A GRAPH-THEORETIC STUDY OF THE NUMERICAL SOLUTION OF SPARSE POSITIVE DEFINITE SYSTEMS OF LINEAR EQUATIONS

Donald J. Rose[†]

Department of Mathematics
University of Denver
Denver, Colorado

† Present address: Aiken Computation Laboratory, Harvard University, Cambridge, Massachusetts.

1. Introduction

The necessity to solve linear systems

$$(1) \qquad Mx = b,$$

where M is an $n \times n$ sparse† symmetric positive-definite matrix, arises frequently in physical applications. These include classical electrical network analysis, analysis of structural systems, and nonlinear hydraulic problems. In such problems understanding and controlling sparsity is essential for efficient computer solution, since, in general, many systems with the same zero–nonzero structure will be solved.

To solve systems like (1) by elimination, it is standard procedure (Forsythe and Moler [12] and Westlake [27]) to decompose, or factor, M as

$$(2) \qquad M = GG^{\mathrm{T}} \quad \text{or} \quad M = LDL^{\mathrm{T}},$$

where G and L are lower triangular, $L = (l_{ij})$ with $l_{ii} = 1$, and $D = (d_{ij})$ is diagonal with $d_{ii} > 0$. Since M is symmetric and positive definite, we may decompose M for any *a priori* ordering of the linear system, that is,

$$A = PMP^{\mathrm{T}},$$

where P is an $n \times n$ permutation matrix. Experience and simple examples show that the choice of ordering is an important consideration in obtaining an efficient elimination scheme. Such considerations lead to the study of the following basic question: what is the effect of the order of elimination upon sparse positive-definite systems?

We have restricted our class of matrices to those M which are symmetric positive definite, because this allows us to examine only the equivalence class PMP^{T} rather than the class PMQ, P and Q being permutation matrices. Equally important is the fact that the decompositions of (1), especially the Cholesky $M = GG^{\mathrm{T}}$ decomposition, are stable with respect to rounding error, see Wilkinson [28, pp. 220, 231–232, 244], for any *a priori* ordering P. Formally, our analysis extends to the more general class of matrices M, such that PMP^{T} can be decomposed as

$$PMP^{\mathrm{T}} = LDU = LU'$$

for any permutation matrix P, and such that $m_{ij} \neq 0 \leftrightarrow m_{ji} \neq 0$. Here, of course, both L and U must be stored.

We use the word "formally" above because the decomposition must be computed in finite precision arithmetic. In the more general case of sparsity

† We leave *sparse* undefined formally, but informally we think of matrices M, many of whose entries m_{ij} are zero.

symmetry above, and in the case where M is simply a nonsingular matrix row, or a column, interchanges are usually effected (see Wilkinson [28, p. 205] or Forsythe and Moler [12, p. 34]), to avoid zero pivot elements and to maintain stability with respect to round-off error. Whereas the decomposition of PMP^{T} for any P is stable when M is positive definite, it is clear that in these more general cases pivoting to control stability and pivoting to control sparsity are not *a priori* compatible.

With M as in (1) we associate an undirected graph G. In Section 2 we formulate the elimination process of decomposition (2) as vertex elimination on this graph. We call this formulation the *combinatorial elimination process*. In addition to formulating the elimination process as vertex elimination on a graph, we seek a graph-theoretic description of those matrices M, such that in the decomposition L or G has exactly the same zero–nonzero structure as the lower triangular part of M. We call such graphs *monotone transitive graphs*.

The suggestion that graph theory might be a convenient way to study elimination is due to Parter [17], although he does not pursue a detailed graph-theoretical study. He analyzes the special case when the matrix M has a graph-theoretic representation as a tree, and he shows that trees can be ordered so that they are monotone transitive. A well-known example of the simple elimination scheme which results from choosing a monotone transitive ordering for such a graph, or matrix, is the case when M is tridiagonal.

There are two interesting implications of Parter's study which were not pursued in the literature. First, since trees are without cycles and can be ordered to be monotone transitive, we are led to investigate monotone transitivity in more general graphs by studying their cycle structure. Second, although a tridiagonal band matrix is represented by a special tree, Parter's elimination scheme applies to any matrix represented by a tree. The elimination process involved has absolutely nothing to do with the bandwidth of the matrix M. This is significant given the recent activity in minimizing bandwidth for sparse matrix calculations [1, 2, 9, 19].

Following the formulation of elimination as a combinatorial process, in Section 3 we gain considerable insight into the elimination process by studying the evolution of the cycle structure and the vertex-separator, or cut-set, structure of a graph under elimination. We show that monotone transitive graphs are triangulated graphs, and conversely, as defined by Berge [3]. This is a cycle characterization. In addition, we characterize monotone transitive graphs by a property of their separators.

In Section 4 we study criteria from which we may define best or good orderings. By counting the arithmetic operations necessary to effect the decompositions, we relate these criteria for optimization to the computational complexity of calculations involving the elimination process. We note that in the literature the study of optimal ordering contains many subjective decisions and implicit assumptions which are not always clearly presented. One reason for some of the confusion which exists is that for practical applications there is an implicit constraint that any ordering algorithm to be used be reasonably efficient. Otherwise, such an algorithm may become too time- or storage-consuming to be feasible. Several authors (Tinney and Walker [26], Tinney [29, p. 25], and Tewarson [29, p. 35]) have developed algorithms which have been partially successful in producing good orderings but which do not, in general, produce optimal orderings for the criterion they choose. Most of this work has been experimental. In Section 4 we discuss these algorithms in view of the results developed in Section 3 and the results of our generalized study of criterion functions.

Another interesting graph-theoretic approach for dealing with sparse systems with respect to Gaussian elimination is to attempt to find permutation matrices P, Q such that

$$A = PMQ \tag{3}$$

is block lower triangular, since in this case it is necessary only to decompose the diagonal blocks of PMQ. Naturally such a transformation does not preserve symmetry. Harary [13, 14] solves this problem algorithmically with the restriction that $Q = P^{\mathrm{T}}$. His results have application to the algebraic eigenvalue problem. Steward [21] and Dulmage and Mendelsohn [10, 11] have solved the more general problem and have algorithms for producing P and Q. These results are not applicable when M is symmetric positive definite and irreducible, since the algorithm would then produce only one diagonal block, M itself. Even when applicable, this theory does not differentiate between reorderings of the system within the diagonal blocks.

Our interest in sparse linear systems was motivated initially by its application to the potential flow network problem [18]. Several examples of sparse linear systems arising in applications can also be found in [29]. Other theoretical considerations on sparse linear systems and numerical linear algebra are reported in Brayton *et al.* [6]. Finally, we wish to emphasize the importance of new approaches to data handling and efficient use of memory hierarchy which are required for the successful machine implementation of sparse matrix methods. While we do not discuss computer implementation here, considerable progress on this aspect of sparse matrix research is reported in Gustavson *et al.* [15].

2. The Elimination Process

In this section, we study the combinatorial nature of the elimination process upon sparse symmetric positive-definite matrices. We will see that it is useful to regard elimination as vertex elimination on a graph. We first review the well-known LDL^T decomposition theorem for positive-definite matrices where L is a lower triangular matrix with unit diagonal elements, and D is a diagonal matrix with positive nonzero entries.

Unfortunately, sparse matrices tend to *fill in* during elimination. That is to say, in general, the number of nonzeros in L of the decomposition $M = LDL^T$ is greater than the number of nonzeros in the lower triangular part of M. In Section 2.2 we ask for the class of matrices so that we can find an ordering (permutation P) such that no zeros are lost in the decomposition of PMP^T. Of course, this class of matrices is special, but we will see that, after elimination, any matrix is transformed into a matrix of this special class. More precisely, we will see that L^T has this property. This question leads us to our notions of elimination graph, monotone transitivity, and perfect elimination processes.

2.1. Decompositions

We begin by stating a well-known (Forsythe and Moler [12, pp. 27–29]), theorem of numerical linear algebra. Let M be a real $n \times n$ matrix and let M_k, $k = 1,\dots,n-1$ denote the principal submatrices of M consisting of the first k rows and columns of M.

THEOREM 1. Let M and M_k be as above, and assume $\det(M_k) \neq 0$, $k = 1, 2,\dots, n-1$. Then there exist unique matrices L, D, U, such that

$$M = LDU, \tag{4}$$

where $L = (l_{ij})$ and $U = (u_{ij})$ are real $n \times n$ unit lower ($l_{ii} = 1$) and unit upper ($u_{ii} = 1$) triangular matrices, respectively; and D is an $n \times n$ real diagonal matrix.

If M is a real symmetric positive-definite matrix, PMP^T satisfies the hypothesis of the theorem for any permutation matrix P. Furthermore, by uniqueness it follows that

$$M = LDL^T, \tag{5}$$

In this case, D has positive diagonal entries. Also,

$$M = GG^T, \tag{6}$$

where $G = LD^{1/2}$. The factorization (6) is due to Cholesky [12, p. 114].

For sparse matrices M, it is significant that L, D, and G of (5) and (6) are unique. This means that the zero–nonzero structure of M uniquely determines the zero–nonzero structure of L or G, independent of the method used to compute L, D, or G. Note also that since M is symmetric, the decompositions (5) or (6) are more efficient than the $M = L'U'$ decomposition, where only

L' or U' is unit triangular, because for the decompositions (5) and (6) we need only store the upper triangular part of M and either L and D of (5) or G of (6).

Since the symmetric Gaussian elimination scheme and Cholesky's method are the two most generally accepted methods for obtaining (5) and (6) respectively, we state them now in algorithmic form which we will need in Section 2.2 and Section 4.

