IMA Journal of Numerical Analysis (1991) **11**, 181–204

The Factorization of Sparse Symmetric Indefinite Matrices

I. S. DUFF, N. I. M. GOULD, J. K. REID, J. A. SCOTT
Atlas Centre, Rutherford Appleton Laboratory, Oxon OX11 0QX

AND

K. TURNER
Department of Mathematics, Utah State University, Logan, Utah 84322-3900, USA

[Received 8 December 1989 and in revised form 11 July 1990]

The Harwell multifrontal code MA27 is able to solve symmetric indefinite systems of linear equations such as those that arise from least-squares and constrained optimization algorithms, but may sometimes lead to many more arithmetic operations being needed to factorize the matrix than is required by other strategies. In this paper, we report on the results of our investigation of this problem. We have concentrated on seeking new strategies that preserve the multifrontal principle but follow the sparsity structure more closely in the case when some of the diagonal entries are zero.

1. Introduction

WE consider the direct solution of sparse symmetric systems of linear equations in which the coefficient matrix is indefinite because some of its diagonal entries are zero. As an example of applications in which such linear systems arise, consider the equality constrained least-squares problem

$$\underset{x \in \mathbb{R}^n}{\text{minimize}} \; \|Bx - b\|_2 \tag{1.1}$$

subject to

$$Cx = d, \tag{1.2}$$

where B is an $m \times n$ matrix, b is a known m-vector, C is a $k \times n$ matrix, and d is a known k-vector. It is presumed that $k \leqslant n \leqslant k + m$. This problem is equivalent to solving the sparse symmetric linear system

$$\begin{bmatrix} I & & B \\ & 0 & C \\ B^T & C^T & 0 \end{bmatrix} \begin{bmatrix} r \\ \lambda \\ x \end{bmatrix} = \begin{bmatrix} b \\ d \\ 0 \end{bmatrix}. \tag{1.3}$$

As another example, consider the quadratic programming problem

$$\underset{x \in \mathbb{R}^n}{\text{minimize}} \; \tfrac{1}{2}x^T H x + c^T x \tag{1.4}$$

subject to the linear equality constraints (1.2), where H is an $n \times n$ symmetric matrix. Such problems arise both in their own right and as subproblems in constrained optimization calculations (see Gill, Murray, & Wright, 1981). Under a suitable inertial condition, the problem of solving (1.4) subject to the constraints (1.2) is equivalent to solving the symmetric but indefinite system of linear equations

$$\begin{bmatrix} H & C^{\mathrm{T}} \\ C & 0 \end{bmatrix}\begin{bmatrix} x \\ \lambda \end{bmatrix} = \begin{bmatrix} -c \\ d \end{bmatrix}. \tag{1.5}$$

The present Harwell Subroutine Library code MA27 uses a multifrontal solution technique to solve sparse symmetric linear systems (Duff & Reid, 1983). The preliminary analysis phase chooses a tentative pivot sequence from the sparsity pattern alone, assuming that the matrix is definite so that all the diagonal entries are nonzero and suitable as 1 × 1 pivots. For the indefinite case, this tentative pivot sequence is modified in the factorization phase to maintain stability by delaying the use of a pivot if it is too small or by replacing two pivots by a 2 × 2 block pivot (Bunch & Parlett, 1971).

The assumption that all the diagonal entries are nonzero is clearly violated in the above examples and for such problems the fill-in during the factorization phase of MA27 can be significantly greater than predicted by the analysis phase (Gill, Murray, & Saunders, 1989). The aim of this study is to improve the effectiveness of MA27 when the coefficient matrix has some zero diagonal blocks by exploiting these blocks during the analysis and factorization phases and, in particular, by allowing the inclusion of 2 × 2 pivots in the tentative pivot sequence selected by the analysis phase. The use of a 2 × 2 pivot with one or both of its diagonal entries zero may preserve a whole block of zeros, a possibility that was not available when the diagonal entries were treated as nonzeros. In Section 2 we briefly review the strategy followed by MA27, and in Section 3 we outline our proposed modifications to the analysis phase. In Section 4 we discuss corresponding modifications to the factorization phase and propose a new stability criterion for 2 × 2 pivots. In Section 5 we give some theoretical results for our proposed pivotal strategies when applied to special classes of problems, and we present numerical results for a range of problems in Section 6. These numerical results were obtained with an experimental code that is much easier to modify than MA27 but runs less efficiently. Finally, in Section 7 concluding comments are made. We will be modifying MA27 itself in the light of our conclusions.

Throughout the paper, we use the notation a_{ij} to refer to an entry in the current reduced matrix, and r_i to denote the number of nonzero entries in row i of the current reduced matrix. This latter quantity is known as the *row count* of row i.

2. The strategy of MA27

MA27 has three distinct phases: the analysis phase chooses a tentative pivot sequence from the sparsity pattern, the factorization phase performs a numerical factorization of a matrix with the same pattern using the tentative pivot sequence as its guide, and the solve phase uses the numerical factorization to solve a set of equations.

Efficiency is achieved during the analysis and factorization phases by the multifrontal technique. At a typical intermediate stage, the reduced matrix is stored as a sum of original entries and of 'generated element matrices', each created during an earlier pivotal step. Each generated element matrix has nonzeros in a limited number of rows and columns and is stored as a dense matrix and an index set. When a pivot is chosen, the original entries of the pivot row and column and the generated element matrices that involve the pivot row are added together (a process usually called 'assembly'). If the pivot is chosen well, the result again has nonzeros in only a small number of rows and columns and so can be stored as a dense matrix and an index set. The pivotal operations are performed within this 'frontal matrix' and a new generated element matrix results. The process is particularly efficient during the analysis phase since only the index sets need to be stored and manipulated.

During the analysis phase of MA27, a tentative pivot sequence is chosen using the minimum degree criterion, assuming that any diagonal entry is nonzero and suitable as a pivot. The tentative pivot sequence of 1 × 1 pivots chosen during the analysis phase may be modified during the factorization phase to maintain stability and 2 × 2 pivots may be used. The stability condition that 1 × 1 pivots are required to satisfy is

$$|a_{kk}| > u \max_{j \neq k} |a_{kj}|, \tag{2.1}$$

where u is a user-specified parameter in the range $0 \leqslant u \leqslant \frac{1}{2}$. If (2.1) is not satisfied and $u > 0$, MA27 attempts to use a 2×2 pivot. For a 2×2 pivot the stability condition

$$\left\| \begin{bmatrix} a_{kk} & a_{k,k+1} \\ a_{k+1,k} & a_{k+1,k+1} \end{bmatrix}^{-1} \right\|_{\infty} \max_{j \neq k,k+1} [\max(|a_{kj}|, |a_{k+1,j}|)] \leqslant u^{-1} \tag{2.2}$$

is used. The limit $\frac{1}{2}$ on the parameter u ensures that a full set of pivots will be chosen. These modifications to the pivot sequence usually lead to generated element matrices that are larger than anticipated by the analysis phase because they contain rows and columns that were expected to have been eliminated but are still present since stable pivots cannot be chosen for them.

3. Modifications to the analysis phase of MA27

In this section we shall discuss the modifications to the analysis phase of MA27 that we have considered. We first introduce some definitions and terminology.

We say that a variable is *defective* if the corresponding diagonal entry of the current reduced matrix is known to be zero; otherwise we say that the variable is *nondefective*. A variable that is initially defective, and becomes nondefective because its diagonal entry fills in, is assumed to remain nondefective until it is eliminated.

The *degree* of a variable is the number of nonzero off-diagonal entries in its row of the current reduced matrix. The degree of a 2 × 2 pivot is the number of columns of the reduced matrix, apart from the columns of the pivot, that have an entry in either of the pivot rows.

We have chosen the term *oxo* pivot for a 2 × 2 pivot of the form

$$\begin{bmatrix} 0 & a \\ a & 0 \end{bmatrix}, \tag{3.1}$$

which involves two defective variables i and j for which the matrix entry $a = a_{ij} = a_{ji}$ is nonzero. We use the term *tile* pivot for a 2 × 2 pivot of the form

$$\begin{bmatrix} h & a \\ a & 0 \end{bmatrix}, \tag{3.2}$$

which involves a nondefective variable i and a defective variable j for which the matrix entry $a = a_{ij} = a_{ji}$ is nonzero. We use the term *full* pivot for a 2 × 2 pivot of the form

$$\begin{bmatrix} h & a \\ a & k \end{bmatrix}, \tag{3.3}$$

which involves two nondefective variables i and j for which the matrix entry $a = a_{ij} = a_{ji}$ is nonzero.

