

## ALGORITHMIC ASPECTS OF BIPARTITE GRAPHS

MIHÁLY BAKONYI

Department of Mathematics  
Georgia State University  
Atlanta, GA 30303

ERIK M. VARNES

Department of Mathematics  
Valparaiso University  
Valparaiso, Indiana 46383

(Received March 19, 1993)

**ABSTRACT.** We generalize previous work done by Donald J. Rose and Robert E. Tarjan [2], who developed efficient algorithms for use on directed graphs. This paper considers an edge elimination process on bipartite graphs, presenting several theorems which lead to an algorithm for computing the minimal fill-in of a given ordered graph.

**KEY WORDS AND PHRASES.** Gaussian elimination, bipartite graph, perfect edge elimination, fill-in.

**1992 AMS SUBJECT CLASSIFICATION CODES.** 68R10, 05C20

### 1. INTRODUCTION

Gaussian elimination is commonly used in matrix theory. If the matrix is sparse, that is, contains a large number of zero entries, it is advantageous to avoid unnecessary operations on those zeros in order to save memory and run time. When we perform Gaussian elimination, some of the zeros in the matrix may be replaced by nonzeros. These new nonzeros are called "fill-in". A perfect ordering is a sequence of Gaussian pivots which produces no fill-in.

Much is known about the case in which the coefficient matrix is symmetric (see Golubic [1]). It is also known how to determine fill-ins for nonsymmetric matrices when the pivots are chosen on the main diagonal. Algorithms have been developed for computing the fill-in for any ordering of pivots, for generating a perfect ordering if such an ordering exists, and for reducing a fill-in to a minimal one (Rose and Tarjan [2]). However, it may be the case that choosing pivots which are not on the main diagonal produces a smaller fill-in than does choosing pivots which are on the main diagonal. For example, choosing any pivot on the main diagonal in the following matrix produces a fill-in of at least one entry. (The entries denoted by an X represent non-zeros in the matrix.)

$$M = \begin{bmatrix} X & X & X & X \\ X & X & X & 0 \\ 0 & X & X & 0 \\ 0 & 0 & X & X \end{bmatrix}$$

But choosing the entry in row two, column one produces no fill-in, and, in fact, there is a perfect elimination ordering for this matrix if pivots off the main diagonal are chosen.

Therefore, this paper generalizes the results of Rose and Tarjan [2] in order to develop an algorithm which takes all nonzero entries of a given matrix into account, as opposed to only the entries on the main diagonal. We draw heavily from several chapters of a book by Golubic [1], and some of the proofs of Rose and Tarjan [2] can be readily adapted for our purposes.

### 2. DEFINITIONS AND NOTATION

We generally follow the same notation as Golubic [1] and Rose et al. [3].

An *undirected graph* is a pair  $G = (V, E)$ , where  $V$  is a finite set of  $n = |V|$  elements called *vertices* and  $E \subseteq \{(v, w) | v, w \in V, v \neq w\}$  is a set of  $e = |E|$  unordered vertex pairs called *edges*. An undirected graph  $G = (V, E)$  is *bipartite* if its vertices can be partitioned into two disjoint sets  $V = X + Y$ , i.e. every edge has one endpoint in  $X$  and the other in  $Y$ . We use the

notation  $\mathbf{H} = (\mathbf{X}, \mathbf{Y}, \mathbf{E})$  to denote bipartite graphs. A bipartite graph  $\mathbf{H} = (\mathbf{X}, \mathbf{Y}, \mathbf{E})$  is *complete* if for every  $\mathbf{x} \in \mathbf{X}$  and  $\mathbf{y} \in \mathbf{Y}$  we have  $(\mathbf{x}, \mathbf{y}) \in \mathbf{E}$ .

