7659, England, UK
Journal of Information and Computing Science
Vol. 2, No. 1, 2007, *pp*. 77-80

Preconditioned Conjugate Gradient is *M*-Error-Reducing

Haiyan Gao [+], Hua Dai

Department of mathematics, Nanjing University of Aeronautics and Astronautics,
Nanjing 210016, P. R. China

(*Received May 07, 2006, accepted July 09, 2006*)

Abstract. The Preconditioned Conjugate Gradient (PCG) method has proven to be extremely powerful for solving symmetric positive definite linear systems. This method can also be applied to nonsymmetric linear systems when combined with the NR/NE techniques. It has been shown in [1] that the CGNR algorithm, which is a nonsymmetric variant of the Conjugate Gradient (CG) method, is error-reducing with respect to the Euclidean norm. However, in practice the simple CGNR algorithm is seldom used because of the squared condition number of the iteration matrix. Preconditioning is frequently needed to overcome this difficulty. In the present paper we give a much richer result concerning the error-reducing property of the CG procedure. Assume that the preconditioner *M* is also symmetric positive definite. It is shown that the PCG method is error-reducing with respect to the *M*-norm.

Keywords: Linear Systems, Preconditioned Conjugate Gradient, CGNR, CGNE, Error.

1. Introduction

The Preconditioned Conjugate Gradient method is designed to solve symmetric positive definite linear systems of the form

$$Ax = b, \quad A \in R^{n\times n}; x, b \in R^n . \tag{1.1}$$

Assume that the preconditioner M is also symmetric positive definite. There are three commonly seen versions of the PCG method:

i. Left-Preconditioned CG: CG applied to the linear system

$$M^{-1}Ax = M^{-1}b ,$$

in which the standard Euclidean inner product is replaced by the M-inner product

$$(x, y)_M \equiv (Mx, y) = (x, My) ;$$

ii. Split-Preconditioned CG: In the case where M is a Cholesky product $M = LL^T$, CG applied to the linear system

$$L^{-1}AL^{-T}u = L^{-1}b, \quad x = L^{-T}u , \tag{1.2}$$

which involves a symmetric positive definite matrix;

iii. Right-Preconditioned CG: CG applied to the linear system

$$AM^{-1}u = b, \quad x = M^{-1}u ,$$

in which the standard Euclidean inner product is replaced by the M^{-1} -inner product.

Interestingly, these algorithms are mathematically equivalent; see [3, Section 9.2]. Let $\hat{A} = L^{-1}AL^{-T}$, $u_j = L^T x_j$ and $\hat{r}_j = L^{-1}r_j$, where x_j is the *j*th PCG iterate and $r_j = b - Ax_j$ is the associated residual. For any symmetric positive definite matrix K , we denote by $\| v \|_K$ the *K*-norm of v , defined as $\| v \|_K = \sqrt{(Kv, v)}$. The three algorithms can be cast in the same form:

Algorithm 1.1. Preconditioned Conjugate Gradient

[+] Corresponding author.
E-mail address: hygao@nuaa.edu.cn.

Compute $r_0 = b - Ax_0$; $\hat{r}_0 = L^{-1} r_0$; and $p_0 = \hat{r}_0$.

For $j = 0,1,\Lambda$, until convergence Do:

$$\alpha_j = \| \hat{r}_j \|_2^2 \Big/ \| p_j \|_{\hat{A}}^2$$

$$u_{j+1} = u_j + \alpha_j p_j$$

$$\hat{r}_{j+1} = \hat{r}_j - \alpha_j \hat{A} p_j$$

$$\beta_j = \| \hat{r}_{j+1} \|_2^2 \Big/ \| \hat{r}_j \|_2^2$$

$$p_{j+1} = \hat{r}_{j+1} + \beta_j p_j$$

End Do.

Although a little complicated, this formulation of the PCG method is quite comprehensible to ordinary minds. It is precisely the Conjugate Gradient method applied to the preconditioned system $\hat{A}u = L^{-1}b$, whose solution u_* is related to the solution of the original system (1.1) by $u_* = L^T x_*$.

In order to solve nonsymmetric linear systems $Ax = b$, we can apply the PCG method directly to the equivalent systems

$$A^T Ax = A^T b, \tag{1.3}$$

or

$$AA^T u = b, \quad x = A^T u. \tag{1.4}$$

The resulted algorithms are known as the Preconditioned CGNR and CGNE, see [2].

