American Journal of Computational Mathematics, 2015, 5, 113-126
Published Online June 2015 in SciRes. http://www.scirp.org/journal/ajcm
http://dx.doi.org/10.4236/ajcm.2015.52010

Comparison of Fixed Point Methods and Krylov Subspace Methods Solving Convection-Diffusion Equations

Xijian Wang

School of Mathematics and Computational Science, Wuyi University, Jiangmen, China
Email: wangxj1980426@163.com

Received 2 March 2015; accepted 5 June 2015; published 9 June 2015

Abstract

The paper first introduces two-dimensional convection-diffusion equation with boundary value condition, later uses the finite difference method to discretize the equation and analyzes positive definite, diagonally dominant and symmetric properties of the discretization matrix. Finally, the paper uses fixed point methods and Krylov subspace methods to solve the linear system and compare the convergence speed of these two methods.

Keywords

Finite Difference Method, Convection-Diffusion Equation, Discretization Matrix, Iterative Method, Convergence Speed

1. Introduction

In the case of a linear system $Ax = b$, the two main classes of iterative methods are the stationary iterative methods (fixed point methods) [1]-[3], and the more general Krylov subspace methods [4]-[13]. When these two classical iterative methods suffer from slow convergence for problems which arise from typical applications such as fluid dynamics or electronic device simulation, preconditioning [14]-[16] is a key ingredient for the success of the convergent process.

The goal of this paper is to find an efficient iterative method combined with preconditioning for the solution of the linear system $Ax = b$ which is related to the following two-dimensional boundary value problem (BVP) [17]:

How to cite this paper: Wang, X.J. (2015) Comparison of Fixed Point Methods and Krylov Subspace Methods Solving Convection-Diffusion Equations. *American Journal of Computational Mathematics*, **5**, 113-126.
http://dx.doi.org/10.4236/ajcm.2015.52010

$$\begin{cases} -\left(\dfrac{\partial^2 u}{\partial x^2}+\dfrac{\partial^2 u}{\partial y^2}\right)+\gamma\left(\dfrac{\partial u}{\partial x}-\dfrac{\partial u}{\partial y}\right)=f(x,y), & (x,y)\in\Omega=(0,1)^2 \\ u=0, & (x,y)\in\partial\Omega. \end{cases} \tag{1}$$

For parameter $\gamma=0,16,64$, we use finite difference method to discretize the Equation (1). Take $n+2$ points in both the *x*- and *y*-direction and number the related degrees of freedom first left-right and next bottom-top. Now, we take $n=32$, thus the grid size $\Delta x=\Delta y=h=\dfrac{1}{n+2-1}=\dfrac{1}{33}$. For $\gamma=0$, the convection term equals to 0, using central difference method to the diffusion term, we get the discretization matrix of the Equation (1)

$$A_0=\begin{pmatrix} B_0 & -I & O & \cdots & O \\ -I & B_0 & -I & \ddots & \vdots \\ O & \ddots & \ddots & \ddots & O \\ \vdots & \ddots & -I & B_0 & -I \\ O & O & O & -I & B_0 \end{pmatrix},\ B_0=\begin{pmatrix} 4 & -1 & 0 & \cdots & 0 \\ -1 & 4 & -1 & \ddots & \vdots \\ 0 & \ddots & \ddots & \ddots & 0 \\ \vdots & \ddots & -1 & 4 & -1 \\ 0 & 0 & 0 & -1 & 4 \end{pmatrix}. \tag{2}$$

For $\gamma=16$, the diffusion term use a central difference and for the convection term use central differences scheme as the following:

$$f(x_{ij},y_{ij})=-\left(\frac{u_{(i+1)j}-2u_{ij}+u_{(i-1)j}}{h^2}+\frac{u_{i(j+1)}-2u_{ij}+u_{i(j-1)}}{h^2}\right)+\gamma\left(\frac{u_{(i+1)j}-u_{(i-1)j}}{2h}-\frac{u_{i(j+1)}-u_{i(j-1)}}{2h}\right), \tag{3}$$

we obtain the discretization matrix of the Equation (1)

$$A_1=\begin{pmatrix} B_1 & -(1+c_1)I & O & \cdots & O \\ -(1-c_1)I & B_1 & -(1+c_1)I & \ddots & \vdots \\ O & \ddots & \ddots & \ddots & O \\ \vdots & \ddots & -(1-c_1)I & B_1 & -(1+c_1)I \\ O & O & O & -(1-c_1)I & B_1 \end{pmatrix}, \tag{4}$$

where

$$B_1=\begin{pmatrix} 4 & -(1-c_1) & 0 & \cdots & 0 \\ -(1+c_1) & 4 & -(1-c_1) & \ddots & \vdots \\ 0 & \ddots & \ddots & \ddots & 0 \\ \vdots & \ddots & -(1+c_1) & 4 & -(1-c_1) \\ 0 & 0 & 0 & -(1+c_1) & 4 \end{pmatrix}, c_1=\frac{\gamma h}{2}=\frac{8}{33}.$$

