IMA Journal of Numerical Analysis (1986) **6**, 447–452

A Generalization of the Conjugate-Gradient Method to Solve Complex Systems

D. A. H. JACOBS

Central Electricity Research Laboratories, Leatherhead, Surrey

[Received 9 February 1984 and in final revised form April 1986]

The conjugate-gradient method has gained favour recently, notably as a procedure for solving large, preconditioned systems of algebraic equations. Preconditioning techniques have been developed for preparing the system for efficient solution by conjugate gradients. The basic conjugate-gradient method was developed for symmetric, positive definite systems. In this paper a generalization to complex systems is described by developing the work of Fletcher. The methods improve on 'symmetrization', solving the normal equations for the asymmetric case, and expanding the complex system to a real one of twice the order. The method has already proved to be effective.

1. Introduction

THE METHOD of conjugate gradients (CG) was devised by Hestenes & Stiefel (1952). In theory, it is a direct method for solving a system of linear algebraic equations whose coefficient matrix is symmetric and positive definite. CG is a gradient method in which the step along each direction is chosen to minimize, along that direction, a function that measures the residual. Its major feature is that the search directions are mutually conjugate, giving the finite-termination property.

Reid (1971) ignited much of the subsequent development of CG methods for solving systems of algebraic equations: in that paper he showed that CG is a powerful iterative method for suitable systems.

Meijerink & van der Vorst (1977) showed how to modify a general system, such as those derived from finite-difference or finite-element models, into a 'desirable' form to which CG could be applied effectively. This was achieved by preconditioning the system of equations with an easily derived approximate inverse to the coefficient matrix. Since then, some even more effective preconditioning methods have been developed.

In its basic form, CG was derived for systems represented by positive definite, symmetric matrices. Not all systems of equations derived from, for example, finite-difference and finite-element models, have a matrix which is positive definite and symmetric; but from an asymmetric system one can obtain a symmetric, normal system. However, the normal system has a condition number equal to the square of that of the original system, making solution significantly less efficient. For asymmetric systems, the biconjugate-gradient method (bi-CG) was developed by Fletcher (1976), drawing on the analogy of work by Lanczos (1950) for finding the eigenvalues of an asymmetric system. The important

feature of bi-CG is that the condition number of the system is unaltered—indeed CG is just a special case of bi-CG when the coefficient matrix is symmetric.

To generalize conjugate-gradient methods to complex systems, one could treat the system as the equivalent real system of double the order. For many systems this leads to strong coupling between the equations for the real and imaginary components which may make solution difficult, and that method is not considered here.

In this paper, the Fletcher algorithm is generalized to a complex form. The algorithm is stated and then proved in Section 2. This complex-biconjugate-gradient method (Comp-bi-CG) is shown to be a generalization of both bi-CG and CG.

2. The complex-biconjugate-gradient method

The problem posed is the solution of

$$\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b},$$

with $\boldsymbol{A} \in \mathbb{C}^{N\times N}$ and $\boldsymbol{b} \in \mathbb{C}^{N}$ given.

Algorithm

To solve $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}$, given an initial estimate $\boldsymbol{x}_0$ of the solution: Define the initial residual $\boldsymbol{r}_0 = \boldsymbol{b} - \boldsymbol{A}\boldsymbol{x}_0$ and initial search direction $\boldsymbol{p}_0 = \boldsymbol{r}_0$. In parallel, define the biresidual $\hat{\boldsymbol{r}}_0 = \boldsymbol{r}_0^*$ and bidirection $\hat{\boldsymbol{p}}_0 = \boldsymbol{p}_0^*$, where * denotes the complex conjugate. For $i = 0, \ldots, N-1$: Calculate the step-length parameter

$$\alpha_i = \frac{\langle \hat{\boldsymbol{r}}_i, \boldsymbol{r}_i \rangle}{\langle \hat{\boldsymbol{p}}_i, \boldsymbol{A}\boldsymbol{p}_i \rangle}, \tag{1}$$

to obtain the new solution estimate $\boldsymbol{x}_{i+1} = \boldsymbol{x}_i + \alpha_i \boldsymbol{p}_i$. The new residual, and biresidual are

$$\boldsymbol{r}_{i+1} = \boldsymbol{r}_i - \alpha_i \boldsymbol{A}\boldsymbol{p}_i, \qquad \hat{\boldsymbol{r}}_{i+1} = \hat{\boldsymbol{r}}_i - \alpha_i^* \boldsymbol{A}^{\mathsf{H}}\hat{\boldsymbol{p}}_i. \tag{2a,b}$$

