Numerical Methods in Engineering

Dr. Shahram Etemadi

Evaluation

Item	Grades	Note:
Regular home works	2	All home works are collected at the beginning of every session.
Computer program home works	4	Computer program home works must be printed on paper, including the computer program, numerical results and graphs, and a short description of the solution algorithm and its theory.
Mid term exam	4	Exams are closed book
Final Exam	10	Any kinds of programmable calculators are forbidden. Simple engineering calculators are allowed. Sharing calculators is forbidden.
Absentees	No Negative points but only three absentees are allowed. More than three absentees → No final exam	

Contents

References

- Hahn D. B., and Vaentine T. D., *Essential MATLAB for Engineers and Scientists*, 3rd ed., Elsevier, 2007.
- Chapra S. C., *Numerical Methods for Engineers*, McGraw–Hill, 2006.
- Moler C., *Numerical Computing with MATLAB*, 2004, http://www.mathworks.com/moler.
- Won Y. Y., *et al.*, *Applied Numerical Methods Using MATLAB*, John Wiley & Sons, 2005.
- Hoffman, D. J., *Numerical Methods for Engineers and Scientists*, Marcel Dekker, 2001.
- Kiusalaas J. , *Numerical Methods in Engineering with MATLAB*, Cambridge university press, 2005.

Chap1: Roots of Equations

- Find the solutions of $f(x) = 0$, *where the function f is given.*
- The solutions (values of x) are known as the *roots* of the equation $f(x) = 0$, or the *zeroes* of the function $f(x)$.

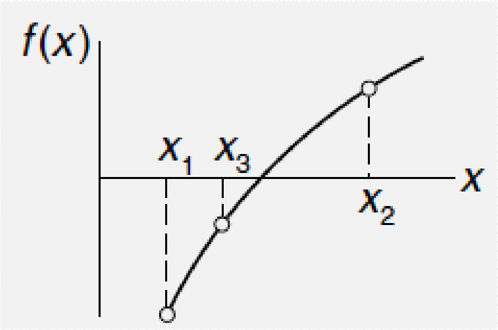
  - *f(x) can be any nonlinear equation*

for example:

$f(x) = \cosh(x)\cos(x) - 1 = 0$

Step 1: Bracketing Roots

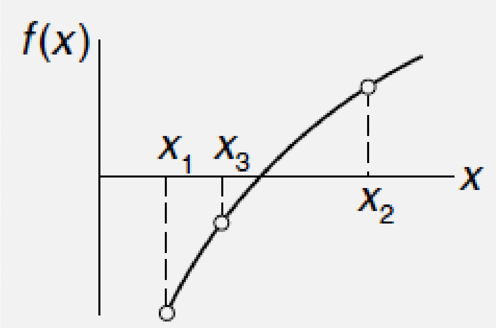
- There is a unique root in the interval (x_1, x_2) if:
  - $f(x_1).f(x_2)<0$
  - f (x) is increasing or f(x) is decreasing in (x_1, x_2)
    - Sign of df/dx must not change in (x_1, x_2)



- After a root of f (x) = 0 has been bracketed in the interval (x_1, x_2), several methods can be used to close in on it.

Bisection Method (Interval Halving Method)

- The method of bisection accomplishes this by successively halving the interval until it becomes sufficiently small.



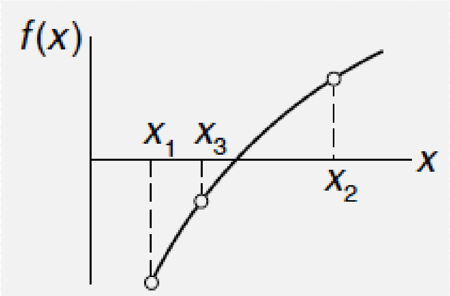
- $x3=(x1+x2)/2$

- Note: Both intervals *(x1, x2) and (x2, x3) are half the size of the original interval*

Bisection Method (Algorithm)

1. Find midpoint of (x_1, x_2)
   - $x_3=(x_1+x_2)/2$

2. Check (x_1, x_3) and (x_3, x_2) to find the interval that contains the root
   - If $f(x_1).f(x_3)<0 \rightarrow$ root lies in (x_1, x_3)
   - If $f(x_3).f(x_2)<0 \rightarrow$ root lies in (x_3, x_2)



3. Replace the original interval (x_1, x_2) by the new interval and go back to 1, ie.:
   - If root lies in (x_1, x_3) $\rightarrow$ $x_4=(x_1+x_3)/2$
   - if root lies in (x_3, x_2) $\rightarrow$ $x_4=(x_3+x_2)/2$

Bisection Method (Termination Criteria)