For $\gamma=64$, the diffusion term use a central difference and for the convection term use upwind differences scheme as the following:

$$f(x_{ij},y_{ij})=-\left(\frac{u_{(i+1)j}-2u_{ij}+u_{(i-1)j}}{h^2}+\frac{u_{i(j+1)}-2u_{ij}+u_{i(j-1)}}{h^2}\right)+\gamma\left(\frac{u_{ij}-u_{(i-1)j}}{h}-\frac{u_{i(j+1)}-u_{ij}}{h}\right), \tag{5}$$

we obtain the discretization matrix of the Equation (1)

$$A_2=\begin{pmatrix} B_2 & -(1+c_1)I & O & \cdots & O \\ -I & B_2 & -(1+c_1)I & \ddots & \vdots \\ O & \ddots & \ddots & \ddots & O \\ \vdots & \ddots & -I & B_2 & -(1+c_1)I \\ O & O & O & -I & B_2 \end{pmatrix}, \tag{6}$$

where

$$B_2=\begin{pmatrix} 4+2c_2 & -1 & 0 & \cdots & 0 \\ -(1+c_1) & 4+2c_2 & -1 & \ddots & \vdots \\ 0 & \ddots & \ddots & \ddots & 0 \\ \vdots & \ddots & -(1+c_1) & 4+2c_2 & -1 \\ 0 & 0 & 0 & -(1+c_1) & 4+2c_2 \end{pmatrix}, c_2=\gamma h=\frac{64}{33}.$$

2. Properties of the Discretization Matrix

In this section, we would first compute the eigenvalues of the discretization matrices A_0, A_1 and A_2 of Equation (1), later analyze positive definite, diagonally dominant and symmetric properties of these matrices.

2.1. Eigenvalues

Using MATLAB, the eigenvalues of the discretization matrices A_0, A_1 and A_2 of Equation (1) are the following:

$$\begin{cases} \lambda(A_0)_{p,q}=4+2\left(\cos\left(\frac{p\pi}{n+1}\right)+\cos\left(\frac{q\pi}{n+1}\right)\right), \\ \lambda(A_1)_{p,q}=4+2\sqrt{1-c_1^2}\left(\cos\left(\frac{p\pi}{n+1}\right)+\cos\left(\frac{q\pi}{n+1}\right)\right), \qquad p,q=1,2,\cdots,n. \\ \lambda(A_2)_{p,q}=4+2c_2+2\sqrt{1+c_2}\left(\cos\left(\frac{p\pi}{n+1}\right)+\cos\left(\frac{q\pi}{n+1}\right)\right), \end{cases} \tag{7}$$

2.2. Definite Positiveness

Matrix A is positive definite if and only if the symmetric part of A *i.e.* $A_S=\frac{A+A^{\mathrm{T}}}{2}$ is positive definite. Since

$$(A_i)_S=\begin{cases} A_0, & i=0, \\ A_0, & i=1, \\ \left(1+\frac{c_2}{2}\right)A_0, i=2, \end{cases}$$

we have all the eigenvalues of $\frac{A_i+A_i^{\mathrm{T}}}{2}$ (i = 0, 1, 2) are the following:

$$\lambda\left((A_i)_S\right)_{p,q}=\begin{cases} \lambda(A_0)_{p,q}=4+2\sqrt{1-c_1^2}\left(\cos\left(\frac{p\pi}{n+1}\right)+\cos\left(\frac{q\pi}{n+1}\right)\right)>0, \qquad i=0,1, \\ \lambda\left(\left(1+\frac{c_2}{2}\right)A_0\right)_{p,q}=\left(1+\frac{c_2}{2}\right)\left(4+2\sqrt{1-c_1^2}\left(\cos\left(\frac{p\pi}{n+1}\right)+\cos\left(\frac{q\pi}{n+1}\right)\right)\right)>0, i=2, \end{cases} \quad p,q=1,2,\cdots,n. \tag{8}$$

Therefore $(A_i)_S$ (i = 0, 1, 2) is positive definite, which means the discretization matrices A_0, A_1 and A_2 of Equation (1) are positive definite.

2.3. Diagonal Dominance

For all the discretization matrices A_0, A_1 and A_2 of Equation (1),

$$\sum_{i\neq j,\, i=1}^{n}\left|a_{ij}\right| \le \begin{cases} 2+2=4=\left|a_{jj}\right|, & A_0=\left(a_{ij}\right); \\ 2(1+c)+2(1-c)=4=\left|a_{jj}\right|, & A_1=\left(a_{ij}\right); \\ 2(1+c)+2=4+2c=\left|a_{jj}\right|, & A_2=\left(a_{ij}\right). \end{cases}$$

Therefore, the discretization matrices A_0, A_1 and A_2 of Equation (1) are diagonally dominant.

2.4. Symmetriness

It is easy to see that only A_0 is symmetric.

3. Stationary Iteration Methods and Krylov Subspace Methods

The goal of this section is to find an efficient iterative method for the solution of the linear system $A_i x = b, i = 0,1,2$. Since $A_i\left(i=0,1,2\right)$ are positive definite and diagonally dominant, we would use fixed point methods and Krylov subspace methods. In this section, first find the suitable convergence tolerance, later use numerical experiments to compare the convergence speed of various iteration methods.

3.1. Convergence Tolerance

Without loss of generality, take $b = Ax_*, x_* = \left(1,1,\cdots,1\right)^{\mathrm{T}}$, convergence tolerance $\delta = \dfrac{10^{-3}}{\|b\|_2 \cdot \left\|A_i^{-1}\right\|_2}$ and assume that $\dfrac{\|Ax_i - b\|_2}{\|b\|_2} \le \delta$, then

$$\begin{aligned}\|x_i - x_*\|_2 &= \left\|A^{-1}\cdot\left(Ax_i - b\right)\right\|_2 \le \left\|A^{-1}\right\|_2 \cdot \|Ax_i - b\|_2 \\ &= \|b\|_2 \cdot \left\|A^{-1}\right\|_2 \cdot \left(\frac{\|Ax_i - b\|_2}{\|b\|_2}\right) \le \delta \cdot \|b\|_2 \cdot \left\|A^{-1}\right\|_2 = 10^{-3}.\end{aligned}$$

In order to achieve $\|x_i - x_*\|_2 \le 10^{-3}$ in the numerical experiments, we would set convergence tolerance $\delta = \dfrac{10^{-3}}{\|b\|_2 \cdot \left\|A_i^{-1}\right\|_2}$.

