Computational Electrical Engineering - Solver Assignment for the Numerical Lab

Lorenzo Schiavone

November 27, 2025

1 Section 8.1

We take the system matrix A and the right-hand side b from the Example 6.1 of the Numerical Lab Notes, together with the exact solution $x0$ obtained with the backslash operator in Matlab, i.e., from a direct LU factorization of A and a backward and forward substitution. It is a symmetric matrix of size 324.

In this exercise we compare the convergence plot of different iterative schemes to solve the system. In particular we test the Preconditioned Conjugate Gradient (PCG) and the Generalized Minimum Residual Method (GMRES) without preconditioner, with the incomplete Cholesky preconditioner and with the incomplete LU preconditioner.

Convergence plot: Figure 1 shows the convergence plot for the different iterative schemes. We may notice that, in this case, the iLU preconditioner and the Cholesky have the same effect in reducing the number of iterations to around 15 both for the PCG and the GMRES. Without preconditioner both GMRES and PCG converge in around the same number of iterations, i.e., 35. The parameter `maxit` is kept high to allow the reaching of the desired convergence, while the `tol` is set to `1e-6` as indicated in the assignment.

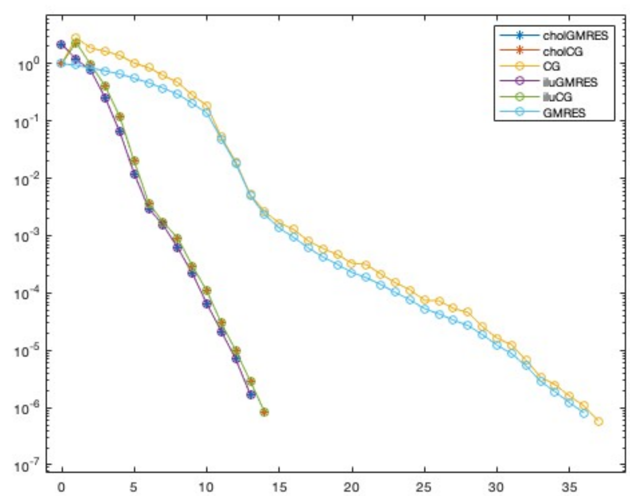

Figure 1: Convergence Plot for the different iterative schemes.

Relative Error: The relative error between the iterative solution and the exact solution is computed as

$$\texttt{err}(x^{\text{it}}) = 100 * \frac{\|x_{\text{exact}} - x^{\text{it}}\|_2}{\|x_{\text{exact}}\|_2}. \quad (1)$$

Even though in real application we have not the exact solution, in Table 1 we may notice that the tolerance on the residue allows to obtain solutions close to the real one, if the system is not too ill conditioned.

method	PCG			GMRES		
preconditioner	none	ichol	ilu	none	ichol	ilu
error	3.42e-06	1.77e-05	1.77e-05	1.05e-05	2.93e-05	2.93e-05

Table 1: Relative error computed as in Equation 1 for the different iterative methods and different preconditioners.

Steepest Descent: We implement the Steepest Descent Algorithm as in Algorithm 26 in the Numerical Lab Notes and we test it on the linear system in exam. Figure 2 shows that it reaches convergence but in a significantly larger number of iterations (592). This is an effect of the spectral radius of the smoother for the Algorithm that is particularly close to one. Indeed, even though the condition number of A is only $k(A) = 198$, the ratio is

$$\frac{k(A)-1}{k(A)+1} = 0.99.$$

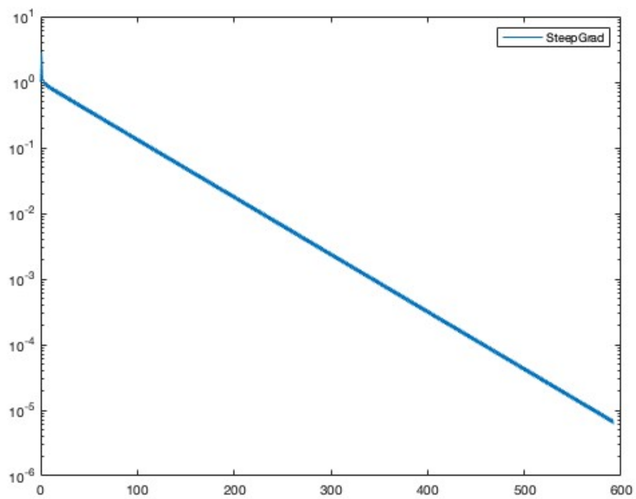

Figure 2: Convergence Plot for the Steepest Descent Algorithm.