SYMMETRIC GAUSSIAN ELIMINATION

This method is also known as the method of congruent transformations (see Westlake [27, p. 21]). To explain the algorithm, it suffices to exhibit the first major step since the algorithm then proceeds recursively on lower order principal submatrices. Let the $n \times n$ positive definite matrix

$$M^{(1)} = \begin{bmatrix} a & r^T \\ r & \overline{M} \end{bmatrix},$$

where a is 1×1, r is $(n-1) \times 1$, and $\overline{M}$ is $(n-1) \times (n-1)$. Then

$$\begin{aligned} M^{(1)} &= \begin{bmatrix} 1 & 0 \\ r/a & I \end{bmatrix} \begin{bmatrix} a & 0 \\ 0 & \overline{M} - rr^{\mathrm{T}}/a \end{bmatrix} \begin{bmatrix} 1 & r^{\mathrm{T}}/a \\ 0 & I \end{bmatrix} \\ &= L_1 \begin{bmatrix} a & 0 \\ 0 & M^{(2)} \end{bmatrix} L_1^{\mathrm{T}}, \end{aligned} \tag{7}$$

where $M^{(2)} = \overline{M} - rr^{\mathrm{T}}/a$ and I is the $(n-1) \times (n-1)$ identity matrix. Note that $M^{(2)}$ is positive definite because

$$(-y^{\mathrm{T}}r/a \,|\, y^{\mathrm{T}})\, M^{(1)} \begin{pmatrix} -y^{\mathrm{T}}r/a \\ y \end{pmatrix} > 0$$

for any $(n-1)$ vector y. We may compute L^{T} and D which replace the strictly upper triangular and diagonal parts of $M^{(1)}$ respectively, by the algorithm

(8)

```
for i: = 1 step 1 until n-1 do;
for j: = i+1 step 1 until n do;
begin;
s: = M[i,j]/M[i,i];
for l: = j step 1 until n do;
M[j,l]: = M[j,l] - s × M[i,l];
M[i,j]: = s;
end.
```

To compute the Cholesky decomposition, note that (7) can be rewritten as

$$M = \begin{bmatrix} \sqrt{a} & 0 \\ r/\sqrt{a} & \tilde{G} \end{bmatrix} \begin{bmatrix} \sqrt{a} & r^{\mathrm{T}}/\sqrt{a} \\ 0 & \tilde{G}^{\mathrm{T}} \end{bmatrix},$$

where $\tilde{G}\tilde{G}^{\mathrm{T}}$ is the Cholesky factorization of $\overline{M} - rr^{\mathrm{T}}/a$ and the algorithm (8) can be changed appropriately. Usually, however, the elements of G are computed column by column, which requires exactly the same operations executed in a different order, as in the following algorithm from Forsythe and Moler [12, p. 114]

(9)
for $j = 1$ step 1 until n do;

begin $G[j,j] := \mathrm{sqrt}\left(M[j,j] - \sum_{k=1}^{j-1} G[j,k]^2\right)$;

for $i = j + 1$ step 1 until n do;

$G[i,j] := \left(M[i,j] - \sum_{k=1}^{j-1} G[i,k] \times G[j,k]\right) \Big/ G[j,j]$;

end.

Finally, to solve $Mx = b$ by symmetric Gaussian elimination, we solve

$$Lz = b, \tag{10}$$

$$Dy = z, \tag{11}$$

$$L^{\mathrm{T}}x = y. \tag{12}$$

Since (10) and (11) involve triangular systems, this is merely back-solving. Similarly for Cholesky's method, we compute

$$Gy = b, \tag{13}$$

$$G^{\mathrm{T}}x = y. \tag{14}$$

Let $M = (m_{ij})$ be an $n \times n$ symmetric positive-definite matrix with decomposition $M = LDL^{\mathrm{T}}$. It is clear, because the unique decomposition can be generated by (7), that the set of pairs $\{i,j\}$ with $l_{ij} = 0$ is, in general, a subset of the pairs with $m_{ij} = 0$, that is, the triangular factor L cannot, in general, be more sparse than the lower triangular part of M.

Given M, if there exists a permutation matrix P such that

$$A = PMP^{\mathrm{T}} = LDL^{\mathrm{T}}$$

and

$$a_{ij} = 0 \Rightarrow l_{ij} = 0, \qquad i > j, \tag{15}$$

then we say M is a *perfect elimination matrix*. It is straightforward from (7) (see also Parter [17, Theorem 1]) that A has this property (property P) if, and only if, for all $1 \leqslant i < j < k \leqslant n$

$$(16) \qquad a_{ij} \neq 0 \quad \text{and} \quad a_{ik} \neq 0 \Rightarrow a_{jk} \neq 0.$$

We give a graph theoretic interpretation of property P in the next section where we introduce monotone transitive graphs and perfect elimination processes.

2.2. *The Combinatorial Elimination Process*

We now temporarily abandon the arithmetic aspects of the elimination process in order to study its combinatorial nature. We begin by associating with each symmetric positive definite matrix an ordered and an unordered graph. First, some graph-theoretic terminology.

For our purposes, a *graph* will be a pair, $G = (X, E)$, where X is a finite set of $|X|$ elements called vertices, and

$$E \subseteq \{\{x, y\} \mid x, y \in X, \qquad x \neq y\}$$

is a set of $|E|$ vertex pairs called *edges*. Given $x \in X$, the set

$$\operatorname{adj}(x) = \{y \in X \mid \{x, y\} \in E\}$$

is the set of vertices *adjacent* to X. For distinct vertices $x, y \in X$ a *chain* from x to y of length $l = n$ is an ordered set of distinct vertices

$$\mu = [p_1, p_2, \ldots, p_{n+1}], \qquad p_1 = x, \quad p_{n+1} = y$$

such that $p_{i+1} \in \operatorname{adj}(p_i)$, $i = 1, \ldots, n$. Similarly, a *cycle* of length $l = n$ is an ordered set of n distinct vertices

$$\mu = [p_1, p_2, \ldots, p_n, p_1]$$

such that $p_{i+1} \in \operatorname{adj}(p_i)$, $i = 1, \ldots, n-1$ and $p_1 \in \operatorname{adj}(p_n)$. We will always assume that the graph G is *connected*, that is, for each pair of distinct vertices $x, y \in X$, there is a chain from x to y.

For a graph $G = (X, E)$ with $|X| = n$ an *ordering* of $|X|$ is a bijection

$$\alpha \colon \{1, 2, \ldots, n\} \leftrightarrow X.$$

We sometimes indicate an ordering by the shorthand $X = \{x_i\}_{i=1}^n$. If $G = (X, E)$ and X is ordered by α, then $G_\alpha = (X, E, \alpha)$ is an ordered graph associated with G. Given an ordering α of X, the set of vertices *monotonely adjacent* to a vertex x is denoted by $\mathrm{M\,adj}(x)$ and defined by

$$\mathrm{M\,adj}(x) = \mathrm{adj}(x) \cap \{z \in X \mid \alpha^{-1}(z) > \alpha^{-1}(x)\}.$$

We associate with each $n \times n$ symmetric matrix $M = (m_{ij})$ an ordered graph $G_\alpha = (X, E, \alpha)$ such that vertex x_i corresponds to row i and $\{x_i, x_j\} \in E$, if and only if $m_{ij} \neq 0$ and $i < j$. The unordered graph $G = (X, E)$ then represents the equivalence class of matrices PMP^{T}, where P is any permutation matrix. For convenience we assume that for no P can PMP^{T} be represented as a direct sum of lower-order matrices, that is, M is irreducible, so that G is connected.

Consider again (7) which represents the first major step of elimination, the elimination of x_1. We proceed to interpret this step graph theoretically. Let $G = (X, E)$ be a graph and α be an ordering of X. The *deficiency*, $D(x)$, is the set of all distinct pairs of $\mathrm{adj}(x)$ which are not themselves adjacent, that is,

$$D(x) = \{\{y, z\} \mid y, z \in \mathrm{adj}(x),\ y \neq z,\ y \notin \mathrm{adj}(z)\}.$$

Similarly, the *monotone deficiency*, $MD(x)$, is the set

$$MD(x) = \{\{y, z\} \mid y, z \in \mathrm{M\,adj}(x),\ y \neq z,\ y \notin \mathrm{adj}(z)\}.$$

Finally, for a graph $G = (X, E)$ and subset $A \subseteq X$, the *section graph* $G(A)$ is the subgraph

$$G(A) = (A, E(A)),$$

where $E(A) = \{\{x, y\} \in E \mid x, y \in A\}$.

Given a vertex y of a graph G, the graph G_y obtained from G by

(1) deleting y and its incident edges;

(2) adding edges such that all vertices in the set $\mathrm{adj}(y)$ are pairwise adjacent

is the *y-elimination graph* of G (compare Parter [17, p. 120]). Thus

$$G_y = (X - \{y\},\ E(X - \{y\}) \cup D(y)).$$

For an ordered graph $G = (X, E, \alpha)$, the *order sequence* of elimination graphs $G_1, \ldots, G_{n-1}$ is defined recursively by $G_1 = G_{x_1}$ and $G_i = (G_{i-1})_{xi}$, $i = 2, \ldots, n-1$.

Since the graphs G_i determine the evolution of the process of vertex elimination, we formally define the *elimination process* on a graph $G = (X, E)$ with ordering α as the ordered set

$$P(G; \alpha) = [G = G_0, G_1, \ldots, G_{n-1}].$$

An elimination process $P(G; \alpha)$ is perfect if

$$G_i = G\left(X - \bigcup_{j=1}^{i} \{x_j\}\right).$$

DEFINITION.† The ordered graph $G = (X, E, \alpha)$ is *monotone transitive* when, for all $x \in X$, we have

$$y \in \mathrm{M\,adj}(x) \quad \text{and} \quad z \in \mathrm{M\,adj}(x) \Rightarrow y \in \mathrm{adj}(z).$$

The significance of monotone transitivity is given in the following lemma which merely summarizes our definitions and relates them to perfect elimination matrices. It is immediate that monotone transitivity is the graph-theoretic interpretation of the perfect elimination matrix condition of (16).

LEMMA 1. Let M be a symmetric positive definite matrix with unordered graph $G = (X, E)$. Then the following are equivalent:

(1) M is a perfect elimination matrix;
(2) there exists an ordering α such that $G_\alpha = (X, E, \alpha)$ is monotone transitive;
(3) in G_α, $MD(x) = \varnothing$ for all $x \in X$;
(4) $P(G; \alpha)$ is a perfect elimination process.

Thus, in a monotone transitive graph, vertex elimination adds no edges. Suppose, however, that $G_\alpha = (X, E, \alpha)$ represents a matrix M which is not a perfect elimination matrix. If elimination is carried out on M, vertex elimination of G_α, then for each $1 = i < j < k$ such that $m_{1j} \neq 0$ and $m_{1k} \neq 0$ but $m_{jk} = 0$, a new nonzero element will be created in the (j, k) position of $M^{(2)} = \overline{M} - rr^{\mathrm{T}}/a$, see (7). Clearly, the graph of $M^{(2)}$ is the elimination graph G_1. Continuing inductively, we see that the study of monotone transitive graphs is interesting even if G_α is not monotone transitive, because the elimination process may be regarded as transforming the graph G_α, matrix M, into its *monotone transitive extension* $\mathrm{M\,TE}(G; \alpha)$, where

$$\mathrm{M\,TE}(G; \alpha) = \left(X, E \bigcup_{i=1}^{n-1} \tau_i\right), \qquad \tau_i = D(x_i) \quad \text{in} \quad G_{i-1}.$$

and $\mathrm{M\,TE}(G; \alpha)$ is the graph of L^{T}.

3. Triangulated Graphs

3.1. Preliminaries

In Section 2 we studied the role of ordered monotone transitive graphs in the elimination process. Here we shall characterize monotone transitive

† By way of motivation, a graph is transitive [8, p. 31] if $y \in \mathrm{adj}(x)$ and $x \in \mathrm{adj}(z)$ implies $y \in \mathrm{adj}(z)$. Since we are dealing with undirected graphs, the adjacency relation is symmetric, that is, $x \in \mathrm{adj}(z) \Leftrightarrow z \in \mathrm{adj}(x)$. It is easy to see that any connected transitive graph on n vertices is the complete graph on n vertices, because then between any two vertices x and y

graphs by their cycle structure and their separating sets of vertices. Monotone transitive graphs are shown to be triangulated graphs as defined by Berge [3]. The theory developed in this section shows very clearly why sparse matrices must fill in during elimination.