Following Gill, Murray, Saunders, & Wright (1989), we say that a symmetric $2n \times 2n$ matrix $\boldsymbol{T}$ is a *tiled* matrix if $\boldsymbol{T}$ is of the form

$$\boldsymbol{T} = \begin{bmatrix} T_{11} & T_{12} & \dots & T_{1n} \\ T_{21} & T_{22} & \dots & T_{2n} \\ \vdots & \vdots & & \vdots \\ T_{n1} & T_{n2} & \dots & T_{nn} \end{bmatrix}, \tag{3.4}$$

where T_{ij}, $1 \leqslant i,j \leqslant n$, is a 2 × 2 matrix

$$T_{ij} = \begin{bmatrix} h & a \\ b & 0 \end{bmatrix} \tag{3.5}$$

and $T_{ij} = T_{ji}^{\mathrm{T}}$. We call each T_{ij} a tile.

Gill, Murray, Saunders, and Wright (1989) report obtaining improved performance on problems of the form (1.5) in which $\boldsymbol{C}$ is an almost square matrix, by using their knowledge of the derivation of the problem to pair most of the variables belonging to the matrix $\boldsymbol{H}$ with variables belonging to the zero block. They then permute the coefficient matrix to the form

$$\begin{bmatrix} \boldsymbol{T} & \boldsymbol{F} \\ \boldsymbol{F}^{\mathrm{T}} & \boldsymbol{E} \end{bmatrix}, \tag{3.6}$$

where $\boldsymbol{T}$ is a *tiled* matrix and the remaining blocks are relatively small. Gill *et al.* gave the structure of the *tiled* matrix to the analysis phase of MA27 by treating T_{ij}

as a single entry that was nonzero if and only if T_{ij} was a nonzero matrix. An ordinary pivot sequence was constructed for the factorization phase by expanding the data structure to consider each tile as a 2 × 2 matrix, with the additional variables placed in arbitrary order at the end. We seek to improve the performance of MA27 on problems of the form (1.5) without the need for any special knowledge of the problem.

3.1 *Minimum Degree Criterion for* 2 × 2 *Pivots*

We considered extending the idea of minimum degree ordering to include 2×2 pivots. Since calculating the degree of a potential 2 × 2 pivot would be expensive, we chose to approximate it with the larger of the degrees of the two variables involved. This allowed us to limit the choice of pivots to 1 × 1 pivots and *oxo* pivots while still implementing a minimum degree strategy. We used the following algorithm for choosing a pivot, which is a minor modification of the existing MA27 algorithm:

```
for $d := 0$ step 1 until $n - 1$ do
begin
  if there is a nondefective variable of degree $d$ then
    accept it as a $1 \times 1$ pivot and goto exit;
  for each defective variable $i$ of degree $d$ do
    if there is a nonzero entry $(i, j)$ in the current reduced matrix such that
    the variable $j$ is defective and of degree $\leq d$ then
      accept $(i, j)$ as a $2 \times 2$ pivot and goto exit
end
exit:
```

We refer to this strategy as the 'minimum degree' strategy. Note that this algorithm gives preference to 1 × 1 pivots, a choice that we made on the grounds of efficiency in pivot selection.

For unconstrained least-squares problems (matrix C in equation (1.3) has no rows), the matrix (1.3) has no *oxo* pivots and nor will the reduced matrix following a 1 × 1 pivot. The same is true for the matrix in equation (1.5) if H has no zero diagonal entries. For such matrices, the above algorithm chooses only 1 × 1 pivots but differs from the original MA27 minimum degree algorithm because it chooses only nondefective variables as pivots.

We considered variations of the above minimum degree strategy, particularly because we wished to include 2 × 2 pivots for unconstrained least-squares problems and quadratic programming problems. We experimented with considering the variables of degree d in the order that the data structure dictates, accepting a 1 × 1 pivot in the nondefective case and performing the search for a 2 × 2 pivot of degree d in the defective case, now allowing a *tile* pivot. We also tried giving preference to 2 × 2 pivots over 1 × 1 pivots, and we considered forcing *tile* and *oxo* pivots as long as there remained any defective variables. These strategies were more expensive to implement than that which gives preference to 1 × 1 pivots and did not yield a significant or consistent improvement. Consequently, the detailed results of these experiments are not reported in this paper.

3.2 *Markowitz Cost Criteria for* 2 × 2 *Pivots*

The use of the 2 × 2 pivot

$$\begin{bmatrix} a_{ii} & a_{ij} \\ a_{ji} & a_{jj} \end{bmatrix} \tag{3.7}$$

is mathematically equivalent to pivoting on a_{ji} and then on a_{ij}. This property is both helpful in understanding the behaviour of the elimination step and in designing improvements. It makes it easy to see that if the reduced matrix has the form

$$\begin{bmatrix} \boldsymbol{A}_1 & \boldsymbol{A}_2 \\ \boldsymbol{A}_2^{\mathrm{T}} & \boldsymbol{0} \end{bmatrix}, \tag{3.8}$$

and we use a 2 × 2 pivot that has a diagonal entry in the zero block, the zero block is preserved. It would therefore seem to be desirable to favour such pivots. The minimum degree strategy, even if implemented exactly, gives little encouragement to such pivots. This has led us to consider the strategy of Markowitz (1957) as an alternative way of extending the strategy of minimum degree to 2 × 2 pivots. For a 1 × 1 pivot in row i, the Markowitz cost is

$$(r_i - 1)^2, \tag{3.9}$$

where r_i is the number of nonzeros in row i. Let us treat a 2 × 2 pivot as if performing ordinary elimination steps pivoting on a_{ji} and then on a_{ij}. The first pivot a_{ji} has Markowitz cost

$$(r_i - 1)(r_j - 1). \tag{3.10}$$

For an *oxo* pivot (3.1), the first pivot step causes no fill-in to row i or column j, so the second pivot has the same Markowitz cost. For a *tile* pivot (3.2), there is no fill-in to column j, but row i has a fill-in for each zero that corresponds to a nonzero of row j. If c_{ij} is the number of columns k such that a_{ik} and a_{jk} are both nonzero, the Markowitz cost of the second pivot a_{ij} is

$$(r_j - 1)(r_i - 2 + r_j - c_{ij}), \tag{3.11}$$

where c_{ij} satisfies the inequalities

$$1 \leq c_{ij} \leq \min(r_j, r_i - 1). \tag{3.12}$$

For a *full* pivot, row i has a fill-in for each zero that corresponds to a nonzero of row j, and column j has a fill-in for each zero that corresponds to a nonzero of column i. The Markowitz cost of the second pivot is

$$(r_i - 2 + r_j - c_{ij})^2, \tag{3.13}$$

where c_{ij} satisfies the inequalities

$$2 \leq c_{ij} \leq \min(r_i, r_j). \tag{3.14}$$

If we take the Markowitz cost of a 2 × 2 pivot to be the cost of the first pivot, it is possible that a large amount of fill-in may occur when the second pivot is used, making the pivot a poor choice. We therefore take the Markowitz cost of a 2 × 2 pivot to be the Markowitz cost of the second pivot. Restricting the pivot choice to 1 × 1, *oxo*, and *tile* pivots, at each stage we choose a pivot to minimize the Markowitz cost using the following algorithm:

```
for $r := 1$ step 1 until $n$ do
begin
  for each variable $i$ with row count $r$ do
  begin
    if variable $i$ is nondefective then
      accept it as a $1 \times 1$ pivot and goto exit
    else for each variable $j$ for which
      there is a nonzero entry $(i, j)$ in the current reduced matrix do
      begin
        if the Markowitz cost $\leq (r-1)^2$ then
          accept $(i, j)$ as a $2 \times 2$ pivot and goto exit;
        if the Markowitz cost is the smallest so far found then
          store it as such
      end
  end
  if there is a stored $2 \times 2$ pivot with Markowitz cost $\leq r^2$ then
    accept the $2 \times 2$ pivot and goto exit
end
exit:
```

We refer to this strategy with the Markowitz cost of a 2 × 2 pivot taken to be the Markowitz cost of the second pivot as the 'Markowitz' strategy. By storing the best 2 × 2 pivot and accepting it as the next pivot if it has Markowitz cost not exceeding r^2 when all variables with row count r have been considered, some preference is given to 2 × 2 pivots. Computing the Markowitz cost of the second pivot requires us to count the number of entries rows i and j have in common, a process that we regarded as prohibitively expensive in Section 3.1. The amount of work involved in this computation is reduced if we assume that the largest possible amount of fill-in occurs (that is, $c_{ij} = 1$ in expression (3.11)). We refer to this strategy as the 'Markowitz-b' strategy. Although taking the Markowitz cost of a 2 × 2 pivot to be the cost of the first pivot can lead to a large amount of fill-in when the second pivot is used, using the cost of the first pivot also avoids the need to count the entries rows i and j have in common. We have therefore experimented with this strategy, which we refer to as the 'Markowitz-1' strategy.

