Semi-Conjugate Direction Methods for Real Positive Definite Systems*

J.Y. Yuan[1], Gene H. Golub[2], Robert J. Plemmons[3], and W. A. G. Cecílio[4]

March 5, 2003

[1] Departamento de Matemática - UFPR
Centro Politécnico, CP: 19.081
CEP: 81531-990, Curitiba, Paraná, Brazil

[2] Department of Computer Science
Stanford University
Stanford, CA94305-9025, USA

[3] Department of Computer Science
Wake Forest University
Winston-Salem, NC27109, USA

[4] Departamento de Matemática
Pontifícia Universidade Catórica do Paraná
Rua Imaculada Conceção, 1155
80215-901, Curitiba, PR, Brazil

Abstract

In this preliminary work, left and right conjugate direction vectors are defined for nonsymmetric, nonsingular matrices A and some properties of these vectors are studied. A left conjugate direction (LCD) method for solving general systems of

*The work of the first author was partially supported by CNPq, CAPES, Fundação Araucária, Brazil. The work of the second author was partially supported by DOE-FC02-01ER4177. The work of the third author was supported in part by the Air Force Office of Scientific Research under grants F49620-02-1-0107, and by the Army Research Office under grant DAAD19-00-1-0540. Some of the paper was written when the first author was visiting Stanford University and Wake Forest University.

linear equations is proposed. The method has no breakdown for real positive definite systems. The method reduces to the usual conjugate gradient method when A is symmetric positive definite. A finite termination property of the semi-conjugate direction method is shown, providing a new simple proof of the finite termination property of the conjugate gradient methods. Some techniques for overcoming breakdown are suggested for general nonsymmetric A. The connection between the semi-conjugate direction method and LU decomposition is established. The semi-conjugate direction method is successfully applied to solve some sample linear systems arising from linear partial differential equations, with attractive convergence rates. Some numerical experiments show the benefits of this method in comparison to well-known methods.

Key Words: *Left conjugate direction vectors, left conjugate direction method, semi-conjugate direction method, LU decomposition, Gaussian elimination, conjugate gradient method, solution of nonsymmetric linear systems, real positive definite system.*
AMS Subject Classification 65F10

1 Introduction.

The conjugate gradient method for solving symmetric and positive definite systems of linear equations

$$Ax = b, \tag{1.1}$$

where $A \in \mathbf{R}^{n\times n}$, is quite attractive, see e.g. [12, p. 520]. A short recurrence relation for computing the iteration vectors can be derived for certain nonsymmetric systems, for example the BiCG method [15]. But there is evidently no short recurrence relation [9] that corresponds to the minimization of any norm. Research on nonsymmetric system solvers has attracted much attention in recent years. Several effective Krylov subspace methods have been established to solve nonsymmetric systems, for example various generalization of the conjugate gradient method [1, 6], GMRES [15], QMR [10] and some of their variants [11, 14]. Generally, Krylov subspace methods look for some basis of the Krylov subspace such that the method still keeps the finite termination property or/and short recurrence for nonsymmetric A. In fact, the conjugate gradient method finds a basis with which the exact solution of the system can be determined by a diagonal system, thanks to the $A-$conjugate orthogonality property when A is symmetric and positive definite. For the nonsymmetric case, the usual $A-$conjugate orthogonality does not hold when A is real positive definite. In general, after obtaining a basis, we can determine the solution over the basis by solving one dense linear system, but this procedure has the same difficulty as the original problem. The methods above find special bases for the Krylov subspace such that one can obtain the solution by cheaply solving a linear system. For example, the GMRES method approaches the solution by minimizing the residual, and the QMR method by minimizing the generalized residual in part. The GMRES method solves a least squares problem, whereas the QMR method solves a simplified one at each step.

From this point of view, we cannot find a basis for the Krylov subspace such that the exact solution can be determined by solving one diagonal system for nonsymmetric systems, but it is possible to find a special basis of the Krylov subspace where the exact solution can be determined by some easily solved systems. Certainly the solution of triangular systems is cheaper than that of solving general least squares problems and other dense systems. The key problem is how to find such a basis. The main work of this paper is to develop this type of method for solving nonsymmetric linear systems.

We introduce the concepts of left conjugate direction vectors and right conjugate direction vectors (see Definitions 2.1 and 2.2 in Section 2), which we called semi-conjugate direction vectors. The new concepts coincide with the concept of the conjugate gradient vector whenever A is symmetric and positive definite. We prove that the set of left (or right) conjugate direction vectors forms a basis of the vector space and the exact solution of the nonsymmetric system (1.1) can uniquely be represented in this basis. The benefit of this concept is that the exact solution can be uniquely determined in this basis by solving triangular systems. We call the method the **Semi-Conjugate Direction Method (SCD)**. Here we mainly discuss the Left Conjugate Direction (LCD) Method. The new method keeps some key properties of the conjugate gradient method, for example the finite termination property. Of course, the new method loses some good properties of the conjugate gradient method, such as $A-$orthogonality and short recurrence relations to create the set of left conjugate direction vectors. In our derivation of the new method we also give a new simple proof of the finite termination property of the conjugate gradient method which may be useful for instructional purposes.

Unlike the conjugate gradient method, we cannot arbitrarily choose the initial left conjugate direction direction vector p_1 for the left conjugate direction method. For symmetric and positive definite systems, every choice of $p_1 \neq 0$ can guarantee $p_k^T A p_k > 0$, but this is not true for nonsymmetric or symmetric but indefinite A. Therefore, we must choose p_1 such that $p_k^T A p_k \neq 0$ for $k = 1, \ldots$, which is still an open problem for the general nonsymmetric case. Note that the LCD concept is the special case of the Stewart's conjugate gradient direction [17] which however is the special case of our SCD concept. Here we concentrate our attention on real positive definite systems. The method can easily be generalized to general nonsymmetric systems. But its stability is not guaranteed.

The outline of this paper is as follows. For general nonsymmetric matrices A, the concepts of the left conjugate direction vectors and right conjugate direction vectors are introduced in Section 2. We study the properties of these vectors in the same section. In Section 3, we propose the Left Conjugate Direction Method for solving nonsymmetric linear systems. We prove the finite termination property and truncation property of the new method as well. In particular we give the explicit form of summation of rank one matrices for the inverse matrix of A. In Section 4, we develop an algorithm to generate the set of left conjugate direction vectors of a real positive definite A and provide some properties of the left conjugate direction algorithm. The determination of left conjugate direction vectors may have breakdown problems for general nonsymmetric matrices, and does not work at all for skew-symmetric matrices A. Some remedies to overcome the breakdown problem are suggested. Hestenes and Stiefel established the connection between the conjugate gradient method and Gaussian Elimination for symmetric and positive definite matrices in 1952 [13, p. 426]. Here, we give a new simple proof for their result and also establish the connection between the left conjugate direction method and the LU decomposition in Section 5. Through this connection, the left conjugate direction method executes the LU decomposition for nonsingular matrices if it runs n steps with a certain choice of p_1. On the other hand, the row vectors of the unit lower triangular matrix given by the LU decomposition form the left conjugate direction vectors for the left conjugate direction method. Finally, in Section 6 we apply this new method to solve the approximation to certain linear partial differential equations. The convergence behavior of this new method appears very attractive. We conclude this paper with some comments.

After the completion of this paper, one of the authors with Dai [8] has shown the existence of the left conjugate direction vectors for all nonsingular matrices except skew-symmetric matrices. They have suggested augmented technique to overcome the breakdown problem for general nonsymmetric systems, and proposed limited-memory left conjugate direction method. Recently Silva, Raupp and Almeida [16] have applied the LCD method to solve linear systems arising from Petrov-Galerkin finite element method for the thermal pollution problem, and compared with traditional methods. From their comparisons, the LCD method seems to be very promising.

2 Left and Right Conjugate Direction Vectors

We begin by giving the definitions and some basic properties of left and right conjugate direction vectors.

Definition 2.1 *Vectors $p_1, p_2, \ldots, p_n \in \mathbf{R}^n$ are called* **left conjugate direction vectors** *(LCD) of an $n \times n$ real nonsingular matrix A if*

$$\begin{cases} p_i^T A p_j = 0 & \text{for} \quad i < j, \\ p_i^T A p_j \neq 0 & \text{for} \quad i = j, \end{cases} \tag{2.1}$$

that is,

$$P^T A P = L = (\,\boxed{\diagdown}^{\,O}\,),$$

where $P = [p_1, p_2, \ldots, p_n]$.

Definition 2.2 *Vectors $p_1, p_2, \ldots, p_n \in \mathbf{R}^n$ are called* **right conjugate direction vectors** *(RCD) of an $n \times n$ real nonsingular matrix A if*

$$\begin{cases} p_i^T A p_j = 0 & \text{for} \quad i > j \\ p_i^T A p_j \neq 0 & \text{for} \quad i = j, \end{cases} \tag{2.2}$$

that is,

$$P^T A P = U = (\,{}_{O}\boxed{\diagdown}\,),$$

where $P = [p_1, p_2, \ldots, p_n]$.

Remark 2.1 *If p_1, p_2, …, p_n are left conjugate direction vectors of A, then they are right conjugate direction vectors of A^T, and also p_n, p_{n-1}, …, p_1 are right conjugate direction vectors of A.*

Definition 2.3 *Vectors* $p_1, p_2, \ldots, p_n \in \mathbf{R}^n$ *are called* **conjugate gradient vectors** *(CG) of an* $n \times n$ *real nonsingular matrix* A *if*

$$\begin{cases} p_i^T A p_j = 0 & \text{for} \quad i \neq j \\ p_i^T A p_j \neq 0 & \text{for} \quad i = j, \end{cases} \tag{2.3}$$

that is,

$$P^T A P = D = ({}_{o}\diagdown^{o}),$$

where $P = [p_1, p_2, \ldots, p_n]$.

Definition 2.4 *Vectors* $p_1, p_2, \ldots, p_n \in \mathbf{R}^n$ *are called* **Semi-conjugate direction vectors** *(SCD) of* A *if they are LCD vectors or RCD vectors of* A.

Remark 2.2 *If* A *is symmetric and nonsingular, then we observe that the left conjugate direction vectors of* A *are also right conjugate direction vectors of* A. *In this case, we call the vectors conjugate gradient vectors of* A.

Lemma 2.1 *Let* $A \in \mathbf{R}^{n \times n}$ *be nonsingular, and* $\{p_1, p_2, \ldots, p_n\} \subset \mathbf{R}^n$ *be left (or right) conjugate direction vectors of* A. *Then* $p_1, p_2, \ldots, p_n$ *are linearly independent.*

Proof: Assume that $P = [p_1, p_2, \ldots, p_n]$. By the definition, there is

$$P^T A P = T$$

where T is lower triangular matrix and $t_{ii} \neq 0$. Then,

$$\det(P^T A P) = \det(T) \neq 0$$

which implies $\det(P) \neq 0$. Therefore, $\{p_1, \ldots, p_n\}$ are linearly independent.¶

Remark 2.3 *Let* $A \in \mathbf{R}^{n \times n}$ *be nonsingular, and* $\{p_1, p_2, \ldots, p_n\}$ *be left (or right) conjugate direction vectors of* A. *For any fixed vector* $x_0 \in \mathbf{R}^n$, *every vector* $x \in \mathbf{R}^n$ *can uniquely be represented with a linear combination of* $\{p_1, p_2, \ldots, p_n\}$ *in the form*

$$x = x_0 + \sum_{i=1}^{n} \alpha_i p_i, \tag{2.3}$$

where each α_i $(i = 1, \ldots, n)$ is uniquely determined by $\{p_1, p_2, \ldots, p_n\}$ and x_0. With $x_0 = 0$, it follows from $Ax = b$ that

$$P^T APP^{-1}x = P^T b,$$

that is,

$$TP^{-1}x = P^T b.$$

Then,

$$T\alpha = P^T b$$

and

$$x = P\alpha.$$

Note that the SCD definition is slightly different from the definition of the conjugate direction given by Stewart in [17]. In terms of Stewart's definition, U and V are $A-$conjugate if $V^T AU$ is lower triangular. Of course Stewart's $A-$conjugate direction is the left conjugate direction when $U = V = P$.

3 The Left Conjugate Direction Method

Suppose that the solution of the system (1.1) is x^*, and $\{p_1, p_2, \ldots, p_n\}$ are left conjugate direction vectors of A. Then it follows that

$$x^* = x_0 + \sum_{i=1}^{n} \alpha_i p_i \tag{3.1}$$

for every fixed vector x_0, where α_i $(i = 1, 2, \ldots, n)$ are unknowns. From $Ax^* = b$ it follows that

$$Ax_0 + \sum_{i=1}^{n} \alpha_i Ap_i = b. \tag{3.2}$$

Let r denote the residual vector defined by

$$r = b - Ax^* = b - Ax_0 - \sum_{i=1}^{n} \alpha_i Ap_i = r_0 - \sum_{i=1}^{n} \alpha_i Ap_i. \tag{3.3}$$

Now we shall determine α_i, for $i = 1, \ldots, n$, such that $r = 0$. Since $p_1, \ldots, p_n$ are linearly independent, $r = 0$ if r is orthogonal to all p_i $(i = 1, \ldots, n)$, that is

$$p_i^T r = 0, \quad \forall i = 1, \ldots, n. \tag{3.4}$$

Since the p_i's are LCD vectors of A, from (3.4) we can obtain

$$\alpha_i = \frac{p_i^T r_{i-1}}{p_i^T A p_i} \tag{3.5}$$

where

$$r_i = b - Ax_i = r_{i-1} - \alpha_i A p_i, \tag{3.6}$$

$$x_i = x_0 + \sum_{k=1}^{i} \alpha_k p_k = x_{i-1} + \alpha_i p_i. \tag{3.7}$$

Therefore we have established the left conjugate direction method to solve (1.1). The formal algorithm can be given as follows.

ALGORITHM 3.1

1. *Input* x_0*,* A *and* b*;*
2. *Choose a left conjugate direction vector set* $\{p_1, \ldots, p_n\}$ *for* A*;*
3. *Calculate* $r_0 = b - Ax_0$*;*
4. *For* $k = 1$ *until stop do*

   $\alpha_k = \frac{p_k^T r_{k-1}}{p_k^T A p_k}$*, if* $p_k^T A p_k \neq 0$*, otherwise stop;*

   $x_k = x_{k-1} + \alpha_k p_k$*,*

   $r_k = b - Ax_k = r_{k-1} - \alpha_k A p_k$*.*

The total number of flops per iteration step of this algorithm is a matrix-vector multiplication plus $6n$. As is the case for the conjugate gradient methods, convergence in at most n iterations is guaranteed, as shown later, and preconditioning can be expected to reduce the number of iterations to an acceptable approximate solution considerably. Note that in the conjugate gradient method, the p_k are computed at the kth step, not before the iteration starts.

We show next that the left conjugate direction method has a finite termination property in the absence of roundoff errors.

Theorem 3.1 *Let $A \in \mathbf{R}^{n\times n}$ be nonsingular, and $\{p_1, \dots, p_n\}$ be a set of left conjugate direction vectors of A. Then the left conjugate direction method obtains the solution of the system (1.1) in at most n steps in the absence of roundoff errors.*

Proof: By Lemma 2.1 and remark 2.3, the solution x^* of the system (1.1) can be uniquely determined by $p_1, \dots, p_n$ with every fixed x_0 as follows

$$x^* = x_0 + \sum_{i=1}^{n} \alpha_i p_i.$$

The method obtains the solution if for some j, $r_j = 0$ where $j < n$. Suppose all $r_j \neq 0$ for $j = 1, 2, \dots, n$. Then

$$r_n = b - Ax_n = b - Ax_0 - \sum_{i=1}^{n} \alpha_i A p_i. \tag{3.8}$$

In the method, all α_i were decided such that r_n is orthogonal to all n linearly independent vectors $p_1, \dots, p_n$. Hence, the vector r_n must be null. Therefore x_n is the solution of the system (1.1).¶

Corollary 3.2 *If A is symmetric and positive definite the conjugate gradient method has a finite termination property.*

Proof: The conjugate gradient method is one special case of the left conjugate direction method.¶

Remark 3.1 *Let the sequence $\{x_k\}$ be generated by the left conjugate direction method. Then x_k is a truncated approximation of x^* in the subspace $x_0 + \mathrm{Span}\{p_1, p_2, \dots, p_k\}$, where x^* is the exact solution of the system (1.1).*

Remark 3.2 *The LCD method is different from other Krylov subspace methods, such as the QMR and GMRES methods, because we do not minimize the residual norm in a Krylov subspace.*

As an application of the Left Conjugate Direction Method, one can obtain a representation of the inverse of A in the form of a summation of rank one matrices.

THEOREM 3.3 *Let P denote the matrix given by $P = [p_1, \ldots, p_n]$ where $p_i\,(i = 1, \ldots, n)$ are left conjugate direction vectors. Then, for every nonsingular matrix A, A^{-1} can be expressed as*

$$A^{-1} = PT^{-1}P^T, \tag{3.9}$$

where T defined by $[T]_{ij} = p_i^T A p_j$ is a nonsingular lower triangular matrix.

Proof: It follows from Lemma 2.1 that P is nonsingular. From the definition of left conjugate direction vectors p_i, we have

$$P^T AP = T, \tag{3.10}$$

where T is triangular and nonsingular.¶

From Theorem 3.3 we have the following expression for A^{-1}:

$$A^{-1} = \sum_{i=1}^{n} p_i y_i^T \tag{3.11}$$

where

$$y_i = \frac{1}{p_i^T A p_i}[p_i - \sum_{j=1}^{i-1} p_j^T A p_i y_j]. \tag{3.12}$$

COROLLARY 3.4 *Assume that A is symmetric and positive definite. Then A^{-1} is given by*

$$A^{-1} = PD^{-1}P^T = \sum_{i=1}^{n} \frac{p_i p_i^T}{p_i^T A p_i}, \tag{3.13}$$

where

$$D = \mathrm{diag}(p_1^T A p_1, \ldots, p_n^T A p_n). \tag{3.14}$$

Note that the existence of the left conjugate direction vector was proved by Stewart in [17].

4 Determination of Left Conjugate Direction Vectors

The main problem for implementing the left conjugate direction method is how to find a set of linearly independent vectors $p_1, \dots, p_n$ which are left conjugate direction vectors of A. For a symmetric and positive definite matrix A, there is a short recurrence to compute the conjugate gradient vectors. Now we are interested in the general nonsymmetric case.

For the nonsymmetric case, there is no short recurrence to compute the left conjugate direction vectors. However, there is still a recurrence relation among $p_1, \dots, p_k$ and r_k to compute the left conjugate direction vector p_{k+1}.

Suppose

$$p_{k+1} = r_k + \sum_{i=1}^{k} \beta_i p_i. \tag{4.1}$$

By the definition of left conjugate direction vectors, for all $j = 1, \dots, k$

$$p_j^T A p_{k+1} = 0, \tag{4.2}$$

that is,

$$\begin{pmatrix} p_1^T A p_1 & 0 & 0 & \dots & 0 \\ p_2^T A p_1 & p_2^T A p_2 & \ddots & \ddots & 0 \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ p_k^T A p_1 & p_k^T A p_2 & \dots & \dots & p_k^T A p_k \end{pmatrix} \begin{pmatrix} \beta_1 \\ \vdots \\ \beta_k \end{pmatrix} = - \begin{pmatrix} p_1^T A r_k \\ \vdots \\ p_k^T A r_k \end{pmatrix}, \tag{4.3}$$

once p_1 is chosen so that $p_1^T A p_1 \neq 0$. It follows from (4.1) and (4.3) that the recurrence can be obtained as follows.

$$\begin{cases} q_0 = r_k \\ \beta_i = -\frac{p_i^T A q_{i-1}}{p_i^T A p_i} \\ q_i = q_{i-1} + \beta_i p_i \quad (i = 1, \dots, k) \\ p_{k+1} = q_k. \end{cases} \tag{4.4}$$

The total number of flops for generating p_{k+1} is $kn^2 + 3kn$. This is certainly a long recurrence formula. We now shall show $p_{k+1}^T A p_{k+1} \neq 0$ where p_{k+1} is defined by (4.4) for matrices with a positive definite symmetric part. In fact, this is obvious for a real positive matrix. It can only fail if $p_{k+1} = 0$.

Theorem 4.1 *Suppose that $A+A^T$ is positive definite. Let $\{p_1,\ldots,p_k\}$ be left conjugate direction vectors of A. Then p_{k+1} is well defined by (4.4), that is, the left conjugate vectors are well defined.*

Proof: In order to prove that the vector p_{k+1} is well defined, we must prove that $p_{k+1}^T A p_{k+1} > 0$ for p_{k+1} given by (4.4). We shall first prove that $r_k, p_1, \ldots, p_k$ are linearly independent, then show that $p_{k+1} \neq 0$. It follows immediately from positive definite $A + A^T$ that $p_{k+1}^T A p_{k+1} > 0$.

Suppose there exist $\gamma_0, \gamma_1, \ldots, \gamma_k$ such that

$$\gamma_0 r_k + \gamma_1 p_1 + \ldots + \gamma_k p_k = 0, \tag{4.5}$$

with all $\gamma_i \neq 0$ ($i = 0, 1, \ldots, k$) or at least $\gamma_0 \neq 0$ because $p_1, \ldots, p_k$ are linearly independent by Lemma 2.1. There is no more to do if all $\gamma_i = 0$. Otherwise, r_k can be linear combination of $p_1, \ldots, p_k$ which means the solution $x^* = x_k$ is in the subspace $x_0 + \mathrm{Span}\{p_1, \ldots, p_k\}$. However the hypothesis $r_k \neq 0$ means x_k is not the solution of the system (1.1) or the solution x^* is not in the subspace $x_0 + \mathrm{Span}\{p_1, \ldots, p_k\}$. Then we get a contradiction. Thus $r_k, p_1, \ldots, p_k$ are linearly independent.

Since $r_k, p_1, \ldots, p_k$ are linearly independent, $p_{k+1} \neq 0$ follows from (4.4) whenever $r_k \neq 0$. Then, p_{k+1} is well defined from $p_{k+1}^T A p_{k+1} > 0$.¶

Corollary 4.2 *Let A be symmetric and positive definite. Then the determination of p_{k+1} in (4.1) will be a short recurrence, that is, the $A-$conjugate vectors p_k in the conjugate gradient method can be determined by a short recurrence.*

Proof: Since A is symmetric and positive definite, take $p_1 = r_0 = b - Ax_0$ and (4.3) is a diagonal system. We obtain the final result by considering the right hand side of (4.3).¶

We now give the complete left conjugate direction method (or semi-conjugate direction method) as follows.

Algorithm 4.1

1. *Input x_0, A, b and Choose p_1 such that $p_1^T A p_1 \neq 0$;*
2. *Calculate $r_0 = b - Ax_0$;*
3. *For $k = 1$ until stop do*

3.1) $u_k^T = p_k^T A$,

$\gamma_k = u_k^T p_k$,

$\alpha_k = \frac{p_k^T r_{k-1}}{\gamma_k}$,

$x_k = x_{k-1} + \alpha_k p_k$,

$r_k = b - Ax_k = r_{k-1} - \alpha_k A p_k$;

3.2) *For* $i = 1, \ldots, k$

$p_{k+1} = r_k$,

$\beta_i = -\frac{u_i^T p_{k+1}}{\gamma_i}$,

$p_{k+1} = p_{k+1} + \beta_i p_i$.

The flop count per iteration step of Algorithm 4.1 is 2 matrix-vector multiplications, $k+2$ vector-vector multiplications, $k+2$ number-vector multiplications, $k+1$ divisions plus $k+2$ vector additions. The $A-$conjugate vectors will be obtained if A is real positive definite because $p_{k+1}^T A p_{k+1} > 0$ always in this case. From the system (4.3), we can get a short recurrence for obtaining the vector p_{k+1} since the system (4.3) is diagonal. In this sense, our proof of the finite termination of the conjugate gradient method is simple.

We now summarize and prove other properties of the left conjugate direction Algorithm 4.1.

THEOREM 4.3 *Suppose that in (1.1)* $A \in \mathbf{R}^{n \times n}$ *is nonsingular and nonsymmetric with positive definite symmetric part, and let* $\mathcal{L}_k = \mathrm{Span}\{p_1, \ldots, p_k\}$ *where* p_i *are left conjugate direction vectors of* A. *Then the left conjugate direction method has the following properties:*

1. *For any* $k \geq 1$,
$$\mathcal{L}_k = \mathrm{Span}\{p_1, r_1, \ldots, r_{k-1}\}.$$

2. *For any* $k \geq 1$,
$$\mathcal{L}_k = \mathrm{Span}\{\delta_0, \delta_1, \ldots, \delta_{k-1}\}$$
where $\delta_k = x_{k+1} - x_k$.

3. *The sequence* x_k *generated by the left conjugate direction Algorithm 4.1 is finite.*

4. *For any* $k < i$, $p_i^T A p_k \neq 0$ *and* $p_k^T A p_i = 0$.

5.

$$x_{k+1} = x_k + \alpha_k p_k,$$

where

$$\alpha_k = \frac{p_k^T r_{k-1}}{p_k^T A p_k}.$$

6.

$$r_{k+1} = r_k - \alpha_k A p_k.$$

7.

$$p_{k+1} = r_k + \sum_{i=1}^{k} \beta_i p_i,$$

where

$$q_0 = r_k,$$

$$\beta_i = -\frac{p_i^T A q_{i-1}}{p_i^T A p_i},$$

$$q_i = q_{i-1} + \beta_i p_i,\ (i = 1, \ldots, k).$$

8. $\mathcal{L}_k$ *is not necessarily a Krylov subspace. But* $\mathcal{L}_n = \mathbf{R}^n$*.*

9. The vectors p_j *for all* $j \leq i-1$ *are orthogonal to all linearly independent vectors* $r_i - r_{i-1}$*, or equivalently* $(r_i - r_{i-1}) \perp Span\{p_1, p_2, ..., p_{i-1}\}$*.*

Proof:

1. Property (1) is from (4.1) and a choice p_1 such that $p_1^T A p_1 \neq 0$. If A is symmetric and positive definite, and $p_1 = r_0$, then the results is the same as the conjugate gradient method, that is, $\mathcal{L}_k = \mathrm{Span}\{r_0, \ldots, r_{k-1}\}$.
2. This follows from (3.7), or $\delta_k = \alpha_k p_k$.
3. This is Theorem 3.1.
4. This follows from the definition of the left conjugate direction vectors (Definition 2.1) and (4.2).

5. This is from (3.7).

6. This follows from (3.7).

7. This is (4.4) from Theorem 4.1.

8. The conclusion comes from the determination of p_i $(i = 1, \ldots, n)$ and 1 or 2 in this theorem.

9. Since $r_i = r_{i-1} - \alpha_i A p_i$, multiplying the equation by p_j, we obtain

$$p_j^T r_i = p_j^T r_{i-1} - \alpha_i p_j^T A p_i.$$

By the definition of left conjugate direction vectors of A we have that $p_j^T A p_i = 0$ for all $j < i$. Then the inner product $p_j^T (r_i - r_{i-1}) = 0$.¶

It is clear that the left conjugate direction method is different from the conjugate gradient method in the following ways when A is not symmetric positive definite.

1. We cannot arbitrarily choose the initial left (or right) conjugate direction vector p_1.

2. $\mathcal{L}_k \neq \mathrm{Span}\{r_0, r_1 \ldots, r_{k-1}\}$ unless $p_1 = r_0$.

3. $r_0, \ldots, r_k$ are not necessarily orthogonal, but are linearly independent.

4. For any $p \in \mathcal{L}_k$, p and r_k are generally not orthogonal.

5. If $k \neq i$ then $\delta_i^T A \delta_k \neq 0$ is possible.

6. There is generally no short recurrence to determine the left conjugate vectors of A.

Remark 4.1 *For general nonsymmetric systems, the method might break down. The method fails for all skew-symmetric matrices A, since then $p_k^T A p_k = 0$ for all p_k. One remedy to overcome the problem is to use regularization, the Lanczos method or row permutation to change skew-symmetric structure.*

Another remedy is to update p_k by $\hat{p}_k = c p_k + p_{k+1}$ where

$$c = \begin{cases} 1 & \text{if } (p_k^T A p_k)(p_{k+1}^T A p_k) \geq 0 \\ 1 - (p_{k+1}^T A p_k)/(p_k^T A p_k) & \text{otherwise} \end{cases}$$

whenever p_{k+1} defined by (4.4) satisfying $p_{k+1}^T A p_{k+1} = 0$.

Also, note that if A is real, positive, $K^T A K$ is also real, positive if $\mathrm{rank}(K) = n$.

5 Connection between the LCD method and Gaussian Elimination

We begin by establishing some basic connections. This is followed by some examples. Here we consider only matrices A having a triangular factorization $A = L^{-1}U$, where L is unit lower triangular and U is upper triangular. Of course always this holds in the special case that A is symmetric positive definite.

LEMMA 5.1 *Let $P = [p_1, p_2, \ldots, p_n]$. Then $P^T AP = T$ nonsingular and lower triangular matrix if and only if $\{p_1, p_2, \ldots, p_n\}$ are LCD vectors.*

LEMMA 5.2 *Let $P = [p_1, p_2, \ldots, p_n]$. Then $P^T AP = T$ nonsingular and upper triangular matrix if and only if $\{p_1, p_2, \ldots, p_n\}$ are RCD vectors.*

THEOREM 5.3 *Let $A \in R^{n\times n}$ be nonsingular. Then, the Left Conjugate Direction method with special choices of p_1 is equivalent to Gaussian elimination.*

Proof: To prove the equivalence, we must show that from the unit lower triangular L given by the Gaussian Elimination, we can define the matrix $P = [p_1, \ldots, p_n]$ whose columns are left conjugate vectors of A, so $P^T AP$ is lower triangular by Lemma 5.1.

It follows from the Gaussian elimination process that $A = LU$ where L is unit lower triangular, and U is upper triangular. Also, the LU factorization is unique if there is no permutation.

Let $P = L^{-T}Q$, where $(e_1, \ldots, e_n)$ are canonical vectors, and

$$Q = \begin{pmatrix} e_n \\ e_{n-1} \\ e_{n-2} \\ \vdots \\ e_1 \end{pmatrix}$$

is a permutation matrix. Then, we have

$$P^T AP = (Q^T L^{-1})A(L^{-T}Q) = (Q^T L^{-1})LU(L^{-T}Q) = Q^T UL^{-T}Q$$

is lower triangular. The nonsingularity is obvious. Another proof can be seen in Example 5.1 and [5]¶

With $P = L^{-T}$, we can prove that the RCD method is equivalent to Gaussian elimination because $P^T AP$ is upper triangular. By Lemma 5.2, we have the following corollary.

Corollary 5.4 *The RCD method is equivalent to triangular decomposition for nonsymmetric matrices. In particular, the CG method is equivalent to triangular decomposition for symmetric and positive definite matrices.*

We can also consider the UL decomposition $A = UL$ where U is unit upper triangular and L is lower triangular. Then, taking $P = U^{-T}$, we can obtain that $P^T AP = LU^{-T}$ is lower triangular. By Lemma 5.1, we have shown that the LCD method is equivalent to the UL decomposition. With this connection, the LCD method executes the Gaussian elimination when p_1 is the last row of L^{-1} for $A = LU$. Of course, there are more choices of p_1 for the LCD method in Algorithm 4.1. The following examples will illustrate the connection. The numerical tests here were carried out using Matlab 6.0. To illustrate the proof of Theorem 5.3, we give examples with some artificial properties. The first matrix A is positive definite, the second one nonsymmetric with negative and complex eigenvalues, and the third is symmetric positive definite.

Example 5.1 *Let*

$$A = \begin{pmatrix} 24 & 1 & 2 \\ 15 & 19 & 6 \\ 12 & 6 & 26 \end{pmatrix}$$

By Gaussian Elimination, we obtain $A = LU$

$$U = \begin{pmatrix} 24 & 1 & 2 \\ 0 & 18.3750 & 4.7500 \\ 0 & 0 & 23.5782 \end{pmatrix}, \quad L^{-1} = \begin{pmatrix} 1 & 0 & 0 \\ -0.6250 & 1 & 0 \\ -0.3129 & -0.2993 & 1 \end{pmatrix}.$$

Choose

$$P = U^{-1}D = \begin{pmatrix} 1.0000 & -0.0417 & -0.0726 \\ 0 & 1 & -0.2585 \\ 0 & 0 & 1 \end{pmatrix},$$

where $D = \mathrm{diag}(24,\ 18.3750,\ 23.5782)$. *Then,*

$$P^T AP = DU^{-T}LD = \begin{pmatrix} 24 & 0 & 0 \\ 14 & 18.3750 & 0 \\ 6.3810 & 0.7500 & 23.5782 \end{pmatrix} = T$$

is lower triangular whose columns $\{p_1, p_2, p_3\}$ are left conjugate direction vectors.

EXAMPLE 5.2 *Consider nonsymmetric matrix A with negative and complex eigenvalues as follows*

$$A = \begin{pmatrix} 6 & 3 & 12 \\ 87 & 7 & 9 \\ 5 & 3 & 9 \end{pmatrix}.$$

By Gaussian Elimination, $A = LU$ where

$$U = \begin{pmatrix} 6 & 3 & 12 \\ 0 & -36.5000 & -165 \\ 0 & 0 & -3.2603 \end{pmatrix}, \quad L^{-1} = \begin{pmatrix} 1 & 0 & 0 \\ -14.5000 & 1 & 0 \\ -1.0320 & 0.0137 & 1 \end{pmatrix}.$$

Choose

$$P = L^{-T}Q = \begin{pmatrix} -1.0320 & -14.5000 & 1 \\ 0.0137 & 1 & 0 \\ 1 & 0 & 0 \end{pmatrix}.$$

It follows that

$$P^T AP = \begin{pmatrix} -3.2603 & 0 & 0 \\ -165.5000 & -36.5000 & 0 \\ 5.8493 & -84 & 6 \end{pmatrix} = T,$$

a lower triangular matrix.

EXAMPLE 5.3 *Let A be symmetric and positive definite given by*

$$A = \begin{pmatrix} 7 & 1 & 2 \\ 1 & 16 & 5 \\ 2 & 5 & 15 \end{pmatrix}.$$

By Gaussian Elimination, we can obtain

$$U = \begin{pmatrix} 7 & 1 & 2 \\ 0 & 15.8571 & 4.7143 \\ 0 & 0 & 13.0270 \end{pmatrix}, \quad L^{-1} = \begin{pmatrix} 1 & 0 & 0 \\ 0.1429 & 1 & 0 \\ 0.2857 & 0.2973 & 1 \end{pmatrix}.$$

Setting $P = L^{-T}Q$, *we have*

$$P = L^{-T}Q \quad = \begin{pmatrix} 1 & -0.1429 & -0.2432 \\ 0 & 1 & -0.2973 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{pmatrix}$$

$$= \begin{pmatrix} -0.2432 & -0.1429 & 1 \\ -0.2973 & 1 & 0 \\ 1 & 0 & 0 \end{pmatrix}$$

and

$$P^T AP = \mathrm{diag}(u_{33}, u_{22}, u_{11}) = \mathrm{diag}(13.0270,\ 15.8571,\ 7).$$

Then $\{p_1, p_2, p_3\}$ *are conjugate gradient vectors.*

It is of interest to compare the relative errors $\epsilon = \|x_i - x^*\|_2/\|x^*\|_2$, $x^* \neq 0$ where x_i was obtained by Algorithm 4.1 with $x_0 = 0$ and $p_1 = r_0$ for the *LCD* method, and using $x_0 = 0$ and p_i given by the choices of p_i in the proof of Theorem 5.3, which is Gaussian Elimination. Assume that $Ax = b$ where

$$A = \begin{pmatrix} 1 & 2 & 7 \\ 2 & 5 & 0 \\ -1 & 0 & 6 \end{pmatrix}, \quad \text{and} \quad b = \begin{pmatrix} 3 \\ 9 \\ 85 \end{pmatrix}.$$

The relative errors are given in the Table 1.

Table 1: Relative Errors

$\epsilon = \|x_i - x^*\|_2/\|x^*\|_2$	LCD Method	GE Method
$\epsilon = \|x_1 - x^*\|_2/\|x^*\|_2$	1.0000	1.0000
$\epsilon = \|x_2 - x^*\|_2/\|x^*\|_2$	1.0081	1.0349
$\epsilon = \|x_3 - x^*\|_2/\|x^*\|_2$	0.6415	0.9506
$\epsilon = \|x_4 - x^*\|_2/\|x^*\|_2$	$7.0094e - 016$	0

6 Numerical Experiments and Conclusions

We first provide an example where the semi-conjugate direction method is successfully applied to solve a linear systems arising from the discretization of a linear partial differential equations. Rapid convergence is obtained.

6.1 Numerical Experiments

EXAMPLE 6.1 *Consider the test problem derived by discretizing the elliptic partial differential equation*

$$-\Delta u + 2\delta_1 u_x + 2\delta_2 u_y - \delta_3 u = f \tag{6.1}$$

with constant coefficients δ_1, δ_2 *and* δ_3 *on the unit square* $\Omega = \{(x, y) : 0 \leq x, y \leq 1\}$, *and with boundary condition* $u(x, y) = 0$ *on* $\partial\Omega$. *The function* f *is chosen so that*

$$u(x, y) = xe^{xy} \sin(\pi x) \cos(\pi y)$$

solves (6.1).

This example was taken from [4]. We use symmetric finite differences on a uniform $(n + 2) \times (n + 2)$ grid, including boundary points, and the standard five-point stencil to approximate Δu to yield a linear system of $N = n^2$ equations for n^2 unknowns $u_{ij} = u(ih, jh)$ $(1 \leq i, j \leq n)$:

$$\begin{aligned}(4 - \delta_3 h^2)u_{ij} - (1 + \delta_1 h)u_{i-1,j} - (1 - \delta_1 h)u_{i+1,j} - (1 + \delta_2 h)u_{i,j-1} \\ -(1 - \delta_2 h)u_{i,j+1} = h^2 f_{ij},\end{aligned} \tag{6.2}$$

where $f_{ij} = f(ih, jh)$ and $h = \frac{1}{n+1}$.

We selected three cases with various value of n for the numerical tests, given as follows (see [4]).

1. Case I: $\delta_1 = 30$, $\delta_2 = 40$, $\delta_3 = 40$.
2. Case II: $\delta_1 = 60$, $\delta_2 = 80$, $\delta_3 = 40$.
3. Case III: $\delta_1 = 80$, $\delta_2 = 80$, $\delta_3 = 40$.

For the purpose of comparison, we also put Calvetti, Golub and Reichel's results (denoted CGR) [4] in Table 2. Here $r = \|r_{last}\|_2/\|r_0\|_2$ and $e = \|x - u\|_2/\|u\|_2$ and s is the time in seconds. All computations were done with the convergence tolerance 10^{-10} using Matlab 5.0. If we set the tolerance to 10^{-8}, all computations were much faster than those reported in Table 3, with almost the same relative errors e in the solution. In this section, the initial vector $x_0 = 0$.

Table 2: Numerical Tests

Case	N $N = n^2$	h	LCD				CGR	
			IT	r	e	CPU	IT	r
I	2500	1/51	107	3.6144e-11	0.0214	34.68(s)	1000	0.52e-4
I	1600	1/41	97	6.0154e-11	0.0201	17.67(s)		
I	900	1/31	78	5.0173e-11	0.0198	6.87(s)		
I	10000	1/101	235	7.7834e-11	0.0209	1009.16(s)		
II	1600	1/41	94	5.7140e-11	0.0111	27.86(s)	231	4.2e-11
II	2500	1/51	122	9.1393e-11	0.0102	42.59(s)	224	3.2e-11
II	10000	1/101	225	8.7524e-11	0.0105	900.9(s)	1000	1.6e-11
III	2500	1/51	119	5.6232e-11	0.0085	41.03(s)		
III	10000	1/101	224	7.7053e-11	0.0086	838.53(s)		

For the second example, we compared BiCGSTAB, QMR, GMRES, GMRES(1), GMRES(5) and GMRES(20) with tolerance convergence 10^{-10} using MATLAB 6.5 for Case III with $N = 2500(n = 50)$. Here the initial guess for x is zero, and p_1 was chosen as random vector for the LCD method. The convergence curves of relative residuals are given in Figure 1. Comparison results of iteration counts and CPU time are given in Table 3. Unfortunately we cannot give a flop count comparison because Matlab 6.5 does not offer such information. The LCD method has nice convergence properties for several different initial left conjugate direction vector p_1 in this case.

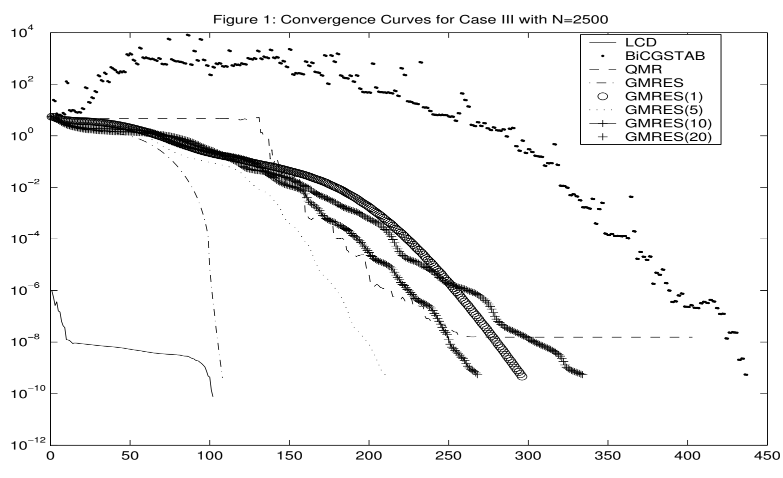
Note that the solution to (6.2) becomes more difficult for large δ_3 since if δ_3 is chosen too large the matrix is no longer positive and definite. Our method works very well for $\delta_i = 2000$ $(i = 1, 2, 3)$.

After finishing this paper, we became aware that Silva, Raupp and Almeida applied the left conjugate direction method to solve linear systems arising from Petrov-Galerkin finite element method for the thermal pollution problem, and gave parallel versions of the left conjugate gradient method in Fortran and compared it with the most traditional methods[16]. By their comparison, the left conjugate direction method seems to be very promising.

Table 3: Comparison Results for Case III with $N = 2500$

Method	IT	CPU	$\frac{\|x-u\|_2}{\|u\|_2}$	$\frac{\|x_{LCD}-x\|_2}{\|u\|_2}$
LCD	102	10.365	0.0093	0.0
QMR	403	101.784	0.0093	5.1803e-10
BiCGSTAB	437	96.808	0.0093	9.7320e-12
GMRES	107	61.547	0.0093	6.2846e-12
GMRES(1)	295	78.472	0.0093	5.1503e-12
GMRES(5)	211	77.610	0.0093	2.0303e-11
GMRES(10)	268	53.846	0.0093	1.5335e-11
GMRES(20)	334	57.4320	0.0093	4.0886e-11

Figure 1: **Convergence Curves for Case III with** $N = 2500$



6.2 Comments and Conclusions

We first provide a simple example to illustrate the importance of the choice of the first left conjugate direction vector p_1. Consider the following small nonsymmetric linear system $Ax = b$:

$$\begin{pmatrix} 1 & 4 & 1 \\ 5 & -1 & 2 \\ 3 & -2 & -1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \begin{pmatrix} 6 \\ 6 \\ 0 \end{pmatrix}.$$

Here, the exact solution is $x = (1, 1, 1)^T$. The left conjugate direction method does not work at all if we choose $p_1 = (\sqrt{5} - 2, 0, 1)^T$ because $p_1^T A p_1 = 0$ in this case. The left conjugate direction method obtains the solution with three steps if $p_1 = b$, and just one step if $p_1 = (-1, -1, -1)$. These numerical tests illustrate that the choice of p_1 is very important for convergence of the left conjugate direction method. In fact, the best choice of p_1 is any vector parallel to the solution $x = A^{-1}b$ whenever $x_0 = 0$, because the method can arrive at the exact solution in just one step in this case. On the other hand, the worst choice is to choose p_1 as any vector orthogonal to the solution $x = A^{-1}b$, because in this case the projection of p_1 is zero in the direction of the solution $x = A^{-1}b$. This means that the choice of p_1 cannot improve the initial choice of the approximate solution. In general, it may not be possible to obtain a very good choice of p_1 for real applications. We want to avoid the worst choice of p_1. Therefore, we must choose p_1 such that $p_1^T x = p_1^T A^{-1} b \neq 0$. We suggest choosing $p_1 = A^T z$ where $z^T b \neq 0$, for example $z = b$ when A is not very ill-conditioned. If A is very ill-conditioned, this choice leads to p_1 almost orthogonal to $x = A^{-1}b$, since $|z^t b|/(\|p_1\| \|A^{-1}b\|) \approx 0$.

Compared with the GMRES method, the work space is the same, but our computation at each step is cheaper because the new method solves triangular systems rather than least squares problems. Our numerical tests confirmed this advantage from Table 3 and Figure 1. Compared with the QMR method, the LCD method is much better because the QMR method did not reduce the error after 293 iterations (see Figure 1). The CPU time of the LCD method is much less than that of all compared methods.

The procedure of the left conjugate direction method can simply and easily be generalized to the right conjugate direction vector set to obtain the right conjugate direction method. The properties and convergence of the right conjugate direction method are the same as the left conjugate direction method. The algorithm is also very similar. We shall not discuss the details here.

Note that the left conjugate direction method is different from Stewart's conjugate direction method when $U = V$[17] in the sense of implementation. Our process is cheaper than his method because at each step Stewart's method needs two triangular system solutions to create one left conjugate direction vector. The proof of the equivalence of the LCD method and the LU decomposition in [17] is simpler than ours.

As mentioned earlier, this paper is expository in nature. A formal process for finding an effective left conjugate vector set for the general nonsymmetric or the symmetric indefinite case is left as a problem for future research.

Acknowledgment

The first author likes to give his sincere thanks to Professor Gene H. Golub for his very kind and helpful hospitality during his visit to Stanford University in Summer of 1997 and from August, 1998 to August, 1999, and to Professor Robert Plemmons for his very kind and helpful hospitality during his visit to Wake Forest University in Summer of 1997 and Spring of 1999. We would like to thank Dr. Yuhong Dai for some helpful comments, which led to Remark 2.1, Remark 2.2 and the breakdown examples. We also thank Robert Bridson for helpful comments on the draft. We thank referees for their very helpful comments and suggestions which improve the quality of the paper greatly.

References

[1] O Axelsson, *Conjugate gradient type methods for nonsymmetric and inconsistent systems of linear equations*, Linear Algebra and its Applications, 29(1980) 1-16.

[2] A. Berman and R. Plemmons, *Nonnegative Matrices in the Mathematical Sciences*, SIAM Classics Series, Philadelphia, 1994.

[3] D.L. Boley and G.H. Golub, *The nonsymmetric Lanczos algorithm and controllability*, Systems Control Lett., 16(1991) 97-105.

[4] D. Calvetti, G.H. Golub and L. Reichel, *Adaptive Chebyshev Iterative Methods for Nonsymmetric Linear Systems Based on Modified Moments*, Numer. Math., 67(1994) 21–40.

[5] Waleria A.G. Cecílio, *Connection between LCD method and Gaussian Elimination*, Master Dissertation, UFPR, Brazil, 2001.

[6] P. Concus and G.H. Golub, *A generalized conjugate gradient method for nonsymmetric systems of linear equations*, in *Computing Methods in Applied Sciences and Engineering (Lecture Notes in Economics and Mathematical Systems 134)*, eds. R. Glowinski and J.L. Lions, Springer, Berlin, 1976, pp56-65.

[7] Y.-H. Dai, *Private communication*, April, 1999.

[8] Y.-H. Dai and J.Y. Yuan, *Further Studies on the Semi-conjugate direction methods*, Technical Report, UFPR, Brazil, 2002.

[9] V. Faber and T. Manteuffel, *Necessary and sufficient conditions for the existence of a conjugate gradient method*, SIAM J. Numer. Anal., 21(1984) 352-362.

[10] R.W. Freund and N.M. Nachtigal, *QMR: a quasi-minimal residual method for non-Hermitian linear systems*, Numer. Math., 60(1991) 315-339.

[11] R.W. Freund, G.H. Golub and N.M. Nachtigal, *Iterative solution of linear systems*, Acta Numerica, Cambridge University Press, 1992, pp 57-100.

[12] G.H. Golub and C.F. Van Loan, *Matrix Computations*, 3rd. Ed., The Johns Hopkins University Press, Baltimore and London, 1996.

[13] M.R. Hestenes and E. Stiefel, *Methods of conjugate gradients for solving linear systems*, Journal of Research of the National Bureau of Standards, B49(1952) 409-436.

[14] Y. Saad, *Krylov subspace methods for solving large unsymmetric linear systems*, Math. Comput., 37(1981) 105-126.

[15] Y. Saad and M.H. Schultz, *GMRES: a generalized minimal residual algorithm for solving nonsymmetric linear systems*, SIAM J. Sci. Stat. Comput., 7(1986) 856-869.

[16] Renato S. Silva, Fernanda M.P. Raupp and Regina C. Almeida, em A Numerical Methodolgy to Solve Thermal Pollution Problems, IX Congresso Brasileiro de

Engenharia e Cincias Trmicas, Caxambu, Brazil, 2002 (http://www.lncc.br/proj-pesq/relpesq-02.html/32/2002).

[17] G.W. Stewart, *Conjugate direction methods for solving systems of linear equations*, Numer. Math. 21(1973)285-297.