NOTICE WARNING CONCERNING COPYRIGHT RESTRICTIONS:
The copyright law of the United States (title 17, U.S. Code) governs the making
of photocopies or other reproductions of copyrighted material. Any copying of this
document without permission of its author may be prohibited by law.
CONJUGATE DIRECTION METHODS FOR SOLVING SYSTEMS OF LINEAR EQUATIONS

by

G. W. Stewart

Department of Computer Science Carnegie-Mellon University Pittsburgh, Pa. 15213

This work was supported by the Office of Naval Research under contract number N00014-67-A-0314-0018, NR 044-452
ABSTRACT

A generalization of the notion of a set of directions conjugate to a matrix is shown to lead to a variety of finitely terminating iterations for solving systems of linear equations. The errors in the iterates are characterized in terms of projectors constructable from the conjugate directions. The natural relations of the algorithms to well known matrix decompositions are pointed out. Some of the algorithms can be used to solve linear least squares problems.
1. **Introduction**

The purpose of this paper is to describe a general class of algorithms for solving the equation

\[(1.1) \quad Ax = b,\]

where \(A\) is a nonsingular matrix of order \(n\) and \(x\) and \(b\) are \(n\)-vectors. The algorithms improve an approximate solution \(x_0\) by stepping along a set of "conjugate" directions \(u_1, u_2, \ldots, u_n\) in such a way that the \(n\)-th vector so produced is the solution of (1.1). We shall show that with a suitable definition of conjugacy many well known methods, and some less well known ones, can be derived as special cases of our general algorithm.

The prototype for the class of conjugate direction algorithms was described by Fox, Huskey, and Wilkinson [1]. They take \(A\) to be Hermitian and define the linearly independent vectors \(u_1, u_2, \ldots, u_n\) to be \(A\)-conjugate if

\[i \neq j \Rightarrow u_i^H A u_j = 0.\]

Equivalently if \(U = (u_1, u_2, \ldots, u_n)\), then \(u_1, u_2, \ldots, u_n\) are \(A\)-conjugate if \(U^H A U\) is diagonal (and of course nonsingular). Starting with \(x_0\), a solution \(x_n\) of (1.1) may be constructed by the following algorithm:

\[(1.2)\]

1) For \(k = 1, 2, \ldots, n\)
   1) \(r_{k-1} = b - Ax_{k-1}\)
   2) \(\nu_k = u_k^H r_{k-1}/u_k^H A u_k\)
   3) \(x_k = x_{k-1} + \nu_k u_k\)
It should be noted that the division in statement 1.2 of (1.2) can always be carried out, since \( u_k^T A u_k \) is the \( k \)-th diagonal entry of \( U^T A U \), which is diagonal and nonsingular.

The above algorithm would be useless without a method for generating conjugate directions. Fox, Huskey, and Wilkinson show how a set of conjugate directions can be constructed recursively as linear combinations of the unit vectors \( e_1, e_2, \ldots, e_n \). At each step the conjugation algorithm requires no more work than the solution of a triangular system, and the algorithm as a whole is therefore not an unreasonable method for solving linear systems.

The same algorithm was rediscovered, apparently independently, by Hestenes and Stiefel [3]. They showed that the conjugation algorithm could be regarded as a variant of Gaussian elimination on the matrix \( A \). Moreover, they pointed out that the set of directions generated by the conjugate gradient algorithm is \( A \)-conjugate, thus exhibiting the method of conjugate gradients as a special conjugate direction algorithm.

In 1955 Householder [4] described a class of iterations which he called orthogonalization methods. Like the original method of conjugate directions, an orthogonalization method steps along a set of specially generated directions until, after a finite number of steps, a solution is reached. However, \( A \) need not be Hermitian and the directions are not \( A \)-conjugate. The algorithm that generates the directions again involves only the solution of triangular systems. An important feature of the method is that is can be used to solve linear least squares problems (cf. the comments at the end of §2 below).

All of these algorithms are closely related to standard factorizations and reductions of matrices. As was mentioned above, Hestenes and Stiefel
have pointed out the relation between their conjugation algorithm and Gaussian elimination, so that the method can be regarded as connected with the LU factorization of $A$ into the product of a lower and an upper triangular matrix. Householder has related a special case of his method to the QR factorization of $A$ into the product of a unitary matrix and an upper triangular matrix. Finally the method of conjugate gradients can be regarded as a variant of the Lanczos biorthogonalization algorithm [6] (for the exact connection see [5, p. 139] or §3.5 below).

