SIAM J. MATRIX ANAL. APPL.
Vol. 12, No. 3, pp. 469–483, July 1991

STABLE SOLVERS AND BLOCK ELIMINATION FOR BORDERED SYSTEMS*

W. GOVAERTS†

Abstract. Linear systems with a fairly well conditioned matrix M of the form

$$\begin{matrix} \begin{pmatrix} A & b \\ c & d \end{pmatrix} \begin{matrix} n \\ 1 \end{matrix}, \\ n \quad 1 \end{matrix}$$

for which a "black-box" solver for A is available, are considered. To solve systems with M, a mixed block elimination algorithm, called BEM, is proposed. It has the following advantages: (1) It is easier to understand and to program than the widely accepted deflated block elimination (DBE) proposed by Chan, yet allows the same broad class of solvers and has comparable accuracy. (2) It requires one less solve with A. (3) It allows a rigorous error analysis that shows why it may fail in exceptional cases (all other black-box methods known to us also fail in these cases).

BEM is also compared to iterative refinement of Crout block elimination (BEC) introduced by Pryce and Govaerts. BEC allows a more restricted class of solvers than BEM but is faster in cases where a solver is given not for A but for a matrix close to A, which is often the case in applications like numerical continuation theory.

Key words. bordered matrix, block elimination, black-box solver

AMS(MOS) subject classification. 65F30

1. Introduction and notation.

Let

$$M = \begin{matrix} \begin{pmatrix} A & b \\ c & d \end{pmatrix} \begin{matrix} n \\ 1 \end{matrix} \\ n \quad 1 \end{matrix}$$

be a bordered matrix. We want to solve

$$M \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} f \\ g \end{pmatrix}, \tag{1}$$

where x, f are n-vectors and y, g are scalars. In applications like numerical continuation theory, a solver for A is often available because A has special structure (banded, symmetric, sparse, or other). It is then advisable to use this solver to solve systems with M. Difficulties arise when A is nearly singular (in the continuation context this means that we are near a turning point; see Rheinboldt [13]).

Various authors (Keller [7], Moore [8]) solved bordered singular systems by altering A or the elimination strategy. Björck [2] suggests rescaling the last row of M in such a way that Gaussian elimination (further denoted by GE) with row interchanges on M does not pivot to the last row. The problem is then that of GE with a badly scaled matrix and Skeel [14] has shown that in most practical cases one iterative refinement leads to a stable algorithm. This is close (but not equivalent) to BEC + 1 (BEC is to be discussed further; see also Govaerts and Pryce [5]). We concentrate, however, on the case where a solver for A is given as a "black box," which in practice is often the case. The spirit is therefore that of Chan and Resasco [3], [4].

* Received by the editors February 6, 1989; accepted for publication (in revised form) March 1, 1990.

† Senior Research Associate of the Belgian National Fund of Scientific Research (N.F.W.O.); Department of Mathematics, University of Ghent, Krijgslaan 281, B-9000 Ghent, Belgium (govaerts@mathanal.rug.ac.be).

To be precise, we assume that a solver S for A is available, i.e., a map $S: R^n \to R^n$ such that $S(r)$ is an approximate solution to $As = r$.

S is called stable if, when it is applied in floating-point arithmetic of unit roundoff u, there exist a modest constant C_S, a matrix ΔA, and a vector Δr such that

$$(A+\Delta A)S(r) = (r+\Delta r),$$
$$\|\Delta A\| \leqq C_S u \|A\|, \qquad \|\Delta r\| \leqq C_S u \|r\|,$$

where $\|\ \|$ is the 2-norm and C_S will be called the stability constant of S.

Block elimination (BE) is a method to solve (1) by decomposing M blockwise. One way is to use the Crout factorization

$$\begin{pmatrix} A & b \\ c & d \end{pmatrix} = \begin{pmatrix} A & 0 \\ c & \delta \end{pmatrix} \begin{pmatrix} I & v \\ 0 & 1 \end{pmatrix} \tag{2}$$

followed by the solution of two block triangular systems.

This leads to the following algorithm.

ALGORITHM BEC.
1. Solve $Av = b$
2. Compute $\delta = d - cv$
3. Solve $Aw = f$
4. Compute $y = (g - cw)/\delta$
5. Compute $x = w - vy$

Another way is to use the Doolittle factorization

$$\begin{pmatrix} A & b \\ c & d \end{pmatrix} = \begin{pmatrix} I & 0 \\ \xi & 1 \end{pmatrix} \begin{pmatrix} A & b \\ 0 & \delta_1 \end{pmatrix}, \tag{3}$$

again followed by two solutions of block triangular systems.

This amounts to the following.

ALGORITHM BED.
1. Solve $\xi A = c$
2. Compute $\delta_1 = d - \xi b$
3. Compute $y = (g - \xi f)/\delta_1$
4. Solve $Ax = f - by$

Both algorithms provide perfectly satisfying answers if M, A are both well conditioned and the solver for A (and in BED, for A^T) is stable. If A is less well conditioned then it is generally a good idea to improve the obtained result by iterative refinement. If Alg is any algorithm that produces x_1, y_1 out of f, g, we define Alg + k($k = 0, 1, 2, \cdots$) as follows.

ALGORITHM Alg+k.
1. Compute x_1, y_1 out of f, g using Alg
2. For $i = 1, 2, \cdots, k$ do steps 3 to 5
3. Compute the residuals $f_1 = f - Ax_1 - by_1$ and $g_1 = g - cx_1 - dy_1$
4. Compute x_2, y_2 out of f_1, g_1 using Alg
5. Compute $x_1 = x_1 + x_2$, $y_1 = y_1 + y_2$

On first thinking, we might expect that:

(i) BEC and BED have roughly the same behaviour (in many treatments of Gaussian elimination, the difference between the Crout and Doolittle decompositions is hardly noticed).

(ii) If M is well conditioned and A tends to singularity, more and more iterations of BEC (respectively, BED) will be necessary to produce accurate values for x and y.

