2

SYSTEMS OF NON-LINEAR EQUATIONS

- Introduction
- Graphical Methods
- Close Methods
- Open Methods
- Polynomial Roots
- System of Multivariable Equations

2.1 Introduction

- Problems involving non-linear equations in engineering include optimisation, solving differential equations and eigen values.

- Usually, a typical problem is to find *roots* for a nonlinear equation, e.g.

$$f(x) = ax^2 + bx + c = 0 \tag{2.1}$$

where its analytical solutions are

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a} \tag{2.2}$$

- However, for other non-linear equations, finding a root may not be a simple task and many can only be determined via numerical approach only.

- Consider the following function:

$$f(r_k) = 0 \qquad \forall\, k = 1, 2, \ldots, n$$

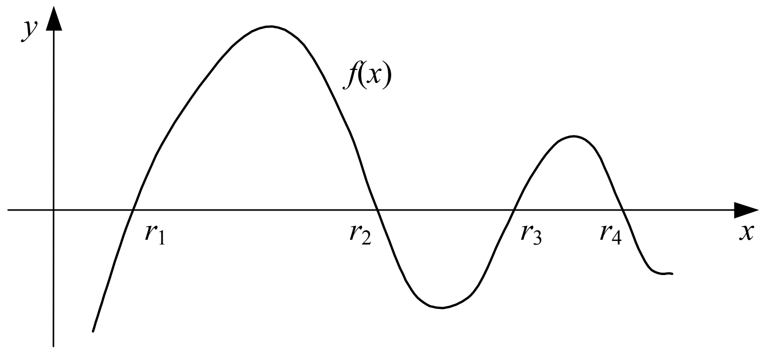

FIGURE 2.1 Roots for $f(x)$

The function $y = f(x)$ may take the form of one of the following categories:

1. **Linear** functions — i.e. $f(x) = ax + b$ which is simple to solve,
2. **Polynomials** or **algebraic** functions — i.e., in the following form:

   $f_n y^n + f_{n+1} y^{n-1} + \cdots + f_1 y + f_0 = 0$ or $f_n(x) = a_0 + a_1 x + \cdots + a_n x^n$
3. **Transcendent** or **non-algebraic** functions — i.e., can be expanded into infinite series in the form:

$$f(x) = \sum_{k=0}^{\infty} a_k x^k \text{ , e.g. } f(x) = e^x = 1 + x + \frac{1}{2!}x^2 + \frac{1}{3!}x^3 + \cdots$$

2.2 Graphical Methods

- The simplest way to estimate the roots or solutions of $f(x)=0$ is from its graph, i.e. from the intersection of the graph with the x-axis.

- Consider the following function:

$$f(c)=1.5\left(e^{-0.2c/5}\right)-0.5\,.$$

c	$f(c)$
10	0.5055
20	0.1740
30	−0.0482
40	−0.1972
50	−0.2970

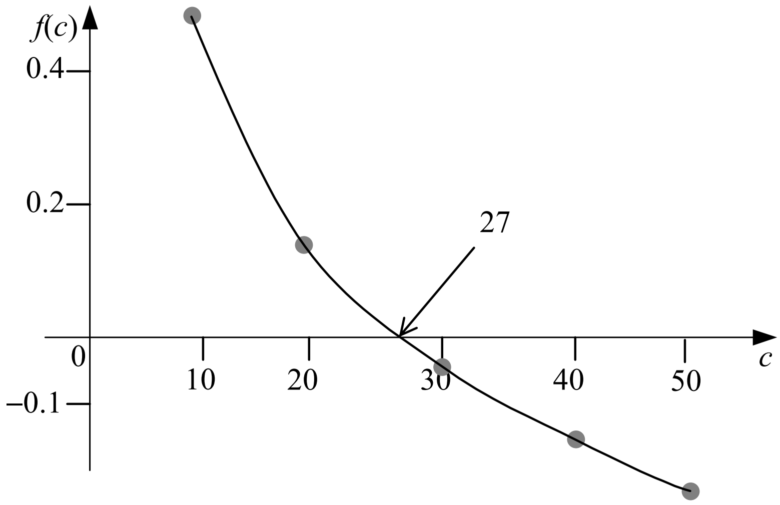

FIGURE 2.2 A graphical method to find the root of a function

- This method is not accurate but can be used to obtain an approximate value.

2.3 Close Methods

- The **close** or *bracketing* methods find roots based on a range specified by two values, which is assumed to contain a root.

- For the **bisection** method, the root which lies in the range of (x_1, x_2) is determined by looking at the sign of the function at both ends:

$$f(x_1)\cdot f(x_2)<0$$

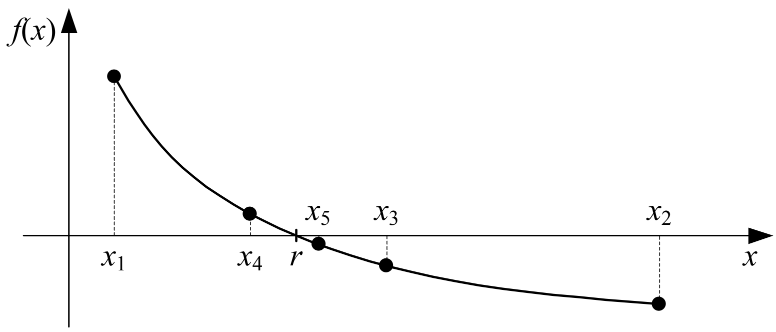

FIGURE 2.3 The bisection method for searching roots

If the product of $f(x_1)\cdot f(x_2)$ is negative, then the next approximate root is