Recall that we are dealing only with connected graphs, and for a graph $G = (X, E)$ and subset $A \subseteq X$, the *section graph* $G(A)$ is the subgraph

$$G(A) = (A, E(A)), \qquad E(A) = \{\{x, y\} \in E \mid x, y \in A\}.$$

A *separator* of a graph $G = (X, E)$ is a subset $S \subset X$ such that the section graph $G(X - S)$ consists of two or more *connected components*, say $C_i = (V_i, E_i)$. The section graphs $G(S \cup V_i)$ are then the *leaves* of G with respect to S. A *minimal separator* is a separator no subset of which is also a separator. Similarly, given $a, b \in X$ with $a \notin \operatorname{adj}(b)$, an a, b separator is a separator such that a and b are in distinct components, say C_a and C_b, respectively. Note that (see the Example) a minimal separator is a minimal a, b separator for some $a, b \in X$, but a minimal a, b separator is not, in general, a minimal separator. A *clique* C of a graph is a subset of vertices which are pairwise adjacent. A *separation clique* is a separator which is also a clique.

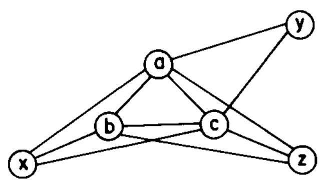


Fig. 1. Graph $G = (X, E)$.

Example: Consider the graph $G = (X, E)$ shown in Fig. 1. The set $S = \{a, b, c\}$ is a separator and $G(X - S)$ consists of the three components (x, ϕ), (y, ϕ), and (z, ϕ). The leaf containing x is the clique on $\{x, a, b, c\}$. Note that S is not minimal because $S' = \{a, c\}$ is also a separator. S' is, however, minimal and it is a minimal x, y separator. On the other hand, S is a minimal x, z separator. In addition, both S and S' are separation cliques.

The following definition is due to Berge [3, p. 158].

Definition. A graph G is *triangulated*, if for every cycle $\mu = [p_1, \ldots, p_n, p_1]$ of length $n > 3$, there is an edge of G joining two nonconsecutive vertices of μ. Such edges are called *chords* of the cycle.

there exists a chain from x to y. Hence, a matrix represented by a transitive graph is a full matrix. In Section 3 we will see how monotone transitive graphs are built up of smaller intersecting complete graphs.

Remark: Note that any section graph of a triangulated graph is triangulated because any cycle in $G = (X - A)$ is a cycle in G itself, and the chord of this cycle in G must be an edge in $G(X - A)$.

3.2. Main Results

THEOREM 2.† For a graph $G = (X, E)$ the following statements are equivalent:

(1) G is triangulated;
(2) every minimal a, b separator is a clique;
(3) there exists an ordering α of X such that $G_\alpha = (X, E, \alpha)$ is monotone transitive.

Theorem 2 and Lemma 1 (Section 2) characterize monotone transitive graphs and thus perfect elimination matrices. Statements (1) and (2) give the structure of the unordered graph while statement (3) is a property of a corresponding ordered graph. Of the three equivalent properties above, statement (3) is clearly the most algorithmic in the sense that its verification is straightforward. In fact Lemma 1, Section 2 shows how to test for monotone transitivity, because in each successive elimination graph there must always exist a vertex with empty deficiency.

THEOREM 3. Let $G = (X, F)$ be triangulated with subgraph $\hat{G} = (X, E)$, $E \subseteq F$. Then $\hat{G}$ is triangulated, if and only if for each $e = \{x, y\} \in F - E$ there exists an x, y separation clique, S_e, of $\hat{G}$.

If $G = (X, F)$ is triangulated, an arbitrary subgraph of G obtained by removing a subset of edges need not remain triangulated. Theorem 3 gives a necessary and sufficient condition that the subgraph be triangulated. It has an important corollary which requires anticipating a notion of Section 4.

Suppose a graph $G = (X, E)$ is not triangulated. Then for any ordering α of X the set $T(\alpha)$ of M TE$(G; \alpha) = (X, E \cup T(\alpha))$ is a *triangulation* of G generated by the ordering α. A *minimum triangulation* would be a triangulation, $T(\hat{\alpha})$, such that

$$|T(\hat{\alpha})| = \min_{\alpha} |T(\alpha)|$$

COROLLARY 1. Let $G = (X, E)$ be a graph with separation clique S with components C_i and leaves L_i. Then any minimum triangulation T of G contains only edges $e = \{x, y\} \in T$ with x and y in the same component C_j, or edges $e = \{x, y\} \in T$ with $x \in C_j$ and $y \in S$.

† For statements of parts of this theorem see Boland and Lekkerkerker [5a], Dirac [9a], and Fulkerson and Gross [12a].

Proof: If the triangulation T contains a nonempty subset of edges with incident edges in C_j and $C_k, j \neq k$, these edges may be deleted, and by Theorem 3 the resulting set $\hat{T}$ is still a triangulation.

Thus, in a graph with a separation clique, the problem of finding a minimum triangulation reduces to finding a minimum triangulation for each leaf.

THEOREM 4. Let $G = (X, E)$ be triangulated, and α be a monotone transitive ordering. If S is a minimal a, b separation clique of G, then $S = \mathrm{M\,adj}(x_j)$ for some $x_j \in X$. Conversely, for any $x_i \in X$, such that the vertices of the elimination graph G_{i-1} are not a clique, $\mathrm{M\,adj}(x_i)$ is a separation clique of G.

Hence, in a triangulated graph, all minimal a, b separators are generated in the elimination process. Note that although the sets $\mathrm{M\,adj}(x_i)$ are separation cliques, if G_{i-1} is not a clique, they need not be minimal a, b separation cliques. For example, in G of Fig. 2 below, {③, ④} is the only minimal a, b separator.

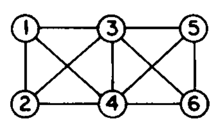


Fig. 2

Remark: One interesting application of Theorem 4 is that in a non-triangulated graph $G = (X, E)$ with $|X| = n$, at most, $n - \mu$ minimal a, b separators of G remain minimal a, b separators of $\hat{G} = (X, E \cup T)$, where T is any triangulation of G and $\mu = \max\{|C|, C \text{ a clique of } G\}$. This follows because any clique in G is also a clique in $\hat{G}$, and because, as we shall see in the proof of Theorem 2 in the next section, a monotone transitive ordering for $\hat{G}$ can be found which orders vertices in any clique last.

3.3. *Proofs and Corollaries*

We begin the proof of Theorem 2 by generalizing slightly Theorem 3 of Berge [3, p. 160].

LEMMA 2. In a triangulated graph $G = (X, E)$ every minimal a, b separator is a clique.

Proof: Let S be a minimal a, b separator and C_a and C_b be the components of $G(X - S)$ containing a and b, respectively. Since S is minimal, each $s \in S$

is adjacent to some vertex in C_a and some vertex in C_b. Let $x,y \in S$, and let μ_i be the shortest chains of the type

$$[x, c_{i,1}, c_{i,2}, \ldots, c_{i,p_i}, y], \qquad i = 1,2, \quad c_{1,j} \in C_a, \quad c_{2,j} \in C_b.$$

The cycle containing x and y formed by μ_1 and μ_2 has length $l \geqslant 4$, and the only possible chord is $\{x,y\}$.

LEMMA 3. Let $G=(X,E)$ be a graph with separation clique S and leaves L_i, $i=1,\ldots,n$. If S_0 is a separator of some L_i, then S_0 is a separator of G. Furthermore, if S_0 is a minimal a,b separator of L_i, then S_0 is a minimal a,b separator of G.

Proof: Let D_j, $j=1,\ldots,m$ be the components of L_i with respect to S_0. Since S is a clique, vertices in S can be in only one component, say D_k. Thus, S_0 is a separator of G, because any chain from a vertex $x \in (L_j - S_0)$ with $j \neq i$ to a vertex $y \in D_l$, $l \neq k$ must contain a vertex of S_0. This proves the first statement.

For the second statement, note that S_0 is a separator of G as we have just shown. It must be an a,b separator, for the same a,b, because $D_i \cap S \neq \phi$ for at most one i. Finally, S_0 must be minimal in G since any a,b separator $\hat{S}_0 \subset S_0$ in G must be an a,b separator in L_i.

LEMMA 4. Let $G=(X,E)$ satisfy statement (2) of Theorem 2. Then either X is a clique, or given any clique $C \subset X$, there exists a vertex $x \notin C$ such that $D(X) = \varnothing$.

Proof: The proof is by induction on $|X|$ and the case $|X|=1$ is clear. Assuming any case with $|X| \leqslant k$, let $G=(X,E)$ be such a graph with $|X| = k+1$ and C be any clique. Either X is a clique or there exists by Lemma 2 some a,b separation clique of G, say C_1. Let D_a, D_b and L_a, L_b be the corresponding components and leaves of G containing a and b respectively. Clearly, the vertices in $C - C_1$ can be in, at most, one component. Suppose such vertices are in D_a. Consider the leaf L_b. By Lemma 3 it inherits statement (2) of Theorem 2. Writing $L_b = (W,F)$, we have $|W| \leqslant k$ and, hence, by induction, either W is a clique or there exists a vertex $x \notin C_1$ such that $D(x) = \varnothing$ in L_b. In either case then, since W must contain at least one vertex not in C_1, there exists an $x \notin C_1$ with $D(x) = \varnothing$ in L_b. Finally, $D(x) = \varnothing$ in G because x is not adjacent to a vertex in any component other than D_b. Clearly, $x \notin C$, the original clique.

Lemmas 3 and 4 yield the following two corollaries concerning the existence of vertices with $D(x) = \varnothing$. The first corollary will imply that the ordering α guaranteed by statement (3) of Theorem 2 is not unique.

COROLLARY 2. Let G be as in Lemma 4, and S be any separation clique of G with components C_i and leaves L_i. Then for each component C_i, there exists a vertex $c_i \in C_i$ with $D(c_i) = \varnothing$ in G.

Proof: By Lemma 3, each L_i has Property (2) of Theorem 2. Thus by Lemma 4, for each L_i there exists a vertex $c_i \notin S$ of L_i with $D(c_i) = \varnothing$ in L_i and, therefore, in G.

COROLLARY 3. Let G be as in Lemma 4. Then, for any $x \in X$, one, and only one, of the following statements is true:

(1) $D(x) = \varnothing$;
(2) $x \in S$, where S is a minimal a, b separation clique.

Proof: If (2) is true, clearly (1) must be false. We show by induction on $|X|$ that (1) or (2) must be true. The case $|X| = 1$ is clear, and we suppose the case $|X| \leqslant k$. Note that if X is a clique, the result is immediate. Assuming otherwise, let S be a minimal a, b separation clique of G. Let $x \in X$. If $x \in S$, the proof ends, so let $x \in (L_a - S)$. By the induction hypothesis and Lemma 3, either $D(x) = \varnothing$ in L_a, and hence in G, or $x \in \hat{S}$, where $\hat{S}$ is a minimal c, d separation clique of L_a, and hence of G.

LEMMA 5. Let $G = (X, E)$ be as in Lemma 4. Then there exists an ordering α of X such that for all $x \in X$, $\mathrm{M}\,D(x) = \varnothing$.

