Applied Mathematical Sciences, Vol. 6, 2012, no. 33, 1643 – 1654

Residual Algorithm with Preconditioner for Linear System of Equations

J. Izadian *

Department of Mathematics, Faculty of Sciences, Mashhad Branch,
Islamic Azad University, Mashhad, Iran.
Jalal_Izadian@yahoo.com

M. Jalili

Department of Mathematics, Neyshabur Branch,
Islamic Azad University, Neyshabur, Iran.
Jalili.maryam@yahoo.com

Abstract

One of the most powerful tools for solving large and sparse systems of linear equation is iterative methods. Their significant advantages like low memory requirements and good approximation properties make them very popular, and they are widely used in applications throughout science and engineering. Residual Algorithm for solving large-scale nonsymmetric linear system of equation which symmetric part is positive (or negative) definite, is evaluated. It uses in a systematic way the residual vector as a search direction, and a spectral steplength. The global convergence is analyzed. A preliminary numerical experimentation is included for showing that the new algorithm is a robust method for solving nonsymmetric linear system and it is competitive with the well-known GMRES and BICGSTAB in number of computed residual and CPU time. The new method for sparse matrix with 10^{12} entries has been successfully examined with we use the two preconditioning strategies ILU and SSOR.

Keywords: linear systems, nonsymmetric matrices, preconditioned iterative methods

1. Introduction

We introduce an iterative method for solving the linear system of equation

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

When $A \in R^{n\times n}$ is nonsymmetric and the symmetric part of A, i.e. , $A_s = \dfrac{A^t + A}{2}$ is positive (or negative) definite, $b \in R^n$ and n is large enough.

Different iterative methods have been developed for solving (1). To solve a very large linear system of the form given in (1), especially those derived from3D soil–structure interaction problems, direct solution methods such as sparse LU factorization are impractical from the perspective of computational time and memory requirement. Krylov subspace iterative methods on the other hand, may overcome these difficulties and they are commonly used for solving large-scale linear systems. Three of the most popular Krylov subspace iterative methods for solving the system (1) are SYMMLQ, MINRES and symmetric QMR (SQMR) method. The SYMMLQ and MINRES methods, developed by Paige and Saunders in1975 [6], could only be used in conjunction with symmetric positive definite preconditioners.

Unlike the optimal SYMMLQ and MINRES methods, the SQMR method proposed by Freund and Nachtigal [2] can be used in conjunction with a symmetric indefinite preconditioner. Because of this flexibility, we choose to use RA throughout this paper. There are also numerical results showing that RA combined with a symmetric indefinite preconditioner is more effective than a positive definite preconditioner .

This paper is organized as follows. In Section 2, we give the model algorithm RA. In Section 3, we give Extensions of the method which we combine RA with a preconditioning method in order to speed up convergence. In Section 4, we consider the application of the RA algorithm for solving large-scale nonsymmetric linear system of equations whose symmetric part is positive (or negative) definite.. In last section, we give a short conclusion.

2 . Model Algorithm

Let the functions f and g are given as

$$\begin{cases} g: R^n \to R^n \\ g(x) = Ax - b \end{cases} \tag{2}$$

$$\begin{cases} f: R^n \to R \\ f(x) = \|g(x)\|_2^2 \end{cases} \tag{3}$$

where $\|\cdot\|_2$ denotes the Euclidian norm. Now assume that $\{\eta_k\}$ is a given sequence such that $\eta_k > 0$ for $k \in N$ (the set of natural numbers) and

$$\sum_{k=0}^{\infty} \eta_k = \eta < \infty . \tag{4}$$

Suppose $\gamma \in (0,1)$, $0 < \sigma_{\min} < \sigma_{\max} < 1$, $\alpha_\circ > 0$ and $x_\circ \in R^n$ is a given arbitary initial point. Combining the systematic use of the search direction, the spectral choice of step length, and the Armijo.test , we obtain the following RA(Residual Algorithm)algorithm.

2-1 RA algorithm

Given: $\alpha_\circ > 0, \gamma \in (0,1)$, $0 < \sigma_{\min} < \sigma_{\max} < 1, \{\eta_k\}$ such that (4) holds, and $x_\circ \in R^n$ be a sufficiently good initial guess.