Given  $\mathbf{v} \in \mathbf{V}$ ,  $Adj(\mathbf{v}) = \{\mathbf{w} \in \mathbf{V} | (\mathbf{v}, \mathbf{w}) \in \mathbf{E}\}$  is the set of vertices adjacent to  $\mathbf{v}$ . For a bipartite graph  $\mathbf{H}$  and  $(\mathbf{x}, \mathbf{y}) \in \mathbf{E}$ ,  $Adj(\mathbf{x}, \mathbf{y}) = \{(\mathbf{x}, \mathbf{k}) | \mathbf{k} \in \mathbf{Y}\} \cup \{(\mathbf{j}, \mathbf{y}) | \mathbf{j} \in \mathbf{X}\}$  is the set of edges adjacent to  $(\mathbf{x}, \mathbf{y})$ .

A *path*  $[\mathbf{v}_1, \dots, \mathbf{v}_k]$  is a sequence of vertices such that  $(\mathbf{v}_j, \mathbf{v}_{j+1}) \in \mathbf{E}$  for  $j = 1, \dots, k-1$ . A *cycle* of length  $k > 2$  is a path  $[\mathbf{v}_1, \dots, \mathbf{v}_k, \mathbf{v}_1]$  in which  $\mathbf{v}_1, \dots, \mathbf{v}_k$  are distinct.

A graph called *chordal* if every cycle of length strictly greater than 3 possesses a *chord*, i.e. an edge joining two nonconsecutive vertices of the cycle. Similarly, a bipartite graph is said to be *bipartite chordal* if every cycle of length strictly greater than four contains a chord.

Given a subset  $\mathbf{A} \subseteq \mathbf{V}$  of the vertices, define the *subgraph induced by A* to be  $\mathbf{G}_A = (\mathbf{A}, \mathbf{E}_A)$  where  $\mathbf{E}_A = \{(\mathbf{x}, \mathbf{y}) | \mathbf{x} \in \mathbf{A} \text{ and } \mathbf{y} \in \mathbf{A}\}$ . A subset  $\mathbf{A} \subseteq \mathbf{V}$  is called a *clique* if it induces a complete subgraph. We call a vertex  $\mathbf{x}$  *simplicial* if  $Adj(\mathbf{x})$  induces a complete subgraph. In a bipartite graph, a subset  $\mathbf{A} \subseteq \mathbf{X} \cup \mathbf{Y}$  is called a *biclique* if it induces a complete bipartite subgraph. An edge  $\mathbf{e} = (\mathbf{x}, \mathbf{y})$  of a bipartite graph  $\mathbf{H}$  is called *bisimplicial* if  $Adj(\mathbf{x}) + Adj(\mathbf{y})$  induces a complete bipartite subgraph of  $\mathbf{H}$ .

For a bipartite graph  $\mathbf{H} = (\mathbf{X}, \mathbf{Y}, \mathbf{E})$ , with  $|\mathbf{X}| = |\mathbf{Y}| = n$ , an *ordering* on  $\mathbf{E}$  is a sequence  $\phi = [(\mathbf{x}_1, \mathbf{y}_1), (\mathbf{x}_2, \mathbf{y}_2), \dots, (\mathbf{x}_n, \mathbf{y}_n)] = [e_1, e_2, \dots, e_n]$  of pairwise nonadjacent edges of  $\mathbf{H} = (\mathbf{X}, \mathbf{Y}, \mathbf{E})$ , where  $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n]$  and  $\mathbf{Y} = [\mathbf{y}_1, \mathbf{y}_2, \dots, \mathbf{y}_n]$ . The graph  $\mathbf{H}_\phi = (\mathbf{X}, \mathbf{Y}, \mathbf{E}, \phi)$  is an *ordered graph*.

Denote by  $S_i$  the set of endpoints of the edges  $e_1, \dots, e_n$  and let  $S_0 = \emptyset$ . We say that  $\phi$  is a *perfect edge elimination scheme* for  $\mathbf{H}$  if each edge  $e_i$  is bisimplicial in the remaining induced subgraph  $\mathbf{H}_{V-S_{i-1}}$  and  $\mathbf{H}_{V-S_n}$  has no edge. Thus we regard the elimination of an edge as the removal of all edges adjacent to  $\mathbf{e}$ . A graph with a perfect edge elimination ordering is called a *perfect edge elimination graph*. Since this paper deals primarily with edge elimination orderings, we will refer to edge elimination orderings as simply "*elimination orderings*" or "*elimination schemes*".

