Course of Numerical Methods for Engineering
Lab 9

Luca Bonaventura, Tommaso Benacchio

MOX - Politecnico di Milano

PHYS-ENG, A.Y. 2020-21

10/11/2020

Topic of this session:

- **Methods for nonlinear systems**

Nonlinear systems

These systems can be written as

$$\begin{aligned} & f_1(x_1, x_2, \cdots, x_n) = 0 \\ & f_2(x_1, x_2, \cdots, x_n) = 0 \\ & \dots \\ & f_n(x_1, x_2, \cdots, x_n) = 0. \end{aligned}$$

In vector notation, setting

$$\mathbf{x} = \left[\begin{array}{c} x_1 \\ x_2 \\ \dots \\ x_n \end{array} \right] \qquad \mathbf{f}(\mathbf{x}) = \left[\begin{array}{c} f_1(x_1, \dots, x_n) \\ f_2(x_1, \dots, x_n) \\ \dots \\ f_n(x_1, \dots, x_n) \end{array} \right]$$

these problems can be rewritten in more compact form as

$$\mathbf{f}(\mathbf{x}) = \mathbf{0}.$$

Fixed point method for systems

Definition

Let Ω denote a convex domain in $\mathbf{R}^n$. Let $\phi \in \mathcal{C}^1(\Omega)$ and such that $\phi : \Omega \to \Omega$ and let $\mathbf{x}^{(0)} \in \Omega$. The fixed point method for the solution of $\mathbf{x} = \phi(\mathbf{x})$ is defined by the recursive sequence

$$\mathbf{x}^{(k+1)} = \phi(\mathbf{x}^{(k)}) \quad k = 1, \ldots, m.$$

The function ϕ is also called iteration function.

Theorem

If the function ϕ is such that

$$\sup_{\mathbf{x} \in \Omega} \left\| \frac{\partial \phi}{\partial \mathbf{x}} \right\| \leq M < 1,$$

there is a unique $\mathbf{x}^* \in \Omega$ such that $\mathbf{x}^* = \phi(\mathbf{x}^*)$ and the sequence defined in (1) converges so that

$$\|\mathbf{x}^{(k)} - \mathbf{x}^*\| \leq M^k \|\mathbf{x}^{(0)} - \mathbf{x}^*\|.$$

Newton's method for systems

Definition

Let $\mathbf{f} \in \mathcal{C}^1(\Omega)$ and let $\mathbf{x}^{(0)} \in \Omega$. Newton's method for the solution of $\mathbf{f}(\mathbf{x}) = \mathbf{0}$ is defined by the recursive sequence $\mathbf{x}^{(k)}, \quad k = 1, \dots, m$ of vectors obtained as solutions of the linear systems

$$\frac{\partial \mathbf{f}}{\partial \mathbf{x}}(\mathbf{x}^{(k)})\mathbf{x}^{(k+1)} = \frac{\partial \mathbf{f}}{\partial \mathbf{x}}(\mathbf{x}^{(k)})\mathbf{x}^{(k)} - \mathbf{f}(\mathbf{x}^{(k)}) \quad k = 0, \dots, m.$$

Equivalently, the recursive sequence can be defined as

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} - \left[\frac{\partial \mathbf{f}}{\partial \mathbf{x}}(\mathbf{x}^{(k)})\right]^{-1} \mathbf{f}(\mathbf{x}^{(k)}) \quad k = 0, \dots, m.$$

Modified Newton's method for systems

Definition

Let $\Omega \subset \mathbf{R}^n$ and $\mathbf{f} : \Omega \to \mathbf{R}^n$ such that $\mathbf{f} \in \mathcal{C}^2(\Omega)$. Given $\mathbf{x}^{(0)} \in \Omega$ the **modified Newton method with constant Jacobian matrix** for the solution of system $\mathbf{f}(\mathbf{x}) = \mathbf{0}$ is defined for $k \geq 0$ by

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + \boldsymbol{\delta}\mathbf{x}^{(k)} \qquad \mathbf{J}^{(0)}\boldsymbol{\delta}\mathbf{x}^{(k)} = -\mathbf{f}(\mathbf{x}^{(k)}),$$

where $\mathbf{J}^{(0)} = \frac{\partial \mathbf{f}}{\partial \mathbf{x}}(\mathbf{x}^{(0)})$.

