A Lanczos - Conjugate Gradient algorithm and the Moore-Penrose pseudoinverse

Giovanni Fasano[a]

Technical Report INSEAN October 22th, 2004

[a] Istituto Nazionale per Studi ed Esperienze di Architettura Navale (INSEAN), Via di Vallerano 139, 00128 Roma, ITALY, E-mail: g.fasano@insean.it.
Istituto di Analisi dei Sistemi ed Informatica \A. Ruberti" - CNR, viale Manzoni 30, 00185 Roma, ITALY, E-mail: fasano@iasi.rm.cnr.it. URL: www.dis.uniroma1.it/
efasano

Abstract

This paper extends some theoretical properties of the Conjugate Gradient-type method FLR [Fas05], for iteratively solving indefinite linear systems of equations. The latter algorithm is a generalization of the Conjugate Gradient (CG) by Hestenes and Stiefel [HS52].
On one hand, here we carry out a complete relationship between algorithm FLR and the Lanczos process, in case of indefinite and possibly singular matrices. On the other hand we develop simple theoretical results for algorithm FLR, in order to construct an approximation of the Moore-Penrose pseudoinverse of an indefinite matrix. Our approach supplies theory for applications within nonconvex optimization.

Keywords : unconstrained optimization, Krilov subspace methods, planar conjugate gradient, Moore-Penrose pseudoinverse.

AMS subject classification: 90C30

1 Introduction

In this paper we consider the solution of the dense linear system

$$Ax = b, \tag{1}$$

where the symmetric matrix $A \in \mathbb{R}^{n \times n}$ is indefinite and possibly singular, $b \in \mathbb{R}^n$ and n is large. Many real large scale problems require the solution of linear system (1) and they often need the use of efficient solvers, along with easy and handable software packages. A great deal of iterative algorithms for solving linear system (1) provide us with useful and efficient tools [GV89]; nevertheless, the selection of the appropriate method is often a stiff problem for non-specialists.
In case Krylov subspace methods are considered [Gre97, SVdV00] and good preconditioners are adopted, the differences among methods become less relevant [Han98]. However, this trivially shifts the problem to the identification of a suitable general purpose preconditioner.

When problem (1) becomes ill-conditioned, the numerical treatment is more complicated and some regularization techniques, which use additional information for stabilizing the solution, are often advisable [Han98]. Moreover, optimization frameworks provide strong motivations for investigating the solution of possibly singular system (1).
In particular, consider the solution of nonlinear least squares problem

$$\min_{x \in \mathbb{R}^n} \frac{1}{2}\|r(x)\|^2, \qquad r : \mathbb{R}^n \to \mathbb{R}^m, \tag{2}$$

by means of the damped Gauss-Newton method [Bjo96]. Let $J(x) \in \mathbb{R}^{m \times n}$ be the Jacobian of vector function $r(x)$, at current point x. Then, at step k the latter method considers the linear approximation $r(x_k) + J(x_k)d_k$ of $r(x)$ at x_k, and computes d_k as a solution of the unconstrained subproblem

$$\min_{d \in \mathbb{R}^n} \|r(x_k) + J(x_k)d\|. \tag{3}$$

Then, the next iterate is $x_{k+1} = x_k + \alpha_k d_k$, where the steplength $\alpha_k \in \mathbb{R}$ is selected by a linesearch procedure [McC83]. Let $J^+(x_k)$ be the Moore-Penrose pseudoinverse of matrix $J(x_k)$ [CM79]: the choice $d_k = -J^+(x_k)r(x_k)$ among the solutions of (3) has a couple of remarkable advantages. It is invariant under linear transformation on x, and it is a descent direction for the objective function in (2) [Bjo96]. In particular, the latter property is used in [LS03, FLS04], where the CG is adopted to compute d_k, i.e. for equivalently solving the linear system $J^T(x_k)J(x_k)d = -J^T(x_k)r(x_k)$ (see also [Hes75]). Observe that in general the matrix $J^T(x_k)J(x_k)$ is rank deficient.

Another application within nonconvex optimization, which involves the solution of possibly singular system (1), is the Newton method for eigenvector computation. Suppose $\lambda \in \mathbb{R}$ is an approximate eigenvalue of the indefinite matrix $H \in \mathbb{R}^{n \times n}$, associated to eigenvector $v \in \mathbb{R}^n$. Then, a non-trivial solution x^* of the linear system $(H - \lambda I)x = 0$ yields an approximation of vector v. The Newton method is often the method of choice to this purpose and gives the iterate [WSS98]

$$x_{k+1} = x_k - (H - \lambda I)^{-1} r_k, \tag{4}$$

where $r_k = (H - \lambda I)x_k$, $x_0 \in \mathbb{R}^n$. Since (4) is not well defined, it is turned into iteration [GV89]

$$x_{k+1} = x_k - (H - \lambda I)^{+} r_k, \tag{5}$$

by introducing the Moore-Penrose pseudoinverse of $(H - \lambda I)$. Under suitable assumptions (5) is convergent to an approximation x^* of eigenvector v. Observe that the pseudoinverse $(H - \lambda I)^+$ is also an inner inverse, i.e. $(H - \lambda I)(H - \lambda I)^+(H - \lambda I) = (H - \lambda I)$, and that $r_k = (H - \lambda I)x_k$. Therefore, on large scale problems, iteration (5) may be solved as the equation

$$(H - \lambda I)(x_{k+1} - x_k) = -r_k, \tag{6}$$

and a Krylov based method may be adopted. Unfortunately, since matrix H is indefinite, the CG may fail. We consider in this paper a generalized CG method, and we prove that under suitable assumptions it provides the pseudoinverse solution of equation (6). An iteration similar to (5) is introduced when the Jacobi-Davidson method [SVdV96] is used, in place of Newton's method, for computing the eigenvector v.

The above examples, along with the low computational cost and the low memory-demand of CG-like methods, induced us to study and consider algorithm FLR in [Fas05], as a possible candidate for solving (1).
We also prove the complete theoretical relationship between algorithm FLR and the Lanczos process. Equivalently, under few assumptions, algorithm FLR is proved to generate, in exact arithmetic, the sequence of Lanczos vectors.

In the following sections we use the symbol $\|\cdot\|$ to denote the Euclidean norm for both a real n-dimensional vector and a real $n \times n$ matrix. We use the notation $x^T y$ for the inner product between vectors $x, y \in \mathbb{R}^n$, so that $x \perp y$ is equivalent to $x^T y = 0$. $0_{[m,n]}$ is the $m \times n$ matrix with all entries equal to zero. With $\mathcal{R}(A)$ and $\mathcal{N}(A)$ we respectively denote the range and the null space of the symmetric matrix $A \in \mathbb{R}^{n \times n}$. With $\mathcal{K}_i(v, A)$ we indicate the Krylov subspace $\mathrm{span}\{v, Av, \ldots, A^{i-1}v\}$ associated to vector $v \in \mathbb{R}^n$ and matrix $A \in \mathbb{R}^{n \times n}$. $Pr_W(v)$ indicates the projection of vector v onto the linear vector space W. Finally, $\lambda_m = \min_j |\lambda_j(A)|$ and $\lambda_M = \max_j |\lambda_j(A)|$, where $\lambda_j(A)$, $j \geq 1$, are the eigenvalues of the symmetric matrix A.

The paper is organized as follows: Section 2 deals with the description of few general preliminaries. Sections 3 and 3.1 provide some relevant features of the CG, when used for solving (1) and the coefficient matrix A is positive semidefinite. Sections 4 and 4.1 extend the results of Sections 3 and 3.1, to the application of planar algorithm FLR in [Fas05]. Here, under mild assumptions the latter algorithm is used to construct an approximation of the Moore-Penrose pseudoinverse A^+. Section 5 provides a noteworthy relation between algorithm FLR and the Lanczos process. Finally, Section 6 contains both conclusions and perspectives related to the treated subject.

Table 1: Algorithm CG for solving the linear system (1).

Step 1.	Set $k = 1$, $x_1 \in \mathbb{R}^n$, $r_1 = b - Ax_1$.
	If $r_1 = 0$, then STOP. Else, set $p_1 = r_1$.
Step k.	Compute $d_k = p_k^T A p_k$, $\alpha_k = r_k^T p_k / d_k$,
	$x_{k+1} = x_k + \alpha_k p_k$, $r_{k+1} = r_k - \alpha_k A p_k$.
	If $r_{k+1} = 0$, then STOP. Else,
	set $\beta_k = -p_k^T A r_{k+1} / d_k$, $p_{k+1} = r_{k+1} + \beta_k p_k$.
	Set $k = k + 1$ go to Step k.

Table 2: The Lanczos process applied to system (1).

Step 0.	$k = 0$, $v_0 = b \in \mathbb{R}^n$,
	$u_0 = 0$, $\delta_0 = \|b\|$.
Step k.	If $\delta_k = 0$, then STOP. Else, $u_{k+1} = v_k / \delta_k$.
	Set $k = k + 1$, $\gamma_k = u_k^T A u_k$,
	$v_k = (A - \gamma_k I) u_k - \delta_{k-1} u_{k-1}$
	$\delta_k = \|v_k\|$, go to Step k.

2 Some general results

In this section we introduce few general results for the solution of (1) which will be largely used in the sequel. Consider the CG-based algorithm FLR described in [Fas05] (see Table 3). The latter algorithm is a general planar method [Lue69, Hes80, LS91, DDS85, MC69] for solving (1), when A is indefinite; i.e. it avoids the possible pivot breakdown of the CG in the indefinite case, by introducing 2×2 pivot elements. We are concerned with proposing some new properties of algorithm FLR in case matrix A in (1) is singular. Tables 1 and 3 briefly recall both the CG and FLR methods for the convenience of the reader.

