PARTITIONING USING PAQ[+]

Thomas D. Howell

TR 75-256

August 1975

Department of Computer Science
Cornell University
Ithaca, New York 14853

[+]This research was supported in part by the Office of Naval Research Grant N00014-67-A-0077-0021, by a National Science Foundation Graduate Fellowship and by Argonne National Laboratory.

PARTITIONING USING PAQ[+]

Thomas D. Howell

Abstract:

The so-called PAQ problem is concerned with the solution of sparse systems of linear equations Ax = b using the transformation PAQy = Pb, x = Qy. An algorithm is given for choosing P and Q to partition the matrix A into its irreducible components. A theorem on which this algorithm is based has long been known, yet no simple, easily understood proof appears in the literature. Such a proof is given here. Remarks are made concerning some unsolved problems related to the PAQ problem.

[+]This research was supported in part by the Office of Naval Research Grant N00014-67-A-0077-0021, by a National Science Foundation Graduate Fellowship and by Argonne National Laboratory.

PARTITIONING USING PAQ[+]

Thomas D. Howell

1. Introduction

Let A be a large, sparse, nonsingular, nonsymmetric matrix. Consider solving the system of linear equations $Ax = b$ by Gaussian elimination. It is often advantageous to permute the equations and variables and solve instead, $(PAQ)y = Pb$, $x = Qy$, where P and Q are permutation matrices. The permutations P and Q may then be chosen to minimize the cost of performing Gaussian elimination on PAQ. This cost may be measured by the number of storage locations or the number of arithmetic operations used, or a combination of the two.

The choice of P and Q may be viewed as a two step process. The first step is to partition the matrix A. This is accomplished by finding P_0 and Q_0 such that P_0AQ_0 is block upper triangular:

$$P_0AQ_0 = \begin{bmatrix} A_{11} & A_{12} & \cdots & A_{1k} \\ 0 & A_{22} & \cdots & A_{2k} \\ \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \\ 0 & \cdots & 0 & A_{kk} \end{bmatrix} . \qquad (1)$$

Each A_{ii} is a nonsingular matrix which can not be further partitioned.

[+]This research was supported in part by the Office of Naval Research Grant N00014-67-A-0077-0021, by a National Science Foundation Graduate Fellowship and by Argonne National Laboratory.

The second step is to find $\bar{P}_i$ and $\bar{Q}_i$, $1 \leq i \leq k$, which permute the equations and variables of the i-th block in a way which minimizes the cost of performing elimination on $\bar{P}_i A_{ii} \bar{Q}_i$. Then,

$$PAQ = P_k \cdots P_1 P_0 A Q_0 Q_1 \cdots Q_k = \begin{bmatrix} \bar{A}_{11} & \bar{A}_{12} & \cdots & \bar{A}_{1k} \\ 0 & & & \\ \cdot & \bar{A}_{22} & \cdots & \bar{A}_{2k} \\ \cdot & \cdot & & \cdot \\ \cdot & \cdot & \cdot & \cdot \\ & & \cdot & \cdot\cdot \\ 0 & \cdots & 0 & \bar{A}_{kk} \end{bmatrix},$$

where P_i and Q_i are $\bar{P}_i$ and $\bar{Q}_i$ extended to the size of A.

The partitioning step may be accomplished as follows:

1. Find a permutation matrix R so that the main diagonal of RA is nonzero.

2. Form the directed graph G(RA) whose adjacency matrix is RA.

3. Find the strong components $C_1,\ldots,C_k$ of G(RA) and order them so that for i<j no path exists from C_j to C_i. Let Q_0^t be a permutation which corresponds to this ordering of vertices.

4. Now, $Q_0^t RAQ_0 \equiv P_0 A Q_0$ is block upper triangular with diagonal blocks $A_{11},\ldots,A_{kk}$ corresponding to components $C_1,\ldots,C_k$ of G(RA).