- The bisection is repeated until the interval has been reduced to a small value ε, *so that*

$$|x2 - x1| \leq \varepsilon$$

- The original interval *x is reduced to x/2 after one bisection,*

  x/2^2 after two bisections

  and after *n bisections it is x/2^n.*

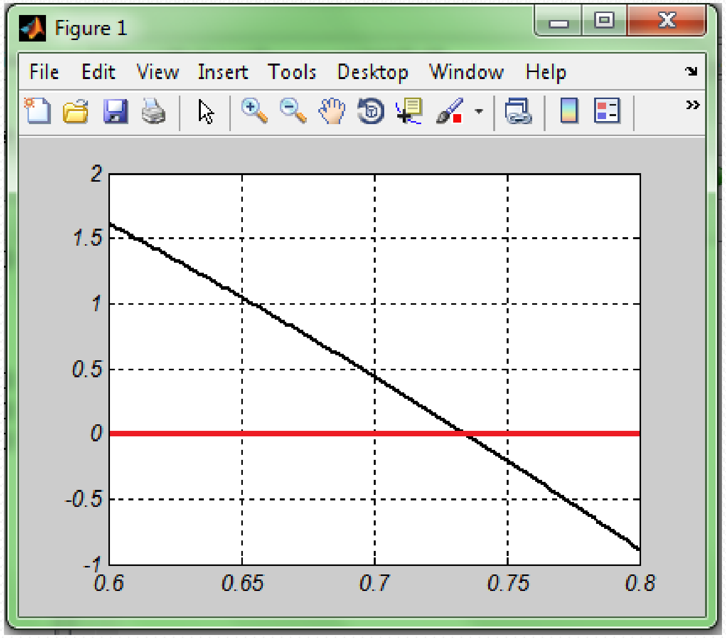
  Setting x/2^n = ε *and solving* for *n, we get*

$$n = ln\,(|x|\,/\varepsilon)/\ln 2$$

Bisection Method (Example)

- Example: Use bisection to find the root of
  $f(x) = x^3 - 10x^2 + 5 = 0$
  that lies in the interval (0.6, 0.8).

x	$f(x)$	Interval
0.6	1.616	–
0.8	−0.888	(0.6, 0.8)
$(0.6+0.8)/2 = 0.7$	0.443	(0.7, 0.8)
$(0.8+0.7)/2 = 0.75$	−0.203	(0.7, 0.75)
$(0.7+0.75)/2 = 0.725$	0.125	(0.725, 0.75)
$(0.75+0.725)/2 = 0.7375$	−0.038	(0.725, 0.7375)
$(0.725+0.7375)/2 = 0.73125$	0.044	(0.7375, 0.73125)
$(0.7375+0.73125)/2 = 0.73438$	0.003	(0.7375, 0.73438)
$(0.7375+0.73438)/2 = 0.73594$	−0.017	(0.73438, 0.73594)
$(0.73438+0.73594)/2 = 0.73516$	−0.007	(0.73438, 0.73516)
$(0.73438+0.73516)/2 = 0.73477$	−0.002	(0.73438, 0.73477)
$(0.73438+0.73477)/2 = 0.73458$	0.000	–



Note: Bisection method is not very fast

Bisection Method (Computer Program)

```
clear all;

x=-2:0.01:2;
y=x.^3+x;

plot(x,y);grid on
f3=1e3;
x1=-0.5;
x2=+1;
%for i=1:100
```

```
while abs(f3)>1e-5
    f1=x1^3+x1;
    f2=x2^3+x2;
     x3=(x1+x2)/2;
    f3=x3^3+x3;
     P1=f1*f3;
    P2=f2*f3;

    if P1<0
        x2=x3;
    else
        x1=x3;
    end
    [x1 x2]
 end
```

Incremental Search (Computer Program)

```
clear all; clc; close all
x=-2:0.1:2;
f=x.^4-x.^3+x.^2-x-1;
 plot(x,f); grid on
 x0=-2;dx=0.1;
for i=-2:dx:+2
   x1=x0+dx;
   f0=x0^4-x0^3+x0^2-x0-1;
   f1=x1^4-x1^3+x1^2-x1-1;
   P=f0*f1;

   if P<=0
      [x0 x1]
   end
   x0=x1;
End
```

- Homework: Find the root of *f(x)* in interval (2.5, 5) by Bisection method up to five place decimal accuracy.

$$f(x) = x^4 - 6.4x^3 + 6.45x^2 + 20.538x - 31.752$$

Newton-Raphson Method (Introduction)

- The Newton–Raphson algorithm can be one of the best-known method of finding roots

  it is simple and fast.

- formula can be derived from the Taylor series expansion of *f (x) about x:*

$$f(x_{i+1}) = f(x_i) + f'(x_i)(x_{i+1} - x_i) + O(x_{i+1} - x_i)^2$$

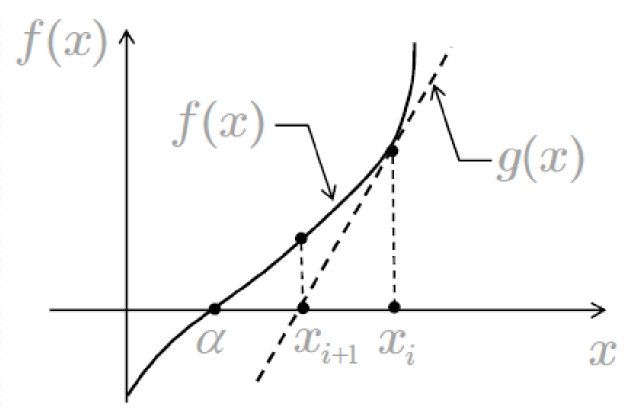
- If x_{i+1} is a root of $f(x) = 0$, then

$$0 = f(x_i) + f'(x_i)\,(x_{i+1} - x_i) + O(x_{i+1} - x_i)^2$$

- Assuming that x_{i+1} is very close to x_i, the last term can be ignored.