Set $r_\circ = b - Ax_\circ$, and $k = 0$.

Step1 : If $r_k = 0$, stop the process.

Step2 : Set $\lambda = 1$.

Step3 : If

$$\left\| r_k - (\frac{\lambda}{\alpha_k}) A r_k \right\|^2 \leq \|r_k\|^2 + \iota_k - \gamma\lambda^2 \|r_k\|^2$$

go to step5.

Step4 : Choose $\sigma \in [\sigma_{\min}, \sigma_{\max}]$, set $\lambda \leftarrow \sigma\lambda$ and go to step3.

Step5 : Set

$$\lambda_k = \lambda,$$

$$x_{k+1} = x_k + (\frac{\lambda}{\alpha_k}) r_k,$$

$$r_{k+1} = r_k - (\frac{\lambda}{\alpha_k}) A r_k$$

Step6 : Set

$$\alpha_{k+1} = \frac{r_k^t A r_k}{r_k^t r_k},$$

$$k = k + 1$$

go to step1.

A global convergence theorem is presented in [4], that guarantee convergence of the solution of problem (1).

3 Extensions of the method

Now, we combine RA with a preconditioning method in order to speed up convergence.

3-1 The preconditioned RA algorithm

We suppose that M be a preconditioning matrix. In stead of solving (1) one may as well solve

$$MAx = Mb \tag{5}$$

or

$$AMy = b \quad \text{with} \quad x = My\,. \tag{6}$$

Therefore by replacing A by MA and $r_\circ = b - Ax_\circ$ by $r_\circ = M\,(b - Ax_\circ)$ we have an algorithm to solve (5) iteratively. In this case the computed residuals r_k are not the real residuals, but $r_k = M\,(b - Ax_k)$ .By replacing A by AM and $X_\circ = x_\circ$ by $X_\circ = M^{-1}x_\circ$, we have an algorithm that solves iteratively (6).

The computed residuals are the real ones, that is, $r_k = b - AMx_k$, but now x_k is not the approximated value that we are interested in, we would like to have Mx_k instead. If we do not want to monitor the approximations of the exact solution x, it suffices to compute Mx_k only after termination. In both variants, the modified RA algorithm may converges faster, due to the preconditioning.

3-2 $SSOR$ preconditioning

The $SSOR$ preconditioner can be derived straight forward from matrix coefficient. The $SSOR$ preconditioner strategies is based on decomposition A as

$$A = L + D + U \tag{7}$$

where L and U, respectively, are the lower and upper triangular part of A and D is the diagonal part of A. The $SSOR$ matrix is defined as :

$$M = (D + L)\,D^{-1}(D + U) \tag{8}$$

3-3 ILU preconditioning

The Incompelete LU factorization [1,3], ILU, is probably the best known general purpose preconditioner .We define :

$$S = \{\ (i,\, j)\,|\, a_{ij} \neq 0\} \tag{9}$$

such that $A=(a_{ij})_{n\times n}$ .The ILU matrix denote $M=\tilde{L}\tilde{U}$. The ILU factorization, yields :

$$A=\tilde{L}\tilde{U}+R \tag{10}$$

where R ,is the residual matrix representation the difference between A and $\tilde{L}\tilde{U}$.

3-3-1 ILU algorithm for $A=(a_{ij})_{n\times n}$

Step1 : Do $r=1:n-1$
Step2 : $d=\dfrac{1}{a_{rr}}$
Step3 : Do $i=r+1:n$
Step4 : If $(i,r)\in S$ then
Step5 : $e=da_{ir}$
Step6 : $a_{ir}=e$
Step7 : Do $j=r+1,n$
Step8 : If $(i,j)\in S$ and $(r,j)\in S$ then
Step9 : $a_{ij}=a_{ij}-ea_{rj}$
Step10 : End
Step11 : End
Step12 : End
Step13 : End
Step14 : End

4. Numerical result

In this section we consider the application of the RA algorithm for solving large-scale nonsymmetric linear system of equations whose symmetric part is positive (or negative) definite. For RA algorithm we use the following parameters :

$$\alpha_\circ=\|b\|,\gamma=10^{-4},\sigma_{\min}=0.1,\sigma_{\max}=0.9,\eta_k=10^{-3}(1\text{-}10^{-7})^k$$

We choose a new λ at step4 by following procedure ,given the current $\lambda_c>0$ we set the new $\lambda>0$

$$\lambda=\begin{cases}\sigma_{\min}\lambda_C & if \quad \lambda_t<\sigma_{\min}\lambda_C\\ \sigma_{\max}\lambda_C & if \quad \lambda_t<\sigma_{\max}\lambda_C\\ \lambda_t & otherwise\end{cases}$$

where

$$\lambda_t = \frac{\lambda_C^2 f(x_k)}{f(x_k+\lambda_C d)+(2\lambda_C-1)f(x_k)} .$$

The iteration terminated if :

$$\frac{\|r_k\|}{\|b\|} \le \varepsilon \quad , \quad 0<\varepsilon<<1 \quad , \quad \varepsilon = 5\times 10^{-15} .$$

In the numerical result given below, the well-known and well-stablished Krylov subspace methods : $BICGSTAB$ and $GMRES$ [5,7], that restarts every 20 steps($m=20$), namely $GMRES(20)$ and $m=40$,namely $GMRES(40)$ have been used to solve the given system of linear equations and is compared with RA.

We have used a PC-Pentium(R)4 ,CPU 2.4 GHz, 448 MB of RAM and MATLAB R2006a for computations. In all of experiences we have used $x_\circ = 0$, on the initial guess. In the following numerical experiments, n and "nnz" denote the order of matrix, and the number of nonzero elements of matrix, respectively. On the other hand iter and CPUtime, refer to number of iteration and elapsed time, respectively. In Table1 we report, the dimension of the problem (n); the number of computed residual ($iter$), and elapsed time (sec) until convergence occurred ($CPUtime$).

Notation : RESALG denote RA in legend figure.

4-1 Example

We consider the matrix by $MATLAB$ code $A = gallery('hanowa', n, n)$. We assume that $n = 7000, b = (1,1,...,1)$ and $nnz = 14000$. The matrix for $n = 10$ is presented in Figure 1. In Figures 2 and 3 we show the behavior of all considered methods when using preconditioning strategies ILU and $SSOR$.

$n=7000$	RA		$BICGSTAB$		$GMRES(20)$		$GMRES(40)$	
	CPUtime	**iter**	**CPUtime**	**iter**	**CPUtime**	**iter**	**CPUtime**	**iter**
ILU	1.48547	3	1.85422	15	2.29685	18	0.34418	18
$SSOR$	0.98623	3	1.01651	7	1.22560	10	0.68507	10