3.2. Numerical Experiments

Computational results using fixed point methods such as Jacobi, Gauss-Seidel, SOR etc. and projection methods such as PCG, BICG, BICGSTAB, CGS, GMRES and QMR are listed out in figures (**Figures 1-21**), for all three different γ values $\left(\gamma = 0,16,64\right)$. The projection methods are performed with different preconditioning methods such as Jacobi preconditioning, luinc and cholinc preconditioning.

The tables (**Table 1** and **Table 2**) containing relevant computational details are also given below.

4. Conclusions

From the figures (**Figures 1-21**) and tables (**Table 1** and **Table 2**), we obtain the following conclusions:

- The convergence speeds of SOR, Backward SOR and SSOR are faster than that of GS and Backward GS; while GS and Backward GS are faster than Jacobi;
- If matrix *A* is symmetric $\left(\gamma = 0\right)$, the convergence speeds of SOR, Backward SOR and SSOR are the same; otherwise $\left(\gamma = 16,64\right)$, SSOR is faster than Backward SOR and SOR;
- The convergence speeds of SOR and Backward SOR are the same, also for GS and Backward GS;
- From **Table 2**, the iteration steps and the time for all fixed point methods of the case in which $\gamma = 64$ are less than the case in which $\gamma = 16$, and also the case in which $\gamma = 16$ are less than the case in which $\gamma = 0$;
- The upwind difference method is more suitable to be applied to convection dominant problem than the cen-

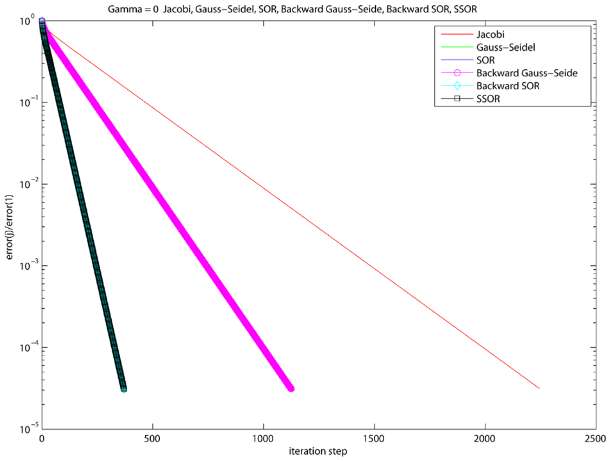

Figure 1. Comparison of convergence speed using fixed point methods when $\gamma = 0$.

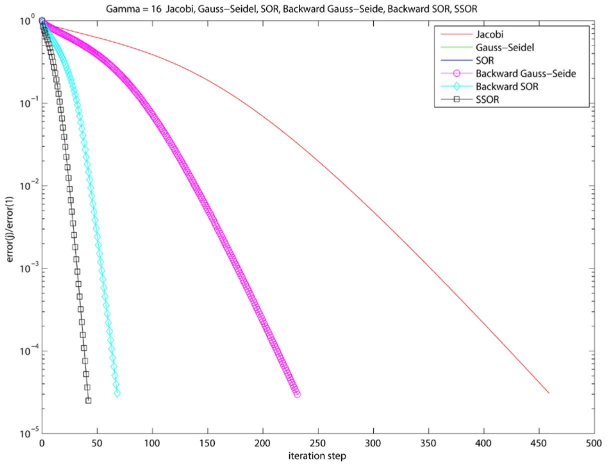

Figure 2. Comparison of convergence speed using fixed point methods when $\gamma = 16$.

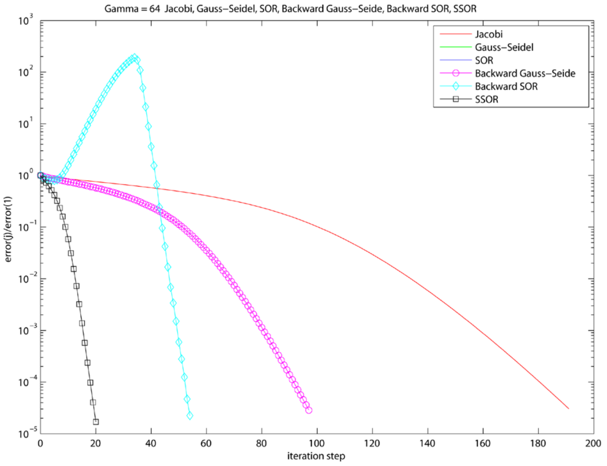

Figure 3. Comparison of convergence speed using fixed point methods when $\gamma = 64$.

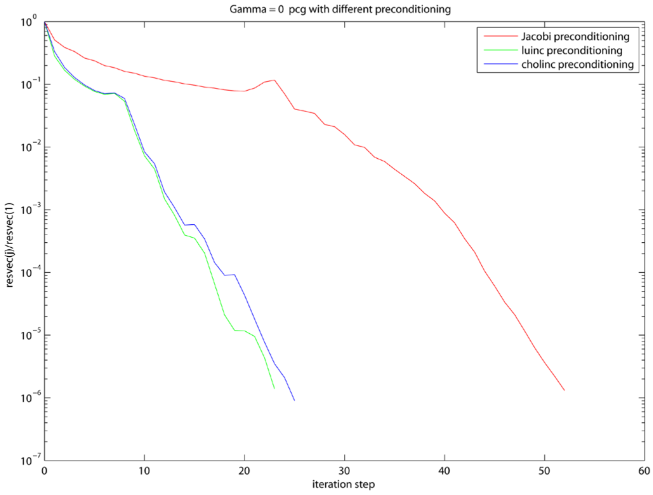

Figure 4. PCG with different preconditionings when $\gamma = 0$.