For a 1 × 1 pivot the Markowitz cost (3.9) bounds the fill-in that may occur in the reduced matrix during the pivot step. This motivated us to consider defining the cost of a 2 × 2 pivot to be the most fill-in that can occur in the reduced matrix during the pivot step. Calculating this cost requires the number of entries rows i and j have in common to be counted for both *oxo* and *tile* pivots so it is at least as expensive to implement as our Markowitz strategy. Our numerical experiments suggest that for some problems using this strategy can give a small improvement in the results compared with those obtained using our Markowitz strategy but for other problems our Markowitz strategy gave better results. The additional expense for *oxo* pivots leads us to prefer our Markowitz strategy. Results of the experiments using the fill-in cost are not presented in this paper.

3.3 *Avoiding Exact Cancellation*

It is possible for numerical dependencies to lead to pivots chosen during the analysis phase never being acceptable during the factorization phase. For example, suppose the leading submatrix has the sparsity pattern

$$\begin{bmatrix} \times & \times & \times & \times \\ \times & 0 & 0 & 0 \\ \times & 0 & 0 & \times \\ \times & 0 & \times & 0 \end{bmatrix}, \tag{3.15}$$

and suppose the analysis phase chooses the $(1, 1)$ entry as a 1×1 pivot. The generated element matrix will be full and the analysis phase can then choose variables 2, 3, and 4 as successive 1×1 pivots. In the factorization phase, choosing a_{11} and a_{22} as successive 1×1 pivots is equivalent to using variables 1 and 2 together as a *tile* pivot, so a_{33} and a_{44} remain zero (or have small values because of round-off) and cannot be used as 1×1 pivots. We can avoid such an occurrence for a 1×1 pivot by ensuring that the analysis phase never chooses one that was initially defective. This requires allowing the inclusion of *full* 2×2 pivots in the analysis phase. Our experience has been that this strategy leads to poorer results, and we conclude that numerical cancellation is not a serious problem. This is confirmed by the results we obtained when numerical values were used for selecting the pivot sequence (see Section 4.3 and Table 6.1).

3.4 *Storing the Generated Element Matrices*

When a 1×1 pivot is selected, the new generated element matrix will be full and all the variables associated with its rows will become nondefective. It is this property that leads to the efficiency of storing the generated element matrices as full submatrices in the definite case. In the indefinite case, a 2×2 pivot with a zero diagonal entry in row *i* produces no fill in the square block of rows and columns corresponding to the zeros of row *i*. We therefore propose to modify MA27 to hold generated element matrices of the forms

$$\begin{bmatrix} \mathbf{A}_1 & \mathbf{A}_2 \\ \mathbf{A}_2^{\mathrm{T}} & \mathbf{0} \end{bmatrix}, \tag{3.16}$$

and

$$\begin{bmatrix} \mathbf{0} & \mathbf{A}_3 \\ \mathbf{A}_3^{\mathrm{T}} & \mathbf{0} \end{bmatrix}. \tag{3.17}$$

Note that the present MA27 form of generated element matrices is the special case of (3.16) that occurs when the zero block is null. A *tile* pivot leads directly to a generated element matrix of the form (3.16). An *oxo* pivot leads to a generated element matrix of the form

$$\begin{bmatrix} A_1 & A_2 & A_3 \\ A_2^T & 0 & A_4 \\ A_3^T & A_4^T & 0 \end{bmatrix}, \qquad (3.18)$$

which can be represented as the sum of a matrix of the form (3.16) for the first block row and column and a matrix of the form (3.17) for the remainder.

This form of storage leads to a slightly more complicated assembly process, but the cost during the analysis phase remains linear in the lengths of the lists. If the leading block of a generated element matrix (3.16) involves a later pivot row, the generated element matrix can be absorbed without any loss of efficiency into the new generated element matrix for that pivot. However, if the trailing block of a generated element matrix (3.16) or a generated element matrix (3.17) involves a later pivot row, it may be impossible to absorb the old generated element matrix without increasing the size of the new generated element matrix. Instead, therefore, we simply remove the pivot row and column (or rows and columns) from the old generated element matrix and keep the rest. Having found the new generated element matrix, we check whether any of these outstanding old element matrices can in fact be absorbed.

4. Modifications to the factorization phase of MA27

The analysis phase of the current MA27 code assumes that all the variables are nondefective and the tentative pivot sequence contains only 1 × 1 pivots. At each stage of the factorization phase, if a pivot is unacceptable for stability reasons, a search is made for a suitable partner for use as a 2 × 2 pivot (details are given by Duff & Reid, 1983). We found that, for some problems, if we pass to the existing factorization phase the tentative pivot sequence obtained using our modified analysis phase without labelling the 2 × 2 pivots, considerable modifications were often made to the pivot sequence and the advantage of having incorporated 2 × 2 pivots within that sequence was largely lost. We therefore need to modify the data passed to the factorization phase to include the 2 × 2 pivots chosen by the analysis phase. During the analysis phase we flag the first entry in a potential 2 × 2 pivot and place its partner next in the pivot sequence. When we encounter a flagged entry in the factorization phase, the flagged entry and the entry immediately following it are tested for use as a 2 × 2 pivot. If this 2 × 2 pivot is unsuitable, we consider the first variable associated with the rejected pivot as if it were an unflagged entry.

Another change that we plan for the factorization phase is to work with generated element matrices of the forms (3.16) and (3.17) as for the analysis phase. Now there will be a substantial storage gain from not storing the zero blocks explicitly.

In addition to these changes, we have modified the stability test for 2 × 2 pivots, and we have introduced an additional test for tentative pivots based on the

Markowitz costs. These modifications are discussed in Sections 4.1 and 4.2. Another possibility is to use the numerical values when selecting the pivot sequence, that is to combine the analysis and factorization phases. We consider this in Section 4.3.

4.1 *A New Stability Condition*

We have found that the stability test (2.2) for 2 × 2 pivots can be too restrictive. It is based on ensuring that the modification

$$\delta_{ij} = (a_{ik} \quad a_{i,k+1})\begin{bmatrix} a_{kk} & a_{k,k+1} \\ a_{k+1,k} & a_{k+1,k+1} \end{bmatrix}^{-1}\begin{bmatrix} a_{kj} \\ a_{k+1,j} \end{bmatrix} \tag{4.1}$$

satisfies the inequality

$$|\delta_{ij}| \leqslant u^{-1} \max_{j} |a_{ij}| \, (\mu_{ik} + \mu_{i,k+1}), \tag{4.2}$$

where μ_{ij} is 0 or 1 according to whether a_{ij} is zero or not. By using the infinity norm in (4.1) we find the inequality

$$|\delta_{ij}| \leqslant (|a_{ik}| + |a_{i,k+1}|)\left\|\begin{bmatrix} a_{kk} & a_{k,k+1} \\ a_{k+1,k} & a_{k+1,k+1} \end{bmatrix}^{-1}\right\|_{\infty} \max_{j\neq k,k+1} (|a_{kj}| + |a_{k+1,j}|), \tag{4.3}$$

from which inequality (4.2) may be deduced if condition (2.2) holds.

This use of norms can lead to unnecessary rejection of pivots. If we replace (4.3) by the inequality

$$|\delta_{ij}| \leqslant (|a_{ik}|, |a_{i,k+1}|)\left|\begin{bmatrix} a_{kk} & a_{k,k+1} \\ a_{k+1,k} & a_{k+1,k+1} \end{bmatrix}^{-1}\right|\begin{bmatrix} |a_{kj}| \\ |a_{k+1,j}| \end{bmatrix}, \tag{4.4}$$

where the modulus notation for a matrix refers to taking the moduli of all its elements, we can replace (2.2) by the less stringent condition

$$\left|\begin{bmatrix} a_{kk} & a_{k,k+1} \\ a_{k+1,k} & a_{k+1,k+1} \end{bmatrix}^{-1}\right|\begin{bmatrix} \max\limits_{j\neq k,k+1} |a_{kj}| \\ \\ \max\limits_{j\neq k,k+1} |a_{k+1,j}| \end{bmatrix} \leqslant \begin{bmatrix} u^{-1} \\ u^{-1} \end{bmatrix}, \tag{4.5}$$

and thereby ensure that (4.2) holds.

