SIAM J. NUMER. ANAL.
Vol. 24, No. 6, December 1987

011

DETERMINATION OF STRIPE STRUCTURES FOR FINITE ELEMENT MATRICES*

RAMI MELHEM†

Dedicated to Werner C. Rheinboldt on the occasion of his 60th birthday.

Abstract. Stripe structures were introduced by the author [Parallel Computing, to appear] as a means for the inclusion of the nonzero elements of a sparse matrix into a regular pattern which allows for efficient parallel manipulation. In this paper the stripe structures of stiffness matrices resulting from irregular domains covered by regular grids are analyzed. It is proved that the nonzero elements in these matrices may be covered by very few stripes and that these stripes may be nonoverlapping if the nodes of the grids are numbered appropriately. The exact number of stripes, which is independent of the size of the problem, is derived for different types of grids and different numbering schemes. The stripe structures of some irregular grids are also examined.

Key words. stripe structures, sparse matrices, finite elements, numbering schemes, computational networks

AMS(MOS) subject classification. 65

1. Introduction. Many techniques have been suggested for the efficient solution of sparse linear systems; they involve often highly irregular storage schemes and manipulation algorithms for the nonzero elements in the matrix [4]. Although these techniques lead to very powerful sequential implementations (see e.g. [3]), they are not very suitable for parallel architectures. In fact, rather regular patterns of computations are favored in parallel processing in order to minimize data conflict and communication delays.

In a previous paper [7], a method was introduced for representing all nonzero elements of a sparse matrix in a stripe structure that provides, in some sense, a compromise between efficiency and regularity. More specifically, the stripe structure is shown to possess enough regularity to allow for the design of some efficient VLSI networks for the parallel manipulation of sparse matrices. Two networks, namely MAT/VEC for the multiplication of a vector by a matrix, and TRIANG, for the solution of triangular systems, are given as examples.

Very briefly, a stripe S_k of an $n \times n$ matrix A is a set of positions that contains at most one position, (i, j), of A for every row i; that is, $S_k = \{(i, \sigma_k(i)): i \in I \subseteq I_n\}$, where $I_n = \{1, \cdots, n\}$, and σ_k is a strictly increasing function. If S_k contains one position for each row of A, then S_k is called a complete stripe. Two stripes S_k and S_q are ordered by $S_k < S_q$ if, for any i and j in the domains of σ_k and σ_q, respectively,

$$i \leqq j \text{ implies } \sigma_k(i) < \sigma_q(j). \tag{1}$$

A stripe structure, Σ_A, of the matrix A is then defined as a disjoint union of stripes S_k, $k = 1, \cdots, \pi$, which satisfies $S_k < S_{k+1}$, and contains all the nonzero elements of A. More specifically if $a_{i,j} \neq 0$, then there should exist a unique k such that $(i, j) \in S_k$. The number π of stripes is called the stripe count of Σ_A. Also, the stripes $S_1, \cdots, S_\pi$

* Received by the editors February 10, 1986; accepted for publication October 14, 1986. This work was supported in part by the Office of Naval Research under contract N00014-85-K-0339.

† Department of Computer Science, University of Pittsburgh, Pittsburgh, Pennsylvania 15260.

are said to be nonoverlapping if

$$\sigma_k(i) \leqq \sigma_{k+m}(i-m) \tag{2}$$

for any integers k, i and m such that $(i, \sigma_k(i)) \in S_k$ and $(i-m, \sigma_{k+m}(i-m)) \in S_{k+m}$. If the inequality in (2) is strict, then the stripes are called strictly nonoverlapping.

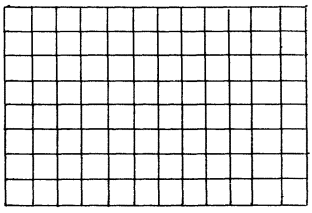
The networks MAT/VEC and TRIANG suggested in [7] are linear networks in which one cell is assigned to perform the computations associated with one stripe of the input matrix. The correctness of the operation of the networks is proved by utilizing the property that stripes are strictly increasing. Also, a performance analysis, which is based on the computation front concept [6], shows that data conflict among the cells of the networks may be eliminated if the input matrices have nonoverlapping stripes. Moreover, the number of fast communication buffers required in each cell may be minimized by minimizing the maximum separation between the stripes of the input matrices. In other words, the peak performance of the networks is obtained for matrices which have a stripe structure with (1) nonoverlapping stripes, and (2) uniform distribution of stripes within the band of the matrix.

In this paper, we consider a major source of large sparse matrices, namely, finite element and finite difference discretizations of partial differential equations (PDE). More specifically, we study the stripe structure of stiffness matrices that result from discretizations on irregular domains using regular grids. First, we specify in § 2 the types of domains and grids used in the study. Then, in § 3, we show that for matrices resulting from these types of grids, a stripe structure with very few stripes may be introduced, but the resulting stripes do, in general, overlap. In order to obtain nonoverlapping stripes, in § 4 we suggest a multicolor numbering scheme that spreads the stripes within a matrix, and thus disengages any overlap between stripes. The multicolor numbering is also shown to decrease the maximum separation between stripes. Finally, in § 5, we briefly review the network MAT/VEC and estimate its execution time for some specific stiffness matrices.

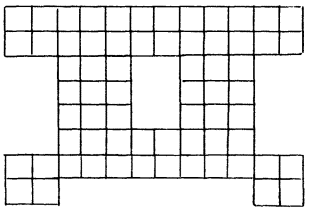
2. Pierced rectangular domains. Let Q be a rectangular domain that is covered by a grid M_Q with lines parallel to the sides of Q. If we remove from Q any number of rectangular subdomains whose boundaries coincide with some lines in M_Q, then we obtain a new domain $\Omega \subset Q$, which we will call a pierced rectangular domain. The part of M_Q that covers Ω is denoted by M_Ω and is called a pierced rectangular grid (see Fig. 1(b)).

If D is an irregular domain, we may approximate D by a pierced rectangular domain and cover it by a pierced rectangular grid (see Fig. 1(c)). Another alternative (usually used in automatic grid generation) is to map D, isoparametrically [10], into a pierced rectangular domain Ω, cover Ω by a pierced rectangular grid M_Ω, and then map M_Ω back to a grid M_D that covers D (see Fig. 1(d)). In this case, the zero pattern of the stiffness matrix that results from the discretization of a PDE on M_D is the same as that resulting from the discretization of the PDE on M_Ω. For this reason, we consider here only discretizations on pierced rectangular grids.

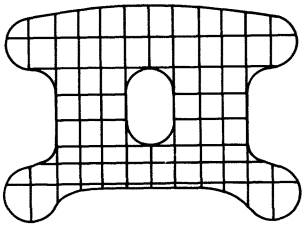
Given a pierced rectangular domain Ω covered by a grid M_Ω which contains n_Ω nodes, let M_Q be a rectangular grid that includes M_Ω and contains n_Q nodes, $n_Q > n_\Omega$. Each node in M_Q may be identified by a unique number λ, $1 \leqq \lambda \leqq n_Q$, assigned to it by some numbering scheme (Greek letters will be used to identify nodes in M_Q). On the other hand, the nodes in M_Ω may be renumbered such that each node λ which is also in M_Ω is assigned a unique number $l = \nu(\lambda)$, $1 \leqq l \leqq n_\Omega$.