This algorithm appears in the literature at least as early as 1962, [1], [2]. The proof of its correctness depends on a theorem which states that any permutation used in step 1 to make the main diagonal nonzero gives rise to the same vertex partition in step 3, although the components themselves may be different. Johnson, Dulmage and Mendelsohn [1] prove a related theorem which has the above theorem as a consequence. Their proof, which is given in graph theoretic terms, is difficult to follow and to apply to the partitioning of sparse matrices. The idea for a simpler proof of this theorem is given by Steward [2]. His argument is only intuitive; he does not state a theorem or give a formal proof. The main result of this paper is to give a new, short, simple proof of the theorem mentioned above.

2. Preliminary Remarks

The reason partitioning is important is that it allows a large system of equations to be replaced by several smaller systems. Suppose P_0 and Q_0 have been found so that P_0AQ_0 is in the form of equation (1). Let y and P_0b be partitioned into $y_1,\ldots,y_k$ and $(P_0b)_1,\ldots,(P_0b)_k$ to correspond to the partitioning of P_0AQ_0 in equation (1). The system $(P_0AQ_0)y = P_0b$, $x = Q_0y$ may now be solved by block back substitution:

$$
\begin{aligned}
A_{kk}\, y_k &= (P_0b)_k \\
&\vdots \\
A_{ii}\, y_i &= (P_0b)_i - \sum_{j=i+1}^{k} A_{ij}\, y_j
\end{aligned}
$$

$$\vdots$$

$$A_{11}\, y_1 = (P_0 b)_1 - \sum_{j=2}^{k} A_{1j}\, y_j$$

$$x = Q_0\, y$$

Now we need only apply Gaussian elimination to the diagonal blocks A_{ii}, $1 \leq i \leq k$. This is certainly no more costly than doing elimination on the entire matrix, and can be much less costly if the partitioning is fine.

Let $R = QP$. Then $PAQ = Q^tRAQ$. The permutation matrices R and Q then have the following interpretations: R permutes the rows of A so that the elements which will be used as "pivot elements" for Gaussian elimination lie on the main diagonal of RA. The symmetric permutation $Q^t(RA)Q$ then determines the order in which these pivot elements will be used. Once R and Q have been chosen, Gaussian elimination is applied to the diagonal blocks of matrix Q^tRAQ with no further pivoting. For this reason we will almost always consider only those permutations R for which the main diagonal of RA (hence also of Q^tRAQ) is entirely nonzero. It would be possible to leave some zeroes on the main diagonal of RA and trust the elimination process to "fill them in" (change them to nonzero) before they are used as pivot elements. We will see in Section 5 that this can only cause the partition generated to be coarser than that which would be generated using a nonzero main diagonal.

The second step in the choice of P and Q is very difficult in general. It requires finding $\bar{P}_i$ and $\bar{Q}_i$ to minimize the cost of Gaussian elimination on the diagonal block A_{ii}. No algorithm

is known which finds the optimum permutations without essentially trying all possible permutations. This problem will be discussed further in Section 6.

As mentioned above, the cost of Gaussian elimination may be measured in a variety of ways. A simple, commonly used cost function is called "fill-in". The fill-in is the number of new nonzero elements created during the elimination process. These elements are said to have been "filled in". Fill-in determines the number of additional storage locations required to perform Gaussian elimination on A. This cost function will be used for the remainder of this paper.

3. Definitions

Definition 1: A diagonal of the nxn matrix A is any set of n elements containing one element from each row and column of A. The set $\{a_{ii} \mid 1 \leq i \leq n\}$ is the main diagonal of A.

Because A is nonsingular, its determinant is not zero, and therefore at least one diagonal of A has all of its elements nonzero.

Definition 2: Let π be a nonzero diagonal of A. The directed graph associated with A and π is $G(A,\pi)$ which is defined as follows: Let R and S be permutation matrices such that the elements of π form the main diagonal of RAS. Then $G(A,\pi) = (V,E)$ where $V = \{v_i \mid 1 \leq i \leq n\}$ and $E = \{(v_i,v_j) \mid i \neq j \text{ and } (RAS)_{ij} \neq 0\}$. When π_0 is the main diagonal of A, $G(A,\pi_0)$ will also be written simply as $G(A)$.

It remains to show that $G(A,\pi)$ is well defined, i.e., that $G(A,\pi)$ is unaffected by the choice of R and S. Note that all graphs used in this paper are directed graphs.