To force the biconjugacy of directions, find the biconjugacy coefficient

$$\beta_i = -\frac{\langle \boldsymbol{A}^{\mathsf{H}}\hat{\boldsymbol{p}}_i, \boldsymbol{r}_{i+1} \rangle}{\langle \hat{\boldsymbol{p}}_i, \boldsymbol{A}\boldsymbol{p}_i \rangle} \tag{3}$$

to obtain the next direction vector and bidirection vector:

$$\boldsymbol{p}_{i+1} = \boldsymbol{r}_{i+1} + \beta_i \boldsymbol{p}_i, \qquad \hat{\boldsymbol{p}}_{i+1} = \hat{\boldsymbol{r}}_{i+1} + \beta_i^* \hat{\boldsymbol{p}}_i. \tag{4a,b}$$

(As usual, H means Hermitian conjugate and the inner product $\langle \boldsymbol{x}, \boldsymbol{y} \rangle$ equals $\boldsymbol{x}^{\mathsf{H}}\boldsymbol{y}$, so that $\langle \boldsymbol{x}, \boldsymbol{y} \rangle = \langle \boldsymbol{y}, \boldsymbol{x} \rangle^*$, $\langle \boldsymbol{x}, \boldsymbol{A}\boldsymbol{y} \rangle = \langle \boldsymbol{A}^{\mathsf{H}}\boldsymbol{x}, \boldsymbol{y} \rangle$, and $\langle \alpha\boldsymbol{x}, \boldsymbol{y} \rangle = \alpha^* \langle \boldsymbol{x}, \boldsymbol{y} \rangle$, where α is a scalar.)

Note that, after the first iteration, the bidirections are not usually the complex conjugates of the directions because it is $\boldsymbol{A}^{\mathsf{H}}$ and not $\boldsymbol{A}^*$ in the expression (2b). The latter choice leads to a form of the standard conjugate-gradient algorithm applied to the norm of the complex system.

The algorithm is based on satisfying four sets of relations, namely biorthogonality of successive residuals:

$$\langle \hat{\boldsymbol{r}}_{i+1}, \boldsymbol{r}_i \rangle = 0, \qquad \langle \boldsymbol{r}_{i+1}, \hat{\boldsymbol{r}}_i \rangle = 0 \tag{5a,b}$$

and the biconjugacy conditions of successive search directions:

$$\langle A^{\mathsf{H}} \hat{\boldsymbol{p}}_{i+1}, \boldsymbol{p}_i \rangle = 0, \qquad \langle A \boldsymbol{p}_{i+1}, \hat{\boldsymbol{p}}_i \rangle = 0. \tag{6a,b}$$

These are proved inductively. They are true for $i = 0$. Assume they are true up to, but not including, the current value of i (>0). Then the form of α_i in (1), together with the assumption of (6a) for $i \leftarrow i-1$, gives (5a); likewise (5b) is deduced using (6b) for $i \leftarrow i-1$. The validity of (6b) is the basis of the expression (3) for β_i, found by using (4a), and (6a) is obtained from (5a), which has been proved.

The procedure produces, in theory, a zero value of $\boldsymbol{r}_{i+1}$ after at most N steps because of the important feature of CG-type algorithms, namely, that the biothogonality and biconjugacy conditions hold not only for successive pairs of vectors but also for all pairs:

THEOREM. *The algorithm detailed above generates sequences of vectors which, for all m and n with* $0 \leqslant m < n \leqslant i+1$, *satisfy:*

$$\langle \hat{\boldsymbol{r}}_n, \boldsymbol{r}_m \rangle = \langle \boldsymbol{r}_n, \hat{\boldsymbol{r}}_m \rangle = 0, \qquad \langle \hat{\boldsymbol{p}}_n, A \boldsymbol{p}_m \rangle = \langle A \boldsymbol{p}_n, \hat{\boldsymbol{p}}_m \rangle = 0, \tag{7a,b}$$

provided the algorithm does not break down, i.e. for all i for which $\langle A^{\mathsf{H}} \hat{\boldsymbol{p}}_i, \boldsymbol{p}_i \rangle \neq 0$ *and* $\langle \hat{\boldsymbol{r}}_i, \boldsymbol{r}_i \rangle \neq 0$.

Proof. The proof is a generalization of that used by Fletcher (1976). We proceed by induction. For $i = 0$ we have $n = 1$ and $m = 0$ and the relationships to be proved are then just those that were forced by the choices (1) of α_i and (3) of β_i: namely (5) and (6).

So assume that the theorem is true when i is replaced by $i-1$, and we will prove it for i. Note that for $m = n-1$ we have already proved the relationships in (5) and (6). So we need to consider only the case, $m < i$.