A pivot which has Markowitz cost zero causes no change to the reduced matrix and needs no test for stability. Hence the stability conditions (2.1) and (4.5) for 1 × 1 and 2 × 2 pivots respectively are only employed if the Markowitz cost of a potential pivot exceeds zero.

We note that, in the case of a *tile* pivot ($a_{k+1,k+1} = 0$), the two parts of the new stability condition (4.5) are exactly those which result from applying the 1 × 1 pivot stability test (2.1) to the successive off-diagonal pivots $a_{k+1,k}$ and $a_{k,k+1}$.

We found that the old stability test (2.2) sometimes rejected a *tile* pivot chosen during the analysis phase and then used its two diagonal entries as successive 1 × 1 pivots. This can be very unsatisfactory from a sparsity point of view because it results in a full generated element matrix. The new test is less likely to split a 2×2 pivot into two successive 1×1 pivots, although it can still happen, as is illustrated by the matrix

$$\begin{bmatrix} 3 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \end{bmatrix} \tag{4.6}$$

with $u = \frac{1}{3}$. Here, with $k = 1$, the left-hand side of inequality (4.5) is

$$\left|\begin{bmatrix} 0 & 1 \\ 1 & -3 \end{bmatrix}\right| \begin{bmatrix} 1 \\ 1 \end{bmatrix} = \begin{bmatrix} 1 \\ 4 \end{bmatrix}, \tag{4.7}$$

so the *tile* is rejected. The first 1×1 pivot results in the reduced matrix

$$\begin{bmatrix} -\frac{1}{3} & \frac{2}{3} \\ \frac{2}{3} & -\frac{1}{3} \end{bmatrix}, \tag{4.8}$$

and it is seen that the second 1×1 pivot is also accepted.

When a 2×2 pivot is advantageous from a sparsity point of view, we avoid splitting it into two 1×1 pivots by accepting a 2×2 pivot that fails to satisfy the stability condition (4.5) if its two diagonal entries are acceptable as successive 1×1 pivots. Note that to implement this, it is necessary to flag the 2×2 pivots passed from the analyse phase to the factorization phase, as we discussed at the beginning of Section 4. We experimented with not passing flags to our new factorization phase. For many of our test problems, if the flags were not employed, there was little deterioration in the results (omitting the flags never improved the results), but for some problems there was a significant increase in the cost of the factorization.

4.2 *A Test on the Markowitz Cost*

With the Markowitz, Markowitz-1, or Markowitz-b strategy in use in the analysis phase, early 2×2 pivots may be chosen with only one row having a small number of entries. For example, a *tile* pivot (i, j) that has only a single entry in row j will have zero Markowitz cost for any number of entries in row i. During the factorization phase, any such pivot may become very undesirable from a sparsity point of view if changes in the pivot sequence have increased the number of entries in the short row.

We need some mechanism for recognizing this situation. We have chosen to add the expected Markowitz cost of each pivot (the cost found during the analysis phase) to the data passed from the analysis phase to the factorization phase. During the factorization phase we then delay any pivot whose actual Markowitz cost exceeds $1\frac{1}{2}$ times its expected Markowitz cost and, additionally, is greater than a prescribed threshold, until its Markowitz cost is less than or equal to $1\frac{1}{2}$ times the expected Markowitz cost of the latest candidate pivot. The threshold is taken to be k^2, for some k, so that any stable 1×1 pivot with not more than k off-diagonal entries in its row is accepted. We have experimented with various values of k. With $k = 2$, we found that too many desirable pivots were rejected, and with $k \geqslant 4$ we accept *oxo* and *tile* pivots which cause a significant amount of fill-in. We therefore chose $k = 3$. The factor of $1\frac{1}{2}$ permits us to accept pivots whose actual Markowitz costs are a limited amount larger than was predicted. Our numerical experiments suggest that the results are not very sensitive to this choice of $1\frac{1}{2}$, but we found that for some of our test problems using a factor as large as 2 did lead to a significant increase in the cost of the factorization.

Towards the end of the elimination, it is possible that the test on the Markowitz cost of a potential pivot prevents the selection of any pivot and at this point we cease to test the Markowitz cost of the pivots.

4.3 *Analysis using Numerical Values*

For unsymmetric problems, it is usual to use numerical values when choosing the pivot sequence. We would be reluctant to switch to this strategy here because it would mean the loss of an analysis phase that is much faster and requires much less storage than the corresponding factorization phase. We have included it in this study because of concern over numerical dependencies leading to pivots, chosen during the analysis phase, never being acceptable during the factorization phase, a possibility discussed in Section 3.3. By using numerical values when choosing pivots, we avoid all such problems.

We will refer to the strategy of using numerical values when selecting pivots following the Markowitz strategy as 'numerical'. This strategy will employ the new stability test (Section 4.1) and the Markowitz cost test (Section 4.2). Note that we may need *full* 2 × 2 pivots if both diagonal entries are too small for use as 1 × 1 pivots. The Markowitz cost of such a pivot was discussed in Section 3.2.

5. Theoretical results

In this section we present theoretical results for some special classes of problems.

THEOREM 5.1 *For a* $2n \times 2n$ *tiled matrix* T *with diagonal tiles*

$$T_{ii} = \begin{bmatrix} h_i & a_i \\ a_i & 0 \end{bmatrix}, \quad h_i \neq 0, \ a_i \neq 0, \ 1 \leqslant i \leqslant n, \tag{5.1}$$

the Markowitz and Markowitz-1 *analysis strategies choose n tile pivots.*

Proof. Since the diagonal tiles are all of the form (5.1), the matrix has no *oxo* pivots and nor will the reduced matrix following a 1 × 1 pivot or a *tile* pivot. It is therefore sufficient to show that *tile* pivots are selected in preference to 1 × 1 pivots.

We begin by showing that the first pivot chosen in the analysis phase using either the Markowitz strategy or the Markowitz-1 strategy is a *tile* pivot. Let a nondefective variable with the smallest row count be k. Since variable k is nondefective, variable $k+1$ is defective and, since T is a *tiled* matrix with nonzero diagonal tiles, r_{k+1} satisfies

$$r_{k+1} < r_k. \tag{5.2}$$

Let t_{ij}, $1 \leqslant i,j \leqslant 2n$, denote the 1×1 entries in T. For any nondefective variable l such that $t_{k+1,l}$ is nonzero, the *tile* pivot $(l, k+1)$ has Markowitz cost

$$(r_{k+1}-1)(r_l-2+r_{k+1}-c_{k+1,l}), \tag{5.3}$$

where

$$1 \leqslant c_{k+1,l} \leqslant \min(r_{k+1}, r_l-1) \tag{5.4}$$

(see (3.11)). Since T is a *tiled* matrix, $c_{k+1,k}=r_{k+1}$. It follows that r_l is minimized and $c_{k+1,l}$ is maximized when $l=k$, and that the best *tile* pivot has cost

$$(r_{k+1}-1)(r_k-2). \tag{5.5}$$

Comparing (3.9) (with $i=k$) with (5.5), we deduce that the best *tile* pivot has smaller Markowitz cost than the best 1×1 pivot, and a *tile* pivot is chosen as the first pivot.

If the Markowitz-1 strategy is used, the *tile* $(l, k+1)$ has cost

$$(r_{k+1}-1)(r_l-1) \tag{5.6}$$

(see (3.10)). This is minimized if $r_l=r_k$, and we again deduce that the first pivot is a *tile* pivot.

If a *tile* pivot is used, the reduced matrix after the first stage of the elimination is a *tiled* matrix with nonzero diagonal tiles. The result follows by induction. □

In Theorem 5.1, if the Markowitz strategy is employed, a *tile* pivot $(l, k+1)$ can have the same cost as the pivot $(k, k+1)$ only if rows k and l have the same sparsity pattern. Therefore, interchanging rows k and l has no effect on the sparsity pattern and it is as if the n *tile* pivots chosen in the analysis phase are all of the form $(k, k+1)$. This is not true if the Markowitz-1 strategy is used. In this case there may exist a nondefective variable $l \neq k$ such that the *tile* pivots $(l, k+1)$ and $(k, k+1)$ have the same cost while rows k and l have different patterns. Interchanging rows k and l then increases the number of nonzero tiles, so using the pivot $(l, k+1)$ may cause more fill-in than the pivot $(k, k+1)$.