Newton-Raphson Method (Algorithm)

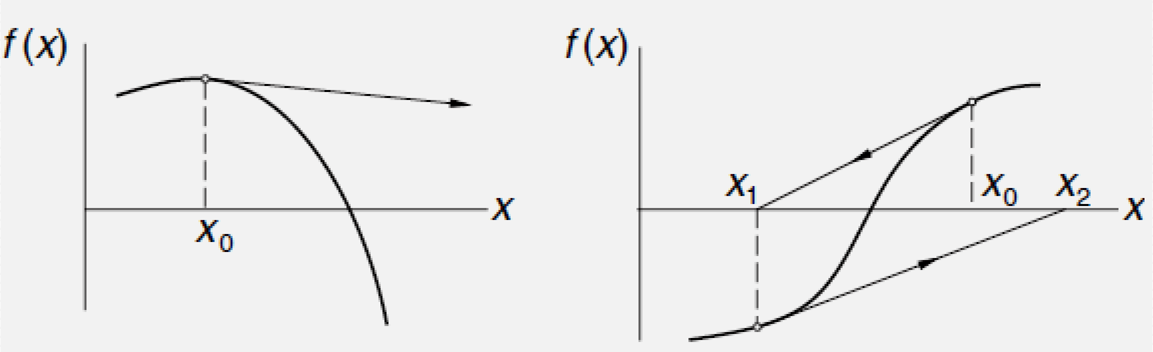
$$x_{i+1} = x_i - \frac{f(x_i)}{f'(x_i)}$$



1. Let *x be a guess for the root of f (x)* = 0.
2. Compute Dx = –*f (x)/ f '(x)*.
3. Let *x* ← *x* +*D x and repeat steps 2-3 until* |*Δx*| < *ε*.

Newton-Raphson Method (Local Convergense)

- Newton-Raphson method is very fast, if it converges.
  - Newton–Raphson method converges fast near the root, but its global convergence characteristics are poor.
  - The reason is that the tangent line is not always an acceptable approximation of the function, as illustrated in these two examples



Newton-Raphson Method (Example)

- A root of $f(x) = x^3 - 10x^2 + 5 = 0$ lies close to $x = 0.7$. Compute this root with the Newton–Raphson method.

$$f(x) = x^3 - 10x^2 + 5 = 0$$

$$f'(x) = 3x^2 - 20x$$

$$x \leftarrow x - \frac{f(x)}{f'(x)} = x - \frac{x^3 - 10x^2 + 5}{3x^2 - 20x} = \frac{2x^3 - 10x^2 - 5}{x(3x - 20)}$$

- It takes only two iterations to reach a five place decimal accuracy.

$$x \leftarrow \frac{2(0.7)^3 - 10(0.7)^2 - 5}{0.7\,[3(0.7) - 20]} = 0.735\,36$$

$$x \leftarrow \frac{2(0.735\,36)^3 - 10(0.735\,36)^2 - 5}{0.735\,36\,[3(0.735\,36) - 20]} = 0.734\,60$$

Newton-Raphson Method (Computational Cost)

- The method needs to compute *f'(x)* in each iteration and sometimes *f'(x)* is a lot more complicated than *f(x)*. In these cases the Newton-Raphson method, although needs few number of iterations, but every iteration has large number of computations.

  For example: *f(x)=cos(x^2) . tan(2x) / ln(sin(x))*

Homework: Find the root of *f(x)* in interval (3.5, 5) by Newton-Raphson method up to five place decimal accuracy. Use a computer program.

$$f(x) = x^4 - 6.4x^3 + 6.45x^2 + 20.538x - 31.752$$

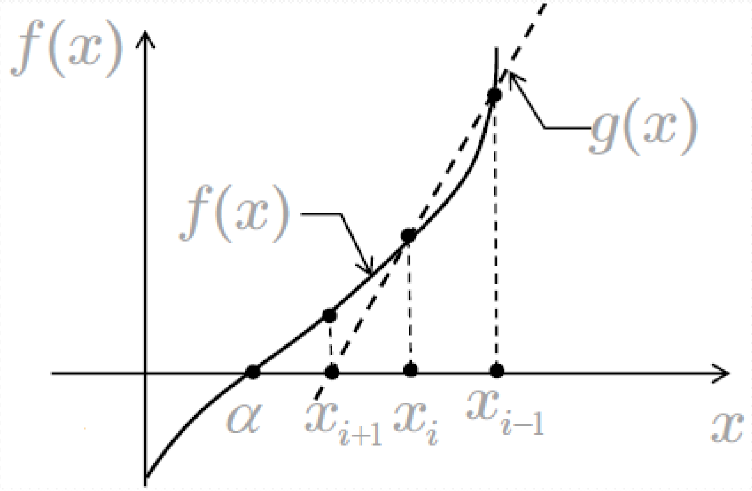
Secant Formula

- If calculating f '(x) is not economic, secant formula replaces the Newton method:

$$x_{i+1} = x_i - \frac{f(x_i)}{g'(x_i)}$$

- Approximate slope of *f(x)*

$$g'(x_i) = \frac{f(x_i) - f(x_{i-1})}{x_i - x_{i-1}}$$



- Note:

  Two initial points ***Xi-1*** and ***Xi*** are necessary to begin iteration.

