FIS Project 3

Eigensolver

Ruei-Bo Chen 416082

1. Introduction

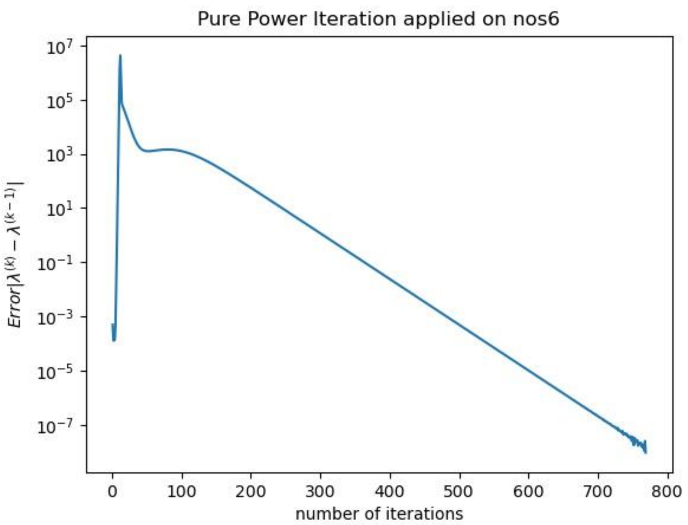
This report covers the application of Lanczos Algorithm and Power Iteration to access the largest eigenvalue of the huge sparse s.p.d. (symmetric positive-definite) matrixes.

First, we use the pure power iteration to obtain the largest eigenvalue of a smaller matrix. We will show the eigenvalue, elapsed time, and the final error in the next chapter.

Next, we utilize the Lanczos algorithm to improve the efficiency. Lanczos algorithm is a modified version of Arnoldi. In the Project 1, we know that Arnoldi method is used to construct the Hessenberg matrix. In this project, all of the input matrixes are s.p.d. Therefore, the Hessenberg matrix of the s.p.d. matrix will be symmetric as well. With the property of Hessenberg matrix and the symmetry combined, one can easily conclude that the Hessenberg matrix of a s.p.d. matrix will be tridiagonal. Moreover, one can prove that the result of power iteration on this tridiagonal matrix will be close to it applied on the original s.p.d. matrix. Thus we can obtain the largest eigenvalue with a much improved efficiency.

2. Result

(1) Pure power iteration



Pure power iteration on Nos6.mtx	
Final Error	9.313e-09
Largest Eigenvalue	7.650603e+6
Iterations	770
Elapsed time	2.65625 seconds

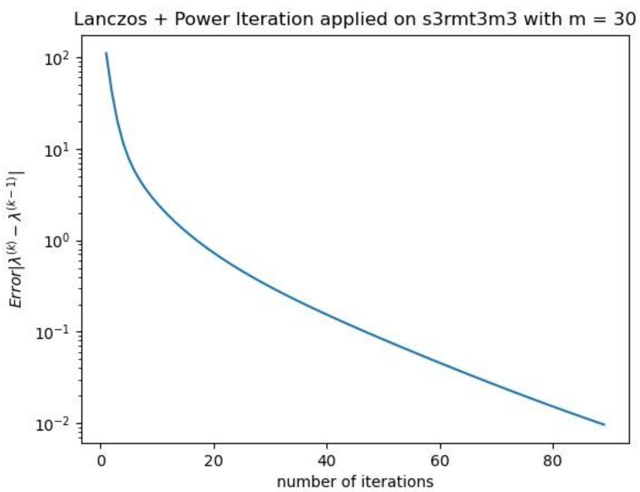
In this test, the convergence is depends on $\left|\lambda^{(k)}-\lambda^{(k-1)}\right|$, i.e., the difference between eigenvalue and the eigenvalue in the previous iteration. The convergence tolerance is set at 10^{-8}.

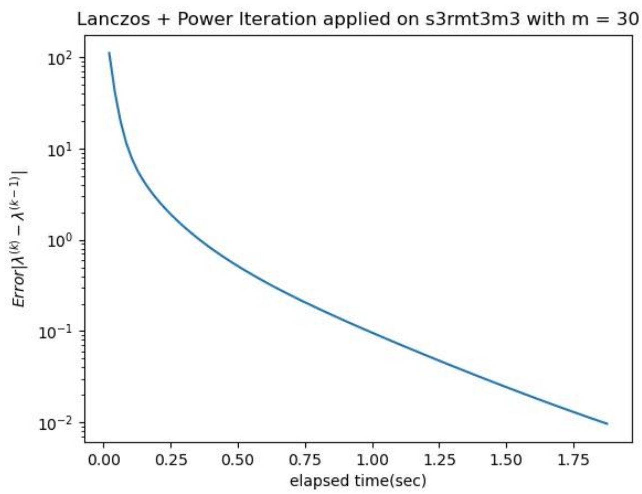
(2) Lanzcos Algorithm

Lanzcos algorithm is a specialized version of Arnoldi method to tackle s.p.d. matrix. Thus, it also has the dimension of Krylov space m.