Finally, for a vertex  $\mathbf{v}$ , the *deficiency*  $\mathbf{D}(\mathbf{v})$  is the set of pairs defined by  $\mathbf{D}(\mathbf{v}) = \{(\mathbf{x}, \mathbf{y}) | \mathbf{v} \in Adj(\mathbf{x}), \mathbf{v} \in Adj(\mathbf{y}), \mathbf{x} \notin Adj(\mathbf{y}), \mathbf{x} \neq \mathbf{y}\}$ . So an alternate definition for a vertex  $\mathbf{v}$  to be *simplicial* is if  $\mathbf{D}(\mathbf{v}) = \emptyset$ .

For an edge  $(\mathbf{x}, \mathbf{y})$ , the *deficiency*  $\mathbf{D}(\mathbf{x}, \mathbf{y})$  is the set of pairs defined by  $\mathbf{D}(\mathbf{x}, \mathbf{y}) = \{(\mathbf{a}, \mathbf{b}) | \mathbf{a}, \mathbf{b} \in Adj(\mathbf{x}) + Adj(\mathbf{y}), (\mathbf{a}, \mathbf{b}) \notin \mathbf{E}\}$ . An alternate definition for an edge  $(\mathbf{x}, \mathbf{y})$  in the bipartite graph  $\mathbf{H}$  to be *bisimplicial* is if  $\mathbf{D}(\mathbf{x}, \mathbf{y}) = \emptyset$ .

Given an arbitrary  $n \times n$  matrix  $\mathbf{M} = (m_{ij})$ , the bipartite graph  $\mathbf{H} = (\mathbf{X}, \mathbf{Y}, \mathbf{E})$ , with  $\mathbf{X} = \{\mathbf{x}_1, \dots, \mathbf{x}_n\}$  and  $\mathbf{Y} = \{\mathbf{y}_1, \dots, \mathbf{y}_n\}$ , and the property that  $m_{ij} = 0$  whenever  $(\mathbf{x}_i, \mathbf{y}_j) \notin \mathbf{E}$  is called a *bipartite graph of M*.

When reducing  $\mathbf{M}$  by Gaussian elimination, the edges of  $\mathbf{D}(\mathbf{x}_i, \mathbf{y}_j)$  correspond exactly to the fill-in entries created when element  $m_{ij}$  is used as a pivot in the matrix  $\mathbf{M}$ . Pivoting on  $m_{ij}$  is equivalent to making  $Adj(\mathbf{x}_i) + Adj(\mathbf{y}_j)$  into a complete subgraph by adding any missing edges and deleting  $(\mathbf{x}_i, \mathbf{y}_j)$ . Therefore, a graph which has a perfect scheme is equivalent to a matrix which can be reduced to the identity without fill-in.

Let  $\mathbf{M}$  be a matrix with bipartite graph  $\mathbf{H}$  and  $\phi$  an ordering of  $\mathbf{H}$ . Define  $\mathbf{F}(\mathbf{H}_\phi)$  to be the set of all fill-ins generated by reducing  $\mathbf{M}$  by Gaussian elimination when the entries corresponding to the ordering  $\phi$  are successively chosen to act as pivots. Then  $\phi$  is a *minimal elimination ordering* of  $\mathbf{H}$  if no other ordering  $\delta$  satisfies  $\mathbf{F}(\mathbf{H}_\delta) \subset \mathbf{F}(\mathbf{H}_\phi)$ . The ordering  $\phi$  is a *minimum elimination ordering* of  $\mathbf{H}$  if no other ordering  $\delta$  satisfies  $|\mathbf{F}(\mathbf{H}_\delta)| < |\mathbf{F}(\mathbf{H}_\phi)|$ . This paper deals with minimal orderings only. It is known that the computation of the minimum fill-in of a graph is NP-complete, Yannakakis [4].