Proof: The proof is by induction on $|X|$. The case $|X| = 1$ is clear, and we suppose the case $|X| = k$. If G is such a graph with $k+1$ vertices, then, by Corollary 2 above, there exists a vertex x_1 such that $D(x_1) = \varnothing$. Let $G_1 = (X_1, E_1)$ be the x_1-elimination graph. Since $\mathrm{adj}(x_1)$ is a separation clique, if X itself is not a clique, G_1 satisfies the hypothesis of the lemma by Lemma 3, and G_1 has $|X_1| = k$. By induction there exists an ordering α_1 of the vertices of G_1 such that

$$\alpha_1(i) = x_{i+1}, \qquad i = 1, \ldots, k, \qquad \text{defining } x_i$$

with $\mathrm{M}\,D(x_i) = \varnothing$. Finally, in G, choose the ordering

$$\alpha(i) = x_i, \quad i = 1, \ldots, k+1.$$

Then $\mathrm{M}\,D(x_i) = \varnothing$ with this ordering in G.

Note that the ordering α assured by Lemma 5 is not unique in view of Corollary 2. This means that if $G = (X, E)$ is not triangulated, any triangulation $T(\alpha)$ generated by an ordering α will also be generated by other orderings α'. Also, note that another way of stating Lemma 5 is that there exists an ordering α such that the order sequence of elimination graphs of G, that is, $G = G_0, G_1, \ldots, G_{n-1}$, has $D(x_i) = \varnothing$ in G_{i-1}. Finally, we shall call any ordering guaranteed by Lemma 5 a *monotone transitive ordering.*

LEMMA 6. A monotone transitive graph is a triangulated graph.

Proof: Let α be the ordering and μ be any cycle with $l > 3$. Let $p^* \in \mu$ be the vertex such that

$$\alpha^{-1}(p^*) = \min_{p \in \mu} \alpha^{-1}(p).$$

Since p^* is adjacent to two nonconsecutive vertices by monotone transitivity, μ has a chord.

Proof of Theorem 2: Statement (1) $\Rightarrow$ statement (2) by Lemma 2, statement (2) $\Rightarrow$ statement (3) by Lemma 5, and statement (3) $\Rightarrow$ statement (1) by Lemma 6 and Lemma 1.

The following corollary shows that in a triangulated graph a monotone transitive ordering can be found such that any given clique is ordered last.

COROLLARY 4. Let $G = (X, E)$ be triangulated with clique $C \subseteq X$. Then there exists a monotone transitive ordering α such that $\alpha(j) \in C$ for $j = k+1, k+2, \ldots, |X|$, where $k = |X| - |C|$.

Proof: The proof follows from Lemma 4 and the induction argument of Lemma 5.

Corollary 4 has the following interesting interpretation. Suppose $G = (X, E)$ is not triangulated, and we wish to find an ordering which generates a triangulation $T(\alpha)$ with a specific property, for example, a minimum triangulation. Since any clique in G remains a clique in the triangulated graph $\hat{G} = (X, E \cup T(\alpha))$, the corollary implies the existence of other orderings α' such that $T(\alpha) = T(\alpha')$, and such that α' orders the clique last. We will see in Section 4 that if only the unknowns represented by the vertices in the clique are desired, ordering the clique last will reduce the number of backsolving operations [see (12) and (14)].

We begin the proof of Theorem 3 with

LEMMA 7. Let $G = (X, F)$ be triangulated with a subgraph $\hat{G} = (X, E)$, $E \subset F$. Suppose S is a separation clique of $\hat{G}$ such that for each edge $e = \{x, y\} \in F - E$, x and y are in different components. Then $\hat{G}$ is triangulated.

Proof: Let μ be any cycle in $\hat{G}$ with $l \geqslant 4$. If μ is entirely with some leaf of $\hat{G}$, then μ contains a chord, because μ is also a cycle in G. If μ has vertices in more than one component, then μ must contain at least two distinct vertices of S. These vertices are adjacent; hence, μ has a chord.

Proof of Theorem 3: The "if" part of the theorem follows by successive applications of Lemma 7. Given some S_e, discard all edges in $F-E$ with incident vertices in different components. S_e is then a separation clique of this new graph $\bar{G}$. By Lemma 7, $\bar{G}$ is triangulated. Continue for each edge in $F-E$ not already discarded. The converse is clear by Lemma 2, because for each $e=\{a,b\}\in F-E$, there exists a minimal a,b separator S_e in $\hat{G}$, and S_e is a clique.

Proof of Theorem 4: To prove the first assertion, let $C_1=(V_1,E_1)$ and $C_2=(V_2,E_2)$ be the components of G with respect to S containing a and b respectively. For each V_i let $v_i{}^*$ be the vertex such that

$$\alpha^{-1}(v^*) = \max_{v\in V_i}\alpha^{-1}(v).$$

Choose $\hat{v}\in\{v_1{}^*,v_2{}^*\}$ such that

$$\alpha^{-1}(\hat{v}) = \min(\alpha^{-1}(v_1{}^*),\alpha^{-1}(v_2{}^*)).$$

Because S is minimal, each $s\in S$ is adjacent to some vertex in C_a and C_b. Hence, if $j=\alpha^{-1}(\hat{v})$ by monotone transitivity and the connectivity of C_a and C_b, we have $S=\mathrm{M\,adj}(x_j)$.

To prove the second assertion, note first that $\mathrm{M\,adj}(x_1)$ is a separation clique of G, unless X is a clique. Also, the elimination graph G_1 is a leaf of G, which is triangulated, with respect to $\mathrm{M\,adj}(x_1)$, and $G_1=(X_1,E_1)$ has $|X_1|=|X|-1$. The assertion then follows by induction on $|X|$ and Lemma 3.

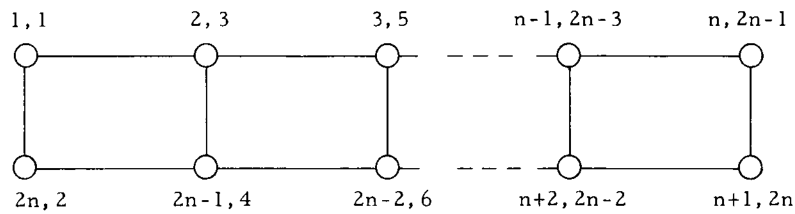
3.4. Examples

As our first example, we will discuss in detail the ladder graph (see Fig. 3), since it illustrates the notions of Section 2 and Corollary 1, as well as anticipating some of the developments in the next section. The remaining examples are classes of graphs which illustrate our theoretical results.

3.4.1. Ladder Graph

Figure 3a shows the ladder graph on $2n$ vertices with two ordering α_1 and α_2, written in the form $(\alpha_1(x),\alpha_2(x))$ at each vertex x. Figure 3b shows the zero-nonzero structure of the matrices corresponding to the two ordered graphs. Finally, Fig. 3c shows the upper triangular factor L^T [see (5)] for each matrix. Clearly M_1 and M_2 are not perfect elimination matrices, because the graph has nonchorded cycles.

Note that the decomposition using α_1 requires $O(n^2)$ cells of storage, while the decomposition using α_2 requires only $O(n)$ cells. We will see in Section 4,



(a)

$$M_1 = \begin{bmatrix} * & * & & & & & & * \\ * & * & * & & & & * & \\ & * & * & * & & * & & \\ & & * & * & * & & & \\ & & & * & * & * & & \\ & & * & & * & * & * & \\ & * & & & & * & * & * \\ * & & & & & & * & * \end{bmatrix} ; \quad M_2 = \begin{bmatrix} * & * & * & & & & & \\ * & * & 0 & * & & & & \\ * & 0 & * & * & * & & & \\ & * & * & * & 0 & * & & \\ & & * & 0 & * & * & * & \\ & & & * & * & * & 0 & * \\ & & & & * & 0 & * & * \\ & & & & & * & * & * \end{bmatrix}$$

(b)

$$L_1^T = \begin{bmatrix} * & * & 0 & 0 & 0 & 0 & 0 & * \\ & * & * & 0 & 0 & 0 & * & * \\ & & * & * & 0 & * & * & * \\ & & & * & * & * & * & * \\ & & & & * & * & * & * \\ & & & & & * & * & * \\ & & & & & & * & * \\ & & & & & & & * \end{bmatrix} ; \quad L_2^T = \begin{bmatrix} * & * & * & & & & & & \\ & * & * & * & & & & & \\ & & * & * & * & & & & \\ & & & * & * & * & & & \\ & & & & * & * & * & & \\ & & & & & * & * & * & \\ & & & & & & * & * & * \\ & & & & & & & * & \end{bmatrix}$$

(c)

Fig. 3. (a) Ladder graph with two orderings. (b) Matrices corresponding to the two orderings, where * indicates nonzero elements. (c) Upper triangular factors.

Theorem 5 that $O(n^3)$ arithmetic operations are required to effect the decomposition with α_1, while only $O(n)$ operations are needed with α_2. The difference is significant.

It follows from Corollary 1 that α_2 generates a minimum triangulation of the ladder graph. Since the pairs of vertices connected by each of the $n-2$ inner vertical rungs form separation cliques, the problem of finding a minimum triangulation for the ladder reduces to finding a minimum triangulation of the cycle on four vertices which requires only one edge. Thus, a minimum triangulation for the $2n$ vertex ladder requires $n-1$ edges, and one such triangulation is generated by α_2.

3.4.2. Trees and k Trees

A *tree* is a connected graph which has no cycles. Equivalently a tree with $|X|=n>1$ vertices is a connected graph with $n-1$ edges. Apparently Parter [17] was the first to realize that the matrix M represented by a tree was a perfect elimination matrix, although he does not use this term. Parter gives a specialized algorithm [17] for Gaussian elimination on such a matrix.

Trees are clearly triangulated graphs, and any tree must have at least two *pendent* vertices, that is, vertices adjacent to only one edge. Pendent vertices x are the only vertices in a tree with $D(x)=\varnothing$, otherwise the tree would have a cycle. A generalization of a tree is a k tree defined recursively as follows: A k tree on k vertices is a clique on k vertices. Given any k tree $T_k(n)$ on n vertices, a k tree on $n+1$ vertices is obtained when the $(n+1)$st vertex is adjacent to the vertices of a clique on k vertices in $T_k(n)$.

If we order the vertices x_i, $i=1,2,\ldots,n$ in the construction of a k tree on n vertices as defined above, then clearly this graph is monotone transitive with ordering $\alpha(i)=x_{n+1-i}$, $i=1,\ldots,n$. Then, k trees are triangulated graphs. They also have the following property.

Proposition 1. Every minimal separator S of a k tree $T_k(n)$ has $|S|=k$.

Proof: Since $T_k(n)$ has a monotone transitive ordering such as α above, $|\mathrm{M\,adj}\,(\alpha(i))|=k$ for $i=1,2,\ldots,n-k$. By Theorem 4, $S=\mathrm{M\,adj}\,(x_i)$ for some such i, since neither the set $W=\{x_i\}_{i=1}^{k-1}$ nor any subset of W is a separator.

3.4.3. The Cycle

Let $C=(X,E)$ be the cycle on $|X|=n$ vertices. With respect to triangulating C, we have

Proposition 2. Let $C=(X,E)$ be a cycle with $|X|\geqslant 3$ vertices. Then a minimum triangulation $\hat{T}$ of C has $|\hat{T}|=|X|-3$. Furthermore, if α is any ordering and $\mathrm{M\,TE}(C;\alpha)=(X,E\cup T(\alpha))$, then $T(\alpha)$ is a minimum triangulation.

Proof: Both conclusions are proved easily by induction on $|X|$, and the case $|X|=3$ is immediate. Let $C=(X,E)$ with $|X|=k+1$ assuming these assertions for such graphs with $|X|\leqslant k$. Let $e\in\hat{T}$, where $\hat{T}$ is a minimum triangulation of G. Clearly the vertices incident on e form a separation clique S in $\hat{C}=(X,E\cup\hat{T})$, by Theorem 3. Hence, by the corollary $\hat{T}=T_1\cup T_2\cup\{e\}$, where T_1 and T_2 are minimum triangulations of the leaves of $\hat{C}$ with respect to S, say $L_1=(V_1,E_1)$ and $L_2=(V_2,E_2)$. L_1 and L_2 are cycles with $|V_i|\leqslant k$,

$i=1,2$, and $|V_1|+|V_2|=|X|+2$. By induction, $|T_i|=|V_i|-3$, $i=1,2$, implying $|T|=|X|-3$. For the second statement, note that $|D(x)|=1$ for any $x\in X$, and that the elimination graph $C_x=(X_1,E_1)$ is a cycle with $|X|=k$ vertices. By induction, any ordering $\hat{\alpha}$ on X_1 gives a minimum triangulation of C_x. The assertion now follows.

3.4.4. Complete Bipartite Graphs

A graph $G=(X,E)$ is bipartite if $X=R\cup B$ with $R\cap B=\emptyset$, and for each $e=\{x,y\}\in E$ either $x\in R$, $y\in B$ or $y\in R$, $x\in B$. Equivalently, G is bipartite if every cycle has even length [8, p. 86]. Because of the second condition, trees are the only bipartite graphs which are triangulated.

Let $G=(X,E)$ be a bipartite graph with $X=B\cup R$ and $|R|\leqslant|B|$. If each vertex $x\in R(x\in B)$ is adjacent to each vertex $y\in B(y\in R)$, the resulting graph is a *complete bipartite graph*, denoted by $C_{n,m}$ ($n=|R|$, $m=|B|$).

By Theorem 2, in any triangulation of $C_{n,m}$, there must exist a vertex with $D(x)=\emptyset$. Hence, to triangulate $C_{n,m}$ at least $n(n-1)/2$ edges are necessary. However, this number of edges is clearly sufficient by taking the MTE generated by the ordering $\alpha(i)=b_i$, $i=1,\ldots,m$, $B=\{b_i\}_{i=1}^{m}$ and $\alpha(i)=r_i$, $i=n+1,\ldots,n+m$, $R=\{r_i\}_{i=m+1}^{n+m}$.

4. Optimal Ordering and Algorithms

In this section we examine carefully several criteria by which we may evaluate "optimal," and we relate these criteria to the computational complexity of the elimination process on sparse matrices. We give, first, a count of the number of operations needed to effect the decompositions and backsolving operations associated with solving symmetric sparse linear systems $Mx=b$. In Section 4.2 we discuss criterion functions in a general setting, and in Section 4.3 we present some results which give bounds for triangulations T of a nontriangulated graph. Finally in Section 4.4 we discuss ordering algorithms.

4.1. Operation Counts and Practical Criteria

Let M be an $n\times n$ symmetric positive definite matrix with ordered graph $G=(X,E,\alpha)$. Denote by $d(\alpha(i))$ the degree of the vertex $\alpha(i)$ in the elimination graph G_{i-1}, that is $d(\alpha(i))=|\text{adj}(\alpha(i))|$ in G_{i-1}. Where it causes no confusion, $d(\alpha(i))$ will be written d_i. Using this notation we present the following:

THEOREM 5. Let M and G be as above. Counting multiplications and divisions as multiplications and operations, $a+0$, $a \neq 0$, which occur whenever $D(x_i) \neq \varnothing$ in G_{i-1} as additions, we have

(a) the LDL^T decomposition [see (8)] requires

$$\sum_{i=1}^{n-1} d_i(d_i+3)/2 \qquad \text{multiplications} \tag{17}$$

and

$$\sum_{i=1}^{n-1} d_i(d_i+1)/2 \qquad \text{additions;} \tag{18}$$

(b) the Cholesky decomposition $M = GG^T$ [see (9)] requires the same number of multiplications and additions as in (a) and also n square roots;

(c) for a general n-vector b, the back-solving operations

(1) $Lz = b$,
(2) $Dy = z$,
(3) $L^T x = y$

require

$$2\sum_{i=1}^{n-1} d_i + n \qquad \text{multiplications} \tag{19}$$

and

$$2\sum_{i=1}^{n-1} d_i \qquad \text{additions.} \tag{20}$$

(d) the back-solving operations $Gy = b$ and $G^T x = y$ require n more multiplications than (19) and the same number of additions as (20).

Proof: By the discussion in Section 2.1 [see (7)–(9)] we see that (b) follows easily from (a). The proof of (a) is by induction on n. The case $n = 2$ is immediate. Suppose the theorem is true for $2 < n = k-1$ and let $G = (X, E, \alpha)$ have $|X| = k$. Referring to (7) and (8), the first step of elimination requires that we compute $s = r/a$ and $M - sr^T$ for all $1 = i \leqslant j \leqslant n$. This requires d_1 multiplications and $d_1(d_1+1)/2$ multiplications and additions. Hence, in total the first step of elimination requires

$$d_1(d_1+3)/2 \quad \text{multiplications} \quad \text{and} \quad d_1(d_1+1)/2 \quad \text{additions.} \tag{21}$$

Since the graph of $M^{(2)}$ is the elimination graph G_1, we have, by induction that the decomposition of $M^{(2)}$ requires

$$\sum_{i=2}^{n-1} d_i(d_i+3)/2 \quad \text{multiplications} \quad \text{and} \quad \sum_{i=2}^{n-1} d_i(d_i+1)/2 \quad \text{additions.} \tag{22}$$

Adding (21) and (22) gives (17) and (18).

To verify (c), recall that the graph of L^T is $\mathrm{MTE}(G;\alpha) = (X, E \cup \tau_i)$ and note that one addition and one multiplication are required for each edge in $E \cup \tau_i$ in the operations (1) and (3). Since

$$\sum_{i=1}^{n-1} d_i = |E \cup \tau_i|,$$

the result follows.

The Cholesky backsolving operations (d) require n more multiplications than the total in (c) because G has, in general, a nonunit diagonal.

These counts show that for a sparse $n \times n$ matrix M as above, the importance of n as a measure of computational complexity is relatively minor. For example, for such an arbitrary irreducible matrix we know, *a priori*, only that the number of multiplicative operations θ for the decomposition A satisfies

$$2(n-1) \leqslant \theta \leqslant \tfrac{1}{6}n(n-1)(n+4).$$

We consider three practical criteria for optimal ordering of a symmetric matrix M for elimination. While the minimum arithmetic criterion is suggested naturally by the operation counts given above, the minimum "fill in" and minimum bandwidth criteria are the two most commonly used.

4.1.1. Minimum Arithmetic

Let M be a symmetric matrix with ordered graph $G = (X, E, \alpha)$, $|X| = n$. Define

$$L(\alpha) = \sum_{i=1}^{n-1} d(\alpha(i)),$$

$$Q(\alpha) = \sum_{i=1}^{n-1} d^2(\alpha(i)),$$

and

$$J(p,q;\alpha) = pL(\alpha) + qQ(\alpha), \qquad p > 0, \quad q > 0.$$

Then, criteria based on minimizing arithmetic operations counted by Theorem 5 can be formulated as attempting to find $\hat{\alpha}$ such that

$$J(p,q;\hat{\alpha}) = \min_{a} J(p,q;\alpha)$$

for specific p and q. For example, to solve $Mx = b$ using the LDL^T decomposition and backsolving requires

$$J(\tfrac{1}{2}, \tfrac{7}{2}; \alpha) + n \quad \text{multiplications} \qquad \text{and} \qquad J(\tfrac{1}{2}, \tfrac{5}{2}; \alpha) \quad \text{additions.}$$

To compute $\det(M)$, which is the product of the diagonal entries of (D) requires

$$J(\tfrac{1}{2},\tfrac{3}{2};\alpha)+(n-1) \quad \text{multiplications} \qquad \text{and} \qquad J(\tfrac{1}{2},\tfrac{1}{2};\alpha) \quad \text{additions.}$$

The operation counts for these two specific computations above suggest a difficulty with the minimum arithmetic criterion. To define "optimal" for either of these computations requires a decision about the relative cost of additions, multiplications, and storage. Furthermore, an optimal ordering for solving $Mx=b$ is not necessarily an optimal ordering for the problem of computing $\det(M)$. Since both computations involve the decomposition $M=LDL^{\mathrm{T}}$, it may be unsatisfactory, from the viewpoint of having a general sparse matrix package, to consider four different criteria in order to define optimal ordering for these two very similar computations. Specifically, the difficulty arises because *a priori* we cannot be assured that there exists an ordering α which minimizes $L(\alpha)$ and $Q(\alpha)$ simultaneously. In practice, it is common to attempt to minimize the less stringent fill in criterion $L(\alpha)$. To relate the $L(\alpha)$ criterion to the more general minimum arithmetic criterion, the following bound is relevant.

PROPOSITION 3. Let $G=(X,E,\alpha)$ be monotone transitive and

$$\mu(\alpha) = \max_{1\leqslant i\leqslant n-1} d(\alpha(i)).^{\dagger}$$

Then

$$Q(\alpha) \leqslant \mu L(\alpha) - (\mu-1)\mu(\mu+1)/6,$$

and there exist graphs for which this bound is sharp.

Proof: The elimination graph G_i must contain a clique with μ vertices if $d(\alpha(i))=\mu$ is the degree of the vertex $\alpha(i)$ in G_{i-1}, since G is monotone transitive. Hence, for all integers $l=\mu-1, \mu-2,\ldots,1$ there exists an integer $k_l>i$ with $d(\alpha(k_l))=l$.

Let

$$p_i = \{j\in I \mid d(\alpha(j))=i\}, \qquad I=\{1,2,\ldots,n-1\}.$$

Then

$$\begin{aligned} Q(\alpha) &= \sum_{i=1}^{n-1} d^2(\alpha(i)) = \sum_{i=1}^{\mu} i^2 + \sum_{i=1}^{\mu} (|p_i|-1)i^2 \\ &\leqslant \sum_{i=1}^{\mu} i^2 + \mu\sum_{i=1}^{\mu}(|p_i|-1)i \\ &= \mu\sum_{i=1}^{n-1} d_i - \sum_{i=1}^{\mu}(\mu-i)i = \mu\sum_{i=1}^{n-1} d_i - (\mu-1)\mu(\mu+1)/6. \end{aligned}$$

Finally, the monotone transitive graph of Fig. 4 shows that equality is possible.