This paper is divided into two parts. In §2 the notion of an $A$-conjugate pair is introduced and a terminating iteration for solving the system (1.1) is described. A theory of conjugate projectors is developed and used to characterize the errors in the successive approximate solutions. Section 3 is devoted to the description of a general conjugation algorithm and its consequences. In particular it is shown that different choices of the parameters in the algorithm lead to various methods, some well known, for solving linear systems, and that these methods are closely related to well known matrix factorizations.

Ideally the paper should end with a section detailing the author's extensive numerical experience with these algorithms. But the number of conjugate direction algorithms is quite literally infinite, and the choice of any single algorithm will probably be indicated by its suitability for the problem at hand. It is hoped that this paper will encourage independent workers to experiment with specific algorithms in various applications.
Another gap in the paper is the absence of a discussion of the application of conjugate direction methods to the solution of nonlinear equations of the form

\[ f(x) = 0 \]

where \( f: \mathbb{C}^n \rightarrow \mathbb{C}^n \). This may be done by identifying the value \( f(x_k) \) with the residual \( r_k \) and the derivative \( F_x \) with the matrix \(-A\). In some of the conjugate direction schemes all that is required of \( A \) is that one be able to evaluate \( Ap \) for any vector \( p \). In solving nonlinear equations this value may be approximated by

\[ \frac{f(x + \alpha p) - f(x)}{\alpha} \]

for some suitable value of \( \alpha \), which circumvents the need of calculating \( F_x \) explicitly. We shall not pursue this line here; however, those who do may find the theory of §2 useful in constructing local convergence proofs.

Throughout the paper we shall use Householder's notational conventions. In addition \( \mathbb{C}^n \) will denote complex n space, and \( \mathbb{C}^{m \times n} \) the set of \( m \times n \) matrices. The column space of \( A \) will be denoted by \( \mathcal{R}(A) \) and the null space by \( \mathcal{N}(A) \). Given any matrix \( A \), the matrices \( A^k \), \( A^k_1 \), \( A^{-k} \), and \( A^k_1 \) will denote the submatrices consisting of respectively the first \( k \) rows, the first \( k \) columns, the last \( k \) rows, and the last \( k \) columns of \( A \). Thus \( A^k_1 \) is the leading principal submatrix of \( A \) of order \( k \).
2. **Conjugacy**

The proof that the vector $x_n$ generated by (1.2) is a solution of (1.1) consists of verifying inductively that the $k$-th residual $r_k$ is orthogonal to $u_1, u_2, \ldots, u_k$. Since $U$ is nonsingular, $r_n$ must be the null vector; i.e. $b - Ax_n = 0$.

The point to be noted is that the vectors $u_1, u_2, \ldots, u_n$ serve two purposes: first they provide directions along which the approximate solutions $x_k$ are to be altered, and second they delineate the subspaces in which the residuals $r_k$ are forced to lie. The essential part of our generalization of the notion of conjugacy is to provide a second set of vectors to serve the second purpose.

**Definition 2.1.** Let $A, U, V \in \mathbb{C}^{n \times n}$ be nonsingular. Then $(U, V)$ is an $A$-conjugate pair if

$$L = V^H AU$$

is lower triangular.

The generalized algorithm for solving (1.1) is a slight variant of (1.2).

**Algorithm 2.2.** Let $A, U, V \in \mathbb{C}^{n \times n}$ be nonsingular with $U = (u_1, u_2, \ldots, u_n)$ and $V = (v_1, v_2, \ldots, v_n)$ forming an $A$-conjugate pair. Let $b, x_0 \in \mathbb{C}^n$.

1) For $k = 1, 2, \ldots, n$

1) $r_{k-1} = b - Ax_{k-1}$

2) $u_k = v_k^R r_{k-1} / u_k^H Au_k$

3) $x_k = x_{k-1} + u_k u_k^*$
Again it should be noted that the algorithm can always be carried to completion; for the denominator $v_k^H A u_k$ in statement 1.2 is the $k$-th diagonal of the lower triangular matrix $L$ and must be nonzero since $L$ is nonsingular. The last vector $x_n$ produced by the algorithm is the solution of (1.1).

