Newton-Raphson Method for Finding Roots of \( f(x) = 0 \)

The Newton-Raphson method uses the slope (tangent) of the function \( f(x) \) at the current iterative solution \( (x_i) \) to find the solution \( (x_{i+1}) \) in the next iteration (see Figure 1). This is different from the Bisection method which uses the sign change to locate the root.

![Figure 1: Newton-Raphson method](image)

The slope at \( (x_i, f(x_i)) \) is given by

\[
f'(x_i) = \frac{f(x_i) - 0}{x_i - x_{i+1}}
\]

which can be solved to find \( x_{i+1} \) as

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

This is known as the Newton-Raphson formula. Using this, the iterative solution is updated at each point.

Example: Find the root of the equation \( e^{-x} - 5x = 0 \).

Then,

\[
f(x) = e^{-x} - 5x
\]

\[
f'(x) = -e^{-x} - 5
\]

With these, the Newton-Raphson solution can be updated as

\[
x_{i+1} = x_i - \frac{e^{-x} - 5x}{-e^{-x} - 5} = x_i + \frac{e^{-x} - 5x}{e^{-x} + 5}
\]

**Termination condition**

The Newton-Raphson iteration can be terminated when the approximate relative error \( \epsilon_a \) is less than a certain threshold (say, 1%). The relative error is given by

\[
\epsilon_a = \frac{x_{i+1} - x_i}{x_{i+1}} \times 100\%
\]
This is similar to the termination condition used in the Bisection method.

**Newton-Raphson formulation from Taylor series**

The Newton-Raphson formula can be derived from the Taylor series expansion as well. This is useful for error analysis (since we know how to evaluate the approximation error in a Taylor series).

In the Taylor series approach, the updated value of the function $f(x_{i+1})$ can be written as

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

Since the function value at $x_{i+1}$ (at the intersection with the x axis) is zero, we can write the above equation as

$$f(x_i) + f'(x_i)(x_{i+1} - x_i) = 0$$

which simplifies as

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

As seen above, the Newton-Raphson formula can be written (in the form of a Taylor series) as

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

However, according to Taylor’s Theorem, the Taylor series can be written as

$$f(x_t) = f(x_i) + f'(x_i)(x_t - x_i) + \frac{f''(\alpha)}{2}(x_t - x_i)^2$$

where $x_t$ is the true solution to $f(x) = 0$ and $\alpha$ is an unknown value between $x_t$ and $x_i$. Note that $f(x_t) = 0$. Subtracting one equation from the other, we have

$$f'(x_i)(x_t - x_{i+1}) + \frac{f''(\alpha)}{2}(x_t - x_i)^2$$

Denoting $e_i = x_t - x_i$, which is the error in the $i$-th iteration, we have

$$0 = f'(x_i)e_{i+1} + \frac{f''(\alpha)}{2}e_i^2$$

With convergence, $x_i \longrightarrow x_t$ and $\alpha \longrightarrow x_t$. Then

$$e_{i+1} = -\frac{f''(x_t)}{2f'(x_t)}e_i^2$$

The above relationship between the errors in two successive iterations indicates that the error in the current iteration is proportional to the square of the previous error. That is, we have quadratic convergence with the Newton-Raphson method. The result is that the number of correct decimal places in a Newton-Raphson solution doubles after each iteration.

In spite of the above property, the Newton-Raphson methods has some drawbacks.
1. It cannot handle multiple roots.

2. It has slow convergence (compared with newer techniques).

3. The solution may diverge near a point of inflection (see Figure 2 or Figure 6.6 in Chapra and Canale).

4. The solution might oscillate near local minima or maxima.

5. With near-zero slope, the solution may diverge or reach a different root.

---

**Example:** Find the McLaurin series up to order 4, Taylor series (around $x = 1$) up to order 4 and the roots of function $f(x) = x^3 - 2x^2 + 0.25x + 0.75$.

