Technical Report for Iterative Subspace Projection Methods for Large scale Linear Systems and Eigenvalue Problems

Matthew Cheung

14 May 2019

Abstract

The Steepest Descent (SD), Conjugate Gradient (CG), Minimal Residual (MR) and Restarted GMRES methods were implemented and the convergence behaviors of these methods were studied for a variant of problems. Furthermore, the eigenvalues accuracies were evaluated for different configurations of the Arnoldi method. The results indicate the required properties of each method are upheld (i.e methods requiring symmetric positive definite (SPD) matrices do not converge for non-SPD), the eigenvalue accuracy for the Arnoldi method increases with increasing iteration, the Ritz values from the Hessenberg matrix are good approximations for the eigenvalues, the non-reorthogonalized Arnoldi method significantly degrades orthogonality of resulting matrices but keep the same eigenvalues, and the shifted-inverse Arnoldi iteration converges to a target faster when specified.

1 Introduction

Iterative subspace projection methods are used for solving large sparse linear systems (i.e. Ax = b). We implement the Steepest Descent (SD), Conjugate Gradient (CG), Minimal Residual (RM) and Restarted GMRES algorithms to study the convergence behaviors of these methods for the matrices bcsstk15, mahindas, nos3 and west0479 from The University of Florida Sparse Matrix Collection. The matrices are shown below.

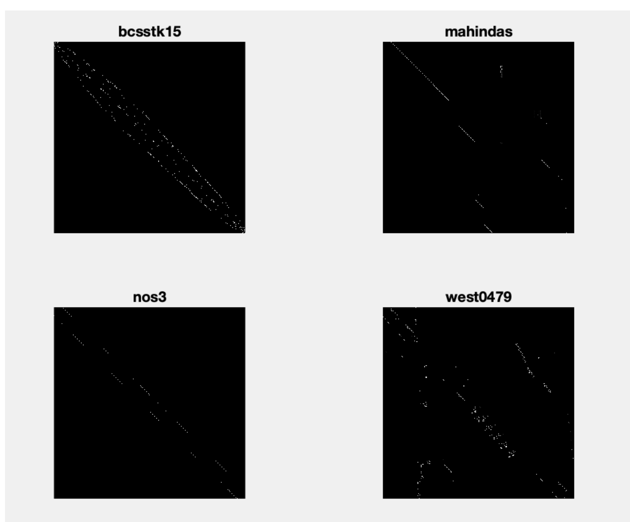

Figure 1: Four Test Matrices from The University of Florida Sparse Matrix Collection

The investigated parameters in this report will be the residuals and the eigenvalues.

2 Methods

2.1 Part I: Iterative subspace projection methods for large scale linear systems

In this section, functions for each of the methods were written using MATLAB. A pseudo code implementation will be presented in the main report and the code will be provided in the appendix. These will be verified using the built in methods (i.e. gmres, cg). Then the implementations will be run on the test matrices.

2.1.1 Steepest Descent (SD)

The Steepest Descent (SD) method solves Ax = b for symmetric positive definite (SPD) matrices. The algorithm is presented in Subspace projection methods for LS Lecture on slide 16. [1]

Pick initial guess x_0
$For\ k = 0, 1, 2, ...until\ convergence\ do$
$r_k = b - Ax_k$
$\alpha_k = \frac{r_k^T r_k}{r_k^T A r_k}$
$x_{k+1} = x_k + \alpha_k r_k$

A is SPD only when $r_k > 0$. When $r_k = 0$, the algorithm has converged. Therefore, the stopping criterion is when $r_k \leq \epsilon$, where ϵ is machine precision, 2.2204e-16.

2.1.2 Conjugate Gradient (CG)

The Conjugate Gradient (CG) method solves Ax = b for SPD A. Theoretically, it yields an exact solution in n steps. The algorithm from *Linear Algebra and Learning from Data by Gilbert Strang* is shown below. [2]

$x_0 = 0, r_0 = b, d_0 = r_0$
$for\ k = 1\ to\ N$
$\alpha_k = \frac{r_{k-1}^T r_{k-1}}{d_{k-1}^T A d_{k-1}}$
$x_k = x_{k-1} + \alpha d_{k-1}$
$r_k = r_{k-1} - \alpha_k A d_{k-1}$
$\beta_k = \frac{r_k^T r_k}{r_{k-1}^T r_{k-1}}$
$d_k = r_k + \beta_k d_{k-1}$

The conjugate method minimizes the error $||x - x_k||_S$ over the kth Krylov subspace, where x_k is the orthogonalized subspace K_m. The stopping criterion is when $r_k \leq \epsilon$, where ϵ is machine precision, 2.2204e-16.