Secant Formula (Algorithm)

The Secant Method.

Given $f(x) = 0$, ε and two initial points x_0, x_1;

Given max =maximum number of iterations;

For $i = 1$ to max

Compute $g'(x_i) = \frac{f(x_i)-f(x_{i-1})}{x_i-x_{i-1}}$;

Compute $x_{i+1} = x_i - \frac{f(x_i)}{g'(x_i)}$;

If $|x_{i+1} - x_i| < \varepsilon$;

Solution $= x_{i+1}$;

Stop the iterations;

Endif

Endfor

Excercise

- Use a computer program to find zero of following function by Secant method

f(x)=exp(x^2)-sin(x)

Moller Method

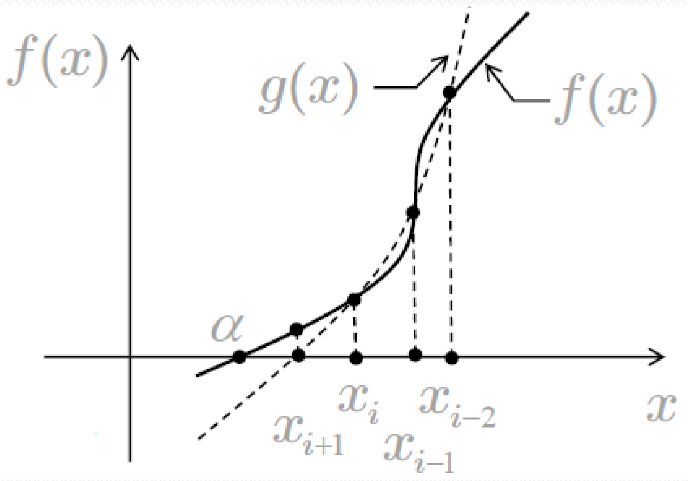
- Uses a second order polynomial to estimate the function ***f(x)*** and find its root in every iteration.

$$g(x) = a(x - x_i)^2 + b(x - x_i) + c$$

- Considering three points on *g(x)*:

$$(x_{i-2}, f_{i-2}),\ (x_{i-1}, f_{i-1}),\ (x_i, f_i)$$

a, *b*, and *c* can be obtained:



$$\left.\begin{array}{l} f_i = c \\ f_{i-1} = ah_1^2 + bh_1 + c \\ f_{i-2} = ah_2^2 + bh_2 + c \end{array}\right\} \rightarrow \begin{cases} ah_1^2 + bh_1 = \delta_1 \\ ah_2^2 + bh_2 = \delta_2 \end{cases}$$

$$h_1 = (x_{i-1} - x_i), \qquad \delta_1 = (f_{i-1} - f_i)$$
$$h_2 = (x_{i-2} - x_i), \qquad \delta_2 = (f_{i-2} - f_i)$$

Moller Method (Algorithm)

$$a=\frac{\delta_1 h_2-\delta_2 h_1}{h_1 h_2 (h_1-h_2)} \qquad b=\frac{\delta_2 h_1^2-\delta_1 h_2^2}{h_1 h_2 (h_1-h_2)}$$

$$h_1=(x_{i-1}-x_i), \qquad \delta_1=(f_{i-1}-f_i)$$
$$h_2=(x_{i-2}-x_i), \qquad \delta_2=(f_{i-2}-f_i)$$

- Algorithm: $$g(x_{i+1})=a(x_{i+1}-x_i)^2+b(x_{i+1}-x_i)+c=0$$

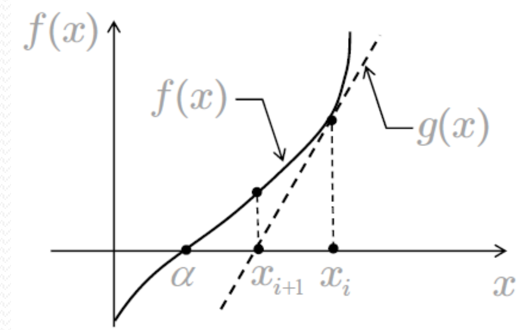
$$x_{i+1}=x_i-\frac{2c}{b\pm\sqrt{b^2-4ac}}$$

Moller Method (Note)

- three initial points ***X1, X2, X3*** are necessary
- \+ or - : choose the one that has a sign similar to *b*
- During the search for real root, if the method results in a complex value, ignore the imaginary part and continue the search.

$$x_{i+1} = x_i - \frac{2c}{b \pm \sqrt{b^2 - 4ac}}$$

False Position Method



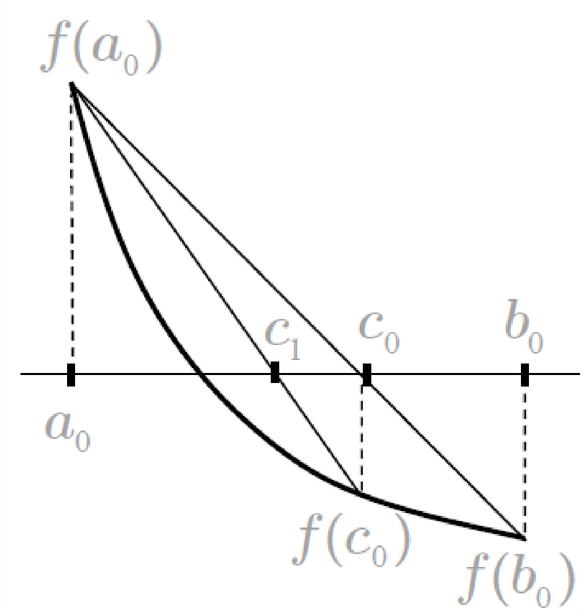
- Instead of the real slope (Newton method), uses a line connecting both ends of the function in the interval [a, b].