We remark that the Krylov based algorithm FLR is a generalization of the CG in case matrix A is indefinite. Indeed from Table 3, as long as the quantity d_k at step k is relatively large, a CG step is performed at step k_A. On the contrary, whenever d_k is relatively small the vector q_k is generated at step k_B, so that the search of the solution for (1) is detected over the 2-dimensional manifold $\mathrm{span}\{p_k, q_k\}$ (see also [BC94]).

Now, on one hand we aim at determining properties of algorithms CG and FLR in case matrix A is singular. Then, we study the relationship between the sets of orthogonal directions generated by the Lanczos process and algorithm FLR, when solving (1). To this end consider algorithms CG, FLR and the Lanczos process (Table 2), where without loss of generality we assumed $v_0 = b$ at Step 0 (see [S03] for a more general choice). Recalling the symmetry of matrix A, let either the first nonzero Lanczos vector u_1 or the first residual r_1 in algorithms CG and FLR be given by

$$\begin{array}{lll} u_1 = y + z; & y = Pr_{R(A)}(u_1); & z = Pr_{N(A)}(u_1); \\ r_1 = y + z; & y = Pr_{R(A)}(r_1); & z = Pr_{N(A)}(r_1). \end{array} \tag{7}$$

Then, the following general result holds:

Table 3: Algorithm FLR for solving the linear system (1).

Step 1.	Set $k = 1$, $x_1 \in \mathbb{R}^n$, $r_1 = b - Ax_1$. If $r_1 = 0$, then STOP. Else, set $p_1 = r_1$.
Step k.	Compute $d_k = p_k^T A p_k$; set $\epsilon_k > 0$. If $\lvert d_k \rvert \ge \epsilon_k \lVert p_k \rVert^2$, go to Step k_A. If $\lvert d_k \rvert < \epsilon_k \lVert p_k \rVert^2$, go to Step k_B.
Step k_A.	Set $a_k = r_k^T p_k / d_k$, $x_{k+1} = x_k + a_k p_k$, $r_{k+1} = r_k - a_k A p_k$. If $r_{k+1} = 0$, then STOP. Else, set $b_k = -p_k^T A r_{k+1} / d_k$ and $p_{k+1} = r_{k+1} + b_k p_k$. Set $k = k + 1$ go to Step k.

Step k_B.	If $k = 1$, then set $q_k = Ap_k$.
	If $k > 1$ and the previous Step is $(k-1)_A$, then
	set $\beta_{k-1} = -(Ap_{k-1})^T A p_k / d_{k-1}$ and $q_k = Ap_k + \beta_{k-1} p_{k-1}$.
	If $k > 1$ and the previous Step is $(k-2)_B$, then
	set $\hat{\beta}_{k-2} = -(Aq_{k-2})^T A p_k$ and $q_k = Ap_k + \hat{\beta}_{k-2}(d_{k-2} q_{k-2} - \delta_{k-2} p_{k-2}) / \Delta_{k-2}$.
	Compute $c_k = r_k^T p_k$, $\delta_k = p_k^T A q_k$, $e_k = q_k^T A q_k$, $\Delta_k = d_k e_k - \delta_k^2$ and
	$\hat{c}_k = (c_k e_k - \delta_k q_k^T r_k) / \Delta_k$, $\hat{d}_k = (d_k q_k^T r_k - \delta_k c_k) / \Delta_k$.
	Set $x_{k+2} = x_k + \hat{c}_k p_k + \hat{d}_k q_k$, $r_{k+2} = r_k - \hat{c}_k A p_k - \hat{d}_k A q_k$.
	If $r_{k+2} = 0$, then STOP. Else,
	compute $\hat{b}_k = -q_k^T A r_{k+2}$ and set $p_{k+2} = r_{k+2} + \hat{b}_k (d_k q_k - \delta_k p_k) / \Delta_k$.
	Set $k = k + 2$ go to Step k.

Lemma 2.1 *Given the symmetric matrix $A \in \mathbb{R}^{n\times n}$, let $P_i(\cdot)$ be a nonzero real polynomial of finite degree $i \geq 1$. Let $y_1, \dots, y_k$ be eigenvectors of matrix A associated to nonzero eigenvalues $\lambda_1, \dots, \lambda_k$ of A.*

1. *If vector y has nonzero orthogonal projection only on eigenvectors $y_{j_1}, \dots, y_{j_l}$, with $j_h \in \{1, \dots, k\}$, $h = 1, \dots, l$, then we have $P_i(A)y = 0$ only if $i \geq \hat{l}$, where $\hat{l} \leq l$ is the number of distinct eigenvalues out of the l eigenvalues associated to $y_{j_1}, \dots, y_{j_l}$.*

2. *The sequence $\{P_i(A)y\}$, which is dependent on index i, contains at most $\hat{l}$ linearly independent vectors.*

Proof.
As regards 1. let the vector y have nonzero orthogonal projection on the l eigenvectors $y_{j_1}, \dots, y_{j_l}$, then the vector $c \in \mathbb{R}^l$ exists such that $y = \sum_{h=1}^{l} c_{j_h} y_{j_h}$, $c_{j_h} \neq 0$, $h = 1, \dots, l$. From the symmetry of matrix A the orthogonal matrix $V \in \mathbb{R}^{n\times n}$ exists such that:

$$A = VDV^T; \qquad D = \mathrm{diag}\{\lambda_1, \dots, \lambda_k, 0_{[n-k]}\}; \qquad V = [y_1 \cdots y_k z_1 \cdots z_{n-k}]; \tag{8}$$

where $z_1, \dots, z_{n-k}$ are orthonormal eigenvectors associated to the zero eigenvalue. Thus, for any i

$$P_i(A) = V P_i(D) V^T; \tag{9}$$

and consequently $P_i(A)y$ is given by:

$$\begin{aligned} P_i(A)y &= V \begin{pmatrix} P_i(\lambda_1) & & & & & & \\ & \ddots & & & & & \\ & & P_i(\lambda_l) & & & & \\ & & & P_i(\lambda_{l+1}) & & & \\ & & & & \ddots & & \\ & & & & & P_i(\lambda_k) & \\ & & & & & & P_i(0_{[n-k;n-k]}) \end{pmatrix} \sum_{h=1}^{l} c_{j_h} V^T y_{j_h} \\ &= Vv; \end{aligned}$$

where $v \in \mathbb{R}^n$ and for $p = 1, \dots, n$

$$v_p = \begin{cases} c_p P_i(\lambda_p) & \text{if } p \in \{j_1, \dots, j_l\} \\ 0 & \text{otherwise.} \end{cases}$$

Since $c_{j_h} \neq 0$, for any $j_h \in \{j_1, \dots, j_l\}$, and V is nonsingular, $P_i(A)y = 0$ if and only if $P_i(\lambda_{j_h}) = 0$, $j_h \in \{j_1, \dots, j_l\}$. In particular this implies that all the $\hat{l}$ distinct eigenvalues in the set $\{\lambda_{j_1}, \dots, \lambda_{j_l}\}$ are roots of the polynomial $P_i(\lambda)$. Consequently, if $P_i(A)y = 0$ then $i \geq \hat{l}$.

As regards 2. let $\{\lambda_{j_1}, \dots, \lambda_{j_{\hat{l}}}\}$ be the distinct eigenvalues out of the l eigenvalues associated to eigenvectors $y_{j_1}, \dots, y_{j_l}$. Then, from the hypothesis

$$y = \sum_{h=1}^{l} c_{j_h} y_{j_h} = \sum_{\sigma=1}^{\hat{l}} \left(\sum_{j_h \in \Lambda_\sigma} c_{j_h} y_{j_h} \right) = \sum_{\sigma=1}^{\hat{l}} w_\sigma; \tag{10}$$

where $\Lambda_\sigma = \{j_h \in \{j_1, \dots, j_l\} \text{ s.t. } \exists \lambda_\sigma \in \mathbb{R}, Ay_{j_h} = \lambda_\sigma y_{j_h}\}$ and $w_\sigma = \sum_{j_h \in \Lambda_\sigma} c_{j_h} y_{j_h}$. Observe that $(\lambda_\sigma, w_\sigma)$ is an eigenpair of matrix A and eigenvectors $w_1, \dots, w_{\hat{l}}$ are independent, therefore from (10) $P_i(A)y \in \mathrm{span}\{w_1, \dots, w_{\hat{l}}\}$, for any $i \geq 1$. This implies that the sequence $\{P_i(A)y\}$ contains at most $\hat{l}$ linearly independent vectors, regardless of the choice of index $i \geq 1$. $\Box$

Remark 2.1 Observe that according with the definitions used in [S03], the integer ℓ of Lemma 2.1 is the grade of y with respect to matrix A, i.e. the lowest degree of the polynomial $P(A)$ such that $P(A)y = 0$. Therefore Lemma 2.1 states a relationship between the grade of y and the eigenpairs of matrix A. Furthermore, connections between the polynomial $P_\ell(A)$ and the minimal polynomial of matrix A were highlighted in [Hes75].

3 Issues on the CG when matrix A is singular

Consider the solution of linear system (1) by means of CG. A sequence of conjugate directions is generated, provided that matrix A is positive definite. We briefly recast a similar result when matrix A is positive semidefinite, using Lemma 2.1 (see also [Hes75]). Let matrix A be positive semidefinite and

$$\lambda_i > 0; \qquad i = 1, \ldots, k; \tag{11}$$
$$\lambda_i = 0; \qquad i = k+1, \ldots, n; \tag{12}$$

where $\{\lambda_i\}$ are the real eigenvalues of A. Thus, for any vector $p \in \mathbb{R}^n$, coefficients $c_i \in \mathbb{R}$, $i = 1, \ldots, k$, exist such that

$$p = z + \sum_{i=1}^{k} c_i y_i; \tag{13}$$

where z is the orthogonal projection of vector p onto the subspace $N(A)$, while y_i, $i = 1, \ldots, k$, are k orthonormal eigenvectors associated to the eigenvalues (11). From (11), (12), (13) and the symmetry of matrix A we obtain

$$p^T A p = \sum_{i=1}^{k} c_i^2 \lambda_i. \tag{14}$$

Thus, if matrix A is positive semidefinite then $p^T A p \neq 0$ if and only if $p \notin N(A)$. This implies that in the semidefinite case, the CG in Table 1 does not stop untimely as long as $p_i \notin N(A)$, $i \geq 1$, where p_i is the conjugate direction generated by the CG at step $i - 1$. In addition we have some further results:

Proposition 3.1 Let matrix A in (1) be positive semidefinite, and let r_1 in Table 1 satisfy (7) and the hypothesis of Lemma 2.1. If algorithm CG generates the mutually conjugate vectors $p_1, \ldots, p_\ell$, with $p_i \notin N(A)$, $i = 1, \ldots, \ell$, then the latter vectors are linearly independent.

(The proof of the above proposition trivially follows from Lemma 2.1 and the guidelines of the positive definite case). The statements of Lemma 2.1 and Proposition 3.1 yield the following result:

Theorem 3.1 Consider the linear system (1) and let matrix A be positive semidefinite. Let in the CG of Table 1

$$r_1 = y + z; \qquad y = Pr_{R(A)}(r_1); \; z = Pr_{N(A)}(r_1). \tag{15}$$

Suppose vector y has nonzero projection on the eigenvectors $y_{j_1}, \ldots, y_{j_l}$ of A, and only $\ell \leq l$ eigenvalues associated to $y_{j_1}, \ldots, y_{j_l}$ are distinct. Then algorithm CG generates the sequences

$$\begin{aligned} r_i &= \gamma_{i-1}(A)y + z; \qquad & i = 1, \ldots, \ell, \\ p_i &= \phi_{i-1}(A)y + \omega_{i-1} z; \qquad & i = 1, \ldots, \ell, \end{aligned} \tag{16}$$

where $\Gamma_j(\epsilon)$ and $\Omega_j(\epsilon)$ are real polynomials with degree j, $\omega_j \in \mathbb{R}$, $j < \hat{l}$. The quantities $\Gamma_j(A)$, $\Omega_j(A)$ and ω_j are recursively defined as follows:

$$
\begin{array}{lll}
\Gamma_0(A) = I, & \Gamma_j(A) = \Gamma_{j-1}(A) - \alpha_j A \Omega_{j-1}(A), & j \ge 1, \\
\Omega_0(A) = I, & \Omega_j(A) = \Gamma_j(A) + \beta_j \Omega_{j-1}(A), & j \ge 1, \\
\omega_0 = 1, & \omega_j = 1 + \beta_j \omega_{j-1}, & j \ge 1,
\end{array}
\tag{17}
$$

where α_j and β_j are calculated in algorithm CG. Finally, directions p_i, $i = 1, \ldots, \hat{l}$, satisfy condition $p_i \notin N(A)$ and are linearly independent.

<u>Proof.</u>
By complete induction, when $i = 1$ it is $r_1 = p_1 = y + z$, $\Gamma_0(A) = \Omega_0(A) = I$ and $\omega_0 = 1$. Now, let

$$
\begin{aligned}
r_{i-1} &= \Gamma_{i-2}(A)y + z; \\
p_{i-1} &= \Omega_{i-2}(A)y + \omega_{i-2} z;
\end{aligned}
$$

with $\Gamma_{i-2}(A) = \Gamma_{i-3}(A) - \alpha_{i-2} A \Omega_{i-3}(A)$, $\Omega_{i-2}(A) = \Gamma_{i-2}(A) + \beta_{i-2}\Omega_{i-3}(A)$, $\omega_{i-2} = 1 + \beta_{i-2}\omega_{i-3}$. Then, from Table 1 and (15) vectors r_i and p_i are given by

$$
r_i = r_{i-1} - \alpha_{i-1} A p_{i-1} = \Gamma_{i-2}(A)y + z - \alpha_{i-1} A \Omega_{i-2}(A) y = \Gamma_{i-1}(A)y + z; \tag{18}
$$

$$
p_i = r_i + \beta_{i-1} p_{i-1} = \Gamma_{i-1}(A)y + z + \beta_{i-1}\Omega_{i-2}(A)y + \beta_{i-1}\omega_{i-2} z = \Omega_{i-1}(A)y + \omega_{i-1} z. \tag{19}
$$

Hence, (16) and (17) hold. It remains to prove that directions $p_1, \ldots, p_{\hat{l}}$ are linearly independent and satisfy $p_i \notin N(A)$, $i \le \hat{l}$. The symmetry of matrix A yields $y^T z = 0$, thus from (16), $p_i \in N(A)$ if and only if $\Omega_{i-1}(A)y = 0$. However, from Lemma 2.1 the latter equality cannot be satisfied as long as $i \le \hat{l}$. Therefore $p_i \notin N(A)$, $i = 1, \ldots, \hat{l}$, so that the results of Proposition 3.1 complete the proof. $\Box$

In other words, if vector y has nonzero projection on eigenvectors $y_{j_1}, \ldots, y_{j_{\hat{l}}}$, then from Lemma 2.1 the CG generates exactly $\hat{l}$ conjugate directions $p_i \notin N(A)$, $i \le \hat{l}$, before stopping.

3.1 The CG and the Moore-Penrose pseudoinverse

Let $r_1 = y + z$ in Table 1, with $y = Pr_{R(A)}(r_1)$, $z = Pr_{N(A)}(r_1)$. If $z \neq 0$ we have from (16) $r_i \neq 0$, for any $i \ge 1$. Thus, if $z \neq 0$ the CG will not converge to a solution of linear system (1).

On the contrary if $r_1 \in R(A)$, then from (16) $r_i = 0$ if and only if $\Gamma_{i-1}(A)y = 0$ (i.e. $i - 1 = \hat{l}$). Moreover from Theorem 3.1 if $r_1 \in R(A)$, then $r_i \neq 0$, $i = 1, \ldots, \hat{l}$, and Lemma 2.1 yields $r_{\hat{l}+1} = 0$. Thus, if $z = 0$ algorithm CG eventually provides a solution x^* of (1).
Consider the Moore-Penrose generalized inverse A^+ [CM79] of the positive semidefinite matrix A in (1). If $Ax^* = b$ then $A^+ b = Pr_{R(A)}(x^*)$ [CM79]. Since by definition $r_1 = b - Ax_1$, we get

$$
Pr_{R(A)}(x^*) = A^+ b = A^+(r_1 + Ax_1) = A^+ r_1 + Pr_{R(A)}(x_1). \tag{20}
$$

Now, let $r_1 = y + z$, with $z = 0$, and let $R^S(r_1; A) = \mathrm{span}\{y_{j_1}, \ldots, y_{j_{\hat{l}}}\}$[1]. We prove that algorithm CG supplies an approximation of matrix A^+ on the linear subspace $R^S(r_1; A)$ (see also [Hes75]). Indeed, we have

$$
x^* = x_1 + \sum_{i=1}^{\hat{l}} \alpha_i p_i; \tag{21}
$$

[1] We remind that according with Theorem 3.1 y_{j_i}, $i = 1, \ldots, \hat{l}$, are eigenvectors of matrix A, associated to the distinct positive eigenvalues on which the initial residual $r_1 = y$ has nonzero projection.

and recalling that $r_i^T p_i = r_1^T p_i$, after a projection of (21) onto $R(A)$ we obtain[2]

$$Pr_{R(A)}(\bar{x} - x_1) = Pr_{R(A)}(\bar{x}) - Pr_{R(A)}(x_1) = \left[\sum_{i=1}^{\ell} \frac{p_i p_i^T}{p_i^T A p_i}\right] r_1. \tag{22}$$

From (20) and (22), observing that $z = 0$ we obtain for any $y \in R^S(r_1; A)$

$$\left[A^+ - \sum_{i=1}^{\ell} \frac{p_i p_i^T}{p_i^T A p_i}\right] y = \left[A^+ - (p_1 \cdots p_\ell) D_\ell^{-1} (p_1 \cdots p_\ell)^T\right] y = 0; \qquad D_\ell = \mathrm{diag}_{1 \le i \le \ell}\{p_i^T A p_i\}; \tag{23}$$

which proves that an approximation of the pseudoinverse matrix A^+ can be iteratively calculated by algorithm CG, on subspace $R^S(r_1; A)$.

Remark 3.1 Observe that $R^S(r_1; A) \equiv K_{\ell-1}(r_1; A)$, i.e. (23) provides an approximation of A^+ over the Krylov subspace spanned by vectors $p_1, \ldots, p_\ell$.

As proved above (see (16)), if $z \neq 0$ the CG in Table 1 does not converge to a solution for the linear system (1). Nevertheless also in this case we are concerned with investigating the results provided.

