MAP55672 (2024-25) — Case studies 2

Krylov subspace methods

(Lecturer: P. Fritzsch, mailto:fritzscp@tcd.ie)

Instructions

- Complete all parts of the assignment by **midnight on Wednesday 26th March 2025**. Create a git repository with your solutions and submit the link to me by e-mail.
- The git repository should include all source files and a summary pdf containing a description of solutions (incl. tables and plots, if necessary). The summary report should reflect your understanding of the material and contain a short description of your submitted code. As such it should contain all necessary information to (compile and) run your code for validation purposes.
- *It is strictly forbidden to use code generating tools such as LLM's. Their use will result in 0 marks for this case study.*

2 The GMRES algorithm

The goal of this case study is the implementation of a variant of the GMRES algorithm for the solution of a square linear system $Ax = b$ for arbitrary regular matrices A and right hand sides $b \neq 0$.

2.1 Basics: Arnoldi iteration.

Use a language of your choice (Matlab, C, C++) to implement the Arnoldi iteration algorithm for Krylov subspaces, `Q,H = arnoldi(A,u,m)`, that performs the Arnoldi iteration for a regular matrix A with vector u to build the Krylov subspace of degree m.
It returns the $m+1$ orthonormal basis and the upper Hessenberg matrix H of size $(m+1)\times m$ for the following linear system.

$$\begin{pmatrix} 3 & 8 & 7 & 3 & 3 & 7 & 2 & 3 & 4 & 8 \\ 5 & 4 & 1 & 6 & 9 & 8 & 3 & 7 & 1 & 9 \\ 3 & 6 & 9 & 4 & 8 & 6 & 5 & 6 & 6 & 6 \\ 5 & 3 & 4 & 7 & 4 & 9 & 2 & 3 & 5 & 1 \\ 4 & 4 & 2 & 1 & 7 & 4 & 2 & 2 & 4 & 5 \\ 4 & 2 & 8 & 6 & 6 & 5 & 2 & 1 & 1 & 2 \\ 2 & 8 & 9 & 5 & 2 & 9 & 4 & 7 & 3 & 3 \\ 9 & 3 & 2 & 2 & 7 & 3 & 4 & 8 & 7 & 7 \\ 9 & 1 & 9 & 3 & 3 & 1 & 2 & 7 & 7 & 1 \\ 9 & 3 & 2 & 2 & 6 & 4 & 4 & 7 & 3 & 5 \end{pmatrix} x = \begin{pmatrix} +0.757516242460009 \\ +2.734057963614329 \\ -0.555605907443403 \\ +1.144284746786790 \\ +0.645280108318073 \\ -0.085488474462339 \\ -0.623679022063185 \\ -0.465240896342741 \\ +2.382909057772335 \\ -0.120465395885881 \end{pmatrix}$$

Test your algorithm and determine Q_9.

2.2 Serial implementation of GMRES.

Implement a serial variant of the GMRES algorithm, `gmres(A,b,m)`, that does m iterations and returns the final solution estimate x and a vector with the history of residual norms. Explain your implementation and apply it to the n-dimensional linear system with

$$A = \begin{pmatrix} -4 & 1 & 0 & \cdots & 0 & 0 & 0 \\ 1 & -4 & 1 & \ddots & \vdots & \vdots & \vdots \\ 0 & 1 & -4 & \ddots & 0 & 0 & 0 \\ 0 & 0 & 1 & \ddots & 1 & 0 & 0 \\ 0 & 0 & 0 & \ddots & -4 & 1 & 0 \\ \vdots & \vdots & \vdots & \ddots & 1 & -4 & 1 \\ 0 & 0 & 0 & \cdots & 0 & 1 & -4 \end{pmatrix}, \qquad b = \begin{pmatrix} 1/n \\ 2/n \\ 3/n \\ \vdots \\ (n-1)/n \\ 1 \end{pmatrix}$$

For $n = 8, 16, 32, 64, 128, 256$, run your function for $m = n/2$ iterations. On a semi-log graph, plot $||r_k||_2/||b||_2$ for all the cases. How does the convergence rate of GMRES seem to depend on n?

2.3 Parallel implementation of GMRES.

Try to implement a parallel variant of the GMRES algorithm of your choice and explain key ingredients of your version as well as possible tests for correctness. What is a good stopping criteria for your algorithm?