These assertions are both incorrect, and the behaviour of iterations of BEC and BED is far more complex. In [5] Govaerts and Pryce consider solvers based on an LU or QR decomposition. They show that BEC + 1 produces x and y accurately no matter how ill conditioned A is (except in rare cases of no practical interest). On the other hand, BED produces y accurately but requires several iterations to find x (if at all). As made clear in [5] the remarkable behaviour of BEC + 1 in this case depends on properties of matrix factorizations like LU and QR.

In § 2 we describe some experiments in the case of a solver based on the preconditioned conjugate gradient algorithm. They show that BEC + 1 no longer works in this case and also support the new algorithm BEM that we propose.

Section 3 gives an error analysis of BEM and shows that it usually produces x, y accurately if M is well conditioned and the solver is stable. It also highlights why exceptional cases may cause a failure. Propositions 3.1 and 3.3 further contain the basic ingredients to prove that in practically arising cases, BEM is stable.

Section 4 describes an "exceptional" situation. The aim is to compare the performance of BEM, BEC, a modified version of BEC, the deflated block elimination of Chan and Resasco [3], [4], and iterative refinements of these algorithms in a critical case.

Section 5 draws the final conclusions on the merits and disadvantages of the algorithms.

2. Tests of block elimination algorithms with a solver based on conjugate gradients. In the tests described in this section, A is an 80-by-80 symmetric nonnegative-definite matrix. It is constructed as

$$A = H_{1000}H_{999}\cdots H_2H_1 \operatorname{diag}(1.49, 1.48, \cdots, 0.71, 0)H_1H_2\cdots H_{999}H_{1000},$$

where each matrix H_i ($1 \leqq i \leqq 1000$) is a Householder elementary reflection matrix $H_i = I - 2h_ih_i^T$ and h_i is a normalized random vector. Except for rounding errors, A has singular values 1.49, 1.48, $\cdots$0.71, 0 and it is made nonsingular only by machine imprecision. Obviously, $\|A\| \simeq 1.49$.

Next, b, c, d, x, y are vectors and scalars with coefficients chosen uniformly random in [0, 1]. We then compute $f = Ax + by$ and $g = cx + dy$ and solve the resulting system of the form (1) by BEC, BED, and their iterations.

All computations are done in the PC-version of the Gauss programming language with no extra precision in the computation of residuals or updating the solutions. Here $u = 2^{-52} \approx 2.2\ 10^{-16}$. In all the examples M is well conditioned (2-norm condition number smaller than 200).

The solver for A is the preconditioned conjugate gradient algorithm in Axelsson and Barker [1, § 1.4] with the diagonal of A as a preconditioner. The stopping criterion is that the norm of the residual must be bounded by 10^{-14} times the norm of the computed solution. This ensures that the system with A is solved in a stable way (see [2]). It is to be remarked, however, that we had similar results with other stopping criteria, e.g., prescribing a fixed number of iterations.

Table 1 gives the logarithms of the relative errors of the computed x and y components by BEC + k and BED + k ($k = 0, 1, \cdots, 6$). For comparison, we also give the relative error in the solution by Gaussian elimination with row interchanges on the full matrix M.

The columns BEC-x, BEC-y, and BED-x apparently support the hypothesis that several iterations of BEC and BED are necessary to produce accurate values for x and y. Since A is very nearly singular it may even seem surprising that the algorithms converge

TABLE 1

Logarithms of relative errors in the computed x and y components by BEC, BED, *and their iterations using a preconditioned conjugate gradient solver.*

Number of iterations	BEC		BED	
	x	y	x	y
	−0.2348	−4.3704	−0.8711	−13.5273
1	−4.3612	−8.6889	−2.3433	$-\infty$
2	−9.0570	−13.4635	−6.7891	−15.2063
3	−13.5508	$-\infty$	−12.2967	−15.8083
4	−15.7204	−15.9106	−14.4728	−14.6622
5	−15.2564	−15.9106	−14.4763	−15.1094
6	−14.7037	−15.9106	−14.9900	−15.5073
Full GE	−14.7330		−14.7424	

at all; however, Jankowski and Wozniakowski [6] have shown that iterative refinement of almost any solution scheme to solve linear systems will ultimately converge to an accurate solution (within the bounds posed by the condition of the system and provided the solution scheme gives a solution with relative error smaller than one).

We can make two other observations:

(1) Without any iteration BED produces y accurately. This result is confirmed by many similar experiments and we shall prove it whenever A is solved in a stable way (§ 3).

(2) The relative error in the x-component of the solution by BEC + k + 1 is of the order of the relative error in the y-component of the solution by BEC + k (i.e., in the preceding iteration) for $k = 0, 1, \cdots$. Again, this is confirmed by many similar experiments and it will be proved in the important case where the y-component by BEC + k is accurate (§ 3).

To test this important case further we organize another experiment. The results are collected in Table 2. Here we perform BEC and two iterations starting with the accurate value for y and a zero vector for x. We also give the norm of the right-hand side vector in step 3 of BEC and the norm of the computed solution (the importance of these quantities will be clarified in § 3).

Again, two things are to be remarked:

(1) The first application of BEC already produces both x and y accurately. This is what we hoped for and it confirms the second observation concerning Table 1.

(2) In the first application of A (step 3 of BEC) the computed solution has the same size as the right-hand side (remember that $\|A\| \simeq 1.49$). This is surprising since A is nearly singular and for a random right-hand side vector the computed solution will typically have the size $u^{-1}\|A\|^{-1}$ times the size of the right-hand side. In § 3 we show that this observation is the key to the understanding of the algorithm.

The preceding experiments naturally lead us to first compute y by BED and to use this value, together with a zero vector as approximation to x, in one step of BEC. The resulting algorithm will be called BEM (block elimination mixed). It is given explicitly by the following algorithm.

