Ann. Inst. Statist. Math.
Vol. 42, No. 1, 187–201 (1990)

A CLASS OF SCALED DIRECT METHODS FOR LINEAR SYSTEMS*

JOZSEF ABAFFY[1] AND EMILIO SPEDICATO[2]

[1]*University of Economics, Budapest, Hungary*
[2]*Istituto Universitario di Bergamo, Dipartimento di Matematica, Statistica, Informatica ed Applicazioni, Bergamo, Italy*

(Received September 29, 1987; revised March 13, 1989)

Abstract. A generalization of the class of direct methods for linear systems recently introduced by Abaffy, Broyden and Spedicato is obtained by applying these algorithms to a scaled system. The resulting class contains an essentially free parameter at each step, giving a unified approach to finitely terminating methods for linear systems. Various properties of the generalized class are presented. Particular attention is paid to the subclasses that contain the classic Hestenes-Stiefel method and the Hegedus-Bodocs biorthogonalization methods.

Key words and phrases: Linear systems, direct methods, scaling of equations, conjugate direction methods, biorthogonalization methods.

1. Introduction

In a series of recent papers, Abaffy (1979), Abaffy and Spedicato (1984), Abaffy *et al.* (1984*a*), have introduced a class of algorithms (here named the ABS class) for solving linear algebraic systems of the form

$$(1.1)\qquad A^Tx = b \quad x \in R^n, \quad b \in R^m, \quad A = (a_1,\dots,a_m), \quad a_i \in R^n,$$

where $m \le n$ and no assumption is made about the rank of A. The algorithms of the class compute a solution x^+ of the given system in a finite number of steps (at most m), generating at each step an approximation x_i of the solution. The algorithms of the ABS class are based upon the following procedure (assuming exact arithmetic):

(A) Let H_1 be an arbitrary nonsingular matrix and let x_1 be an arbitrary vector in R^n; set $i = 1$.

(B) Compute $s_i = H_i a_i$.

---

*This work was partially supported by CNR under contract 85.02648.01.

(C) If $s_i = 0$ and $a_i^T x_i - b_i = 0$, set $x_{i+1} = x_i$, $H_{i+1} = H_i$, increment i by one and go to (B) (in this case, the i-th equation depends linearly on the previous equations); if $s_i = 0$ and $a_i^T x_i - b_i \neq 0$, stop (the i-th equation is incompatible); if $s_i \neq 0$ go to (D).

(D) Compute the search vector p_i by formula

$$p_i = H_i^T z_i \,, \tag{1.2}$$

where $z_i \in R^n$ is an arbitrary parameter vector save for the condition that

$$z_i^T H_i a_i \neq 0 \,. \tag{1.3}$$

(E) Compute the approximation x_{i+1} of the solution by the following formula

$$x_{i+1} = x_i - (a_i^T x_i - b_i)/(a_i^T p_i) p_i \,. \tag{1.4}$$

(F) If $i = m$, stop (x_{m+1} is the solution); otherwise update H_i by the following formula

$$H_{i+1} = H_i - H_i a_i w_i^T H_i \,, \tag{1.5}$$

where $w_i \in R^n$ is an arbitrary parameter vector save for the following condition:

$$w_i^T H_i a_i = 1 \,. \tag{1.6}$$

(G) Increment i by one and go to (B).

Particular algorithms in the ABS class are obtained by making specific choices of the available parameters, say H_1, z_i and w_i. Of interest are the following algorithms:

(i) The implicit LQ or symmetric algorithm, previously considered by Huang (1975):

$$H_1 = I, \qquad z_i = a_i, \qquad w_i = a_i/(a_i^T H_i a_i) \,. \tag{1.7}$$

(ii) The pseudosymmetric algorithm, a version of the symmetric algorithm (to which it is mathematically equivalent), which has performed best in numerical experiments of Abaffy and Spedicato (1983):

$$H_1 = I, \qquad z_i = H_i a_i, \qquad w_i = H_i a_i / \| H_i a_i \|^2 \,. \tag{1.8}$$

(iii) The implicit Gauss-Choleski or $LU - LL^T$ factorization algorithm (well-defined iff all principal minors of A are nonsingular, otherwise row pivoting has to be performed):

$$H_1 = I, \quad z_i = e_i/(|e_i^T H_i a_i|)^{1/2}, \quad w_i = e_i/(e_i^T H_i a_i) . \tag{1.9}$$

When x_1 is the null vector, it can be shown that the above algorithm generates the same sequence of iterates x_i as the classical escalator method.