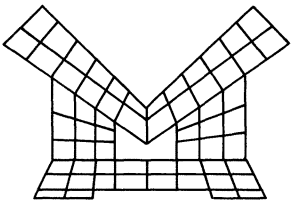
(a) *A rectangular grid* M_Q.



(b) *A pierced rectangular grid* M_Ω.



(c) *An irregular domain covered by* M_Ω.



(d) *An irregular grid* M_D *isomorphic to* M_Ω.

FIG. 1. *Examples of finite element grids.*

DEFINITION 1. A renumbering of M_Ω is said to be deduced from the numbering of M_Q if the number, $l = \nu(\lambda)$, assigned to any node $\lambda \in M_\Omega$ is derived as follows:

$l = 0$
FOR $\lambda = 1, \cdots, n_Q$ DO
  IF $\lambda \in M_\Omega$ THEN $\{l = l+1: \nu(\lambda) = l\}$
    ELSE $\{\nu(\lambda)$ is undefined$\}$

Clearly, the renumbering function ν satisfies the following relations:

$$\lambda > \mu \Leftrightarrow \nu(\lambda) > \nu(\mu), \tag{3a}$$

$$\lambda > \mu \Rightarrow \lambda - \mu \geqq \nu(\lambda) - \nu(\mu). \tag{3b}$$

The inverse ν^{-1} of the function ν will be used to map the number l of any node in M_Ω into its identity $\lambda = \nu^{-1}(l)$ in M_Q. It is also useful to define a function which determines, for each node $\delta \notin M_\Omega$, the smallest node larger than δ that is in M_Ω. For uniformity, we define such a function for any $\delta \in M_Q$ as follows:

DEFINITION 2. The function Next $(\delta): M_Q \to M_Q$ is defined by:

$$\text{Next}(\delta) = \begin{cases} \min\{\mu: \mu \geqq \delta \text{ and } \mu \in M_\Omega\} & \text{if such } \mu \text{ exists,} \\ n_Q & \text{otherwise.} \end{cases}$$

Note that the minimum does not exist if no node $\mu \geqq \delta$ is in M_Ω, which may happen only if $\delta > \nu^{-1}(n_\Omega)$.

Without entering into the details of the generation of stiffness matrices, we just mention that the matrix A generated from the discretization of a PDE on M_Ω is an $n_\Omega \times n_\Omega$ matrix in which each row l corresponds to a node $\lambda = \nu^{-1}(l)$ in M_Ω. The only nonzero elements in row l of A are those at positions (l, m), where $\mu = \nu^{-1}(m)$ is a node that is a neighbor to node λ in M_Ω. The definition of neighboring nodes depends

on the specific discretization used. For example, in finite element discretizations, two nodes are neighbors if there exists an element that contains the two nodes.

From the above discussion, it is clear that the scheme used to number the nodes determines the zero structure of the matrix A. In the following sections we consider two different numbering schemes. For ease of reference, we refer to the 5-point finite difference discretization by FD_5, and to finite element discretizations with 3-node triangles, 6-node triangles, 4-node rectangles and 9-node rectangles by FE_3, FE_6, FE_4 and FE_9, respectively.

3. Regular node numbering. A regular node numbering is one in which the nodes are numbered sequentially, column-wise or row-wise. We will consider only column-wise numbering and note that our results apply to row-wise numberings, as well.

Let M_Q contain H horizontal lines and W vertical lines, and identify each node in M_Q by the number assigned to it by the column-wise numbering of M_Q, that is, identify the node located at the intersection of the ith horizontal line and the jth vertical line of M_Q by the integer $(j-1)H+i$. It is easy to see that the column-wise numbering of M_Ω is the one deduced from the above numbering of M_Q. Let ν be the renumbering function introduced in Definition 1.

Depending on the specific discretization, we may introduce a few functions that define the neighbors of each node λ in M_Q. For example, for FE_4 discretization, the following nine neighboring functions may be defined for each $\lambda \in M_Q$ (see Fig. 2):

$$
\begin{aligned}
&\rho_{-4}(\lambda) = \lambda - H - 1, && \rho_4(\lambda) = \lambda + H + 1,\\
&\rho_{-3}(\lambda) = \lambda - H, && \rho_3(\lambda) = \lambda + H,\\
&\rho_{-2}(\lambda) = \lambda - H + 1, && \rho_2(\lambda) = \lambda + H - 1,\\
&\rho_{-1}(\lambda) = \lambda - 1, && \rho_1(\lambda) = \lambda + 1,\\
&\rho_0(\lambda) = \lambda.
\end{aligned} \tag{4}
$$

Similar neighboring functions may be defined for other discretizations, and then used to determine the stripe structure of the corresponding matrix as illustrated by the following theorem.

THEOREM 1. *Let the numbering of M_Ω be deduced from that of M_Q, and let A be the stiffness matrix which results from a specific discretization of a* PDE *on M_Ω (with a specific definition of neighboring nodes). Assume that there exist π functions $\rho_k : M_Q \to M_Q$, $k = 1, \cdots, \pi$ such that for any two neighboring nodes λ and μ in M_Q, $\rho_k(\lambda) = \mu$, for some k, $1 \leqq k \leqq \pi$, and the functions ρ_k satisfy*

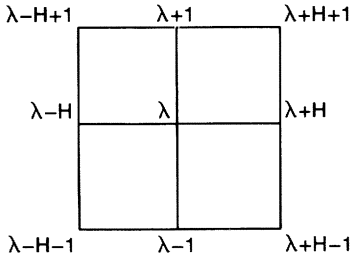


FIG. 2. *The neighbors of node λ.*

$$\rho_k(\lambda) < \rho_k(\lambda+1), \qquad k = 1, \cdots, \pi, \tag{5a}$$

$$\rho_k(\lambda) < \rho_{k+1}(\lambda), \qquad k = 1, \cdots, \pi-1. \tag{5b}$$

Then it is possible to construct a stripe structure for A with stripe count π.

Proof. Define, for $k = 1, \cdots, \pi$, the following sets:

$$S_k = \{(l, \sigma_k(l)); 1 \leqq l \leqq n_\Omega\} \tag{6}$$

where

$$\sigma_k(l) = \begin{cases} \nu(\rho_k(\nu^{-1}(l))) & \text{if } \rho_k(\nu^{-1}(l)) \in M_\Omega, \\ \text{not defined} & \text{otherwise.} \end{cases}$$

It is readily seen that if the (l, m)th element of A is nonzero, then nodes $\nu^{-1}(l)$ and $\nu^{-1}(m)$ are neighbors, and there exists a k such that $\nu^{-1}(m) = \rho_k(\nu^{-1}(l))$. Thus, $(l, m) \in S_k$. In other words, every position of A that has a nonzero element is in some set S_k, $1 \leqq k \leqq \pi$.