TABLE 2

Logarithms of relative errors in the computed x and y components by BEC *and two iterations where a correct y and a zero vector for x are introduced (preconditioned conjugate gradient solver).*

	x	y	Norm of right-hand side in step 3	Norm of solution in step 3
Introduced	0	$-\infty$		
BEC	−14.0759	−15.4734	6.2112	5.2935
+1	−15.4510	−15.7744	1.9271E − 15	0.06403
+2	−15.4143	$-\infty$	8.3257E − 16	0.04805
Full GE	−15.1916			

ALGORITHM BEM

1. Solve $\xi A = c$
2. Compute $\delta_1 = d - \xi b$
3. Compute $y = (g - \xi f)/\delta_1$
4. Solve $Av = b$
5. Compute $\delta = d - cv$
6. Compute $f_1 = f - by$
7. Compute $g_1 = g - dy$
8. Solve $Aw = f_1$
9. Compute $y_1 = (g_1 - cw)/\delta$
10. Compute $x = w - vy_1$
11. Compute $y = y + y_1$

Remark that steps 1–3 of BEM are identical to steps 1–3 of BED. Steps 4–5 of BEM are identical to steps 1–2 of BEC. Steps 6–7 compute the residuals given y (from step 3) and a zero vector for x as first approximations to the solution. Steps 8–10 correspond to steps 3–5 of BEC applied to the new right-hand side components f_1, g_1. Finally, step 11 updates y.

Remark that steps 4–5 of BEM are interchangeable with steps 6–7 of BEM. Step 4 of BED is omitted to avoid one solve with A (if included, steps 6–7 have to be adapted and a step 12 is necessary to update x).

TABLE 3

Logarithms of relative errors in the computed x and y components by BEM *and two iterations with* BEC *(preconditioned conjugate gradient solver).*

	x	y	Norm of right-hand side in step 8(BEM), step 3(BEC)	Norm of solution in step 8(BEM), step 3(BEC)
Step 3	0	−15.8359		
Steps 10–11	−13.9947	−14.9328	6.4435	5.2883
+BEC	−15.1867	−15.8359	5.1164E − 14	0.5475
+BEC + 1	−15.5406	$-\infty$	2.3295E − 15	0.01477
Full GE	−15.3589			

Table 3 gives the result of a test with BEM (A, b, c, d, x, y, f, g, as before). Note that BEM produces x, y accurately, as we hoped, and that the right-hand side and computed solution in step 8 have the same order of magnitude (cf. the discussion of Fig. 2).

For completeness, Table 3 also shows the effect of two further iterations with BEC. The improvement so obtained is small and apparently not worth the effort.

3. Error analysis of BEM. Throughout this section we assume that M is well conditioned, i.e., $\kappa(M) = \|M\| \cdot \|M^{-1}\|$ is modest.

Proposition 3.1 and its Corollary 3.2 contain the analysis of steps 1–3 of BEM. The important result is that y, as computed in step 3 of BEM, is accurate even if A is very ill conditioned. This is consistent with the numerical evidence in Table 3 and also explains the observation (1) made in § 2 while discussing Table 1.

Proposition 3.3 is the backward error analysis of BEC. Its Corollary 3.4 draws the important conclusion: the accuracy of the solution obtained by BEC depends exclusively on the size of $\|\bar{w}\|$. This explains why we choose to represent this quantity in Tables 2 and 3.

Now the second part of BEM is precisely an application of BEC to a system transformed by Steps 1–3 and 6–7. Theorem 3.5 shows that in this transformed system, $\|\bar{w}\|$ is usually of order $\|A\|^{-1}\,\|M\|\,\|z\|$ (even for nearly singular A), and therefore BEM produces x, y accurately. This confirms the numerical results of Table 3. From the proof of Theorem 3.5 it is clear that the essential results (a modest $\|\bar{w}\|$ and accurate x, y) remain true if steps 1–3 of BEM are replaced by any method that produces y accurately. This explains the observations (1), (2) in the discussion of Table 2 in § 2.

All computations described in this paper are done in the same floating-point precision u. In general, $\bar{a}$ denotes the computed value of the quantity a (so $\bar{a}$ need not be close to a in any sense).

In the error analysis, we use the notation introduced by Pryce [12] for manipulating the relative error metric introduced by Olver [10] in the scalar case and generalized by Pryce [11] to the vector case. Throughout the analysis, θ_1, θ_2, $\cdots$ denote scalar or $n \times n$ matrix quantities close to the identity. In the scalar case the notation $\theta \in 1(\delta)$ where δ is a nonnegative constant, means $\theta = e^{\varepsilon}$ where $|\varepsilon| \leqq \delta$. In the matrix case, it means that θ is a product of a finite number of matrices $\exp(E_i)$ where $\sum_i \|E_i\| \leqq \delta$.

With this understanding we have

$$\mathrm{fl}(x \text{ op } y) = \theta(x \text{ op } y), \qquad \theta \in 1(u)$$

whenever x, y are scalars and "op" is one of the four basic operations. This remains true if x, y are vectors and "op" is a componentwise combination. It is also true when "op" denotes multiplication of a vector by a scalar.

Furthermore, there is a constant C_{IP} such that

$$\mathrm{fl}(x^T y) = x^T \theta y, \qquad \theta \in 1(C_{IP}u),$$

where θ is a diagonal matrix and $C_{IP} \leqq n$ (cf. [5]; in case of double precision accumulation we have $C_{IP} \simeq 1$).

The obvious bounds

$$\|e^{\theta}\| \leqq e^{\|\theta\|}, \qquad \|e^{\theta} - I\| \leqq \|\theta\|\, e^{\|\theta\|}$$

will often be used without notice.