The ABS class can be applied to overdetermined linear systems for determining a least squares generalized solution (see Spedicato (1984)), or to nonlinear algebraic equations (see Abaffy *et al.* (1984)). In this paper we present a natural generalization of the ABS class for linear systems, which results in a larger class, where an additional parameter is available. Such a generalized class contains, in a new formulation, well-known methods like conjugate direction methods; in fact, the class is essentially a realization of the general finitely terminating iterative algorithm of Stewart (1973), which is implemented in terms of usual factorizations, and of Broyden (1985), where no determination is given of vectors p_i. The application of the generalized ABS class to nonlinear systems is considered by Abaffy and Galantai (1986).

2. A generalization of the ABS class

Let us assume, for simplicity of formulation, that A is full rank. Consider, instead of system (1.1), the following scaled system:

$$V^T A^T x = V^T b , \tag{2.1}$$

where $V = (v_1, v_2, \ldots, v_m)$, $v_i \in R^m$, is an arbitrary nonsingular matrix. Systems (1.1) and (2.1) are equivalent, any solution of one being a solution of the other. If we apply the ABS algorithm to system (2.1), we find that equations (1.3)–(1.6) take the following form (s_i being nonzero since A is full rank):

$$z_i^T H_i A v_i \neq 0 , \tag{2.2}$$

$$x_{i+1} = x_i - (v_i^T r_i)/(v_i^T A^T p_i) p_i , \tag{2.3}$$

$$H_{i+1} = H_i - H_i A v_i w_i^T H_i , \tag{2.4}$$

$$w_i^T H_i A v_i = 1 , \tag{2.5}$$

where $r_i \in R^m$ is the residual vector of system (1.1) in x_i, say

$$r_i = A^T x_i - b . \tag{2.6}$$

We can make the following important observations:

(i) Equations (2.2)–(2.5) are obtained from equations (1.3)–(1.6) just by substituting a_i by Av_i and b_i by $v_i^T b$.

(ii) At step i only the i-th column v_i of matrix V is used. As V is an arbitrary nonsingular matrix, v_i can be interpreted as a new parameter available at the i-th step, arbitrary save for the condition of being linearly independent from the previously chosen parameters $v_1, v_2, \ldots, v_{i-1}$.

(iii) The vector x_{m+1} computed by the equations (1.2), (2.2)–(2.6) solves not only the scaled system $V^T A^T x = V^T b$ but also system $A^T x = b$; thus, if v_i is interpreted as a new parameter, the ABS algorithm with equations (2.2)–(2.6) can be interpreted as a generalized algorithm for solving (1.1). We shall call this generalized class the ABSg class of algorithms for linear systems.

Observing that every property valid for the ABS class of the form $Q(H_i, a_i, z_i, w_i)$ can be reformulated for the ABSg class, if A and V are full rank, as a property of the form $Q(H_i, Av_i, z_i, w_i)$, we can state that the following relations are true, being the reformulation of similar properties proved for the ABS class:

$$H_i H_1^{-1} H_j = H_i \quad j \leq i , \tag{2.7}$$

$$H_j H_1^{-1} H_i = H_i \quad j \leq i , \tag{2.8}$$

$$H_i A v_j = 0 \quad j < i , \tag{2.9}$$

$$H_i A v_j \neq 0 \quad j \geq i , \tag{2.10}$$

$$\sum_{j=i}^{m} \beta_j H_i A v_j = 0 \quad \Rightarrow \quad \beta_j = 0 , \tag{2.11}$$

$$\text{rank } (H_i) = n - i + 1 , \tag{2.12}$$

$$\sum_{j=1}^{m} \beta_j p_j = 0 \quad \Rightarrow \quad \beta_j = 0 , \tag{2.13}$$

$$V^T A^T P = L , \tag{2.14}$$

where L is a nonsingular lower triangular matrix and $P = (p_1, p_2, \ldots, p_m)$. If $m = n$, (2.14) gives the following (implicit) factorization of A^T:

$$A^T = (V^T)^{-1} L P^{-1} . \tag{2.15}$$

The property valid in the ABS class that x_{i+1} is a solution of the first i equations of system (1.1) is not generally true for the ABSg class, x_{i+1} being instead a solution of the first i equations of the scaled system $V^T A^T x = V^T b$. However, we can characterize the sequence x_i with the following property:

THEOREM 2.1. *Let $s \in R^n$ be an arbitrary vector; let x_i be the vector generated at the $(i-1)$-th step of the ABSg algorithm. Then the vector $\tilde{x}$ given by the following relation*

$$\tilde{x} = x_i + H_i^T s \,, \tag{2.16}$$

satisfies the $i-1$ equations

$$v_j^T(A\tilde{x} - b) = 0 \quad j < i \,. \tag{2.17}$$

PROOF. Immediate by induction.

