AME40541/60541: Finite Element Methods
Homework 5: Due Wednesday, April 20, 2020

Problem 1: (50 points) Consider a general nonlinear vector-valued function of m variables and its Jacobian matrix

$$\begin{aligned} \boldsymbol{R} : \mathbb{R}^m &\to \mathbb{R}^m \\ \boldsymbol{x} &\mapsto \boldsymbol{R}(\boldsymbol{x}), \end{aligned} \qquad \begin{aligned} \boldsymbol{J} : \mathbb{R}^m &\to \mathbb{R}^{m\times m} \\ \boldsymbol{x} &\mapsto \frac{\partial \boldsymbol{R}}{\partial \boldsymbol{x}}(\boldsymbol{x}), \end{aligned}$$

where $\left[\frac{\partial \boldsymbol{R}}{\partial \boldsymbol{x}}\right]_{ij} = \frac{\partial R_i}{\partial x_j}$. To find a root of $\boldsymbol{R}$, i.e., find $\boldsymbol{x}^* \in \mathbb{R}^m$ such that $\boldsymbol{R}(\boldsymbol{x}^*) = 0$, we will apply the Newton-Raphson iteration

$$\boldsymbol{x}_{k+1} = \boldsymbol{x}_k - \left[\frac{\partial \boldsymbol{R}}{\partial \boldsymbol{x}}(\boldsymbol{x}_k)\right]^{-1} \boldsymbol{R}(\boldsymbol{x}_k),$$

where $\boldsymbol{x}_0 \in \mathbb{R}^m$ is some initial guess for $\boldsymbol{x}^*$ and $\boldsymbol{x}_k \in \mathbb{R}^m$ is the approximation of $\boldsymbol{x}^*$ at the kth iteration for $k = 0, 1, \ldots$. The Newton-Raphson iterations will converge quadratically to a root of $\boldsymbol{R}$ under some assumptions. For the algorithm to be practical, we must specify a maximum number of iterations N (in cases where the iterations diverge) and a convergence condition. We will consider iteration k to be sufficiently close to $\boldsymbol{x}^*$ (converged) if

$$\|\boldsymbol{R}(\boldsymbol{x}_k)\|_\infty \leqslant \tau,$$

where $\|\cdot\|_\infty$ is the infinity norm (the entry with the maximum magnitude) and $\tau \in \mathbb{R}$ is the convergence tolerance. If none of the iterations $\{\boldsymbol{x}_0, \ldots, \boldsymbol{x}_N\}$ satisfy the convergence conditions, we assume the algorithm is diverging and terminate the iteration.

(a) Implement the Newton-Raphson method in `solve_newtraph.m`. Starter code is provided in Homework 5 code distribution. Be sure to save convergence iteration (see comments in starter code) in order to investigate the convergence properties of the method.

(b) Test your function on any linear system you choose $\boldsymbol{R}(\boldsymbol{x}) = \boldsymbol{A}\boldsymbol{x} - \boldsymbol{b}$ and verify it converges to machine precision after one iteration.

(c) Consider the following nonlinear function

$$\boldsymbol{R}(\boldsymbol{x}) = \begin{bmatrix} e^{-e^{-(x_1+x_2)}} - x_2(1+x_1^2) \\ x_1\cos(x_2) + x_2\sin(x_1) - \frac{1}{2}. \end{bmatrix}$$

- Derive the Jacobian of $\boldsymbol{R}(\boldsymbol{x})$. As always, you are welcome to use symbolic mathematics software.
- Use your code to find a root $\boldsymbol{R}(\boldsymbol{x})$ from the starting point $\boldsymbol{x} = (0, 0)^T$. Use $\tau = 10^{-10}$ and $N = 20$.
- Verify quadratic convergence by computing the ratio

$$m_k = \frac{\log\|\boldsymbol{R}(\boldsymbol{x}_k)\|_\infty}{\log\|\boldsymbol{R}(\boldsymbol{x}_{k-1})\|_\infty}$$

  for $k = 1, \ldots, N$ and confirming $m_k \to 2$.

(d) Repeat the previous tasks for the following nonlinear function

$$\boldsymbol{R}(\boldsymbol{x}) = \begin{bmatrix} x_1 + 2x_2 + 4x_3 \\ \sin x_1 \sin x_2 \sin x_3 \\ x_1^2 + 2x_2^2 + 3x_3^2 \end{bmatrix}$$

with the starting point $\boldsymbol{x} = (0.5, 0.5, 0.5)^T$ with $\tau = 10^{-10}$ and $N = 100$. Does the method still converge quadratically? If not, explain why quadratic convergence breaks down in this case.

(e) Check your code to make sure your implementation is reasonably *efficient*. In particular, make sure there are no unnecessary evaluations of $\boldsymbol{R}(\boldsymbol{x})$ and its Jacobian. Even though we have only considered small problems in this assignment, you will use this code in your project to solve 2- and 3-dimensional PDEs using the finite element method.