PROPOSITION 3.1. *Let S be a stable solver for A^T with stability constant C_S. Let $\bar{y}$ be the result computed in step* 3 *of Algorithm* BEM. *Then $\bar{y}$ is the y-component of the*

exact solution of a system near $Mz = h$. *More precisely, there exist* ΔA, Δb, Δc, Δd, Δf, Δg, *and* x_∞ *such that*

$$\begin{pmatrix} A+\Delta A & b+\Delta b \\ c+\Delta c & d+\Delta d \end{pmatrix}\begin{pmatrix} x_\infty \\ \bar{y} \end{pmatrix}=\begin{pmatrix} f+\Delta f \\ g+\Delta g \end{pmatrix} \tag{4}$$

and

$$b+\Delta b=\theta_b b, \qquad \theta_b \in 1((1+C_{IP})u), \tag{4a}$$

$$d+\Delta d=\theta_d d, \qquad \theta_d \in 1(u), \tag{4b}$$

$$f+\Delta f=\theta_f f, \qquad \theta_f \in 1((2+C_{IP})u), \tag{4c}$$

$$g+\Delta g=\theta_g g, \qquad \theta_g \in 1(2u), \tag{4d}$$

$$\|\Delta A\| \leqq C_S u \|A\|, \tag{4e}$$

$$\|\Delta c\| \leqq C_S u \|c\|. \tag{4f}$$

Proof. We have

$$\bar{\xi}(A+\Delta A)=c+\Delta c, \qquad \|\Delta A\| \leqq C_S u\|A\|, \qquad \|\Delta c\| \leqq C_S u\|c\|, \tag{5}$$

$$\theta_1\bar{\delta}_1 = d-\bar{\xi}\theta_2 b, \qquad \theta_1 \in 1(u), \qquad \theta_2 \in 1(C_{IP}u), \tag{6}$$

$$\theta_3\overline{(g-\xi f)} = g-\bar{\xi}\theta_4 f, \qquad \theta_3 \in 1(u), \qquad \theta_4 \in 1(C_{IP}u), \tag{7}$$

$$\theta_5\bar{y}=\overline{(g-\xi f)}/\bar{\delta}_1, \qquad \theta_5 \in 1(u). \tag{8}$$

Combining (6), (7), and (8), we obtain

$$\bar{y}=\frac{\theta_5^{-1}\theta_3^{-1}g-\bar{\xi}\theta_5^{-1}\theta_3^{-1}\theta_4 f}{\theta_1^{-1}d-\bar{\xi}\theta_1^{-1}\theta_2 b}. \tag{9}$$

So $\bar{y}$ is the exact y-component of the solution of

$$\begin{pmatrix} A+\Delta A & \theta_1^{-1}\theta_2 b \\ c+\Delta c & \theta_1^{-1}d \end{pmatrix}\begin{pmatrix} x \\ y \end{pmatrix}=\begin{pmatrix} \theta_5^{-1}\theta_3^{-1}\theta_4 f \\ \theta_5^{-1}\theta_3^{-1}g \end{pmatrix}, \tag{10}$$

from which the proposition follows. □

COROLLARY 3.2. *Let the assumptions of Proposition* 3.1 *be satisfied. Suppose, in addition, that* M *is nonsingular and*

$$uC_M\kappa(M)<1 \tag{11a}$$

where

$$C_M=(2+C_{IP}+2C_S)\exp((1+C_{IP})u). \tag{11b}$$

Then

$$|y-\bar{y}| \leqq C_y u\|z\| \tag{11c}$$

where

$$C_y=\frac{(C_h+C_M)\kappa(M)}{1-uC_M\kappa(M)} \tag{11d}$$

and

$$C_h=(4+C_{IP})\exp((2+C_{IP})u). \tag{11e}$$

Proof. By Proposition 3.1 we have

$$(M+\Delta M)(z+\Delta z)=h+\Delta h \tag{12}$$

where

$$z+\Delta z=\begin{pmatrix} x+\Delta x \\ y+\Delta y \end{pmatrix}=\begin{pmatrix} x_\infty \\ \bar{y} \end{pmatrix} \tag{13}$$

and

$$\Delta M=\begin{pmatrix} \Delta A & \Delta b \\ \Delta c & \Delta d \end{pmatrix}, \qquad \Delta h=\begin{pmatrix} \Delta f \\ \Delta g \end{pmatrix}.$$

Now standard perturbation arguments yield

$$\frac{\|\Delta z\|}{\|z\|}\leqq uC_y,$$

where we have used the bounds (4a)–(4f). From this, (11c) follows. □

PROPOSITION 3.3. *Let S be a stable solver for A with stability constant C_S and let $\bar{x}$, $\bar{y}$ be the components of* (1) *by* BEC. *Then $\bar{x}$, $\bar{y}$ exactly satisfy the matrix equality*

$$\begin{pmatrix} A+\Delta A & b+\Delta b \\ c+\Delta c & d+\Delta d \end{pmatrix}\begin{pmatrix} \bar{x} \\ \bar{y} \end{pmatrix}=\begin{pmatrix} f+\Delta f \\ g \end{pmatrix}+\begin{pmatrix} T \\ U \end{pmatrix}\bar{w} \tag{14}$$

where

$$\|\Delta A\|\leqq(2+C_S)u\exp(2u)\|A\|, \tag{14a}$$

$$\|\Delta b\|\leqq C_S u\|b\|, \tag{14b}$$

$$\|\Delta c\|\leqq(5+C_{IP})u\exp((5+C_{IP})u)\|c\|, \tag{14c}$$

$$\|\Delta d\|\leqq 3u\exp(3u)\|d\|, \tag{14d}$$

$$\|\Delta f\|\leqq C_S u\|f\|, \tag{14e}$$

$$\|T\|\leqq(2C_S+(1+C_S u)\exp u)u\|A\|, \tag{14f}$$

$$\|U\|\leqq(4+2C_{IP})u\exp((6+C_{IP})u)\|c\|. \tag{14g}$$

Proof. The computed quantities $\bar{v}$, $\bar{w}$, $\bar{y}$, $\bar{x}$ satisfy

$$(A+\Delta_v A)\bar{v}=b+\Delta b, \qquad \|\Delta_v A\|\leqq C_S u\|A\|, \qquad \|\Delta b\|\leqq C_S u\|b\|, \tag{15}$$

$$(A+\Delta_w A)\bar{w}=f+\Delta f, \qquad \|\Delta_w A\|\leqq C_S u\|A\|, \qquad \|\Delta f\|\leqq C_S u\|f\|, \tag{16}$$

$$\theta_6\bar{y}=\frac{g-c\theta_7\bar{w}}{d-c\theta_8\bar{v}}, \qquad \theta_6\in 1(3u), \qquad \theta_7\in 1(C_{IP}u), \qquad \theta_8\in 1(C_{IP}u), \tag{17}$$

$$\theta_9\bar{x}=\bar{w}-\theta_{10}\bar{v}\bar{y}, \qquad \theta_9\in 1(u), \qquad \theta_{10}\in 1(u). \tag{18}$$