The proof of the following theorem can be established in similar way as the proof of Theorem V in Abaffy *et al.* (1984*a*):

THEOREM 2.2. *The ABSg class is well-defined; say for every choice of the nonsingular matrix V there exist choices of z_i and w_i such that conditions* (2.2) *and* (2.5) *are satisfied.*

THEOREM 2.3. *For any nonsingular V the following choice of w_i for $i = 1, 2, \ldots, m$*

$$w_i = Av_i/(v_i^T A^T H_i A v_i) \tag{2.18}$$

is well-defined, satisfies condition (2.5) *and implies that the generated matrices H_{i+1} are symmetric.*

PROOF. Immediate by using properties (2.7) and (2.8).

Remark 1. The update obtained by the above choice of w_i will be called the generalized symmetric update.

THEOREM 2.4. *Let $H_1 = I$ and x_1 have the form $x_1 = \sum_{j=1}^{k} \beta_j A v_j$ with $k < m$. Choose w_i as in* (2.18) *and z_i as $z_i = Av_i$. Then for $i > k$, x_i is the vector of minimum euclidean norm among all vectors $\tilde{x}$ such that $v_j^T(A\tilde{x} - b) = 0$ for $j < i$.*

PROOF. Clearly vector x_i is of the form

$$x_i = \sum_{j=1}^{k} \beta_j A v_j + \sum_{j=1}^{i-1} \gamma_j p_j, \quad \gamma_j = (v_j^T r_j)/(v_j^T A^T p_j) \,. \tag{2.19}$$

Now equation (2.16) gives for arbitrary s the most general expression for a vector $\tilde{x}$ satisfying conditions $v_j^T(A\tilde{x} - b) = 0$, $j < i$. Taking the norm of $\tilde{x}$ we get

$$\tilde{x}^T\tilde{x} = x_i^Tx_i + s^TH_iH_i^Ts + 2s^T\left[\sum_{j=1}^{k} \beta_j H_iAv_j + \sum_{j=1}^{i-1} \gamma_j H_ip_j\right]. \tag{2.20}$$

In the summation the first term is null because of (2.9); in the second term we have, from the choice of z_j, the symmetry of the update and (2.7) that $H_ip_j = H_iH_j^TAv_j = H_iH_jAv_j = H_iAv_j$, which is null again because of (2.9). Thus it follows that the minimum value of $\tilde{x}^T\tilde{x}$ is $x_i^Tx_i$, corresponding to any choice of s in the null space of H_i.

Remark 2. The algorithm where z_i and w_i are chosen as in Theorem 2.4 will be called the generalized symmetric algorithm.

An additional characterization of the generalized symmetric algorithm is given by the following theorem (where norms are euclidean norms):

THEOREM 2.5. *Let the sequence x_i be generated by the symmetric algorithm. Then, for $i = 2,\ldots,m + 1$, the sequence $\|x_i\|$ is monotonically increasing.*

PROOF. Let S_i be the orthogonal complement to the space spanned by $v_1,\ldots,v_{i-1}$ and Z_i be the set of vectors x such that the residual in x belongs to S_i. From Theorem 2.4 x_i is the minimum euclidean norm vector in Z_i. As $S_{i+1} \subseteq S_i$ and $Z_{i+1} \subseteq Z_i$ the inequality $\|x_{i+1}\| < \|x_i\|$ would contradict the minimality of $\|x_i\|$ in Z_i.

Remark 3. Theorem 2.5 indicates that the solution x_{m+1} is approached by the symmetric algorithm from below, a regularization property of great interest.

THEOREM 2.6. *Let $H_1 = I$; then among all possible choices of w_i in* (2.4) *subject to* (2.5) *the one which minimizes the Frobenius norm of the correction to H_i is given by the symmetric update choice* (2.18); *moreover, for such a choice the Frobenius norm of H_i satisfies the following relation*

$$\|H_i\|^2 = n - i + 1\ . \tag{2.21}$$

PROOF. The first statement is just a reformulation of a similar result proved for the symmetric algorithm by Abaffy and Spedicato (1983). To prove the second statement let $s_i = H_iAv_i$ and note that $H_is_i = s_i$ from (2.7). Now we have

$$\begin{aligned}(2.22)\qquad \|H_{i+1}\|_F^2 &= \mathrm{Tr}\,[(H_i^T - s_is_i^T/s_i^Ts_i)(H_i - s_is_i^T/s_i^Ts_i)] \\ &= \mathrm{Tr}\,(H_i - s_is_i^T/s_i^Ts_i) \\ &= \mathrm{Tr}\,(H_i) - 1 \\ &= \|H_i\|_F^2 - 1\,,\end{aligned}$$

and the result follows since $\|I\|_F^2 = n$.