† Note that $\mu+1$ is the number of vertices in the largest clique of G.

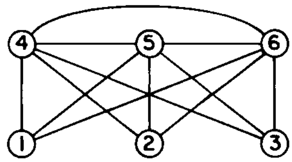


Fig. 4. Monotone transitive graph, where $d(1) = d(2) = d(3) = 3$, $d(4) = 2$, $d(5) = 1$.

4.1.2. MINIMUM FILL-IN

Let $G = (X, E)$ be a graph with monotone transitive extention $\hat{G} = (X, E \cup T(\alpha))$. Then, since $\hat{G}$ is monotone transitive,

$$L(\alpha) = \sum_{i=1}^{n-1} d(\alpha(i)) = |E| + |T(\alpha)| \tag{23}$$

because each edge in $E \cup T(\alpha)$ is counted once, and only once, in some $d(\alpha(i))$. Thus, by minimizing $L(\alpha)$ over all orderings, we minimize the fill in $T(\alpha)$ caused by elimination. Then, $T(\alpha)$ is a minimum triangulation of the graph.

Various authors [20, 23, 25, 26, 29, p. 25] have taken the criterion of minimum fill in as the "appropriate" criterion for defining optimal orderings. However, the effect of minimizing $L(\alpha)$ upon the count of necessary arithmetic operations for certain computations seems to have been overlooked in the literature. It is certainly not the case, as is evident from the discussion above, that minimizing $L(\alpha)$ necessarily minimizes arithmetic. Note, however, that for any α we must store $L(\alpha)+|X|$ nonzero numbers for D and L in the decomposition of M corresponding to G. We call this *primary storage*, as opposed to the secondary storage necessary to determine which elements of L are nonzero.

We think the advantages of using $L(\alpha)$ as a criterion for optimal ordering are as follows:

(1) minimizing $L(\alpha)$ minimizes primary storage;
(2) minimizing $L(\alpha)$ minimizes the backsolving operations of Theorem 5(c);
(3) for a graph in which $\mu(\alpha)$ of Proposition 3 can be bounded independent of $|X| = n$, a satisfactory bound on arithmetic operations can be given which is minimized with $L(\alpha)$;
(4) if $L(\alpha)$ is minimum, the triangulation $T(\alpha)$ of G is a minimum triangulation, that is, the function $L(x)$ on a graph has graph-theoretic significance.

4.1.3. MINIMUM BANDWIDTH

For an $n \times n$ symmetric matrix M, it is natural to define M to have *bandwidth* $k \geqslant 0$ if

$$k = \max_{\{i,j\} \in A} (j-i), \tag{24}$$

where

$$A = \{\{i,j\} \mid i < j \text{ and } m_{ij} \neq 0\}.$$

Thus, a symmetric tridiagonal matrix has bandwidth $k = 1$. Equation (24) is consistent with the recent paper of Cuthill and McKee [9], although other authors count the diagonal and subdiagonals in their definition of bandwidth [12, p. 15]. If $G = (X, E, \alpha)$ is the ordered graph of M, clearly M has bandwidth k, if and only if

$$k = \max_{1 \leqslant i \leqslant n-1} \max_{y \in \mathrm{M\,adj}(x_i)} (\alpha^{-1}(y) - i).$$

Furthermore, since $|\mathrm{adj}(x)| = |\mathrm{M\,adj}(x)| + |\{y \in X \mid x \in \mathrm{M\,adj}(y)\}|$, it follows easily that

$$k \geqslant \max_{x \in X} \{[\tfrac{1}{2} |\mathrm{adj}(x)|]\}, \tag{25}$$

where $[p]$ is the least integer $l \geqslant p$.

Matrix bandwidth minimization has enjoyed considerable popularity in matrix methods of structural analysis, see, for example, Livesley [16], McCormick [29, p. 155], Cuthill and McKee [9], and Rosen [19]. By using bandwidth methods, these authors attempt to limit fill in and arithmetic to a level acceptable for their applications. The popularity of bandwidth methods is partially justified by the following two properties of bandwidth analysis. First, for a symmetric matrix M of bandwidth k and ordered graph $G = (X, E, \alpha)$, all the fill in of M due to elimination is constrained within the bandwidth. That is, the graph $\mathrm{M\,TE}(G; \alpha)$ also has bandwidth k.† Second, the special elimination scheme for a symmetric matrix of bandwidth k is relatively easy to implement on a digital computer primarily because necessary data handling and indexing is simplified (see the discussions in McCormick [29, p. 155] and Cuthill and McKee [9, Section 1]).

Note, however, that the effectiveness of bandwidth implicitly presupposes that the band width k of a symmetric $n \times n$ matrix M will be small relative to n. Bandwidth analysis is crude, in general, because k need not be small relative to $|X| = n$, and because this analysis takes no account of the zero–nonzero structure within the band. To substantiate this claim, we appeal to Theorem 4

† This is clear since $k \geqslant |\mathrm{M\,adj}(x)|$, and making $\mathrm{M\,adj}(x)$ a clique does not increase k.

and the corollary to Theorem 3. Theorem 4 states that the sets $\mathrm{M\,adj}(x_i)$ are separation cliques in the extension graph $\mathrm{MTE}(G;\alpha)$. Suppose that the vertices are being ordered by some sequential scheme to attain a minimum, or approximate minimum, bandwidth.† If, in the elimination graph G_i, the set $S = \mathrm{M\,adj}(x_i)$ is a separation clique breaking G into $c \geqslant 2$ components, Theorem 3 and Corollary 1 (Section 3) implies that in G_i the vertices in S should be ordered after those in all but one component. Bandwidth minimization, however, will tend to order the vertices in S immediately. This causes redundant edges in successive elimination graphs which may increase the bound (25) in subsequent elimination graphs.‡ We illustrate this phenomenon in the following example.

Example: The snowflake graph shown in Fig. 5 provides an example where bandwidth ordering (Cuthill–McKee algorithm) orders separating sets too early. Note also that for this graph, (25) gives the overly optimistic bound $k \geqslant 3$. In fact, this ordering gives $k = 6$, where k is not small relative to $|X| = 18$.

4.2. *Criterion Functions*

Let $G = (X, E, \alpha)$ be a monotone transitive graph with $|X| = n$. Then

$$\sum_{i=1}^{n-1} d(\alpha(i)) = |E|,$$

that is, the $(n-1)$ integers $d(\alpha(i))$ form a *partition*, or degree partition, of $|E|$.

For two ordered monotone transitive graphs $G_\alpha = (X, E, \alpha)$ and $G_\beta = (X, F, \beta)$ with $|X| = n$, the partitions of $|E|$ and $|F|$ generated by the $d(\alpha(i))$ and the $d(\beta(i))$, respectively, will be called *equal*, if there exists a permutation π on the integers $1, 2, \ldots, n-1$ such that

$$d(\alpha(i)) = d(\beta(\pi(i))), \qquad i = 1, 2, \ldots, n-1.$$

Similarly, the partition generated by the $d(\alpha(i))$ *dominates* the partition generated by the $d(\beta(i))$, if

$$d(\alpha(i)) \geqslant d(\beta(\pi(i))), \qquad i = 1, 2, \ldots, n-1.$$

† As, for example, the algorithm presented by Cuthill and McKee [9], which is probably the best available for large order graphs ($|X| = 10^3$–10^5). It can be combined with the recent algorithm of Rosen [19] for further improvements (see Cuthill and McKee [9, p. 12]). See Akyuz and Utku [1] and Alway and Martin [2] for other bandwidth algorithms.

‡ This analysis explains the results of Cuthill and McKee [9, p. 15] where it is reported that, for several sets of graphs generated randomly with $|X| = 50$ and $100 \leqslant |E| \leqslant 150$, the average bandwidth, after using the Cuthill–McKee algorithm or the Cuthill–McKee–Rosen modification, ranged from $k = 17$ to $k = 28.2$. k is not found to be small relative to $|X|$ in these experiments.

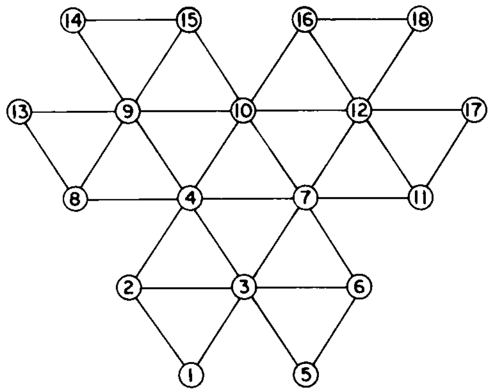


Fig. 5. Snowflake graph with bandwidth ordering given by the Cuthill–McKee algorithm [9], where $k=6$. Vertices ③ and ④ are ordered too early.

We consider a class of functions defined on the quantities $d(\alpha(i))$ each of which may represent a cost of elimination, or if the graph $G=(X,E)$ is not triangulated, these functions can be considered as *criterion functions* for choosing an optimal ordering.

As criterion functions for the graph $G=(X,E)(|X|=n)$, we choose the class of symmetric isotone functions, that is, real valued functions

$$F(a_1,a_2,\ldots,a_{n-1}),\qquad a_i \ \text{integer}$$

such that

(1) $F(a_1,a_2,\ldots,a_{n-1})=F(a_{\sigma(1)},a_{\sigma(2)},\ldots,a_{\sigma(n-1)})$, where σ is any permutation on $\{1,2,\ldots,n-1\}$;

(2) $F(a_1,a_2,\ldots,a_{n-1})\geqslant F(b_1,b_2,\ldots,b_{n-1})$ when $a_i\geqslant b_i$, $i=1,\ldots,n-1$.

We now show, see Theorem 6, that if F is a criterion function for a triangulated graph $G=(X,E)$ with distinct monotone transitive orderings α and β, then

$$F(d(\alpha(1)),d(\alpha(2)),\ldots,d(\alpha(n-1))) = F(d(\beta(1)),d(\beta(2)),\ldots,d(\beta(n-1))).$$

Furthermore, by Theorem 7, if γ is any nonmonotone transitive ordering of X, then $G_\alpha=(X,E,\alpha)$ is a subgraph of $\mathrm{MTE}(G_i;\ \gamma)$ and we show

$$F(d(\gamma(1)),d(\gamma(2)),\ldots,d(\gamma(n-1))) > F(d(\alpha(1)),d(\alpha(2)),\ldots,d(\alpha(n-1))).$$

Thus, with respect to criterion functions on triangulated graphs, monotone transitive orderings may be regarded as *optimal*.

THEOREM 6. Let $G=(X,E)$ be triangulated and let α and β be two distinct monotone transitive orderings of X. Then the two partitions of $|E|$ generated by the $d(\alpha(i))$ and the $d(\beta(i))$ are equal.