Definition

Let $\Omega \subset \mathbf{R}^n$ and $\mathbf{f} : \Omega \to \mathbf{R}^n$ such that $\mathbf{f} \in \mathcal{C}^2(\Omega)$. Given $\mathbf{x}^{(0)} \in \Omega$ the **modified Newton method with constant approximate Jacobian** for the solution of system $\mathbf{f}(\mathbf{x}) = \mathbf{0}$ is defined for $k \geq 0$ by

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + \boldsymbol{\delta}\mathbf{x}^{(k)} \qquad \tilde{\mathbf{J}}^{(0)}\boldsymbol{\delta}\mathbf{x}^{(k)} = -\mathbf{f}(\mathbf{x}^{(k)}),$$

where $\tilde{\mathbf{J}}^{(0)}$ denotes an appropriate approximation of the Jacobian matrix $\mathbf{J}^{(0)} = \frac{\partial \mathbf{f}}{\partial \mathbf{x}}(\mathbf{x}^{(0)})$.

The Broyden quasi-Newton method

Definition

Given $\mathbf{x}^{(0)}$ and $\mathbf{J}_0 = \frac{\partial \mathbf{f}}{\partial \mathbf{x}}(\mathbf{x}^{(0)})$, the Broyden quasi-Newton method is defined for $k \geq 0$ by

$$
\begin{aligned}
\mathbf{J}_k \boldsymbol{\delta}\mathbf{x}^{(k)} &= -\mathbf{f}(\mathbf{x}^{(k)}) \\
\mathbf{x}^{(k+1)} &= \mathbf{x}^{(k)} + \boldsymbol{\delta}\mathbf{x}^{(k)} \\
\boldsymbol{\delta}\mathbf{f}^{(k)} &= \mathbf{f}(\mathbf{x}^{(k+1)}) - \mathbf{f}(\mathbf{x}^{(k)}) \\
\mathbf{J}_{k+1} &= \mathbf{J}_k \\
&\quad + \frac{(\boldsymbol{\delta}\mathbf{f}^{(k)} - \mathbf{J}_k \boldsymbol{\delta}\mathbf{x}^{(k)})(\boldsymbol{\delta}\mathbf{x}^{(k)})^T}{(\boldsymbol{\delta}\mathbf{x}^{(k)})^T \boldsymbol{\delta}\mathbf{x}^{(k)}}
\end{aligned}
$$

Definition

Given $\mathbf{x}^{(0)}$ and $\mathbf{J}_0 = \frac{\partial \mathbf{f}}{\partial \mathbf{x}}(\mathbf{x}^{(0)})$ and $\mathbf{B}_0 = \mathbf{J}_0^{-1}$, the Broyden quasi-Newton method with inverse update is defined for $k \geq 0$ by

$$
\begin{aligned}
\boldsymbol{\delta}\mathbf{x}^{(k)} &= -\mathbf{B}_k \mathbf{f}(\mathbf{x}^{(k)}) \\
\mathbf{x}^{(k+1)} &= \mathbf{x}^{(k)} + \boldsymbol{\delta}\mathbf{x}^{(k)} \\
\mathbf{u}_k &= \frac{\mathbf{f}(\mathbf{x}^{(k+1)})}{\|\boldsymbol{\delta}\mathbf{x}^{(k)}\|_2} \qquad \mathbf{v}_k = \frac{\boldsymbol{\delta}\mathbf{x}^{(k)}}{\|\boldsymbol{\delta}\mathbf{x}^{(k)}\|_2} \\
\mathbf{B}_{k+1} &= \mathbf{B}_k - \frac{\mathbf{B}_k \mathbf{u}_k \mathbf{v}_k^T \mathbf{B}_k}{1 + \mathbf{v}_k^T \mathbf{B}_k \mathbf{u}_k}.
\end{aligned}
$$