3. Alternative representations of the update matrix

The following theorem is a reformulation of Theorems 6, 9, 10 in Abaffy *et al.* (1984*a*):

THEOREM 3.1. *Define the matrices* $V_i = (v_1, v_2, \ldots, v_i)$ *and* $W_i = (w_1, w_2, \ldots, w_i)$. *Then*

(a) W_i *is full rank.*

(b) *Matrix* $W_i^T H_1 A V_i$ *is nonsingular and LU decomposable.*

(c) *Update* (2.4) *can be written in the form.*

$$(3.1)\qquad H_{i+1} = H_1 - H_1AV_i(W_i^TH_1AV_i)^{-1}W_i^TH_1\,.$$

(d) *If* $H_1 = I$, *then for* $1 \le j \le m$ *the vectors* H_jAv_j *and* $H_j^Tw_j$ *satisfy the follwing biorthogonality relation*

$$(3.2)\qquad w_j^TH_jH_iAv_j = 0 \qquad j \neq i\,.$$

We show now that for the subclass of the ABSg class where z_i is proportional to w_i, it is not necessary to update at step i a full square matrix H_i but just a set of $n - i$ vectors in R^n, or, in other words, a rectangular matrix whose number of columns decreases by one at every step. This result, of great theoretical and computational interest, had not been disclosed in the previous analysis of the ABS class.

THEOREM 3.2. *Consider the ABSg algorithm with the following parameter choices:* H_1 *arbitrary nonsingular,* $v_1, \ldots, v_n$ *arbitrary linearly independent,* $z_i = u_i$ *and* $w_i = u_i/u_i^TH_iAv_i$ *with* u_i *arbitrary such that* $u_i^TH_iAv_i \neq 0$. *Then the algorithm is well-defined and it generates the same sequence* x_i *which is produced by the following algorithm:*

(A′) *Let* H_1 *and* x_1 *be given as in the above defined ABSg algorithm; set* $i = 1$.

(B′) *For* $j = 1, 2, \ldots, m$ *compute vectors* $u_j^1 \in R^n$ *by formula*

$$(3.3)\qquad u_j^1 = H_1^Tu_j\,.$$

(C′) *Compute the new approximation to the solution by*

$$x_{i+1} = x_i - (v_i^T r_i)/(v_i^T A^T u_i^i) u_i^i . \tag{3.4}$$

(D′) *If* $i = m$ *stop,* x_{m+1} *is the solution; otherwise for* $j = i + 1$, $i + 2,\dots, m$ *compute vectors* $u_j^{i+1} \in R^n$ *by the formula*

$$u_j^{i+1} = u_j^i - (v_i^T A^T u_j^i)/(v_i^T A^T u_i^i) u_i^i . \tag{3.5}$$

(E′) *Increment the index i by one and go to* (C′).

PROOF. It is obvious that the ABSg algorithm with the above parameter choices is well-defined. To establish the identity of the sequences x_i it is enough to prove that $p_i = u_i^i$ and that the denominators in (3.4) and (3.5) are nonzero. For $i = 1$ this is true, since $u_1^1 = H_1^T u_1 = H_1^T z_1 = p_1$ and $v_1^T A^T u_1^1 = v_1^T A^T H_1^T u_1$ is nonzero by assumption. For $i > 1$ the result follows by identifying $p_i = H_i^T z_i$ with u_i^i and verifying that the update of p_i according to formulas (1.2) and (2.4) is identical to the update of u_i^i according to formula (3.5), and that the denominator in (3.4) and (3.5) is identical to $u_i^T H_i A v_i$.

The subclass of the ABSg algorithm defined by equations (3.3), (3.4) and (3.5) will be called the condensed ABSg class.

THEOREM 3.3. *The vectors* u_j^i, $i \le j \le m$, *defined in* (3.5) *are nonzero and linearly independent for* $1 \le i \le m$.

PROOF. It follows from the structure of update (2.4) that every property of the form $Q(H_i, Av_i, w_i)$ can be reformulated as a property of the form $Q(H_i^T, w_i, Av_i)$. Under the assumption that $Av_1,\dots, Av_i$ are linearly independent, it follows, see (2.10) and (2.11), that $H_i A v_j$, $j \le i$, is nonzero and linearly independent. Since $w_1,\dots, w_i$ are linearly independent (see Theorem 3.1) it follows similarly that $H_i^T w_j$, $j \le i$, is nonzero and linearly independent. The result follows since u_j^i and $H_i^T w_j$ are proportional by a nonzero factor.