- Slope of the line: $$g'(x) = \frac{f(b) - f(a)}{b - a}$$
- The line: $f(x) = f(a_0) + g'(x)(x - a_0)$

  meets x-axis in *C* where $f(C) = 0$. *So:*

$$f(c) = 0 \rightarrow c = a - \frac{f(b)}{g'(x)}$$

C is the first guess.



False Position Algorithm

1. Determine the slope g': $g'(x) = \frac{f(b) - f(a)}{b - a}$
2. Calculate *C*: $c = a - \frac{f(b)}{g'(x)}$
3. Check a termination criteria, and if the result is satisfactory, then stop

$$|b - a| \le \varepsilon_1 \quad and/or \quad |f(c)| \le \varepsilon_2$$

4. Replace b by the value of C and go back to step 1.

Fixed Point Method

- Turns the function $f(x)=0$ to a recursive formula like $x=g(x)$ where x converges to $\boldsymbol{C}$.

$$f(x) = 0 \rightarrow x = g(x)$$

- **Algorithm:** $\boldsymbol{X_{i+1}=g(X_i)}$

1. Choose an arbitrary value of x *like* $\boldsymbol{C_0}$ in $[a, b]$.
2. Evaluate $\boldsymbol{C_1}$ ***from*** $\boldsymbol{C_1= g(C_0)}$.
3. Check a termination criteria like $f(C_1)<\varepsilon$, and if the result is satisfactory, then stop.
4. If not repeat step 2.

Fixed Point Method (Note)

- *g(x)* is not necessarily unique. Eg:

$$x^2 + \sqrt[3]{x} - 10 = 0 \Rightarrow \begin{cases} x = \sqrt{10 - \sqrt[3]{x}} \\ x = \left(10 - x^2\right)^3 \\ x = 10 + x - x^2 - \sqrt[3]{x} \end{cases}$$

- *g(x)* is acceptable only if $|g'(X_i)| < 1$ in *[a, b]*

  (Convergence Criteria)

Fixed Point Method (Example)

- ***f(x)=x^2-1=0*** in *[1, 1.5]*
- *First guess:*

$$x = \frac{2}{x} = g_a(x)? \quad \rightarrow g_a'(x) = -\frac{2}{x^2} \quad \rightarrow |g_a'(x)| = \left|\frac{2}{x^2}\right| \overset{?}{<} 1 \ \forall x \in I$$

- *Second guess: (x+x=x+2/x)*

$$x = \frac{1}{2}\left(x + \frac{2}{x}\right) = g_a(x)? \qquad |g_a'(x)| = \frac{1}{2}\left|1 - \frac{2}{x^2}\right| \overset{?}{<} 1 \quad \forall x \in I$$

- *The second guess meets the expectations*

Fixed point Method (Exercise)

- Find the root of $x^3+x^2-1=0$ in [-1, +1] by using fixed point method.

Aitken Method

- A method to accelerate convergence:

$$x_{k+3} = x_k - \frac{(\Delta x_k)^2}{\Delta^2 x_k}, \qquad \Delta x_k = x_{k+1} - x_k, \qquad \Delta^2 x_k = x_{k+2} - 2x_{k+1} + x_k$$

- Can be used once in every four steps in a stable iteration. Eg:
  - $x_{k+1} = g(x_k)$
  - $x_{k+2} = g(x_{k+1})$
  - $x_{k+3} = from\ Aitken\ method,$
  - $x_{k+4} = g(x_{k+3})$
  - $x_{k+5} = g(x_{k+4})$
  - $x_{k+6} = from\ Aitken\ method,$
  - ...

Set of Nonlinear Equations

Example

$$\begin{cases} x+y+z=3 \\ x^2+y^2+z^2=5 \\ e^x+xy-xz=1 \end{cases} \xrightarrow{????} \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \\ 2 \end{bmatrix}$$

Set of Nonlinear Equations

- Remember the 1D-Newton-raphson method

$$x^{(k+1)} = x^{(k)} - \frac{f\left(x^{(k)}\right)}{f'\left(x^{(k)}\right)} = x^{(k)} - \left[f'\left(x^{(k)}\right)\right]^{-1} f\left(x^{(k)}\right)$$

- Is Generalization possible?