$$x_3=\frac{x_1+x_2}{2} \tag{2.3}$$

For the next iteration, if $f(x_1)\cdot f(x_3)<0$ then the root lies in (x_1, x_3), otherwise if $f(x_3)\cdot f(x_2)<0$ then the root lies in (x_3, x_2). In Fig. 2.3:

$$x_4=\frac{x_1+x_3}{2}, \quad x_5=\frac{x_3+x_4}{2}, \quad x_6=\frac{x_4+x_5}{2}$$

until $f(x_3)\approx 0$ as defined by a convergence or termination criterion.

- For the **false position** method, a *linear interpolation* is used to give a better approximation (see Fig. 2.4), thus leading to faster convergence than the bisection method:

$$\frac{ST}{PT}=\frac{RQ}{PQ}$$

$$\frac{x_2-x_3}{f(x_2)}=\frac{x_2-x_1}{f(x_2)-f(x_1)}$$

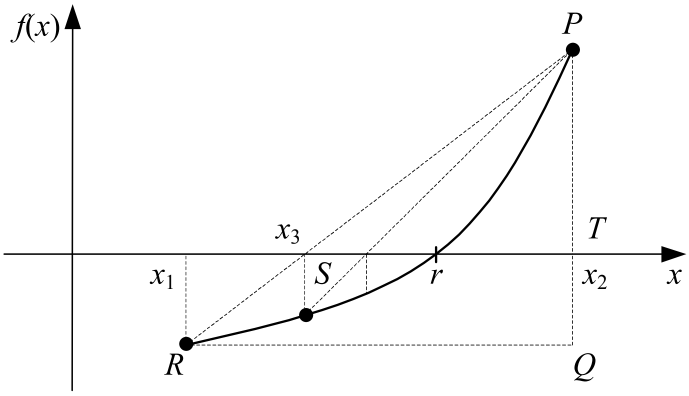

FIGURE 2.4 The false position method for searching roots

Upon rearrangement:

$$x_3 = x_2 - \frac{(x_2 - x_1)f(x_2)}{f(x_2) - f(x_1)}$$

Hence, the formula for searching roots can be generalised as followed:

$$x_{k+1} = x_{\text{fixed}} - \frac{(x_{\text{fixed}} - x_k)f(x_{\text{fixed}})}{f(x_{\text{fixed}}) - f(x_k)} \tag{2.4}$$

where $k = 2, 3, 4, \ldots$

Example 2.3

Use the bisection method to find the root of $e^{-x} - x$ in the range of [0, 1]. Compare with the real value of 0.567143.

Solution

$$f(x) = e^{-x} - x$$
$$f(0) = 1, \quad f(1) = -0.63212$$

Lelaran	x_1	x_2	x_3	$f(x_3)$	ε_a %	ε_t %
1	0	1	0.5	0.10651	-	11.839
2	0.5	1	0.75	−0.27763	33.333	32.242
3	0.5	0.75	0.625	−0.08974	20.000	10.201
4	0.5	0.625	0.5625	0.00728	11.111	0.819
5	0.5625	0.625	0.59375	−0.04149	5.263	4.691
⋮	⋮	⋮	⋮	⋮	⋮	⋮
20	0.56714	0.56714	0.56714	−1.75E−06	3.35E−04	1.95E−04

Example 2.4

Use the false position method to find the root of $e^{-x} - x$ in the range of [0, 1]. Compare with the real value of 0.567143.

Solution

$$f(x) = e^{-x} - x$$

$$f(0) = 1, \quad f(1) = -0.63212$$

For $k = 2,3,4,\ldots$, Eq. (2.4) can be written as

$$x_{k+1} = 0 - \frac{(1)(0 - x_k)}{1 - f(x_k)}$$

$$x_3 = 0 - \frac{1(0-1)}{1-(-0.63212)}$$

$$= 0.61270$$

and,

$$f(x_3) = -0.07081$$

Lelaran	x_1	x_2	x_3	$f(x_3)$	ε_a %	ε_t %
1	0	1	0.61270	−0.07081	-	8.033
2	0	0.61270	0.57218	−0.00789	7.082	0.888
3	0	0.57218	0.56770	−0.00087	0.789	0.098
4	0	0.56770	0.56720	−0.00009	0.088	9.99E−05
5	0	0.56720	0.56715	−0.00001	0.009	1.18E−05

2.4 Open Method

- The **open** method finds roots through *iterations* using one or two points as initial points.

- The simplest method is the **fixed point iteration** method, where the original equation $f(x) = 0$ is modified to be:

$$x_{i+1} = g(x_i) \tag{2.5}$$

and, the iteration starts with an initial value x_0. This method leads to a very slow convergence and may be diverge.

Example 2.6

User the fixed point iteration method to obtain the root of $e^{-x} - x$ accurate to three decimal places. Take the initial value of $x_0 = 0$.