Remark 4. Parameter choices which satisfy the requirements of Theorem 3.1 are the following:

(i) The generalized symmetric algorithm

$$u_i = A v_i . \tag{3.6}$$

(ii) The generalized pseudosymmetric algorithm

$$u_i = H_i A v_i \,. \tag{3.7}$$

(iii) The generalized implicit LU algorithm (under the additional assumption that all principal minors of AV be nonsingular)

$$u_i = e_i / (e_i^T H_i A v_i) \,. \tag{3.8}$$

Remark 5. When Av_i is known, the number of multiplications required by the condensed ABSg algorithm at step i is no more than $2n(n-i) + O(n)$, implying a total number of multiplications, for $m = n$, equal to $n^3 + O(n^2)$. Note that the formulation of the ABSg algorithm in terms of matrices H_i in general requires $3n^3 + O(n^2)$ multiplications (for z_i proportional to w_i), dropping to $3/2n^3 + O(n^2)$ for the symmetric algorithm and $n^3/3 + O(n^2)$ for the implicit $LU - LL^T$ algorithm. The condensed ABSg algorithm still requires only $n^3/3 + O(n^2)$ multiplications for the generalized implicit $LU - LL^T$ algorithm (if Av_i is known); indeed, if u_i is proportional to e_i, equation (3.6) implies that vectors u_j^{i+1} have only $i+1$ nonzero components. Thus step i requires only $2i(n-i+1) + O(n)$ multiplications and the result follows.

THEOREM 3.4. *Let x_i and u_j^i, $j \geq i$, be generated by the condensed ABSg algorithm. Then the set of vectors $\tilde{x}$ such that $v_j^T(A\tilde{x} - b) = 0, j \leq i$, has the following form (for $m = n$)*

$$\tilde{x} = x_i + \sum_{j=i}^{n} \alpha_j u_j^i \,, \tag{3.9}$$

where the α_j are arbitrary.

PROOF. We know from Theorem 2.1 that vectors $\tilde{x}$ have the form $\tilde{x} = x_i + H_i^T s$, where x_i, H_i are generated by any method in the ABSg class and s is arbitrary. As vectors w_j are linearly independent from Theorem 3.1, we can write $s = \sum_{j=1}^{n} \beta_j w_j$. Since any property of the form $Q(H_i, Av_i, w_i)$ corresponds to a property of the form $Q(H_i^T, w_i, Av_i)$ it follows from relation (2.9) that $H_i^T w_j = 0$ for $j < i$. Thus we have $s = \sum_{j=i}^{n} \beta_j w_j$ and the result follows from the definition of u_j^i.

4. Generating A-conjugate search vectors

Algorithms generating search vectors that are A-conjugate can be obtained in the ABSg class when A is symmetric positive definite and the choice $v_i = p_i$ is made.

THEOREM 4.1. *Let A be symmetric and positive definite. Then the subclass of the ABSg class where $v_i = p_i$ is well-defined. Moreover, the following relation is true*:

$$P^T A P = D , \tag{4.1}$$

where D is a diagonal matrix with positive diagonal elements.

PROOF. For any vector z_i such that $p_i = H_i^T z_i$ is nonzero, condition (2.2) is satisfied, for $v_i = p_i$, since $z_i^T H_i A v_i = p_i^T A p_i$ is positive from the assumption on A. Moreover, $z_i^T H_i A v_i > 0$ implies $H_i A v_i \neq 0$, so that condition (2.5) can be satisfied by a suitable choice of w_i. As the p_i's are linearly independent, so are the v_i's, implying that the subclass is well-defined. From (2.14) we have $P^T A^T P = L$. Taking the transpose we have, from symmetry of A, $P^T A^T P = L = P^T A P$, which implies the diagonality of L, $L = D$; moreover, for $j = 1,\ldots, n$, $D_{jj} = p_j^T A p_j$ is positive since A is positive definite.

The subclass of the ABSg class where $v_i = p_i$ still contains as free parameters H_1, z_i and w_i. A sequence of symmetric matrices H_i is obtained by the following choice of w_i

$$w_i = A p_i / p_i^T A H_i A p_i . \tag{4.2}$$

Formula (4.2) for w_i is well-defined, since the denominator is positive. With the further choice $z_i = A p_i / \eta_i$, η_i arbitrary nonzero scalar, a realization would be obtained of the generalized symmetric algorithms with search vectors that are simultaneously orthogonal and A-conjugate (as the eigenvectors of A are). Since the definition of p_i and the considered choice of z_i imply $H_i^T A p_i = \eta_i p_i$, the determination of p_i is not possible in explicit form, being equivalent to the computation of the eigenvectors of $H_i^T A$.