**Theorem 2.3.** In Algorithm 2.2

$$A x_n = b.$$ 

This theorem can be proved in three ways, each of which has advantages. The simplest way is to show inductively that $r_k$ is orthogonal to $v_1, v_2, \ldots, v_k$, which implies that $r_n = 0$.

A second proof may be had by regarding $A$ as a linear transformation of $\mathbb{C}^n$ into $\mathbb{C}^n$. If the domain of $A$ is equipped with the basis formed from the columns of $U$ and the range with the basis formed from the columns of $V^H$, then the matrix representing the transformation $A$ is the lower triangular matrix $L$. Moreover, in this coordinate system, Algorithm 2.2 becomes nothing more than the usual recursive algorithm for solving $L y = V^H b$.

The third proof follows from a detailed investigation of the errors in the $x_k$, which we now give. Let

$$e_k = x - x_k.$$ 

Then

$$e_k = x - (x_{k-1} + u_k v_k^H)$$

$$= e_{k-1} - \frac{u_k v_k^H}{v_k^H A u_k}.$$
Now
\[ r_{k-1} = A e_{k-1}. \]

Hence
\[ e_k = (I - P_k)e_{k-1}, \]

where
\[ p_k = \frac{u_k v_k^H}{v_k^H A u_k}, \quad (2.1) \]
and generally
\[ e_k = (I - P_k)(I - P_{k-1}) \ldots (I - P_1)e_0. \quad (2.2) \]

Thus the problem of characterizing \( e_k \) becomes one of characterizing the matrix \( (I - P_k)(I - P_{k-1}) \ldots (I - P_1) \).

It is easily verified that
\[ p_k^2 = p_k; \]
i.e., \( p_k \) is a projector. In fact from (2.1) it is seen that \( p_k \) is the rank one projector onto the space spanned by \( u_k \) along the orthogonal complement of the space spanned by \( A^H v_k \). Moreover, by the A-conjugacy of \( U \) and \( V \), we have
\[ i < k \Rightarrow v_i^H A u_k = 0. \]

It follows that
\[ i < k \Rightarrow p_i p_k = 0. \quad (2.3) \]
It turns out that the property (2.3) is sufficient to enable us to describe
\((I - P_k)(I - P_{k-1})... (I - P_1)\) regardless of the dimensionality of the
projectors \(P_i\). Thus we make the following definition.

**Definition 2.4.** Let \(P_1, P_2, \ldots\) be a sequence of projectors. Then
\(P_1, P_2, \ldots\) are **conjugate projectors** if

\[ i < k \implies P_i P_k = 0. \]

The following lemma is an immediate consequence of Definition 2.4.

**Lemma 2.5.** Let \(P_1, P_2, \ldots\) be conjugate projectors. Then

\[ (2.4) \quad i < k \quad P_i (I - P_k)(I - P_{k-1})... (I - P_1) = 0. \]

**Proof.** By conjugacy, we have for \(i < j\),

\[ (2.5) \quad P_i (I - P_j) = P_i - P_i P_j = P_i; \]

and since \(P_i\) is a projector,

\[ (2.6) \quad P_i (I - P_i) = P_i - P_i^2 = P_i - P_i = 0. \]

Together (2.5) and (2.6) imply (2.4).

**Theorem 2.6.** Let \(P_1, P_2, \ldots\) be conjugate projectors. Then

\[ (I - P_k)(I - P_{k-1})... (I - P_1) = I - \tilde{P}_k, \]

where \(\tilde{P}_k\) is the projector onto \(\bigoplus_{i=1}^k \mathcal{R}(P_i)\) along \(\bigcap_{i=1}^k \mathcal{N}(P_i)\).
Proof. The theorem is trivial for \( k = 1 \). Assume its truth for \( P_1, P_2, \ldots, P_{k-1} \) and let \( \tilde{Q}_k = (I - P_k)(I - P_{k-1}) \cdots (I - P_1) \). Now from Lemma 2.5 it follows that

\[ i \leq k \implies (I - P_i)\tilde{Q}_k = \tilde{Q}_k, \]

and hence that \( \tilde{Q}_k = \tilde{Q}_k^2 \); i.e. \( \tilde{Q}_k \) is a projector.