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} - \left[\mathbf{f}'\left(\mathbf{x}^{(k)}\right)\right]^{-1} \mathbf{f}\left(\mathbf{x}^{(k)}\right)$$

$$\mathbf{J} = \mathbf{f}'(\mathbf{x}) = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_1}{\partial x_2} & \cdots & \frac{\partial f_1}{\partial x_n} \\ \frac{\partial f_2}{\partial x_1} & \frac{\partial f_2}{\partial x_2} & \cdots & \frac{\partial f_2}{\partial x_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial f_n}{\partial x_1} & \frac{\partial f_n}{\partial x_2} & \cdots & \frac{\partial f_n}{\partial x_n} \end{bmatrix}$$

$$\left[\mathbf{f}'\left(\mathbf{x}^{(k)}\right)\right] \equiv Jacobian$$

Set of Nonlinear Equations

- General Form:

$$\begin{cases} f_1(x_1, x_2, \cdots, x_n) = 0 \\ f_2(x_1, x_2, \cdots, x_n) = 0 \\ \vdots \\ f_n(x_1, x_2, \cdots, x_n) = 0 \end{cases}$$

- Using Teylor series expansion:

$$f_i(x_1 + h_1, x_2 + h_2, \cdots, x_n + h_n) = f_i(x_1, x_2, \cdots, x_n) +$$

$$h_1 \frac{\partial f_i}{\partial x_1} + h_2 \frac{\partial f_i}{\partial x_2} + \cdots + h_n \frac{\partial f_i}{\partial x_n} + HOTs$$

$$\boldsymbol{h} = (h_1, h_2, \cdots, h_n)^T$$

Set of Nonlinear Equations

- Assume $\boldsymbol{x}^{(0)}$ as initial guess
  - $\boldsymbol{x}^{(1)} = \boldsymbol{x}^{(0)} + \boldsymbol{h}$

$$\underbrace{\begin{bmatrix} f_1\left(x_1^{(1)}+h_1, x_2^{(1)}+h_2, \cdots, x_n^{(1)}+h_n\right) \\ f_2\left(x_1^{(1)}+h_1, x_2^{(1)}+h_2, \cdots, x_n^{(1)}+h_n\right) \\ \vdots \\ f_n\left(x_1^{(1)}+h_1, x_2^{(1)}+h_2, \cdots, x_n^{(1)}+h_n\right) \end{bmatrix}}_{\boldsymbol{f}\left(\boldsymbol{x}^{(1)}+\boldsymbol{h}\right)} = \underbrace{\begin{bmatrix} f_1\left(x_1^{(0)}, x_2^{(0)}, \cdots, x_n^{(0)}\right) \\ f_2\left(x_1^{(0)}, x_2^{(0)}, \cdots, x_n^{(0)}\right) \\ \vdots \\ f_n\left(x_1^{(0)}, x_2^{(0)}, \cdots, x_n^{(0)}\right) \end{bmatrix}}_{\boldsymbol{f}\left(\boldsymbol{x}^{(0)}\right)} + \underbrace{\begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_1}{\partial x_2} & \cdots & \frac{\partial f_1}{\partial x_n} \\ \frac{\partial f_2}{\partial x_1} & \frac{\partial f_2}{\partial x_2} & \cdots & \frac{\partial f_2}{\partial x_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial f_n}{\partial x_1} & \frac{\partial f_n}{\partial x_2} & \cdots & \frac{\partial f_n}{\partial x_n} \end{bmatrix}}_{\boldsymbol{J}=\boldsymbol{f}'\left(\boldsymbol{x}^{(0)}\right)} \underbrace{\begin{bmatrix} h_1 \\ h_2 \\ \vdots \\ h_n \end{bmatrix}}_{\boldsymbol{h}}$$

$$\boldsymbol{f}\left(\boldsymbol{x}^{(1)}+\boldsymbol{h}\right) = \boldsymbol{f}\left(\boldsymbol{x}^{(0)}\right) + \boldsymbol{f}'\left(\boldsymbol{x}^{(0)}\right)\boldsymbol{h}$$

Set of Nonlinear Equations

$$f\left(x^{(1)}+h\right)=f\left(x^{(0)}\right)+f'\left(x^{(0)}\right)h\approx 0$$

$$h\approx -\left[f'\left(x^{(0)}\right)\right]^{-1}f\left(x^{(0)}\right)$$

- Recursive Formula

$$x^{(k+1)}=x^{(k)}-\left[f'\left(x^{(k)}\right)\right]^{-1}f\left(x^{(k)}\right)$$

Set of Nonlinear Equations

- Example:

$$\begin{cases} x+y+z=3 \\ x^2+y^2+z^2=5 \\ e^x+xy-xz=1 \end{cases} \Rightarrow \boldsymbol{J}(\boldsymbol{f}) = \begin{bmatrix} 1 & 1 & 1 \\ 2x & 2y & 2z \\ e^x+y-z & x & -x \end{bmatrix}$$

$$\begin{bmatrix} x^{(k+1)} \\ y^{(k+1)} \\ z^{(k+1)} \end{bmatrix} = \begin{bmatrix} x^{(k)} \\ y^{(k)} \\ z^{(k)} \end{bmatrix} - \begin{bmatrix} 1 & 1 & 1 \\ 2x^{(k)} & 2y^{(k)} & 2z^{(k)} \\ \left(e^x\right)^{(k)}+y^{(k)}-z^{(k)} & x^{(k)} & -x^{(k)} \end{bmatrix}^{-1} \begin{bmatrix} x^{(k)}+y^{(k)}+z^{(k)}-3 \\ \left(x^{(k)}\right)^2+\left(y^{(k)}\right)^2+\left(z^{(k)}\right)^2-5 \\ \left(e^x\right)^{(k)}+x^{(k)}y^{(k)}-x^{(k)}z^{(k)}-1 \end{bmatrix}$$

