Eigensolvers

Fast Iterative Solvers, Project 3

Johannes Leonard Grafen, 380149

August 10, 2023

Contents

List of Figures

List of Tables

1 Remarks on used architecture and compiler options

The code was written in C++, compiled using the Clang compiler and executed on an Apple Silicon M1 Pro Chip featuring the ARM64 architecture. To measure timings, the high resolution clock of the std chrono library was employed. The following flags where passed to the compiler to enhance code performance of the aforementioned architecture: -march=native, -O3. To suppress the output to measure runtime accurately the flag "DISABLEIO" was introduced and passed to the compiler via the "-D" option.

2 Power Iteration

2.1 "power_test_msr" - Matrix

A power iteration method was implemented as discussed in the lecture. For the **power_test_msr** matrix a maximum eigenvalue of $\lambda_{max} = 7.65060331390989758e + 06$ was found after 770 iterations using the stopping criterion

$$|\lambda^{(k)} - \lambda^{(k-1)}| < 10^{-8} \tag{1}$$

an an initial guess of $\mathbf{x} = \frac{1}{\sqrt{n}}(1, 1, ..., 1)^T$ as described in the project's instructions.

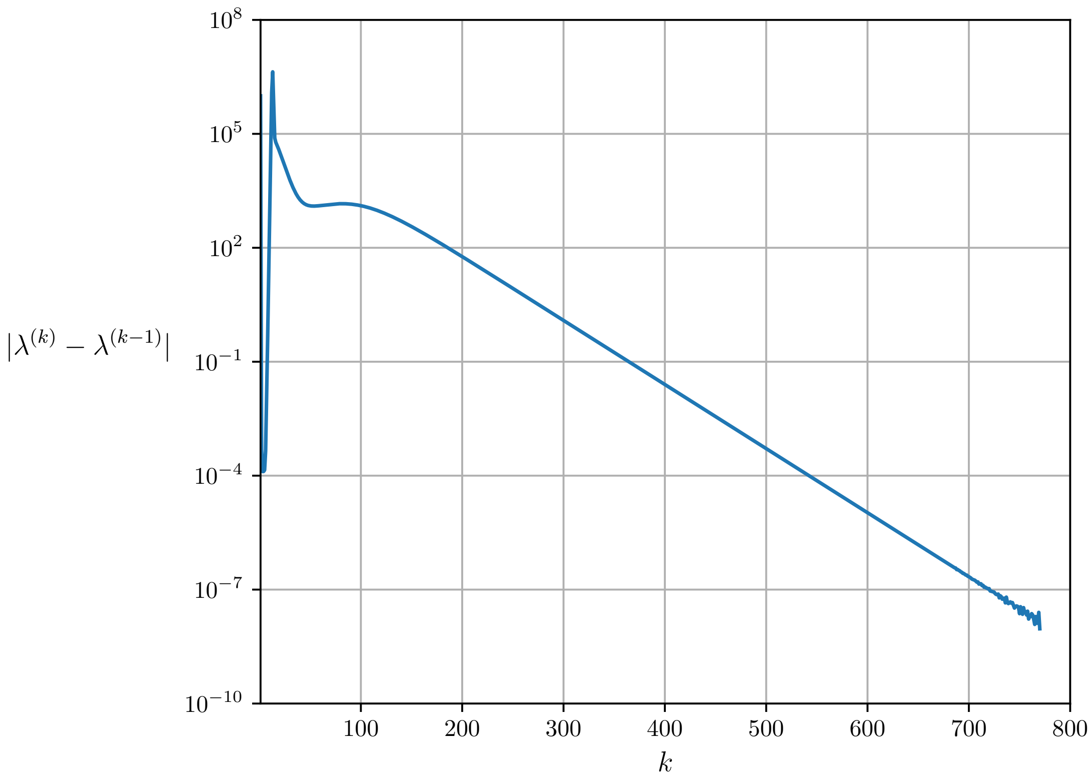
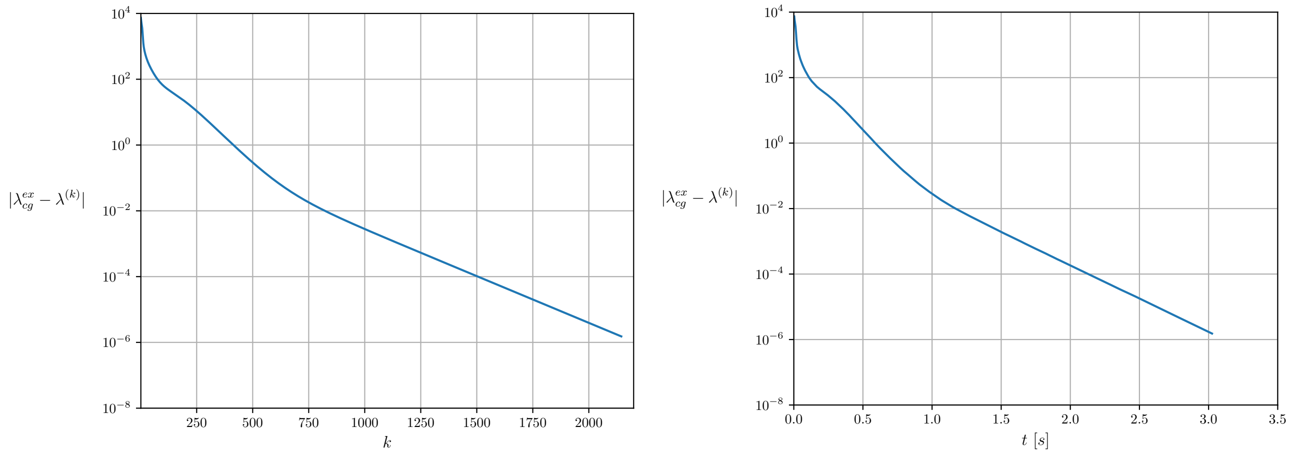