The column space of \( \tilde{Q}_k \) is given by

\[ \mathcal{R}(\tilde{Q}_k) = \{ x : \tilde{Q}_k x = x \}. \]

Let \( \tilde{Q}_k x = x \). Then by Lemma 2.5

\[ P_i x = P_i \tilde{Q}_k x = 0, \quad (i = 1, 2, \ldots, k). \]

Hence \( x \in \bigcap_{i=1}^k \mathcal{N}(P_i) \). On the other hand if \( x \in \bigcap_{i=1}^k \mathcal{N}(P_i) \), then it is easily verified \( \tilde{Q}_k x = \tilde{Q}_k x \). This shows that \( \mathcal{R}(\tilde{Q}_k) = \bigcap_{i=1}^k \mathcal{N}(P_i) \).

Since the sequence \( P_1^H, P_2^H, \ldots, P_k^H \) is conjugate it follows that

\[ \mathcal{R}(\tilde{Q}_k^H) = \bigcap_{i=1}^k \mathcal{N}(P_i^H). \]

But \( \bigcap_{i=1}^k \mathcal{N}(P_i^H) \) is the orthogonal complement of

\[ \bigcap_{i=1}^k \mathcal{R}(P_i). \]

Now any projector \( P \) is the projector onto \( \mathcal{R}(P) \) along the orthogonal complement of \( \mathcal{R}(P^T) \). Hence \( \tilde{Q}_k \) is the projector onto \( \bigcap_{i=1}^k \mathcal{N}(P_i) \) along \( \bigcap_{i=1}^k \mathcal{R}(P_i) \). The complementary projector \( \tilde{P}_k = I - \tilde{Q}_k \)

is therefore the projector onto \( \bigcap_{i=1}^k \mathcal{R}(P_i) \) along \( \bigcap_{i=1}^k \mathcal{N}(P_i) \).
We must still justify the use of the direct sum in the characterization of $\mathcal{R}(\tilde{P}_k)$. By the induction hypothesis, it is sufficient to show that 

$$\mathcal{R}(P_k) \cap \mathcal{R}(\tilde{P}_{k-1}) = \{0\}.\text{ Now from the conjugacy conditions it follows that } Q_{k-1}P_k = P_k \text{ or } \tilde{P}_{k-1}P_k = 0. \text{ Hence if } P_k x = x (x \in \mathcal{R}(P_k)) \text{ and } \tilde{P}_{k-1} \tilde{x} = x (x \in \mathcal{R}(\tilde{P}_{k-1})) \text{ then }

$$x = \tilde{P}_{k-1} x = \tilde{P}_{k-1} P_k x = 0.$$

Returning to the characterization (2.2) of $e_k$, we obtain the following result as a consequence of Theorem 2.6.

**Theorem 2.7.** In Algorithm 2.2, the errors $e_k = x - x_k$ are given by

$$e_k = (I - \tilde{P}_k)e_0,$$

where $\tilde{P}_k$ is the projector onto $\mathcal{R}(U^1)^k$ along $\mathcal{R}(U^{n-k})$ (or equivalently along the orthogonal complement of $\mathcal{R}(A_hV^1)^k$).

**Proof.** It follows immediately from theorem 2.6 and the form of the projectors $P_i$ that $\tilde{P}_k$ is the projector onto $\mathcal{R}(U^1)^k$ along the orthogonal complement of $\mathcal{R}(A_hV^1)^k$. By the conjugacy of $(U, V)$, the orthogonal complement of $\mathcal{R}(A_hV^1)^k$ is $\mathcal{R}(U^{n-k})$, which establishes the theorem.

Since $\tilde{P}_n = I$, it follows that $e_n = 0$, which proves Theorem 2.3.

The projector $P_k$ can be represented as follows. Let $W = U^{-1}$. Then it is easy to verify that

$$\tilde{P}_k = U^1 W^k.$$
and

\[ I - P_k = U^{\perp} \in \mathbb{R}^{n-k} \times \mathbb{R}^{n-k}. \]

Since the residual vector \( r_k \) is simply \( A e_k \), we have

\[ r_k = (I - AP_k A^{-1}) r_0. \]

The matrix \( I - AP_k A^{-1} \) is a projector. In fact, we have the following easy corollary of Theorem 2.7.

**Corollary 2.8.** The residuals in Algorithm 2.2 are given by

\[ r_k = Q_k r_0, \]

where \( Q_k \) is the projector onto the orthogonal complement of \( (V^l_k) \) along the orthogonal complement of \( (A U^l_k) \).

We conclude this section with two extensions of the notion of conjugacy.