Eliminating $\bar{v}\bar{y}$ from (17) and (18) we get

$$\theta_6 d\bar{y}+\theta_6 c\theta_8\theta_{10}^{-1}\theta_9\bar{x}=g+c(\theta_6\theta_8\theta_{10}^{-1}-\theta_7)\bar{w}. \tag{19}$$

Combining (18), (15), and (16), we get

$$(A+\Delta_v A)\theta_{10}^{-1}\theta_g\bar{x}+(b+\Delta b)\bar{y}=f+\Delta f+[(\Delta_v A-\Delta_w A)+(A+\Delta_v A)(\theta_{10}^{-1}-I)]\bar{w}. \tag{20}$$

Now (19) and (20) may be rewritten as

$$\begin{pmatrix} A+\Delta A & b+\Delta b \\ c+\Delta c & d+\Delta d \end{pmatrix}\begin{pmatrix} \bar{x} \\ \bar{y} \end{pmatrix}=\begin{pmatrix} f+\Delta f \\ g \end{pmatrix}+\begin{pmatrix} T \\ U \end{pmatrix}\bar{w},$$

where bounds for $\|\Delta A\|$, $\|\Delta b\|$, $\|\Delta c\|$, $\|\Delta d\|$, $\|\Delta f\|$, $\|T\|$, and $\|U\|$ can be computed from the bounds in (15)–(18). □

COROLLARY 3.4. *Let the assumptions of Proposition* 3.3 *be satisfied and define*

$$C'_h=(5+2C_S+uC_S+2C_{IP})\exp((6+C_{IP})u), \tag{21a}$$

$$C'_M=(10+C_{IP}+2C_S)\exp((5+C_{IP})u). \tag{21b}$$

Assume that $uC'_M\kappa(M)<1$ *and define*

$$C_z=\frac{(C'_M+C_S)\kappa(M)}{1-uC'_M\kappa(M)}, \tag{21c}$$

$$C''_h=\frac{C'_h\kappa(M)}{1-uc'_M\kappa(M)}. \tag{21d}$$

Then

$$\|\bar{z}-z\|\leqq uC_z\|z\|+uC''_h\|\bar{w}\|. \tag{22}$$

So if M is well conditioned, then the accuracy of the computed solution $\bar{z}$ *is determined by the size of* $\bar{w}$.

Proof. Rewrite (14) as

$$(M+\Delta M)\bar{z}=\begin{pmatrix} f+\Delta f \\ g \end{pmatrix}+\begin{pmatrix} T \\ U \end{pmatrix}\bar{w},$$

where bounds for $\|\Delta M\|$, $\|\Delta f\|$, $\|T\|$, $\|U\|$ follow from (14a)–(14g).

The result now follows by standard perturbation arguments. □

THEOREM 3.5. *Let* $\bar{x}$ *be the x-component obtained by* BEM. *Let* $\bar{y}$ *be the y-component obtained by* BEM *with step* 11 *omitted, and* $\bar{y}_2$ *the y-component obtained by* BEM *with step* 11 *included. Assume that*

$$uC'_M\kappa(M)<1. \tag{23}$$

Then

$$\|\bar{x}-x\|\leqq uC''_z\|z\|+u^2CC''_hz\|\|(A+\Delta_wA)^{-1}\|\|M\|\|z\|, \tag{24a}$$

$$\|\bar{y}-y\|\leqq uC_y\|z\|, \tag{24b}$$

$$\|\bar{y}_2-y\|\leqq ue^u(C''_z+1)\|z\|+u^2CC''_he^u\|(A+\Delta_wA)^{-1}\|M\|\|z\|, \tag{24c}$$

where we have defined

$$C'''_h=2e^u+(1+uC_y)(4e^{2u}+C_S(1+e^u+2e^{2u})), \tag{25a}$$

$$C'_z=\frac{\kappa(M)(C'_M(1+uC_y)+C'''_h)}{1-uC'_M\kappa(M)}, \tag{25b}$$

$$C''_z=C'_z+C''_h, \tag{25c}$$

$$C=C_S+C_y+e^u+(1+uC_y)(2e^{2u}+C_S(1+e^u+2e^{2u})). \tag{25d}$$

Proof. Let $\bar{y}$ be the y-component computed in step 3 of BEM. Define

$$f_{1,0} = f - b\bar{y}, \tag{26a}$$

$$g_{1,0} = g - d\bar{y}, \tag{26b}$$

$$y_1 = y - \bar{y}. \tag{26c}$$

Then

$$M\begin{pmatrix} x \\ y_1 \end{pmatrix} = \begin{pmatrix} f_{1,0} \\ g_{1,0} \end{pmatrix}. \tag{27}$$

First note that

$$\overline{f_1} = \theta_{11}(f - \theta_{12} b\bar{y}), \qquad \theta_{11}, \theta_{12} \in 1(u), \tag{28a}$$

$$\overline{g_1} = \theta_{13}(g - \theta_{14} d\bar{y}), \qquad \theta_{13}, \theta_{14} \in 1(u). \tag{28b}$$