In order to prove that each set S_k, $1 \leqq k \leqq \pi$, is a stripe, we consider any two elements $(l, \sigma_k(l))$ and $(m, \sigma_k(m))$ in S_k. If $l = \nu(\lambda)$ and $m = \nu(\mu)$, then by the definition of σ_k, both $\rho_k(\lambda)$ and $\rho_k(\mu)$ are in M_Ω, and hence both $\nu(\rho_k(\lambda))$ and $\nu(\rho_k(\mu))$ are defined. Now if $l > m$, then from (3a), $\lambda > \mu$ and from (5a), $\rho_k(\lambda) > \rho_k(\mu)$. Thus $\nu(\rho_k(\lambda)) > \nu(\rho_k(\mu))$, that is $\sigma_k(l) > \sigma_k(m)$, which proves that σ_k is a strictly increasing function and that S_k is a stripe.

Finally, we need to show that $S_k < S_{k+1}$. For this, we consider the two elements $(l, \sigma_k(l)) \in S_k$, and $(m, \sigma_{k+1}(m)) \in S_{k+1}$. Following the same steps as above, we may show that if $l \geqq m$, then $\lambda \geqq \mu$ and $\rho_{k+1}(\lambda) \geqq \rho_{k+1}(\mu)$. But from (5b), $\rho_{k+1}(\mu) > \rho_k(\mu)$, which leads to $\sigma_{k+1}(l) > \sigma_k(m)$. □

Note that the above theorem does not depend on the specific numbering of M_Q. For column-wise numbering, the functions (4) may be used (assuming $H > 2$) to prove the following corollary.

COROLLARY 1. *If the nodes in a pierced rectangular grid* M_Ω *are numbered column-wise, then the matrix that results from* FE_4 *on* M_Ω *may be striped with nine stripes.*

Given that the matrix resulting from FE_4 on M_Q has nine parallel stripes, Corollary 1 proves that piercing M_Q and renumbering the nodes do not change the stripe count of the matrix (however, the stripes are no longer parallel).

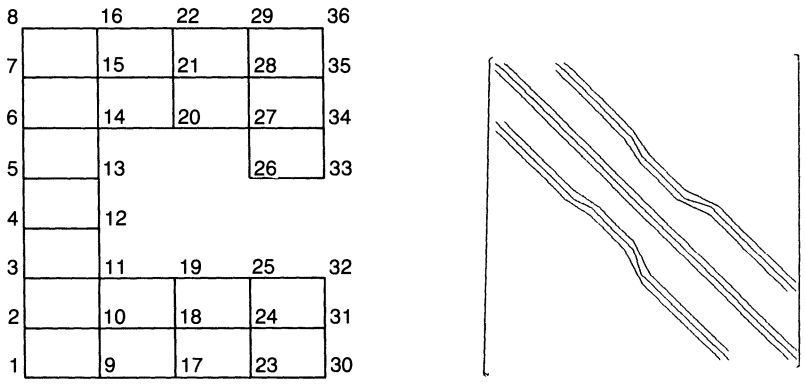
Results similar to Corollary 1 may be proved for other discretizations (see Table 1 for a summary). Although these results indicate that the networks MAT/VEC and TRIANG may be used with the corresponding stiffness matrix, they do not guarantee that the stripes of the matrix are nonoverlapping, and thus, that the operation of MAT/VEC or TRIANG is not delayed due to internal data conflict. For example, the matrix shown in Fig. 3(b), which has overlapping stripes, is obtained from the column-wise numbering of the pierced rectangular domain shown in Fig. 3(a).

4. Multi-color node numbering. Many multicolor numbering schemes have been used by different authors to obtain reorderings of stiffness matrices that have some

TABLE 1
Stripe count for different numbering schemes.

	FD_5	FE_3	FE_4	FE_6	FE_9
regular	5	7	9	19	25
3-color	7(NO)	9(NO)	11(NO)	23	29
5-color	×	×	×	23(NO)	29(NO)

* (NO) = Not overlapping if Lemma 2 applies.



(a) *The grid.* (b) *The FE_4 matrix.*

FIG. 3. *Column-wise numbering.*

desirable properties (see e.g. [1], [8], [9]). In this section, we introduce a multicolor scheme that spreads apart the stripes of A such that they do not overlap. We consider only 3-color numbering, and we assume that $H = 3h - 1$, for some integer h. This may be satisfied, always, by increasing the height of M_Q appropriately.

In order to explain the 3-color numbering scheme, we assume that each horizontal line in M_Q is given a color. Namely, lines $1, 4, \cdots, 3h-2$ are white, lines $2, 5, \cdots, 3h-1$ are black, and lines $3, 6, \cdots, 3h-3$ are red. Numbers are then assigned to the nodes in M_Q as follows:

FOR each column $j = 1, \cdots, W$ DO

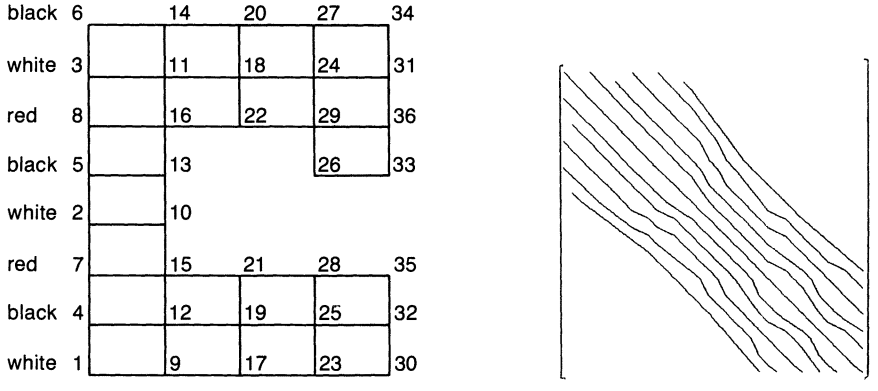
(1) number the nodes in the white lines of column j,

(2) number the nodes in the black lines of column j,

(3) number the nodes in the red lines of column j.

The 3-color numbering of the nodes of M_Ω is then the numbering deduced from the 3-color numbering of M_Q. As an example, we show in Fig. 4(a) the 3-color numbering of the same grid of Fig. 3(a).