Theorem 1: Let $G_1(A,\pi)$ and $G_2(A,\pi)$ be the graphs described by Definition 2 using pairs of permutations (R_1,S_1) and (R_2,S_2) respectively. Then $G_1(A,\pi)$ and $G_2(A,\pi)$ are isomorphic.

Proof: Let Π be the nxn permutation matrix whose 1's are in the positions corresponding to the locations of the elements of π in A. Since (R_1,S_1) and (R_2,S_2) move π to the main diagonal, $R_1\Pi S_1 = R_2\Pi S_2 = I$. Then $S_1 = \Pi^t R_1^t$, $S_2 = \Pi^t R_2^t$. Let $C = R_2 R_1^t$. Then,

$$\begin{aligned} C(R_1AS_1)C^t &= R_2R_1^tR_1AS_1R_1R_2^t \\ &= R_2A\Pi^tR_2^t \\ &= R_2AS_2 \end{aligned}$$

The matrices R_1AS_1 and R_2AS_2 are the same up to a renumbering of the rows and columns by the permutation C, hence $G_1(A,\pi) \cong G_2(A,\pi)$ with isomorphism given by C.

A consequence of Theorem 1 is that with no loss of generality we may always move the pivot elements to the main diagonal using row permutations only, i.e., using $R = \Pi^t$, $S = I$.

Definition 3: A directed graph is strongly connected if there is a path from each vertex to each other vertex. A directed graph which is not strongly connected can be divided in a unique way into maximal strongly connected subgraphs which are called strong components.

These strong components $C_1,\ldots,C_k$ can be ordered in such a way that for $i<j$ no path exists from vertices in C_j to vertices in C_i. An efficient algorithm for finding the strong components in this order has been given by Tarjan [3]. This algorithm can also be found in [4], as can definitions of other graph theoretic terms used here with which the reader may not be familiar.

Definition 4: If $PAQ = \begin{bmatrix} A_{11} & A_{12} \\ 0 & A_{22} \end{bmatrix}$ where A_{11} and A_{22} are square and of order at least one, and P and Q are permutation matrices, then A is bireducible. Otherwise A is fully indecomposable. If

$$PAQ = \begin{bmatrix} A_{11} & A_{12} & \cdots & A_{1k} \\ 0 & A_{22} & \cdots & A_{2k} \\ \cdot & \cdot & \cdot & \cdot \\ \cdot & & \cdot & \cdot \\ \cdot & & \cdot & \cdot \\ 0 & \cdots & 0 & A_{kk} \end{bmatrix},$$

and each A_{ii} is fully indecomposable, then the submatrices A_{ii}, $1 \leq i \leq k$, are the indecomposable components of A.

Definition 5: Let A be an nxn matrix with nonzero main diagonal. Let G(A) be its graph. We say that A (or G(A)) has property C if the following condition holds:

Matrix statement - Every subset of $k < n$ rows of A has nonzero entries in at least k+1 columns.

Graph statement - Every subset of $k < n$ vertices of G(A) has at least one directed edge from a vertex in the subset to a vertex outside the subset.

4. Main Theorem

Theorem 2: Let A be an nxn matrix with nonzero main diagonal and G(A) be its graph. The following three statements are equivalent:

(1) A has property C.

(2) A is fully indecomposable.

(3) G(A) is strongly connected.

Proof: not (1) => not (2) : Let S be a subset of rows of A on which property C fails. Let P be a permutation matrix which orders rows in S last. Then

$$PAP^t = \begin{bmatrix} A_{11} & A_{12} \\ 0 & A_{22} \end{bmatrix} \} \text{ rows in S} \quad ,$$

and A is bireducible.

not (2) => not(1) : Let $PAQ = \begin{bmatrix} A_{11} & A_{12} \\ 0 & A_{22} \end{bmatrix}$.

Then property C fails on the rows of A_{22}, hence also on the rows of A from which they were formed.

not (1) => not(3) : Let S be a subset of rows of A on which property C fails.

Then no path exists in G(A) from vertices corresponding to rows in S to vertices outside this set, hence G(A) is not strongly connected.

not (3) => not(1) : If G(A) is not strongly connected it must have a proper strong component C_k with no outward edges. The rows of A corresponding to vertices in C_k have nonzero entries only in columns corresponding to vertices in C_k, violating property C.

