The antitriangular factorisation of saddle point matrices

J. Pestana and A. J. Wathen

August 29, 2013

Abstract

Mastronardi and Van Dooren [this journal, 34 (2013) pp. 173–196] recently introduced the block antitriangular (“Batman”) decomposition for symmetric indefinite matrices. Here we show the simplification of this factorisation for saddle point matrices and demonstrate how it represents the common nullspace method. We show the relation of this factorisation to constraint preconditioning and how it transforms but preserves the block diagonal structure of block diagonal preconditioning.

1 Introduction

The antitriangular factorisation of Mastronardi and Van Dooren [9] converts a symmetric indefinite matrix $H \in \mathbb{R}^{p \times p}$ into a block antitriangular matrix $M$ using orthogonal similarity transforms. The factorisation can be performed in a backward stable manner and linear systems with the block antitriangular matrix may be efficiently solved. Moreover, the orthogonal similarity transforms preserve eigenvalues, and preserve and reveal the inertia of $H$. Thus, from $M$ one can determine the triple $(n_-, n_0, n_+)$ of $H$, where $n_-$ is the number of negative eigenvalues, $n_0$ is the number of zero eigenvalues and $n_+$ is the number of positive eigenvalues.

The antitriangular factorisation takes the form

$$H = QMQ^T, \quad Q^{-1} = Q^T, \quad M = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & Y^T \\ 0 & 0 & X & Z^T \\ 0 & Y & Z & W \end{bmatrix} \begin{bmatrix} n_0 \\ n_1 \\ n_2 \\ n_1 \end{bmatrix}, \quad (1)$$

where $n_1 = \min(n_-, n_+)$, $n_2 = \max(n_-, n_+) - n_1$, $Z \in \mathbb{R}^{n_1 \times n_2}$, $W \in \mathbb{R}^{n_1 \times n_1}$ is symmetric, $X = \epsilon LL^T \in \mathbb{R}^{n_2 \times n_2}$ is symmetric definite whenever $n_2 > 0$ and $Y \in \mathbb{R}^{n_1 \times n_1}$ is nonsingular and antitriangular, so that entries above the main antidiagonal are zero. Additionally,

$$\epsilon = \begin{cases} 1 & \text{if } n_+ > n_- \\ -1 & \text{if } n_- > n_+ \end{cases}$$
The matrix $M$ is strictly antitriangular whenever $n_2 = 0,1$, i.e., whenever the number of positive and negative eigenvalues differs by at most one. However, the “bulge” $X$ increases in dimension as $H$ becomes closer to definite. In the extreme case that $H$ is symmetric positive (or negative) definite $n_0 = n_1 = 0$, i.e., $X$ is itself a $p \times p$ matrix. Accordingly, the antitriangular factorisation is perhaps best suited to matrices that have a significant number of both positive and negative eigenvalues. We emphasise, however, its generality for real symmetric matrices.

Saddle point matrices are symmetric and indefinite, so that the antitriangular factorisation can be applied. These matrices arise in numerous applications [1, Section 2] and have the form

$$A = \begin{bmatrix} A & B^T \\ B & 0 \end{bmatrix} \begin{bmatrix} n \\ m \end{bmatrix},$$

where $A \in \mathbb{R}^{n \times n}$ is symmetric and $B \in \mathbb{R}^{m \times n}$. The matrix $A$ is nonsingular with $n$ positive eigenvalues and $m$ negative eigenvalues when $A$ is positive definite on the nullspace of $B$ and $B$ has full rank. We only consider this most common situation here.

The algorithm for computing an antitriangular factorisation proposed by Mastronardi and Van Dooren is designed to be applicable to all symmetric indefinite matrices and so is unduly complicated for the special case of saddle point matrices. In this note we describe a simple procedure for obtaining an antitriangular factorisation of saddle point matrices that involves only permutation matrices and a QR factorisation of $B^T$. This QR factorisation of $B$ allows us to show that solving a saddle point system with the antitriangular factorisation is equivalent to applying the nullspace method [1, Section 6][13, Section 15.2]. In other words, the factorisation proposed by Mastronardi and Van Dooren allows the nullspace method to be represented not just as a procedure but also as a matrix decomposition, similarly to other well known methods for solving linear systems like Gaussian elimination.

If the matrix $A$ is large, we may solve the saddle point system by an iterative method rather than by a direct method like the antitriangular factorisation. Preconditioning is usually required, with block preconditioners, such as block diagonal and constraint preconditioners popular choices for saddle point systems. We show that the same orthogonal transformation matrix that converts $A$ into an antitriangular matrix can be applied to these preconditioners and that relevant structures are preserved.