Set of Nonlinear Equations

- Iterative method for a set of equations:

$$\begin{cases} x^2 - 2x + 0.5 + y = 0 \\ x^2 + 4y^2 - 4 = 0 \end{cases} \xrightarrow[\text{of } 2^{nd} \text{ eq.}]{\text{Add } \pm 8y \text{ to leftside}} \begin{cases} x = \dfrac{x^2 - y + 0.5}{2} \\ y = \dfrac{-x^2 - 4y^2 + 8y + 4}{8} \end{cases}$$

$$\begin{cases} x_{k+1} = \dfrac{x_k^2 - y_k + 0.5}{2} \\ y_{k+1} = \dfrac{-x_k^2 - 4y_k^2 + 8y_k + 4}{8} \end{cases} \rightarrow \begin{cases} x_0 = 0,\ y_0 = 1 \\ x_9 = -0.22222146, y_9 = 0.9938084 \end{cases}$$

Set of Nonlinear Equations

- Fixed point method

$$\begin{cases} f_1(x_1, x_2, \cdots, x_n) = 0 \\ f_2(x_1, x_2, \cdots, x_n) = 0 \\ \vdots \\ f_n(x_1, x_2, \cdots, x_n) = 0 \end{cases} \rightarrow \begin{cases} x_1 = g_1(x_1, x_2, \cdots, x_n) \\ x_2 = g_2(x_1, x_2, \cdots, x_n) \\ \vdots \\ x_n = g_n(x_1, x_2, \cdots, x_n) \end{cases}$$

- Iterative Formula:

$$\begin{cases} x_1^{(k+1)} = g_1\left(x_1^{(k)}, x_2^{(k)}, \cdots, x_n^{(k)}\right) \\ x_2^{(k+1)} = g_2\left(x_1^{(k)}, x_2^{(k)}, \cdots, x_n^{(k)}\right) \\ \vdots \\ x_n^{(k+1)} = g_n\left(x_1^{(k)}, x_2^{(k)}, \cdots, x_n^{(k)}\right) \end{cases}$$

Set of Nonlinear Equations

Converges if:

$$\begin{cases} \sum_{i=1}^{n} \left| \frac{\partial g_1}{\partial x_i}(x_1, x_2, \cdots, x_n) \right| < 1 \\ \sum_{i=1}^{n} \left| \frac{\partial g_2}{\partial x_i}(x_1, x_2, \cdots, x_n) \right| < 1 \\ \vdots \\ \sum_{i=1}^{n} \left| \frac{\partial g_n}{\partial x_i}(x_1, x_2, \cdots, x_n) \right| < 1 \end{cases}$$

- Example

$$\begin{cases} \left| \frac{\partial g_1}{\partial x} \right| + \left| \frac{\partial g_1}{\partial y} \right| < 1 \\ \left| \frac{\partial g_2}{\partial x} \right| + \left| \frac{\partial g_2}{\partial y} \right| < 1 \end{cases}$$

Set of Nonlinear Equations

- Example:

$$\begin{cases} 3x_1 - \cos(x_2 x_3) - \frac{1}{2} = 0 \\ x_1^2 - 81(x_2 + 0.1)^2 + \sin(x_3) + 1.06 = 0 \\ e^{-x_1 x_2} + 20x_3 + \frac{10\pi - 3}{3} = 0 \end{cases}$$

- Fixed point model:

$$\begin{cases} x_1 = \frac{1}{3}\cos(x_2 x_3) + \frac{1}{6} \\ x_2 = \frac{1}{9}\sqrt{x_1^2 + \sin(x_3) + 1.06} - 0.1 \\ x_3 = -\frac{1}{20} e^{-x_1 x_2} + \frac{10\pi - 3}{3} \end{cases}$$

- Assume that D= -1<x1, x2, x3<+1

Set of Nonlinear Equations

- Existence of solution:

$$\begin{cases} \left|g_1(x_1,x_2,x_3)\right| \le \frac{1}{3}\left|\cos(x_2 x_3)\right| + \frac{1}{6} \le \frac{1}{2} \\ \left|g_2(x_1,x_2,x_3)\right| = \left|\frac{1}{9}\sqrt{x_1^2+\sin(x_3)+1.06}-0.1\right| \le \frac{1}{9}\sqrt{1+\sin(1)+1.06}-0.1 < 0.90 \\ \left|g_3(x_1,x_2,x_3)\right| = \left|-\frac{1}{20}e^{-x_1x_2}+\frac{10\pi-3}{3}\right| \le \frac{1}{20}e+\frac{10\pi-3}{3} < 0.61 \end{cases}$$

- $\forall i = 1,2,3$ ، $-1 \le g_i(x_1,x_2,x_3) \le 1$ Hence
- $x \in D$, $G(x) \in D$

