Otto-von-Guericke-Universität Magdeburg
Max Planck Institute for Dynamics of Complex Technical Systems
Prof. Dr. Peter Benner, Dr. Ion Victor Gosea, Jennifer Przybilla

Chemnitz University of Technology
Faculty of Mathematics
Dr. Sara Grundel, Dr. Jan Blechta

winter semester 2021/2022

Please send your solutions (including a MATLAB®implementation if applicable) by **Oct 31, 11:59 PM (Magdeburg students), Oct 25, 11:59 PM (Chemnitz students)** to przybilla@mpi-magdeburg.mpg.de (Magdeburg students) or jan.blechta@math.tu-chemnitz.de (Chemnitz students) with subject `NLA-HW02`. Late submissions are only possible if requested by email before the due date for a valid reason.

Numerical Linear Algebra – homework #02

Problem 1 (QR decomposition)

With the help of the Gram–Schmidt method we can determine for a matrix $A \in \mathbb{R}^{m,n}$ a (sparse) QR decomposition $A = QR$, $Q \in \mathbb{R}^{m,n}$, $R \in \mathbb{R}^{n,n}$ using the following algorithm:

Algorithm 1 Gram-Schmidt

1: **for** $j = 1, 2, \ldots$ **do**
2: $\quad \widetilde{q}_j = a_j$
3: $\quad$ **for** $k = 1, 2, \ldots, j-1$ **do**
4: $\quad\quad r_{kj} = q_k^{\mathrm{T}} a_j$
5: $\quad\quad \widetilde{q}_j = a_j - r_{k,j} q_k$
6: $\quad$ **end for**
7: $\quad r_{jj} = \|\widetilde{q}_j\|_2$
8: $\quad q_j = \widetilde{q}_j / r_{jj}$
9: **end for**

a) Compute the QR decomposition of the matrix $A = \begin{bmatrix} 1 & 1 \\ 1 & 2 \\ 1 & 2 \\ 1 & 5 \end{bmatrix}$ using the Gram–Schmidt method.

b) Implement the computation of a QR decomposition in Matlab or Octave using the Gram–Schmidt method.

Problem 2 (Gaussian elimination)

We consider Gaussian elimination (GE), LU decomposition and pivoting for solving the linear system $Ax = b$.

a) Explain how we can modify GE from the lecture notes such that the transformations L_i can be applied directly to the right-hand side b.

b) Implement GE without pivoting in MATLAB.

c) Let the following matrix be given:

$$A = \begin{bmatrix} 3 & 17 & 10 \\ 2 & 4 & -2 \\ 6 & 18 & -12 \end{bmatrix}.$$

Compute the factorization $PA = LU$ and compare the results with the MATLAB's `lu` routine.

Solve $Ax = b$ with `b=sum(A,2)` with your implementation and compare the results with the solution that was computed using MATLAB's `lu`. For that you can use the norm of the residual `norm(b-Ax)` or the error for both cases.

Repeat the experiment with $a_{11} = 10^{-6}$. Explain your observation.