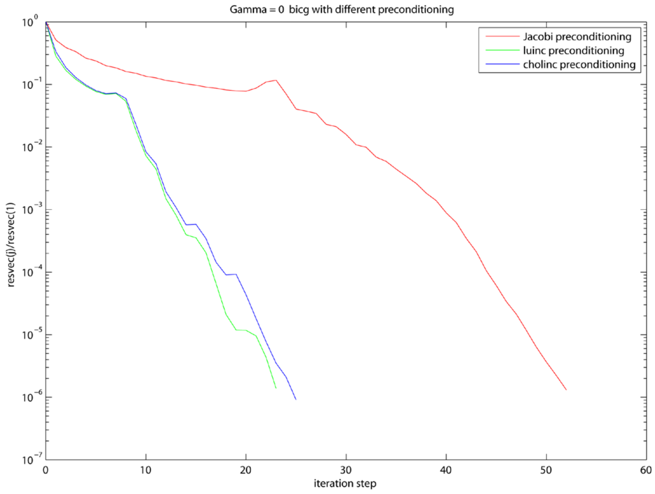

Figure 5. BICG with different preconditionings when $\gamma = 0$.

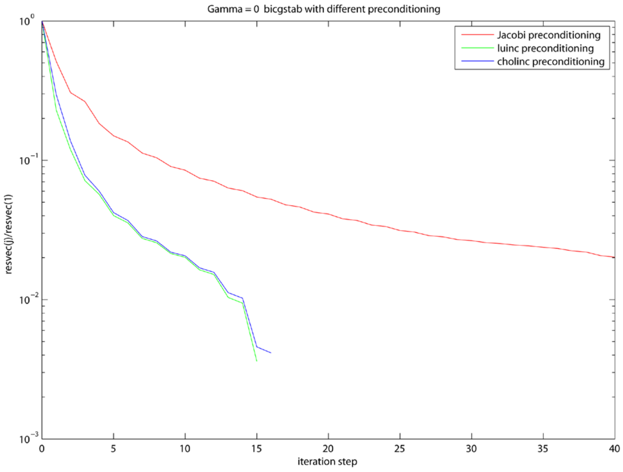

Figure 6. BICGSTAB with different preconditionings when $\gamma = 0$.

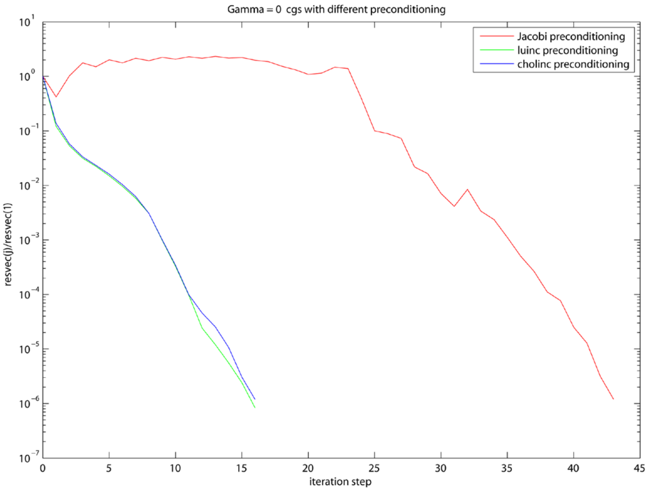

Figure 7. CGS with different preconditionings when $\gamma = 0$.

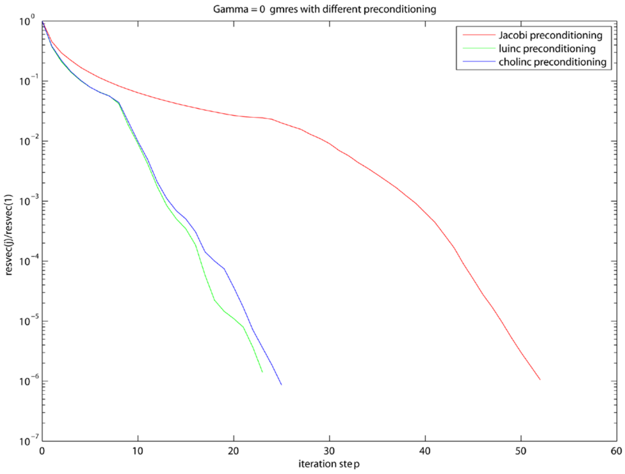

Figure 8. GMRES with different preconditionings when $\gamma = 0$.

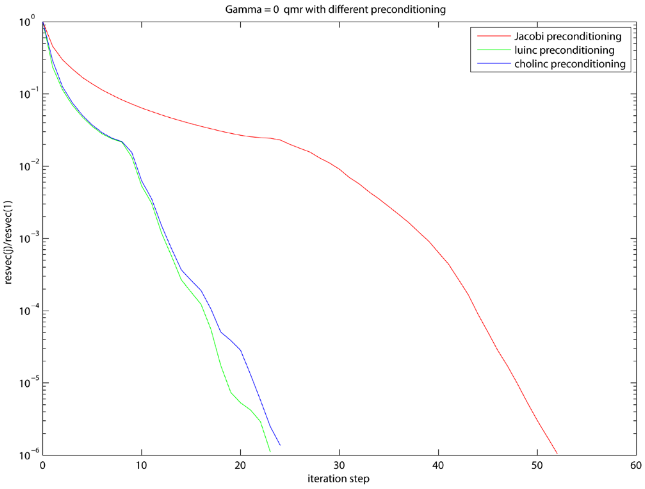

Figure 9. QMR with different preconditionings when $\gamma = 0$.