Table 1

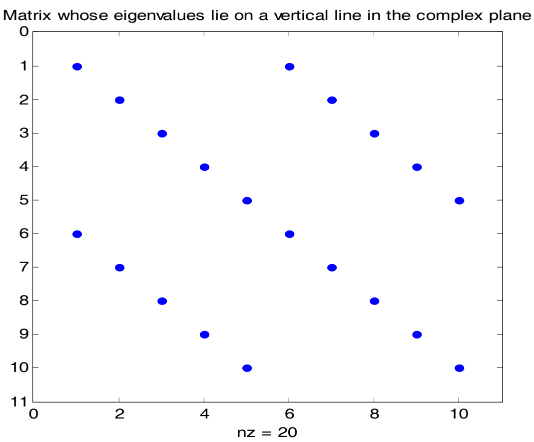

Figure 1

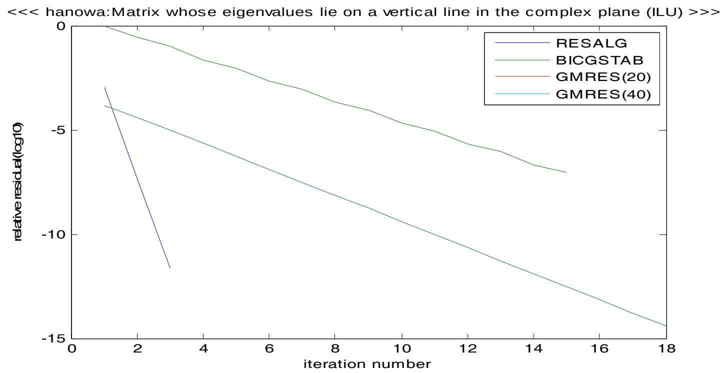

Figure 2:behavior of all method when use ILU preconditioner

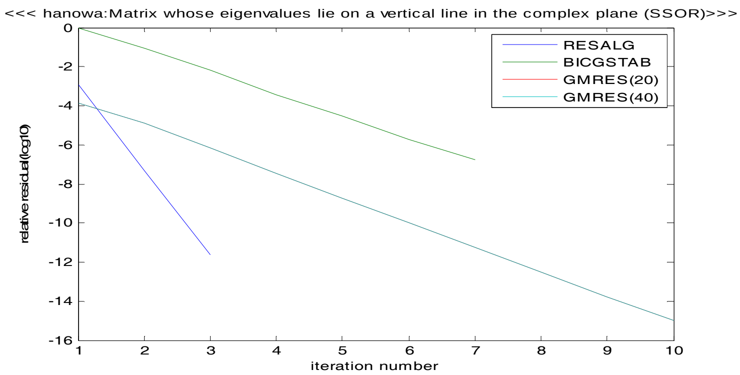

Figure 3:behavior of all method when use SSOR preconditioner

4-2 Example

Consider the matrix by $MATLAB$ code $A = gallery\ ('toeppen', n, 1, 10, n, -10, -1)$. Let assume that $n = 10^6$, $b = (1,1,\dots,1)$ and $nnz = 4999994$. The matrix in the case $n = 10$ has been presented in Figure 4, and in Figure 5 we show the behavior of all considered methods when using preconditioning strategies ILU.

$n = 10^6$	RA		$BICGSTAB$		$GMRES(20)$		$GMRES(40)$	
	CPUtime	**iter**	**CPUtime**	**iter**	**CPUtime**	**iter**	**CPUtime**	**iter**
ILU	14.04	2	186.379	2	402.538	3	*	*
$SSOR$	6.0553	2	269.234	1	388.271	2	*	*

Table 2

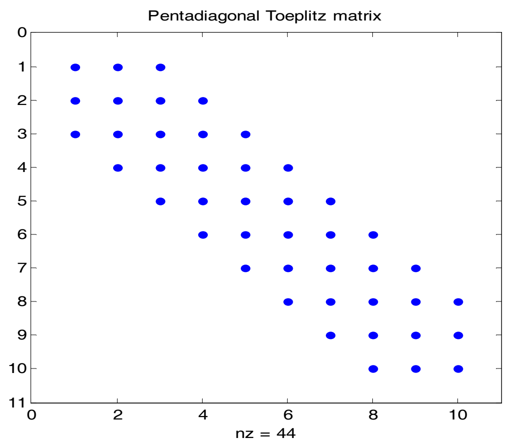

Figure 4

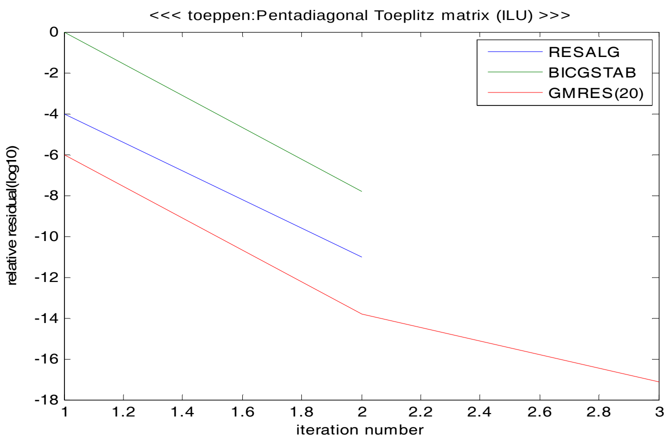

Figure 5:behavior of all method when use ILU preconditioner

4-3 Example

We consider the elliptic problem $u_{xx}+u_{yy}=e^{x^2-y^2}\sin(x+y)$ on the unit square, whit homogeneous Dirichlet boundary conditions, $u=0$, on the border of the region. The discretization grid has 100 internal nodes per axis producing an $n\times n$ matrix where $n=10^4$. In Figures 6,7 we show the behavior of all considered methods when using preconditioning strategies $ILU, SSOR$; Figure 8 shows $(x_k, y_k, u(x_k, y_k))$.