Applying Proposition 3.3 we get

$$(M + \Delta M)\begin{pmatrix} \bar{x} \\ \bar{y}_1 \end{pmatrix} = \begin{pmatrix} \overline{f_1} + \Delta\overline{f_1} \\ \overline{g_1} \end{pmatrix} + \begin{pmatrix} T \\ U \end{pmatrix}\bar{w} \tag{29}$$

with bounds for ΔM, $\Delta\bar{f}_1$, T, U as in (14a)–(14g). Put

$$\bar{f}_1 = f_{1,0} + \Delta f_1, \tag{30a}$$

$$\bar{g}_1 = g_{1,0} + \Delta g_1. \tag{30b}$$

Then by (26a) and (28a)

$$\|\Delta f_1\| \leqq ue^u \|f\| + 2ue^{2u}(1 + uC_y)\|M\|\|z\|, \tag{31a}$$

$$\|\Delta g_1\| \leqq ue^u \|g\| + 2ue^{2u}(1 + uC_y)\|M\|\|z\|. \tag{31b}$$

Now rewrite (34) as

$$(M + \Delta M)\begin{pmatrix} \bar{x} \\ \bar{y}_1 \end{pmatrix} = \begin{pmatrix} f_{1,0} \\ g_{1,0} \end{pmatrix} + \Delta h. \tag{32}$$

By straightforward computation we obtain from (11c), (14e), and (26a)–(31b)

$$\|\Delta h\| \leqq uC_h''' \|M\|\|z\| + uC_h' \|M\|\|\bar{w}\|. \tag{33}$$

Using (27), (32), (33), the assumption (23), and (11c) again, we obtain

$$\left\| \begin{pmatrix} \bar{x} \\ \bar{y}_1 \end{pmatrix} - \begin{pmatrix} x \\ y_1 \end{pmatrix} \right\| \leqq uC_z'' \|z\| + uC_h'' \|\bar{w}\|. \tag{34}$$

Clearly, the size of $\|\bar{w}\|$ is all-important. By the stability assumption we have

$$(A + \Delta_w A)\bar{w} = \overline{f_1} + \Delta\overline{f_1}, \tag{35}$$

$$\|\Delta_w A\| \leqq uC_S \|A\|, \tag{35a}$$

$$\|\Delta\overline{f_1}\| \leqq uC_S \|\overline{f_1}\|. \tag{35b}$$

By straightforward computations using (26a), (30a), (31a), and (28a) we find

$$\|\bar{w}\| \leqq \|x\| + Cu\|(A + A_w A)^{-1}\|\|M\|\|z\|. \tag{36}$$

Inserting (36) into (34) we get

$$\left\| \begin{pmatrix} \bar{x} \\ \bar{y}_1 \end{pmatrix} - \begin{pmatrix} x \\ y_1 \end{pmatrix} \right\| \leqq uC_z'' \|z\| + u^2 CC_h'' \|(A + \Delta_w A)^{-1}\| \|M\| \|z\|. \tag{37}$$

This implies (24a). Of course (24b) is just (11c).

To prove (24c) first remark that

$$\begin{aligned} \|\bar{y}_2 - y\| &= \|\overline{(\bar{y}_1 + \bar{y})} - y\| \\ &= \|\theta_{15}(\bar{y}_1 + \bar{y}) - y\|, \theta_{15} \in 1(u) \\ &= \|\theta_{15}(\bar{y}_1 - y_1) + \theta_{15}(y_1 + \bar{y}) - y\| \\ &\leqq \|\theta_{15}(\bar{y}_1 - y_1)\| + \|\theta_{15} y - y\| \\ &\leqq e^u \|\bar{y}_1 - y_1\| + ue^u \|y\|. \end{aligned}$$

Formula (24c) follows by inserting the bound in (37) for $|\bar{y}_1 - y_1|$ in this inequality.

DISCUSSION 3.6. (1) The error bounds in (24b) and (24c) suggest that step 11 of BEM can be omitted. This is indeed true for perfectly well conditioned M. Since in practice we deal with less extreme cases, we strongly recommend retaining step 11, whose computing cost is negligible anyway.

(2) The bound in (24a) shows that the accuracy of the x-component computed by BEM depends entirely on the size of $\|(A + \Delta_w A)^{-1}\|$. In particular, the x-component is accurate whenever $\|(A + \Delta_w A)^{-1}\| \lesssim u^{-1}\|M\|^{-1}$. This is the case that typically occurs in practice because roundoff errors in the computation of A and in the solution of systems with A tend to produce this bound.

(3) It is possible to construct highly artificial situations where BEM produces x accurately and $\|(A + \Delta_w A)^{-1}\|$ is arbitrarily large (provided there is no overflow or underflow in the computations). This may be achieved by choosing all components of A, b, c, d, x, y as appropriate integers in such a way that no roundoff error occurs. Typically, however, BEM will produce a completely nonaccurate x-component whenever $\|(A + \Delta_w A)^{-1}\| \gtrsim u^{-2}\|M\|^{-1}$. This is best seen from (35). Indeed, $\bar{f}_1 + \Delta\bar{f}_1$ will probably contain a vector of size at least $u\|M\|\|z\|$ in the singular direction of $(A + \Delta_w A)$. Therefore we expect

$$\|\bar{w}\| \gtrsim u^{-2}\|M\|^{-1} u\|M\|\|z\| \simeq u^{-1}\|z\|.$$

This means that $\bar{x}$ may have a relative error of order one.

(4) In the intermediate case

$$u^{-1}\|M\|^{-1} \leqq \|(A + \Delta_w A)^{-1}\| < u^{-2}\|M\|^{-1},$$

we infer from (24a) that x has a relative error of order less than one. In this case iterative refinement of BEM is in practice very satisfactory (cf. Jankowski and Wozniakowski [6]).

4. A series of experiments in an unusual situation. In this section we describe a series of experiments with four algorithms to solve bordered singular systems.

These methods are BEM, DBE, BEC + 1, and BEC2.

BEM (block elimination mixed) was introduced in § 2 and studied in § 3.

DBE (deflated block elimination) is the method introduced by Chan [3], [4]. We used the form proposed in [4].

BEC + 1 (block elimination (Crout)) is the BEC algorithm described in § 1 with one iterative refinement. It was studied by Govaerts and Pryce [5].

BEC2 is a modification of BEC + 1 in which step 5 of BEC is replaced by simply making all components of x zero. In the iteration, however, step 5 is retained.

As remarked before, in most practically occurring cases, the solver has norm bounded by $u^{-1}\|M\|^{-1}$. This is typically caused by roundoff error even if A is theoretically singular.

Tests with such solvers are described in [3] (DBE) and [5] (BEC + 1). Section 2 of this paper describes a test with BEM in the case of a conjugate gradient solver. These and similar experiments show that all four methods produce accurate results in the cases of practical interest (BEC + 1 and BEC2 only for solvers based on decompositions like LU or QR, not for solvers based on the conjugate gradient method).

Since our error analysis shows that in certain cases of little practical interest BEM may fail, it is of interest to know whether the other methods might do better. Since mildly pathological cases might also arise, we can further ask whether iterative refinement is useful in such cases.

To get insight into the critical cases we consider the ill-reputed matrix $A = W_n$.

$$W_n(i,j) = \begin{cases} 1 & \text{if } i = j, \\ -1 & \text{if } i > j, \\ 0 & \text{if } i < j. \end{cases}$$

If $2^n \gtrsim u^{-1}$, this triangular matrix has a unique small singular value of order 2^{-n}; the near null vector is $(2^{-n+1}, 2^{-n+2}, \cdots, 1)$. Moreover, small perturbations of the nonzero elements of W_n do not essentially change this behaviour and $\|(W_n + \Delta W_n)^{-1}\|$ is of order 2^n in this case (small random perturbations in all elements of W_n, however, tend to reduce $\|(W_n + \Delta W_n)^{-1}\|$ to order u^{-1}). The solver for W_n is forward elimination in all cases and $u \sim 10^{-16}$.

In all the experiments, b, c, d, x, y are chosen uniformly random in [0, 1] and f, g are computed in the same precision as $f = Ax + by$ and $g = cx + dy$. The resulting system is then solved by the four algorithms and for each of them two iterative improvements are performed as well. This is done for $n = 20, 40, 60, 80, 100, 120, 140$, and 160. Since the computed $\bar{y}$ is always accurate ($|\bar{y} - y|/\|z\|$ is of the order of u), only the logarithmic relative error $\log(\|\bar{x} - x\|/\|x\|)$ in the x-component is represented.

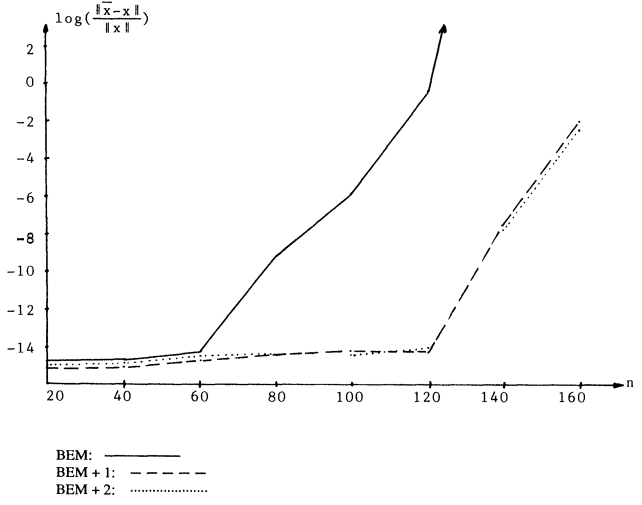

FIG. 1. BEM *with an ill-conditioned triangular A.*

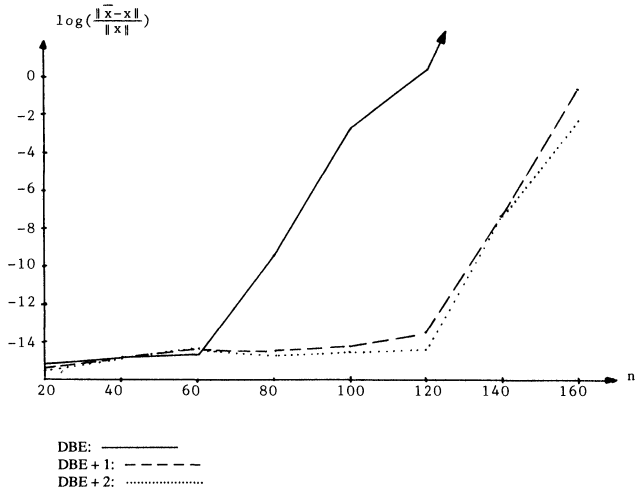

FIG. 2. DBE *with an ill-conditioned triangular* A.

The following should be noted.

(1) Figure 1 shows that BEM produces an accurate x-component for $n \lesssim 60$. Since $2^{60} \sim 10^{18}$, this confirms Discussion 3.6(2).

(2) Figure 1 also shows that BEM + 1 produces accurate results for $n \lesssim 120$ and that more iterations will not further improve the accuracy for higher n.

This is consistent with Discussion 3.6(2) and 3.6(3). Actually, the numerical results are even better than expected from theory. This might be due, however, to the special nature of A.

(3) Figure 2 shows that numerically, DBE behaves very much like BEM. In particular, it is not always a stable method. In the (admittedly rare) cases where it is not, it may be improved greatly by one iterative refinement.

(4) BEC + 1 as such is an inferior method in the case where $\|(A + \Delta_w A)^{-1}\| > u^{-1}\|M\|^{-1}$, since it does not improve by iterative refinement. The reason is obviously

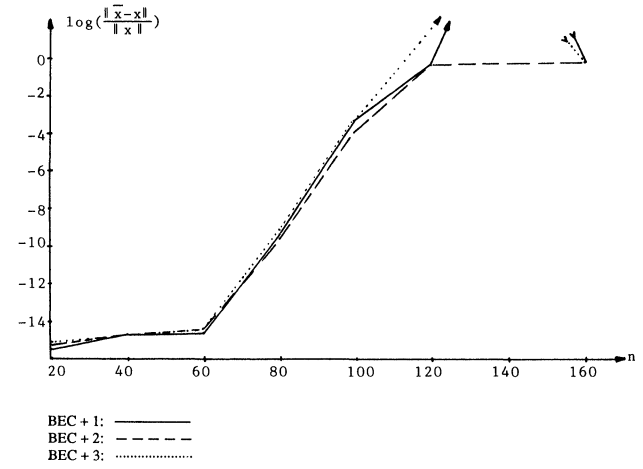

FIG. 3. BEC + k *with an ill-conditioned triangular* A.