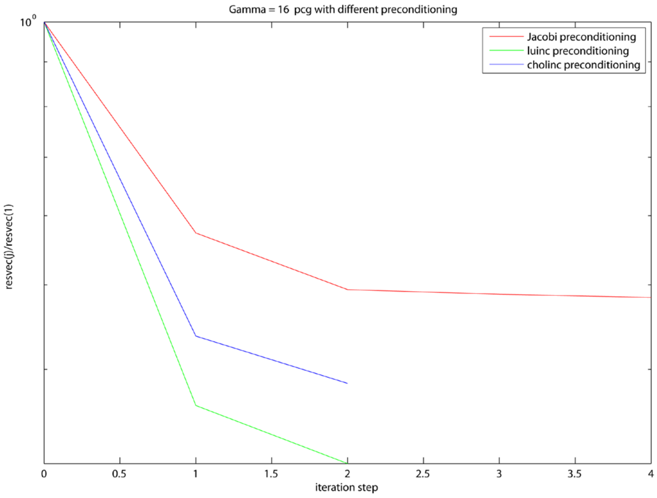

Figure 10. PCG with different preconditionings when $\gamma = 16$.

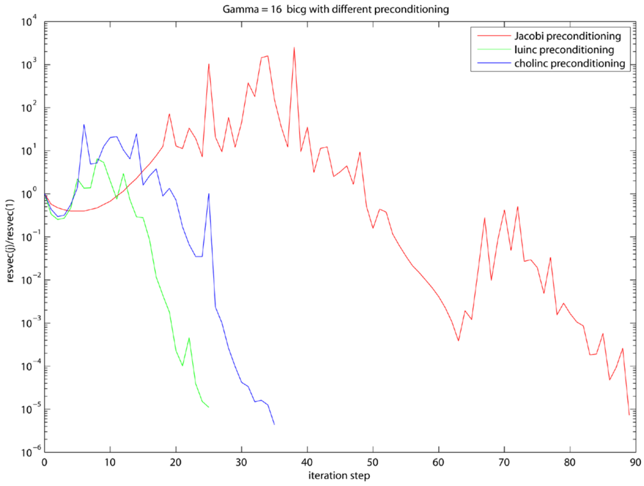

Figure 11. BICG with different preconditionings when $\gamma = 16$.

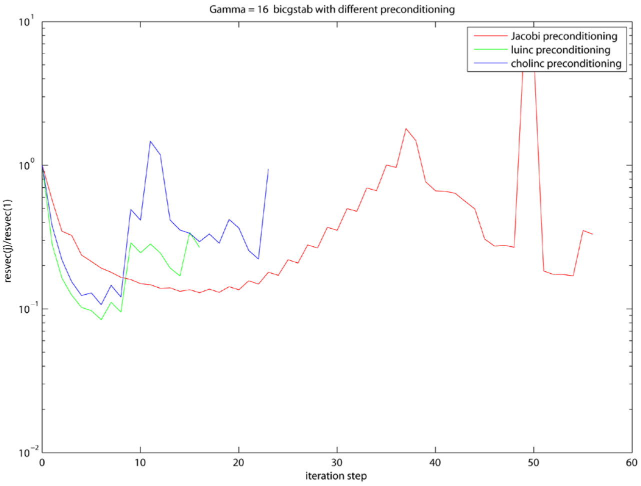

Figure 12. BICGSTAB with different preconditionings when $\gamma = 16$.

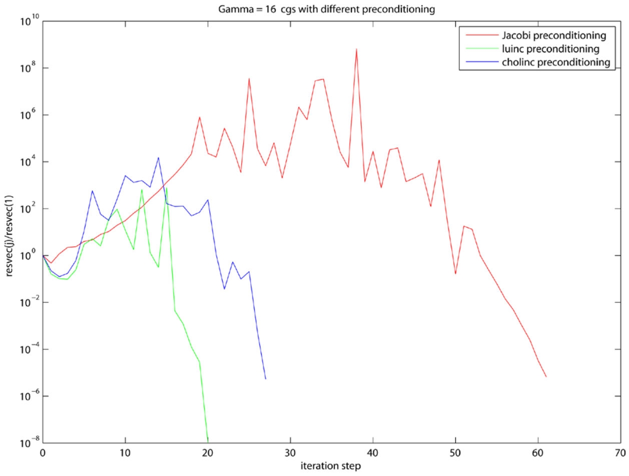

Figure 13. CGS with different preconditionings when $\gamma = 16$.