In this task, we try 4 different m, which are 30, 50, 75, 100, respectively. Moreover, different m has different tolerance respectively, which are 10^{-2} (m = 30), 10^{-4} (m = 50), 10^{-6} (m = 75), 10^{-10} (m = 100).

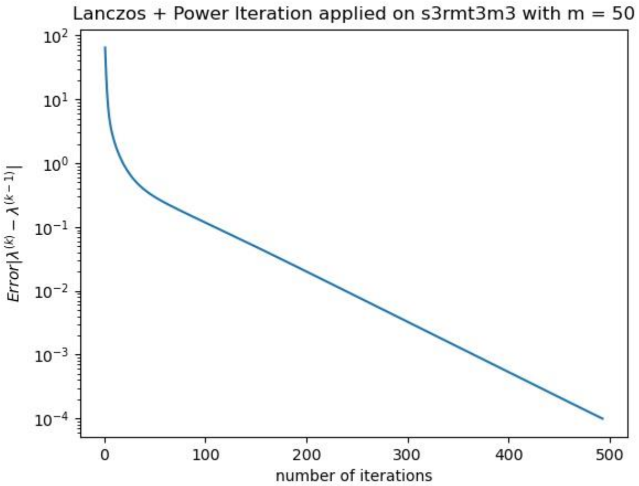
The largest eigenvalue of ***s3rmt3m3.mtx*** is given on MatrixMarket website, which is 9.598608e+06.

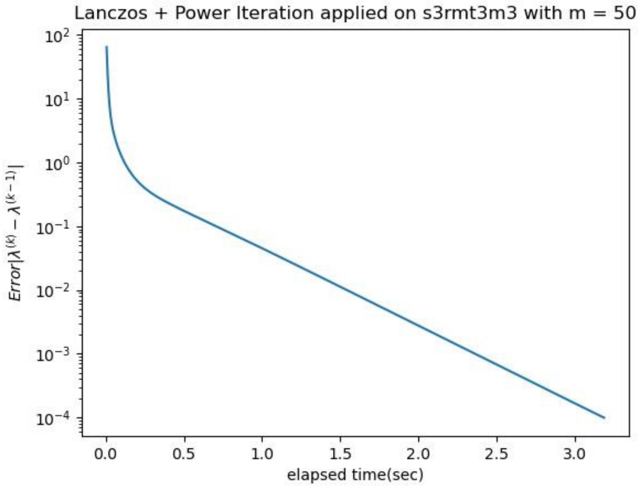




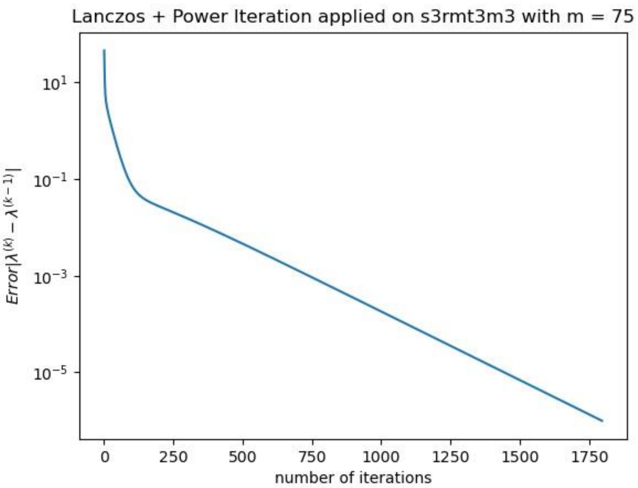
E+03.

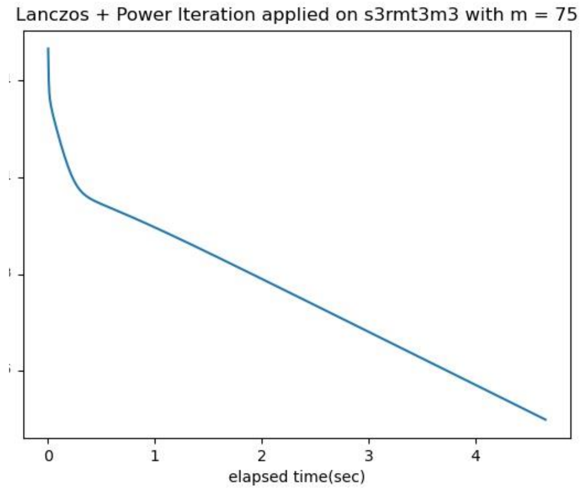
Lanzcos + power iteration on s3rmt3m3.mtx with m = 30	
Final Error	9.6722e-03
Largest Eigenvalue	9582.9197
Iterations	90
Elapsed time	1.875 seconds