Corollary: Let R and S be permutation matrices such that RA and SA both have nonzero main diagonal. Then G(RA) is strongly connected if and only if G(SA) is strongly connected.

Proof: By Theorem 2 property C is equivalent to strong connectedness. Property C depends only on the set of rows in the matrix not on their order. The matrices RA and SA have the same set of rows.

The essence of the main theorem is contained in Theorem 2 and its corollary. It remains only to apply these results to matrices whose graphs have several strong components.

Theorem 3: Every nonzero diagonal π of A induces the same strong component vertex partition in $G(A,\pi)$.

Proof: The proof is by induction on the number of strong components of $G(A,\pi)$. Let k be this number, and let the components be $C_1,\ldots,C_k$ where no path exists from C_j to C_i

with $i<j$. The corollary to Theorem 2 takes care of the case $k=1$. For the inductive step assume the theorem holds whenever $G(A,\pi')$ has $k-1$ components for some nonzero diagonal π'. Let X be the set of indices of the vertices in C_k, the last strong component. Let $\bar{X}$ be the rest of the indices: $X \cup \bar{X} = \{1, \ldots, n\}$. Let A_X and $A_{\bar{X}}$ be square submatrices of A consisting of the rows and columns whose indices are in X and $\bar{X}$ respectively. Let ρ be another diagonal of A. Let π_X, $\pi_{\bar{X}}$, ρ_X and $\rho_{\bar{X}}$ be the elements of π and ρ in A_X and $A_{\bar{X}}$ respectively. Consider the rows of A whose indices lie in X. Because C_k has no outward edges in $G(A,\pi)$, the only nonzero entries in these rows are in A_X. Since ρ_X contains one nonzero entry from each of these rows, ρ_X forms a diagonal of A_X. It follows that $\rho_{\bar{X}}$ forms a diagonal of $A_{\bar{X}}$. The inductive hypothesis implies that $G(A_{\bar{X}},\rho_{\bar{X}})$ has the same strong component vertex partition as $G(A_{\bar{X}},\pi_{\bar{X}})$. The corollary to Theorem 2 implies that $G(A_X,\rho_X)$ is strongly connected. The rows of A containing A_X are zero outside of A_X, so $G(A_X,\rho_X)$ remains as a separate strong component of $G(A,\rho)$. Therefore $G(A,\pi)$ and $G(A,\rho)$ have the same strong component vertex partition, and the theorem follows by induction.

5. Time Bounds

Let A be a nonsingular nxn matrix with f(n) nonzero elements ($n \leq f(n) \leq n^2$). The partitioning algorithm can be rewritten as follows:

1. Find a nonzero diagonal π and permutation R moving π to the main diagonal.

2. Form $G(A,\pi)$.

3. Find strong components of $G(A,\pi)$ in an order such that for $i<j$, no path exists from C_j to C_i. Let the permutation Q^t order row k before row ℓ whenever $V_k \in C_i$, $V_\ell \in C_j$, and $i<j$.

4. Form $PAQ \equiv Q^tRAQ$ which is block upper triangular.

Steps 2 and 4 are straight forward and take time and storage bounded by a constant multiple of f(n). Step 3 can be accomplished using the algorithm of Tarjan also in time and storage proportional to f(n). The fastest known algorithm to do step 1 in general is due to Hopcroft and Karp [5] and requires time proportional to $n^{2.5}$. This may be too costly to be feasible for very large values of n, however for many specific problems a diagonal may be found more easily. In many cases A has a ready-made nonzero main diagonal.

Remark: Suppose A has a zero on its main diagonal which will be filled in by the elimination process before it is used as a divisor. One could still construct G(A) and do the partitioning as before. By Theorem 3 this graph would have the same strong component vertex partition as the one generated as follows:

1. Color the zero on the main diagonal red.

2. Find R so that RA has nonzero main diagonal and and construct G(RA) as before.

3. Add an edge corresponding to the red zero which is now somewhere off the main diagonal of RA.

This added edge may or may not change the strong component structure of G(RA). It is possible for it to merge two or more components of G(RA) into a single component. For this reason it makes sense to use a nonzero main diagonal whenever it is feasible to find one.