- $f(x) = x^3 - 2x^2 + 0.25x + 0.75$
- $f'(x) = 3x^2 - 4x + 0.25$
- $f''(x) = 6x - 4$
- $f'''(x) = 6$

<table>
<thead>
<tr>
<th>$f(x)$</th>
<th>$f(0) = 0.75$</th>
<th>$f(1) = 0$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$f'(x)$</td>
<td>$f'(0) = 0.25$</td>
<td>$f'(1) = -0.75$</td>
</tr>
<tr>
<td>$f''(x)$</td>
<td>$f''(0) = -4$</td>
<td>$f''(1) = 2$</td>
</tr>
<tr>
<td>$f'''(x)$</td>
<td>$f'''(0) = 6$</td>
<td>$f'''(1) = 6$</td>
</tr>
</tbody>
</table>

Figure 2: Problems in the Newton-Raphson method
McLaurin series can be written as

\[ f(x) \approx \sum_{i=0}^{n} \frac{f^{(i)}(0)}{i!} x^i \]

\[ = f(0) + f'(0)x + \frac{f''(0)}{2} x^2 + \frac{f'''(0)}{6} x^3 \]

The fourth order term is ignored since \( f^{(iv)}(0) = 0 \). Then, the third order McLaurin series expansion

\[ f_{M3}(x) = 0.75 + 0.25x - 2x^2 + x^3 \]

which is the same as the original polynomial function. The lower order expansions may be written as

\[ f_{M2}(x) = 0.75 + 0.25x - 2x^2 \]
\[ f_{M1}(x) = 0.75 + 0.25x \]
\[ f_{M0}(x) = 0.75 \]

The Taylor series (of order 3) can be written as

\[ f_{T3}(x) \approx \sum_{i=0}^{n} \frac{f^{(i)}(r)}{i!} (x - r)^i \]

\[ = f(r) + f'(r)(x - r) + \frac{f''(r)}{2} (x - r)^2 + \frac{f'''(r)}{6} (x - r)^3 \]

where \( r = 1 \).

Then

\[ f_{T3}(x) = f(1) + f'(r)(x - 1) + \frac{f''(1)}{2}(x - 1)^2 + \frac{f'''(1)}{6}(x - 1)^3 \]

\[ = f(1) + f'(r)(x - 1) + \frac{f''(1)}{2}(x^2 - 2x + 1) + \frac{f'''(1)}{6}(x^3 - 3x^2 + 3x - 1) \]

\[ = \left[ f(1) - f'(1) + \frac{f''(1)}{2} - \frac{f'''(1)}{6} \right] + \left[ f'(1) - f''(1) + \frac{f'''(1)}{2} \right] x + \left[ \frac{f''(1)}{2} - \frac{f'''(1)}{2} \right] x^2 + \frac{f'''(1)}{2} x^3 \]

The third order Taylor series can be written as

\[ f_{T3}(x) = 0.75 + .25x - 2x^3 + x^3 \]

which is the same as the original function.

The lower order Taylor series have to be derived from first principles. In contrast, lower order McLaurin series can be found from higher order ones by truncating the higher order terms.

The second order Taylor series can be written as

\[ f_{T2}(x) = f(1) + f'(r)(x - 1) + \frac{f''(1)}{2}(x - 1)^2 \]

\[ = f(1) + f'(r)(x - 1) + \frac{f''(1)}{2}(x^2 - 2x + 1) \]

\[ = \left[ f(1) - f'(1) + \frac{f''(1)}{2} \right] + \left[ f'(1) - f''(1) \right] x + \frac{f''(1)}{2} x^2 \]

\[ = 1.75 - 2.75x + x^2 \]
The first order Taylor series can be written as

$$f_{T1}(x) = f(1) + f'(r)(x - 1)$$

$$= f(1) + f'(r)(x - 1)$$

$$= [f(1) - f'(1)] + f'(1)x$$

$$= 0.75 - 0.75x$$

The zeroth order Taylor series can be written as

$$f_{T0}(x) = f(1)$$

$$= 0$$

which indicates that $x = 1$ is a root of $f(x)$.

First, we’ll find the roots exactly by factorizing $f(x)$ (using the fact that $x = 1$ is a root).

To find the roots exactly,

$$f(x) = 0$$

$$x^3 - 2x^2 + 0.25x + 0.75 = 0$$

$$(x - 1)(x^2 - x - 0.75) = 0$$

$$(x - 1)(x - 1.5)(x + 0.5) = 0$$

Thus, $x = -0.5$, $x = 1$ and $x = 1.5$ are the exact roots of $f(x) = 0$.

To find the roots using the Newton-Raphson method, write the N-R formula as

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

$$= x_i - \frac{x^3 - 2x^2 + 0.25x + 0.75}{3x^2 - 4x + 0.25}$$

which can be programmed in MATLAB or C.

In the figures note the convergence properties of different algorithms depending on the starting points.