$n=10^4$	RA		$BICGSTAB$		$GMRES(20)$		$GMRES(40)$	
	CPUtime	iter	CPUtime	iter	CPUtime	iter	CPUtime	iter
ILU	0.02297	15	*	*	*	*	*	*
$SSOR$	5.59485	46	*	*	*	*	218.165	501

Table 3

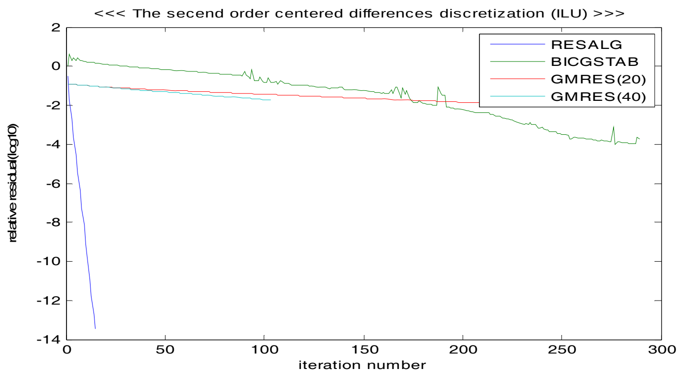

Figure 6:behavior of all method when use ILU preconditioner

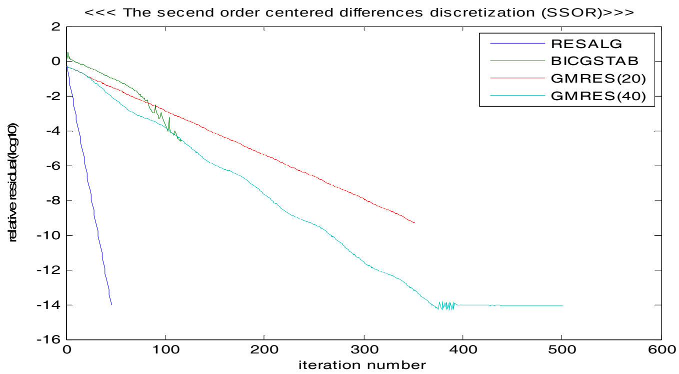

Figure 7:behavior of all method when use SSOR preconditioner

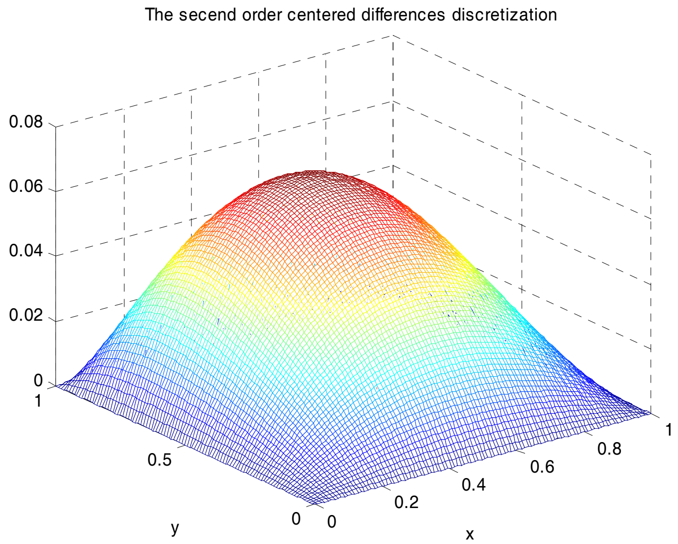

Figure 8

4-3 Example

We consider the following simple boundary-value problem of second order. We subdivide [-1,1] into $n=10^4$ equal subintervals, the difference method give n $n\times n$ matrix where $n=10^4$ and matrix is nonsymmetric positive definite that arise from the discretization of the boundary value problem. In Figure 9 we show the behavior of all considered methods when using preconditioning strategies ILU ; Figure 10 shows $(x_k, y_k, u(x_k, y_k))$

$$\begin{cases} y''(x)=\dfrac{1}{(2-x)}y'(x)+\dfrac{1}{(2(2-x)^2)}y(x)+x & -1\leq x\leq 1 \\ y(-1)=0 \\ y(1)=0 \end{cases}$$