**The linear least squares problem.** Let \( A \in \mathbb{C}^{m \times n} \), \( V \in \mathbb{C}^{m \times n} \), and \( U \in \mathbb{C}^{n \times n} \), where \( m \geq n \). Then \((U,V)\) will be said to be \( A\)-conjugate if \( V^H AU \) is nonsingular and lower triangular. If \( b \in \mathbb{C}^m \) and \( x_0 \in \mathbb{C}^n \), then Algorithm 2.2 may be applied to yield a sequence of vectors \( x_1, x_2, \ldots, x_n \). The theory developed above applies to the errors and residuals associated with these vectors. In particular, by Corollary 2.8 the associated residual vector \( r_n \) lies in the orthogonal complement of \( \mathcal{R}(V) \). If \( V \) is chosen so that \( \mathcal{R}(V) = \mathcal{R}(A) \), then \( r_n \) lies in the orthogonal complement of the column space of \( A \), which is sufficient (see, e.g., [5, p. 8]) for \( x_n \) to be a solution of the linear least squares problem of minimizing \( ||b - Ax|| \), where \( || \cdot || \) denotes the usual Euclidean norm.
Block conjugacy. Let $A, U, V \in \mathbb{C}^{n \times n}$ be nonsingular, and let $U$ and $V$ be partitioned in the forms

$$U = (U_1, U_2, \ldots, U_r),$$

$$V = (V_1, V_2, \ldots, V_r),$$

where

$$U_i, V_i \in \mathbb{C}^{n \times n_i}, (i = 1, 2, \ldots, r).$$

Then $(U, V)$ is block $A$-conjugate if

$$i < j \Rightarrow V_i^H AU_j = 0.$$

Given block $A$-conjugate $U$ and $V$, an algorithm along the lines of Algorithm 2.2 may be given as follows.

1) For $k = 1, 2, \ldots, r$
   1) $r_{k-1} = b - Ax_{k-1}$
   2) $m_k = (V_k^H A_{k-1})^{-1} V_k r_{k-1}$
   3) $x_k = x_{k-1} + U_k m_k$.

As was done above, it can be shown that

$$e_k = (I - P_k) e_{k-1}, \quad (k = 1, 2, \ldots, r),$$

where $P_k$ is the projector

$$P_k = U_k (V_k^H A_{k-1})^{-1} V_k^H A.$$
In fact $P_1, P_2, \ldots, P_r$ is a sequence of conjugate projectors, and the above theory applies. In particular $e_k$ is the projection of $e_0$ onto $\mathcal{P} \{(U_1, \ldots, U_k)\}$ along $\mathcal{P} \{(U_{k+1}, \ldots, U_r)\}$.

3. Conjugation.

Algorithm 2.2 is of no practical value for solving the equation $Ax = b$, unless an $A$-conjugate pair $(U, V)$ can be found. In this section we shall describe an algorithm for conjugating a set of linearly independent vectors with respect to another set of vectors. By varying the choice of the vectors to be conjugated and using the results in Algorithm 2.2, one may obtain various finitely terminating iterations for solving linear equations. Moreover, the conjugate pairs so obtained are related to standard matrix decompositions.

The idea of the conjugation technique is simple. Given nonsingular matrices $V$, $A$, and $P$, we attempt to determine $u_k$ as a linear combination of $p_1, p_2, \ldots, p_k$ in such a way that $U$ and $V$ are $A$-conjugate. We shall call this process the $A$-conjugation of $P$ with respect to $V$.

To determine when conjugation can be carried out, note that the process is equivalent to finding an upper triangular matrix, which we shall denote by $S^{-1}$, such that $U = PS^{-1}$. The $A$-conjugacy of $(U, V)$ requires that

$$L = V^H AU = V^H APS^{-1}$$

be lower triangular. In other words $V^H AP = LS$ must be factorizable into the produce of a lower triangular matrix and an upper triangular matrix ($V^H AP$ has an "LU factorization"). Since $V$, $A$, and $P$ are nonsingular, the matrix $S$ is uniquely determined up to the scaling of its rows (see [5, §1.4),
which implies that U is uniquely determined up to the scaling of its columns. We summarize these results in the following theorem.

**Theorem 3.1.** Let V, A, and P be nonsingular. Then a necessary and sufficient condition that P can be A-conjugated with respect to V is that $V^H A P$ have an LU factorization. In this case the conjugate vectors so obtained are unique up to scaling.