6. Directions for further research

Consider for a moment the case in which A is symmetric and positive definite. Then A has a ready-made nonzero diagonal. As long as this main diagonal is used, which means that only reorderings of the form Q^tAQ are considered, symmetry and positive definiteness are preserved. Because Q^tAQ is positive definite, pivoting is not necessary to insure the numerical stability of Gaussian elimination. This leaves us free to choose Q to minimize the cost of doing the elimination on Q^tAQ. This case has been studied extensively by Rose and others [6], [7], [8].

When A is not symmetric, positive definite there is no reason to restrict reorderings to the form Q^tAQ, so we consider PAQ (or Q^tRAQ). In contrast to the previous case, the ready-made nonzero main diagonal and the guaranteed numerical stability have been lost, but the freedom to choose a nonzero main diagonal (via R) has been gained. The partitioning problem using PAQ is fairly easy, and a reasonably good solution has been described above. Theorem 3 shows that for partitioning

it does not matter which nonzero diagonal is chosen. This leaves us free to choose both P and Q for each indecomposable component of A in a way which attempts to solve two more difficult problems which we have ignored up to now.

One problem is numerical stability. In the case of dense matrices the main purpose of pivoting is to insure stability. In the sparse, positive definite, symmetric case pivoting is used to minimize the cost of Gaussian elimination. In the sparse, unsymmetric case one would like to use the freedom to choose both P and Q in order to achieve both goals: stability and low cost. This paper has ignored the first goal. By distinguishing only between zero and nonzero numbers we have factored out the problem of numerical stability. Any practical algorithm for the PAQ problem will have to face this problem.

The second difficult problem is concerned with the goal of low cost. How should $\bar{P}_i$ and $\bar{Q}_i$ be chosen to minimize the cost of doing elimination on $\bar{P}_i A_{ii} \bar{Q}_i$ where A_{ii} is an indecomposable component of A? Rose and Tarjan [9] have shown that after $\bar{R}_i = \bar{Q}_i \bar{P}_i$ has been chosen so that $\bar{R}_i A_{ii}$ has a nonzero main diagonal, the problem of choosing $\bar{Q}_i$ to minimize the cost (measured in terms of fill-in) of doing elimination on $\bar{Q}_i^t R_i \bar{A}_{ii} \bar{Q} \equiv \bar{P}_i A_{ii} \bar{Q}_i$ is NP-complete. For a discussion of the meaning of NP-complete see [4], [10], or [11]. The problem of choosing both $\bar{P}_i$ and $\bar{Q}_i$ to minimize cost seems at least as hard as choosing $\bar{Q}_i$ given $\bar{R}_i$; it would be interesting to learn whether this problem is also NP-complete. The problem is hard

enough that practical algorithms probably will have to use heuristics to help find "good" permutations $\bar{P}_i$ and $\bar{Q}_i$ rather than insisting on finding the best ones. One strategy is to fix $\bar{R}_i$, then choose $\bar{Q}_i$ using heurstics developed for the symmetric Q^tAQ problem. Two such heuristics are "minimum degree" and "minimum deficiency", and are discussed in [6].

7. Examples

The first example will illustrate the entire process of solving Ax = b using the method described above.

$$n = 5 \qquad \begin{bmatrix} 2 & 0 & 4 & 0 & 0 \\ 1 & 1 & 0 & 0 & 3 \\ 0 & 0 & 3 & 4 & 0 \\ 2 & 2 & 0 & 1 & 5 \\ 0 & 0 & 1 & 1 & 0 \end{bmatrix} \qquad x = \begin{bmatrix} 0 \\ 2 \\ 2 \\ 2 \\ 1 \end{bmatrix}$$

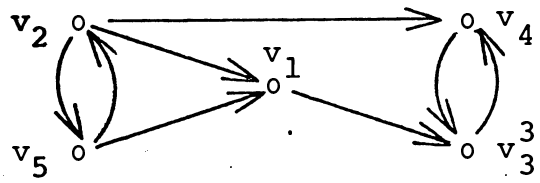
$$Ax = b$$

1. Choose any nonzero diagonal, say $\pi = (a_{11}, a_{25}, a_{34}, a_{42}, a_{53})$. Let R move it to the main diagonal.