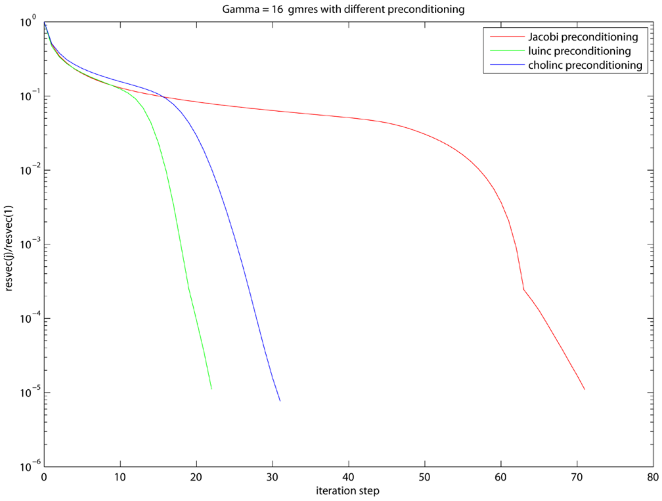

Figure 14. GMRES with different preconditionings when $\gamma = 16$.

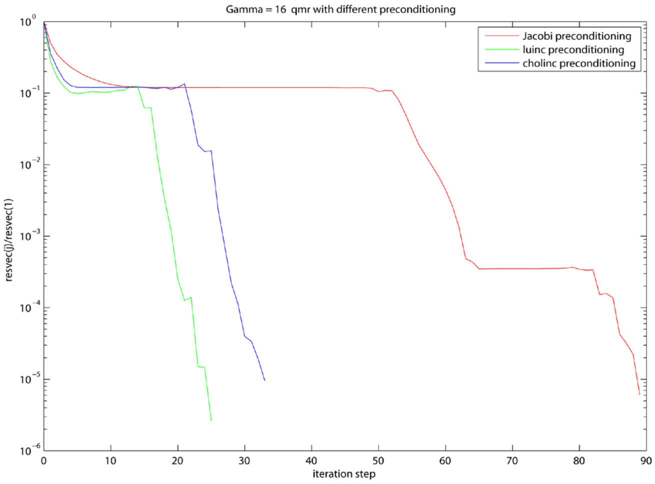

Figure 15. QMR with different preconditionings when $\gamma = 16$.

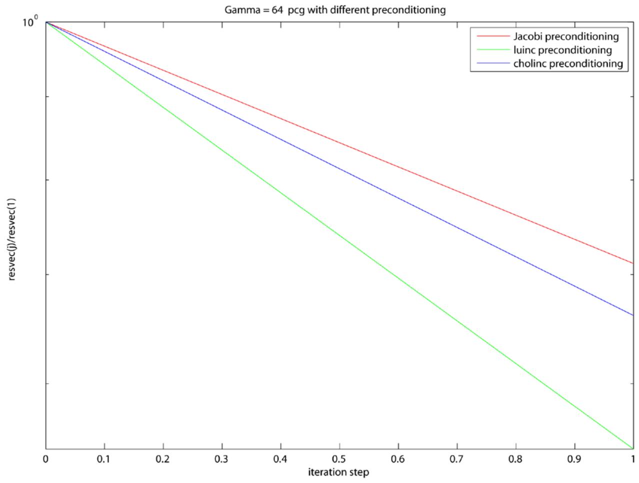

Figure 16. PCG with different preconditionings when $\gamma = 64$.

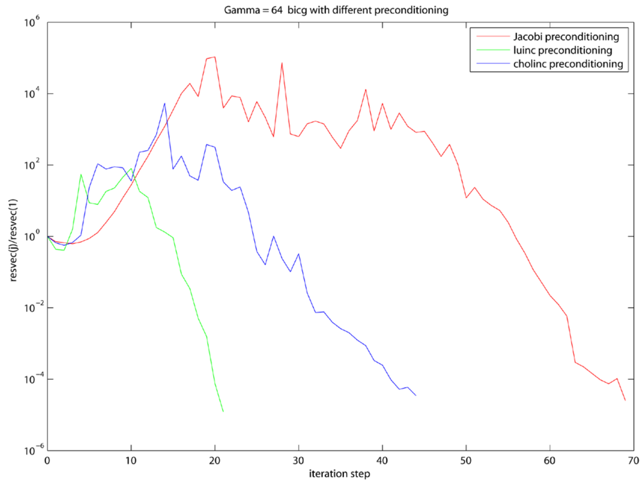

Figure 17. Bicg with different preconditionings when $\gamma = 64$.

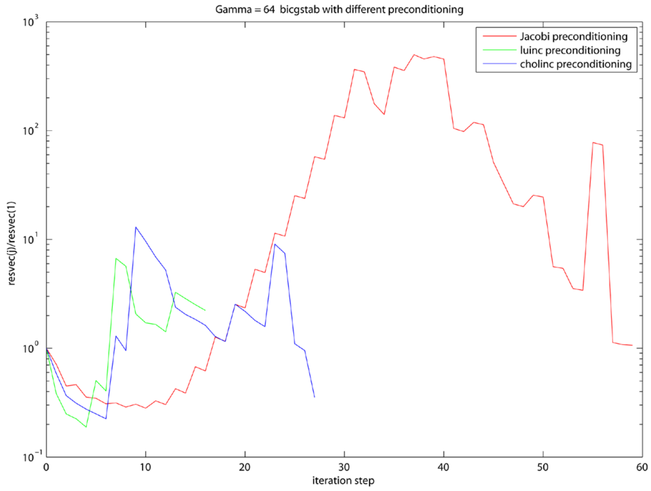

Figure 18. BICGSTAB with different preconditionings when $\gamma = 64$.