First consider $\langle \hat{\boldsymbol{r}}_{i+1}, \boldsymbol{r}_m \rangle$; then using (2b) followed by (4a):

$$\langle \hat{\boldsymbol{r}}_{i+1}, \boldsymbol{r}_m \rangle = \langle \hat{\boldsymbol{r}}_i - \alpha_i^* A^{\mathsf{H}} \hat{\boldsymbol{p}}_i, \boldsymbol{r}_m \rangle = \langle \hat{\boldsymbol{r}}_i, \boldsymbol{r}_m \rangle - \langle \alpha_i^* A^{\mathsf{H}} \hat{\boldsymbol{p}}_i, \boldsymbol{p}_m - \beta_{m-1} \boldsymbol{p}_{m-1} \rangle,$$

assuming $\beta_{-1} = 0$ if necessary. Since we need to consider only the case $m < i$, the validity follows from (7a) and (7b) by our induction hypothesis with $n = i$; hence the result holds for all $m < i+1$.

In a like manner, it is easy to prove that $\langle \boldsymbol{r}_{i+1}, \hat{\boldsymbol{r}}_m \rangle = 0$ for all $m < i+1$, and so the relations (7a) hold for $n = i+1$.

Now consider $\langle \hat{\boldsymbol{p}}_{i+1}, A \boldsymbol{p}_m \rangle$; by using (4b) and then (2a), we have

$$\langle \hat{\boldsymbol{p}}_{i+1}, A \boldsymbol{p}_m \rangle = \langle \hat{\boldsymbol{r}}_{i+1}, A \boldsymbol{p}_m \rangle + \beta_i \langle \hat{\boldsymbol{p}}_i, A \boldsymbol{p}_m \rangle = \langle \hat{\boldsymbol{r}}_{i+1}, \boldsymbol{r}_m - \boldsymbol{r}_{m+1} \rangle / \alpha_m + \beta_i \langle \hat{\boldsymbol{p}}_i, A \boldsymbol{p}_m \rangle.$$

Again we need consider only the case $m < i$. Using (7a) with $n = i+1$, which has already been proved, and (7b) with $n = i$, which is assumed by the induction

hypothesis, we have $\langle \hat{\boldsymbol{p}}_{i+1}, A\boldsymbol{p}_m \rangle = 0$; hence this result holds for all $m \leqslant i$. Likewise, it can be proved that $\langle A\boldsymbol{p}_{i+1}, \hat{\boldsymbol{p}}_m \rangle = 0$ for all $m \leqslant i$. □

Note that the proof assumes that the algorithm does not break down.

As with bi-CG, one can immediately deduce from equation (4a) applied recursively to itself that

$$\boldsymbol{p}_{i+1} = \boldsymbol{r}_{i+1} + \beta_i \boldsymbol{r}_i + \beta_i \beta_{i-1} \boldsymbol{r}_{i-1} + \ldots,$$

so the orthogonality condition (7a) implies

$$\langle \hat{\boldsymbol{r}}_i, \boldsymbol{p}_j \rangle = 0 \quad \text{for } i > j. \tag{8}$$

Likewise using (4b) followed by (7a) gives

$$\langle \boldsymbol{r}_i, \hat{\boldsymbol{p}}_j \rangle = 0 \quad \text{for } i > j. \tag{9}$$

These are the cross-orthogonality conditions.

Forms for α_i that enable cheaper computer implementation are obtained by (i) substituting (4a) into (1) to eliminate $\boldsymbol{r}_i$ and using (8), or (ii) substituting (4b) into (1) to eliminate $\boldsymbol{r}_i$ and using (9):

$$\text{(i)} \quad \alpha_i = \frac{\langle \hat{\boldsymbol{r}}_i, \boldsymbol{p}_i \rangle}{\langle \hat{\boldsymbol{p}}_i, A\boldsymbol{p}_i \rangle}, \qquad \text{(ii)} \quad \alpha_i = \frac{\langle \hat{\boldsymbol{p}}_i, \boldsymbol{r}_i \rangle}{\langle \hat{\boldsymbol{p}}_i, A\boldsymbol{p}_i \rangle}.$$

3. Discussion

There are several special cases that arise.

(i) If A is Hermitian, i.e. $A^{\mathrm{H}} = A$, then the complex bi-CG algorithm becomes the complex form of the basic CG algorithm, as one would expect, provided we choose $\hat{\boldsymbol{r}}_0 = \boldsymbol{r}_0^*$ and $\hat{\boldsymbol{p}}_0 = \boldsymbol{p}_0^*$. In this case, the analogous result on the strictly monotonic decay of the error measure $h^2 = \langle \boldsymbol{r}, A^{-1}\boldsymbol{r} \rangle$ is obtained.

(ii) If the system is real, then the algorithm reduces immediately to the bi-CG algorithm of Fletcher (1976), and his comments concerning possible breakdown and/or loss of accuracy are relevant here also.

(iii) If the system is real, and A is positive definite and symmetric, then the standard CG algorithm is recovered.

Hence the algorithm described is a generalization of the CG algorithm to general systems with complex elements.

