\rho}$ (P3); or no greater than 0 if $c_{i+1,l} = \underline{\rho}$. Thus $k - k' \le \bar{\rho}$.

(b) If $l' = l + 1$, then $\sum_{s=1}^{l} c_{is} = j$ and $\sum_{s=1}^{l} c_{i+1,s} = j - 1$ (similar reasoning as in subcase (b) of Case A using (P2)). Further, we must have $c_{i,l} = c_{i+1,l+1} = \bar{\rho}$, and upon applying (P3) we also get that $c_{i+1,l} = c_{i,l+1} = \underline{\rho}$ (see Figure 6(a)).

$$k = \sum_{r=1}^{i-1} c_{rl} + \begin{cases} c_{il} & \text{for } l \text{ odd,} \\ 1 & \text{for } l \text{ even,} \end{cases}$$

$$\begin{aligned} k' &= \sum_{r=1}^{i} c_{r,l+1} + \begin{cases} j - \sum_{s=1}^{l} c_{i+1,s} & \text{for } l \text{ even,} \\ 1 + \sum_{s=1}^{l+1} c_{i+1,s} - j & \text{for } l \text{ odd,} \end{cases} \\ &= \sum_{r=1}^{i} c_{r,l+1} + \begin{cases} 1 & \text{for } l \text{ even,} \\ c_{i+1,l+1} & \text{for } l \text{ odd,} \end{cases} \end{aligned}$$

$$\begin{aligned} k' - k &= \sum_{r=1}^{i} c_{r,l+1} - \sum_{r=1}^{i-1} c_{rl} + \begin{cases} c_{i+1,l+1} - c_{il} & \text{for } l \text{ odd,} \\ 0 & \text{for } l \text{ even,} \end{cases} \\ &= \sum_{r=1}^{i} c_{r,l+1} - \sum_{r=1}^{i-1} c_{rl} + \begin{cases} 0 & \text{for } l \text{ odd,} \\ 0 & \text{for } l \text{ even,} \end{cases} \\ &= c_{i,l+1} \qquad \text{(P5)} \\ &= \underline{\rho}. \end{aligned}$$

Hence $\delta = \bar{\rho}$.

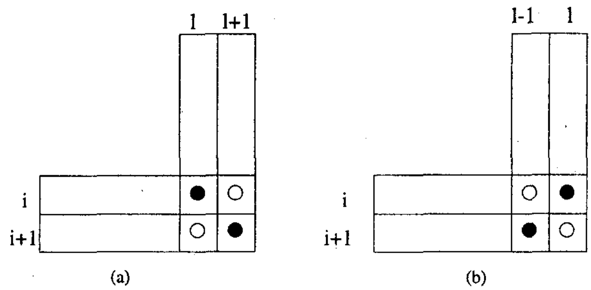


Fig. 6. Two scenarios in the proof of Theorem 5.

(c) Similar to (b). If $l' = l - 1$, then $\sum_{s=1}^{l-1} c_{i+1,s} = j$, $\sum_{s=1}^{l-1} c_{is} = j - 1$, and we must have $c_{i,l-1} = c_{i+1,l} = \underline{\rho}$, $c_{i,l} = c_{i+1,l-1} = \bar{\rho}$ (see Figure 6(b)).

$$
\begin{aligned}
k &= \sum_{r=1}^{i-1} c_{rl} + \begin{cases} 1 & \text{for } l \text{ odd}, \\ c_{il} & \text{for } l \text{ even}, \end{cases} \\
k' &= \sum_{r=1}^{i} c_{r,l-1} + \begin{cases} j - \displaystyle\sum_{s=1}^{l-2} c_{i+1,s} & \text{for } l \text{ even}, \\ 1 + \displaystyle\sum_{s=1}^{l-1} c_{i+1,s} - j & \text{for } l \text{ odd}, \end{cases} \\
&= \sum_{r=1}^{i} c_{r,l-1} + \begin{cases} c_{i+1,l-1} & \text{for } l \text{ even}, \\ 1 & \text{for } l \text{ odd}, \end{cases} \\
k' - k &= \sum_{r=1}^{i} c_{r,l-1} - \sum_{r=1}^{i-1} c_{rl} + \begin{cases} 0 & \text{for } l \text{ odd}, \\ 0 & \text{for } l \text{ even}, \end{cases} \\
&= c_{i,l-1} \qquad \text{(P5)} \\
&= \underline{\rho}.
\end{aligned}
$$

