Fachbereich Mathematik & Informatik
Freie Universität Berlin
Prof. Dr. Ralf Kornhuber, Lasse Hinrichsen-Bischoff

Exercise 12 for the lecture

Numerical Mathematics II

WS 2021/2022

http://numerik.mi.fu-berlin.de/wiki/WS_2021/NumericsII.php

Due: 11:59pm on Monday, January 31, 2022

Problem 1

Prove Proposition 4.4.3:
Let e_i be the i-th Cartesian unit vector, i.e. $(e_i)_j = \delta_{ij}$.

a) Show that for this particular choice, the Parallel direction correction (PDC) method is equivalent to the Jacobi method.

b) Similarly, show that the SDC method is equivalent to the Gauss–Seidel method.

Problem 2

The symmetric Gauß-Seidel method for the solution of a linear system with a s.p.d. matrix is obtained by applying one normal Gauß-Seidel step and one Gauß-Seidel step with the components in reversed order alternatingly. Give the iteration matrix and the generated preconditioner of the symmetric Gauß-Seidel method and show that the preconditioner is s.p.d..

Problem 3

Consider the the linear system

$$AU = b \tag{1}$$

with the symmetric positive definite matrix $A \in \mathbb{R}^{n,n}$ and $b \in \mathbb{R}^n$.

a) Implement the Jacobi and the Gauß-Seidel methods

`Jacobi(A, b, u0, tol, uexact)`

and

`GaussSeidel(A, b, u0, tol, uexact).`

The iteration should stop if the energy norm $\|\cdot\|_A = \langle A\cdot, \cdot\rangle^{0.5}$ of the error is smaller than the tolerance `tol`. The methods should return a pair `(u, n)` where `u` is the approximated solution and `n` is the number of iterationsteps performed.

b) Define $h := \frac{1}{N-1}$. Test your program with the matrix $A \in \mathbb{R}^N$

$$A = \frac{1}{h^2}\begin{pmatrix} 1 & 0 & 0 & & \dots & & 0 \\ 1/h^2 & -2/h^2 & 1/h^2 & 0 & \dots & & 0 \\ 0 & 1/h^2 & -2/h^2 & 1/h^2 & 0 & \dots & 0 \\ & & \vdots & & & & \\ 0 & 0 & \dots & 0 & 1/h^2 & -2/h^2 & 1/h^2 \\ 0 & \dots & & & & 0 & 1 \end{pmatrix},$$

i. e. having $2/h^2$ on the diagonal and $-1/h^2$s on the off-diagonals execpt for the first and last row.

Hint: You can use `spdiags` in Matlab or `scipy.sparse.spdiags` in Python to create the sparse matrix.

Use the right hand side

$$b = \frac{1}{h}\begin{pmatrix} 0 \\ 1 \\ \vdots \\ 1 \\ 0 \end{pmatrix}.$$

c) Solve the system with varying values of N. Plot the number of required iteration steps over the number of unknowns N. Comment on what you found out.

d) Explain how a Gauss–Seidel step can be performed in $\mathcal{O}(N)$ if the matrix is sparse.