Solution of Non-Linear Set of Equations – Lecture 01

Dr. Serhat Hosder

Associate Professor of Aerospace Engineering

Mechanical and Aerospace Engineering

290B Toomey Hall

Missouri S&T

Rolla, MO 65409

Phone: 573-341-7239

E-mail: hosders@mst.edu

Outline

- We will discuss solving systems of non-linear equations. We will look at an extension of Newton's method to a system of unknowns.
- This gives rise to a linear problem on each iteration, which can be solved using both direct or indirect methods.
- This topic has broad application in the numerical solution of partial differential equations.

Non-Linear Equations

A system of non-linear equations has the form

$$
\begin{aligned}
f_1(x_1, x_2, x_3, \ldots, x_n) &= 0 \\
f_2(x_1, x_2, x_3, \ldots, x_n) &= 0 \\
f_3(x_1, x_2, x_3, \ldots, x_n) &= 0 \\
\vdots \quad \vdots \quad \vdots \quad \vdots \quad &= 0 \\
f_n(x_1, x_2, x_3, \ldots, x_n) &= 0
\end{aligned}
$$

Each function $\boldsymbol{f_i}$ is dependent on the vector $\boldsymbol{x}=(x_1, x_2, x_3, \ldots x_n)$.

In matrix form, we can write this as

$$\boldsymbol{f(x)=0}$$

where $\boldsymbol{f}$, $\boldsymbol{x}$, and $\boldsymbol{0}$ are column vectors of length n, the number of unknowns (or equivalently, the number of equations in the system).

Review of Newton's Method for a Scalar

We derived Newton's method for a scalar by Taylor series expansion

$$f(x_{k+1}) = f(x_k) + f'(x_k).(x_{k+1}-x_k)+f''(x_k).(x_{k+1}-x_k)^2/2!+\ldots$$

Since we want $f(x_{k+1})=0$, we set it to zero and neglected high order terms in the expansion to get

$$0 = f(x_k) + f'(x_k).(x_{k+1}-x_k)$$

i.e.,

$\Delta x_k = - f(x_k) \,/\, f'(x_k)$ where $\Delta x_k = x_{k+1}-x_k$

The procedure for a vector set of equations is essentially equivalent.

Newton's Method For a System

In n dimensions, Taylor's formula can be written as

$$f(x_{k+1}) = f(x_k) + f'(x_k).(x_{k+1}-x_k) + \textit{higher order terms}$$

where $f'(x_k)$ is a nxn matrix, called the ***Jacobian matrix*** of function f with respect to x. The elements of the Jacobian matrix are:

$$f'_{ij} = \frac{\partial f_i}{\partial x_j} \quad for \;\; i, j = 1,2,\ldots,n$$

Setting $f(x_{k+1}) = 0$ and neglecting higher order terms yields

$$[f']^k \{\Delta x\}^k = -\{f\}^k \qquad \text{on the } k^{th} \text{ iteration step}$$

- **Note that each iteration set yields a linear problem for the update vector $\{\Delta x\}^k$ and can be solved using standard techniques.**
- Newton's Method is generally expected to give quadratic convergence, provided that a **sufficiently accurate starting point** is known and $[f'(p)]^{-1}$ exists (p is the solution vector or point)

Newton's Method For a 3x3 System

Consider 3 non-linear equations system with 3 unknowns,

$$f_1(x_1, x_2, x_3) = 0$$

$$f_2(x_1, x_2, x_3) = 0$$

$$f_3(x_1, x_2, x_3) = 0$$

The Jacobian matrix for this system is given by $$[f'] = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_1}{\partial x_2} & \frac{\partial f_1}{\partial x_3} \\ \frac{\partial f_2}{\partial x_1} & \frac{\partial f_2}{\partial x_2} & \frac{\partial f_2}{\partial x_3} \\ \frac{\partial f_3}{\partial x_1} & \frac{\partial f_2}{\partial x_2} & \frac{\partial f_3}{\partial x_3} \end{bmatrix}$$

Re-writing it in the form: $[f']^k \{\Delta x\}^k = -\{f\}^k$

$$\begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_1}{\partial x_2} & \frac{\partial f_1}{\partial x_3} \\ \frac{\partial f_2}{\partial x_1} & \frac{\partial f_2}{\partial x_2} & \frac{\partial f_2}{\partial x_3} \\ \frac{\partial f_3}{\partial x_1} & \frac{\partial f_2}{\partial x_2} & \frac{\partial f_3}{\partial x_3} \end{bmatrix}^k \begin{bmatrix} \Delta x_1 \\ \Delta x_2 \\ \Delta x_3 \end{bmatrix}^k = \begin{bmatrix} -f_1 \\ -f_2 \\ -f_3 \end{bmatrix}^k$$

and $$\{x\}^{k+1} = \{x\}^k + \{\Delta x\}^k$$

Example – Newton's Method

Solve the system of non-linear equations:

$$x^2+y^2=1+\pi^2 \qquad (1)$$

$$\cos x + \ln y = -1 \qquad (2)$$

Solution:

Rewrite the equations in the form $f(x,y)=0$,

$$f_1(x,y) \Rightarrow x^2+y^2-(1+\pi^2) = 0$$

$$f_2(x,y) \Rightarrow \cos x + \ln y + 1 = 0$$

The Jacobian matrix is: $f' = \begin{bmatrix} \frac{\partial f_1}{\partial x} & \frac{\partial f_1}{\partial y} \\ \frac{\partial f_2}{\partial x} & \frac{\partial f_2}{\partial y} \end{bmatrix}$

Apply Newton's method to this problem: $[f']^k \{\Delta x\}^k = -\{f\}^k$

$$\begin{bmatrix} 2x & 2y \\ -\sin x & 1/y \end{bmatrix}^k \begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix}^k = \begin{bmatrix} -f_1(x_k y_k) \\ -f_2(x_k y_k) \end{bmatrix}^k$$

Numerical Results

- Starting with an initial guess of $x = 3$ and $y = 0.5$, we tabulate the following

k	x	y	Δx	Δy	f_1	f_2
0	3	0.5	0.211	0.352	-1.62	-0.683
1	3.211	0.852	-0.063	0.1385	0.1668	-0.1577
2	3.1483	0.9905	-0.0066	-0.0066	0.0232	-0.0095

- The solution is clearly approaching to the actual solution of (π,1).

- For a large class of practical problems involving non-linear systems, Newton's method will converge quadratically to the root. However, since multiple roots may exist, there is no guarantee that it will do so.

Exit Criteria

Exit Criteria for the iterative algorithm:

Exit criteria for iterative algorithms depends upon the required precision of the result.

As we have seen for the Jacobi and Gauss-Seidel methods, an iterative algorithm may be terminated if

1. An error norm is satisfied (or)
2. The maximum number of permissible iterations is reached

Let $\|f(x_k)\|_p$ be the L_p norm of f in the numerical procedure on the k^{th} iteration. Typically, the relative and absolute error tolerances are set by

$$\frac{\| f(x_k) \|_p}{\| f(x_0) \|_p} < \varepsilon_r \qquad\qquad \| f(x_k) \|_p < \varepsilon_a$$

Common values of p are 1, **2** and ∞.

Summary

- We have discussed solving systems of non-linear equations by using an extension of Newton's method to a system of unknowns.
- In Newton's method, at each iteration you will have a linear problem, which can be solved both using direct or indirect methods.
- We have worked on a simple example
- We have seen the convergence criteria for Newton's method applied to a system of non-linear equations.