As we did for regular numbering, we assume that FE_4 discretization is used and we introduce the appropriate neighboring functions in M_Q. However, in this case, the



(a) *The grid.* (b) *The FE_4 matrix.*

FIG. 4. *3-colors numbering scheme.*

neighbors of a given node λ depend on the color of λ. In order to be more specific, we show in Fig. 5 the numbers that are assigned by the 3-color numbering to the nine neighbors of λ. Clearly, at least eleven functions are needed in order to include all the neighbors, namely:

$$
\begin{aligned}
&\rho_5(\lambda)=\lambda+5h-2, && \rho_{-5}(\lambda)=\lambda-5h+2,\\
&\rho_4(\lambda)=\lambda+4h-1, && \rho_{-4}(\lambda)=\lambda-4h+1,\\
&\rho_3(\lambda)=\lambda+3h-1, && \rho_{-3}(\lambda)=\lambda-3h+1,\\
&\rho_2(\lambda)=\lambda+2h-1, && \rho_{-2}(\lambda)=\lambda-2h+1,\\
&\rho_1(\lambda)=\lambda+h, && \rho_{-1}(\lambda)=\lambda-h.\\
&\rho_0(\lambda)=\lambda,
\end{aligned}
\tag{7}
$$

If $h\geqq 2(H\geqq 5)$, then the functions (7) satisfy the conditions of Theorem 1, and hence, the resulting matrix may be covered by eleven stripes, which is more than the number of stripes resulting from the regular numbering. However, the stripes in this case are nonoverlapping provided that M_Ω does not have very narrow regions. These conditions on M_Ω are better phrased in the following lemmas.

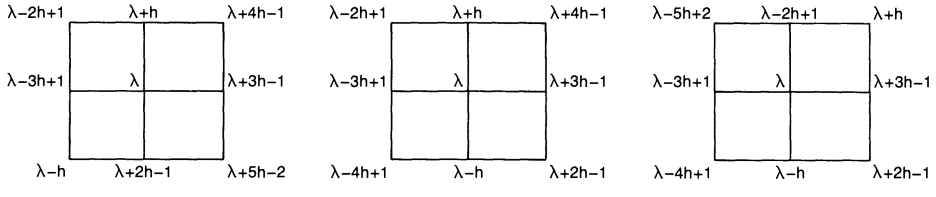
LEMMA 1. *Assuming 3-color node numbering and 4-node quadrilateral elements, if each column in M_Ω contains at least four elements that are either contiguous or divided into two groups of two contiguous elements each, then*

$$\text{Next}(\delta)-\delta\leqq h-2 \quad \textit{for any } \delta\in M_Q \tag{8}$$

where the function Next *is as given in Definition* 2.

Proof. For ease of reference, we indicate the position of a node that lies on the intersection of the zth horizontal line, $1\leqq z\leqq 3h-1$, and the νth vertical line, $1\leqq \nu\leqq W$, of M_Q by the pair (z, ν). If $\delta\in M_\Omega$, then $\text{Next}(\delta)=\delta$ and (8) is trivial. Hence, let $\delta\notin M_\delta$ be at position (z, ν) and have the color R. Let R1 be the color that follows R, that is R1 = black, red or white, if R = white, black or red, respectively. From the hypothesis, the νth column of elements in M_Ω contains either four contiguous elements or two pairs of contiguous elements. That is, there exist two horizontal lines a and b with $b\geqq a+2$ such that all the nodes at positions (c, ν) and $(c, \nu+1)$, for $a\leqq c\leqq a+2$ and $b\leqq c\leqq b+2$, are in M_Ω (see Fig. 6). Clearly, we may have one of three cases:

Case 1: $z<a$. In this case, a node $\mu\in M_\Omega$ with color R should exist at a position (c, ν), $a\leqq c\leqq a+2$, and $\mu-\delta-1=$ the number of lines with color R between lines z and a. In other words, $\mu-\delta-1$ is less than the number of lines with color R below line a. Since this number is $\lfloor(a-1)/3\rfloor$, the largest integer less than $(a-1)/3$, and given that $b+2\leqq 3h-1$ and $a\leqq b-2$, we obtain $\mu-\delta-1<\lfloor(3h-6)/3\rfloor=h-2$. By definition, $\text{Next}(\delta)\leqq\mu$, and hence $\text{Next}(\delta)-\delta\leqq h-2$.



(a) λ *is white.* (b) λ *is black.* (c) λ *is red.*

FIG. 5. *The neighbors of* λ.

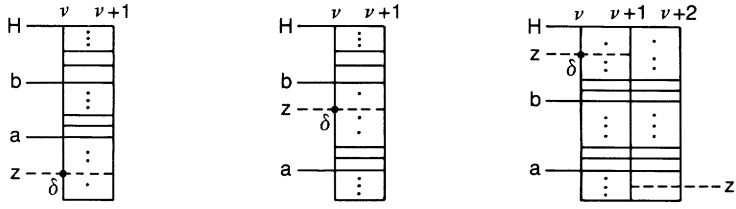


FIG. 6. *Column ν of* M_Ω.

Case 2: $a < z < b$. In this case a node $\mu \in M_\Omega$ with color R should exist at a position (c, ν), $b \leqq c \leqq b+2$, and $\mu - \delta - 1$ is less than the number of lines with color R between lines $a+2$ and b, that is, less than $\lfloor (b-(a+2)-1)/3 \rfloor + 1$. Since $b+2 \leqq 3h-1$ and $a \geqq 1$, then $\mu - \delta - 1 < \lfloor (3h-7)/3 \rfloor + 1 = h-2$. That is, Next $(\delta) - \delta \leqq \mu - \delta \leqq h-2$.

Case 3: $z > b+2$. If $\nu = W$ and R = red, then there is no node in M_Ω larger than δ. Hence, Next $(\delta) = n_Q$, and Next $(\delta) - \delta =$ the number of lines of color R above line $z = \lfloor (3h-1-z)/3 \rfloor$. But $z \geqq b+3 \geqq a+5 \geqq 6$, which gives Next $(\delta) - \delta \leqq h-3$.

On the other hand, if $\nu < W$, or $R \neq$ red, then a node $\mu \in M_\Omega$ with color R1 should exist at a position $(c, \nu+1)$ if R = red or at a position (c, ν) if R = white or black, where $a \leqq c \leqq a+2$. Here, $\mu - \delta - 1 =$ [the number of lines of color R above line z] + [the number of lines of color R1 below line a]. But at most $\lfloor (3h-1-z)/3 \rfloor$ lines with color R may be above line z, and at most $\lfloor (a-1)/3 \rfloor$ lines with color R1 may be below line a. Hence, $\mu - \delta - 1 \leqq \lfloor (3h-1-z)/3 \rfloor + \lfloor (a-1)/3 \rfloor \leqq \lfloor (3h-2-z+a)/3 \rfloor$.. Given that $z \geqq a+5$, then $\mu - \delta - 1 \leqq h-3$, which gives Next $(\delta) - \delta \leqq h-2$. □

LEMMA 2. *Assume* 3*-color node numbering and* 4*-node quadrilateral elements. If each column in* M_Ω *contains at least seven contiguous elements or two separated groups of at least three contiguous elements each, then*

$$\text{Next}(\delta) - \delta \leqq h-3 \quad \textit{for any } \delta \in M_Q. \tag{9}$$

Sketch of the proof. Let $\delta \notin M_\Omega$ be at position (z, ν). Then from the hypothesis, there exist two horizontal lines a and b, $b > a+3$, such that all the nodes at positions (c, ν) and $(c, \nu+1)$, for $a \leqq c \leqq a+3$ and $b \leqq c \leqq b+3$ are in M_Ω. The rest of the proof proceeds in a way similar to the proof of Lemma 1. □

LEMMA 3. *If for any* $\delta \in M_Q$, Next $(\delta) - \delta \leqq p$, *then,*

$$\nu^{-1}(l+1) - \nu^{-1}(l) \leqq p+1 \quad \textit{for } l = 1, \cdots, n_\Omega - 1. \tag{10}$$

Proof. Let $l = \nu(\lambda)$ and $l+1 = \nu(\bar{\lambda})$. Given that $\bar{\lambda}$ is numbered right after λ, then any node δ with $\lambda < \delta < \bar{\lambda}$ is not in M_Ω, and hence, Next $(\delta) = \bar{\lambda}$. But from the hypothesis $\bar{\lambda} - \delta \leqq p$, and hence if $\bar{\lambda} > \lambda + 1$, then $\bar{\lambda} - \lambda < \bar{\lambda} - \delta \leqq p$. That is, $\bar{\lambda} - \lambda \leqq p+1$. □

Condition (10) may be translated to an upper limit on the number of columns that a stripe in a stiffness matrix may jump in one row. More specifically, if $a_{l,\sigma_k(l)}$ and $a_{l+1,\sigma_k(l+1)}$ are in the same stripe, then condition (10) limits the value of $\sigma_k(l+1) - \sigma_k(l)$. This may be used to prove that the stripes will not overlap if they are adequately separated from each other.

THEOREM 2. *Let the nodes in a pierced rectangular grid* M_Ω *be numbered using the* 3*-color numbering scheme, and let A be the matrix that results from an* FE_4 *discretization on* M_Ω. *If* M_Ω *satisfies the conditions of Lemma* 1, *then A may be covered by eleven*

nonoverlapping stripes. Moreover, if M_Ω satisfies the conditions of Lemma 2, then the eleven stripes are strictly nonoverlapping.

Proof. Consider the eleven functions given by (7). It is straightforward to check that

$$\rho_k(\lambda+1)-\rho_k(\lambda)=1, \qquad k=-5,\cdots,5, \tag{11a}$$

$$\rho_{k+1}(\lambda)-\rho_k(\lambda)\geqq h-1, \quad k=-5,\cdots,4. \tag{11b}$$

Clearly, these functions satisfy the conditions in Theorem 1, and hence, A has eleven stripes S_k, $k=-5,\cdots,5$, of the form

$$S_k=\{(l,\sigma_k(l));\ 1\leqq l\leqq n_\Omega\} \tag{12}$$

where

$$\sigma_k(l)=\begin{cases}\nu(\rho_k(\nu^{-1}(l))) & \text{if } \rho_k(\nu^{-1}(l))\in M_\Omega,\\ \text{not defined} & \text{otherwise.}\end{cases}$$

In order to prove that these stripes do not overlap, we consider any integers l, m and k such that $(l,\sigma_k(l))\in S_k$ and $(l-m,\sigma_{k+m}(l-m))\in S_{k+m}$, and we let $l-m=\nu(\bar{\lambda})$ and $l=\nu(\lambda)$. From (12), we get

$$\sigma_{k+m}(l-m)-\sigma_k(l)=\nu(\rho_{k+m}(\bar{\lambda}))-\nu(\rho_k(\lambda)). \tag{13}$$

But from (11b), $\rho_{k+m}(\bar{\lambda})-\rho_k(\bar{\lambda})\geqq m(h-1)$, and from (11a), $\rho_k(\lambda)-\rho_k(\bar{\lambda})=(\lambda-\bar{\lambda})$. That is,

$$\rho_{k+m}(\bar{\lambda})-\rho_k(\lambda)\geqq m(h-1)-(\lambda-\bar{\lambda}).$$

Now, if the conditions of Lemma 1 are satisfied, then from (8) and (10), $\lambda-\bar{\lambda}\leqq m(h-1)$ and thus $\rho_{k+m}(\bar{\lambda})-\rho_k(\lambda)\geqq 0$. By property (3) of the renumbering function and (13), we finally obtain $\sigma_{k+m}(l-m)\geqq\sigma_k(l)$, which is the condition (2) for nonoverlapping stripes. On the other hand, if the conditions of Lemma 2 are satisfied, then from (9) and (10), $\lambda-\bar{\lambda}<m(h-1)$. This leads to $\sigma_{k+m}(l-m)>\sigma_k(l)$, which is the condition for strictly nonoverlapping stripes. □

The result of Theorem 2 proves that if an $n\times n$ stiffness matrix generated by FE_4 and 3-color numbering is used as input to MAT/VEC or TRIANG, then execution will not be delayed by internal data conflict.

The 3-color numbering scheme introduced here also causes the stripes of the matrices obtained from FD_5 and FE_3 discretizations to be nonoverlapping (as indicated in Table 1). However for FE_6 and FE_9 discretizations, this numbering does not spread the stripes enough and overlap may still occur. A 5-color numbering scheme is needed in this case to guarantee nonoverlapping stripes. The analysis of the 5-color scheme is similar to that discussed in this section.

Although the property of nonoverlapping stripes is important for the efficient operation of MAT/VEC and TRIANG, the multicolor numbering scheme has an additional advantage over the regular scheme. Namely, it produces matrices in which the stripes are uniformly spread, thus minimizing the maximum separation between stripes. This property, which will be analyzed in § 5.3, may be observed by comparing Figs. 3(b) and 4(b).

5. Performance of MAT/VEC applied to stiffness matrices. In this section, we first review briefly the principle of operation of the network MAT/VEC (the operation of TRIANG is very similar). We also describe a method which may be used for the estimation of the performance of the network for any specific input matrix. This method is then applied to the case in which input matrices result from finite difference or finite element analysis.

5.1. The network MAT/VEC. The network MAT/VEC is designed specifically for the multiplication of a vector x by an $n \times n$ matrix A, assuming that a stripe structure, $\Sigma_A = \{S_{-\pi_1}, \cdots, S_{\pi_2}\}$, for A is given. MAT/VEC is a linear network which consists of $\pi_1 + \pi_2 + 1$ cells labeled by the integers $-\pi_1, \cdots, \pi_2$. Each cell has five input ports and two output ports labeled as shown in Fig. 7. Successive nonzero elements in a particular stripe S_k, along with their column and row indices, are supplied on ports I_3, I_4 and I_5, respectively, of cell k.

Every two consecutive cells k and $k+1$ in MAT/VEC are connected by two unidirectional communication links, where a link is regarded as a queue that may buffer data between the two cells. One of the links is directed from $k+1$ to k and transmits the elements of the input vector x, and the other is directed from k to $k+1$ and transmits the elements of the result vector $y = Ax$. The network is data driven in the sense that the operation of each cell is initiated by the availability of its input.