$$R = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \end{bmatrix}$$

$$RAx = Rb \qquad \begin{bmatrix} 2 & 0 & 4 & 0 & 0 \\ 2 & 2 & 0 & 1 & 5 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 3 & 4 & 0 \\ 1 & 1 & 0 & 0 & 3 \end{bmatrix} \qquad x = \begin{bmatrix} 0 \\ 2 \\ 1 \\ 2 \\ 2 \end{bmatrix}$$

2. Form $G(A,\pi) = G(RA)$



3. Find strong components in order. Let Q^t reflect this order.

$$\{v_2,v_5\}\ ,\ \{v_1\},\ \{v_3,v_4\}$$

$$Q^t = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \end{bmatrix} \qquad Q = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 \end{bmatrix}$$

4. Form $PAQ = Q^tRAQ$.

$$PAQy = Pb \left[\begin{array}{cc|c|c} 2 & 5 & 2 & 0 \\ 1 & 3 & 1 & 0 \\ \hline 0 & 0 & \frac{1}{2} & 4 \\ \hline 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 3 \end{array} \begin{array}{c} 1 \\ 0 \\ \hline 0 \\ \hline 1 \\ 4 \end{array}\right] y = \begin{bmatrix} A_{11} & A_{12} & A_{13} \\ 0 & A_{22} & A_{23} \\ 0 & 0 & A_{33} \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix} = \left[\begin{array}{c} 2 \\ 2 \\ \hline 0 \\ \hline 1 \\ 2 \end{array}\right]$$

5. Block back substitution.

$$A_{33}y_3 = (Pb)_3$$

$$\begin{bmatrix} 1 & 1 \\ 3 & 4 \end{bmatrix} y_3 = \begin{bmatrix} 1 \\ 2 \end{bmatrix} \quad \Rightarrow \quad y_3 = \begin{bmatrix} 2 \\ -1 \end{bmatrix}$$

$$A_{22}y_2 = (Pb)_2 - A_{23}y_3$$

$$\begin{bmatrix} 2 \end{bmatrix} y_2 = \begin{bmatrix} -8 \end{bmatrix} \quad \Rightarrow \quad y_2 = \begin{bmatrix} -4 \end{bmatrix}$$

$$A_{11}y_1 = (Pb)_1 - A_{12}y_2 - A_{13}y_3$$

$$\begin{bmatrix} 2 & 5 \\ 1 & 3 \end{bmatrix} y_1 = \begin{bmatrix} 11 \\ 6 \end{bmatrix} \quad \Rightarrow \quad y_1 = \begin{bmatrix} 3 \\ 1 \end{bmatrix}$$

$$y = \begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix} = \begin{bmatrix} 3 \\ 1 \\ -4 \\ 2 \\ -1 \end{bmatrix} \qquad x = Qy = \begin{bmatrix} -4 \\ 3 \\ 2 \\ 1 \\ -1 \end{bmatrix}$$

Remark: If we had used the main diagonal of A without applying the permutation R to make it nonzero, the resulting graph would have been strongly connected, and no partitioning would have resulted.

The second example shows that the choice of a nonzero main diagonal can affect the cost of Gaussian elimination on a component of A. Suppose,

$$A_{ii} = \begin{bmatrix} x & x & 0 & x \\ 0 & x & x & 0 \\ 0 & x & x & x \\ x & 0 & 0 & x \end{bmatrix} \qquad x = \text{arbitrary nonzero entry}$$

The choice $\bar{R}_i = I$ leads to fill-in of at least one, regardless of the choice of $\bar{Q}_i$. The choice,

$$\bar{R}_i = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix},$$

leads to no fill-in if $\bar{Q}_i$ is chosen to be I, for example.