Penyelesaian

$$f(x) = e^{-x} - x = 0$$

$$x_{i+1} = e^{-x_i}$$

i	x_i	ε_t %	i	x_i	ε_t %
0	0	100	8	0.560	1.24
1	1	76.3	9	0.571	0.705
2	0.368	35.1	10	0.565	0.399
3	0.692	22.0	11	0.568	0.227
4	0.500	11.8	12	0.566	0.128
5	0.606	6.89	13	0.568	0.073
6	0.545	3.84	14	0.567	0.041
7	0.580	2.20	15	0.567	0.023

- The most popular method for searching roots is the **Newton-Raphson** method, which usually leads to a fast convergence (see Fig 2.4).

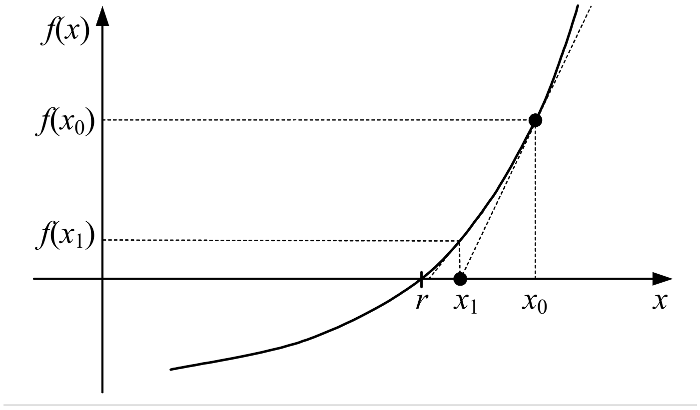

FIGURE 2.5 The Newton-Raphson method for searching roots

This is a gradient-based method, where the formula can be derived from the first order derivative:

$$f'(x_0) = \frac{f(x_0)}{x_0 - x_1} \quad \text{or} \quad x_1 = x_0 - \frac{f(x_0)}{f'(x_0)}$$

The generalised form for the *Newton-Raphson formula* is

$$x_{i+1} = x_i - \frac{f(x_i)}{f'(x_i)} \tag{2.6}$$

where $i = 0, 1, 2, \ldots$. The error for this method can be derived from the Taylor series expansion assuming the real root as $r = x_k + \varepsilon$, then

$$\begin{aligned} f(r) &= f(x_k + \varepsilon), \\ &= f(x_k) + \varepsilon f'(x_k) + \frac{\varepsilon^2}{2!} f''(x_k) + \cdots \end{aligned}$$

By neglecting ε^2 and other higher order terms, the relative error can be obtained:

$$f(x_k) + \varepsilon f'(x_k) \approx 0$$

$$\varepsilon \approx -\frac{f(x_k)}{f'(x_k)} \quad , \quad r = x_k - \frac{f(x_k)}{f'(x_k)}$$

- However, the Newton-Raphson method has limitations in the following cases:
  - o Some functions may have its derivatives difficult to derive and may require lengthy steps,
  - o Some simple functions may have a small tangent gradient and thus need many iterations to converge, e.g. $f(x) = x^{10} - 1$ with $x_0 = 0.5$,
  - o Some cases may lead to divergence (see Fig. 2.6),

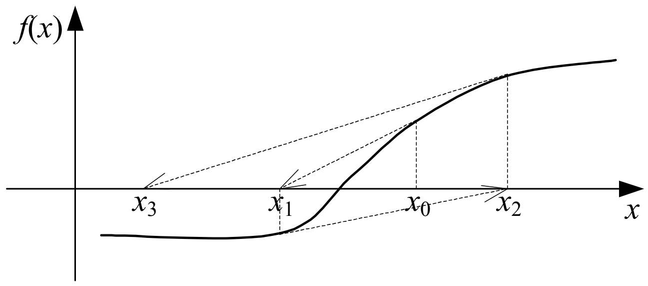

FIGURE 2.6 The Newton-Raphson iteration process which is diverged

  - o Oscillations may happen and this will not terminate (see Fig. 2.7),

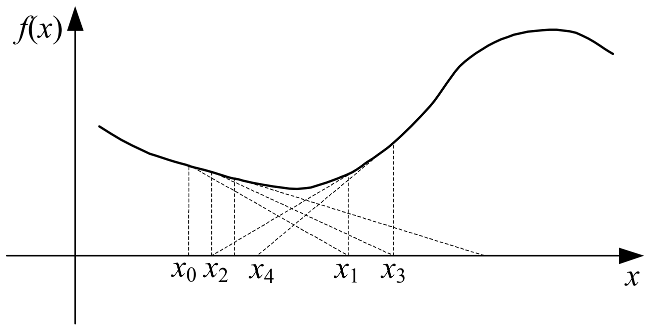

FIGURE 2.7 Oscillations in the Newton-Raphson method

  - o The method may be not accurate in the case of multiple roots (see Fig. 2.8).