### 3. CHARACTERISTICS OF BIPARTITE GRAPHS

In the case of symmetric matrices, Gaussian elimination is modelled by vertex elimination instead of edge elimination. It is known that a symmetric matrix  $\mathbf{M}$  has a perfect vertex elimination scheme if and only if its associated graph is chordal (Golumbic, [1]). But there is no character-

ization of graphs with edge elimination schemes in terms of some forbidden subgraphs. While it is true that every chordal bipartite graph is a perfect elimination bipartite graph (Golubic, [1]), the converse of this statement is false. That is, it may be the case that a perfect elimination bipartite graph is not chordal bipartite. In this paper, there will be no restriction that the graph must be bipartite chordal.

The bisimpliciality of an edge is a hereditary property. That is, if an edge is bisimplicial, it is also bisimplicial in an induced subgraph. In order to construct perfect schemes, it is possible to choose an arbitrary bisimplicial edge, remove it, and then repeat the process in the induced subgraph. A theorem from Golubic [1] formalizes this observation.

**THEOREM 1.** If  $e = (x, y)$  is a bisimplicial edge of a perfect elimination bipartite graph  $H = (X, Y, E)$ , then  $H_{X - \{x\} + Y - \{y\}}$  is also a perfect elimination bipartite graph.

In other words, if  $H$  is a perfect elimination bipartite graph and  $(x, y)$  is any edge with  $D(x, y) = \emptyset$ , then there is a perfect elimination ordering  $\phi$  with  $\phi(1) = (x, y)$ .

The following theorems, then, are generalizations from Rose and Tarjan [2]. Theorem 2 is mentioned as Ex.4, p.285 in Golubic [1] but for the sake of completeness it is proven here. Theorem 2 essentially says that we can add the deficiency of an edge to a perfect elimination graph, and new graph will have the same perfect elimination scheme.

**THEOREM 2.** Let  $H = (X, Y, E)$  be a bipartite graph with perfect edge elimination scheme  $\phi = [(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)]$ . Let  $e \in E$  and let  $H' = (X, Y, E \cup D(e))$ . Then  $\phi$  is a perfect edge elimination scheme for  $H'$ .

**PROOF.** We must show that for any  $(x_p, y_k), (x_k, y_m) \in E \cup D(e)$  with  $m, p > k, (x_p, y_m) \in E \cup D(e)$ . There are three cases to consider. In the following illustrations, the dotted lines indicate an edge in  $D(e)$ , while solid lines indicate edges in  $E \cup D(e)$ .

Case 1: Both  $(x_p, y_k)$  and  $(x_k, y_m)$  are in  $E$ . Then since  $\phi$  is perfect for  $H = (X, Y, E)$ ,  $(x_p, y_m)$  must be in  $E$ . Therefore,  $(x_p, y_m) \in E \cup D(e)$ .

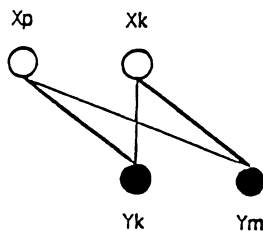


Fig. 1

Case 2: Both  $(x_p, y_k)$  and  $(x_k, y_m)$  are in  $D(e)$ . Then let  $e = (x_t, y_q)$ . So  $(x_k, y_m) \in D(e)$  implies  $(x_k, y_q) \in E$  and  $(x_t, y_m) \in E$ , and  $(x_p, y_k) \in D(e)$  implies  $(x_p, y_q) \in E$  and  $(x_t, y_k) \in E$ . Then  $(x_t, y_m)$  and  $(x_p, y_q) \in E$  implies  $(x_p, y_m) \in E \cup D(e)$ .