Hence $\delta = \bar{\rho}$. □

LEMMA 12. *If the compression ratio is at most* 1.5, *the compression process from Q to R does not dilate both edges AB and BC, where $A = (i, j - 1)$, $B = (i, j)$, and $C = (i, j + 1)$.*

Proof sketch. Suppose both edges AB and BC are dilated, $\varphi(A)$, $\varphi(B)$, and $\varphi(C)$ must be in three consecutive columns of R. Let these columns be $l-1$, l, and $l+1$. $\varphi(A)$ and $\varphi(B)$ must also in adjacent rows. There are four cases to be considered.

Case 1: $\varphi(i, j-1) = (k+1, l-1)$, $\varphi(i, j) = (k, l)$, $\varphi(i, j+1) = (k-1, l+1)$. By using the definition of φ and properties (P1) and (P2), we have $c_{il} = 1$, $\sum_{s=1}^{l-1} c_{is} = j-1$,

$\sum_{s=1}^{l} c_{is} = j$, and $\sum_{s=1}^{l+1} c_{is} \geq j+1$. Hence

$$(1) \qquad k+1 = \sum_{r=1}^{i-1} c_{r,l-1} + \begin{cases} c_{i,l-1} & \text{if } l \text{ is even,} \\ 1 & \text{if } l \text{ is odd.} \end{cases}$$

$$(2) \qquad k = \sum_{r=1}^{i-1} c_{rl} + \begin{cases} 1 & \text{if } l \text{ is odd,} \\ 1 & \text{if } l \text{ is even.} \end{cases}$$

$$(3) \qquad k-1 = \sum_{r=1}^{i-1} c_{r,l+1} + \begin{cases} 1 & \text{if } l \text{ is even,} \\ c_{i,l+1} & \text{if } l \text{ is odd.} \end{cases}$$

From (1) and (2) we have

$$\begin{matrix} \text{(a)} \\ \\ \text{(b)} \end{matrix} \qquad \begin{cases} \displaystyle\sum_{r=1}^{i-1} c_{r,l-1} - \sum_{r=1}^{i-1} c_{rl} = 1 & \text{if } l \text{ is odd,} \\ \displaystyle\sum_{r=1}^{i} c_{r,l-1} - \sum_{r=1}^{i} c_{rl} = 1 & \text{if } l \text{ is even.} \end{cases}$$

From (2) and (3) we have

$$\begin{matrix} \text{(c)} \\ \\ \text{(d)} \end{matrix} \qquad \begin{cases} \displaystyle\sum_{r=1}^{i-1} c_{rl} - \sum_{r=1}^{i-1} c_{r,l+1} = 1 & \text{if } l \text{ is even,} \\ \displaystyle\sum_{r=1}^{i} c_{rl} - \sum_{r=1}^{i} c_{r,l+1} = 1 & \text{if } l \text{ is odd.} \end{cases}$$

Denote (a) by $s_1 - s_2 = 1$ for odd l, and (c) by $s_2 - s_3 = 1$ for even l. If l is odd, then (a) holds, and (d) implies that $(s_2+1)-(s_3+c_{i,l+1}) = 1$, i.e., $s_2 - s_3 = c_{i,l+1}$, hence $s_1 - s_3 = 1 + c_{i,l+1} \geq 2$ contradicts (P6). If l is even, from (b), $(s_1 + c_{i,l-1}) - (s_2+1) = 1$, then $s_1 - s_2 = 2 - c_{i,l-1}$, since $c_{il} = 1$, we must have $c_{i,l-1} = 2$ and $s_1 = s_2$ by (P3). From (c), $s_2 - s_3 = 1$, then $s_1 - s_3 = 3 - c_{i,l-1}$ thus $s_1 = s_3 + 1$, therefore $s_2 = s_3 + 1$, which contradicts $s_2 = s_3$. The reason for $s_2 = s_3$ is $c_{il} = c_{i,l+1} = 1$ and (P3). Otherwise if $c_{i,l+1} = 2$, then trace back this two and $c_{i,l-1} = 2$ to the original elements in the first row (the generating vector) of the base matrix. If the original two elements move in different directions, they should have odd distance (even gap), but $c_{i,l-1}$ and $c_{i,l+1}$ have even distance (odd gap), a contradiction. If the two original elements move in the same direction, then there must be exactly another two between them, but for a compression ratio less than 1.5, there are no three consecutive twos in the generating vector. It is also not possible to have two twos in the first two positions in the base vector because of the way it is generated.

Case 2: $\varphi(i, j-1) = (k-1, l-1), \varphi(i, j) = (k, l), \varphi(i, j+1) = (k+1, l+1)$. This case is similar to Case 1.

The other two cases do not occur (even if they occur, the dilation between A and C is 2). □

A.1. *General Algorithm for Rectangle Embedding.*

NOTATION. M^h denotes matrix M reflected along the horizontal axis, M^v denotes matrix M reflected along the vertical axis, and M^τ denotes the transpose of matrix M.

Algorithm for embedding a rectangle into its optimal rectangle

/* Embeds $h \times w$ into $h' \times w'$ where $w' < w$ and $h' = \lceil hw/w' \rceil$*/

- Let $a = \lfloor w/w' \rfloor$—the small entry in the embedding matrix,
  $b = \lceil w/w' \rceil$—the big entry in the embedding matrix,
  $r = w \bmod w'$—the number of big entries in each row or column of the matrix,
  $d = \gcd(w', r)$—the greatest common divisor of w' and r,
  $s = w'|d$—w' divided by the gcd, make s relatively prime to k defined next,
  $k = r|d$—r divided by the gcd,
  $z = \lceil h/s \rceil$—the number of vertical repititions needed later.

Assume $k < s/2$, otherwise let $k = s - k$ and flip a and b later in the embedding matrix.

1. Generate vector $U(s, k) = (u_1, u_2, \ldots, u_s)$ as follows:
   If k is even, define
   $$P_0 = \left\{ \left\lfloor \frac{(2i-1)s}{k} \right\rfloor \mid i = 1, 2, \ldots, \frac{k}{2} \right\},$$
   $$P_1 = \{p + 1 \mid p \in P_0\}, \qquad P = P_0 \cup P_1.$$
   If k is odd, define
   $$P_0 = \left\{ \left\lfloor \frac{(2i)s}{k} \right\rfloor \mid i = 0, 1, \ldots, \frac{k-1}{2} \right\},$$
   $$P_1 = \{p + 1 \mid p \in P_0\}, \qquad P = P_0 \cup P_1.$$
   For $0 < i \leq s$, let $u_i = b$ **if** $i \in P$, and a **otherwise**.
2. Adjust $U(s, k)$ into $V(s, k) = (v_1, v_2, \ldots, v_s)$ so that it becomes even gapped: From left to right, **if** the number of a's between two successive pairs of b's is odd, **then** swap the first b of the second pair with the a immediately following this pair. Define $P'_0 = \{i \mid v_i = v_{i+1} = b\}$.
3. Generate $s \times s$ matrix $C = (c_{ij})$ (initially $c_{ij} = a$ for all i, j):
   If $v_1 = b$, **then** set $c_{ii} = b$ for $1 \leq i \leq s$.
   $\forall\, p \in P'_0$, set $c_{i,p+1-i} = c_{s-p+i,s-i+1} = b$ for $1 \leq i \leq p$,
   set $c_{p+j,j} = c_{j,p+j} = b$ for $1 \leq j \leq s - p$.
4. Horizontal expansion: $D = (C_1, C_2, \ldots, C_d)$, where $C_{2i} = C^v$, $C_{2i-1} = C$.
5. Vertical expansion: $E = (D_1, D_2, \ldots, D_z)^\tau$, where $D_{2j} = D^h$, $D_{2j-1} = D$.
6. Use the first h rows of $E_{zs \times w'}$ to define the embedding function φ from $h \times w$ into $h' \times w'$: $\varphi(i, j) = (k, l)$ where

$$l = \min\left\{ \lambda \colon \sum_{s=1}^{\lambda} c_{is} \geq j \right\}, \qquad k = \sum_{r=1}^{i-1} c_{rl} + \begin{cases} j - \displaystyle\sum_{s=1}^{l-1} c_{is} & \text{if } l \text{ is odd,} \\ 1 + \displaystyle\sum_{s=1}^{l} c_{is} - j & \text{if } l \text{ is even.} \end{cases}$$