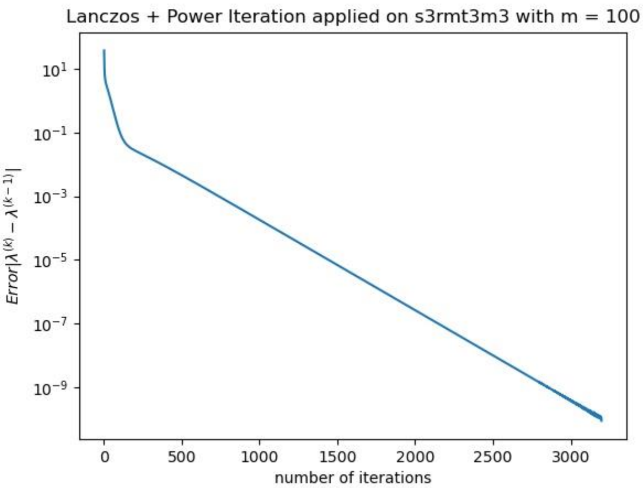


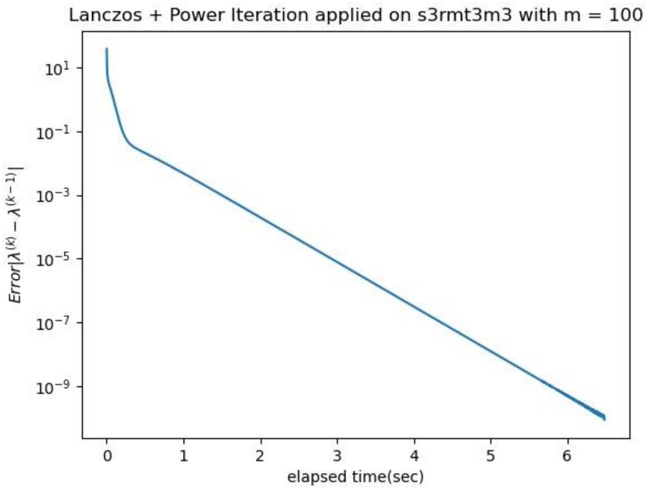
Lanczos + power iteration on s3rmt3m3.mtx with m = 50	
Final Error	9.9446e-05
Largest Eigenvalue	9598.5312
Iterations	494
Elapsed time	3.1875 seconds





Lanczos + power iteration on s3rmt3m3.mtx with m = 75	
Final Error	9.9707e-07
Largest Eigenvalue	9598.6079
Iterations	1797
Elapsed time	4.65625 seconds

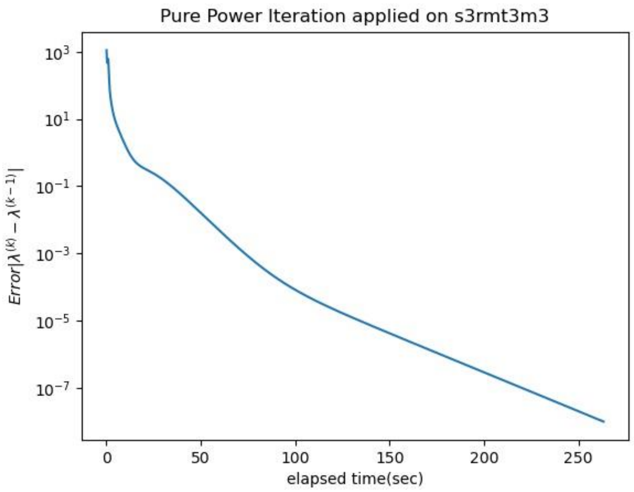


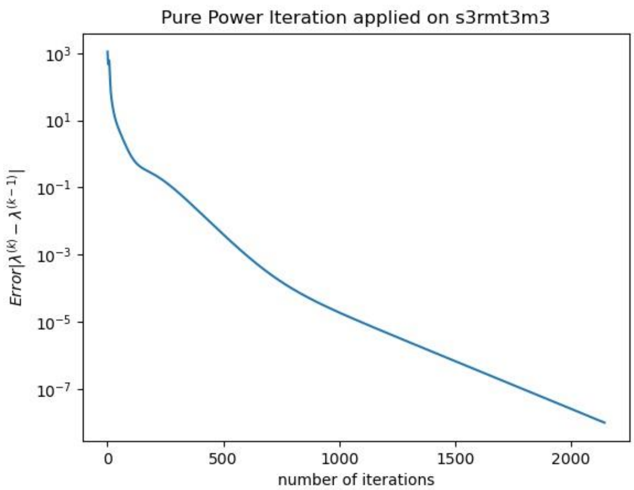


Lanczos + power iteration on s3rmt3m3.mtx with m = 100	
Final Error	8.7311e-11
Largest Eigenvalue	9598.6080
Iterations	3199
Elapsed time	6.484375 seconds

(3). Comparison

To get a better comparison between Lanczos algorithm and pure power iteration, we also apply pure power iteration on the same matrix, ***s3rmt3m3.mtx.***





Pure power iteration on Nos6.mtx	
Final Error	9.9844e-09
Largest Eigenvalue	9598.6080
Iterations	2145
Elapsed time	243.28125 seconds

We can notice that, power iteration can access the correct result as well. However, to reach this accuracy, it consumes much more time than Lanzcos algorithm, which takes less than 10 seconds. Lanzcos can achieve more than 20 times efficiency.