Set of Nonlinear Equations

- Convergence investigation:

$$\left|\frac{\partial g_1}{\partial x_1}\right| = 0$$

$$\left|\frac{\partial g_1}{\partial x_2}\right| \le \frac{1}{3}|x_3||\sin(x_2 x_3)| \le \frac{1}{3}\sin(1) < 0.281$$

$$\left|\frac{\partial g_1}{\partial x_3}\right| = \frac{1}{3}|x_2||\sin(x_2 x_3)| \le \frac{1}{3}\sin(1) < 0.281$$

$$\left|\frac{\partial g_2}{\partial x_1}\right| = \frac{|x_1|}{9\sqrt{x_1^2 + \sin(x_3) + 1.06}} \le \frac{1}{9\sqrt{0.218}} < 0.238$$

$$\left|\frac{\partial g_2}{\partial x_2}\right| = 0$$

$$\left|\frac{\partial g_2}{\partial x_3}\right| = \frac{|\cos(x_3)|}{18\sqrt{x_1^2 + \sin(x_3) + 1.06}} \le \frac{1}{18\sqrt{0.218}} < 0.119$$

$$\left|\frac{\partial g_3}{\partial x_1}\right| = \frac{|x_2|}{20} e^{-x_1 x_2} \le \frac{1}{20} e \le 0.14$$

$$\left|\frac{\partial g_3}{\partial x_2}\right| = \frac{|x_1|}{20} e^{-x_1 x_2} \le \frac{1}{20} e \le 0.14$$

$$\left|\frac{\partial g_3}{\partial x_3}\right| = 0$$

$$\left|\frac{\partial g_i(\mathbf{x})}{\partial x_j}\right| \le 0.281, \qquad i,j = 1,2,3$$

Set of Nonlinear Equations

- Homework
- Use the fixed point method to solve the given set of equations:

$$\begin{cases} x+y+z=3 \\ x^2+y^2+z^2=5 \\ e^x+xy-xz=1 \end{cases} \xrightarrow{\text{Solution}} \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \\ 2 \end{bmatrix}$$

Roots of Polynomials

- Horner Laws

$$P_n(x) = a_n x^n + a_{n-1} x^{n-1} + \cdots + a_1 x^1 + a_0$$

$$= (x - z)\left(b_{n-1} x^{n-1} + b_{n-2} x^{n-2} + \cdots + b_1 x^1 + b_0\right) + R$$

$$\begin{cases} a_n = b_{n-1} \\ a_{n-1} = b_{n-2} - z b_{n-1} \\ a_{n-2} = b_{n-3} - z b_{n-2} \\ \quad \vdots \\ a_1 = b_0 - z b_1 \\ a_0 = R - z b_0 \end{cases} \Leftrightarrow \begin{cases} b_{n-1} = a_n \\ b_{n-2} = a_{n-1} + z b_{n-1} \\ b_{n-3} = a_{n-2} + z b_{n-2} \\ \quad \vdots \\ b_0 = a_1 + z b_1 \\ R = a_0 + z b_0 \end{cases} \rightarrow \begin{cases} b_{(n)} = 0 \\ b_{(i)} = a_{(i+1)} + z b_{(i+1)} \end{cases}$$

Roots of Polynomials
Horner method

- Example:

$$P(x) = x^3 - 3x^2 + 4x - 2$$

$$= (x-1)\,Q(x)$$

$$\begin{cases} b_2 = a_3 = 1.0 \\ b_1 = a_2 + zb_2 = -3.0 + (1.0)(1.0) = -2.0 \\ b_0 = a_1 + zb_1 = 4.0 + (1.0)(-2.0) = 2.0 \end{cases}$$

$$Q(x) = x^2 - 2x + 2$$

Roots of Polynomials
Newton-Raphson mehtod

$$P(x) = x^3 - 3x^2 + 4x - 2$$

$$Q(x) = 3x^2 - 6x + 4$$

$$x_{i+1} = x_i - \frac{P(x)}{Q(x)}$$

$$x_0 = 1.5,$$

$$x_1 = x_0 - \frac{P(x_0)}{P'(x_0)} = 1.5 - \frac{0.6250}{1.750} = 1.142857$$

Repeated Roots of Polynomials

- 1^{st} Modified Newton-Raphson

$$x_{i+1} = x_i - m \frac{f(x_i)}{f'(x_i)}$$

$$f(x) = (x-\alpha)^m \, h(x)$$

- 2^{nd} Modified Newton-Raphson

$$x_{i+1} = x_i - \frac{f(x_i)\, f'(x_i)}{\left[f'(x_i) \right]^2 - f(x_i)\, f''(x_i)}$$

Repeated Roots of Polynomials

❖ Example

$$f(x) = x^3 - x^2 - x + 1, \qquad x_0 = 1.5$$

$$= (x+1)(x-1)(x-1)$$

$$f'(x) = 3x^2 - 2x - 1, \qquad f''(x) = 6x - 2$$

$$f(1.5) = 0.6250 \qquad f'(1.5) = 2.750, \qquad f''(1.5) = 7.0$$

$$x_1 = x_0 - \frac{f(x_0)}{f'(x_0)} \rightarrow x_1 = 1.5 - \frac{0.6250}{2.750} = 2.272727$$

$$x_1 = x_0 - m\frac{f(x_0)}{f'(x_0)} \rightarrow x_1 = 1.5 - 2.0\frac{0.6250}{2.750} = 1.045355$$

$$x_1 = x_0 - \frac{f(x_0)\,f'(x_0)}{\left[f'(x_0)\right]^2 - f(x_0)\,f''(x_0)} \rightarrow$$

$$x_1 = 1.5 - \frac{(0.6250)\,(2.750)}{[2.750]^2 - (0.6250)\,(7.0)} = 0.960784$$