1 Experiment No. 3

2 Experiment Title

Solving Nonlinear Equations by the False Position Method Using MATLAB

3 Objective

The objectives of this lab are:

- To determine two approximations between which the root of the equation exists.
- To display the approximations, root and error for each iteration by the false position method in a tabular form.

4 Theory

The False Position Method, also known as the Regula Falsi Method, is a numerical method used to find the root of a nonlinear equation of the form $f(x) = 0$. It is a bracketing method that combines aspects of the Bisection Method and the Secant Method. Like the Bisection Method, it starts with two initial guesses, b and c, such that $f(b)$ and $f(c)$ have opposite signs (i.e., $f(b) \cdot f(c) < 0$), ensuring that a root lies between them. Unlike the Bisection Method, the False Position Method uses a weighted average to estimate the root more accurately by considering the function values at the interval endpoints.
The method proceeds by calculating a new approximation m using the formula:

$$m = \frac{b \cdot f(c) - c \cdot f(b)}{f(c) - f(b)}$$

If $f(b) \cdot f(m) < 0$, the root lies in the interval $[b, m]$; otherwise, it lies in $[m, c]$. This process repeats until the function value at m becomes sufficiently close to zero or until a predefined tolerance level for the error is reached.

4.1 Algorithm:

The False Position Method follows these steps:

1. Define the function $f(x)$.
2. Choose two initial points b and c such that $f(b) \cdot f(c) < 0$.
3. Compute the new approximation:

$$m = \frac{b \cdot f(c) - c \cdot f(b)}{f(c) - f(b)}$$

4. Check the sign of $f(m)$:
   (a) If $f(b) \cdot f(m) < 0$, set $c = m$.
   (b) Else, set $b = m$.
5. Repeat steps 3 and 4 until $|f(m)| <$ tolerance or maximum iterations are reached.

4.2 Solving Non-linear Equation Using False Position Method

Here $f(x) = x^3 - 2x - 5$, error is assumed to be 0.001

4.2.1 MATLAB Code;

```matlab

% False Position Method
clc;
clear;

f = @(x) x^3 - 2*x - 5;
error = 0.001;
N = 500;

for i = -1000:1000
if f(i) * f(i+1) < 0
b = i;
c = i + 1;
break;
end
end

for k = 1:N
m = (b*f(c) - c*f(b)) / (f(c) - f(b));
er = abs(c - b);
fprintf('Iteration %d: b = %.6f, c = %.6f, m = %.6f, f(m) = %.6f, error = %.6f\n', k, b, c, m, f(m), er);

if f(b) * f(m) < 0
c = m;
else
b = m;
end


if abs(f(m)) < error || er <= error
fprintf('The root of the equation is %.6f\n', m);
break;
end
end
```

Listing 1: Solving Non-linear Equation Using False Position Method in MATLAB.

4.2.2 Result Shown in Command Window

```
Iteration 1: b = 2.000000, c = 3.000000, m = 2.058824,f(m) = -0.390800,
    error = 1.000000
Iteration 2: b = 2.058824, c = 3.000000, m = 2.081264,f(m) = -0.147204,
    error = 0.941176
Iteration 3: b = 2.081264, c = 3.000000, m = 2.089639,f(m) = -0.054677,
    error = 0.918736
Iteration 4: b = 2.089639, c = 3.000000, m = 2.092740,f(m) = -0.020203,
    error = 0.910361
Iteration 5: b = 2.092740, c = 3.000000, m = 2.093884,f(m) = -0.007451,
    error = 0.907260
Iteration 6: b = 2.093884, c = 3.000000, m = 2.094305,f(m) = -0.002746,
    error = 0.906116
Iteration 7: b = 2.094305, c = 3.000000, m = 2.094461,f(m) = -0.001012,
    error = 0.905695
Iteration 8: b = 2.094461, c = 3.000000, m = 2.094518,f(m) = -0.000373,
    error = 0.905539
The root of the equation is 2.094518
```

Listing 2: Command Window for False Position Method

5 Discussion

In this session, the False Position Method was implemented in MATLAB to solve a non-linear equation of the form $f(x) = x^3 - 2x - 5$. The method begins with two initial guesses between which the root lies and iteratively updates the interval using a formula based on the values of the function at the endpoints. In this experiment, the program automatically determined an appropriate interval and proceeded with the iteration until the error was below the tolerance level of 0.001. The root was found to be approximately 2.094518 after 8 iterations, indicating efficient convergence. The command window displayed all iterations along with intermediate values of the approximations, root estimates, and error.

This lab effectively demonstrated the application of the False Position Method in MATLAB.