Sources for presentations:

- Textbook
- [http://math.fullerton.edu/mathews/numerical.html](http://math.fullerton.edu/mathews/numerical.html)
- [http://numericalmethods.eng.usf.edu](http://numericalmethods.eng.usf.edu)
- MIT Open Courseware “Introduction to Numerical Analysis for Engineering”

The Solution of a Nonlinear Equation

\[ f(x) = 0 \]

Example – Square root

\[ x^2 - a = 0 \Rightarrow x = \sqrt{a} \]

Example:

Figure 2.1 The portion of a sphere of radius \( r \) that is to be submerged to a depth \( d \).

\[ \frac{\pi}{3} (r^3 - 3y^2r + 4y^3) = 0. \]

Figure 2.2 The curve \( y = 2555 - 2965 \cdot e^{-x} \).
General procedure for solving nonlinear equations/root finding/finding zero

- Plot the function
- Make an initial guess
- Iteratively refine the initial guess with a root-finding algorithm

Iteration: a process is repeated until an answer is achieved.

Bisection Method

An equation $f(x)=0$, where $f(x)$ is a real continuous function, has at least one root between $x_l$ and $x_u$ if $f(x_l) f(x_u) < 0$. 

![Bisection Method Diagram]
Step 1

• Choose $x_l$ and $x_u$ as two guesses for the root such that $f(x_l) f(x_u) < 0$, or in other words, $f(x)$ changes sign between $x_l$ and $x_u$.

Step 2

Estimate the root, $x_m$ of the equation $f(x) = 0$ as the mid-point between $x_l$ and $x_u$ as

$$x_m = \frac{x_l + x_u}{2}$$
Step 3

Now check the following

If \( f(x_l) f(x_m) < 0 \), then the root lies between \( x_l \) and \( x_m \); then
\[
x_l = x_l; \quad x_u = x_m.
\]

If \( f(x_l) f(x_m) > 0 \), then the root lies between \( x_m \) and \( x_u \); then
\[
x_l = x_m; \quad x_u = x_u.
\]

If \( f(x_l) f(x_m) = 0 \); then the root is \( x_m \).
Stop the algorithm if this is true.

Iteration continues until stopping criteria are met.

Stopping criteria:

- \( |x_k - x_{k-1}| < \delta \) Machine Accuracy
- \( |f(x) - f(x_k)| < \delta \) Accuracy

Use combination of the two criteria

Cannot require
\[
|x_k - x_{k-1}| < \delta
\]

Cannot require
\[
\delta |f(x) - f(x_k)| < \delta
\]
Example 1. Find all the real solutions to the cubic equation $x^3 + 4x^2 - 10 = 0$.

There appears to be only one real root which lies in the interval $[1,2]$.

The Bisection Method
Approximate a solution of the equation $f(x) = 0$.

$y = f(x) = x^3 + 4x^2 - 10$
Use the starting interval \([a, b] = [-1, 2]\):

<table>
<thead>
<tr>
<th>(k)</th>
<th>(a_k)</th>
<th>(c_k)</th>
<th>(b_k)</th>
<th>(f(c_k))</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>-1.0</td>
<td>0.5</td>
<td>2.0</td>
<td>-6.975</td>
</tr>
<tr>
<td>1</td>
<td>0.5</td>
<td>1.25</td>
<td>2.0</td>
<td>-1.79875</td>
</tr>
<tr>
<td>2</td>
<td>1.25</td>
<td>1.625</td>
<td>2.0</td>
<td>4.853515625</td>
</tr>
<tr>
<td>3</td>
<td>1.25</td>
<td>1.4375</td>
<td>1.625</td>
<td>1.326083984375</td>
</tr>
<tr>
<td>4</td>
<td>1.25</td>
<td>1.34375</td>
<td>1.4375</td>
<td>-0.350983866815625</td>
</tr>
<tr>
<td>5</td>
<td>1.34375</td>
<td>1.390625</td>
<td>1.4375</td>
<td>0.4265945791930306</td>
</tr>
<tr>
<td>6</td>
<td>1.34375</td>
<td>1.3671875</td>
<td>1.390625</td>
<td>0.03233578536397308</td>
</tr>
<tr>
<td>7</td>
<td>1.34375</td>
<td>1.355357142857</td>
<td>1.3671875</td>
<td>-0.109421113660286</td>
</tr>
<tr>
<td>8</td>
<td>1.355357142857</td>
<td>1.361328125</td>
<td>1.3671875</td>
<td>-0.0643102452158279</td>
</tr>
<tr>
<td>9</td>
<td>1.361328125</td>
<td>1.3642578125</td>
<td>1.3671875</td>
<td>-0.01504669075453242</td>
</tr>
<tr>
<td>10</td>
<td>1.3642578125</td>
<td>1.36572265625</td>
<td>1.3671875</td>
<td>0.00813717257910581</td>
</tr>
</tbody>
</table>