We note that Theorem 5.1 does not hold if pivots are selected using the Markowitz-b strategy. For the Markowitz-b strategy, the best possible *tile* pivot $(l, k+1)$ with $r_l=r_k$ has cost (see (3.11))

$$(r_{k+1}-1)(r_k-3+r_{k+1}). \tag{5.7}$$

In this case it is possible that a 1×1 pivot has a smaller Markowitz cost. For example, suppose $k=1$ and

$$T_{11}=\begin{bmatrix} \times & \times \\ \times & 0 \end{bmatrix}, \qquad T_{1i}=\begin{bmatrix} \times & 0 \\ \times & 0 \end{bmatrix}, \quad i \geqslant 2. \tag{5.8}$$

In this example $r_k=r_{k+1}+1$, the *tile* pivot $(k, k+1)$ has cost

$$2(r_{k+1}-1)^2, \tag{5.9}$$

and the best 1×1 pivot has cost

$$r_{k+1}^2. \tag{5.10}$$

For $r_{k+1} \geqslant 4$ the inequality $2(r_{k+1}-1)^2 > r_{k+1}^2$ holds, and a 1 × 1 pivot is chosen as the first pivot in preference to the best *tile* pivot. A 1 × 1 pivot destroys the tiled structure.

THEOREM 5.2 *Let $\boldsymbol{B}$ be an $m \times n$ $(m \geqslant n)$ matrix of rank n and let $\boldsymbol{H}$ be a symmetric $m \times m$ matrix with nonzero diagonal entries. Let the sparsity pattern of each row of $\boldsymbol{H}$ contain the sparsity pattern of at least one row of $\boldsymbol{B}^{\mathrm{T}}$. If either the Markowitz strategy or the Markowitz-*1 *strategy is applied to the matrix with sparsity pattern*

$$\begin{bmatrix} \boldsymbol{H} & \boldsymbol{B} \\ \boldsymbol{B}^{\mathrm{T}} & \boldsymbol{0} \end{bmatrix}, \tag{5.11}$$

then in the analysis phase n tile pivots are chosen.

Proof. Let the nondefective variable with the smallest row count be k. Assuming the kth row of $\boldsymbol{B}$ has at least one nonzero entry, the number of entries in the kth row of $\boldsymbol{H}$ does not exceed $r_k - 1$. Since row k of $\boldsymbol{H}$ contains the sparsity pattern of row i, say, of $\boldsymbol{B}^{\mathrm{T}}$, row i of $\boldsymbol{B}^{\mathrm{T}}$ has row count r_{i+m} satisfying $r_{i+m} \leqslant r_k - 1$, and if $(\boldsymbol{B}^{\mathrm{T}})_{ij}$ is nonzero, $(\boldsymbol{H})_{kj}$ is nonzero. We deduce from (3.11) that, if the Markowitz strategy is used, the *tile* pivot $(k, i+m)$ has cost

$$(r_{i+m}-1)(r_k-2). \tag{5.12}$$

If the Markowitz-1 strategy is used, the *tile* pivot $(k, i+m)$ has cost (see (3.10))

$$(r_{i+m}-1)(r_k-1). \tag{5.13}$$

Comparing (3.9) (with $i=k$) with (5.12) and (5.13), we deduce that the *tile* pivot $(k, i+m)$ has a smaller Markowitz cost than the best 1 × 1 pivot, and a *tile* pivot is therefore chosen as the first pivot.

A *tile* pivot preserves the zero block and the reduced matrix after the first stage of the elimination process is of the form

$$\begin{bmatrix} \boldsymbol{H}_1 & \boldsymbol{B}_1 \\ \boldsymbol{B}_1^{\mathrm{T}} & \boldsymbol{0} \end{bmatrix}, \tag{5.14}$$

where $\boldsymbol{H}_1$ has nonzero diagonal entries and each row of $\boldsymbol{H}_1$ contains the sparsity pattern of at least one row of $\boldsymbol{B}_1^{\mathrm{T}}$. By induction we continue to choose *tile* pivots until all the defective variables have been selected. A 1 × 1 pivot can only be chosen before n *tile* pivots have been selected if, at some stage l of the elimination process, the reduced matrix $\boldsymbol{B}_l$ has a row, say row r, of all zeros. The reduced matrix $\boldsymbol{H}_l$ has nonzero diagonal entries, therefore variable r is nondefective and may be used as a 1 × 1 pivot. Since row r has no entries in $\boldsymbol{B}_l$, this pivot preserves the zero block. □

COROLLARY 5.2 *Let $\boldsymbol{B}$ be an $m \times n$ matrix of rank n with no null rows. If the $m \times m$ matrix $\boldsymbol{H}$ has the same sparsity pattern as $\boldsymbol{B}\boldsymbol{B}^{\mathrm{T}}$ and the Markowitz or Markowitz-*1 *strategy is applied to the matrix* (5.11), *then in the analysis phase n tile pivots are chosen.*

We observe that if $m = n$, Theorem 5.2 may be deduced from Theorem 5.1. For if $m = n$, there exists a permutation $\boldsymbol{P}$ such that

$$\boldsymbol{T} = \boldsymbol{P}^{\mathrm{T}} \begin{bmatrix} \boldsymbol{H} & \boldsymbol{B} \\ \boldsymbol{B}^{\mathrm{T}} & \mathbf{0} \end{bmatrix} \boldsymbol{P}, \tag{5.15}$$

is a *tiled* matrix. The permutation $\boldsymbol{P}$ is found by permuting the rows of $\boldsymbol{B}^{\mathrm{T}}$ to place nonzeros on the diagonal, and then pairing the rows of $\boldsymbol{H}$ with the rows of the permuted matrix $\boldsymbol{B}^{\mathrm{T}}$. Since h_{ii} is nonzero for all i, $1 \leqslant i \leqslant n$, $\boldsymbol{T}$ has nonzero diagonal tiles.

THEOREM 5.3 *For nonsingular matrices of the form*

$$\begin{bmatrix} \boldsymbol{I} & & \boldsymbol{B}_1 \\ & \mathbf{0} & \boldsymbol{B}_2 \\ \boldsymbol{B}_1^{\mathrm{T}} & \boldsymbol{B}_2^{\mathrm{T}} & \mathbf{0} \end{bmatrix}, \tag{5.16}$$

where $\boldsymbol{B}_1$ is an $(m-k) \times n$ matrix $(1 \leqslant k \leqslant n)$, $\boldsymbol{B}_2$ is a $k \times n$ matrix with $\boldsymbol{B}_2 = [\boldsymbol{D}_k \ \ \mathbf{0}]$, and $\boldsymbol{D}_k$ is a $k \times k$ diagonal matrix with nonzero diagonal entries (see Section 6), *the Markowitz, Markowitz*-b, *and Markowitz*-1 *analysis strategies choose exactly k oxo pivots, all with Markowitz cost zero.*

Proof. The defective variables $m-k+l$, $1 \leqslant l \leqslant k$, each have row count 1. For each l, $1 \leqslant l \leqslant k$, the 2×2 pivot $(m-k+l,\ m+l)$ is an *oxo* pivot with Markowitz cost zero. If row i of $\boldsymbol{B}_1$ has no nonzero entries, variable i is a 1×1 pivot with Markowitz cost zero. A pivot with Markowitz cost zero preserves the sparsity structure, and we therefore eventually choose k *oxo* pivots associated with the defective variables $m-k+1, m-k+2, \ldots, m$. □

6. Numerical experiments

In this section we describe the results of testing the proposed modifications to MA27 on a variety of problems. All the results were obtained using our experimental code. The data for the problems was taken from the extensive set of linear programming test problems available from Netlib (Dongarra & Grosse, 1987) as collected by Gay (1985). All computation was performed on the IBM 3084Q at Harwell.

Each problem was constructed to be of the form

$$\begin{bmatrix} \boldsymbol{H} & \boldsymbol{B} \\ \boldsymbol{B}^{\mathrm{T}} & \mathbf{0} \end{bmatrix}, \tag{6.1}$$

where $\boldsymbol{B}$ is an $m \times n$ matrix $(m \geqslant n)$ that is a permutation of a Netlib matrix. With the given Netlib data, we scaled the rows and columns of the matrix (6.1) using the scheme of Curtis & Reid (1972), implemented as subroutine MC19 in the Harwell Subroutine Library. The results presented in the tables of this section are representative of all the problems we tried.