Lemma 3.1 Let $b \notin R(A)$ and let the hypothesis of Theorem 3.1 hold. The solution $\bar{x} = x_1 + \sum_{i=1}^{\ell} \alpha_i p_i$ provided by CG when solving (1) is not a least square solution of (1).

<u>Proof.</u>
Indeed (see (15) and (16)) from Theorem 3.1 directions $p_1, \ldots, p_\ell$ are generated. Then, setting $\tilde{p}_i = \phi_{i-1}(A)y$, $i = 1, \ldots, \ell$, we have $p_i = \tilde{p}_i + \omega_{i-1} z$ and recalling that $r_i^T p_i = r_1^T p_i$,

$$\begin{aligned}
\bar{x} &= x_1 + \sum_{i=1}^{\ell} \alpha_i p_i = x_1 + \sum_{i=1}^{\ell} \frac{p_i p_i^T}{p_i^T A p_i} r_1 \\
&= x_1 + \sum_{i=1}^{\ell} \left[\frac{\tilde{p}_i \tilde{p}_i^T}{\tilde{p}_i^T A \tilde{p}_i} r_1 + \omega_{i-1} \frac{\|z\|^2 \tilde{p}_i + \tilde{p}_i^T y z + \omega_{i-1} \|z\|^2 z}{\tilde{p}_i^T A \tilde{p}_i} \right] \\
&= x_1 + \sum_{i=1}^{\ell} \left[\frac{\tilde{p}_i \tilde{p}_i^T}{\tilde{p}_i^T A \tilde{p}_i} r_1 + \omega_{i-1} \frac{\|z\|^2}{\tilde{p}_i^T A \tilde{p}_i} \tilde{p}_i \right] + \sum_{i=1}^{\ell} \omega_{i-1} \left[\frac{\tilde{p}_i^T y + \omega_{i-1} \|z\|^2}{\tilde{p}_i^T A \tilde{p}_i} \right] z.
\end{aligned} \tag{24}$$

Then, since $r_1 = y + z$ and y has nonzero orthogonal projection only on eigenvectors $y_{j_1}, \ldots, y_{j_\ell}$, we get from (23) and (24)

$$\begin{aligned}
Pr_{R(A)}(\bar{x}) &= Pr_{R(A)}(x_1) + \left[\sum_{i=1}^{\ell} \frac{\tilde{p}_i \tilde{p}_i^T}{\tilde{p}_i^T A \tilde{p}_i}\right] y + \left[\sum_{i=1}^{\ell} \omega_{i-1} \frac{\|z\|^2}{\tilde{p}_i^T A \tilde{p}_i} \tilde{p}_i\right] \\
&= Pr_{R(A)}(x_1) + A^+ y + \left[\sum_{i=1}^{\ell} \omega_{i-1} \frac{\|z\|^2}{\tilde{p}_i^T A \tilde{p}_i} \tilde{p}_i\right]; \qquad \forall y \in R^S(r_1; A).
\end{aligned} \tag{25}$$

---

[2] We remark that for the linear space $R(A)$ the following property holds:

$$Pr_{R(A)}(y_1 - y_2) = Pr_{R(A)}(y_1) - Pr_{R(A)}(y_2); \qquad \forall y_1, y_2 \in \mathbb{R}^n.$$

Now, by contradiction let $\bar{x}$ be a least square solution of system (1), then it should be $\bar{x} = A^+b + \bar{z}$ with $\bar{z} \in N(A)$, hence

$$Pr_{R(A)}(\bar{x}) = A^+b = A^+(r_1 + Ax_1) = A^+y + Pr_{R(A)}(x_1). \tag{26}$$

Comparing (25) and (26) we realize that (26) does not hold, because the right most term in (25) is nonzero in general. Therefore, $\bar{x}$ cannot be a least square solution of system (1). 2

4 Issues on algorithm FLR when matrix A is singular

Here we aim at extending the results in [Hes75, Hes80] and the previous section, when considering algorithm FLR in Table 3 for solving (1), in the case of indefinite and possibly singular matrix A. When the indefinite matrix A is nonsingular and algorithm FLR has not yet stopped, at step k we have either $d_k \neq 0$ or $\Delta_k \neq 0$ [Fas05] (i.e. we are ensured thet either step k_A or step k_B can be performed). In this section we are concerned with recasting an analogous result, under the hypothesis that matrix A is singular. Observe that at step k of algorithm FLR, $d_k = 0$ implies [Fas05]

$$\Delta_k = -\|Ap_k\|^4.$$

Hence if $d_k = 0$ and matrix A is singular, then Δ_k is nonzero as long as

$$p_k \notin N(A); \qquad k < n. \tag{27}$$

The following theorem yields some results in order to satisfy condition (27).

Theorem 4.1 Consider the linear system (1) and let matrix A be indefinite and possibly singular. Let in algorithm FLR $r_1 = y + z$, with $y = Pr_{R(A)}(r_1)$ and $z = Pr_{N(A)}(r_1)$. Suppose y has nonzero projection on l eigenvectors $y_{j_1}, \ldots, y_{j_l}$ of A, and only $\hat{l} \le l$ eigenvalues associated to $y_{j_1}, \ldots, y_{j_l}$ are distinct. Then algorithm FLR generates the sequences:

$$\begin{aligned} r_i &= P_{i-1}(A)y + z; & i \le \hat{l}, \\ p_i &= Q_{i-1}(A)y + m_{i-1}z; & i \le \hat{l}, \\ q_i &= R_i(A)y + n_i z; & i \le \hat{l}, \end{aligned} \tag{28}$$

where $P_j(\epsilon)$, $Q_j(\epsilon)$ and $R_j(\epsilon)$ are real polynomials of degree j; m_j, $n_j \in \mathbb{R}$. Moreover, directions p_i and q_i satisfy relations:

$$\begin{aligned} p_i &\notin N(A); & i \le \hat{l}, \\ q_i &\notin N(A); & i \le \hat{l} - 1. \end{aligned} \tag{29}$$

<u>Proof.</u>
By complete induction, when $i = 1$ then $r_1 = p_1 = y + z$, and if step 1_B is performed $q_1 = Ay$, according with (28). Moreover, let

$$\begin{aligned} r_{i-1} &= P_{i-2}(A)y + z; \\ p_{i-1} &= Q_{i-2}(A)y + m_{i-2}z; \\ q_{i-1} &= R_{i-1}(A)y + n_{i-1}z; \end{aligned}$$

then the following cases must be considered, depending on whether algorithm FLR performs step i_A or step i_B.

- Step i_A is performed, then

$$
\begin{aligned}
r_i &= P_{i-2}(A)y + z - a_{i-1}AQ_{i-2}(A)y = P_{i-1}(A)y + z;\\
p_i &= P_{i-1}(A)y + z + b_{i-1}[Q_{i-2}(A)y + m_{i-2}z] = Q_{i-1}(A)y + m_{i-1}z.
\end{aligned}
$$

- Step i_B is performed, then

$$
\begin{aligned}
r_i &= P_{i-3}(A)y + z - \hat c_{i-2}AQ_{i-3}(A)y - \hat d_{i-2}AR_{i-2}(A)y = P_{i-1}(A)y + z;\\
p_i &= P_{i-1}(A)y + z + \frac{\hat b_{i-2}}{\Delta_{i-2}}[d_{i-2}(R_{i-2}(A)y + n_{i-2}z) - \delta_{i-2}(Q_{i-3}(A)y + m_{i-3}z)]\\
&= Q_{i-1}(A)y + m_{i-1}z;
\end{aligned}
$$

and depending on whether the previous step was $(i-1)_A$ or $(i-2)_B$, we obtain for vector q_i at step i_B the relations:

$$
\begin{aligned}
q_i &= Ap_i + \beta_{i-1}p_{i-1} = AQ_{i-1}(A)y + \beta_{i-1}[Q_{i-2}(A)y + m_{i-2}z] = R_i(A)y + n_iz;\\
q_i &= Ap_i + \frac{\hat\psi_{i-2}}{\Delta_{i-2}}(d_{i-2}q_{i-2} - \delta_{i-2}p_{i-2}) = A[Q_{i-1}(A)y + m_{i-1}z] + \frac{\hat\psi_{i-2}}{\Delta_{i-2}}(d_{i-2}q_{i-2} - \delta_{i-2}p_{i-2})\\
&= R_i(A)y + n_iz;
\end{aligned}
$$

according with (28). As regards (29), the hypotheses ensure that $p_i \in N(A)$ if and only if $Q_{i-1}(A)y = 0$. By Lemma 2.1 the latter equality cannot hold as long as $i \le \hat\ell$. Similarly we have $q_i \in N(A)$ if and only if $R_i(A)y = 0$, hence, as long as $i < \hat\ell - 1$, $q_i \notin N(A)$. $\square$

Now consider algorithm FLR in Table 3 and let vectors t_k, $k \le n$, be defined in the following way:

$$
\begin{array}{llll}
\text{if} & |d_k| \ge \epsilon_k\|p_k\|^2 & \text{then set} & \alpha_k = a_k \ \text{ and } \ t_k = p_k;\\
\text{if} & |d_k| < \epsilon_k\|p_k\|^2 & \text{then set} & \begin{array}{l}\alpha_k = \hat c_k \ \text{ and } \ t_k = p_k;\\ \alpha_{k+1} = \hat d_k \ \text{ and } \ t_{k+1} = q_k.\end{array}
\end{array}
\tag{30}
$$

Proposition 4.1 Let matrix A in (1) be indefinite and possibly singular, let r_1 satisfy (7) and the hypothesis of Lemma 2.1. Then algorithm FLR generates directions $t_1, \ldots, t_{\hat\ell}$, with $t_i \notin N(A)$, $i = 1, \ldots, \hat\ell$, and these vectors are linearly independent.