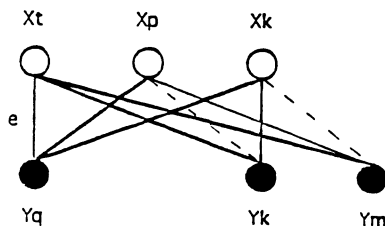


Fig. 2

Case 3: One of the edges, say  $(x_k, y_m)$ , without loss of generality, is in  $E$ , while  $(x_p, y_k)$  is in  $D(e)$ . Then let  $e = (x_t, y_q)$ . Since  $(x_p, y_k) \in D(e)$ ,  $(x_p, y_q)$  is in  $E$  and  $(x_t, y_k)$  is in  $E$ .

If  $t > k$ , then  $(x_t, y_m) \in E$ , since  $\phi$  is a perfect ordering for  $H$ . But then we have  $(x_t, y_m) \in E$ , and  $(x_p, y_q) \in E$ , which implies  $(x_p, y_m) \in E \cup D(e)$ .

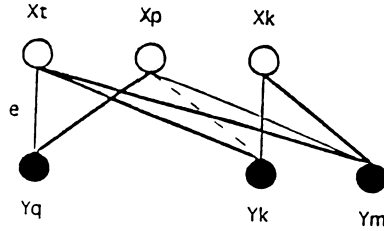


Fig. 3

If  $t < k$ , then  $(x_p, y_t) \in E \cup D(e)$ , which implies  $(x_p, y_k) \in E$ . But in that case we return to Case 1, and  $(x_p, y_m) \in E \cup D(e)$ .

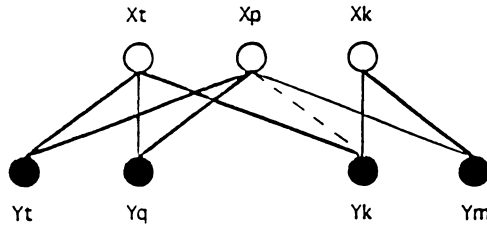


Fig. 4

**COROLLARY 1.** If  $H = (X, Y, E)$  is a perfect elimination bipartite graph, then for any edge  $(x, y)$  the bipartite graph  $H_{(x,y)} = (X \setminus \{x\}, Y \setminus \{y\}, E_{X \setminus \{x\} + Y \setminus \{y\}} \cup D(x, y))$  is also a perfect elimination graph.

**PROOF.** By Theorem 2, the graph  $H' = (X, Y, E \cup D(x, y))$  has a perfect edge elimination scheme, and  $(x, y)$  is bisimplicial in  $H'$ . Then, Theorem 1 implies that  $H_{(x,y)}$  is also a perfect edge elimination graph.

The next theorem provides a characterisation of bipartite graphs with nonminimal fill-in, a theorem which is utilized extensively in the algorithm at the end of this paper.

**THEOREM 3.** Let  $H = (X, Y, E)$  be a bipartite graph and let  $F \neq \emptyset$  be a fill-in for  $H$ . If  $F$  is not minimal, then there exists an edge  $e = (x, y)$  such that  $D(x, y) \subset F$  and  $D'(x, y) = \emptyset$ , where  $D'(x, y)$  is the deficiency of  $e$  in  $H' = (X, Y, E \cup F)$ .

**PROOF.** Let  $S = \{(x, y) \in E \cup F : D'(x, y) = \emptyset\}$ . For any edge  $(x, y)$ ,  $D(x, y) \subseteq D'(x, y)$ , so  $D(x, y) \setminus F \subseteq D'(x, y)$ . Hence if  $(x, y)$  is in  $S$ , then  $D(x, y) \setminus F \subseteq \emptyset$ , or  $D(x, y) \subseteq F$ . Since  $H'$  is a perfect elimination graph,  $S \neq \emptyset$ . Let  $(w, z) \in S$ . If  $D(w, z) \subset F$ , then the theorem holds with  $(x, y) = (w, z)$ .

Suppose, on the other hand, that  $D(w, z) = F$ . Let  $F_0$  be a fill-in for  $H$  such that  $F_0 \subset F$ , and let  $\phi$  be a perfect elimination ordering for  $H_0 = (X, Y, E \cup F_0)$ . Then  $D(\phi(1)) \subseteq F_0 \subset F$ . By Theorem 2,  $\phi$  is a perfect ordering for  $H' = (X, Y, E \cup D(w, z))$ . Thus  $D'(\phi(1)) = \emptyset$ . Hence the theorem holds with  $(x, y) = \phi(1)$ .

The next theorem states that given a bipartite graph with a perfect scheme and a supergraph with the same property, we can remove an edge from the supergraph such that the resulting graph is also a perfect elimination graph.

**THEOREM 4.** Let  $H = (X, Y, E)$  be a perfect elimination bipartite graph. If  $H' = (X, Y, E \cup J)$  and  $J \neq \emptyset$ , and  $|J| \geq 2$ , then there exists an  $f \in J$  such that  $H'' = (X, Y, E \cup J \setminus \{f\})$  is also a perfect elimination graph.

**PROOF.** Proof is by induction on  $n = |E|$ . If  $n \leq 1$ , the result is obvious. Suppose the result is true for all  $n \leq n_0$  and let  $n = n_0 + 1$ . By Theorem 3, there exists an edge  $e = (x, y)$  such that  $D(x, y) \subset J$  and  $D'(x, y) = \emptyset$ . We have two cases.

Case 1: There is an edge  $e_1 \in J$  which has  $x$  or  $y$  as an endpoint. By Theorem 2, there

exists a perfect elimination ordering  $\phi$  for  $H'$  with  $\phi(1) = (x, y)$ , but this ordering is also perfect for  $H' = (X, Y, E \setminus \{e\})$ .

Case 2: Case 1 does not apply. Then by Theorem 3, since  $F$  is not minimal, there exists an edge  $e = (x, y)$  such that  $D'(x, y) = \emptyset$  and  $D(x, y) \subset J$ . Then  $H_{(x,y)} = (X \setminus \{x\}, Y \setminus \{y\}, E_{X \setminus \{x\} + Y \setminus \{y\}} \cup D(x, y))$  and  $H'_{(x,y)} = (X \setminus \{x\}, Y \setminus \{y\}, E_{X \setminus \{x\} + Y \setminus \{y\}} \cup J)$  are perfect elimination graphs.

Then by the induction hypothesis, there exists  $f \in J \setminus D(x, y)$  such that  $H''_{(x,y)} = (X \setminus \{x\}, Y \setminus \{y\}, E_{X \setminus \{x\} + Y \setminus \{y\}} \cup (J \setminus \{f\}))$  is a perfect elimination graph. But then  $H' = (X, Y, E \cup (J \setminus \{f\}))$  is a perfect elimination graph since  $f \notin D(x, y)$ .

#### 4. MINIMAL FILL PROCEDURE

The previous theorems allow us to construct an efficient algorithm for determining the minimal fill-in of a given bipartite graph. The algorithm is most easily stated as a recursive procedure. By Theorem 3, if the fill-in for a graph is not minimal, we can find an edge  $(x, y)$  whose deficiency in  $H$  is not all of  $F$  and whose deficiency in  $H'$  is empty. If some edge  $e$  in  $F$  is adjacent to  $(x, y)$ , we can delete  $e$  from  $F$  and repeat the procedure. Otherwise, we form the graph  $H_1 = (X \setminus \{x\}, Y \setminus \{y\}, E_{X \setminus \{x\} + Y \setminus \{y\}} \cup D(x, y))$ , which has a fill-in of  $F - D(x, y)$ . The procedure is applied recursively so that  $F - D(x, y)$  is reduced to a minimal fill-in  $F_1$  in  $H_1$ . We then form the graph  $H_2 = (X, Y, E \cup F_1)$ , which has fill-in  $D(x, y)$ . Again, we apply the procedure recursively to reduce  $D(x, y)$  to a minimal fill-in  $F_2$  in  $H_2$ . Then  $F_0 = F_1 \cup F_2$  is a minimal fill-in for  $H$ . The following algorithm summarizes this procedure.