It is the purpose of this paper to show that PCG is an error-reducing method with respect to the *M*-norm, i.e., the *M*-norm of the error $e_k \equiv x_* - x_k$ monotonically decreases during its iteration. The proof of our theorem follows closely the lines of a proof given in [1], where the simple CGNR algorithm is considered.

2. Main results

We begin with some known facts. Recall that Algorithm 1.1 is precisely the CG method applied to (1.2) with respect to the u-variable. There are two characterizing properties of the CG procedure:

$$(\hat{r}_k, \hat{r}_j) = \begin{cases} 0 & k \neq j \\ \| \hat{r}_j \|_2^2 & k = j \end{cases}; \tag{2.1}$$

$$\| u_* - u_j \|_{\hat{A}} = \min_{u \in u_0 + K_j(\hat{r}_0, \hat{A})} \| u_* - u \|_{\hat{A}}, \tag{2.2}$$

where u_* is the exact solution of the system (1.2) and $K_j(\hat{r}_0, \hat{A}) = span\{\hat{A}^i \hat{r}_0\}_{i=0}^{j-1}$ is the associated Krylov subspace. Observe that

$$\begin{aligned} \| u_* - u_j \|_{\hat{A}} &= \sqrt{(\hat{A}(u_* - u_j), u_* - u_j)} = \sqrt{(L^{-1}AL^{-T}(u_* - u_j), u_* - u_j)} \\ &= \sqrt{(AL^{-T}(u_* - u_j), L^{-T}(u_* - u_j))} = \sqrt{(A(x_* - x_j), x_* - x_j)} = \| x_* - x_j \|_A . \end{aligned}$$

(2.2) can be equivalently shown as

$$\| x_* - x_j \|_A = \min_{x \in x_0 + K_j(M^{-1}r_0, M^{-1}A)} \| x_* - x \|_A . \tag{2.3}$$

We will prove a theorem to show that the PCG method is *M*-error-reducing. The following lemmas are needed first.

Lemma 2.1. The direction vector p_k in Algorithm 1.1 is a linear combination of the residuals $\hat{r}_0, \hat{r}_1, \Lambda, \hat{r}_k$:

$$p_k = \xi_k \left(\frac{\hat{r}_0}{\xi_0} + \frac{\hat{r}_1}{\xi_1} + \Lambda + \frac{\hat{r}_k}{\xi_k} \right), \quad k = 0,1,\Lambda \;, \tag{2.4}$$

where

$$\xi_j = \| \hat{r}_j \|_2^2, \ j = 0,1,\Lambda\ ,k .$$

Proof: From lines 7 and 6 of Algorithm 1.1, we have

$$p_{j+1} = \hat{r}_{j+1} + \beta_j p_j = \xi_{j+1}\left(\frac{p_j}{\xi_j} + \frac{\hat{r}_{j+1}}{\xi_{j+1}} \right). \tag{2.5}$$

Observe that $p_0 = \hat{r}_0$. By an inductive argument, (2.4) follows form (2.5).

Lemma 2.2. Let $\hat{e}_k = u_* - u_k$ be the error of the CG iterate u_k generated in Algorithm 1.1. Then, $(p_k, \hat{e}_{k+1}) \geq 0, \quad k = 0,1,\Lambda$. (2.6)

Proof: We assume exact arithmetic. It is well known that the CG method gives the exact solution in at most n iterations. Hence, $u_* = u_n$ and then $(p_{n-1}, \hat{e}_n) = 0$. When $k < n-1$, from the update of u_k in Algorithm 1.1, we have

$$u_* = u_n = u_{k+1} + \alpha_{k+1} p_{k+1} + \Lambda\ + \alpha_{n-1} p_{n-1} .$$

Then

$$(p_k, \hat{e}_{k+1}) = (p_k, u_* - u_{k+1}) = \alpha_{k+1}(p_k, p_{k+1}) + \Lambda\ + \alpha_{n-1}(p_k, p_{n-1}) . \tag{2.7}$$

By (2.4) and (2.1), we have

$$(p_k, p_j) = \xi_k \xi_j \left(\frac{\hat{r}_0}{\xi_0} + \Lambda\ + \frac{\hat{r}_k}{\xi_k}, \quad \frac{\hat{r}_0}{\xi_0} + \Lambda\ + \frac{\hat{r}_j}{\xi_j} \right) = \xi_k \xi_j \left(\frac{1}{\xi_0} + \Lambda\ + \frac{1}{\xi_k} \right) \geq 0 , \tag{2.8}$$

$$(j = k+1, \Lambda\ , n-1) .$$

(2.6) then follows from combining (2.7, 2.8) with the fact that $\alpha_j, \ j = k+1, k+2, \Lambda\ , n-1$ are all nonnegative.