At this point it is worth noting the qualifications in the statement of the Theorem, in particular, that $\langle \hat{\boldsymbol{p}}_i, A\boldsymbol{p}_i \rangle \neq 0$. The possible breakdown of the basic CG method was considered by Fletcher (1976); the use of hyperbolic pairs of directions, to avoid division by zero, was a major part of that paper. However, Fletcher concluded that such an approach has potential stability problems, and indeed it was these that motivated him to obtain bi-CG, which encompasses the minimum-residual and orthogonal-direction algorithms. The latter is a special case of bi-CG, expressed in a form that evidently avoids the potential danger of division by zero. The analogous implementation could be derived for the complex case, but it has not been found necessary to use this in our work.

Regarding convergence, the theoretical finite-termination property is still obtained, as implied by the general biorthogonality and biconjugacy relations (7). However, except for the special cases detailed above, the monotonic decay of the residual measure $h^2 = (\hat{\boldsymbol{r}}, A^{-1}\boldsymbol{r})$ cannot be proved; this, of course, is also the case with Fletcher's bi-CG: the version applicable only to real systems. Indeed, Faber & Manteuffel (1984) have proved that CG-type methods with monotonic convergence to minimization exist only for a special class of asymmetric real systems, and this result naturally extends to complex systems. However, the method has proved in practice to have convergence properties similar to those of the basic CG method, for comparable systems.

The potential difficulties arising from a breakdown or near breakdown of the algorithm, when the denominator of the expressions for α_i and β_i becomes zero or near zero, have not appeared in practice. One can monitor for such a breakdown and employ an alternative strategy in the event of it arising: for example, recommencement of the algorithm from the current solution estimate. Moreover, the method does not suffer from the drawback of squaring the condition number, which occurs in solving the normal equations $A^{\mathrm{H}}A\boldsymbol{x} = A^{\mathrm{H}}\boldsymbol{b}$; a system with real coefficients but complex unknowns and right-hand-side.

The method described forms the basis of two suites of computer software employing preconditioned CG methods (Jacobs, 1983), one of which solves five-and seven-diagonal complex systems such as derived from finite-difference models; the other (Markham, 1983) solves systems of general sparse equations very efficiently. These have been employed to solve a range of both finite-difference and finite-element problems, with many different meshes and boundary conditions. The results obtained show that the method has convergence properties analogous to those of the bi-CG method applied to comparable systems of real equations.

4. Conclusions

This paper has detailed a generalization of the conjugate-gradient method applicable to the solution of general linear systems of algebraic equations with complex coefficients and variables.

This new algorithm overcomes the limitations of the basic conjugate-gradient method which is restricted to systems of equations governed by real, symmetric, positive definite matrices, but at the expense of a potential theoretical breakdown which has not been a problem in practice.

The biconjugate-gradient method was devised by Lanczos (1950) for finding the eigenvalues of an asymmetric matrix. Fletcher (1976) developed it as a method for solving asymmetric and/or indefinite systems.

This paper has described and proved the extension of the biconjugate-gradient method to complex systems of equations, the new method being called complex biconjugate gradients. The method has been proved in computer software implementations.

Acknowledgement

The work was carried out at the Central Electricity Research Laboratories and is published by permission of the Central Electricity Generating Board.

References

FABER, V., & MANTEUFFEL, T. 1984 Necessary and sufficient conditions for the existence of a conjugate gradient method. *SIAM J. numer. Anal.* **21,** 352–362.

FLETCHER, R. 1976 Conjugate gradient methods for indefinite systems. In: *Proceedings of Dundee Conference on Numerical Analysis—1975* (Ed. G. A. Watson), Springer-Verlag, Berlin.

HESTENES, M. R., & STIEFEL, E. 1952 Methods of conjugate gradients for solving linear systems, *J. Res. Nat. Bur. Standards* **49,** 404–436.

JACOBS, D. A. H. 1983 *Preconditioning Techniques in the Numerical Solution of Partial Differential Equations* (Ed. D. J. Evans). Gordon & Breach, London.

LANCZOS, C. 1950 An iteration method for the solution of the eigenvalue problem of linear differential and integral operators. *J. Res. Nat. Bur. Standards* **45,** 255–282.

MARKHAM, G. 1983 A Users Guide to CGS, BCGS, CCGS, Sparse code packages using preconditioned conjugate gradient methods for solving systems of algebraic equations. CEGB Program Note TPRD/L/2394/P82.

MEIJERINK, J. A., & VAN DER VORST, H. A., 1977 An iterative solution method for linear systems of which the coefficient matrix is a symmetric M-matrix. *Math. Comp.* **31,** 148–162.

REID, J. K. 1971 On the method of conjugate gradients for the solution of large sparse systems of linear equations. In: *Large Sparse Sets of Linear Equations* (Ed. J. K. Reid). Academic Press, London.