Figure 1: Convergence according to Eq. (1) for the test matrix **power_test_msr**

Fig. 1 shows the evolution of the convergence criterion as in Eq. (1). After an initial peak, the power iteration method converges linearly with an increasing iteration index. Small oscillations are observed in the region close to the stopping criterion (10^{-8}).

2.2 "cg_test_msr" - Matrix

Using the power iteration method to determine the maximum eigenvalue for the **cg_test_msr** matrix yields a maximum eigenvalue of $\lambda_{max} = 9.59860808796396850e + 03$. The pure power iteration method required 2.83192 seconds suppressing any I/O. Fig. 2b required I/O operations to track the runtime and the corresponding error. The error is defined as $\epsilon = |\lambda_{cg}^{ex} - \lambda^{(k)}|$, where $\lambda_{cg}^{ex} = 9.5986080894852857e + 3$ denotes the exact maximum eigenvalue of the **cg_test_msr** matrix, as given in the instructions. The error is plotted against the iteration index and runtime in Fig. 2. The minimum and final error is ϵ = 1.52132e-6 for the **cg_test_msr** matrix.



(a) Error against iteration index

(b) Error against runtime

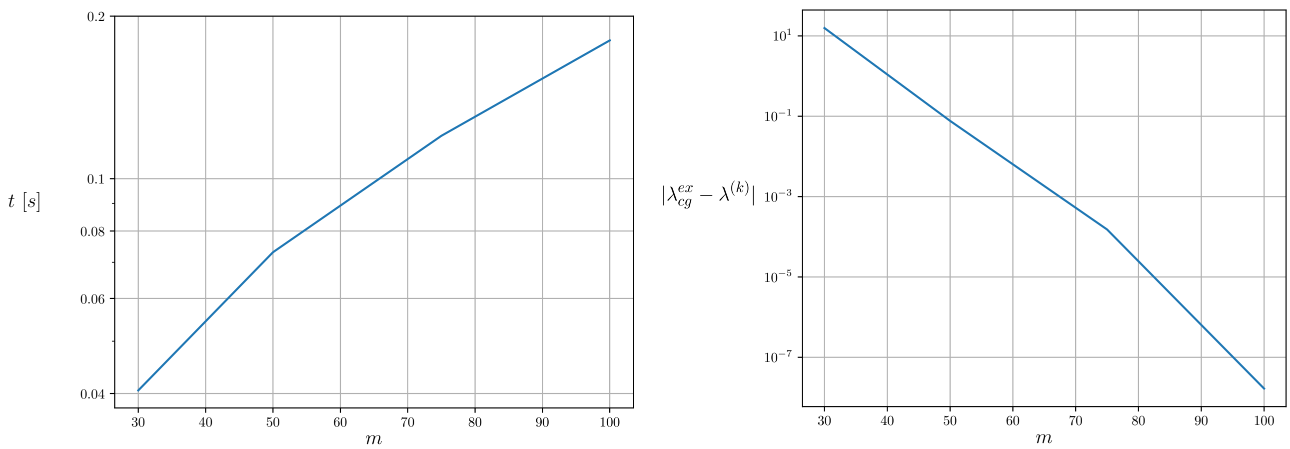
Figure 2: Error against iteration index and runtime on a semi-log scale for the **cg_test_msr** matrix