Lemma 2.3. In Algorithm 1.1, the 2-norm of $\hat{e}_k$ decreases monotonically, i.e.

$$\| \hat{e}_{k+1} \|_2 \leq \| \hat{e}_k \|_2, \quad k = 0,1,\Lambda\ . \tag{2.9}$$

Proof: Form lines 4 of Algorithm 1.1, we have

$$\begin{aligned} \| \hat{e}_k \|_2^2 &= \| u_* - u_k \|_2^2 \\ &= \| u_* - u_{k+1} + \alpha_k p_k \|_2^2 \\ &= \| u_* - u_{k+1} \|_2^2 + 2a_k (p_k, u_* - u_{k+1}) + \| \alpha_k p_k \|_2^2 \\ &= \| \hat{e}_{k+1} \|_2^2 + 2\alpha_k (p_k, \hat{e}_{k+1}) + \| \alpha_k p_k \|_2^2 \end{aligned}$$

Using (2.6) we then obtain (2.9).

We can now apply Lemma 2.3 to give the following theorem.

Theorem 2.4. The PCG method is M-error-reducing, i.e.,

$$\| e_{k+1} \|_M \leq \| e_k \|_M , \ k = 0,1,\Lambda\ .$$

Proof: Observe that

$$\begin{aligned} \| \hat{e}_k \|_2 &= \sqrt{(\hat{e}_k, \hat{e}_k)} = \sqrt{(u_* - u_k, u_* - u_k)} \\ &= \sqrt{(L^T(x_* - x_k), L^T(x_* - x_k)} = \sqrt{(LL^T(x_* - x_k), x_* - x_k)} \\ &= \sqrt{(Me_k, e_k)} = \| e_k \|_M, \quad k = 0,1,\Lambda\ . \end{aligned}$$

Using (2.9) we then obtain the result.

CGNR is nothing but the CG method applied to the normal equations (1.3). Letting $M = I$, the next corollary follows form Theorem 2.4 immediately.

Corollary 2.5. CGNR is 2-error-reducing.

Note that this is the main result of [1], i.e., Proposition 2.2.

Concerning the CGNE algorithm, we have the following corollary.

Corollary 2.5. CGNE is 2-error-reducing with respect to the u-variable.

3. Concluding remarks

The Preconditioned Conjugate Gradient method has been known as a robust method because of its coding simplicity and its error-minimizing property with the *A*-norm i.e., (2.3). In this short note we have shown another attractive property of this method: it is also error-reducing in the *M*-norm when the preconditioner M is symmetric positive definite.

The CGNR and CGNE algorithms have proven to be very effective for nonsymmetric linear systems in many situations. Typically, it has been observed that if the coefficient matrix of a nonsymmetric linear system is of high defectiveness with the related eigenvalues less that 1, the usual Krylov subspace methods (in restarted version) will be in great risks of breakdown or stagnation, see [4]. In this situation, the CGNR and CGNE algorithms may be two promising substitutes because they are based on normal equations. Preconditioning is needed to reduce the squared condition number of their iteration matrices. The results presented in this paper indicate that when the preconditioner is chosen to be symmetric positive definite [2], it will be potentially advantageous that both the error-minimizing property and the error-reducing property of the CG procedure are retained.

4. References

[1] C. Li. CGNR is an Error Reducing Algorithm. *SIAM J. Sci. Comput.*, 2001, **22**:2109-2112.

[2] Y. Saad. Preconditioning Techniques for Indefinite and Nonsymmetric Linear Systems. *J. Comput. Appl. Math.*, 1988, **24**:89-105.

[3] Y. Saad. *Iterative Methods for Sparse Linear Systems*. PWS Publishing, Boston, 1996.

[4] B. J. Zhong. On the Breakdowns of the Galerkin and Least-squares Methods, *Numer. Math.- A J. of Chinese Universities.* 2002, **11**:137-148