Throughout, we use Matlab notation to denote submatrices. Thus $K(q : r, s : t)$ is the submatrix of $K$ comprising the intersection of rows $q$ to $r$ with columns $s$ to $t$.

## 2 An antitriangular factorisation for saddle point matrices

We are interested in applying orthogonal transformations to the saddle point matrix $A$ in (2) to obtain the antitriangular matrix $M$ in (1). Since $A$ is
nonsingular with \( n \) positive eigenvalues and \( m \) negative eigenvalues, in this case the antitriangular matrix has the specific form

\[
\mathcal{M} = \begin{bmatrix}
0 & 0 & Y^T \\
0 & X & Z^T \\
Y & Z & W
\end{bmatrix}
\]

where \( Y \in \mathbb{R}^{m \times m} \) is antitriangular, \( X \in \mathbb{R}^{(n-m) \times (n-m)} \) is symmetric positive definite and \( W \in \mathbb{R}^{m \times m} \) is symmetric. Because of its nonzero profile, this factorisation is also called the “Batman” factorisation. We note that linear systems with \( \mathcal{M} \) can be solved with the obvious “antitriangular” substitution (finding the last variable from the first equation, the second-last variable from the second equation and so forth) twice, with a solve with the positive definite matrix \( X \) (using, for example, a Cholesky factorisation) in between.

The algorithm of Mastronardi and Van Dooren proceeds outwards from the \((1,1)\) entry of \( H \). Thus, at the \( k \)-th step we use the factorisation \( M^{(k)} = Q^{(k)}H^{(k)}(Q^{(k)})^T \) of \( H^{(k)} = H(1 : k, 1 : k) \) to obtain an antitriangular factorisation of \( H^{(k+1)} = H(1 : k + 1, 1 : k + 1) \), where \( Q^{(k)} \in \mathbb{R}^{k \times k} \) is orthogonal and

\[
M^{(k)} = \begin{bmatrix}
0 & 0 & Y^T \\
0 & X & Z^T \\
Y & Z & W
\end{bmatrix}.
\]

However, the inertia of \( H^{(k)} \) and \( H^{(k+1)} \) must differ and as a result a number of different cases must be examined at each step. In particular, updating \( X \) so that it remains positive or negative definite can be rather involved. An additional issue when applying this factorisation to saddle point matrices is that the algorithm in [9] ignores the zero \((2,2)\) block entirely.

We show here that for saddle point matrices a far simpler approach can be taken to achieve an antitriangular factorisation. As a first step, we use the matrix

\[
Q_1^T = \begin{bmatrix}
0 & I_m \\
I_n & 0
\end{bmatrix}
\]

to permute the blocks of \( A \) so that the zero matrix is now in the top left corner:

\[
Q_1^T A Q_1 = \begin{bmatrix}
0 & B \\
B^T & A
\end{bmatrix}
\]

Our next task is to make the matrix \( B^T \) antitriangular. To achieve this we compute the (full) QR decomposition of \( B^T \),

\[
B^T = \underbrace{\begin{bmatrix}
U_1 \\
U_2
\end{bmatrix}}_{U} \begin{bmatrix}
R \\
0
\end{bmatrix},
\]

where \( U \in \mathbb{R}^{n \times n} \) and \( R \in \mathbb{R}^{m \times m} \). Note that the columns of \( U_1 \) form an orthonormal basis for the range of \( B \) while the columns of \( U_2 \) form an orthonormal basis for its nullspace.
Next, we embed this orthogonal $U$ in a larger matrix

$$Q_2^T = \begin{bmatrix} I_m & 0 \\ 0 & U^T \end{bmatrix}$$

which we apply to our permuted saddle point matrix to give

$$Q_2^T Q_1^T A Q_2 = \begin{bmatrix} 0 & R^T & 0 \\ R & \hat{A}_{11} & \hat{A}_{12} \\ 0 & \hat{A}_{12}^T & \hat{A}_{22} \end{bmatrix} \begin{bmatrix} m \\ m \end{bmatrix}$$

where $\hat{A}_{ij} = U_i^T A U_j$, $i, j = 1, 2$. Note that since $A$ is positive definite on the nullspace of $B$, and the columns of $U_2$ form a basis for this nullspace, $\hat{A}_{22}$ is positive definite.

To obtain the desired antitriangular form, all that remains is to permute the last $n - m$ rows and columns so that the $R$ matrix is transformed to an antitriangular matrix that sits in the last $m$ rows. One matrix that accomplishes this is

$$Q_3^T = \begin{bmatrix} I_m & 0 & 0 \\ 0 & 0 & I_{n-m} \\ 0 & S_m & 0 \end{bmatrix},$$

where

$$S_m = \begin{bmatrix} 1 \\ \vdots \\ 1 \end{bmatrix}$$

is the reverse identity matrix, also known as the standard involuntary matrix or flip matrix, of dimension $m$. We drop the subscripts on the identity and reverse identity matrices when the dimension is clear from the context. Note that $S^{-1} = S^T = S$.

Application of $Q_3$ then gives the desired antitriangularisation:

$$M = Q^T A Q = \begin{bmatrix} 0 & 0 & Y^T \\ 0 & X & Z^T \\ Y & Z & W \end{bmatrix}, \quad Q = Q_1 Q_2 Q_3 = \begin{bmatrix} 0 & U_2 & U_1 S_m \\ I_m & 0 & 0 \end{bmatrix},$$

where $Y = SR \in \mathbb{R}^{m \times m}$, $X = \hat{A}_{22} \in \mathbb{R}^{(n-m) \times (n-m)}$, $Z = S \hat{A}_{12}$ and $W = S \hat{A}_{11} S$. The matrices $X$ and $W$ are symmetric and $X = \hat{A}_{22}$ is positive definite.

Thus, a block antitriangularisation of $A$ can be computed using only permutations and a QR decomposition of $B^T$. The method is clearly backward stable provided the QR decomposition is computed in a backward stable manner, say by Householder transformations or Givens rotations. In contrast, if we were to begin at the upper-left corner of $A$ and proceed outwards as is done in [9] for general symmetric indefinite matrices, we would need to separate out the positive definite part of $A$, i.e., its projection onto the nullspace of $B$, by updating a Cholesky factorisation at each stage. We would then require orthogonal
transformations to convert $B$ to upper triangular form and move the zero block to the upper left corner.

Another benefit of this simpler approach is that it provides an orthonormal basis of the nullspace of $B$. This allows us to link the antitriangular factorisation with the nullspace method and with preconditioning, as we show in subsequent sections.

### 3 Comparison with the nullspace method

In the previous section we saw that an antitriangular factorisation for saddle point matrices can be easily computed. In this section we show that using this factorisation to solve the linear system

$$
Ax = \begin{bmatrix} A & B^T \\ B & 0 \end{bmatrix} \begin{bmatrix} u \\ p \end{bmatrix} = \begin{bmatrix} f \\ g \end{bmatrix},
$$

is equivalent to applying the nullspace method.

The nullspace method requires a basis for the nullspace of $B$, such as $U_2$, and a particular solution $\hat{u}$ of $Bu = g$. (Note that any basis of the nullspace of $B$ in general would be sufficient but we use $U_2$ here as this common choice corresponds to the antitriangular factorisation.) With these two quantities, the nullspace method proceeds as follows [1, Section 6][13, Section 15.2]:

1. Solve $U_2^T A U_2 v = U_2^T (f - A \hat{u})$;
2. Set $u_* = U_2 v + \hat{u}$;
3. Solve $BB^T p_* = B(f - Au_*)$,

then $(u_*, p_*)$ solves (5).

On the other hand, applying the antitriangularisation (4) to (5) gives

$$
(Q^T AQ)(Q^T x) = \mathcal{M}(Q^T x) = (Q^T b)
$$

or

$$
\begin{bmatrix}
0 & 0 & Y^T \\
0 & X & Z^T \\
Y & Z & W
\end{bmatrix}
\begin{bmatrix}
p \\
U_2^T u \\
SU_1^T u
\end{bmatrix} = 
\begin{bmatrix}
g \\
U_2^T f \\
SU_1^T f
\end{bmatrix}.
$$

To recover $u$ and $p$ we must solve

$$
Y^T SU_1^T u = g \quad \text{(6a)}
\quad \quad
XU_2^T u + Z^T SU_1^T u = U_2^T f \quad \text{(6b)}
\quad \quad
Yp + ZU_2^T u + WSU_1^T u = SU_1^T f, \quad \text{(6c)}
$$

using the antitriangular substitution described in the previous section. This is equivalent to applying the inverse of $\mathcal{M}$, which has upper block antitriangular
structure, that is,

\[
M^{-1} = \begin{bmatrix}
Y^{-1}(ZX^{-1}Z^T - W)Y^T & -Y^{-1}ZX^{-1} & Y^{-1} \\
-X^{-1}Z^TY^T & X^{-1} & 0 \\
Y^{-T} & 0 & 0
\end{bmatrix}
\]_{m \times n-m}.