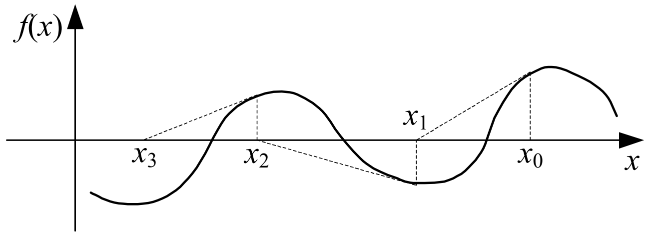

FIGURE 2.8 A case of multiple root using the Newton-Raphson method

Example 2.7

Use the Newton-Raphson method to determine the root for $e^{-x}-x$. Take the initial value of $x_0 = 0$.

Solution

$$f(x) = e^{-x} - x$$
$$f'(x) = -e^{-x} - 1$$

The Newton-Raphson formula:

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

$$x_{i+1} = x_i - \frac{e^{-x_i} - x_i}{-e^{-x_i} - 1}$$

i	x_i	$f(x_i)$	$f'(x_i)$	$f(x_i)/f'(x_i)$	$\lvert\varepsilon_t\rvert$%
0	0	1	−2	−0.5	100
1	0.50000000	0.10653066	−1.60653066	−0.06631100	11.839
2	0.56631100	0.00130451	−1.56761551	−0.00083216	0.147
3	0.56714317	1.965E−07	−1.56714336	−1.254E−07	2.205E−05
4	0.56714329	4.441E−15	−1.56714329	−2.834E−15	7.225E−04

- The **modified Newton-Raphson** method simplifies the formula such that the derivative has to be evaluated once only but leads to more iterations:

$$x_{i+1} = x_i - \frac{f(x_i)}{f'(x_0)} \tag{2.7}$$

Moreover, $f'(x_0)$ may even evaluated through a small change of Δx:

$$f'(x_0) \approx \frac{\Delta f(x)}{\Delta x}$$

Example 2.8

Repeat Example 2.7 using the modified Newton-Raphson method 2.7.

Solution

From Example 2.7:

$$f(x) = e^{-x} - x$$
$$f'(0) = -2$$

Hence, the formula for the modified Newton-Raphson is

$$x_{i+1} = x_i - \frac{e^{-x} - x}{-2}$$

i	x_i	$f(x_i)$	$f(x_i)/f'(x_i)$	$\lvert\varepsilon_t\rvert\%$
0	0	1	–0.5	100
1	0.5000000	0.1065307	–0.0532653	11.839
2	0.5532653	0.0218604	–0.0109018	2.4470
3	0.5641671	0.0046666	–0.0023332	0.5248
4	0.5665004	0.0010076	–0.0005038	0.1134
5	0.5670042	0.0002180	–0.0001090	0.02452
6	0.5671132	4.717E–05	–2.358E-05	5.307E–03
7	0.5671368	1.021E–05	–5.104E-06	1.149E–03
8	0.5671419	2.210E–06	–1.105E-06	2.486E–04
9	0.5671430	4.782E–07	–2.391E-07	5.380E–05
10	0.5671432	1.035E–07	–5.174E-08	1.164E–05
11	0.5671433	2.240E–08	–1.120E-08	2.520E–06
12	0.5671433	4.847E–09	–2.424E-09	5.454E–07

- The **secant** method can avoid using any the derivative and the gradient is taken from the formula of *finite divided difference*:

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

Hence the complete formula for the secant method is

$$x_{i+1} = x_i - \frac{f(x_i)\cdot(x_{i-1} - x_i)}{f(x_{i-1}) - f(x_i)} \tag{2.8}$$

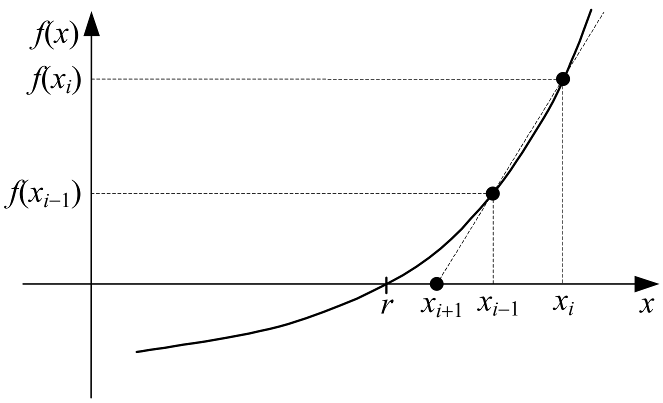

FIGURE 2.9 The secant method for searching roots

However, it needs two initial conditions, i.e. for x_i and x_{i-1}.

Example 2.9

Use the secant method to determine the root for $e^{-x} - x$. Take the initial value of $x_{-1} = 0$ and $x_0 = 1.0$.