| \(c\) | 1.36523001328506 |
| \(\Delta c\) | \(-1.39698 \times 10^{-8}\) |
| \(f[0]\) | \(-1.10300845439774 \times 10^{-6}\) |

**Theorem 2.4 (Bisection Theorem).** Assume that \(f \in C[a, b]\) and that there exists a number \(r \in [a, b]\) such that \(f(r) = 0\). If \(f(a)\) and \(f(b)\) have opposite signs, and \(\{c_n\}_{n=0}^{\infty}\) represents the sequence of midpoints generated by the bisection process of (8) and (9), then

\[
|r - c_n| \leq \frac{b - a}{2^{n+1}} \quad \text{for } n = 0, 1, \ldots.
\]

\[
\begin{array}{c}
\text{midpoint} \\
\text{interval} \\
\text{midpoint} \\
\text{interval}
\end{array}
\]

Observe that the successive interval widths form the pattern

\[
\begin{align*}
b_1 - a_1 &= \frac{b_0 - a_0}{2^1}, \\
b_2 - a_2 &= \frac{b_1 - a_1}{2} = \frac{b_0 - a_0}{2^2}.
\end{align*}
\]

It is left as an exercise for the reader to use mathematical induction and show that

\[
b_n - a_n = \frac{b_0 - a_0}{2^n}.
\]
A virtue of the bisection method is that formula (10) provides a predetermined estimate for the accuracy of the computed solution. In Example 2.7 the width of the starting interval was \( b_0 - a_0 = 2 \). Suppose that Table 2.1 were continued to the thirty-first iterate; then, by (10), the error bound would be \( |E_{31}| \leq (2 - 0)/2^{32} \approx 4.656613 \times 10^{-10} \). Hence \( c_{31} \) would be an approximation to \( c \) with nine decimal places of accuracy. The number \( N \) of repeated bisections needed to guarantee that the \( N \)th midpoint \( c_N \) is an approximation to a zero and has an error less than the preassigned value \( \delta \) is

\[
N = \text{int} \left( \frac{\ln(b - a) - \ln(\delta)}{\ln(2)} \right),
\]

where \( |r - c_N| = \delta \).

Ex. If we want to reduce the error to less than 0.1% of the original interval we need \( N = 9 \) iterations.

---

**Convergence**

- Solution to \( f(x) = 0 \) involves a series of approximations.

If \( x_0, x_1, \ldots, x_n \to x^* \) ; \( n \to \infty \)

then the numerical method *converges* (is convergent)

Otherwise diverges (is divergent)

- We are interested in
  - Conditions of convergence
  - Speed of convergence

Ex. Bisection method is always convergent.
Order of Convergence

\[ \lim_{n \to \infty} \frac{|p - p_{n+1}|}{|p - p_n|^R} = \lim_{n \to \infty} \frac{|E_{n+1}|}{|E_n|^R} = A \]

R : order of convergence (R>0)
A : asymptotic error constant (A≠0)
p : root
E: error