The parameter choices corresponding to the generalized implicit LU algorithm are $H_1 = I$, z_i proportional to e_i and

$$w_i = e_i / e_i^T H_i A p_i . \tag{4.3}$$

It is easy to show by induction that the above algorithm is well-defined. Indeed it corresponds to applying the standard implicit LU algorithm to the problem with coefficient matrix $P^T A^T$. Such a matrix is strongly nonsingular, since its i-th principal minor is the i-th principal minor of A premultiplied by the matrix comprising the first i columns and rows of P_i^T, which is a nonsingular lower triangular matrix. The number of multiplications required by the algorithm is $5/6n^3 + O(n^2)$, $n^3/2$ multiplications coming from the evaluation of the vector $A p_i$.

We show now that it is possible to determine the parameters z_i and w_i such that the sequence x_i can be built using only two vectors, the algorithm becoming identical with the Hestenes and Stiefel method.

THEOREM 4.2. *Let A be symmetric positive definite. Let $H_1 = I$, and suppose that, for $i \geq 1$, $r_i \neq 0$ (otherwise stop the algorithm at the first index i for which $r_i = 0$; x_i is the solution). Take the following parameter choices at step i: $z_i = r_i$, $v_i = p_i = H_i^T r_i$, $w_i = p_i / p_i^T H_i A p_i$. Then*

(A) *The algorithm is well-defined and $w_i = p_i / p_i^T A p_i$.*

(B) *The sequence of vectors x_i, p_i is identical with that generated by the Hestenes and Stiefel method (with the same starting point).*

(C) *The scalar products $v_i^T A^T u_j^i$ in (3.5) are identically zero for $j > i+1$.*

PROOF. To prove (A) we note that condition (2.5) is satisfied if $p_i^T H_i A p_i \neq 0$. Now $p_i^T H_i A p_i = r_i^T H_i H_i A p_i = r_i^T H_i A p_i = p_i^T A p_i$ because of (2.7). Thus from positive definiteness of A it follows that $p_i^T H_i A p_i > 0$ if $p_i \neq 0$. Condition (2.2) is satisfied if $r_i^T H_i A p_i = p_i^T A p_i \neq 0$, which is again true if $p_i \neq 0$. For $i = 1$ this is true from the assumptions. For $i > 1$ it follows from the proof of statement (B), where it is shown that p_i is identical to the i-th search vector generated by the Hestenes-Stiefel method, which is nonzero if r_i is nonzero. To prove (B) let x_i', p_i' be the vectors generated by the Hestenes-Stiefel method. Since $x_1' = x_1$ and $p_1' = r_1 = p_1$ it follows immediately that $x_2' = x_2$. To extend this result to other indices, let us write the formulas defining the Hestenes-Stiefel iteration for general i:

$$x_{i+1}' = x_i' + (p_i'^T r_i') / (p_i'^T A p_i') p_i' , \tag{4.4}$$

$$p_{i+1}' = r_{i+1}' - (p_i'^T A r_{i+1}') / (p_i'^T A p_i') p_i' . \tag{4.5}$$

With the given parameter choices, the ABSg algorithm can be written in the condensed form, giving the following relations

$$x_{i+1} = x_i - (p_i^T r_i) / (p_i^T A p_i) p_i , \tag{4.6}$$

$$u_j^{i+1} = u_j^i - (p_i^T A u_j^i) / (p_i^T A p_i) p_i \quad j = i+1, \ldots, n . \tag{4.7}$$

Note that u_j^1, $1 \leq j \leq n$, has the form $u_j^1 = r_j$, and these vectors cannot be actually computed at the beginning of the iteration, since only r_1 is known. However, it is a consequence of statement (C) that the computation of u_j^1 is not actually needed.

Equations (4.4) and (4.6) give the same vectors if vectors p_i, p_i', x_i, x_i' are the same. For $i = 1$ this was observed to be true. For $i = 2$ and $p_2 = u_2^2$ relation (4.7) becomes

(4.8) $$u_2^2 = u_2^1 - (p_1^T A u_2^1)/(p_1^T A p_1) p_1 ,$$

since $u_2^1 = r_2 = r_2'$ and $p_1 = p_1'$ then $p_2 = p_2'$ and $x_3 = x_3'$. For general indices we proceed by induction, assuming that $u_j^j = p_j = p_j'$ and $x_j = x_j'$ for $j \leq i$. It follows immediately that $x_{i+1} = x_{i+1}'$. From (4.7) we get

(4.9) $$u_{i+1}^{i+1} = u_{i+1}^i - (p_i'^T A u_{i+1}^i)/(p_i'^T A p_i') p_i' ,$$

implying that $p_{i+1} = u_{i+1}^{i+1} = p_{i+1}'$ if $u_{i+1}^i = r_{i+1}$. Applying (4.7) backwards we have