The operation of each cell k may be described by the following cycle which is executed repeatedly:

(1) Read $a_{i,\sigma_k(i)}$, $\sigma_k(i)$ and i from I_3, I_4 and I_5, respectively,

(2) Write consecutive elements of y received on I_2 onto O_2 until y_i is received, and write consecutive elements of x received on I_1 onto O_1 until $x_{\sigma_k(i)}$ is received,

(3) Compute $y_i = y_i + a_{i,\sigma_k(i)} * x_{\sigma_k(i)}$, and write y_i and $x_{\sigma_k(i)}$ onto O_2 and O_1, respectively.

With this, each element of the result vector will travel toward the right, picking up, at cell k, the contribution of stripe S_k. In order to estimate the performance of the network, a hypothetical mode of operation is assumed in which execution alternates between communication phases and processing phases [6], [7]. In a communication phase, each cell executes steps 1 and 2 of the above cycle until it receives its appropriate data or it is blocked, temporarily, due to the unavailability of input. At this point, all the cells which have received their appropriate data execute step 3 while the other cells remain idle. This is a processing phase. A communication phase followed by a processing phase is called a global cycle. Note that this mode of operation is slower than the actual data-driven operation, and hence, it allows for a worst case analysis of actual execution.

Clearly, the time for a processing phase is the time needed to complete a floating point multiply/add, which is a constant (say τ_m) for a given architecture of the cells. On the other hand, the time for a communication phase depends on the stripe structure of the matrix. More specifically, if τ_c is the time needed to transmit a single data item between two cells, and N is the total number of global cycles, then the total execution time of MAT/VEC is

$$T = \tau_m N + \tau_c \sum_{t=1}^{N} \xi_t \tag{14}$$

where ξ_t, which will be determined later, is an indication of the communication activity during the tth communication phase. It is proved in [7] that for input matrices with

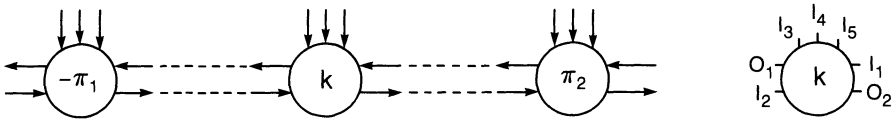


FIG. 7. *The network* MAT/VEC.

strictly nonoverlapping stripes and nonzero diagonal elements, the number of global cycles N is equal to n, the order of the input matrix.

If the time for a floating point operation τ_m is much larger than the time for the transmission of one data item τ_c, then the total execution time of MAT/VEC is approximately $N\tau_m$. However, if τ_c cannot be neglected, then ξ_t, $t=1,\cdots,N$ should be calculated as explained in the following section.

5.2. Performance estimation for specific input matrices. The progress of the computation in MAT/VEC may be modeled by the propagation of a computation front [6]. More specifically, let M_t be the subset of cells which are not idle during the tth processing phase. With this, we define the computation front at cycle t, denoted by CF_t, as the set that consists of the elements of A which are operated upon by the cells in M_t during the processing phase of cycle t. Computation fronts may be constructed systematically for specific input matrices and communication queue capacities [7]. In the remainder of this section, we will, for simplicity, assume that each y-stream communication queue may buffer only one data item, and each x-stream communication queue may buffer $d_{\min}$ data items, where $d_{\min}$ is arbitrarily large. A formula for obtaining a bound on $d_{\min}$ will be given in § 5.3.

The locations of the elements of the vector x in MAT/VEC during the tth processing phase may be modeled by a profile function xP_t, which is defined such that $xP_t(k)=j$ if x_j is at cell k during the tth processing phase. xP_t may be derived from CF_t according to the following:

$$xP_t(k)=\sigma_k(l) \quad \text{if } a_{l,\sigma_k(l)}\in CF_t, \tag{15a}$$

$$xP_t(k)\begin{cases} <xP_t(k+1) & \text{if } k+1\in M_t,\\ \leqq xP_t(k+1) & \text{if } k+1\notin M_t,\end{cases} \tag{15b}$$

$$xP_t(k)\begin{cases} <xP_{t+1}(k) & \text{if } k\in M_t,\\ \leqq xP_{t+1}(k) & \text{if } k\notin M_t,\end{cases} \tag{15c}$$

where (15a) guarantees the consistency of the computation and (15b) and (15c) guarantee that the order of elements in the x data stream are preserved in both space and time. Given the profile functions, the value of ξ_t in (14) may, then, be computed from (see [7] for a proof)

$$\xi_t=\max_k\{xP_t(k)-xP_{t-1}(k)\}. \tag{16}$$

For example, consider the computation of MAT/VEC on an $n\times n$ input matrix A which results from an FE_4 discretization of a pierced rectangular grid M_Ω. If the 3-color scheme of § 4 is used to number the nodes of the grid, then A may be covered with eleven strictly nonoverlapping stripes, and it may be shown that the computation fronts in this case are given by

$$CF_t=\{a_{t-k,\sigma_k(t-k)}\,|\,-5\leqq k\leqq 5\},\qquad t=1,\cdots,n \tag{17}$$

where σ_k, $k=-5,\cdots,5$, are as given in (12). In order to find profile functions which correspond to the fronts (17), consider the functions

$$xP_t(k)=\nu(\text{Next}\,(\rho_k(\nu^{-1}(t-k)))),\qquad t=1,\cdots,n \tag{18}$$

where the function Next is defined in Definition 2. Clearly, if $\sigma_k(t-k)$ is not defined, then from (12), $\rho_k(\nu^{-1}(t-k))\in M_\Omega$, and hence $\text{Next}\,(\rho_k(\nu^{-1}(t-k)))=\rho_k(\nu^{-1}(t-k))$, which by (12) and (18) gives

$$xP_t(k)=\sigma_k(t-k) \quad \text{if } \sigma_k(t-k) \text{ is defined}$$

thus satisfying (15a). The verification of (15b) and (15c) is straightforward. In other words, the functions in (18) are valid profile functions which may be used in (16) to compute ξ_t, $t = 1, \cdots, n$. Moreover, a bound on ξ_t may be obtained for arbitrary pierced rectangular domains M_Ω, namely:

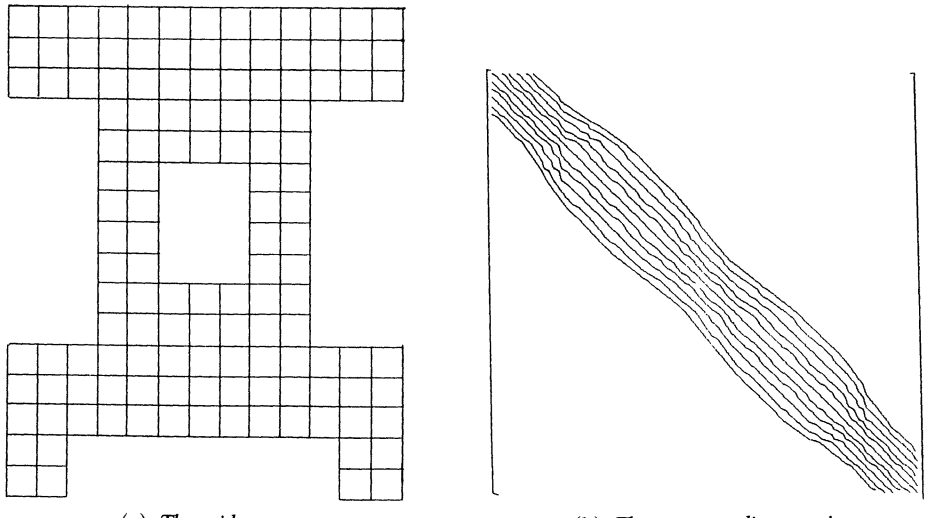
$$\xi_t \leqq \frac{H}{3}, \qquad t = 1, \cdots, n$$

where H is the height of M_Ω. However, this bound is relatively large, and the actual value of ξ_t for specific grids is usually much smaller than $H/3$, as shown by the following examples,