In the preceeding example A_{ii} was nonsymmetric. It has been conjectured that when the zero-nonzero structure of A_{ii} is symmetric with nonzero main diagonal the cost cannot be reduced by using a diagonal such that the zero-nonzero structure of $\bar{R}_i A_{ii}$ is nonsymmetric instead of one for which it is symmetric. While not disproving this conjecture, a final example will show that the situation is not as simple as might be imagined. Consider:

$$A = R_0A = \begin{bmatrix} x & 0 & x & x & x \\ 0 & x & x & x & x \\ x & x & x & 0 & 0 \\ x & x & 0 & x & 0 \\ x & x & 0 & 0 & x \end{bmatrix} \qquad R_1A = \begin{bmatrix} x & x & x & 0 & 0 \\ 0 & x & x & x & x \\ x & 0 & x & x & x \\ x & x & 0 & x & 0 \\ x & x & 0 & 0 & x \end{bmatrix} \qquad R_2A = \begin{bmatrix} x & x & x & 0 & 0 \\ x & x & 0 & x & 0 \\ x & 0 & x & x & x \\ 0 & x & x & x & x \\ x & x & 0 & 0 & x \end{bmatrix}$$

m	R_0A	R_1A	R_2A
2	36	12	0
3	0	48	48
4	36	12	48
5	0	24	24
6	48	24	0
Avg. cost	4.2	4.0	3.8
Min. cost	2	2	3

Each entry of the table gives the number of the 120 possible permutations Q such that the cost for $Q^t R_i A Q$ is m. The cost is still measured by fill-in.

The average cost over all permutations Q can be reduced below that of R_0A by using a nonsymmetric permutation R_1A, even though the minimum cost over all Q is unaffected. The average cost can be reduced still further by using another nonsymmetric permutation R_2A, yet in this case the minimum cost actually rises!

8. Conclusion

An algorithm has been given for choosing permutation matrices P_0 and Q_0 to partition a sparse matrix A into its indecomposable components. A new proof of a theorem underlying this algorithm has been given which is simpler and clearer than the original proof. Bounds on the running time of the algorithm are stated and an example is given illustrating the algorithm. Areas for further research are indicated, and some remarks are made concerning the conjecture that for symmetric matrices reorderings of the form Q^tAQ can do as well as those of the form PAQ.

References

[1] Johnson, D.M., A.L. Dulmage, and N.S. Mendelsohn, "Connectivity and Reducibility of Graphs", Can. J. Math., 14 (1962), pp. 529-439.

[2] Steward, D.V., "On an Approach to Techniques for the Analysis of the Structure of Large Systems of Equations", SIAM Review, 4 (1962), pp. 321-342.

[3] Tarjan, R.E., "Depth First Search and Linear Graph Algorithms", SIAM J. Computing, 1 (1972), pp. 146-160.

[4] Aho, A.V., J.E. Hopcroft, and J.D. Ullman, The Design and Analysis of Computer Algorithms, Addison-Wesley, Reading, Mass. (1974), pp. 189-195.

[5] Hopcroft, J.E. and R.M. Karp, "An $n^{5/2}$ algorithm for Maximum Matchings in Bipartite Graphs", SIAM J. Computing, 2 (1972).

[6] Rose, D.J., "A Graph-theoretic Study of the Numerical Solution of Sparse Positive Definite Systems of Linear Equations", Graph Theory and Computing, R. Read, ed., Academic Press, N.Y. (1973), pp. 183-217.

[7] Bunch, J.R. and D.J. Rose, "Partitioning, Tearing and Modification of Sparse Linear Systems", J. Math. Anal. and Applications, 48 (1974), pp. 574-593.

[8] Rose, D.J., and R.E. Tarjan, "Algorithmic Aspects of Vertex Elimination", Proceedings Seventh Annual ACM Symposium on Theory of Computing (1975), pp. 245-254.

[9] Rose, D.J. and R.E. Tarjan, "Algorithmic Aspects of Vertex Elimination on Directed Graphs", joint Harvard-Stanford Technical Report.

[10] Cook, S., "The Complexity of Theorem-proving Procedures", Proceedings Third Annual ACM Symposium on Theory of Computing (1975), pp. 245-254.

[11] Karp, R.M., "Reducibility Among Combinatorial Problems", Complexity of Computer Computations, R.E. Miller and J. W. Thatcher, eds., Plenum Press, N.Y. (1972), pp. 85-104.