We report in detail the results for four classes of problems:

(i) $\boldsymbol{H} = \boldsymbol{I}_m$.

Problems of this type arise from the solution of unconstrained least-squares problems ($\boldsymbol{C}$ null in equation (1.3)).

(ii)
$$\begin{bmatrix} \boldsymbol{I} & & \boldsymbol{B}_1 \\ & \mathbf{0} & \boldsymbol{B}_2 \\ \boldsymbol{B}_1^{\mathrm{T}} & \mathbf{B}_2^{\mathrm{T}} & \mathbf{0} \end{bmatrix}, \tag{6.2}$$

where $\boldsymbol{B}_2$ is an $n \times n$ submatrix of $\boldsymbol{B}$ of full rank. Our motivation for considering problems of this type is equality constrained least-squares problems (see Section 1) and Karmarkar's interior point linear programming algorithm (Karmarkar 1984). The matrix (6.2) is a limiting form of those which occur during the implementation of Karmarkar's algorithm.

(iii) As (ii) with $\boldsymbol{B}_2$ a $k \times n$ ($k \leq n$) submatrix of $\boldsymbol{B}$ such that

$$\boldsymbol{B}_2 = [\boldsymbol{D}_k \quad \mathbf{0}], \tag{6.3}$$

where $\boldsymbol{D}_k$ is a $k \times k$ diagonal matrix. Matrices of this type arise in constrained least-distance problems for which there is a mixture of equality and inequality constraints.

(iv) $\boldsymbol{H}$ has the sparsity pattern of $\boldsymbol{B}\boldsymbol{B}^{\mathrm{T}}$, and the values of the nonzero entries in the matrix (6.1) are obtained using subroutine FA01 from the Harwell Subroutine Library, which generates uniformly distributed pseudo-random numbers in the range $[-1, 1]$. Problems of this structure were proposed to us by Gill, Murray, & Saunders (1989).

In addition to the above classes of test problems, we have tested our proposed modifications to MA27 on a small number of *tiled* matrices. For $2n \times 2n$ *tiled* matrices we found that the Markowitz and Markowitz-1 strategies chose n *tile* pivots (as predicted by Theorem 5.1), and although the Markowitz-b strategy did not always choose all the pivots to be *tiles*, for the *tiled* matrices we considered, the predicted cost of the factorization phase using the Markowitz-b strategy did not exceed that of the Markowitz strategy by more than 12 per cent. The Markowitz-b strategy gave results which compared very favourably with those obtained using the original MA27 strategy and the minimum degree strategy.

In Table 6.1 a comparison of the different strategies we have considered for the analysis phase of MA27 is presented. We used the default value 0.1 for the relative threshold parameter u. The predicted number of flops is the number of flops the analysis phase predicts will be required by the factorization phase, and the actual number of flops is the number of flops actually used by the factorization phase. For the numerical strategy (Section 4.3), the predicted number of flops is equivalent to the actual number of flops. Results for FFFFF800 are not included in class (iv) since this problem was prohibitively large for our experimental code. It may be seen that for problems belonging to classes (ii) and (iii), the Markowitz strategies are far more successful than the original and minimum degree strategies. Furthermore, in the factorization phase, little gain is lost through pivoting for stability and the results for the Markowitz and Markowitz-b strategies differ from those of the numerical strategy by less than 12 per cent. For the Karmarkar problems (class (ii)) with data Capri, FFFFF800,

TABLE 6.1

A comparison of the proposed strategies for the analysis phase of MA27

Problem		Capri	Share1b	FFFFF800	E226	Beaconfd
No. rows in $B(m)$		466	253	1028	472	295
No. cols in $B(n)$		271	117	524	223	173
Least-squares problems (i)						
No. nonzeros		2330	1432	7429	3240	3703
Predicted flops	Original	136381	34658	1433091	196208	158850
	Min. degree	148720	26040	962303	173032	207900
	Markowitz	87022	33244	601059	189033	95623
	Markowitz-b	87003	33244	600799	189033	95623
	Markowitz-1	99745	33264	1160955	203775	95597
Actual flops	Original	199130	36540	2483074	226655	365065
	Min. degree	178704	26216	1423918	202355	228090
	Markowitz	108530	33804	884798	221700	145822
	Markowitz-b	108522	33804	884393	221700	145822
	Markowitz-1	127181	38278	1861678	237054	145822
	Numerical	97807	33599	848905	197407	128390
Karmarkar problems (ii)						
No. nonzeros		2059	1315	6905	3017	3530
Predicted flops	Original	136381	34658	1433091	196208	158850
	Min. degree	174608	39127	766100	183255	119741
	Markowitz	2330	3644	7429	3240	3703
	Markowitz-b	2330	3718	7429	3240	3703
	Markowitz-1	2330	3700	7429	3240	3703
Actual flops	Original	462532	95851	5530512	633861	383227
	Min. degree	167441	41423	1351778	154177	106786
	Markowitz	2330	5764	7429	3240	3703
	Markowitz-b	2330	6476	7429	3240	3703
	Markowitz-1	2330	8375	7429	3240	3703
	Numerical	2330	5781	7429	3240	3703
Least-distance problems (iii)						
No. nonzeros		2190	1387	7247	3050	3633
Predicted flops	Original	136381	34658	1433091	196208	158850
	Min. degree	156035	22439	711888	129388	132468
	Markowitz	35355	9632	139289	11557	31990
	Markowitz-b	35355	9632	139376	11557	31990
	Markowitz-1	35245	10315	144171	12053	31990
Actual flops	Original	246713	48862	2849355	440361	387451
	Min. degree	167038	22413	740922	99113	121234
	Markowitz	37844	10556	164501	11582	43077
	Markowitz-b	37844	10556	164579	11582	43077
	Markowitz-1	37776	11271	175644	12181	43077
	Numerical	34269	10025	160440	11547	39310
Problems (iv)						
No. nonzeros		7201	4788	82920	17839	19096
Predicted flops	Original	309195	116728		1509463	1812501
	Min. degree	371352	128324		1546434	2396527
	Markowitz	199205	113939		1404884	482549
	Markowitz-b	200041	114986		1410030	482711
	Markowitz-1	198010	113919		1405420	482549
Actual flops	Original	825582	172504		1894308	3228859
	Min. degree	722132	194411		1786313	2487538
	Markowitz	401356	149491		2025084	757254
	Markowitz-b	426404	146888		1918331	740454
	Markowitz-1	429570	152483		2026426	757254
	Numerical	341710	146158		1667520	650552

E226, and Beaconfd the Markowitz strategies chose optimal pivot sequences, that is, pivot sequences in which the expected Markowitz cost of each pivot is zero. Since a pivot with zero expected Markowitz cost is not tested for stability in the new factorization phase, the actual number of flops was equal to the predicted number of flops for these problems. The comparison between the different strategies is less clear cut for classes (i) and (iv), but for class (i) good gains using the Markowitz and Markowitz-b strategies are observed for Capri, FFFFF800, and Beaconfd, and for class (iv) there are substantial gains for Capri and Beaconfd. The use of a numerical analysis phase gives improvements of less than 20 per cent compared with the Markowitz strategy. For all the problems considered, except E226 in class (iv), the Markowitz and Markowitz-b strategies give improvements over the original strategy.

In a recent paper, George & Liu (1989) discuss the effects of tie-breaking on the minimum degree ordering algorithm. For a 180 × 180 grid problem, George & Liu report that when MA27 was employed with ten different random initial orderings there was a difference of approximately 30 per cent in terms of the factorization operation counts between the best and worst orderings. We have performed some experiments on our test problems by randomly permuting the rows and columns of the test matrices before passing them to our experimental codes for the analysis and factorization phases. For the problems tried, we found that the predicted and actual flop counts were not very sensitive to the order in which the data was presented. Typically, the flop counts varied by less than 10 per cent. For the Karmarkar problems with Capri, FFFFF800, E226, and Beaconfd data, the optimality of the pivot sequences chosen using the Markowitz strategies was not dependent on the order of the input data.

The actual flops for the original strategy are those of the present factorization phase of MA27. If the expected Markowitz costs of the pivots chosen using the original strategy are passed to the new factorization phase, some improvement occurs because of the new stability test (4.5), the Markowitz cost test, and the new storage scheme for the generated element matrices, but the original strategy still compares unfavourably with those of Sections 3.1 and 3.2.