Example 1:

Consider the pierced rectangular grid shown in Fig. 8(a). It contains 130 4-node rectangular elements and 174 nodes. The stiffness matrix corresponding to the 3-color numbering scheme (Fig. 8(b)) has a band width $b = 49$ and, in accordance with Theorem 2, has 11 strictly nonoverlapping stripes. The construction of the data profiles (18) and the application of (16) gives $\sum_{t=1}^{174} \xi_t = 241$. That is, MAT/VEC completes the matrix/vector multiplication, using 11 cells, in time $T = 174\tau_m + 241\tau_c$. Note that if the multiplication is performed on a systolic network [5], then 49 cells are needed and the computation is completed in time $T_s = 348(\tau_m + \tau_c)$. The saving in both the number of cells and the execution time is obvious. Also, the number of cells π in MAT/VEC is independent of the size of the grid, while the number of cells used in the systolic approach [5], namely b, depends on the size of the grid (usually, $b = O(\sqrt{n})$).

In order to observe the effect of the numbering scheme, we also consider the matrix corresponding to the column-wise numbering of the grid of Fig. 8(a). This matrix has a bandwith $b_n = 35$ and may be covered by nine stripes. However, the stripes are not "strictly nonoverlapping," and the construction of the computation front shows that 326 global cycles are needed for the completion of the execution of MAT/VEC. Hence, the size of MAT/VEC is smaller for regular numbering than in the 3-color numbering (9 cells instead of 11 cells), but execution is slower (326 global cycles instead of 174 cycles).



(a) *The grid.* (b) *The corresponding matrix.*

FIG. 8. *Example* 1.

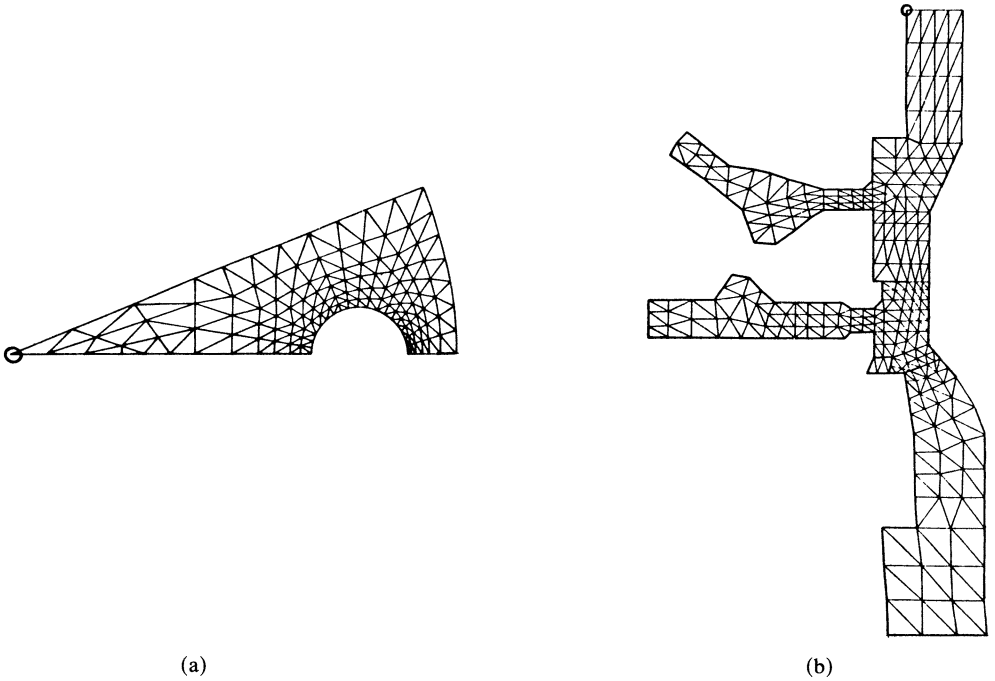
Clearly, general results, of the type proved in the previous sections, may only be obtained for grids that are, in some sense, regular. However, given any sparse matrix, and in particular a stiffness matrix, a stripe structure may be constructed for the matrix and the number of computation fronts needed for the execution of MAT/VEC may be estimated.

Example 2:

Highly irregular grids may be obtained if triangular elements are used. Consider, for example, the two grids shown in Fig. 9 that are extracted from [10]. The Cuthill-McKee numbering scheme [2] is used for both grids starting from the encircled nodes. The stiffness matrix corresponding to the grid of Fig. 9(a) is of order 145 and has a bandwith 25. The minimum number of stripes that may cover the matrix is 9 (overlapping) and the number of computation fronts is found to be 283. In other words, MAT/VEC uses 9 cells and terminates in 283 global cycles.

For the grid of Fig. 9(b), the order of the matrix is 289 and the bandwidth is 49. The number of stripes is found to be 13 and the corresponding number of computation fronts is 533. That is, MAT/VEC uses 13 cells and terminates in 533 cycles. By comparison with systolic multiplication, in which 25 cells and 290 cycles are required for the first matrix, and 49 cells and 578 cycles are required for the second matrix, it is clear that the organization of the nonzero elements into a stripe structure reduces the hardware needed for the completion of the multiplication, without slowing down execution.

Finally, we note that the grids in Fig. 9 are constructed without any consideration for the regularity of the stripe structure. More specifically, the same domains may be easily covered by grids that have the same element-density distribution as the given grids, but that are isomorphic to some pierced rectangular grids. The matrices generated from these grids should obey the results of §§ 3 and 4.



(a) (b)

FIG. 9. *Irregular grids.*

5.3. The maximum separation between stripes. The formula (17) for the computation fronts was determined assuming that each x-stream communication queue may buffer at least $d_{\min}$ data items, where $d_{\min}$ is found in [7] to be a measure of the maximum separation between the stripes. More specifically, if the stripes of the matrix are complete, then $d_{\min}$ may be estimated from

$$d_{\min} = \max_{k,t} \{\sigma_{k+1}(t) - \sigma_k(t)\}.$$

On the other hand, if the stripes of the matrix are not complete, then

$$d'_{\min} = \max_{k,t} \{xP_t(k+1) - xP_t(k)\} \tag{19}$$

where xP_t, $t = 1, \cdots, n$, are the x-stream data profiles.