Solution

i	x_i	$f(x_i)$	$\lvert\varepsilon_t\rvert$%
−1	0	1	100
0	1.0	−0.63212056	76.3
1	0.61269984	−0.07081395	8.03
2	0.56383839	0.00518236	0.564
3	0.56717036	−4.242E−05	0.00477
4	0.56714331	−2.538E−08	2.928E−06

- The following graph gives the comparison for the processes:

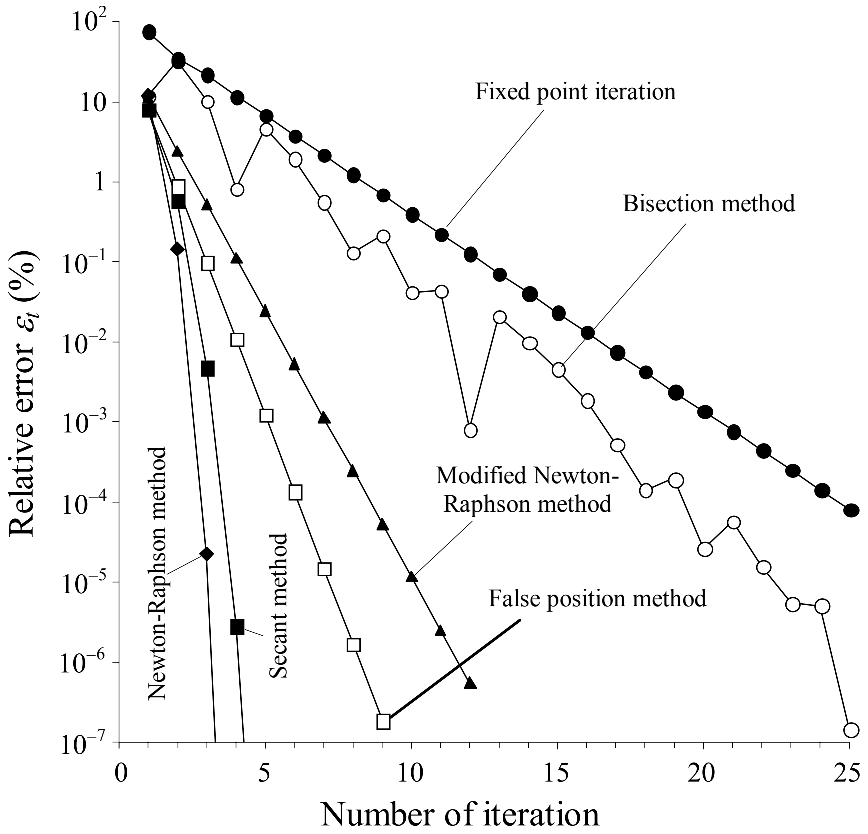

FIGURE 2.10 Comparison of root searching methods for $f(x) = e^{-x} - x$

2.5 Polynomial Roots

- The typical form of n-th order polynomial function of having n roots:

$$f_n(x) = a_0 + a_1 x + a_2 x^2 + \cdots + a_n x^n \tag{2.9}$$

- The **Müller** method uses the similar approach as the secant method, but requires *three* initial points (see Fig. 2.11).

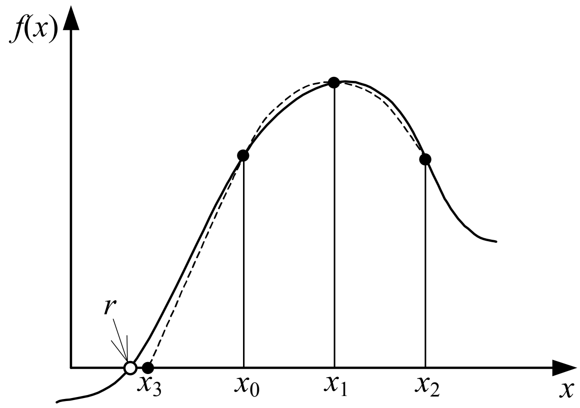

FIGURE 2.11 Parabolic projection in the Müller method

In this method, the quadratic polynomial has been used:

$$f_2(x) = a(x - x_2)^2 + b(x - x_2) + c \tag{2.10}$$

With the three initial points $[x_0, f(x_0)]$, $[x_1, f(x_1)]$ and $[x_2, f(x_2)]$:

$$\begin{aligned} f(x_0) &= a(x_0 - x_2)^2 + b(x_0 - x_2) + c \\ f(x_1) &= a(x_1 - x_2)^2 + b(x_1 - x_2) + c \\ f(x_2) &= a(x_2 - x_2)^2 + b(x_2 - x_2) + c \end{aligned}$$

By solving these simultaneous equations:

$$\begin{aligned} a &= \frac{\dfrac{f(x_2) - f(x_1)}{x_2 - x_1} - \dfrac{f(x_1) - f(x_0)}{x_1 - x_0}}{x_2 - x_0} \\ b &= a(x_2 - x_1) + \frac{f(x_2) - f(x_1)}{x_2 - x_1} \\ c &= f(x_2) \end{aligned} \tag{2.11}$$

By using the values of a, b and c, Eq. (2.11) can be reused to obtain x_3:

$$f(x_3) = a(x_3 - x_2)^2 + b(x_3 - x_2) + c = 0$$

$$x_3 = x_2 + \frac{-2c}{b \pm \sqrt{b^2 - 4ac}} \tag{2.12}$$

Example 2.10

Use the Müller method to get the roots for the following cubic polynomial:

$$f(x) = x^3 - 2.5x^2 + 1.5x - 1$$

Take $x_0 = 2.5$, $x_1 = 2.4$ dan $x_2 = 2.6$ as the initial values and perform iteration until the relative error is less than 0.05%.