Proof: We use induction on $|X|$. The case $|X| = 2$ is clear. Suppose that, in the case $|X| = k-1$, we consider G with $|X| = k$. If $\alpha(1) = \beta(1) = x$, the result follows immediately from the induction hypothesis on the elimination graph G_x. Suppose, then, that $\alpha(1) = y$ and $\beta(1) = z$. Note that if $y \in \mathrm{adj}(z)$ then $\mathrm{adj}(y) - \{z\} = \mathrm{adj}(z) - \{y\}$ by monotone transitivity and $|\mathrm{adj}(y)| = |\mathrm{adj}(z)|$. Consider the new monotone transitive orderings $\hat{\alpha}$ and $\hat{\beta}$ defined by

$$\hat{\alpha}(1) = y, \qquad \hat{\beta}(1) = z,$$
$$\hat{\alpha}(2) = z, \qquad \hat{\beta}(2) = y,$$
$$\hat{\alpha}(i) = \hat{\beta}(i) = \gamma(i-2), \qquad i = 3,\ldots,n,$$

where γ is any monotone transitive ordering of the triangulated section graph $G(X - \{x,y\})$. By the first part of the proof, the partitions generated by $d(\hat{\alpha}(i))$ and $d(\alpha(i))$ are equal, as are those generated by $d(\hat{\beta}(i))$ and $d(\beta(i))$. It remains to show that the partitions generated by $d(\hat{\alpha}(i))$ and $d(\hat{\beta}(i))$ are equal. Now $\hat{\alpha}$ yields the partition $|\mathrm{adj}(y)|$, $|\mathrm{adj}(z)|$, $d(\gamma(i-2))$, $i = 3,\ldots,n$, if $y \notin \mathrm{adj}(z)$ and $|\mathrm{adj}(y)|$, $|\mathrm{adj}(z)| - 1$, $d(\gamma(i-2))$, $i = 3,\ldots,n$, if $y \in \mathrm{adj}(z)$. But β gives an equal partition in each case because $|\mathrm{adj}(y)| = |\mathrm{adj}(z)|$ if $y \in \mathrm{adj}(z)$.

LEMMA 8. Let $G = (X, F)$ be triangulated with triangulated subgraph $\hat{G} = (X, E)$, $E \subseteq F$. If α is any monotone transitive ordering for both G and $\hat{G}$, then the degree partition of F dominates the degree partition of E.

Proof: Clearly, $\hat{d}(x_1) \leqslant d(x_1)$, and the elimination graphs $\hat{G}_1 = (X - \{x_1\}, E_1)$ and $G_1 = (X - \{x_1\}, F_1)$ are triangulated with $E_1 \subseteq F_1$. The proof then follows by induction on $|X|$.

LEMMA 9. Let $G = (X, E)$ be triangulated and $x \in X$. Then $\hat{G} = (X, E \cup D(x))$ is triangulated.

Proof: Assuming $D(x) \neq \varnothing$, we need only show that cycles in $\hat{G}$ of the form

$$\mu = [x_1, y_1, p_1, \ldots, p_n, x_1], \qquad n \geqslant 2$$

with $\{x_1, y_1\} \in D(x)$ have a chord. These are two cases.

(1) if some $p_i \in \mathrm{adj}(x)$, then there is a chord $\{x_1, p_i\}$ (or $\{y_1, p_n\}$, if $i = n$) in $E \cup D(x)$;

(2) if no $p_i \in \mathrm{adj}(x)$,the cycle $\mu' = [x_1, x, y_1, p_1, \ldots, p_n, x_1]$ in G has a chord $\{x_1, p_i\}$, $\{y_1, p_j\}$, or $\{p_i, p_j\}$ in E, which is also in chord in $\hat{G}$.

LEMMA 10. Let $\hat{G} = (X, E)$ and $G = (X, F)$ be triangulated with strict inclusion $E \subset F$. Then there exists a monotone transitive ordering α for G such that in $\hat{G}$, M TE$(\hat{G}; \alpha) = (X, E \cup T(\alpha))$ with strict inclusion $(E \cup T(\alpha)) \subset F$.

Proof: If X is a clique in G, the lemma is true for any α which is a monotone transitive ordering for $\hat{G}$. Hence, we assume X is not a clique in G and prove the assertion by induction on $|X|$. One easily verifies the cases when $|X| = 4$, and, assuming the case $|X| = k-1$, we consider such graphs G and $\hat{G}$ with $|X| = k$.

Let $S = \{y \in X \mid D(y) = \varnothing \text{ in } G\}$. We first dispense with two cases.

(1) If for some $y \in S$, $[\mathrm{adj}(y)]_{\hat{G}} \subset [\mathrm{adj}(y)]_G$, that is, there is an edge $e = \{y, x\} \in F - E$, then by choosing any monotone transitive ordering for G with $\alpha(1) = y$ we have $(E \cup T(\alpha)) \subset F$.

(2) If some $y \in S$ with $[\mathrm{adj}(y)]_{\hat{G}} = [\mathrm{adj}(y)]_G$ has $D(y) = \varnothing$ in $\hat{G}$ also, then by choosing $\alpha(1) = y$, the lemma follows by the induction hypothesis on the elimination graph G_y and $\hat{G}_y$.

These cases being dismissed, we may assume that for each $y \in S$, $[\mathrm{adj}(y)]_{\hat{G}} = [\mathrm{adj}(y)]_G$, and that the clique $\mathrm{adj}(y)$ in G contains at least one pair of vertices $e_y = \{v_1, v_2\} \in F - E$. By Corollary 2 and Lemma 6 (Section 3), since X in G is not a clique, there exists $y, z \in S$ with $y \notin \mathrm{adj}(z)$. For such vertices $e_y \neq e_z$, because if $e_y = e_z = \{v_1, v_2\}$, the cycle $\mu = [y, v_1, z, v_2, y]$ has no chord in E, so $\hat{G}$ could not be triangulated.

Hence, for some $y \in S$, choose $\alpha(1) = y$ and consider the y-elimination graphs $\hat{G}_y = (X - \{y\}, E_1)$ and $G_y = (X - \{y\}, F_1)$. It is clear from the above that strict inclusion $E_1 \subset F_1$ holds because there exists a $z \in S$ with $y \notin \mathrm{adj}(z)$ and such that $e_z \in F_1$, but $e_z \notin E_1$. Also, by Lemma 9, $\hat{G}_y$ is triangulated, as is G_y. Hence, the lemma follows by using induction on the graphs G_y and $\hat{G}_y$.

These lemmas give us Theorem 7.

THEOREM 7. Let $\hat{G} = (X, E)$ and $G = (X, F)$ be triangulated with $E \subseteq F$. Let α and β be monotone transitive orderings of $\hat{G}$ and G, respectively. Then the degree partition of $|F|$ dominates the degree partition of $|E|$.

Proof: We use induction of $|F|$. If $|F| = |X| - 1$, that is, G is a tree, then $E = F$, because both G and $\hat{G}$ are assumed connected, and the conclusion follows from Theorem 6. Suppose the theorem is true whenever $|X| - 1 < |F| \leqslant k - 1$, and let G and $\hat{G}$ be as above with $|F| = k$. If the subgraph $\hat{G} = (X, E)$ has $E = F$, then again the conclusion follows from Theorem 6. Assume then $E \subset F$ (strict). By Lemma 10, there exists a monotone transitive ordering $\hat{\alpha}$ for G such that $\mathrm{MTE}(\hat{G}; \hat{\alpha}) = (X, E \cup T(\hat{\alpha}))$ and $(E \cup T(\hat{\alpha})) \subset F$ (strict). By the induction hypothesis, the degree partition of $|E \cup T(\hat{\alpha})|$, generated by $\hat{\alpha}$ in the triangulated graph $\mathrm{MTE}(\hat{G}; \hat{\alpha})$, dominates the degree partition of $|E|$, and by Lemma 8 the degree partition of $|F|$ generated by $\hat{\alpha}$ dominates the degree partition of $|E \cup T(\hat{\alpha})|$. By Theorem 6, the degree partitions of $|F|$ generated by β and $\hat{\alpha}$ are equal.

Theorem 7 has the following important implication: If T is a triangulation of a graph $G = (X, E)$, the T is *minimal* if no $\hat{T} \subset T$ is also a triangulation of G. Clearly, a minimum triangulation is minimal, but a minimal triangulation need not be minimum. Theorem 7 implies that if T is a nonminimal triangulation of G, and $\hat{T} \subset T$ is also a triangulation, then the cost of elimination with a monotone transitive ordering of $G_1 = (X, E \cup T)$ is greater than the cost of elimination with a monotone transitive ordering of $G_2 = (X, E \cup \hat{T})$ for any criterion function.

4.3. *Bounds for Triangulations*

Since the size of a triangulation T of a nontriangulated graph G is one indication of the computational complexity of G, that is, M, with respect to elimination, we seek bounds on $|T|$ which are related to the structure of G. Corollary 5 below relates the size of a minimal triangulation to the size of minimal a, b separators in G. Theorem 9 shows that if k edges of G can be deleted to yield a triangulated graph, then G itself can be triangulated with $|T| \leqslant kn$.

THEOREM 8. Let $G = (X, E)$ be a graph with minimal triangulation T. Then every minimal a, b separator of $\hat{G} = (X, E \cup T)$ is a minimal a, b separator of G.

Proof: If S is an a, b separator of $\hat{G}$, clearly it is an a, b separator of G. Suppose S is minimal in $\hat{G}$ but not in G, that is, $S' \subset S$ is also an a, b separator in G. Let C_i be the components of G with respect to S'. Since some vertices in S are in the C_i, returning to $\hat{G}$ where S is minimal implies there must be edges $T_0 \subset T$ with vertices in different components C_i. S' is a clique in $\hat{G}$, and by removing edges in T_0 the graph $\tilde{G} = (X, E \cup T - T_0)$ is triangulated by Theorem 3. Thus, T is not minimal, which contradicts our hypothesis.

COROLLARY 5. Let $G = (X, E)$ be a graph, $|X| = n$, such that every minimal a, b separator S of G satisfies $|S| \leqslant k$. If T is a minimal triangulation of G, then

$$|T| \leqslant (n - \mu)(k(k+1)/2)$$

where

$$\mu = \max\{|C|, C \text{ a clique of } G\}.$$

Proof: By the remark following the statement of Theorem 4, there are, at most, $n - \mu$ minimal a, b separators in $\hat{G} = (X, E \cup T)$. The proof then follows by the above theorem and the hypothesis $|S| \leqslant k$.

THEOREM 9.† Let $T=(X, E)$ be triangulated and $G=(X, E \cup F)$. Let $V \subseteq X$ be a set of vertices covering F, that is, if $f=\{x, y\} \in F$, then $x \in V$ or $y \in V$. If $|X|=n$ and $|V|=m$, G can be triangulated with, at most,

$$nm-(m(m+1)/2) \quad \text{edges.}$$

Proof: Let $H=(X, E \cup F \cup T)$, where $T=\{\{v, x\} \mid v \in V, x \in X, v \neq x\}$. Note that $|T|=nm-(m(m+1)/2)$. We show H is triangulated. If μ is a cycle of H with length $l \geqslant 4$ and

(1) μ contains no vertex in V, then μ is a cycle of T and, hence, has a chord;
(2) μ contains a vertex $v \in V$, then H contains the chord $\{vx\}$ for any $x \in \mu$ not adjacent to v.

Note that the bound in Theorem 9 can be improved, if G has some separation cliques which are also separation cliques of T.