(4.10) $$u_{i+1}^i = u_{i+1}^1 - \sum_{j=1}^{i-1} (p_j'^T A u_{i+1}^j)/(p_j'^T A p_j') p_j' ,$$

as $u_{i+1}^1 = r_{i+1}$ the identity $p_{i+1} = p_{i+1}'$ is established (and statement (A)) if we show that $p_j'^T A u_{i+1}^j = 0$, $j \leq i-1$ (and so proving also statement (C)). Applying (4.7) again backwards we have

(4.11) $$u_{i+1}^j = u_{i+1}^1 - \sum_{k=1}^{j-1} (p_k'^T A u_{i+1}^k)/(p_k'^T A p_k') p_k' ,$$

or from the choice of z_i and obvious definition of β_k

(4.12) $$u_{i+1}^j = r_{i+1} - \sum_{k=1}^{j-1} \beta_k p_k' .$$

Since vectors $p_1', \ldots, p_j'$ are A-conjugate, it follows from (4.12) and the induction that

(4.13) $$p_j^T A u_{i+1}^j = p_j'^T A r_{i+1} ,$$

which is zero from a well-known property of the Hestenes-Stiefel method.

Remark 6. Theorem 4.2 clearly establishes the equivalence with the various forms of the Hestenes-Stiefel method which have appeared in the literature (Fletcher-Reeves, Polak-Ribière etc.). Along similar lines it is possible to derive explicit expressions for parameters z_i, w_i in the ABSg class which generate algorithms equivalent to many other conjugate direction methods.

5. Relations with the Hegedus-Bodocs algorithm for A-conjugate vector pairs

In a series of recent papers Hegedus (1982) and Hegedus and Bodocs

(1982) have introduced recursions for generating, for a given symmetric matrix A, sets of A-conjugate or A-biorthogonal vector pairs (v_i, u_i), $i = 1,\dots,n$, which satisfy the following relation

$$(5.1) \qquad v_j^T A u_k = 0 \quad j \neq k .$$

Hegedus and Bodocs' recursions are of the following type. Suppose that vectors v_j and u_j satisfy (5.1) for $j \leq i$; then two additional vectors v_{i+1} and u_{i+1} satisfying (5.1) are obtained by the formulas

$$(5.2) \qquad v_{i+1} = P_i^T r_{i+1} ,$$

$$(5.3) \qquad u_{i+1} = Q_i q_{i+1} ,$$

where P_i and Q_i are nonorthogonal projectors of the form

$$(5.4) \qquad P_i = I - \sum_{j=1}^{i} A u_j v_j^T / (v_j^T A u_j) ,$$

$$(5.5) \qquad Q_i = I - \sum_{j=1}^{i} u_j v_j^T A / (v_j^T A u_j) ,$$

and r_{i+1} and q_{i+1} are essentially arbitrary vectors, save for the condition

$$(5.6) \qquad r_{i+1}^T P_i A Q_i q_{i+1} \neq 0 .$$

In the following theorem we show that, if the columns of matrix V in the ABSg class are identified with vectors v_i in the Hegedus-Bodocs relations, then it is possible to choose the parameters z_i, w_i in such a way that vectors u_i become identical with vectors p_i. Thus the Hegedus-Bodocs recursions appear as a special case of the recursions associated with the ABSg class.

THEOREM 5.1. *Let A be symmetric and let r_i, q_i, $1 \leq i \leq n$, be the vectors chosen in the Hegedus-Bodocs recursions satisfying condition* (5.6). *Consider the subclass of the ABSg class corresponding to the following parameter choices*: $H_1 = I$, $v_i = P_i^T r_i$, $z_i = q_i$, $w_i = q_i / q_i^T H_i A v_i$. *Then such parameter choices are well-defined and the identity $u_i = p_i$ is true.*

PROOF. For $i = 1$ the result is immediate. Assume now that the sequence p_j, H_j is well-defined and that $u_j = p_j$ for $j \leq i$. In order that H_{i+1} be well-defined, (2.5) must be satisfied, which is true if $q_i^T H_i A v_i \neq 0$. From the definition of p_i and the induction we have identically $q_i^T H_i A v_i = p_i^T A v_i = u_i^T A v_i$, which is nonzero due to (5.2), (5.3) and (5.6). Thus p_{i+1} can be determined and we prove first that it equals u_{i+1} and then that it satisfies (2.2). From (5.3) and (5.5) we have, using the induction

(5.7) $$u_{i+1} = q_{i+1} - \sum_{j=1}^{i} p_j v_j^T A q_{i+1} / (v_j^T A p_j) .$$

Observing that with the assumed parameter choices the ABSg algorithm can be written in the condensed form, we can write