References

[AR] R. Aleliunas and A. L. Rosenberg, On Embedding Rectangular Grids in Square Grids, *IEEE Trans. Comput.*, Vol. 31, No. 9, pp. 907–913, 1982.

[BCLR1] S. Bhatt, F. Chung, T. Leighton, and A. L. Rosenberg, Optimal Simulations of Tree Machines, *Proc. 27th Symp. on Foundations of Computer Science*, pp. 274–282, 1986.

[BCLR2] S. Bhatt, F. Chung, T. Leighton, and A. L. Rosenberg, Efficient Embeddings of Trees in Hypercubes, *SIAM J. Comput.*, Vol. 21, No. 1, pp. 151–162, February 1992.

[C1] M. Y. Chan, Dilation-2 Embeddings of Grids into Hypercubes, *Proc. Internat. Conf. on Parallel Processing*, Vol. 3, pp. 295–298, 1988.

[C2] M. Y. Chan, Embedding of *D*-dimensional Grids into Optimal Hypercubes, *Proc. Symp. on Parallel Algorithms and Architectures*, pp. 52–57, 1989.

[C3] M. Y. Chan, Embedding Grids into Optimal Hypercubes, *SIAM J. Comput.*, Vol. 20, No. 5, pp. 833–864, October 1991.

[CC] M. Y. Chan and F. Y. L. Chin, On Embedding Rectangular Grids in Hypercubes, *IEEE Trans. Comput.*, Vol. 37, No. 10, pp. 1285–1288, 1988.

[DEL] R. A. DeMillo, S. C. Eisenstat, and R. J. Lipton, Preserving Average Proximity in Arrays, *Comm. ACM*, Vol. 21, pp. 228–231, 1978.

[E] J. A. Ellis, Embedding Rectangular Grids into Square Grids, *IEEE Trans. Comput.*, Vol. 40, No. 1, pp. 46–52, 1991.

[HLV] S. H. S. Huang, H. Liu, and R. M. Verma, On Embedding Rectangles into Optimal Squares, *Proc. 22nd Internat. Conf. on Parallel Processing*, Vol. III, pp. 73–76, 1993.

[KA1] S. R. Kosaraju and M. J. Atallah, Optimal Simulations Between Mesh-Connected Arrays of Processors, *Proc. ACM Symp. on Theory of Computing*, pp. 264–272, 1986.

[KA2] S. R. Kosaraju and M. J. Atallah, Optimal Simulations Between Mesh-Connected Arrays of Processors, *J. Assoc. Comput. Mach.*, Vol. 35, pp. 635–650, 1988.

[L] C. E. Leiserson, Area-Efficient Graph Layouts (for VLSI), *Proc. 21st IEEE Symp. on Foundations of Computer Science*, pp. 270–281, 1980.

[LED] R. J. Lipton, S. C. Eisenstat, and R. A. DeMillo, Space and Time Hierarchies for Classes of Control Structures and Data Structures, *J. Assoc. Comput. Mach.*, Vol. 23, pp. 720–732, 1976.

[LSS] F. Lombardi, D. Sciuto, and R. Stefanelli, An Algorithm for Functional Reconfiguration of Fixed-Size Arrays, *IEEE Trans. Comput. Aided Design*, Vol. 10, pp. 1114–1118, 1988.

[RS] A. L. Rosenberg and L. Snyder, Bounds on the Costs of Data Encoding, *Math. Systems Theory*, Vol. 12, pp. 9–39, 1978.

[R] A. L. Rosenberg, Encoding Data Structures in Trees, *J. Assoc. Comput. Mach.*, Vol. 26, pp. 668–689, 1979.