4.4. *Ordering Algorithms*

4.4.1. DYNAMIC PROGRAMMING

Given any criterion function F as defined in Section 4.2 and a graph G, it is possible to find an ordering $\hat{\alpha}$ which minimizes F by using the dynamic programming technique of Bertele, Brioschi, and Even [4, 5, 7], who consider the specific criterion function

$$y(\alpha)=\max_{1 \leqslant i \leqslant n-1} d(\alpha(i)).$$

However, for a graph with n vertices, the complexity of this algorithm and the storage requirements increase as 2^n. Hence, this algorithm is not feasible for large graphs, and no other general algorithm ensuring optimality is known.

In practice, it is tacitly agreed that a near optimal ordering is acceptable if the ordering algorithm is efficient. For example, the complexity grows only as n^p, p small. In the literature‡ it is assumed that the next two algorithms to be discussed produce near optimal orderings. While some experimental results reported by Tinney [29, p. 25], confirm this assumption, no detailed study of these algorithms has been reported.

† From a private communication from A. Hoffman, IBM research, Yorktown Heights, New York.

‡ See, for example, Sato and Tinney [20], Tinney and Walker [26], and the summary paper by Tinney [29, p. 25], who use these algorithms for ordering sparse symmetric matrices. For similar algorithms applied to the nonsymmetric case, see Tewarson [29, p. 35].

4.4.2. MINIMUM DEGREE ALGORITHM

Let $G_0 = G = (X, E)$. The minimum degree algorithm orders X as follows:

(1) set $i = 1$;
(2) in the elimination graph G_{i-1}, choose x_i to be any vertex such that

$$|\mathrm{adj}(x_i)| = \min_{y \in X_{i-1}} |\mathrm{adj}(y)|$$

where

$$G_{i-1} = (X_{i-1}, E_{i-1});$$

(3) set $i = i+1$;
(4) if $i > |X|$, stop;
(5) go to Step (2).

The advantage of this algorithm is its speed. $n(n+1)/2$ vertices are tested, and each test simply counts adjacent vertices. The disadvantages of the algorithm are

(1) the algorithm does not, in general, produce a monotone transitive ordering when the graph is triangulated (see Fig. 6)†;

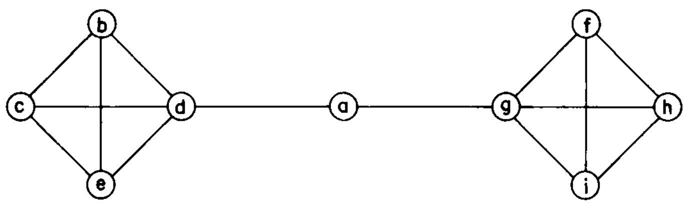


Fig. 6. Vertex ⓐ has minimum degree in the triangulated graph G. Since $\{a\}$ is a separation clique, ordering ⓐ first leads to a nominimal triangulation.

(2) the algorithm does not, in general, produce a minimal triangulation (again see Fig. 6);
(3) there exist examples when the triangulation produced by this ordering is arbitrarily greater than a minimum triangulation (see following Example and Fig. 7).

Example: Let $n < m$ and C_{m-1} be a clique on $m-1$ vertices. Each of the vertices a_i is adjacent to each vertex of the clique C_{m-1}. Vertex x is adjacent to each a_i. Vertex x has minimum degree, $|\mathrm{adj}(x)| = n$, and the elimination graph G_x is the clique C_{n+m-1}. This triangulation obtained by ordering x first requires $n(n-1)/2$ edges. However, the triangulation obtained by ordering the a_i first (note that $|\mathrm{adj}(a_i)| = m-1$) requires only $m-1$ edges.

† It is easy to see that the algorithm will produce a monotone transitive ordering when G is a k tree.

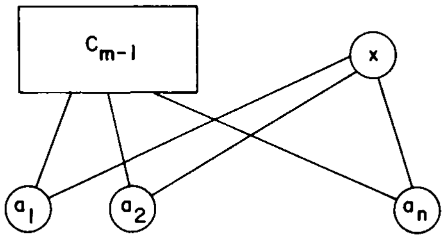


Fig. 7

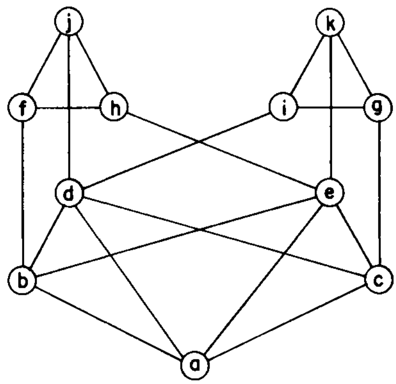


Fig. 8. Vertex ⓐ has $|D(a)| = 2$ and is a minimum deficiency vertex. However, in the elimination graph G_a, the edge $\{b,c\}$ is redundant in a triangulation given edge $\{d, e\}$ since $S = \{a, d, e\}$ is then a,b,c separation clique of G_a. Thus, given a deficiency set $D(x)$, only some subset of $D(x)$ may be necessary in a minimal triangulation.

4.4.3. Minimum Deficiency Algorithm

Letting $G_0 = G = (X, E)$, the minimum deficiency algorithm orders X as follows:

(1) set $i = 1$;
(2) in the elimination graph G_{i-1} choose x_i to be any vertex such that

$$|D(x_i)| = \min_{y \in X_{i-1}} |D(y)|,$$

where

$$G_{i-1} = (X_{i-1}, E_i);$$

(3) set $i = i+1$;
(4) if $i > |X|$, stop;
(5) go to Step (2).

The advantages of the minimum deficiency algorithm are

(1) only $n(n+1)/2$ deficiency counts are needed to compute the ordering;
(2) the algorithm produces a monotone transitive ordering when the graph is triangulated; also, in this case, ordering a vertex as soon as the $D(x)=\varnothing$ condition is recognized leads to *fewer* than $n(n+1)/2$ deficiency counts.

The disadvantages are

(1) the algorithm is slower than the minimum degree algorithm, because in addition counting, or listing, vertices in $\mathrm{adj}(x)$, pairs of vertices in $\mathrm{adj}(x)$ must be edge tested;
(2) the algorithm does not, in general, produce a minimal triangulation (see Fig. 6, with edges $\{c,d\}$, $\{b,e\}$, $\{f,i\}$, $\{g,h\}$ deleted; see Fig. 8).

References

1. Akyuz, F. A. and Utku, S. An automatic relabeling scheme for bandwidth minimization of stiffness matrices, *AIAA J. 6*, 728–730 (1968).
2. Alway, G. and Martin, D., An algorithm for reducing the band width of a matrix of symmetrical configuration, *Comput. J.* **8**, 264–272 (1965).
3. Berge, C., Some classes of perfect graphs, "Graph Theory and Theoretical Physics" (F. Harary, ed.), pp. 155–166. Academic Press, New York, 1967.
4. Bertele, U. and Brioschi, F., A new algorithm for the solution of the secondary optimization problem in nonserial dynamic programming, *J. Math. Anal. Appl.* **27**, 565–574 (1969).
5. Bertele, U., and Brioschi, F., Contributions to nonserial dynamic programming, *J. Math. Anal. Appl.* **28**, 313–325 (1969).

5a. Boland J. and Lekkerkerker C., Representation of a finite graph by a set of intervals on the real line, *Fundam. Math.* **51**, 45–64 (1962).

6. Brayton, R. K., Gustavson, F. G., and Willoughby, R. A., Some results on sparse matrices, *Math. Comp.* (on press).
7. Brioschi, E., and Even, S., Minimizing the number of operations in certain discreet variable optimization problems, *Tech. Rep. 567, Harvard University, Div. Eng. and Applied Physics.*
8. Busacker, R. G., and Saaty, T. L., "Finite Graphs and Networks." McGraw-Hill, New York, 1965.
9. Cuthill, E., and McKee, J., Reducing the bandwidth of sparse symmetric matrices, *Proc. ACM 23rd National Conf.* (1969).

9a. Dirac, G. A., On rigid circuit graphs, *Abh. Math. Sem. Univ. Hamburg* **25**, 71–76 (1961).

10. Dulmage, A. L., and Mendelsohn, N. S., On the inversion of sparse matrices, *Math. Comp.* **16**, 494–496 (1962).
11. Dulmage, A. L., and Mendelsohn, N. S.,Two algorithms for bipar ite graphs, *SIAM J. Appl. Math.* **11**, 183–194 (1963).

12. Forsythe, G. E., and Moler, C. B., "Computer solution of Linear Algebraic Systems." Prentice-Hall, Englewood Cliffs, New Jersey, 1967.
12a. Fulkerson, D., and Gross, O., Incidence matrices and interval graphs, *Pacific J. Math.* **15**, 835–855 (1965).
13. Harary, F., A graph theoretic method for the complete reduction of a matrix with a view toward finding its eigenvalues, *J. Math. and Phys.* **38**, 104–111 (1959).
14. Harary, F., A graph theoretic approach to matrix inversion by partitioning, *Numer. Math.* **4**, 128–135 (1962).
15. Gustavson, F. G., Liniger, W., and Willoughby, R.,Symbolic generation of an optimal Crout algorithm for sparse systems of linear equations, *J. Assoc. Comput. Mach.* **17**, 87–109 (1970).
16. Livesley, R. K.,The analysis of large structural systems, *Comput. J.* **3**, 34–39 (1960).
17. Parter, S., The use of linear graphs in Gauss elimination, *SIAM Rev.* **3**, 119–130 (1961).
18. Rose, D. J., Symmetric elimination on sparse positive definite systems and the potential flow network problem, Ph.D. Thesis, Harvard University (1970).
19. Rosen, R., Matrix bandwidth minimization, *Proc. ACM 23rd National Conf.* 585–595 (1968).
20. Sato, N., and Tinney, W. F., Techniques for exploiting the sparsity of the network admittance matrix, *IEEE, PAS*, 944–950 (1963).
21. Steward, D. V., Partitioning and tearing systems of equations, *SIAM J. Numer. Anal.* **2**, 345–365 (1965).
22. Tewarson, R. P., On the product form of inverses of sparse matrices, *SIAM Rev.* **8**, 336–342 (1966).
23. Tewarson, R. P., The product form of inverses of sparse matrices and graph theory *SIAM Rev.* **9**, 91–99 (1967).
24. Tewarson, R. P., Row–column permutation of sparse matrices, *Comput. J.* **10**, 300–305 (1967).
25. Tewarson, R. P., Solution of a system of simultaneous linear equations with a sparse coefficient matrix by elimination methods, *BIT* **7**, 226–239 (1967).
26. Tinney, W. F., and Walker, J. W., Direct solutions of sparse network equations by optimally ordered triangular factorization, *Proc. IEEE* **55**, 1801–1809 (1967).
27. Westlake, J. R., "A Handbook of Numerical Matrix Inversion and Solution of Linear Equations." Wiley, New York, 1968.
28. Wilkinson, J. H., "The algebraic Eigenvalue Problem." Oxford Univ. Press (Clarendon), London and New York, 1965.
29. Willoughby, R. A. (ed.), IBM sparse matrix proceedings, *IBM Report RAI*, No. 11707 (1969).