(5.8) $$p_{i+1} = u_{i+1}^i - (v_i^T A u_{i+1}^i) / (v_i^T A p_i) p_i .$$

Applying (5.4) backwards we have

(5.9) $$p_{i+1} = u_{i+1}^1 - \sum_{j=1}^{i} (v_j^T A u_{i+1}^j) / (v_j^T A p_j) p_j .$$

Again applying (5.4) backwards we have, with β_k some coefficients

(5.10) $$u_{i+1}^j = u_{i+1}^1 - \sum_{k=1}^{j-1} \beta_k p_k .$$

From the induction and the A-conjugacy of the v_j and $u_j = p_j$ we have

(5.11) $$v_j^T A u_{i+1}^j = v_j^T A u_{i+1}^1 ,$$

and thus

(5.12) $$p_{i+1} = u_{i+1}^1 - \sum_{j=1}^{i} (v_j^T A u_{i+1}^1) / (v_j^T A p_j) p_j ,$$

implying that $p_{i+1} = u_{i+1}$, since $u_{i+1}^1 = H_1^T z_{i+1} = q_{i+1}$. We can now immediately prove inequality (2.2) observing that $p_{i+1}^T A v_{i+1} = u_{i+1}^T A v_{i+1} = q_{i+1}^T Q_{i+1}^T A P_{i+1}^T r_{i+1}$ which is nonzero because of (5.6).

Remark 7. The well-definiteness condition (5.6) is satisfied if A is positive definite and $V = P$.

6. Final remarks and conclusions

In this paper we have presented a generalization of the ABS algorithm, obtained by applying it to a scaled system. The columns of the scaling matrix play the role of additional parameters available at each iteration, allowing the generation of infinitely more algorithms. We have shown that conjugate direction algorithms (including the classic Hestenes-Stiefel algorithm) and the general biorthogonal direction algorithm of Hegedus and Bodocs can be obtained by particular choices of the available parameters. It is actually possible to show that essentially all algorithms with finite termination for linear systems correspond to particular parameter choices in the scaled ABS algorithm. More about this question will appear in a forthcoming monograph by Abaffy and Spedicato (1989). It is also possible to apply the generalized ABS algorithm for solving nonlinear systems. For convergence results in such a case, see Abaffy and Galantai (1986). Numerical experiments are presently being performed to find whether better algorithms than the classic ones can be determined in the generalized ABS class.

REFERENCES

Abaffy, J. (1979). A linearis egyenletrendszerek altalanos megoldsanak egy direkt modszerosztalya, *Alkalmaz. Mat. Lapok*, **4**, 233–240.

Abaffy, J. and Galantai, A. (1986). Conjugate direction methods for linear and nonlinear systems of algebraic equations, Report DMSIA 7/86, Univ. of Bergamo, Italy.

Abaffy, J. and Spedicato, E. (1983). On the symmetric algorithm in the ABS class of direct methods for linear systems, Report SOFMAT 7/83, Univ. of Bergamo, Italy.

Abaffy, J. and Spedicato, E. (1984). A generalization of Huang's method for solving systems of linear equations, *Boll. Un. Mat. Ital. B*, **3**, 517–529.

Abaffy, J. and Spedicato, E. (1989). *ABS Projection Algorithms: Mathematical Techniques for Linear and Nonlinear Equations*, Ellis Horwood.

Abaffy, J., Broyden, C. G. and Spedicato, E. (1984*a*). A class of direct method for linear systems, *Numer. Math.*, **45**, 361–376.

Abaffy, J., Galantai, A. and Spedicato, E. (1984*b*). Convergence properties of the ABS algorithm for nonlinear algebraic equations, Report DMSIA 7/84, Univ. of Bergamo, Italy.

Broyden, C. G. (1985). On the numerical stability of Huang's and related methods, *J. Optim. Theory Appl.*, **47**, 401–412.

Hegedus, C. (1982). Generalization of the method of conjugate gradients: The method of conjugate pairs, Report KFKI 79/82, Central Res. Inst. for Physics, Budapest.

Hegedus, C. and Bodocs, L. (1982). General recursions for *A*-conjugate pairs, Report KFKI 56/82, Central Res. Inst. for Physics, Budapest.

Huang, H. Y. (1975). A direct method for the general solution of a system of linear equations, *J. Optim. Theory Appl.*, **16**, 429–445.

Spedicato, E. (1984). On the solution of linear least squares through the ABS class for linear systems, Report DMSIA 8/84, Univ. of Bergamo, Italy.

Stewart, G. (1973). Conjugate direction methods for solving systems of linear equations, *Numer. Math.*, **21**, 285–297.