Solution

In the first iteration:

The values of $f(x)$ for the three initial values are

$$f(x_0) = (2.5)^3 - 2.5(2.5)^2 + 1.5(2.5) - 1 = 2.75$$
$$f(x_1) = (2.4)^3 - 2.5(2.4)^2 + 1.5(2.4) - 1 = 2.024$$
$$f(x_2) = (2.6)^3 - 2.5(2.6)^2 + 1.5(2.6) - 1 = 3.576$$

From Eq. (2.11):

$$a = \frac{\dfrac{3.576 - 2.024}{2.6 - 2.4} - \dfrac{2.024 - 2.75}{2.4 - 2.5}}{5 - 4.5} = 5$$

$$b = 5(2.6 - 2.4) + \frac{3.576 - 2.024}{2.6 - 2.4} = 8.76$$

$$c = 3.576$$

Hence, using Eq. (2.12):

$$x_3 = 5 + \frac{-2(3.576)}{8.76 + \sqrt{(8.76)^2 - 4(5)(3.576)}} = 1.95242$$

The rest of the processes are as follows:

i	x_i	$f(x_i)$	a	b	c	ε_a (%)
0	2.5	2.75	-	-	-	-
1	2.4	2.024	-	-	-	-
2	2.6	3.576	5	8.76	3.576	-
3	1.95242	–0.15871	4.45242	2.88389	–0.15871	33.17
4	2.00344	0.01207	4.05586	3.55452	0.01207	2.546
5	2.00003	0.00010	3.45588	3.50036	0.00010	0.170
6	2.00000	0.00000	3.50346	3.50000	0.00000	0.001

- The **Bairstow** method combines both the Müller and Newton-Raphson methods and enables the determination of all roots, either real or complex.

Dividing Eq. (2.9) with a quadratic factor $x^2 - rx - s$ produces

$$f_{n-2}(x) = b_2 + b_3 x + \cdots + b_{n-1} x^{n-3} + b_n x^{n-2} \tag{2.13}$$

resulting in the residual term as followed

$$R = b_1(x - r) + b_0 \tag{2.14}$$

Hence, Eq. (2.9) can be rewritten as

$$f_n(x) = f_{n-2}(x) \cdot (x^2 - rx - s) + R$$

$$a_0 + a_1 x + a_2 x^2 + \cdots + a_n x^n = \\ (b_2 + b_3 x + \cdots + b_{n-1} x^{n-3} + b_n x^{n-2}) \cdot (x^2 - rx - s) + (b_1(x - r) + b_0)$$

Hence,

$$\begin{aligned} b_n &= a_n \\ b_{n-1} &= a_{n-1} + r b_n \\ b_i &= a_i + r b_{i+1} + s b_{i+2} \qquad \text{for } i = n-2, \cdots, 2, 1, 0 \end{aligned} \tag{2.15}$$

If r^* dan s^* is an approximation of r dan s, then Δr dan Δs are their respective changes, i.e.

$$r^* = r + \Delta r \qquad s^* = s + \Delta s$$

Since b_0 dan b_1 are functions of r dan s, the following equations can be established

$$b_1\left(r^*, s^*\right) = b_1 + \frac{\partial b_1}{\partial r}\Delta r + \frac{\partial b_1}{\partial s}\Delta s = 0$$

$$b_0\left(r^*, s^*\right) = b_0 + \frac{\partial b_0}{\partial r}\Delta r + \frac{\partial b_0}{\partial s}\Delta s = 0$$

$$\begin{aligned} \frac{\partial b_1}{\partial r}\Delta r + \frac{\partial b_1}{\partial s}\Delta s &= -b_1 \\ \frac{\partial b_0}{\partial r}\Delta r + \frac{\partial b_0}{\partial s}\Delta s &= -b_0 \end{aligned} \tag{2.16}$$

Then, dividing Eq. (2.13) with the same quadratic factor yields:

$$\begin{aligned} c_n &= b_n \\ c_{n-1} &= b_{n-1} + rc_n \\ c_i &= b_i + rc_{i+1} + sc_{i+2} \qquad \text{for } i = n-2, \cdots, 3,2,1 \end{aligned} \tag{2.17}$$

This produces

$$\frac{\partial b_1}{\partial r} = c_2 \qquad \frac{\partial b_1}{\partial s} = c_3$$
$$\frac{\partial b_0}{\partial r} = c_1 \qquad \frac{\partial b_0}{\partial s} = c_2$$

which can be solved as followed

$$\begin{aligned} c_2\,\Delta r + c_3\,\Delta s &= -b_1 \\ c_1\,\Delta r + c_2\,\Delta s &= -b_0 \end{aligned} \tag{2.18}$$

These equations are iterated until Δr dan Δs are within the convergence criteria, and then the root can be estimated as followed:

$$x = \frac{r \pm \sqrt{r^2 + 4s}}{2} \tag{2.19}$$

Example 2.11

Use the Bairstow method to obtain all roots of the following polynomials:

$$f(x) = x^3 - 2.5x^2 + 1.5x - 1$$

Take $r = s = 0$ as the initial values and perform iterations for the convergence criteria of 0.05%.