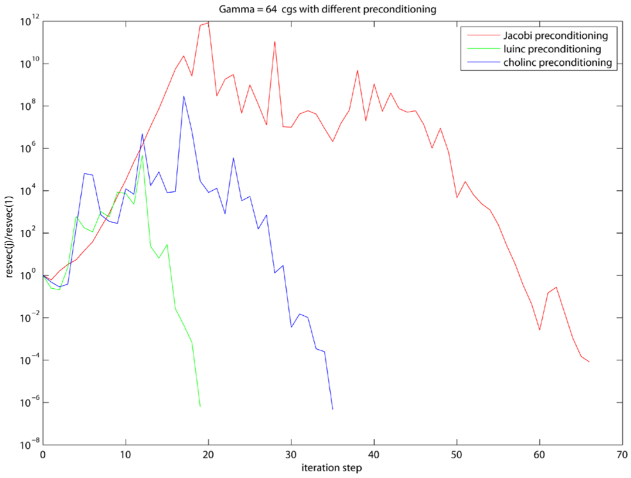

Figure 19. CGS with different preconditionings when $\gamma = 64$.

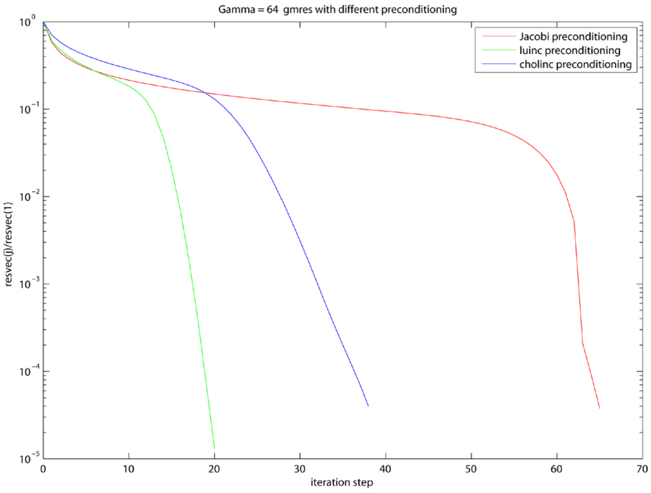

Figure 20. GMRES with different preconditionings when $\gamma = 64$.

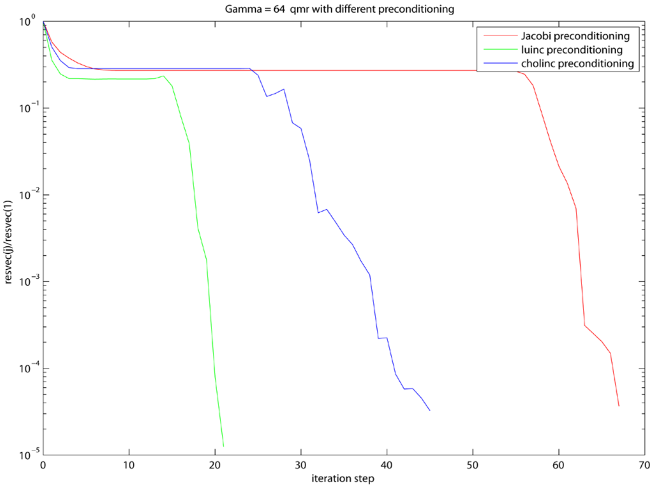

Figure 21. QMR with different preconditionings when $\gamma = 64$.

Table 1. Computational details of different projection methods with different preconditioning.

Gamma	Function	Preconditioning	flag	No. of iterations	Relres	Delta (tol)
0	PCG	jacobi	0	52	1.31E−06	1.55E−06
		luinc	0	23	1.38E−06	
		cholinc	0	25	9.05E−07	
	BICG	jacobi	0	52	1.31E−06	
		luinc	0	23	1.38E−06	
		cholinc	0	25	9.05E−07	
	BICGSTAB	jacobi	0	40.5	1.42E−06	
		luinc	0	15.5	9.36E−07	
		cholinc	0	16	1.54E−06	
	CGS	jacobi	0	43	1.20E−06	
		luinc	0	16	8.27E−07	
		cholinc	0	16	1.18E−06	
	GMRES	jacobi	0	49	1.32E−06	
		luinc	0	23	9.01E−07	
		cholinc	0	24	1.21E−06	
	QMR	jacobi	0	52	1.05E−06	
		luinc	0	23	1.12E−06	
		cholinc	0	24	1.37E−06	
16	PCG	jacobi	1	4	0.483607757	1.14E−05
		luinc	1	2	0.312145391	
		cholinc	1	2	0.38563672	
	bicg	jacobi	0	89	7.33E−06	
		luinc	0	25	1.11E−05	
		cholinc	0	35	4.37E−06	
	BICGSTAB	jacobi	0	56.5	1.03E−05	
		luinc	0	16.5	6.30E−06	
		cholinc	0	23.5	3.98E−06	
	CGS	jacobi	0	61	6.61E−06	
		luinc	0	20	1.01E−08	
		cholinc	0	27	5.29E−06	
	GMRES	jacobi	0	68	9.70E−06	
		luinc	0	22	7.08E−06	
		cholinc	0	30	1.10E−05	
	QMR	jacobi	0	89	6.43E−06	
		luinc	0	25	2.64E−06	
		cholinc	0	33	9.57E−06	

Continued

64	PCG	jacobi	1	1	0.710978037	4.13E−05
		luinc	1	1	0.547110057	
		cholinc	1	1	0.660611555	
	BICG	jacobi	0	70	3.15E−05	
		luinc	0	21	1.21E−05	
		cholinc	0	44	3.47E−05	
	BICGSTAB	jacobi	0	59.5	2.12E−05	
		luinc	0	16.5	5.27E−06	
		cholinc	0	27.5	1.60E−05	
	CGS	jacobi	1	87	0.000123357	
		luinc	0	19	6.36E−07	
		cholinc	0	35	3.14E−06	
	GMRES	jacobi	0	63	2.66E−05	
		luinc	0	19	2.50E−05	
		cholinc	0	37	3.11E−05	
	QMR	jacobi	0	69	3.56E−05	
		luinc	0	21	1.25E−05	
		cholinc	0	45	3.31E−05	

Table 2. Computational details of different fixed point methods.