Procedure MINFILL  $(X, Y, E, F, F_0)$

BEGIN

declare  $F_1, F_2$  set variables local to procedure MINFILL

IF  $H = (X, Y, E)$  has no edge  $(x, y)$  with  $D'(x, y) = \emptyset$  and  $D(x, y) \subset F$

    THEN  $F_0 := F$ .

    ELSE BEGIN

        Let  $(x, y)$  be an edge with  $D'(x, y) = \emptyset$  and  $D(x, y) \subset F$ .

        IF  $F$  contains some edge  $e$  incident to  $(x, y)$

            THEN (Call 1) MINFILL  $(X, Y, E, F - \{e\}, F_0)$

        ELSE BEGIN

            (Call 2) MINFILL  $(X \setminus \{x\}, Y \setminus \{y\}, E_{X \setminus \{x\} + Y \setminus \{y\}} \cup D(x, y), F - D(x, y), F_1)$

            (Call 3) MINFILL  $(X, Y, E \cup F_1, D(x, y), F_2)$

$F_0 := F_1 \cup F_2$

        END

    END

END

**THEOREM 5.** If  $F$  is a fill-in for a graph  $H = (X, Y, E)$ , then the set  $F_0$  computed by the execution of  $MINFILL(X, Y, E, F, F_0)$  is a minimal fill-in contained in  $F$ .

**PROOF.** The proof largely follows that of Rose and Tarjan [2]. We proceed by induction on the number of edges in  $F$ . If  $F = \emptyset$ , the theorem is obviously true. Suppose the theorem is true is  $|F| \leq k$  and let  $|F| = k + 1$ .

Now, if  $H = (X, Y, E)$  has no edge  $(x, y)$  such that  $D'(x, y) = \emptyset$  and  $D(x, y) \subset F$ , then  $F_0 = F$  is minimal by Theorem 3. So suppose  $(x, y)$  is an edge satisfying  $D'(x, y) = \emptyset$  and  $D(x, y) \subset F$ .

If  $F$  contains an edge  $e$  incident to  $(x, y)$ , then  $F - \{e\}$  is a fill-in by the proof of Theorem 4. Then by the induction hypothesis, the set  $F_0 \subseteq F - \{e\}$  computed by the recursive call (Call 1) is a minimal fill-in.

Suppose  $\mathbf{F}$  contains no edge  $\mathbf{e}$  incident to  $(\mathbf{x}, \mathbf{y})$ . Since  $\mathbf{D}'(\mathbf{x}, \mathbf{y}) = \emptyset$ ,  $\mathbf{H}'_1 = (\mathbf{X} \setminus \{\mathbf{x}\}, \mathbf{Y} \setminus \{\mathbf{y}\}, \mathbf{E}_{\mathbf{X} - \{\mathbf{x}\} + \mathbf{Y} - \{\mathbf{y}\}} \cup \mathbf{F})$  has a perfect ordering by Theorem 1. Thus  $\mathbf{F} - \mathbf{D}(\mathbf{x}, \mathbf{y})$  is a fill-in for  $\mathbf{H}_1$ . And by the induction hypothesis, the set  $\mathbf{F}_1 \subseteq \mathbf{F} - \mathbf{D}(\mathbf{x}, \mathbf{y})$  computed by the recursive call (Call 2) is a minimal fill in for  $\mathbf{H}_1$ .

Since  $\mathbf{D}(\mathbf{x}, \mathbf{y})$  is a fill-in for  $\mathbf{H}_2 = (\mathbf{X}, \mathbf{Y}, \mathbf{E} \cup \mathbf{F}_1)$ , the induction hypothesis also implies that the set  $\mathbf{F}_2 \subseteq \mathbf{D}(\mathbf{x}, \mathbf{y})$  computed by the recursive call (Call 3) is a minimal fill-in for  $\mathbf{H}_2$ .