Solution

In the first iteration:

Use Eq. (2.15) and Eq. (2.17):

$$b_3 = a_3 = 1$$
$$b_2 = a_2 + rb_3 = (-2.5) + (0)(1) = -2.5$$
$$b_1 = a_1 + rb_2 + sb_3 = (1.5) + (0)(-2.5) + (0)(1) = 1.5$$
$$b_0 = a_0 + rb_1 + sb_2 = (-1) + (0)(1.5) + (0)(-2.5) = -1$$

$$c_3 = b_3 = 1$$
$$c_2 = b_2 + rc_3 = (-2.5) + (0)(1) = -2.5$$
$$c_1 = b_1 + rc_2 + sc_3 = (1.5) + (0)(-2.5) + (0)(1) = 1.5$$

Then, use Eq. (2.18):

$$\begin{aligned} -2.5\,\Delta r \quad + \qquad \Delta s &= -1.5 \\ 1.5\,\Delta r \quad - \quad 2.5\,\Delta s &= 1 \end{aligned}$$

$$\Delta r = 0.57895, \qquad \Delta s = -0.05263,$$
$$r = 0.57895. \qquad s = -0.05263.$$

The rest of the processes are as follows:

i	r	s	c_1	c_2	c_3	Δr	Δs	$\varepsilon_{a,r}$ (%)	$\varepsilon_{a,s}$ (%)
0	0	0	1.5	–2.5	1	0.5789	–0.0526	-	-
1	0.5789	–0.0526	–0.4945	–1.3421	1	–0.1111	–0.4843	19.2	920
2	0.4679	–0.5369	–1.2564	–1.5643	1	0.0313	0.0367	6.70	6.84
3	0.4992	–0.5002	–1.2488	–1.5016	1	0.0008	0.0002	0.156	0.037
4	0.5000	–0.5000	–1.2500	–1.5000	1	0.0000	0.0000	0.000	0.000

From this table, the quadratic factor is $x^2 - 0.5x + 0.5$. From Eq. (2.19):

$$x = \frac{0.5 \pm \sqrt{(0.5)^2 + 4(-0.5)}}{2},$$
$$= 0.25 \pm \sqrt{0.4375}\, i.$$

One more root can be obtained as followed (use the values of $b_2 = -2$ dan $b_3 = 1$):

$$x^3 - 2.5x^2 + 1.5x - 1 = \left(x^2 - 0.5x + 0.5\right) \cdot \left(b_2 + b_3 x\right),$$
$$= \left(x^2 - 0.5x + 0.5\right) \cdot \left(-2 + x\right)$$

Hence, the last root is

$$x = 2$$

2.6 System of Multivariable Equations

- Consider the following system of non-linear equations:

$$\begin{aligned} f_1(x_1, x_2, \ldots, x_n) &= 0 \\ f_2(x_1, x_2, \ldots, x_n) &= 0 \\ &\vdots \\ f_n(x_1, x_2, \ldots, x_n) &= 0 \end{aligned} \tag{2.20}$$

- This system can be solved using the Newton-Raphson method as followed (via the first order Taylor series):

$$\begin{aligned} f_1\left(x_{1_{i+1}}, x_{2_{i+1}}, \ldots, x_{n_{i+1}}\right) = {} & f_1\left(x_{1_i}, x_{2_i}, \ldots, x_{n_i}\right) + \Delta x_{1_i} \frac{\partial}{\partial x_1} f_1\left(x_{1_i}, x_{2_i}, \ldots, x_{n_i}\right) \\ & + \Delta x_{2_i} \frac{\partial}{\partial x_2} f_1\left(x_{1_i}, x_{2_i}, \ldots, x_{n_i}\right) + \cdots + \Delta x_{n_i} \frac{\partial}{\partial x_n} f_1\left(x_{1_i}, x_{2_i}, \ldots, x_{n_i}\right) \end{aligned}$$

$$\begin{aligned} f_2\left(x_{1_{i+1}}, x_{2_{i+1}}, \ldots, x_{n_{i+1}}\right) = {} & f_2\left(x_{1_i}, x_{2_i}, \ldots, x_{n_i}\right) + \Delta x_{1_i} \frac{\partial}{\partial x_1} f_2\left(x_{1_i}, x_{2_i}, \ldots, x_{n_i}\right) \\ & + \Delta x_{2_i} \frac{\partial}{\partial x_2} f_2\left(x_{1_i}, x_{2_i}, \ldots, x_{n_i}\right) + \cdots + \Delta x_{n_i} \frac{\partial}{\partial x_n} f_2\left(x_{1_i}, x_{2_i}, \ldots, x_{n_i}\right) \end{aligned}$$

$$\vdots$$

$$\begin{aligned} f_n\left(x_{1_{i+1}}, x_{2_{i+1}}, \ldots, x_{n_{i+1}}\right) = {} & f_n\left(x_{1_i}, x_{2_i}, \ldots, x_{n_i}\right) + \Delta x_{1_i} \frac{\partial}{\partial x_1} f_n\left(x_{1_i}, x_{2_i}, \ldots, x_{n_i}\right) \\ & + \Delta x_{2_i} \frac{\partial}{\partial x_2} f_n\left(x_{1_i}, x_{2_i}, \ldots, x_{n_i}\right) + \cdots + \Delta x_{n_i} \frac{\partial}{\partial x_n} f_n\left(x_{1_i}, x_{2_i}, \ldots, x_{n_i}\right) \end{aligned}$$

where,

$$\begin{aligned} \Delta x_{1_i} &= x_{1_{i+1}} - x_{1_i} \\ \Delta x_{2_i} &= x_{2_{i+1}} - x_{2_i} \\ &\vdots \\ \Delta x_{n_i} &= x_{n_{i+1}} - x_{n_i} \end{aligned}$$