A reasonably efficient conjugation algorithm may be derived as follows. From the equation

$$P = US,$$

it follows that

$$p_k = U^{k-1} s_k + \sigma_k u_k,$$

where

$$s_k = (\sigma_{1k}, \sigma_{2k}, \ldots, \sigma_{k-1,k})^T.$$

Now the conjugacy of U and V require that

$$(V^{k-1})^H A U^{k-1} = L^{k-1}$$

be lower triangular and that

$$(V^{k-1})^H A u_k = 0.$$
Hence upon multiplying (3.1) by $(V^H A)^{-1}$, we obtain

$$(3.2) \quad L^{-1} s_k = (V^H A)^{-1} \quad \text{where} \quad \sigma_{kk} \neq 0$$

Thus $s_k$ may be determined by solving the lower triangular system. The vector $u_k$ may be determined from (3.1), where $\sigma_{kk} \neq 0$ is chosen to give $u_k$ some predetermined scaling.

**Algorithm 3.2.** Let $V, A, P \in \mathbb{C}^{n \times n}$ be nonsingular, and let $V^H A P$ have an LU factorization. The following algorithm $A$-conjugates $P$ with respect to $V$, returning the conjugated vectors as the columns of $U = (u_1, u_2, \ldots, u_n)$

1. For $k = 1, 2, \ldots, n$
   1. $s_k = (L^{-1}) s_k^T = (V^H A P)^{-1}$
   2. $u_k = \sigma_{kk}^{-1} (p_k - U^T s_k)$, $\sigma_{kk} \neq 0$.
   3. $L^{-1} u_k = (V^H A U)^{-1}$

Of course when $k = 1$, statement 1.1 is skipped and $u_1$ is determined as a scalar multiple of $p_1$.

An important feature of the conjugation technique is that the vectors $v_k, v_{k+1}, \ldots, v_n$ are not needed to determine $u_1, u_2, \ldots, u_k$. This means that the choice of $v_k$ can be deferred until after $u_k$ has been computed, and thus can be made to depend on $u_1, u_2, \ldots, u_k$.

We shall now consider some of the algorithms that may be obtained by varying $V$ and $P$ in the conjugation algorithm. Each choice leads to a well-known matrix decomposition and it is convenient to list the choices by the decompositions they determine.
1. LU decomposition of $A$. There are two choices of $V$ and $P$ that lead to the LU decomposition of $A$ into the product of a lower triangular matrix and an upper triangular matrix. The first choice is $P = V = I$. In this case $U = IS^{-1}$ is upper triangular, and

$$L = V^HAU = AU$$

is lower triangular. Hence

$$A = LS$$

is the required decomposition. In this case the conjugation algorithm is related to various methods for the triangular factorization of a matrix.

The second choice is $P = I$ and $V = U$. Again $U$ is upper triangular and

$$A = (S^H L)S$$

is the required decomposition. When $A$ is Hermitian, $L = U^HAU$ is also Hermitian, and hence diagonal. This is the usual conjugate direction algorithm, whose connection with the LU factorization of $A$ has already been pointed out by Hestenes and Stiefel [3].

2. LU decomposition of $A^HA$. For this method take $P = I$ and $V = A$. Since $V^HA = A^HA$ is positive definite and always has an LU decomposition, the conjugating algorithm can always be carried out for this choice of $P$ and $U$. The matrix $U = S^{-1}$ is upper triangular, and from the conjugacy conditions

$$A^HA = LS,$$

which exhibits an LU factorization of $A^HA$. 


3. Orthogonalization methods and the QR decomposition. This class of methods has been treated in detail by Householder [4]. Take $P$ arbitrary and $V = AU$. Since $P^H A^H A P$ is positive definite, it can be written in the form

$$P^H A^H A P = S^H S,$$

where $S$ is upper triangular [5, §1.4]. Hence with $U = PS^{-1}$, $V^H A P$ has an LU decomposition and the conjugation algorithm can be carried out. Moreover

$$I = S^{-H} P^H A^H A P S^{-1} = U^H A^H A U = V^H V,$$

so that the columns of $V$ are orthonormal. In addition $\mathcal{R}(V) = \mathcal{R}(A)$, and the algorithm can be used to solve least squares problems, as was pointed out in the §2.