In order to observe the effect of the node numbering scheme on the separation between stripes, we consider a rectangular grid M_Q, with $H = 3h - 1$ horizontal lines and W vertical lines, and we assume that A is the stiffness matrix that results from FE_4 discretization on M_Q. If regular node numbering is applied, then the functions (4) may be used to construct nine parallel complete stripes S_k, $k = -4, \cdots, 4$, that satisfy for any t

$$\sigma_{k+1}(t) - \sigma_k(t) = \begin{cases} 1 & \text{if } k = -4, -3, -1, 0, 2, 3, \\ H-2 & \text{if } k = -2, 1. \end{cases}$$

This gives $d_{\min} < H - 2 = 3(h-1)$. On the other hand, if 3-color numbering is applied, then the functions (7) may be used to produce eleven parallel complete stripes S_k, $k = -5, \cdots, 5$, that satisfy for any t

$$\sigma_{k+1}(t) - \sigma_k(t) = \begin{cases} h-1 & \text{if } k = -5, -2, 1, 4, \\ h & \text{if } k = -4, -3, -1, 0, 2, 3. \end{cases}$$

That is $d_{\min} < h$. Hence, although the multi-color numbering produces a matrix with a larger bandwidth ($5h-1$ instead of $3h+1$), the stripes are spread within the band almost uniformly, thus decreasing the maximum separation between the stripes from $3(h-1)$ to h.

The natural question to ask is: does $d_{\min}$ remain unchanged if M_Q is pierced and the nodes in the resulting pierced domain M_Ω are renumbered? In order to answer the above question positively, we substitute the profile functions (18) in (19) and get

$$d'_{\min} = \max_{k,l} \{\nu(\text{Next}\,(\rho_{k+1}(\bar{\lambda}))) - \nu(\text{Next}\,(\rho_k(\lambda)))\} \tag{20}$$

where $\lambda = \nu^{-1}(l)$ and $\bar{\lambda} = \nu^{-1}(l-1)$. Now, let $\phi = \max\{\mu \mid \mu \leqq \rho_{k+1}(\bar{\lambda}) \text{ and } \mu \in M_\Omega\}$. That is, ϕ is the first node before $\rho_{k+1}(\bar{\lambda})$ that is in M_Ω (take $\phi = 0$ if no such μ exists). Given that $\text{Next}\,(\rho_{k+1}(\bar{\lambda}))$ is the first node in M_Ω after $\rho_{k+1}(\bar{\lambda})$, we get $\nu(\text{Next}\,(\rho_{k+1}(\bar{\lambda}))) = \nu(\phi) + 1$. Hence

$$\nu(\text{Next}\,(\rho_{k+1}(\bar{\lambda}))) - \nu(\text{Next}\,(\rho_k(\lambda))) = \nu(\phi) - \nu(\text{Next}\,(\rho_k(\lambda))) + 1. \tag{21}$$

But $\phi \leqq \rho_{k+1}(\bar{\lambda})$ and $\text{Next}\,(\rho_k(\lambda)) \geqq \rho_k(\lambda)$. This gives

$$\begin{aligned} \phi - \text{Next}\,(\rho_k(\lambda)) &\leqq \rho_{k+1}(\bar{\lambda}) - \rho_k(\lambda) \\ &\leqq \rho_k(\bar{\lambda}) - \rho_k(\lambda) + h \end{aligned} \tag{22}$$

where we used $\rho_{k+1}(\bar{\lambda}) - \rho_k(\bar{\lambda}) \leqq h$, which may be verified from (7). Also, from (7), $\lambda > \bar{\lambda}$ implies that $\rho_k(\bar{\lambda}) < \rho_k(\lambda)$, which together with the property (3b) of the renumbering function ν and (22) gives

$$\nu(\phi) = \nu(\text{Next}(\rho_k(\lambda))) < h. \tag{23}$$

From (20) and (21) in (23), we finally get $d'_{\min} \leqq h$. That is, for matrices which are generated from FE_4 discretizations on pierced rectangular grids with height H and 3-color node numbering, a buffer capacity of $H/3$ is sufficient to ensure that MAT/VEC terminates execution in n global cycles. Similar results may be obtained for other discretizations.

6. Conclusion. It is shown that the number of stripes π in the stripe structure of a stiffness matrix is independent of the size of the problem, and is much smaller than the bandwidth of the matrix. For pierced rectangular domains, the stripe count π may be estimated analytically and the stripe structure of the matrix may be constructed from the finite element grid.

The multicolor node numbering presented in this paper has two favorable effects on the resulting matrix: First, it produces nonoverlapping stripes, which prevents any data conflict during the execution of MAT/VEC, and second, it distributes the stripes uniformly, which reduces the maximum separation between stripes and thus minimizes the number of buffers needed in MAT/VEC.

In brief, the construction of stripe structures for stiffness matrices allows for the efficient utilization of computational networks which are independent of the size of the problem.

Acknowledgment. It is my pleasure to dedicate this work to Werner Rheinboldt, my former advisor, who taught me how to do research and showed me how to enjoy it.

REFERENCES

[1] L. ADAMS, *Iterative algorithms for large sparse linear systems on parallel computers*, Ph.D. thesis, Univ. of Virginia, Charlottesville, VA, 1982.

[2] E. CUTHILL AND J. MCKEE, *Reducing the bandwidth of sparse symmetric matrices*, Proc. ACM National Conference, New York, 1969, pp. 157–172.

[3] S. EISENSTAT, M. GURSKY, M. SCHULTZ AND A. SHERMAN, *Yale sparse matrix package*, Tech. Reports 112, 114, Dept. of Computer Science, Yale University, New Haven, CT, 1977.

[4] A. GEORGE AND J. LIU, *Computer solutions of large sparse positive definite systems*, in Computational Mathematics, Prentice-Hall, Englewood Cliffs, NJ, 1981.

[5] H. T. KUNG AND C. E. LEISERSON, *Systolic arrays for* VLSI, in Introduction to VLSI Systems, C. Mead and L. Conway, eds., Addison-Wesley, Reading, MA, 1980.

[6] R. MELHEM, *A study of data interlock in computational networks for sparse matrix multiplication*, Tech. Report TR-CS-505, Dept. of Computer Science, Purdue Univ., West Lafayette, IN; IEEE Trans. Comput., 36 (1987).

[7] ———, *Parallel solution of linear systems with striped sparse matrices*, Tech. Report ICMA-86-91, January 1986, Parallel Computing, to appear.

[8] D. O'LEARY, *Ordering schemes for parallel processing of certain mesh problems*, SIAM J. Sci. Statist. Comput., 5 (1984), pp. 620–632.

[9] Y. SAAD AND M. SCHULTZ, *Parallel implementations of preconditioned conjugate gradient methods*, Tech. Report YALEU/DCS/RR425, Dept. of Computer Science, Yale University, New Haven, CT, Oct. 1985.

[10] O. C. ZIENKIEWICZ, *The Finite Element Method*, McGraw-Hill, New York, 1979.