2.1.3 Minimal Residual (MR)

The Minimal Residual (MR) Iteration Method solves Ax = b for a nonsymmetric and nonsingular matrix A. This is similar to the SD method. But in this case, $\alpha_k = \frac{r_k^T A^T r_k}{r_k^T A^T A r_k}$. he algorithm is presented in Subspace projection methods for LS Lecture on slide 21. [1]

Pick initial guess x_0
$For\ k = 0, 1, 2, ...until\ convergence\ do$
$r_k = b - Ax_k$
$\alpha_k = \frac{r_k^T A^T r_k}{r_k^T A^T A r_k}$
$x_{k+1} = x_k + \alpha_k r_k$

Each iteration minimizes $f(x) \equiv ||r||_2^2 = ||b - Ax||_2^2$. The stopping criterion is when $r_k \leq \epsilon$, where ϵ is machine precision, 2.2204e-16.

2.1.4 Arnoldi and Restarted GMRES

The Restarted GMRES (GMRES(k)) uses Krylov subspaces as pair of projection subspaces. It is a generalization of the one-dimensional MR iteration. The Krylov subspaces are generated and orthogonalized using the Arnoldi Iteration. This can be represented by the equations

$$AV_m = V_m H_m + h_{m+1,m} v_{m+1} e_m^T = V_{m+1} \hat{H}_m$$

$$V_{m+1} = \begin{bmatrix} V_m & v_{m+1} \end{bmatrix}$$

$$\hat{H}_m = \begin{bmatrix} H_m \\ h_{m+1,m} e_m^T \end{bmatrix}$$

where A has dimensions $n \times n$, V_m has dimensions $n \times m$, V_{m+1} has dimensions $n \times (m+1)$, the Hessenberg H_m has dimensions $m \times m$, $\hat{H}_m$ has dimensions $(m+1) \times m$, $h_{m+1,m}$ is a scalar, v_{m+1} is a columns vector of dimensions $(m+1) \times 1$ and e_m is a unit vector of size $m \times 1$.

The Arnoldi algorithm is presented in Subspace projection methods for LS Lecture on slide 25.[1]

v$_1 = \frac{v}{||v||}$
$for\ j = 1, 2, ..., m$
  $w = Av_j$
  $for\ i = 1, 2, ..., j$
    $h_{ij} = v_i^T w$
    $w = w - h_{ij} v_i$
  end
  $h_{j+1,j} = ||w||_2$
  $if\ h_{j+1,j} = 0, stop$
  $v_{j+1} = \frac{1}{h_{j+1,j}}$

$H_k = Q_k^T A Q_k$ is the projection of A onto the Krylov space using the columns of Q. [2] A version for re-orthogonalization was implemented.

The GMRES algorithm is presented in Subspace projection methods for LS Lecture on slide 19.[1]

for i = 1,2,... until convergence do
  $r_0 = b - Ax_0$
  $\beta = ||r_0||_2$
  $v1 = \frac{r0}{\beta}$
  $arnoldi(A, v1, m)$
  $solve\ min_y ||\beta e_1 H_m y||_2$
  $x_m = x_0 + V_m y_m$
  $test\ for\ convergence,\ if\ satisfied,\ then\ stop$
  $set\ x_0\ :=\ x_m\ and\ go\ to\ 1$

The least squares problem provides a solution. The Arnoldi procedure breaks down when $h_{j+1,j} = 0$ at some step j and this is when x_j is the exact solution of the linear system Ax = b.

2.1.5 Residuals

The residuals for the Steepest Descent, Minimal Residuals and Conjugate Gradient methods were computed using the formula

$$r = \frac{||r_0||_2}{||A||_1 ||x||_2 + ||b||_2}$$

The residuals for the Conjugate Gradient methods were computed using the formula

$$r = \frac{||\hat{H}_m y - \beta e_1||_2}{||b||_2}$$

2.2 Part II: Iterative subspace projection methods for large scale eigenvalue problems

In this part, the Arnoldi method with and without reorthogonalization are used to to compute eigenpairs of large sparse matrix A (west0479) for different iterations. This was compared to the "exact" eigenvalues computed using the eig function in MATLAB. The real and imaginary values are plotted and discussed.

2.2.1 Reorthogonalization/No Reorthogonalization and Residual Comparison

The Arnoldi (with Reorthogonalization) code is modified so that he reorthogonalization is not done. The correctness of the code is verified using $||AV_m - V_{m+1}\hat{H}_m||_2$. The orthogonality is evaluated using the residual $||I - V_{j+1}^H V_{j+1}||_2$.