Proof.
The result straightforwardly holds from [Fas05], Theorem 4.1 and Lemma 2.1. $\square$

4.1 Algorithm FLR and the Moore-Penrose pseudoinverse

In this section we complete the analogy between algorithms CG and FLR, when they are applied for solving (1) and matrix A is singular. In particular we aim at obtaining for algorithm FLR relations similar to (23) and (25). Consider Theorem 4.1 and suppose FLR has generated $\hat\ell$ directions $t_1, \ldots, t_{\hat\ell}$ before stopping. Then, if $z = 0$ we prove that algorithm FLR can provide an approximation of the Moore-Penrose pseudoinverse A^+ (where A is indefinite and possibly singular).
More specifically, we introduce the following linear subspace, dependent on matrix A and vector r_1

$$
\mathcal{R}^P(r_1; A) = \mathrm{span}\{w_1, \ldots, w_{\hat\ell}\}, \tag{31}
$$

where $w_1, \ldots, w_{\hat\ell}$ are eigenvectors of matrix A, associated to distinct nonzero eigenvalues, on which the initial residual r_1 has nonzero projection. Now, since $r_1 = y + z$, from relation (28) algorithm

FLR can give the solution x^* of (1) provided that $z = 0$. Moreover, if $b \in R(A)$ (i.e. $z = 0$) exactly ℓ directions will be generated by algorithm FLR before converging to x^*. Indeed, Lemma 2.1 and Theorem 4.1 ensure that algorithm FLR generates exactly the independent directions $t_1, \ldots, t_\ell$, since the last step performed by FLR is either step $(\ell - 1)_A$ or step $(\ell - 2)_B$. As a consequence, if x^* is a solution of linear system (1) detected by algorithm FLR, by the definition of Moore-Penrose pseudoinverse [CM79]

$$Pr_{R(A)}(x^*) = A^+ b = A^+(r_1 + Ax_1) = A^+ r_1 + Pr_{R(A)}(x_1), \tag{32}$$

where matrix A is indefinite and possibly singular. Moreover, from (30)

$$x^* = x_1 + \sum_{i=1}^{\ell} \alpha_i t_i, \tag{33}$$

and assuming $z = 0$, from (28) of Theorem 4.1:

$$Pr_{R(A)}(x^*) = Pr_{R(A)}(x_1) + \sum_{i=1}^{\ell} \alpha_i t_i. \tag{34}$$

Finally, combining (32) and (34), and considering again relation $z = 0$, along with the expression of coefficients α_i, $i = 1, \ldots, \ell$ in (30), we have[3]:

$$\begin{aligned} A^+ y &= \sum_{i=1}^{\ell} \alpha_i t_i = \sum_{i \in S_1} a_i p_i + \sum_{i \in S_2} \left[\hat{c}_i p_i + \hat{d}_i q_i \right] \\ &= \sum_{i \in S_1} \frac{p_i^T r_i}{p_i^T A p_i} p_i + \sum_{i \in S_2} \left[\frac{(e_i p_i - \delta_i q_i)^T r_i}{\Delta_i} p_i + \frac{(d_i q_i - \delta_i p_i)^T r_i}{\Delta_i} q_i \right]. \end{aligned} \tag{35}$$

Now, it can be readily proved that $p_i^T r_i = p_i^T r_1$, $q_i^T r_i = q_i^T r_1$ [Fas05]. Thus, recalling that $\Delta_i \neq 0$ in (35), Table 3 and (31) yield for any $y \in R^P(r_1; A)$

$$\begin{aligned} 0 &= A^+ y - \sum_{i \in S_1} \frac{p_i p_i^T}{p_i^T A p_i} r_1 - \sum_{i \in S_2} \left[\frac{p_i (e_i p_i - \delta_i q_i)^T}{\Delta_i} + \frac{q_i (d_i q_i - \delta_i p_i)^T}{\Delta_i} \right] r_1 \\ &= A^+ y - \left[\sum_{i \in S_1} \frac{p_i p_i^T}{p_i^T A p_i} + \sum_{i \in S_2} \frac{(p_i \ \ q_i) \begin{pmatrix} e_i & -\delta_i \\ -\delta_i & d_i \end{pmatrix} \begin{pmatrix} p_i^T \\ q_i^T \end{pmatrix}}{\Delta_i} \right] y \\ &= \left[A^+ - \sum_{i \in S_1} \frac{p_i p_i^T}{p_i^T A p_i} - \sum_{i \in S_2} (p_i \ \ q_i) \begin{pmatrix} d_i & \delta_i \\ \delta_i & e_i \end{pmatrix}^{-1} \begin{pmatrix} p_i^T \\ q_i^T \end{pmatrix} \right] y \end{aligned} \tag{36}$$

$$= \left[A^+ - (t_1 \cdots t_\ell) B_\ell^{-1} (t_1 \cdots t_\ell)^T \right] y; \qquad B_\ell = \mathrm{diag}_{i \in S_1, j \in S_2} \left\{ d_i \, ; \begin{pmatrix} d_j & \delta_j \\ \delta_j & e_j \end{pmatrix} \right\}. \tag{37}$$

Observe that in (36), whenever the pairs (p_i, q_i), $i \in S_2$, are conjugate (i.e. $\delta_i = 0$, for any $i \in S_2$), then relation (37) reduces exactly to (23).

[3] In the following relations we have introduced the pair of disjoint sets S_1 and S_2 such that: S_1 is the set of indices $h \leq \ell$ for which algorithm FLR performs step h_A, while S_2 is the set of indices $h \leq \ell$ for which algorithm FLR performs step h_B. Thus, for the cardinality of the sets S_1 and S_2 relation $|S_1| + 2|S_2| = \ell$ holds.

In addition, let (λ_i, v_i), $i = 1, \ldots, n$, be the eigenpairs of the symmetric nonsingular matrix $C \in \mathbb{R}^{n \times n}$. Then the spectral form of C^{-1} is simply [GV89]

$$C^{-1} = \sum_{i=1} \frac{1}{\lambda_i} v_i v_i^T = (v_1 \cdots v_n)\Lambda^{-1}(v_1 \cdots v_n)^T; \qquad \Lambda = \mathrm{diag}_{1 \le i \le n}\{\lambda_i\};$$

which is clearly generalized by (37) in the singular case. Finally, likewise CG we prove the following

Theorem 4.2 Let $b \notin R(A)$ and let the hypothesis of Theorem 4.1 hold. Then the solution $x^* = x_1 + \sum_{i=1}^{h} \alpha_i t_i$, calculated by algorithm FLR when solving (1) is not a least square solution of (1).

Proof.

Consider relation (28) and let $b \notin R(A)$ (i.e. $z \neq 0$). From Lemma 2.1 algorithm FLR provides in exact arithmetic $r_{h+1} = z$, after the generation of directions $t_1, \ldots, t_h$. Now, by means of the substitutions $\tilde p_i = Q_{i-1}(A)y$ and $\tilde q_i = R_i(A)y$ in relations (28), we obtain from Table 3

$$\begin{aligned}
x^* &= x_1 + \sum_{i=1}^{h} \alpha_i t_i = x_1 + \left[\sum_{i \in S_1} \frac{p_i p_i^T}{p_i^T A p_i} + \sum_{i \in S_2} (p_i \ \ q_i) \begin{pmatrix} d_i & \delta_i \\ \delta_i & e_i \end{pmatrix}^{-1} \begin{pmatrix} p_i^T \\ q_i^T \end{pmatrix}\right] r_1 \\
&= x_1 + \sum_{i \in S_1} \frac{[\tilde p_i + m_{i-1}z][\tilde p_i + m_{i-1}z]^T}{\tilde p_i^T A \tilde p_i} r_1 + \\
&\quad + \sum_{i \in S_2} (\tilde p_i + m_{i-1}z \ \ \ \tilde q_i + n_i z) \begin{pmatrix} d_i & \delta_i \\ \delta_i & e_i \end{pmatrix}^{-1} \begin{pmatrix} \tilde p_i^T + m_{i-1}z^T \\ \tilde q_i^T + n_i z^T \end{pmatrix} r_1;
\end{aligned}$$

and since $\tilde p_i^T z = \tilde q_i^T z = 0$, $z^T r_1 = \|z\|^2$, we obtain

$$\begin{aligned}
x^* = x_1 &+ \sum_{i \in S_1} \left[\frac{\tilde p_i \tilde p_i^T}{\tilde p_i^T A \tilde p_i} r_1 + \|z\|^2 \frac{m_{i-1}}{\tilde p_i^T A \tilde p_i} \tilde p_i \right] + \lambda_1 z + \\
&+ \sum_{i \in S_2} (\tilde p_i \ \ \tilde q_i) \begin{pmatrix} d_i & \delta_i \\ \delta_i & e_i \end{pmatrix}^{-1} \begin{pmatrix} \tilde p_i^T \\ \tilde q_i^T \end{pmatrix} r_1 + \\
&+ \|z\|^2 (\tilde p_i \ \ \tilde q_i) \begin{pmatrix} d_i & \delta_i \\ \delta_i & e_i \end{pmatrix}^{-1} \begin{pmatrix} m_{i-1} \\ n_i \end{pmatrix} \Bigg] + \lambda_2 z;
\end{aligned} \tag{38}$$

where $\lambda_1, \lambda_2 \in \mathbb{R}$ summarize the dependency of the solution point x^* from vector z. Now, observe that x^* can be a least squares solution of (1) if and only if $x^* = A^+ b + \bar z$, with $\bar z \in N(A)$. Thus, projecting x^* in (38) onto the subspace $R(A)$, we simply have

$$\begin{aligned}
Pr_{R(A)}(x^*) = Pr_{R(A)}(x_1) &+ \sum_{i \in S_1} \left[\frac{\tilde p_i \tilde p_i^T}{\tilde p_i^T A \tilde p_i} y + \|z\|^2 \frac{m_{i-1}}{\tilde p_i^T A \tilde p_i} \tilde p_i \right] + \\
&+ \sum_{i \in S_2} (\tilde p_i \ \ \tilde q_i) \begin{pmatrix} d_i & \delta_i \\ \delta_i & e_i \end{pmatrix}^{-1} \begin{pmatrix} \tilde p_i^T \\ \tilde q_i^T \end{pmatrix} y + \qquad (39) \\
&+ \|z\|^2 (\tilde p_i \ \ \tilde q_i) \begin{pmatrix} d_i & \delta_i \\ \delta_i & e_i \end{pmatrix}^{-1} \begin{pmatrix} m_{i-1} \\ n_i \end{pmatrix} \Bigg] . \qquad (40)
\end{aligned}$$

Finally, recalling (32) and (36), and considering in (39) the terms which contain $\|z\|^2$, we conclude that if $b \notin R(A)$, x^* is not a least square solution of the linear system (1). $\square$

5 The Lanczos process and algorithm FLR

In this section we describe a twofold result: first we report some theoretical properties of the Lanczos process (Table 2) in case matrix A in (1) is singular. This aims at investigating possible similarities with the results of Section 4, where algorithm FLR is studied in the singular case. Then, a relevant relationship between the Lanczos vectors $\{u_i\}$ and the residuals $\{r_i\}$ calculated by the algorithm FLR is accomplished. We prove that the proper choice of parameter ε_k, at step k of algorithm FLR, plays a key role for the latter purpose.

Theorem 5.1 Consider the linear system (1) where A is indefinite and possibly singular. Consider the Lanczos process and let $u_1 = y + z$, with $y = Pr_{R(A)}(u_1)$ and $z = Pr_{N(A)}(u_1)$. Let y have nonzero projection on l eigenvectors $y_{j_1}, \ldots, y_{j_l}$ of A, and only $\hat{l} \leq l$ eigenvalues associated to $y_{j_1}, \ldots, y_{j_l}$ are distinct. Then, the Lanczos process generates the sequence of orthonormal vectors

$$u_i = U_{i-1}(A)y + \eta_{i-1}z; \qquad 1 \leq i \leq \hat{l}; \tag{41}$$

where $U_j(\epsilon)$ is a real polynomial of degree j and $\eta_j \in \mathbb{R}$, with $(j \geq 3)$

$$\begin{aligned}
&U_0(A) = \frac{I}{\delta_0}; && \eta_0 = \frac{1}{\delta_0};\\
&U_1(A) = \frac{1}{\delta_1}(A - \gamma_1 I)U_0(A); && \eta_1 = -\frac{\gamma_1\eta_0}{\delta_1};\\
&U_{j-1}(A) = \frac{1}{\delta_{j-1}}\left[(A - \gamma_{j-1}I)U_{j-2}(A) - \delta_{j-2}U_{j-3}(A)\right]; && \eta_{j-1} = -\frac{\gamma_{j-1}\eta_{j-2} + \delta_{j-2}\eta_{j-3}}{\delta_{j-1}}.
\end{aligned} \tag{42}$$

Moreover $u_i \notin N(A)$, for any $i \leq \hat{l}$.

Proof.
From the hypothesis and Lemma 2.1, the Lanczos process performs exactly $\hat{l}$ iterations before stopping. Finally, considering the guidelines of Theorem 4.1, complete induction yields (41) and (42). $\square$

Theorem 5.2 Let matrix A in (1) be indefinite and possibly singular. Suppose the Lanczos process and algorithm FLR are applied to solve (1), with $x_1 = 0$ in algorithm FLR. Then in exact arithmetic algorithms Lanczos and FLR perform the same number of iterations.

Proof.
Evidently, if at the step k both the Lanczos process and algorithm FLR have not yet stopped, they have respectively generated the orthogonal sequences $u_1, \ldots, u_k$ and $t_1, \ldots, t_k$, in the Krylov subspaces $K_k(u_1; A)$ and $K_k(r_1; A)$. Since $x_1 = 0$

$$K_k(u_1; A) \equiv K_k(r_1; A); \tag{43}$$

so that the statement holds from (28), (41) and Lemma 2.1. $\square$

Theorem 5.3 The vectors u_i, $i \geq 1$, and $r_i/\|r_i\|$, $i \geq 1$, generated respectively by the Lanczos process and algorithm FLR with $x_1 = 0$, in exact arithmetic satisfy relation:

$$u_i = s_i \frac{r_i}{\|r_i\|}; \qquad s_i \in \{+1, -1\}. \tag{44}$$

Proof.
By complete induction, $x_1 = 0$ yields

$$u_1 = \frac{r_1}{\|r_1\|} = \frac{b}{\|b\|}. \tag{45}$$

Now suppose $u_{i-1} = s_{i-1} r_{i-1}/\|r_{i-1}\|$, we prove that $u_i = s_i r_i/\|r_i\|$. On this purpose, let $\hat{h}$ be the number of iterations performed by Lanczos process and algorithm FLR, according with Theorem 5.2. Recall that the Lanczos vectors $u_1, \dots, u_{\hat{h}}$, satisfy $u_i^T u_j = 0$, $\hat{h} \ge i \ne j \ge 1$ [GV89]. Furthermore, considering at step k_B of algorithm FLR the dummy residual [Fas01, BC94]

$$r_{k+1} = -\alpha_k r_k - (1+\alpha_k)\,\mathrm{sgn}(d_k) A p_k; \qquad \alpha_k = -\frac{|d_k|}{\|r_k\|^2 + |d_k|}; \qquad \mathrm{sgn}(d_k) = \begin{cases} 1 & d_k \ge 0 \\ -1 & d_k < 0; \end{cases} \tag{46}$$

the sequence $r_1, \dots, r_{\hat{h}}$ satisfies $r_i^T r_j = 0$, where $\hat{h} \ge i \ne j \ge 1$ [Fas05]. Now observe that

$$\begin{cases} u_i \in \mathcal{K}_i(u_1; A) \\ u_i \perp \mathcal{K}_{i-1}(u_1; A) = \mathrm{span}\{\mathcal{K}_{i-2}(u_1; A); u_{i-1}\}; \end{cases} \qquad \begin{cases} r_i \in \mathcal{K}_i(r_1; A) \\ r_i \perp \mathcal{K}_{i-1}(r_1; A) = \mathrm{span}\{\mathcal{K}_{i-2}(r_1; A); r_{i-1}\}; \end{cases} \tag{47}$$

and from (45) and the inductive hypothesis $\mathcal{K}_{i-1}(u_1; A) = \mathcal{K}_{i-1}(r_1; A)$. Thus, from (45) and (47) u_i and r_i are parallel. Finally, since $\|u_i\| = 1$ relation (44) holds. $\square$

Theorem 5.4 Consider algorithm FLR in Table 3. Let $x_1 = 0$ and let at step i_B the dummy residual (46) be calculated. If at step i the parameter ϵ_i is chosen according with

$$\begin{array}{ll} 0 < \bar{\epsilon} \le \epsilon_i; & \text{step } i_A; \\ 0 < \bar{\epsilon} \le \epsilon_i < \min\left\{ \dfrac{\|Ap_i\|^2 \|r_i\|^2}{\|p_i\|^4}; \dfrac{\|Ap_i\|^4}{\lambda_M \|p_i\|^2 \|q_i\|^2} \right\}; & \text{step } i_B; \end{array} \tag{48}$$

then in exact arithmetic the sequences $\{u_i\}$ and $\{r_i/\|r_i\|\}$ generated by algorithms Lanczos and FLR satisfy

$$u_i = s_i \frac{r_i}{\|r_i\|}; \qquad i \ge 1; \tag{49}$$

where

$$\begin{array}{ll} s_1 = 1; & \\ s_i = -s_{i-1}\,\mathrm{sgn}(p_{i-1}^T A p_{i-1}) & \text{if step } (i-1)_A \text{ is performed;} \\ \left.\begin{array}{l} s_{i-1} = -s_{i-2}\,\mathrm{sgn}(p_{i-2}^T A p_{i-2}) \\ s_i = -s_{i-2} \end{array}\right\} & \text{if step } (i-2)_B \text{ is performed.} \end{array} \tag{50}$$

Proof.
The hypothesis $x_1 = 0$ trivially yields $u_1 = r_1/\|r_1\|$, i.e. $s_1 = 1$. Now, by complete induction we prove (49) and (50) with $i = 2$ (step $(i-1)_A$) and $i = 3$ (step $(i-2)_B$). Then, we assume they

hold for $i-1$ and we prove them for i.
On one hand, in case $i = 2$ and step 1_A was performed, then it is:

$$\begin{aligned}
u_2^T\left(\frac{r_2}{\|r_2\|}\right) &= \left(\frac{v_1}{\|v_1\|}\right)^T\frac{r_2}{\|r_2\|} = \frac{1}{\|v_1\|\|r_2\|}[(A-\gamma_1 I)u_1]^T r_2 \\
&= \frac{1}{\|v_1\|\|r_2\|}(Au_1)^T r_2 = \frac{s_1}{\|v_1\|\|r_2\|\|r_1\|}(Ar_1)^T(r_1 - a_1 Ar_1) \\
&= -s_1\,\mathrm{sgn}(r_1^T Ar_1)\frac{\|r_1\|^2\|Ar_1\|^2 - (r_1^T Ar_1)^2}{\|v_1\|\|r_2\|\|r_1\||r_1^T Ar_1|} \\
&= -s_1\,\mathrm{sgn}(p_1^T Ap_1)\frac{\|r_1\|^2\|Ar_1\|^2 - (r_1^T Ar_1)^2}{\|v_1\|\|r_2\|\|r_1\||r_1^T Ar_1|};
\end{aligned}$$

which implies from Theorem 5.3

$$s_2 = -s_1\,\mathrm{sgn}(p_1^T Ap_1).$$

On the other hand, in case $i = 3$ and step 1_B was performed, then we have:

$$\begin{aligned}
u_2^T\left(\frac{r_2}{\|r_2\|}\right) &= \left(\frac{v_1}{\|v_1\|}\right)^T\frac{r_2}{\|r_2\|} = \frac{s_1}{\|v_1\|\|r_2\|}[(A-\gamma_1 I)u_1]^T r_2 \\
&= \frac{1}{\|v_1\|\|r_2\|}(Au_1)^T r_2 = \frac{s_1}{\|v_1\|\|r_2\|\|r_1\|}(Ar_1)^T\left[-\alpha_1 r_1 - (1+\alpha_1)\mathrm{sgn}(p_1^T Ap_1)Ar_1\right] \\
&= \frac{-s_1}{\|v_1\|\|r_2\|\|r_1\|}\left[\alpha_1 r_1^T Ar_1 + (1+\alpha_1)\mathrm{sgn}(p_1^T Ap_1)\|Ar_1\|^2\right] \\
&= \frac{-s_1\,\mathrm{sgn}(p_1^T Ap_1)}{\|v_1\|\|r_2\|\|r_1\|}\left[-\frac{(p_1^T Ap_1)^2}{\|r_1\|^2+|d_1|} + \frac{\|r_1\|^2\|Ar_1\|^2}{\|r_1\|^2+|d_1|}\right];
\end{aligned}$$

which again implies from Theorem 5.3

$$s_2 = -s_1\,\mathrm{sgn}(p_1^T Ap_1);$$

and

$$\begin{aligned}
u_3^T\left(\frac{r_3}{\|r_3\|}\right) &= \frac{[(A-\gamma_2 I)u_2 - \delta_1 u_1]^T r_3}{\|v_2\|\|r_3\|} = \frac{(Au_2)^T r_3}{\|v_2\|\|r_3\|} \\
&= \frac{s_2(Ar_2)^T r_3}{\|v_2\|\|r_3\|\|r_2\|} = \frac{s_2\left(\frac{(1+\alpha_1)\mathrm{sgn}(p_1^T Ap_1)[r_3 - r_1 + \hat{c}_1 Ap_1]}{\hat{d}_1} - \alpha_1 Ar_1\right)^T r_3}{\|v_2\|\|r_3\|\|r_2\|} \\
&= \frac{s_2(1+\alpha_1)\mathrm{sgn}(p_1^T Ap_1)\|r_3\|^2}{\hat{d}_1\|v_2\|\|r_3\|\|r_2\|};
\end{aligned}$$

which implies from Theorem 5.3 (the choice of ε_1 yields $\hat{d}_1 > 0$)

$$s_3 = s_2\,\mathrm{sgn}(p_1^T Ap_1) = -s_1[\mathrm{sgn}(p_1^T Ap_1)]^2 = -s_1.$$

Let us now prove (49) and (50) for index i. On this purpose, from the inductive hypothesis:

$$\begin{aligned}
u_i^T\left(\frac{r_i}{\|r_i\|}\right) &= \left(\frac{v_{i-1}}{\|v_{i-1}\|}\right)^T\frac{r_i}{\|r_i\|} = \frac{1}{\|v_{i-1}\|\|r_i\|}[(A-\gamma_{i-1}I)u_{i-1} - \delta_{i-2}u_{i-2}]^T r_i \\
&= \frac{1}{\|v_{i-1}\|\|r_i\|}(Au_{i-1})^T r_i = \frac{s_{i-1}}{\|v_{i-1}\|\|r_i\|}\left(A\frac{r_{i-1}}{\|r_{i-1}\|}\right)^T r_i. \qquad (51)
\end{aligned}$$

Now we analize two subcases. If step $(i-1)_A$ was performed, then (51) becomes

$$u_i^T\left(\frac{r_i}{\|r_i\|}\right) = \frac{s_{i-1}}{\|v_{i-1}\|\|r_i\|}\left(\frac{Ar_{i-1}}{\|r_{i-1}\|}\right)^T r_i = \frac{s_{i-1}}{\|v_{i-1}\|\|r_i\|}\left(\frac{\frac{r_{i-1}-r_i}{a_{i-1}} - b_{i-2}Ap_{i-2}}{\|r_{i-1}\|}\right)^T r_i$$

$$= \frac{-s_{i-1}\|r_i\|^2}{\|v_{i-1}\|\|r_i\|\|r_{i-1}\|a_{i-1}} = -s_{i-1}\,\mathrm{sgn}(p_{i-1}^TAp_{i-1})\frac{\|r_i\||p_{i-1}^TAp_{i-1}|}{\|v_{i-1}\|\|r_{i-1}\|^3};$$

which implies from Theorem 5.3

$$s_i = -s_{i-1}\,\mathrm{sgn}(p_{i-1}^TAp_{i-1}).$$

If step $(i-2)_B$ was performed we have two further cases. On one hand, using (30), (46) and relation (51) it is

$$u_{i-1}^T\left(\frac{r_{i-1}}{\|r_{i-1}\|}\right) = \frac{s_{i-2}}{\|v_{i-2}\|\|r_{i-1}\|}\left(\frac{-\frac{\alpha_{i-2}r_{i-2}+r_{i-1}}{(1+\alpha_{i-2})\mathrm{sgn}(p_{i-2}^TAp_{i-2})} - A\omega_1(t_{i-4},t_{i-3})}{\|r_{i-2}\|}\right)^T r_{i-1}$$

$$= \frac{s_{i-2}\|r_{i-1}\|^2}{\|v_{i-2}\|\|r_{i-1}\|\|r_{i-2}\|}\frac{-1}{(1+\alpha_{i-2})\mathrm{sgn}(p_{i-2}^TAp_{i-2})},$$

where $\omega_1(t_{i-4},t_{i-3})$ is a linear combination of vectors t_{i-4} and t_{i-3}. The previous relation and Theorem 5.3 imply

$$s_{i-1} = -s_{i-2}\,\mathrm{sgn}(p_{i-2}^TAp_{i-2}).$$

Furthermore, considering that

$$\begin{aligned}
r_i &= r_{i-2} - \hat\gamma_{i-2}Ap_{i-2} - \hat\delta_{i-2}Aq_{i-2};\\
q_{i-2} &= Ap_{i-2} + \omega_2(t_{i-4},t_{i-3});\\
r_{i-1} &= -\alpha_{i-2}r_{i-2} - (1+\alpha_{i-2})\mathrm{sgn}(p_{i-2}^TAp_{i-2})Ap_{i-2};
\end{aligned}$$

where again $\omega_2(t_{i-4},t_{i-3})$ is a linear combination of vectors t_{i-4} and t_{i-3}, it is

$$r_i = r_{i-2} - \hat\gamma_{i-2}Ap_{i-2} - \hat\delta_{i-2}A\left[-\frac{r_{i-1}+\alpha_{i-2}r_{i-2}}{(1+\alpha_{i-2})\mathrm{sgn}(p_{i-2}^TAp_{i-2})} + \omega_2(t_{i-4},t_{i-3})\right];$$

hence

$$Ar_{i-1} = \frac{\left[r_i - r_{i-2} + \hat\gamma_{i-2}At_{i-2} + \hat\delta_{i-2}A\omega_2(t_{i-4},t_{i-3})\right]}{\hat\delta_{i-2}}(1+\alpha_{i-2})\mathrm{sgn}(p_{i-2}^TAp_{i-2}) + \alpha_{i-2}Ar_{i-2}.$$

Therefore relation (51) becomes

$$u_i^T\left(\frac{r_i}{\|r_i\|}\right) = \frac{s_{i-1}}{\|v_{i-1}\|\|r_i\|}\left(A\frac{r_{i-1}}{\|r_{i-1}\|}\right)^T r_i = \frac{s_{i-1}}{\|v_{i-1}\|\|r_i\|\|r_{i-1}\|}\frac{(1+\alpha_{i-2})\mathrm{sgn}(p_{i-2}^TAp_{i-2})}{\hat\delta_{i-2}}\|r_i\|^2$$

$$= \frac{\|r_i\|s_{i-1}}{\|v_{i-1}\|\|r_{i-1}\|}\frac{\frac{\|r_{i-2}\|^2}{\|r_{i-2}\|^2+|d_{i-2}|}\mathrm{sgn}(t_{i-2}^TAt_{i-2})}{\hat\delta_{i-2}}; \tag{52}$$

and according with the choice of ε_{i-2}, the coefficient $\hat\delta_{i-2}$ is positive, so that (52) and Theorem 5.3 yield

$$s_i = s_{i-1}\mathrm{sgn}(p_{i-2}^TAp_{i-2}) = -s_{i-2}[\mathrm{sgn}(p_{i-2}^TAp_{i-2})]^2 = -s_{i-2}. \qquad \Box$$

Remark 5.1 Observe that condition (48) on ϵ_i is slightly less restrictive in respect to condition (12) proposed in [Fas05], since it does not require the knowledge of λ_m. As regards the apparently cumbersome computation of q_i in (48), refer to the considerations in [Fas05].
We also highlight that the approximation of the Moore-Penrose pseudoinverse A^+, provided in (37) by algorithm FLR, is not inexpensively available from the Lanczos process. In particular, the set of directions $t_1, \ldots, t_h$ should be ad hoc generated by the Lanczos process.

Note that relations (49) and (50) are also a generalization of the results reported in [CGT00], by replacing CG with algorithm FLR. In particular, in matrix terms the Lanczos process gives at step k [CGT00]

$$T_k^{(L)} = U_k^T A U_k, \tag{53}$$

where

$$T_k^{(L)} = \begin{pmatrix} \delta_0 & \gamma_1 & & & \\ \gamma_1 & \delta_1 & \cdot & & \\ & \cdot & \cdot & \cdot & \\ & & \cdot & \delta_{k-1} & \gamma_k \\ & & & \gamma_k & \delta_k \end{pmatrix}, \qquad U_k = (u_1 \cdots u_k);$$

and relations (49)-(50) can be restated as

$$U_k = R_k S_k, \tag{54}$$

where

$$R_k = \left(\frac{r_1}{\|r_1\|} \cdots \frac{r_k}{\|r_k\|} \right), \qquad S_k = \mathrm{diag}_{1 \le i \le k}\{s_i\}.$$

From (53) and (54) we obtain

$$T_k^{(L)} = S_k^T (R_k^T A R_k) S_k = S_k T_k^{(FLR)} S_k, \tag{55}$$

where the tridiagonal matrix $T_k^{(FLR)}$ is available at step k of algorithm FLR. The explicit expression of $T_k^{(FLR)}$, in terms of the coefficients of algorithm FLR, is given in [Fas01].

Proposition 5.1 In the hypothesis of Theorem 5.4 and in exact arithmetic, the tridiagonal matrix $T_k^{(L)}$ by the Lanczos process is a straightforward by-product of algorithm FLR, as indicated in (55).

Furthermore, in the hypothesis of Theorems 5.1 and 5.4, the solution x^* of (1) provided by the Lanczos process, may be given in terms of algorithm FLR quantities. Indeed, $x^* = U_h [T_h^{(L)}]^{-1} \|b\| e_1$ [S03] and from (55)

$$x^* = U_h S_h \left[T_h^{(FLR)} \right]^{-1} S_h \|b\| e_1 = R_h \left[T_h^{(FLR)} \right]^{-1} S_h \|b\| e_1 = R_h \left[T_h^{(FLR)} \right]^{-1} \|b\| e_1,$$

where the last equality follows from relation $e_1^T = (1\ 0 \cdots 0)^T$ and (50).

6 Conclusions and Perspectives

This paper describes several properties of the planar-CG algorithm FLR proposed in [Fas05], for solving indefinite linear systems. We have proved that the sequence of orthogonal residuals $\{r_i\}$ by algorithm FLR, yields the sequence of orthogonal vectors $\{u_i\}$ from the Lanczos process, provided

that parameter ϵ_i at step i of FLR is chosen according with Theorem 5.4. Since algorithm FLR is a cheap CG-type method, this result encourages to consider a numerical comparison of these methods within nonconvex optimization frameworks, where efficient tools for the solution of indefinite linear systems are claimed.
On the other hand we have studied the solution of linear system $Ax = b$, $A \in \mathbb{R}^{n \times n}$ indefinite and possibly singular, by means of the algorithm FLR: this extended the results provided by the CG in the positive semidefinite case [Hes75].

We conclude that algorithm FLR proved to be a general tool for the solution of symmetric linear systems, i.e. for the search of stationary points of quadratic forms, in unconstrained optimization frameworks. In addition, the approximation of the Moore-Penrose pseudoinverse A^+ provided by algorithm FLR, may be a fruitful instrument for the construction of preconditioners [K02]. Finally as Section 1 reported, the Newton method for the computation of real eigenvectors, could gain advantage from considering algorithm FLR.

Acknowledgments

This work has been supported by the U.S. Office of Naval Research under the grant No. 000140210489, through Dr. Pat Purtell.

References

[Bjo96] A.BJORCK (1996) \Numerical Methods for Least Squares Problems", SIAM, Philadelphia, PA.

[BC94] R.E.BANK, T.F.CHAN (1994)\A Composite step bi-conjugate gradient algorithm for non-symmetric linear systems", Numerical Algorithms, Vol. 7, pp. 1-16.

[CM79] S.L.CAMPBELL, C.D.MEYER Jr. (1979) \Generalized inverses of linear transformations", Dover Publications, Inc., New York.

[CGT00] A.R.CONN, N.I.M.GOULD, PH.L.TOINT (2000) \Trust-Region Methods", SIAM, Philadelphia, PA.

[DDS85] L.C.W.DIXON, P.G.DUCKSBURY, P.SINGH (1985)\A New Three-Term Conjugate Gradient Method", Technical Report 130, Numerical Optimization Centre, Hat¯eld Polytechnic, Hat¯eld, Hertfordshire, England, 1985.

[Fas01] G.FASANO (2001)\Use of Conjugate Directions inside Newton-type Algorithms for Large-Scale Unconstrained Optimization", PhD Dissertation, Rome, Italy, 2001.

[Fas05] G.FASANO (2005)\Planar-Conjugate Gradient Algorithm for Large-Scale Unconstrained Optimization, Part 1: Theory", JOTA, vol. 125, n. 3, 2005.

[FLS04] G.FASANO, F.LAMPARIELLO, M.SCIANDRONE (2004)\A Truncated Nonmonotone Gauss-Newton Method for Large-Scale Nonlinear Lest-Squares Problems", Technical Report n. 590, IASI-CNR, Rome.

[Gre97] A.GREENBAUM (1997) \Iterative Methods for Solving Linear Systems", SIAM , Frontiers in Applied Mathematics, Philadelphia.

[GV89] G.H.GOLUB, C.F.VAN LOAN (1989) \Matrix computations - 3rd edition", The John Hopkins Press, Baltimore.

[Han98] P.C.HANSEN (1998) \Rank-De¯cient and Discrete Ill-Posed Problems", SIAM, Philadelphia.

[HS52] M.R.HESTENES, E.STIEFEL (1952) \Methods of Conjugate Gradients for solving Linear Systems", J. Res. Nat. Bur. Standards, vol. 49, n. 6, pp. 409-436.

[Hes75] M.R.HESTENES (1975) \Pseudoinverses and Conjugate Gradients", Communications of the ACM, vol. 18, issue 1, pp. 40-43.

[Hes80] M.R.HESTENES (1980) \Conjugate Direction Methods in Optimization", Springer Verlag, New York Heidelberg Berlin.

[K02] C.KOSCHINSKI (2002)\New Methods for Adapting and for Approximating Inverses as Preconditioners", Applied Numerical Mathematics, Vol. 41, Issue 1, pp. 179-218.

[LS03] F.LAMPARIELLO, M.SCIANDRONE (2003) \Use of the Minimum-norm search direction in a nonmonotone version of the Gauss-Newton method", Journal of Optimization Theory and Applications, vol. 119, pp. 65-82.

[LS91] Y.LIU, C.STOREY (1991) \E±cient generalized Conjugate Gradient Algorithms, Part 1: Theory", Journal of Optimization Theory and Applications, Vol. 69, pp. 129-137, 1991.

[Lue69] D.G.LUENBERGER (1969)\Hyperbolic pairs in the Method of Conjugate Gradients", SIAM J. Appl. Math., vol. 17, n.6, pp. 1263-1267.

[MC69] A.MIELE, J.W..CANTRELL (1969)\Study on a Memory Gradient Method for the Minimization of Functions", Journal of Optimization Theory and Applications, Vol. 3, pp. 459-470, 1969.

[McC83] G.P.McCORMICK (1983) \Nonlinear Programming: Theory, Algorithm and Application", Wiley & Sons, New York.

[S03] Y.SAAD (2003)\Iterative Methods for Sparse Linear Systems - Second Edition", SIAM, Philadelphia, PA.

[SVdV00] Y.SAAD, H.A.VAN DER VORST (2000)\Iterative Solution of Linear Systems in the 20-th Century", J. Comp. Appl. Math. 123, pp. 1-33.

[SVdV96] G.L.G.SLEIJPEN, H.A.VAN DER VORST (1996)\The Jacobi-Davidson method for eigenvalue problems as an accelerated inexact Newton scheme", in Iterative Methods in Linear Algebra II: proceedings of the ImACS International Symposium on Iterative Methods in Linear Algebra, June 17-20, pp. 377-389, 1995, Blagoevgrad. Elsevier, Amsterdam; New York, March 1996.

[WSS98] K.WU, Y.SAAD, A.STATHOPOULOS (1998)\Inexact Newton Preconditioning Techniques for Large Symmetric Eigenvalue Problems", Electronic Transactions on Numerical Analysis, vol. 7, pp. 202-214.