Now let  $\mathbf{W} \subseteq \mathbf{F}_1 \cup \mathbf{F}_2$  be a fill-in for  $\mathbf{H}$ . Then  $(\mathbf{X}, \mathbf{Y}, \mathbf{E} \cup \mathbf{W})$  is a perfect elimination graph. By Theorem 2, so is  $(\mathbf{X}, \mathbf{Y}, \mathbf{E} \cup \mathbf{W} \cup \mathbf{D}(\mathbf{x}, \mathbf{y}))$ , and by Theorem 1, so is  $\mathbf{H}_\mathbf{W} = (\mathbf{X} \setminus \{\mathbf{x}\}, \mathbf{Y} \setminus \{\mathbf{y}\}, \mathbf{E}_{\mathbf{X} - \{\mathbf{x}\} + \mathbf{Y} - \{\mathbf{y}\}} \cup \mathbf{D}(\mathbf{x}, \mathbf{y}) \cup \mathbf{W})$ . Since  $\mathbf{F}_1$  is a minimal fill-in for  $\mathbf{H}_\mathbf{W}$ , we have  $\mathbf{F}_1 \subseteq \mathbf{W}$ .

Finally, we have that  $\mathbf{F}_1 \subseteq \mathbf{W} \subseteq \mathbf{F}_1 \cup \mathbf{F}_2$ , and  $\mathbf{W}$  is a fill-in for  $\mathbf{H}$ , and  $\mathbf{F}_2$  is a minimal fill-in for  $\mathbf{H}_2$ . This all implies that  $\mathbf{W} = \mathbf{F}_1 \cup \mathbf{F}_2$ . Thus  $\mathbf{F}_1 \cup \mathbf{F}_2$  is a minimal fill-in for  $\mathbf{H}$ .

The above algorithm, as we have described, examines all entries in a matrix to determine a minimal fill-in. The following example illustrates the usefulness of this algorithm.

$$\mathbf{M} = \begin{bmatrix} X & X & 0 & X & 0 & X \\ X & X & X & 0 & X & 0 \\ 0 & 0 & X & X & X & 0 \\ X & 0 & 0 & X & 0 & X \\ X & 0 & X & 0 & X & X \\ X & X & X & 0 & 0 & X \end{bmatrix}$$

If we take as in Rose and Tarjan [2] an ordering of pivots down the main diagonal so  $\phi = [(x_1, y_1), (x_2, y_2), \dots, (x_6, y_6)]$ , where  $(x_i, y_j)$  corresponds to the entry  $m_{ij}$ , then the fill-in  $\mathbf{F}$  produced by this ordering contains 9 edges. However, in this example, a different ordering of pivots, again only choosing entries from the main diagonal, produces a minimal fill-in of  $\mathbf{F}$  with 5 edges. Finally, if we choose from among all the entries in the matrix instead of only the entries on the diagonal, there is an ordering which produces a minimal fill-in of  $\mathbf{F}$  which has only 3 edges. So it is clear that this algorithm, which chooses entries from among all entries in the matrix, is capable of producing the smallest minimal fill-ins.

ACKNOWLEDGEMENT. This research was performed at The College of William and Mary, Williamsburg, Virginia, in Summer 1992, funded by NSF REU Grant DMS 91-00894.

#### REFERENCES

- [1] M.C. Golumbic, *Algorithmic Graph Theory and Perfect Graphs*, Academic Press, New York, 1980.
- [2] D.J. Rose and R.E. Tarjan, Algorithmic aspects of vertex elimination on directed graphs, *SIAMJ.Appl.Math.*34 (1978), 176-197.
- [3] D.J. Rose, R.E. Tarjan, and G.S. Lueker, Algorithmic aspects of vertex elimination on graphs, *SIAMJ.Comput.*5 (1976), 266-283.
- [4] M. Yannakakis, Computing the minimum fill-in is NP- complete, *SIAMJ.Alg.Disc. Math.*2 (1981), 77-79.



# Hindawi

Submit your manuscripts at  
<http://www.hindawi.com>