We now show that solving (6a)–(6c) is equivalent to applying the nullspace method. Substituting for \(Y = SR\) in (6a), we find that

\[
R^TS^T SU_1^Tu = R^TU_1^Tu = [R^T \ 0^T] \begin{bmatrix} U_1^T \\ U_2^T \end{bmatrix} u = Bu = g.
\]

Moreover,

\[
g = B(UU^T)u = B(U_1U_1^T + U_2U_2^T)u = BU_1U_1^Tu
\]
since the columns of \(U_2\) span the kernel of \(B\). Thus, a particular solution of \(Bu = g\) is \(\hat{u} = U_1U_1^Tu\).

Let us now examine (6b). Since \(X = \hat{A}_{22}\) and \(Z = S\hat{A}_{12}\), where \(\hat{A}_{ij} = U_i^TAU_j\), \(i, j = 1, 2\) as before, we have that

\[
XU_2^Tu + Z^TSU_1^Tu = U_2^Tf
\]

\[
(U_2^TAU_2)(U_2^Tu) + U_2^TAU_1S^TSU_1^Tu = U_2^Tf
\]

\[
(U_2^TAU_2)(U_2^Tu) = U_2^T(f - AU_1U_1^Tu).
\]

Thus, \(U_2^TAU_2v = U_2^T(f - \hat{A}\hat{u})\), where \(v = U_2^Tu\).

Substituting for \(\hat{u}, v\) and \(W = S\hat{A}_{11}S\) in (6c) then gives that

\[
Yp + ZU_2^Tu + WSU_1^Tu = SU_1^Tf
\]

\[
SRp + SU_1^TAU_2U_2^Tu + SU_1^TAU_1S^TSU_1^Tu = SU_1^Tf
\]

\[
Rp = U_1^T[f - A(U_2v + \hat{u})]
\]

\[
R^TRp = [R^T \ 0^T] \begin{bmatrix} U_1^T \\ U_2^T \end{bmatrix} [f - A(U_2v + \hat{u})]
\]

\[
BB^Tp = B(f - Au),
\]

where \(u_* = U_2v + \hat{u}\).

Thus, solving a system with the antitriangular factorisation is equivalent to applying the nullspace method with the QR nullspace basis and with \(\hat{u} = U_1U_1^Tu\) and \(v = U_2^Tu\). It is straightforward to show that

\[
u_* = U_2v + \hat{u} = (U_2U_2^T + U_1U_1^T)u = \begin{bmatrix} U_1 & U_2 \end{bmatrix} \begin{bmatrix} U_1^T \\ U_2^T \end{bmatrix} u = u
\]
as we would expect, while \(p\) coincides with the vector computed in step 3 in the nullspace method.

Note that no antitriangular solves are required in the nullspace method, even though we are solving a linear system with a block antitriangular matrix.
This is because the permutation matrix $S$ that transforms the upper triangular matrix $R$ to antitriangular form occurs as $S^2 = I$ in (6a) and can be eliminated from (6c).

We have seen that the factorisation of Mastronardi and Van Dooren allows us to view the nullspace method as a factorisation rather than as a procedure. It also puts the nullspace method into the same framework as other direct solvers, such as Gaussian elimination, which can be written as the product of structured matrices. This alternative viewpoint may be useful.

Application of the nullspace method is only viable when it is not too costly to solve linear systems with $U_2^T A U_2$. Otherwise, iterative methods may be more attractive. In the next section we show that constraint preconditioners for iterative methods applied to saddle point systems can also be related to the factorisation (4).

4 The antitriangular factorisation and preconditioning

When the saddle point system (5) is too large to be solved by a direct method like the antitriangular factorisation or the nullspace method, an iterative method such as a Krylov subspace method is usually applied. Unfortunately, however, these iterative methods typically converge slowly when applied to saddle point problems unless preconditioners are used. Many preconditioners for saddle point matrices have been proposed [1, Section 10][2], but we focus here on block preconditioners and show how they can be factored by the antitriangular factorisation in Section 2. We first discuss block diagonal preconditioners and then describe constraint preconditioners, showing that in this latter case the same orthogonal transformation converts $A$ and $P$ to antitriangular form. We assume throughout that $A$ in (5) is factorised in antitriangular form (4), i.e., that $A = Q M Q^T$.