$n=10^4$	RA		$BICGSTAB$		$GMRES(20)$		$GMRES(40)$	
	CPUtime	iter	CPUtime	iter	CPUtime	iter	CPUtime	iter
ILU	0.13051	43	*	*	*	*	*	*
$SSOR$	*	*	*	*	*	*	*	*

Table 4

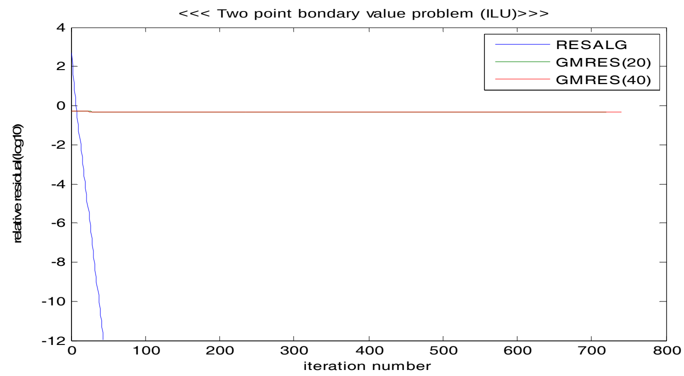

Figure 9:behavior of all method when use ILU preconditioner

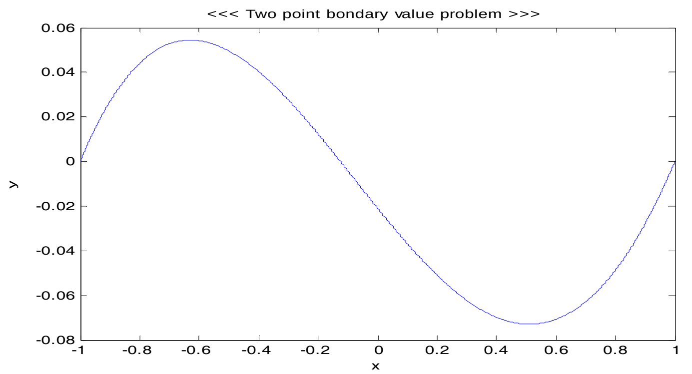

Figure 10

5. Conclusions

We have interpreted in this paper the RA algorithm, and compare with well-known Krylo subsspase methods, with using two preconditioning strategies ILU and $SSOR$. We observe that using the residual and spectral step length decrease nonmonotonically norm of the residual and also guarantees convergence. Finally, the numerical results confirm that the RA algorithm presented in this article is worth further application and apply for nonlinear system of equation.

References

[1] M. Benzi, D.B.Szyld and Arno.V. Duin (1999)." Orderings for Incompeleted Factorization Preconditioning of nonsymmetric problems".SIAM J.SCI.Compute.Vol.20,No.5,PP.1652-1670.

[2] R.W. Freund, N.M .Nachtigal. A new Krylov-subspace method for symmetric indefinite linear system. In Proceedings

[3] S.M. Hosseini and M.Rezghi (2006)." An ILU preconditioner for nonsymmetric positive definite matrices by using conjugate Gram-Schmidt process".Journal of Computational and Applied Mathematics.188-150-164 .

[4] W. La Cruzez and M. Raydan (2006)." Residual iterative scheme for large-scale linear systems". Lecturas en Ciencias de la Computación .ISSN 1316-6239.

[5] Y. Saad and M. H. Shultz (1986). "GMRES: generalized minimal residual algorithm for solving nonsymmetric linear systems. SIAM J. Sci. Stat". Comput., 7,856–869.

[6] CC . Paige, Saunders MA. Solution of sparse indefinite systems of linear equations. SIAM Journal on Numerical Analysis 1975; **12**:617– 629.

[7] H. A. Van der Vorst (1992). "Bi-CGSTAB: A fast and smoothly convergent variant Bi-CG for the solution of non-symmetric linear systems". SIAM J. Sci. Stat.Comput., 13, 631–644.

Received: October, 2011