	The fixed point method	No. of iterations
Gamma = 0	Jacobi	2246
	Gauss-Seidel	1124
	SOR	370
	Backward Gauss-Seidel	1124
	Backward SOR	370
	SSOR	370
Gamma = 16	Jacobi	459
	Gauss-Seidel	231
	SOR	68
	Backward Gauss-Seidel	231
	Backward SOR	68
	SSOR	42
Gamma = 64	Jacobi	191
	Gauss-Seidel	97
	SOR	54
	Backward Gauss-Seidel	97
	Backward SOR	54
	SSOR	20

tral difference method;

- The convergence speed of the six projection methods including PCG, BICG, BICGSTAB, CGS, GMRES and QMR under luinc preconditioning are faster than under cholinc preconditioning, while under cholinc preconditioning are faster than Jacobi preconditioning;
- The six projection methods under Jacobi, luinc and cholinc are convergent when $\gamma = 0$, however, for $\gamma = 16, 64$, the PCG method are not convergent and also the CGS method under Jacobi preconditioning are not convergent when $\gamma = 64$.

Acknowledgements

I thank the editor and the referee for their comments. I would like to express deep gratitude to my supervisor Prof. Dr. Mark A. Peletier whose guidance and support were crucial for the successful completion of this paper. This work was completed with the financial support of Foundation of Guangdong Educational Committee (2014KQNCX161, 2014KQNCX162).

References

[1] Saad, Y. (2003) Iterative Methods for Sparse Linear Systems. Siam, Bangkok. http://dx.doi.org/10.1137/1.9780898718003

[2] Varga, R.S. (2009) Matrix Iterative Analysis. Volume 27, Springer Science & Business Media, Heidelberger.

[3] Young, D.M. (2014) Iterative Solution of Large Linear Systems. Elsevier, Amsterdam.

[4] Lanczos, C. (1952) Solution of Systems of Linear Equations by Minimized Iterations. *Journal of Research of the National Bureau of Standards*, **49**, 33-53. http://dx.doi.org/10.6028/jres.049.006

[5] Hestenes, M.R. and Stiefel, E. (1952) Methods of Conjugate Gradients for Solving Linear Systems.

[6] Walker, H.F. (1988) Implementation of the GMRES Method Using Householder Transformations. *SIAM Journal on Scientific and Statistical Computing*, **9**, 152-163. http://dx.doi.org/10.1137/0909010

[7] Sonneveld, P. (1989) CGS, a Fast Lanczos-Type Solver for Nonsymmetric Linear Systems. *SIAM Journal on Scientific and Statistical Computing*, **10**, 36-52. http://dx.doi.org/10.1137/0910004

[8] Freund, R.W. and Nachtigal, N.M. (1991) QMR: A Quasi-Minimal Residual Method for Non-Hermitian Linear Systems. *Numerische Mathematik*, **60**, 315-339. http://dx.doi.org/10.1007/BF01385726

[9] van der Vorst, H.A. (1992) Bi-CGSTAB: A Fast and Smoothly Converging Variant of Bi-CG for the Solution of Nonsymmetric Linear Systems. *SIAM Journal on Scientific and Statistical Computing*, **13**, 631-644. http://dx.doi.org/10.1137/0913035

[10] Brezinski, C., Zaglia, M.R. and Sadok, H. (1992) A Breakdown-Free Lanczos Type Algorithm for Solving Linear Systems. *Numerische Mathematik*, **63**, 29-38. http://dx.doi.org/10.1007/BF01385846

[11] Chan, T.F., Gallopoulos, E., Simoncini, V., Szeto, T. and Tong, C.H. (1994) A Quasi-Minimal Residual Variant of the Bi-CGSTAB Algorithm for Nonsymmetric Systems. *SIAM Journal on Scientific Computing*, **15**, 338-347. http://dx.doi.org/10.1137/0915023

[12] Gutknecht, M.H. (1992) A Completed Theory of the Unsymmetric Lanczos Process and Related Algorithms, Part I. *SIAM Journal on Matrix Analysis and Applications*, **13**, 594-639. http://dx.doi.org/10.1137/0613037

[13] Gutknecht, M.H. (1994) A Completed Theory of the Unsymmetric Lanczos Process and Related Algorithms. Part II. *SIAM Journal on Matrix Analysis and Applications*, **15**, 15-58. http://dx.doi.org/10.1137/S0895479890188803

[14] Eisenstat, S.C. (1981) Efficient Implementation of a Class of Preconditioned Conjugate Gradient Methods. *SIAM Journal on Scientific and Statistical Computing*, **2**, 1-4. http://dx.doi.org/10.1137/0902001

[15] Meijerink, J.V. and van der Vorst, H.A. (1977) An Iterative Solution Method for Linear Systems of Which the Coefficient Matrix Is a Symmetric M-Matrix. *Mathematics of Computation*, **31**, 148-162.

[16] Ortega, J.M. (1988) Efficient Implementations of Certain Iterative Methods. *SIAM Journal on Scientific and Statistical Computing*, **9**, 882-891. http://dx.doi.org/10.1137/0909060

[17] Fowler, A.C. (1997) Mathematical Models in the Applied Sciences. Volume 17, Cambridge University Press, Cambridge.