If $P = I$, then

$$A = VS,$$

which exhibits the QR factorization of $A$ into the product of a unitary matrix and an upper triangular matrix. If the conjugation algorithm is augmented to include the explicit calculation of $V$, then it becomes essentially the modified Gram-Schmidt method.

4. Reduction by similarity transformations to Hessenberg form. The columns of $P$ are taken to be the vectors in the Krylov sequence defined by

$$p_i = A^{-1} p_{i-1} \quad (i = 1, 2, \ldots, n).$$
Necessary and sufficient conditions that \( P \) be nonsingular are that the Jordan form of \( A \) contains precisely one block for each distinct eigenvalue of \( A \) (\( A \) is nonderogatory) and that in the coordinate system associated with the Jordan form of \( A \) the vector \( p_1 \) has nonzero components in the principal vectors of maximal height.

Now from (3.3),

\[
p_{i+1} = Ap_i \quad (i = 1, 2, \ldots, n-1),
\]

and since the \( p_1 \) form a basis for \( n \)-space

\[
Ap_{n+1} = \gamma_1 p_1 + \gamma_2 p_2 + \ldots + \gamma_n p_n
\]

for some constants \( \gamma_1, \gamma_2, \ldots, \gamma_n \). It follows that

\[
(3.4) \quad AP = PC,
\]

where \( C \) is the companion matrix

\[
C = \begin{pmatrix}
0 & 0 & \cdots & 0 & \gamma_1 \\
1 & 0 & \cdots & 0 & \gamma_2 \\
0 & 1 & \cdots & 0 & \gamma_3 \\
\vdots & \vdots & \ddots & \vdots \\
0 & 0 & \cdots & 1 & \gamma_n
\end{pmatrix}.
\]

If \( V \) is chosen arbitrarily and \( P \) is conjugated with respect to \( V \), then the resulting \( U \) is equal to \( PS^{-1} \) where \( S \) is upper triangular. From (3.4)
and since $C$ is upper Hessenberg and $S$ is upper triangular, $H$ is also upper Hessenberg.

5. Biconjugation and reduction to tridiagonal form. Let the Hessenberg conjugation of $P$ just described be carried out with respect to any matrix $Q$. Then

$$Q^H AU = L$$

and if we define $V = QL^{-H}$, then

$$V^H AU = I.$$ 

Thus not only is $(U,V)$ $A$-conjugate, but $(V,U)$ is $A^H$-conjugate.

Now if $Q$ is generated by a Krylov sequence on $A^H$,

$$Q = (q_1, Aq_1, \ldots, (A^H)^{n-1} q_1),$$

then

$$K = V^{-1} A^H V$$

is upper Hessenberg. But

$$H = U^{-1} AU = (V^H AU)(U^{-1} AU)(U^{-1} A^{-1} V^{-H}) = V^H AV^{-H} = K^H.$$ 

Thus $H = K^H$ is both upper and lower Hessenberg, and hence $H$ is tridiagonal.
In principle it is possible to apply the conjugating algorithm directly to $P$ and $Q$ to yield the biconjugate pair $(U,V)$. However the fact that $U^{-1}AU$ and $V^HAU^{-1}$ are tridiagonal implies that the columns of $U$ and $V$ each satisfy three term recurrences, and the coefficients of these recurrences may be calculated by using the biconjugacy relation $V^HAU = I$. Specifically if the first columns of the matrices on each side of the equality

\begin{equation}
AU = UH
\end{equation}

are calculated, the result is

\[ Au_1 = \eta_{11}u_1 + \eta_{21}u_2. \]

Since $v_1^HAu_2 = 0$ and $v_1^HAu_1 = 1$,

\[ \eta_{11} = v_1^HAu_2. \]

Thus $u_2$ may be calculated in the form

\[ \eta_{21}u_2 = Au_1 - \eta_{11}u_2, \]

where $\eta_{21} \neq 0$ is arbitrary. Generally, from (3.5)

\[ Au_k = \eta_{k-1,k}u_{k-1} + \eta_{kk}u_k + \eta_{k+1,k}u_{k+1}. \]

Hence

\[ \eta_{k+1,k}u_{k+1} = Au_k - \eta_{kk}u_{k+1} - \eta_{k-1,k}u_{k-1}, \]

where $\eta_{k+1,k} \neq 0$ is arbitrary and
\[ n_{kk} - v_k^H A u_k, \]
\[ n_{k-1,k}^H = v_{k-1}^H A u_k. \]

Similarly from the equation \( A^H v = vH \), it follows that the columns of \( V \) may be calculated by the recurrences

\[ n_{12} v_2 = A^H v_1 = n_{11} v_1, \]
\[ n_{k,k+1} v_{k+1} = A^H v_k - n_{kk} v_k + n_{k,k-1} v_{k-1}, \]

where \( n_{kk} \) and \( n_{k,k+1} \) are defined as above and \( n_{k,k+1} \) is chosen so that

\[ v_{k+1}^H A u_{k+1} = 1. \]

The vectors \( u_k \) and \( v_k \) can be generated simultaneously. At no stage is it necessary to retain more than two of the vectors \( u_k \) and two of the vectors \( v_k \), which suggests that the algorithm may find application to large sparse linear systems.

When \( A \) is Hermitian and \( u_1 = v_1 = r_1 \), the above method, combined with Algorithm 2.2, is essentially the method of conjugate gradients [3], with some differences in scaling.

The general method is closely related to the Lanczos biorthogonalization algorithm [6]. In fact since \( U^{-1} = V^H A \), it follows that the tridiagonal matrix \( H \) is the one that would be obtained by biorthogonalizing the columns of the matrices \( P \) and \( A^H Q \). It should be noted that algorithms of this type tend to suffer from numerical instabilities. For a good discussion of the practical use of the Hessenberg and biorthogonalization algorithms, see [7, Ch. 6].
6. **Reduction to bidiagonal form.** The decomposition treated here seems not to originate with the conjugating algorithm. Rather it is based on the fact that if \( v_1 \) is any vector of Euclidean length unity, then there are unitary matrices \( U \) and \( V \) such that the first column of \( V \) is \( v_1 \) and the matrix

\[
L = V^H A U
\]

is lower bidiagonal (lower triangular and upper Hessenberg). The matrices \( U \) and \( V \) of course form an \( A \)-conjugate pair.

The columns of \( U \) and \( V \) may be generated successively by an algorithm which may be derived in much the same way as the biconjugation algorithm. From the equation

\[
UL^H = A^H V
\]

it follows that

\[
\lambda_{11} u_1 = A^H v_1,
\]

where \( \lambda_{11} \) is chosen to make \( u_1 \) of length unity. Generally

\[
\bar{\lambda}_{kk} u_k = A^H v_k - \bar{\lambda}_{k,k-1} u_{k-1},
\]

where orthonormality requires that

\[
(3.6) \quad \lambda_{k,k-1} = v_k^H A u_{k-1},
\]

and \( \lambda_{kk} \) is chosen so that \( u_k \) is of length unity.
From the equation

\[ AU = VL \]

it follows that

\[ \lambda_{k+1,k} v_{k+1} = A u_k - \lambda_{kk} v_k. \]

The orthonormality conditions for \( V \) give the same values for the \( \lambda \)'s as (3.6) and (3.7). Again the reduction can proceed stepwise and only the most recent vectors need be stored. This reduction was proposed by Golub and Kahan [2].

If \( v_1 \in \mathcal{R}(A) \), then \( \mathcal{R}(V) = \mathcal{R}(A) \), and the algorithm can be used to solve least squares problems.
References

1. L. Fox, H. D. Huskey, and J. H. Wilkinson, Notes on the solution of algebraic linear simultaneous equations, Quart. J. Mech. Appl. Math. 1 (1948) 149-173.

2. G. H. Golub and W. Kahan, Calculating the singular values and pseudo inverse of a matrix, J. SIAM Ser. B. Numer. Anal. 2 (1965) 205-224.

3. Magnus R. Hestenes and Eduard Stiefel, The method of conjugate gradients for solving linear systems, J. Res. Nat. Bur. Standards 49 (1952) 409-436.

4. A. S. Householder, Terminating and nonterminating iterations for solving linear systems, J. Soc. Indust. Appl. Math. 3 (1955) 67-72.

5. The Theory of Matrices in Numerical Analysis, Blaisdell, New York, 1964.

6. C. Lanczos, An iteration for the solution of the eigenvalue problem of linear differential and integral operators, J. Res. Nat. Bur. Standards 45 (1950) 164-206.

7. J. H. Wilkinson, The Algebraic Eigenvalue Problem, Clarendon, Oxford, 1965.