2.2.2 Ritz Values

A method to attain eigenvalues is to use the eigenvalues of the Hessenberg matrix H_j. These are computed using the eig command. The Ritz values and the exact eigenvalues together. In theory, the outlying eigenvalues are good approximations for the Ritz values.

2.2.3 Arnoldi with Shift and Inversion

The shift and invert spectral transform are used to find the eigenvalues closest to a target. Instead of multiplying the initial vector by A, the vector is multiplied by $(A - \tau I)^{-1}$, where τ is the target. This is similar to the power method with shift and inversion. The eigenvalues problem now becomes

$$(A - \tau I)^{-1} x = \frac{1}{\lambda - \sigma} x$$

The eigenvalue (λ_j) closest to τ of the shifted and inverted problem $\sigma = \frac{1}{\lambda_j - \tau}$ is now dominant and thus leads to a smaller residual. The residuals can be proved to be [3]

$$||r_j|| \leq h_{m+1,m} |\lambda_j - \tau| \; ||A - \tau b|| \; |e_m^T y_j|$$

From this form, we see that the residual (r_j) is bounded by zero if λ is a good estimate for τ.

3 Numerical Experiment Results

3.0.1 Steepest Descent (SD)

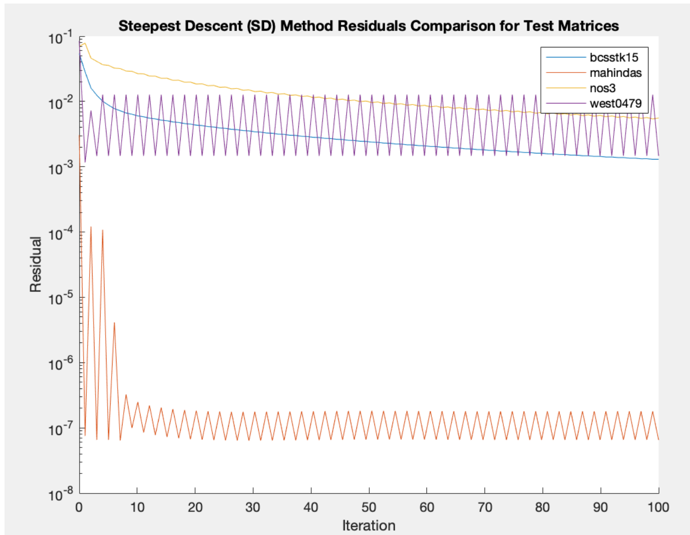

Figure 2: Steepest Descent Method Comparison for 4 test matrices

From the plot, we see that only the bcsstk15 and the nos3 display convergent behavior. This is expected because mahindas and west0479 are non SPD matrices.

3.0.2 Conjugate Gradient (CG)

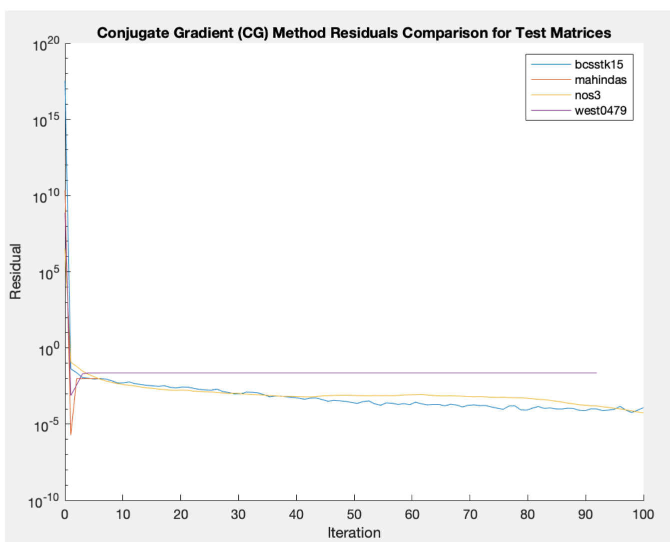

Figure 3: Conjugate Gradient (CJ) Method Comparison for 4 test matrices

Similar to the Steepest Descent method, from the plot, we see that only the bcsstk15 and the nos3 display convergent behavior. This is expected because mahindas and west0479 are non SPD matrices.