R=1 linear convergence
R=2 quadratic convergence

Ex. bisection method
has a linear convergenc(R=1,A=0.5):

\[ e_n = \frac{b_n - a_n}{2}, e_{n+1} = \frac{b_{n+1} - a_{n+1}}{2} = \frac{b_n - a_n}{4} \]

\[ \frac{e_{n+1}}{e_n} = \frac{1}{2} \]

Matlab Code for bisection method

• function [c,err,yc]=bisect(f,a,b,delta)

  %Input - f is the function input as a string 'f'
  % - a and b are the left and right endpoints
  % - delta is the tolerance
  %Output - c is the zero
  %- yc = f(c)
  % - err is the error estimate for c

  ya=feval(f,a);
  yb=feval(f,b);
  if ya*yb > 0,break,end
  max1=1+round((log(b-a)-log(delta))/log(2));
  for k=1:max1
    c=(a+b)/2;
    yc=feval(f,c);
    if yc==0
      a=c;
      b=c;
    elseif yb*yc>0
      b=c;
      yb=yc;
    else
      a=c;
      ya=yc;
    end
    if b-a < delta, break,end
  end
  c=(a+b)/2;
  err=abs(b-a);
  yc=feval(f,c);
Advantages

• Always convergent
• The root bracket gets halved with each iteration - guaranteed.

Drawbacks

• Slow convergence
• If one of the initial guesses is close to the root, the convergence is slower

Drawbacks (continued)

• If a function $f(x)$ is such that it just touches the x-axis it will be unable to find the lower and upper guesses.
Drawbacks (continued)

- Function changes sign but root does not exist

\[ f(x) = \frac{1}{x} \]

Method of False Position (Regula Falsi)

- Start with an initial interval **bracketing** the root
- Calculate \( c (\ast) \)
  - If \( f(a) \) and \( f(c) \) have opposite signs, a zero lies in \([a, c]\).
  - If \( f(c) \) and \( f(b) \) have opposite signs, a zero lies in \([c, b]\).
  - If \( f(c) = 0 \), then the zero is \( c \).
- Continue iteration until stopping criteria are satisfied
For the same stopping criteria False position converges in 15 steps (bisection 30 steps)
False position’s convergence is faster than linear

c = 1.365239619798665
r(c) = -4.368872977233973 × 10⁻¹

The Regula Falsi Method
Approximate a solution of the equation f(x) = 0

Drawback

Stationary end point for the false position method
Fixed Point Iteration

– Rewrite \( f(x) = 0 \) as \( x=g(x) \) so that finding the root of \( f(x) = 0 \) becomes equivalent to finding the fixed point of \( g(x) \).
– Start with initial guess \( p_0 \)
– Iterate according to

\[
\begin{align*}
  p_1 &= g(p_0) \\
  p_2 &= g(p_1) \\
  & \\
  p_K &= g(p_{K-1}) \\
  p_{K+1} &= g(p_K)
\end{align*}
\]

Note 1: no initial interval – open method
Note 2: Check if the number of iterations has exceeded the maximum number of iterations (additional stopping criteria)

Convergence conditions for the fixed-point iterations:

If

1. \( g(x) \) is continuous and \( g(x) \) maps \([a,b]\) into \([a,b]\),
2. \( g'(x) \) is continuous on \([a,b]\), and
3. there is a number \( K<1 \) such that \( |g'(x)| \leq K \) for all \( x \) in \([a,b]\),

then

• \( x=g(x) \) has exactly one solution (say \( x^* \)) in \([a,b]\), and
• the fixed-point iteration converges to \( x^* \), for any initial guess in \([a,b]\).

Note 1: if 1 and 2 hold, but \( |g'(x)| > 1 \) then fixed-point iteration diverges
Note 2: since \( g(x) \) is continuous in \([a,b]\) we can also use \( |g'(x^*)| \leq K<1 \) and \( |g'(x^*)| > 1 \).
Example 2.4. Consider the iteration \( p_{n+1} = g(p_n) \) when the function \( g(x) = 1 + x - x^2/4 \) is used. The fixed points can be found by solving the equation \( x = g(x) \). The two solutions (fixed points of \( g \)) are \( x = -2 \) and \( x = 2 \). The derivative of the function is \( g'(x) = 1 - x/2 \), and there are only two cases to consider.