3 Lanczos Method

In addition to the pure power iteration, the Lanczos method was also used to determine the eigenvalues of the **cg_test_msr** matrix, using $m = 30, 50, 75, 100$, where m corresponds to the dimension of the Krylov space. A power iteration method is used to compute the maximum eigenvalue of the tridiagonal Lanczos matrix. In Tab. 1 the runtime for different number of Krylov vectors as well as the final error are presented. The error decreases with an increasing number of Krylov vectors (compare Fig. 3b). On the other hand, the runtime increases approximately linear with the number of Krylov vectors as Fig. 3a illustrates. Compared to the pure powerIteration the runtime for the Lanczos Method with $m = 100$ is significantly faster, requiring only 6.23 % of the runtime of the pure power iteration for the **cg_test_msr** matrix. Furthermore, the Lanczos Method with $m = 100$ results in an error that is two orders of magnitude lower than the error obtained by the pure power iteration.

m	runtime $[s]$	λ_{max}	$\epsilon = \lvert\lambda_{cg}^{ex} - \lambda^{(k)}\rvert$
30	0.041932	9.58291974574370579e+03	1.56883437415799563e+01
50	0.070548	9.59853122138019717e+03	7.68681050885788864e-02
75	0.119099	9.59860793744963121e+03	1.52035654537030496e-04
100	0.176473	9.59860808946861471e+03	1.66710378834977746e-08

Table 1: Runtime, maximum eigenvalue and final error for different number of Krylov vectors



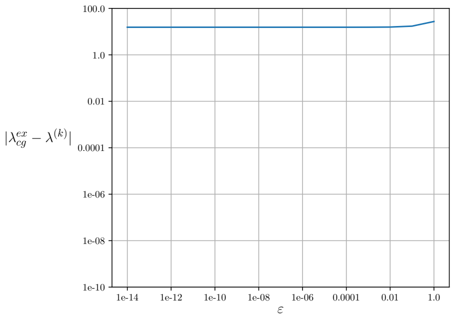
(a) Runtime against dimension of the Krylov space

(b) λ_{max} against dimension of the Krylov space

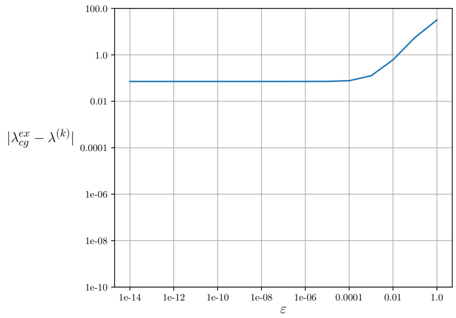
Figure 3: Runtime and maximum eigenvalue against dimension of the Krylov space

3.1 Optional Task: Investigation of different tolerances

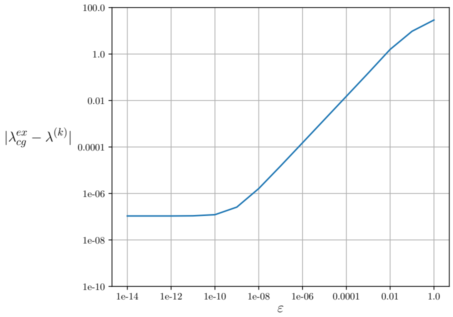
In order to determine optimal tolerances for different number of Krylov vectors m a parameter study with different tolerances $\varepsilon = |\theta^{(k)} - \theta^{(k-1)}|$ for the power iteration of the tridiagonal Lanczos matrix was performed. Fig. 4 shows the results of this study. For $m = 30$ and $m = 50$ the tolerances as given in the instructions proved to be an appropriate bound as no significant changes are recorded below these tolerances (compare Fig. 4a and Fig. 4b). For $m = 75$ and $m = 100$, the prescribed tolerances are 10^{-6} and 10^{-8} respectively. In contrast we can observe no significant changes only below an ε of 10^{-10} for $m = 75$ and 10^{-12} for $m = 100$ from the corresponding evolutions of the error. Adjusted tolerances ε^* are presented in Tab. 2.



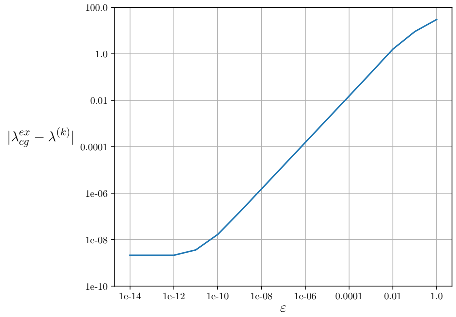
(a) $m = 30$



(b) $m = 50$



(c) $m = 75$



(d) $m = 100$

Figure 4: Comparison of error $\epsilon = |\lambda_{cg}^{ex} - \lambda^{(k)}|$ for different tolerances $\varepsilon = |\theta^{(k)} - \theta^{(k-1)}|$ for different number of Krylov vectors m

m	ε	ε^*
30	10^{-2}	10^{-2}
50	10^{-4}	10^{-4}
75	10^{-6}	10^{-10}
100	10^{-10}	10^{-12}

Table 2: prescribed tolerance ε and newly suggested tolerance ε^* for different number of Krylov vectors m