3.0.3 Minimal Residual (MR)

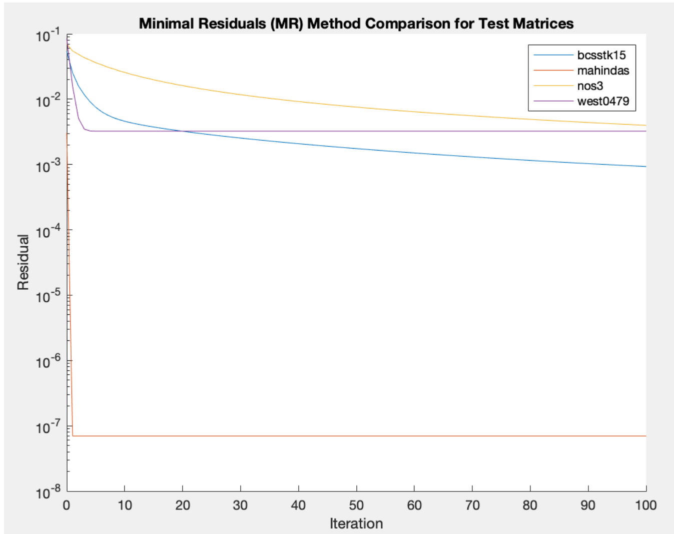

Figure 4: Minimal Residual (MR) Method Comparison for 4 test matrices

From the plot, we see that only bcsstk15, nos3 and west0479 are converging. mahindas is not because it is non invertible. This can be seen by taking the determinant of mahindas, which is computed to be -1.0008e-20.

3.0.4 Restarted GMRES

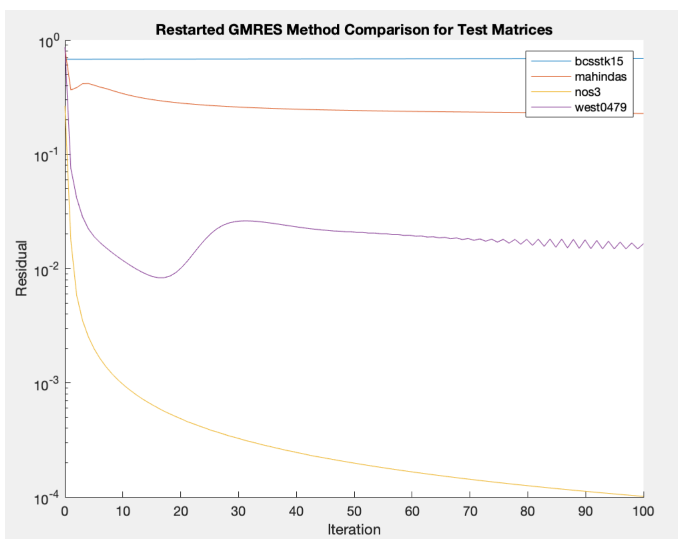

Figure 5: Restarted GMRES Method Comparison for 4 test matrices

From the plot, we see that only nos3, mahindas and west0479 are converging. Interestingly, bcsstk15 does not appear to converge quickly.

3.0.5 Reorthogonalization/No Reorthogonalization and Residual Comparison

To check my correctness of the Reorthogonalized Arnoldi Algorithm, $||AV_m - V_{m+1}\hat{H}_m||_2$ was computed for 30 iterations to be 2.6297e-12. Similarly, for the Arnoldi without reorthogonalization, $||AV_m - V_{m+1}\hat{H}_m||_2$

was computed to be 2.2025e-12. Both of these are approximately zero, despite being 4 orders of magnitude smaller than machine precision.

The orthogonality was evaluated using the residual $||I - V_{j+1}^H V_{j+1}||_2$. At 30 iterations, these were computed to be 1.1814e-15 for the Reorthogonalized Arnoldi Algorithm and 4.3886e-13 for the Non-Reorthogonalized Algorithm. Also, at 60 iterations, the Non-Reorthogonalized Algorithm residual the residual $||I - V_{j+1}^H V_{j+1}||_2$ was 8.9316e-12. From these results, it is evident that V_{j+1} loses orthogonality. Therefore, in practice, it is a good idea to implement the reorthogonalization. Furthermore, the residuals were computed to be 2.2025e-12 and 7.0673e-11 for 30 and 60 iterations.

3.0.6 Ritz Values

The Ritz values were computed using the eig command and compared to the exact eigenvalues.