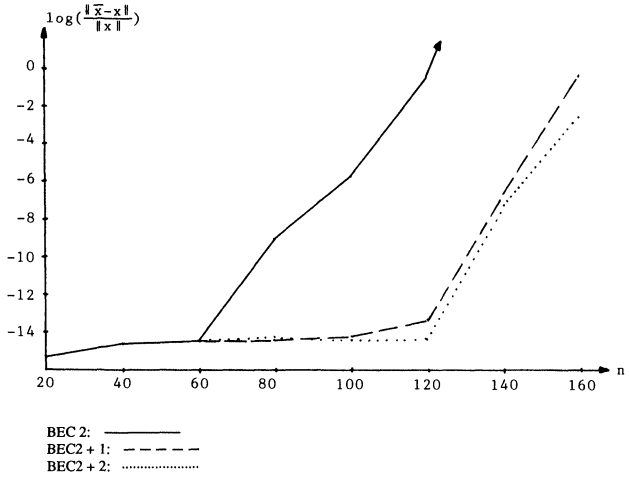

FIG. 4. BEC2 *with an ill-conditioned triangular* A.

that the large size of $\bar{x}$, computed in step 5 of BEC, causes catastrophic roundoff error in the computation of the residual (Fig. 3).

(5) BEC2 avoids this problem by simply omitting the computation of $\bar{x}$ in the first round of BEC. The results are then strikingly similar to those of BEM and DBE (Fig. 4).

5. Conclusions. BEC + 1 is implemented very easily. It has the further possible advantage that it only needs a solver for A, not for A^T. The computational cost is minimal: essentially three solves with A. Next, it fits well in applications like numerical continuation theory where a solver for a matrix close to A might be available. Therefore it is highly recommended in most practical cases.

BEC + 1 has the disadvantage that it requires a solver based on a decomposition like LU, QR, or a similar one. It fails, e.g., for a solver based on the conjugate gradient method for a symmetric positive-semidefinite matrix A.

BEC2 is an alternative to BEC + 1 and has the same properties. Its one advantage is that it can be improved by iterative refinement in some exceptional cases where BEC + 1 fails because $\|(A + \Delta_w A)^{-1}\|$ is excessive. Remark that BEC2 + 1 requires five solves with A.

Now let the solver be general, i.e., not necessarily based on an LU or QR decomposition. A solver for A^T is often also available in practice. Then BEM has the same cost as BEC + 1. A solver for a matrix close to A can be used only if BEM is iterated once. Remark that the cost of an iterative refinement is only one solve with A and that BEM + 1 is also more accurate in the cases with excessive $\|(A + \Delta_w A)^{-1}\|$.

DBE has roughly the same requirements as has BEM and similar performance as well. It uses, however, four solves with A and we think BEM also allows an easier implementation (see also Moore [9] for the error analysis of DBE).

Acknowledgments. The stimulating influence of J. D. Pryce (RMCS, England) led us to write this paper. We also thank L. Paquet (Mons, Belgium) for many critical remarks. The comments of two anonymous referees also led to a substantial improvement in the presentation, in particular, in that of § 2.

REFERENCES

[1] O. AXELSSON AND V. A. BARKER, *Finite Element Solution of Boundary Value Problems*, Academic Press, New York, 1984.

[2] A. BJÖRCK, *Iterative refinement and reliable computing*, in Advances in Reliable Numerical Computing, M. Cox and S. Hammarling, eds., Oxford University Press, Oxford, U.K., 1990.

[3] T. F. CHAN, *Deflation techniques and block-elimination algorithms for solving bordered singular systems*, SIAM J. Sci. Statist. Comput., 5 (1984), pp. 121–134.

[4] T. F. CHAN AND D. C. RESASCO, *Generalized deflated block-elimination*, SIAM J. Numer. Anal., 23 (1986), pp. 913–924.

[5] W. GOVAERTS AND J. D. PRYCE, *Block elimination with one iterative refinement solves bordered linear systems accurately*, BIT, 30 (1990), pp. 490–507.

[6] M. JANKOWSKI AND H. WOZNIAKOWSKI, *Iterative refinement implies numerical stability*, BIT, 17 (1977), pp. 303–311.

[7] H. B. KELLER, *The bordering algorithm and path following near singular points of higher nullity*, SIAM J. Sci. Statist. Comput., 4 (1983), pp. 573–582.

[8] G. MOORE, *The application of Newton's method to simple bifurcation and turning point problems*, Ph.D. thesis, University of Bath, Bath, U.K., 1979.

[9] ———, *Some remarks on the deflated block elimination method*, in Bifurcation: Analysis, Algorithms, Applications, T. Küpper, R. Seydel, and H. Troger, eds., Birkhauser-Verlag, Basel, Switzerland, 1987.

[10] F. W. J. OLVER, *A new approach to error arithmetic*, SIAM J. Numer. Anal., 15 (1978), pp. 368–393.

[11] J. D. PRYCE, *A new measure of relative error for vectors*, SIAM J. Numer. Anal., 21 (1984), pp. 202–215.

[12] ———, *Multiplicative error analysis of matrix transformation algorithms*, IMA J. Numer. Anal., 5 (1985), pp. 437–445.

[13] W. C. RHEINBOLDT, *Numerical Analysis of Parametrized Nonlinear Equations*, John Wiley, New York, 1986.

[14] R. D. SKEEL, *Iterative refinement implies numerical stability for Gaussian elimination*, Math. Comp., 35 (1980), pp. 817–832.