For the problems in Table 6.1 the matrix $\boldsymbol{B}$ in (6.1) is of order $m \times n$ with $m \gg n$. For class (i) and (iv) problems if $m \gg n$ and the tentative pivot sequence is selected using the Markowitz, Markowitz-1, or Markowitz-b strategy, the sequence will in general contain few *tile* pivots. We anticipate that these strategies will give a more significant improvement over the minimum degree strategy for these problems when $\boldsymbol{B}$ is almost square since the zero block is then proportionally larger. Moreover, $\boldsymbol{B}^{\mathrm{T}}$ is more likely to contain rows with a single nonzero, which may be used to form *tile* pivots with Markowitz cost zero. We have therefore constructed almost square matrices $\boldsymbol{B}$ by deleting rows of $\boldsymbol{B}$ to leave a full rank submatrix of order $n^* \times n$ with $n^* = n$. The results of applying the different strategies to class (i) and (iv) problems with $\boldsymbol{B}$ almost square are given in Table 6.2.

Tables 6.1 and 6.2 suggest that, of the Markowitz strategies discussed in Section 3.2, selecting the pivots on the basis of the Markowitz cost of the second pivot in a 2 × 2 pivot gives the most efficient factorization. However, for most

TABLE 6.2

A comparison of the proposed strategies for the analysis phase of MA27 *applied to 'almost' square problems* ($u = 0.1$)

Problem		Capri	Share1b	FFFFF800	E226	Beaconfd
No. cols (n)		271	117	524	223	173
Least-squares problems (I)						
No. rows (n^*)		280	118	567	225	181
No. nonzeros		1419	579	2811	989	1698
Predicted flops	Original	40763	4496	79144	1163	123044
	Min. degree	49071	5560	101301	12847	187239
	Markowitz	1777	1685	17674	1248	1895
	Markowitz-b	1777	1715	17742	1248	1895
	Markowitz-1	1777	1682	17226	1248	1895
Actual flops	Original	126036	9538	279874	22035	196302
	Min. degree	77061	5502	130160	16162	205364
	Markowitz	1777	1814	25952	1248	1895
	Markowitz-b	1777	1796	26100	1248	1895
	Markowitz-1	1777	1887	28672	1248	1895
	Numerical	1777	2411	23571	1248	1895
Problems (iv)						
No. rows (n^*)		280	127	542	233	174
No. nonzeros		3291	1164	9682	2091	5360
Predicted flops	Original	85980	9642	804595	26639	211157
	Min. degree	122154	12172	1024847	35243	325421
	Markowitz	3802	6551	29295	2741	5539
	Markowitz-b	3802	7264	29462	2745	5539
	Markowitz-1	3802	6592	29481	2741	5539
Actual flops	Original	599210	23288	6396947	154712	703723
	Min. degree	194606	18119	1876333	44612	147801
	Markowitz	3802	8840	84764	2823	5539
	Markowitz-b	3802	9586	100480	2785	5539
	Markowitz-1	3802	9229	100743	2823	5539
	Numerical	3802	8297	101250	2811	5539

problems using the upper bound on the Markowitz cost of the second pivot yields results which are comparable with those where the exact Markowitz cost is computed and, since the exact Markowitz cost is expensive to compute (involving a merge of the two pivotal rows), we will adopt the Markowitz-b strategy which uses this upper bound.

The choice of the relative pivot threshold u and the scaling of the rows and columns of the matrix can have a profound effect on the measure of agreement between the predicted and actual numbers of floating-point operations. A small value of u such as 0.01 is sometimes used despite the attendant risk of numerical instability (Gill, Murray, & Saunders, 1989). This has motivated us to consider using smaller values for u, in particular, $u = 0.01$. In Table 6.3(a) we report the results of experiments with the original MA27 strategy using $u = 0.01$. In addition, we consider taking the pattern of the matrix $\boldsymbol{B}$ and generating the values of the nonzero entries using subroutine FA01 from the Harwell Subroutine Library. We refer to the resulting problems as the *random* test problems. In Table 6.3(b) we report the results corresponding to those in Table 6.3(a) using

TABLE 6.3
(a) Results using the original MA27 *strategy. (b) Results using the Markowitz*-b *strategy*

(a)

Problem		Capri	Share1b	E226	Beaconfd
No. rows (m)		466	253	472	295
No. cols (n)		271	117	223	173
Least-squares problems (i)					
No. nonzeros		2330	1432	3240	3703
Predicted flops		136381	34658	196208	158850
Actual flops	$u = 0.1$, scaled B	199130	36540	226655	365065
	$u = 0.01$, scaled B	167534	36379	197572	211638
	$u = 0.1$, random B	211216	36258	201835	205907
Karmarkar problems (ii)					
No. nonzeros		2059	1315	3017	3530
Predicted flops		136381	34658	196208	158850
Actual flops	$u = 0.1$, scaled B	462532	95851	633861	383227
	$u = 0.01$, scaled B	270547	75919	349879	202237
	$u = 0.1$, random B	464378	109948	482372	354140
Least-distance problems (iii)					
No. nonzeros		2190	1387	3050	3633
Predicted flops		136381	34658	196208	158850
Actual flops	$u = 0.1$, scaled B	246713	48862	440361	387451
	$u = 0.01$, scaled B	198375	43579	265825	221094
	$u = 0.1$, random B	227484	44380	320398	231228

(b)

Problem		Capri	Share1b	E226	Beaconfd
No. rows (m)		466	253	472	295
No. cols (n)		271	117	223	173
Least-squares problems (i)					
No. nonzeros		2330	1432	3240	3703
Predicted flops		87003	33244	189033	95623
Actual flops	$u = 0.1$, scaled B	108530	33804	221700	145822
	$u = 0.01$, scaled B	88351	33699	189167	98893
	$u = 0.1$, random B	114840	33895	208526	107284
Karmarkar problems (ii)					
No. nonzeros		2059	1315	3017	3530
Predicted flops		2330	3718	3240	3703
Actual flops	$u = 0.1$, scaled B	2330	6476	3240	3703
	$u = 0.01$, scaled B	2330	3714	3240	3703
	$u = 0.1$, random B	2330	4105	3240	3703
Least-distance problems (iii)					
No. nonzeros		2190	1387	3050	3633
Predicted flops		35355	9632	11557	31990
Actual flops	$u = 0.1$, scaled B	37844	10556	11582	43077
	$u = 0.01$, scaled B	35355	9632	11581	32680
	$u = 0.1$, random B	41606	9784	11647	33630

the Markowitz-b strategy to select the tentative pivot sequence and employing the proposed modifications to the factorization phase. The results for FFFFF800 are omitted from these tables (and from subsequent tables) since this problem was significantly larger and was too expensive to run using our experimental code. Additionally, from the experiments we did run on this problem, it appeared to provide no new information. We also omit results for class (iv) since these problems were also expensive to run using our experimental code, and they appeared to behave in a similar manner to the class (i) problems (see Tables 6.1 and 6.2).

Tables 6.3(a) and (b) demonstrate that it is difficult to advise the user on how accurate the predicted cost returned by the symbolic analysis phase is likely to be. The reliability of the predicted cost depends upon the number of zero diagonal entries in the matrix, the values of the nonzero entries, and the value of the relative pivot threshold u. The results using the original MA27 strategy confirm the findings of Gill, Murray, & Saunders (1989) that for problems with some zero diagonal entries a smaller value of u can yield a substantial reduction in the factorization cost. If the problem is well scaled, the Markowitz-b strategy which takes into consideration the zeros on the diagonal is generally less sensitive to the value of u than the original strategy. This is particularly true if the tentative pivot sequence contains a large number of 2×2 pivots with Markowitz cost zero. However, for the Karmarkar problem with Share1b data, we found that the number of arithmetic operations needed by the factorization phase was more than 1.7 times the number anticipated by the analysis phase. For the Markowitz-b strategy with $u=0.01$ we found that for all our test problems the predicted cost was a good estimate of the actual cost. If the problem is not scaled, the discrepancy between the predicted and actual cost can be considerable. For example, for the least-squares problem Beaconfd with $u=0.1$, if MC19 was not employed we found that, for the Markowitz-b strategy, the actual number of flops rose from 145822 to 298057.

The effect of using the new stability test (4.5) is demonstrated in Table 6.4. The Markowitz-b strategy is used to select the tentative pivot sequence and the default value $u=0.1$ is employed. It may be seen that for some problems the new stability test can give a worthwhile gain, and for all the test problems the new stability test gave results which were at least as good as those achieved with the old stability test. For problems Capri, E226, and Beaconfd in class (ii) where the pivotal sequence is optimal, the stability test is not applied since all the pivots have Markowitz cost zero.