We briefly mention the block diagonal matrix

$$P_D = \begin{bmatrix} T & 0 \\ 0 & V \end{bmatrix},$$

where $T \in \mathbb{R}^{n \times n}$ and $V \in \mathbb{R}^{m \times m}$ are symmetric. Often $T$ is chosen to approximate $A$ and $V$ to approximate the Schur complement $B A^{-1} B^T$. Indeed, if $T = A$ and $V = B A^{-1} B^T$ then $P_D^{-1} A$ has three eigenvalues, 1 and $(1 \pm \sqrt{5})/2$ [7, 10].

Applying $Q$ in a similarity transform gives

$$Q^T P_D Q = \begin{bmatrix} V & 0 & 0 \\ 0 & \hat{T}_{22} & \hat{T}_{12}^T S \\ 0 & S \hat{T}_{12} & S \hat{T}_{11} S \end{bmatrix},$$

where $\hat{T}_{ij} = U_i^T T U_j$, $i, j = 1, 2$. Thus, the transformed preconditioner is also block diagonal, with an $m \times m$ block followed by an $n \times n$ block. Note that
since $P_D$ is positive definite, $Q^T P_D Q$ can not have significant block antidiagonal structure.

Constraint preconditioners \([6, 8, 11, 12]\)

$$P_C = \begin{bmatrix} T & B^T \\ B & 0 \end{bmatrix},$$

on the other hand, preserve the constraints of $A$ exactly but replace $A$ by a symmetric approximation $T$. The preconditioned matrix $P^{-1}A$ has at least $2m$ unit eigenvalues, with the remainder being the eigenvalues $\lambda$ of $U_2^T A U_2 v = \lambda U_2^T T U_2 v$ \([6, 8]\) . (Note that we could use any basis for the nullspace of $B$ in place of $U_2$.) Since $A$ is positive definite on the nullspace of $B$, any non-unit eigenvalues are real, although negative eigenvalues will occur when $U_2^T T U_2$ is not positive definite.

Precisely because the constraints are preserved,

$$Q^T P_C Q = \begin{bmatrix} 0 & 0 & R^T S \\ 0 & \widehat{T}_{22} & \widehat{T}_{12} S \\ SR & ST_{12} & ST_{11} S \end{bmatrix},$$

where $\widehat{T}_{ij} = U_i^T T U_j$, $i, j = 1, 2$, is an antitriangular matrix when $\widehat{T}_{22}$ is positive definite. Indeed, Keller et al. \([6]\) use a flipped version of (3) to investigate the eigenvalues of $P_C^{-1} A$. The matrix $Q^T P_C Q$ may be applied using the procedure outlined in Section 3, and it makes clear the equivalence between constraint preconditioners and the nullspace method that has previously been observed \([5, 8, 14]\) . Conversely, any matrix $N$ in antitriangular form with $Y = SR$ defines a constraint preconditioner $QNQ^T$ for $A$.

We note that alternative factorisations of constraint preconditioners, some of which rely on a basis for the nullspace of $B$, have also been proposed. In particular, the Schilders’ factorisation \([3, 4, 15]\) re-orders the variables so that $B = [B_1 \ B_2]$, with $B_1 \in \mathbb{R}^{m \times m}$ nonsingular. In this case

$$N = \begin{bmatrix} -B_1^{-1}B_2 \\ I \end{bmatrix}$$

is a basis for the nullspace, and is important to the factorisation.

For a true, inertia-revealing antitriangular factorisation $T$ must be positive definite on the nullspace of $B$, so that $\widehat{T}_{22}$ is symmetric positive definite. If we are only interested in preconditioning (and not in the inertia of $P$) we only require that $\widehat{T}_{22}$ is invertible, although the spectrum may be less favourable in this case.

In summary, we see that applying the same orthogonal similarity transform that makes $A$ antitriangular to $P_D$ and $P_C$ results in preconditioners with specific structures. The antitriangular form of $P_C$ makes the equivalence between constraint preconditioners and the nullspace method clear, and may provide other insights into the properties of constraint preconditioners.
5 Conclusions

We have considerably simplified the antitriangular factorisation for symmetric indefinite matrices of Mastronardi and Van Dooren in the specific and common case of saddle point matrices. This leads to the observation that this factorisation is equivalent to the well known nullspace method. Additionally, we have considered the form of this antitriangular factorisation for popular constraint preconditioning and block diagonal preconditioning, showing how specific structures are preserved.

Acknowledgements The authors would like to thank Gil Strang for inspiring this work and for valuable suggestions. This publication was based on work supported in part by Award No KUK-C1-013-04, made by King Abdullah University of Science and Technology (KAUST).

References