For $f_1\left(x_{1_{i+1}}, x_{2_{i+1}}, \ldots, x_{n_{i+1}}\right) = f_2\left(x_{1_{i+1}}, x_{2_{i+1}}, \ldots, x_{n_{i+1}}\right) = \cdots = f_n\left(x_{1_{i+1}}, x_{2_{i+1}}, \ldots, x_{n_{i+1}}\right) = 0$, the above relation can be rearranged into a matrix equation:

$$\begin{bmatrix} \partial f_1/\partial x_1 & \partial f_1/\partial x_2 & \cdots & \partial f_1/\partial x_n \\ \partial f_2/\partial x_1 & \partial f_2/\partial x_2 & \cdots & \partial f_2/\partial x_n \\ \vdots & \vdots & \ddots & \vdots \\ \partial f_n/\partial x_1 & \partial f_n/\partial x_2 & \cdots & \partial f_n/\partial x_n \end{bmatrix}_i \cdot \begin{Bmatrix} \Delta x_{1_i} \\ \Delta x_{2_i} \\ \vdots \\ \Delta x_{n_i} \end{Bmatrix} = -\begin{Bmatrix} f_1\left(x_{1_i}, x_{2_i}, \ldots, x_{n_i}\right) \\ f_2\left(x_{1_i}, x_{2_i}, \ldots, x_{n_i}\right) \\ \vdots \\ f_n\left(x_{1_i}, x_{2_i}, \ldots, x_{n_i}\right) \end{Bmatrix} \quad (2.21)$$

or in a more compact form,

$$\mathbf{J}_i \cdot \left(\mathbf{x}_{i+1} - \mathbf{x}_i\right) = -\mathbf{f}_i$$

where the left-hand side matrix **J** is known as the *Jacobian* matrix. Eq. (2.21) can be iterated until converged.

Example 2.12

Use the Newton-Raphson method to solve the following system:

$$x^2 + xy = 3$$
$$y + 2xy^2 = 22$$

Take $x_0 = 1$ dan $y_0 = -1$ as the initial values and perform iterations for the until the relative error norm is less than 0.05%.

Solution

For the given system:

$$f_1(x, y) = 0 = x^2 + xy - 3$$
$$f_2(x, y) = 0 = y + 2xy^2 - 22$$

Then, the Jacobian matrix **J** can be formed as followed:

$$\mathbf{J} = \begin{bmatrix} \partial f_1/\partial x & \partial f_1/\partial y \\ \partial f_2/\partial x & \partial f_2/\partial y \end{bmatrix} = \begin{bmatrix} 2x + y & x \\ 2y^2 & 1 + 4xy \end{bmatrix}$$

and the iteration formulation is as followed:

$$\begin{bmatrix} 2x_i + y_i & x_i \\ 2y_i^2 & 1 + 4x_i y_i \end{bmatrix} \cdot \begin{Bmatrix} x_{i+1} - x_i \\ y_{i+1} - y_i \end{Bmatrix} = \begin{Bmatrix} -f_1(x_i, y_i) \\ -f_2(x_i, y_i) \end{Bmatrix}$$

By using the initial values of $x_0 = 1$ dan $y_0 = -1$:

i	x_i	y_i	$f_1(x_i, y_i)$	$f_2(x_i, y_i)$	Δx_i	Δy_i	$\|\|\varepsilon_a\|\|_e$ (%)
0	1	−1	−3	−21	6	−3	83.21
1	7	−4	18	198	−2.53673	1.05247	51.35
2	4.46327	−2.94753	3.76516	52.6054	−1.11122	0.64501	31.59
3	3.35204	−2.30252	0.51807	11.2397	−0.31822	0.26330	11.30
4	3.03382	−2.03921	0.01748	1.19237	−0.03336	0.03853	1.413
5	3.00047	−2.00068	−0.00017	0.01939	−0.00047	0.00068	0.023
6	3	−2	0.00000	0.00000	0.00000	0.00000	0.000

Exercises

1. Determine the intersection of the two following equations:

$$g(s) = s^6$$
$$h(s) = s + 1$$

using the false position method and then the Newton-Raphson method. By using the range of [1, 1.5], perform calculation until it converges. Also, at each iteration, calculate the approximate and actual errors if the actual solution is $s = 1.13472$.

2. The relationship for friction factor f for a flow in a damping element with the Reynolds number R_e is given by:

$$\frac{1}{\sqrt{f}} = \frac{1}{k}\ln\left(R_e\sqrt{f}\right) + \left(14 - \frac{5.6}{k}\right)$$

where k is a constant for internal wall roughness for the damping element and is equal to 0.28. Calculate the value of f if $R_e = 3{,}750$.

3. Solve the following system of non-linear equations:

$$xyz - x^2 + y^2 = 1.34$$
$$xy - z^2 = 0.09$$
$$e^x - e^y + z = 0.41$$

Use the initial values of $(x_0, y_0, z_0) = (0, -1, 0)$ and the termination criateria of 0.1% for the approximate error norm.