In Table 6.5 we illustrate the effects of testing the Markowitz cost of a potential pivot. The recommended strategy passes the expected Markowitz costs to the factorization phase; the Markowitz cost of a potential pivot is then tested against the expected cost, as discussed in Section 4.2. In addition, any pivot with Markowitz cost zero is not tested for stability. This strategy is compared with not passing the expected Markowitz costs to the factorization phase, but still testing for a Markowitz cost of zero, and with performing no test on the Markowitz cost of potential pivots during the factorization phase. It may be seen that for most of the test problems it is important to pass the expected Markowitz costs to the

TABLE 6.4

*A comparison of the new and old stability tests (Markowitz-*b *pivotal strategy with u* = 0.1 *is used)*

Problem		Capri	Share1b	E226	Beaconfd
No. rows (m)		466	253	472	295
No. cols (n)		271	117	223	173
Least-squares problems (i)					
No. nonzeros		2330	1432	3240	3703
Predicted flops		87003	33244	189033	95623
Actual flops	New stability test	108522	33804	221700	145822
	Old stability test	109868	33804	222058	158405
Karmarkar problems (ii)					
No. nonzeros		2059	1315	3017	3530
Predicted flops		2330	3718	3240	3703
Actual flops	New stability test	2330	6476	3240	3703
	Old stability test	2330	10041	3240	3703
Least-distance problems (iii)					
No. nonzeros		2190	1387	3050	3633
Predicted flops		35355	9632	11557	31990
Actual flops	New stability test	37844	10556	11582	43077
	Old stability test	37844	10556	11582	44848

TABLE 6.5

*The effect of testing the Markowitz cost of potential pivots (Markowitz-*b *pivotal strategy with u* = 0.1 *is used)*

Problem		Capri	Share1b	E226	Beaconfd
No. rows (m)		466	253	472	295
No. cols (n)		271	117	223	173
Least squares problems (i)					
No. nonzeros		2330	1432	3240	3703
Predicted flops		87003	33244	189033	95623
Actual flops	Recommended strategy	108522	33804	221700	145822
	Test for zero Markowitz cost	162714	33804	240458	202944
	No Markowitz cost test	162714	33804	241253	317657
Karmarkar problems (ii)					
No. nonzeros		2059	1315	3017	3530
Predicted flops		2330	3718	3240	3703
Actual flops	Recommended strategy	2330	6476	3240	3703
	Test for zero Markowitz cost	2330	29580	3240	3703
	No Markowitz cost test	374084	47179	2167494	134028
Least-distance problems (iii)					
No. nonzeros		2190	1387	3050	3633
Predicted flops		35355	9632	11557	31990
Actual flops	Recommended strategy	37844	10556	11582	43077
	Test for zero Markowitz cost	37844	10556	11582	57296
	No Markowitz cost test	38007	10556	574364	1137893

TABLE 6.6
*A comparison of the original and new storage schemes for the generated element matrices (Markowitz-*b *pivotal strategy and* $u = 0.1$)

Problem		Capri	Sharelb	E226	Beaconfd
No. rows (m)		466	253	472	295
No. cols (n)		271	117	223	173
Least-squares problems (i)					
No. nonzeros		2330	1432	3240	3703
New storage scheme	Predicted flops	87003	33244	189033	95623
	Actual flops	108522	33804	221700	145822
Original storage scheme	Predicted flops	204995	33594	199209	91297
	Actual flops	239138	34154	225426	125072
Karmarkar problems (ii)					
No. nonzeros		2059	1315	3017	3530
New storage scheme	Predicted flops	2330	3718	3240	3703
	Actual flops	2330	6476	3240	3703
Original storage scheme	Predicted flops	481044	42706	307035	635379
	Actual flops	896758	52842	444555	555759
Least-distance problems (iii)					
No. nonzeros		2190	1387	3050	3633
New storage scheme	Predicted flops	35355	9632	11557	31990
	Actual flops	37844	10556	11582	43077
Original storage scheme	Predicted flops	968217	43544	775318	606296
	Actual flops	1204350	47754	785233	659184

factorization phase and to test for a Markowitz cost of zero. For the class (ii) problems with Capri, E226, and Beaconfd data for which the Markowitz-b strategy chooses optimal pivot sequences, the number of flops required by the numerical factorization increases dramatically if no test is made for a Markowitz cost of zero.

All the results for the Markowitz strategies presented in Tables 6.1–6.5 have employed the new storage scheme for the generated element matrices discussed in Section 3.4. In Table 6.6 we consider what happens if the original storage scheme in which the generated element matrices are stored as full matrices is used with the Markowitz-b strategy. It may be seen that for many of the test problems, particularly those in classes (ii) and (iii), if the original storage scheme is used there is a sharp deterioration in the results.

7. Conclusions

Our numerical experiments indicate that, in general, the proposed modifications to MA27 yield worthwhile gains compared with the MA27 strategy when applied to matrices with some zero diagonal entries. Extending the minimum degree ordering to include 2 × 2 pivots does give some improvement over the original MA27 strategy but larger gains are obtained if the pivots are selected using the Markowitz criterion. The gains were found to be particularly significant for the Karmarkar and least-distance test problems (classes (ii) and (iii)). For the unconstrained least-squares problems (class (i)) and the class (iv) problems the improvements in the performance using a Markowitz strategy in place of a minimum degree strategy were considerable if the matrix B was almost square. Of the Markowitz strategies discussed in Section 3.2, our numerical experiments suggest that selecting the pivot sequence on the basis of the Markowitz cost of the second pivot in a 2 × 2 pivot gives the best results. However, for most problems using an upper bound on the Markowitz cost of the second pivot yields results which are comparable with those where the exact Markowitz cost is computed and, since the exact Markowitz cost is expensive to compute, we will adopt the Markowitz-b strategy which uses this upper bound.

We propose therefore to modify the analysis phase of MA27 to use generated element matrices of the form (3.16) and (3.17), to employ the Markowitz-b pivoting strategy (Section 3.2), to provide the factorization phase with the expected Markowitz costs of the pivots, and to indicate which pivots are intended 2 × 2 pivots. The factorization phase will be modified to accept the new data, use generated element matrices of the form (3.16) and (3.17), use the new stability test (Section 4.1), and delay pivots whose Markowitz costs are much larger than anticipated. We believe that these changes will avoid the present poor performance of MA27 on problems with zero diagonal entries.

Acknowledgements

We would like to express our thanks to Philip Gill, Michael Saunders, and Walter Murray for drawing our attention to this problem and for the time they spent explaining their experiences.

Most of the work for this paper was conducted while the authors were at the Harwell Laboratory, Oxfordshire.

REFERENCES

BUNCH, J. R., & PARLETT, B. N. 1971 Direct methods for solving symmetric indefinite systems of linear equations. *SIAM J. Numer. Anal.* **8,** 639–655.

CURTIS, A. R., & REID, J. K. 1972 On the automatic scaling of matrices for Gaussian elimination. *Journal of the Institute of Mathematics and its Applications* **10,** 118–124.

DONGARRA, J. J., & GROSSE, E. 1987 Distribution of mathematical software via electronic mail. *Communications ACM* **30,** 403–407.

DUFF, I. S., & REID, J. K. 1983 The multifrontal solution of indefinite sparse symmetric linear equations. *ACM Trans. Math. Softw.* **9,** 302–325.

GAY, D. M. 1985 Electronic mail distribution of linear programming test problems. *Mathematical Programming Society COAL Newsletter.*

GEORGE, A., & LIU, J. W. H. 1989 The evolution of the minimum degree ordering algorithm. *SIAM Review* **31,** 1–19.

GILL, P. E., MURRAY, W., & SAUNDERS, M. A. 1989 Private communications.

GILL, P. E., MURRAY, W., & WRIGHT, M. H. 1981 *Practical Optimization.* Academic Press.

GILL, P. E., MURRAY, W., SAUNDERS, M. A., & WRIGHT, M. H. 1990 A Schur-complement method for sparse quadratic programming. In: *Reliable Numerical Computation* (M. G. Cox and S. J. Hammarling, eds). Oxford University Press.

KARMARKAR, N. 1984 A new polynomial time algorithm for linear programming. *Combinatoria* **4,** 373–395.

MARKOWITZ, H. M. 1957 The elimination form of the inverse and its application to linear programming. *Management Sci.* **3,** 255–269.