Exercise 1

Write the nonlinear system whose solution is the intersection of the three-dimensional sphere of radius 3, of the quartic paraboloid $z = x^4 + y^4$ and of the plane $x = y$. Using the initial guess $\mathbf{x}_0 = [1, 1, 2]^T$ and imposing an absolute error tolerance of 10^{-12}, solve the system using:

(a) **the Newton method with exact Jacobian;**

(b) **the Newton method with approximate Jacobian (use forward finite differences with increment $\delta = 0.01$);**

(c) **the Newton method with constant approximate Jacobian ;**

(d) **the quasi Newton Broyden method ;**

(e) **the quasi Newton Broyden method with inverse update.**

In all cases, compare the numerical solution obtained with the different variants of Newton's method with the reference solution given by the Octave/MATLAB solver `fsolve` and compare the number of iterations and execution time of each method.

Exercise 2

Repeat the previous exercise using as initial guesses $\mathbf{x}_0 = [2, 2, 2]^T$ and $\mathbf{x}_0 = [10, 10, 2]^T$.

Exercise 3

Consider the nonlinear system

$$x - \frac{y}{4} = 0$$
$$x^2 + y^2 + z^2 - \frac{1}{2} = 0$$
$$z - x^2 - y^2 = 0$$

Solve it with the initial guess $\mathbf{x}_0 = [0.1, 0.3, 0.3]^T$ using all the methods used in the previous exercises. Reformulate the problem as fixed point problem with appropriate iteration function and compare the solutions, the number of iterations and execution time of the Newton methods with those of the fixed point method. Verify that the convergence conditions for the fixed problem method are satisfied at the numerically computed solution.

Exercise 4

Let $\mathbf{A}$ be the $n \times n$ tridiagonal matrix with elements equal to 5 on the main diagonal and -1 on the first upper and lower diagonals and consider the nonlinear system:

$$\mathbf{Ax} + \mathbf{g}(\mathbf{x}) = \mathbf{0},$$

where $\mathbf{g}(\mathbf{x}) = [-2\sin(x_1) + \pi, \ldots, -2\sin(x_n) + \pi]$. In the case $n = 10$, solve the system using all the previously considered methods. For the fixed point method, set $\mathbf{\Phi}(\mathbf{x}) = -\mathbf{A}^{-1}\mathbf{g}(\mathbf{x})$ and compute each iteration by solving the linear system $\mathbf{Ax}^{(k+1)} = \mathbf{g}(\mathbf{x}^{(k)})$. Use the initial guess $\mathbf{x}_0 = [1, \ldots, 1]^T$ and an absolute error tolerance of 10^{-8}. In all cases, compare the numerical solution obtained with the different variants of Newton's method with the reference solution given by the Octave/MATLAB solver `fsolve` and compare the number of iterations and execution time of each method.

Exercise 5

Repeat the previous exercise using elements equal to 50 on the main diagonal. Repeat again using $n = 200$ and elements equal to 5 and then 50 on the main diagonal.

Exercise 6

Let $\mathbf{A}$ be the $n \times n$ banded matrix with elements equal to 10 on the main diagonal, -2 on the first upper and lower diagonals and -1 on the 10-th upper and lower diagonals and consider the nonlinear system:

$$\mathbf{Ax} + \mathbf{g}(\mathbf{x}) = \mathbf{0},$$

where $\mathbf{g}(\mathbf{x}) = -\mathbf{x}/\|\mathbf{x}\|_2^{\alpha}$. In the case $n = 200, \alpha = 1.5$, solve the system using all the previously considered methods that do not require the knowledge of the exact Jacobian. For the fixed point method, set $\mathbf{\Phi}(\mathbf{x}) = -\mathbf{A}^{-1}\mathbf{g}(\mathbf{x})$ and compute each iteration by solving the linear system $\mathbf{Ax}^{(k+1)} = \mathbf{g}(\mathbf{x}^{(k)})$. Use the initial guess $\mathbf{x}_0 = [1, \ldots, 1]^T$ and an absolute error tolerance of 10^{-10}. In all cases, compare the numerical solution obtained with the different variants of Newton's method with the reference solution given by the Octave/MATLAB solver `fsolve` and compare the number of iterations and execution time of each method. Repeat the exercise in the case $n = 20, \alpha = 2.5$ and discuss the difference in the results.