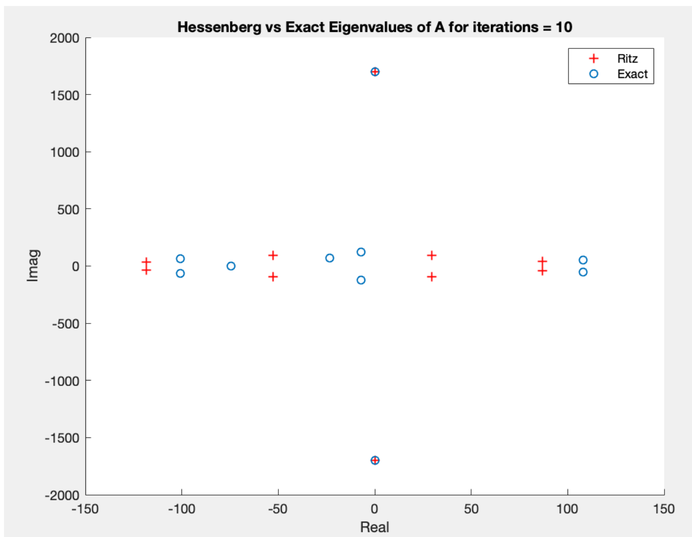

Figure 6: Hessenberg vs Exact Eigenvalues for iterations = 10

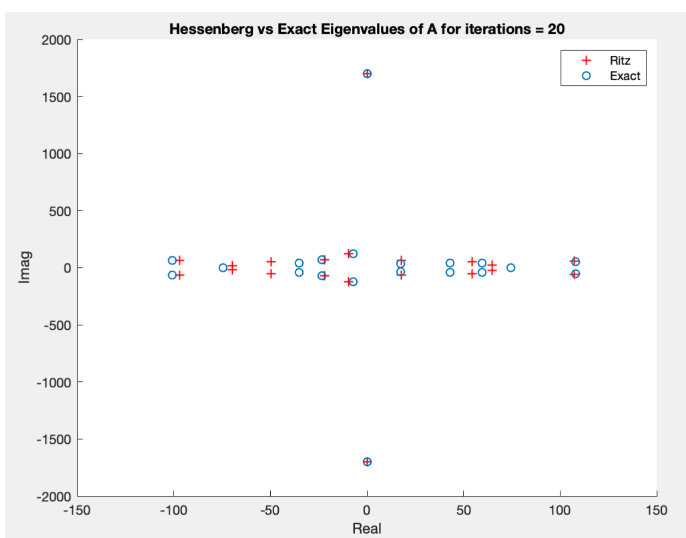

Figure 7: Hessenberg vs Exact Eigenvalues for iterations = 20

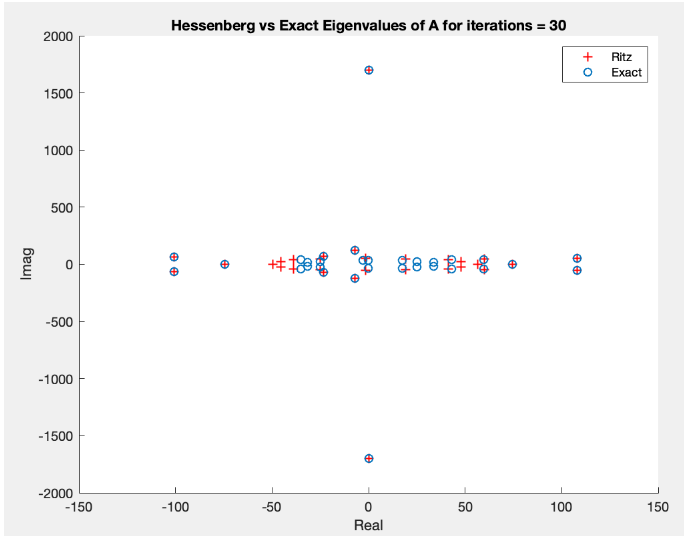

Figure 8: Hessenberg vs Exact Eigenvalues for iterations = 30

From the figures above, the eigenvalues are approximated well by the Ritz values. However, the number of iterations does not improve the eigenvalue approximations. The norm of the difference between the Reorthogonalized and the Non-Reorthogonalized eigenvalues were computed to be 528.5324 and 275.3189 for both algorithms. The shift-inversion spectral transformation converges to the dominant eigenvalue. The implementation was unsuccessful and did not yield good results.

4 Acknowledgements

I would like to thank Professor Zhaojun Bai who provided me with a wealth of knowledge about Numerical Linear Algebra techniques to undertake investigation.

5 References

[1] Bai, Zhaojun. "Subspace Projection Methods for LS." ECS231 Slides, web.cs.ucdavis.edu/ bai/ECS231/Slides/ls.pdf.

[2] Strang, Gilbert. Linear Algebra and Learning from Data. Wellesley-Cambridge Press, 2019.

[3] Jia, Zhongxiao, and Yong Zhang. "A Refined Shift-and-Invert Arnoldi Algorithm for Large Unsymmetric Generalized Eigenproblems." Computers amp; Mathematics with Applications, vol. 44, no. 8-9, 2002, pp. 1117–1127., doi:10.1016/s0898-1221(02)00220-1.