**Case (i):** \( P = -2 \)
- Start with \( p_0 = -2.05 \)
- then get
  - \( p_1 = -2.100625 \)
  - \( p_2 = -2.20378155 \)
  - \( p_3 = -2.41794441 \)
  - \( \ldots \)
  - \( \lim_{n \to \infty} p_n = -\infty \)

Since \( |g'(x)| > \frac{1}{2} \) on \([-3, -1]\), by Theorem 2.3, the sequence will not converge to \( P = -2 \).

**Case (ii):** \( P = 2 \)
- Start with \( p_0 = 1.6 \)
- then get
  - \( p_1 = 1.96 \)
  - \( p_2 = 1.9996 \)
  - \( p_3 = 1.9999996 \)
  - \( \ldots \)
  - \( \lim_{n \to \infty} p_n = 2 \)

Since \( |g'(x)| < \frac{1}{2} \) on \([1, 3]\), by Theorem 2.3, the sequence will converge to \( P = 2 \).
Convergence: Monotone Increasing

Fixed Point Iteration
Approximate a solution of the equation $x = g(x)$

$y = g(x) = \sqrt{2 + \sqrt{x}}$
Divergence: Monotone Increasing

Fixed Point Iteration
Approximate a solution
of the equation $x = g(x)$

Newton-Raphson Method

$$\tan(\alpha) = \frac{AB}{AC}$$

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

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

The Newton-Raphson Method
Approximate a solution
of the equation $f(x) = 0$

$$y = f(x) = 3e^x - 4\cos(x)$$

Cycling divergence

The Newton-Raphson Method
Approximate a solution
of the equation $f(x) = 0$

$$y = f(x) = x^3 - x + 1$$
Convergence rate of Newton-Raphson

- quadratic at a simple root
  \[ |E_{k+1}| = \frac{1}{2} \left| \frac{f''(p)}{f'(p)} \right| \left( |E_k| \right)^2 \]

- linear at a multiple root
  \[ |E_{k+1}| = \frac{m-1}{m} |E_k| \]

Definition: \( f(x) \) has a root of order \( m \) at \( x=p \) if

\[ f(p) = 0, \quad f'(p) = 0, \quad f''(p) = 0, \quad \ldots, \quad f^{(n-1)}(p) = 0, \quad f^{(n)}(p) \neq 0 \]
Advantages

- Converges fast, if it converges
- Requires only one guess

Drawbacks

\[ f(x) = (x - 1)^3 = 0 \]

Inflection Point
Drawbacks (continued)

- Division by zero

\[ f(x) = x^3 - 0.03x^2 + 2.4 \times 10^{-6} = 0 \]

- Root Jumping

\[ f(x) = \sin x = 0 \]
Drawbacks (continued)

Oscillations near Local Maxima or Minima

Secant Method

Newton’s Method

\[ x_{j+1} = x_j - \frac{f(x_j)}{f'(x_j)} \]

Approximate the derivative

\[ f'(x_j) = \frac{f(x_j) - f(x_{j-1})}{x_j - x_{j-1}} \]

\[ x_{j+1} = x_j - \frac{f(x_j)(x_j - x_{j-1})}{f(x_j) - f(x_{j-1})} \]
Secant Method

Similar Triangles

\[
\frac{AB}{AE} = \frac{DC}{DE}
\]

\[
\frac{f(x_i)}{x_i - x_{i+1}} = \frac{f(x_{i-1})}{x_{i-1} - x_{i+1}}
\]

\[
x_{i+1} = x_i - \frac{f(x_i)(x_i - x_{i-1})}{f(x_i) - f(x_{i-1})}
\]

Oscillating convergence

The Secant Method
Approximate a solution of the equation \( f(x) = 0 \)
Divergence to infinity

Advantages

- Converges fast, if it converges \((R=1.618)\)
- Requires two guesses that do not need to bracket the root
Drawbacks

Division by zero