Sample MATLAB source codes are given below, but note that you cannot use the advanced features of MATLAB in your solution. In addition, the solutions given below can be improved significantly. Use this as a starting point for your program.

```matlab
%----------------------------------------
% McLaurin series for x^3-2x^2+0.25x+0.75

% find McLaurin series
x=-2:.05:2;
x=x(:);
```
m0 = .75*ones(length(x),1);
m1 = m0 + .25*x;
m2 = m1 - 2*x.^2;
m3 = m2 + x.^3;

% plot McLaurin series
f1 = figure;
plot(x,m0, x,m1, ':', x,m2, '--', x,m3, ':-:');
l = legend('Zero Order', 'First Order', 'Second Order', 'Third Order');
set(l, 'FontSize', 16);
set(gca, 'FontSize', 16);
xlabel('x', 'FontSize', 16);
ylabel('f(x)', 'FontSize', 16);
title('McLaurin Series', 'FontSize', 16)
Figure 5: Roots of $f(x)$ from Newton-Raphson method

Figure 6: Roots of $f(x)$ from Newton-Raphson method

% find Taylor series
t0=0*ones(length(x),1);
t1=.75*ones(length(x),1)-.75*x;
t2=1.75*ones(length(x),1)-2.75*x+x.^2;
t3=.75*ones(length(x),1)+.25*x-2*x.^2+x.^3;

% plot Taylor series
f2=figure;
plot(x,t0, x,t1,':', x,t2, '-.', x,t3, '--')
l=legend('Zero Order','First Order','Second Order','Third Order');
set(l,'FontSize', 16);
set(gca, 'FontSize',16);
xlabel('x', 'Fontsize',16);
ylabel('f(x)', 'Fontsize',16);
title('Taylor Series', 'Fontsize',16)
%----------------------------------------
% Newton-Raphson method solution for \(x^3-2x^2+0.25x+0.75=0\)

% form x and f(x)
x=-5:.05:5;
x=x(:);
t3=.75*ones(length(x),1)+.25*x-2*x.^2+x.^3;
xn=-5;
xo=10;

% final error criterion
e=.0001;

% plot the function
f2=figure;
fx=xn^3-2*xn^2+.25*xn+.75;
plot(x,t3, '--', xn, fx, 's')
set(gca, 'FontSize',16);
xlabel('x', 'Fontsize',16);
ylabel('f(x)', 'Fontsize',16);
set(gca, 'XTick', -5:.5:5);
title(['Newton-Raphson Method (from ', num2str(xn), ')'], 'Fontsize',16)
grid on
hold on

% do the iteration until convergence
while abs((xn-xo)/xn) > e
    fx=xn^3-2*xn^2+.25*xn+.75;
    fpx=3*xn^2-4*xn+.25;
    xn=xn-(fx)/(fpx);
    plot(xn, fx, 's');
    pause
end

%----------------------------------------
% Bisection method solution for x^3-2x^2+0.25x+0.75

% form x and f(x)
x=-5:.05:5;
x=x(:);
t3=.75*ones(length(x),1)+.25*x-2*x.^2+x.^3;

% initial range
xo=10;
e=.0001;
xl=-4.5;
xu=5;
xn=(xl+xu)/2;

% plot f(x) and current solution
f2=figure;
fx=xn^3-2*xn^2+.25*xn+.75;
fxl=xl^3-2*xl^2+.25*xl+.75;
fxu=xu^3-2*xu^2+.25*xu+.75;
plot(x,t3, '--', xn, fx, 's', xl, fxl, '<', xu, fxu, '>')
set(gca, 'FontSize',16);
xlabel('x', 'FontSize',16);
ylabel('f(x)', 'FontSize',16);
set(gca, 'XTick', -5:.5:5);
title(['Bisection Method (between ', num2str(xl), ' and ', num2str(xu), ')'], 'FontSize',16)
grid on
hold on

if (fxl*fx <=0)
    xu=xn;
elseif (fxu*fx <=0)
    xl=xn;
else
    return;
end

% iteratively improve the range and find the solution
while abs((xn-xo)/xn) > e
    fxl=xl^3-2*xl^2+.25*xl+.75;
    fxu=xu^3-2*xu^2+.25*xu+.75;
    xo=xn
    xn=(xl+xu)/2
    fx=xn^3-2*xn^2+.25*xn+.75;

    if (fxl*fx <=0)
        xu=xn;
    elseif (fxu*fx <=0)
        xl=xn;
    else
        break
    end
plot(xn, fx, 's', xl, fxl, '<', xu, fxu, '>');
end

Resources:
