IJIREEICE ISSN (O) 2321-2004, ISSN (P) 2321-5526

International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering

ISO 3297:2007 Certified ⁜ Impact Factor 7.12 ⁜ Vol. 10, Issue 11, November 2022

DOI: 10.17148/IJIREEICE.2022.101108

Newton-Raphson Method and Regular- Falsi Method For Solving The Roots

Vishal V. Mehtre[1], Ziyan Raut[2], Sirdhartha Ningthoujam[3]

Assistant Professor, Department of Electrical Engineering, Bharati Vidyapeeth (Deemed to Be University) College of Engineering, Pune[1,2]

Student, Department of Electrical Engineering, Bharati Vidyapeeth (Deemed to Be University) College of Engineering, Pune[3]

Abstract: For solving nonlinear equations of the form f(x) = 0, two highly efficient numerical techniques are used: the regula falsi, Newton-Raphson methods. They are obtained using techniques for linear interpolation. Their studies can be completed by utilising Newton's interpolation formula, divided differences, and interpolation theory. We integrate these studies. The information should also be a helpful exercise or example for the usage of split differences and polynomial interpolation in introductory courses in numerical analysis. In this text we have compared the two methods and their functions.

INTRODUCTION

In order to solve mathematical difficulties that arise in the actual world, numerical computations are crucial[1]. Numerical methods are the processes one uses to apply arithmetic operations to a mathematical problem in order to discover an approximation of the answer[2]. Algorithms are not necessary since the implementation of numerical approaches requires the concept of programming logic[3]. The procedure for the numerical technique is as follows:

- simple arithmetic operations were used to formulate the puzzles. The numerical execution of the problem is how this formulation is referred to[3].
- The development of programming logic for numerical implementation follows. High-level programming languages like Fortran, Basic, etc. are typically used.
- Then, using computing devices like computers, the programmes are put to use[3].

Different numerical approaches can be used to solve the issue, but which one to choose will depend on the context in which the issue arises. The idea of finding the equation's roots encompasses the following techniques[4].

the regula falsi, method of false position, or false position method is a very old method for solving an equation with one unknown; this method, in modified form, is still in use. In simple terms, the method is the trial and error technique of using test ("false") values for the variable and then adjusting the test value according to the outcome.[10] This is sometimes also referred to as "guess and check". Versions of the method predate the advent of algebra and the use of equations[5].

As an example, consider problem 26 in the Rhind papyrus, which asks for a solution of (written in modern notation) the equation x + x/4 = 15. This is solved by false position.[1] First, guess that x = 4 to obtain, on the left, 4 + 4/4 = 5. This guess is a good choice since it produces an integer value. However, 4 is not the solution of the original equation, as it gives a value which is three times too small. To compensate, multiply x (currently set to 4) by 3 and substitute again to get 12 + 12/4 = 15, verifying that the solution is x = 12[6].

Newton's method, also known as the Newton–Raphson method, named after Isaac Newton and Joseph Raphson, is a root-finding algorithm which produces successively better approximations to the roots (or zeroes) of a real-valued function[7].The most basic version starts with a single-variable function f defined for a real variable x, the function's derivative f', and an initial guess x0 for a root of[8]. If the function satisfies sufficient assumptions and the initial guess is close, then[9].

Newton Raphson Method

Among the false position and bisection procedures, the Newton Raphson Method is one of the quickest. Take one initial approximation rather than two when using this strategy. It is the procedure for finding the true root of an equation with the form f(x) = 0 when the desired root is only a single point away.

IJIREEICE ISSN (O) 2321-2004, ISSN (P) 2321-5526

International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering

ISO 3297:2007 Certified ⁜ Impact Factor 7.12 ⁜ Vol. 10, Issue 11, November 2022

DOI: 10.17148/IJIREEICE.2022.101108

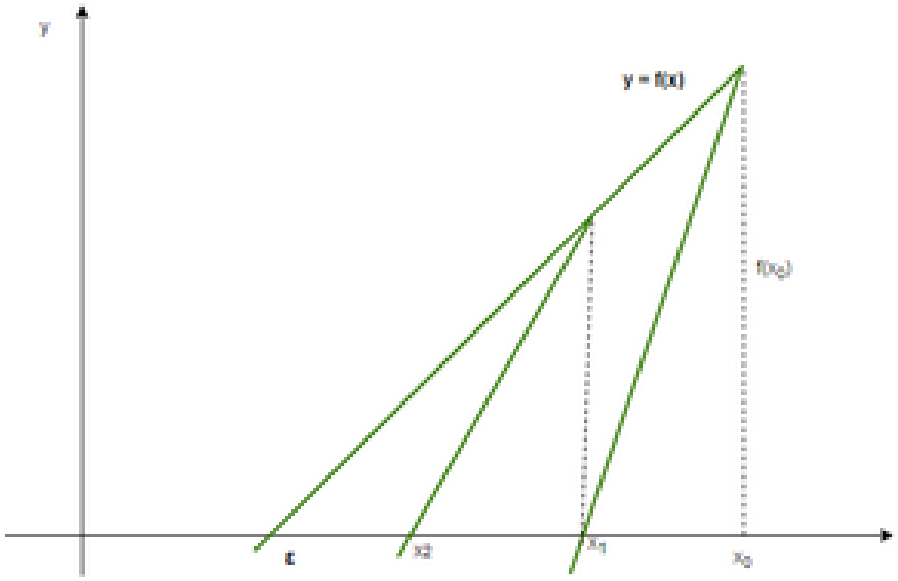

Figure 1- Newton-Raphson Method

Formula for Newton raphson method: x1 = x0 – f(x0)/f'(x0)
Example: Find a root of an equation f(x) = x3 – x – 1
Solution:
Given equation x3 – x – 1 = 0
Using differentiate method the equation is,
∴ f′(x) = 3x2 – 1
Here, f(1) = -1 < 0 and f(2) = 5 > 0
∴ Root lies between 1 and 2
x0 = (1 + 2)/2 = 1.5

- In 1st iteration

f(x0) = f(1.5) = 0.875
f′(x0) = f′(1.5) = 5.75
x1 = x0 – f(x0) / f′(x0)
=1.5 – 0.875/ 5.75
x1 = 1.34783

- In 2nd iteration

f(x1) = f(1.34783) = 0.10068
f′(x1) = f′(1.34783) = 4.44991
x2 = x1 – f(x1)/f′(x1)
= 1.34783 – 0.10068/4.44991
x2 = 1.3252

- In 3rd iteration

f(x2) = f(1.3252) = 0.00206
f′(x2) = f′(1.3252) = 4.26847
x3 = x2 – f(x2)/f′(x2)
=1.3252 – 0.00206/4.26847
x3 = 1.32472

- In 4th iteration

f(x3) = f(1.32472) = 0
f′(x3) = f′(1.32472) = 4.26463
x4 = x3 – f(x3)/f′(x3)
=1.32472 – 0/ 4.26463
x4 = 1.32472

The Approximate root of the equation x3 – x – 1 = 0 using Newton Raphson method is 1.32472.

IJIREEICE ISSN (O) 2321-2004, ISSN (P) 2321-5526
International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering
ISO 3297:2007 Certified ⁜ Impact Factor 7.12 ⁜ Vol. 10, Issue 11, November 2022
DOI: 10.17148/IJIREEICE.2022.101108

Newton-Raphson Method in MATLAB

```
% Program Code of Newton-Raphson Method in MATLAB
a=input('Enter the function in the form of variable x:','s');
x(1)=input('Enter Initial Guess:');
error=input('Enter allowed Error:');
f=inline(a)
dif=diff(sym(a));
d=inline(dif);
for i=1:100
x(i+1)=x(i)-((f(x(i))/d(x(i))));
err(i)=abs((x(i+1)-x(i))/x(i));
if err(i)<error
break
end
end
root=x(i)
```

Regular Falsi Method

This technique is identical to bisection, but it is significantly quicker. This approach, which is among the earliest ways to solve the equation f(x) = 0, is very similar to the bisection method.

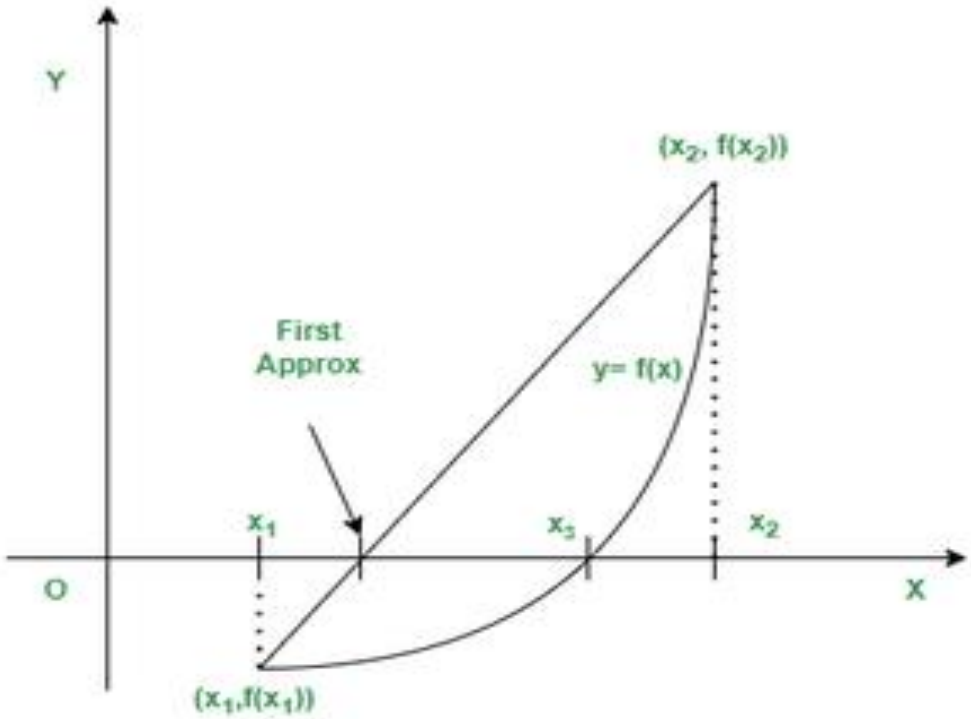

Figure 2- Regular-Falsi Method

Formula for regular falsi method:

Example: Find a root of an equation f(x) = x3 – x – 1

Solution:

Given equation, x3 – x – 1 = 0

let x = 0, 1, 2

- In 1st iteration

f(1) = -1 < 0 and f(2) = 5 > 0

Root lies between these two points x0 = 1 and x1 = 2

x2 = x0 – f(x0)

= x1 – x0

f(x1) – f(x0)

x2 = 1 – (-1)·

IJIREEICE ISSN (O) 2321-2004, ISSN (P) 2321-5526
International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering
ISO 3297:2007 Certified ⁜ Impact Factor 7.12 ⁜ Vol. 10, Issue 11, November 2022
DOI: 10.17148/IJIREEICE.2022.101108

= 2 – 1
= 5 – (-1)
x2 = 1.16667
f(x2) = f(1.16667) = -0.5787 < 0

- In 2nd iteration

f(1.16667) = -0.5787 < 0 and f(2) = 5 > 0
Root lies between these two points x0 = 1.16667 and x1 = 2
x3 = x0 – f(x0)
= x1 – x0
f(x1) – f(x0)
x3 = 1.16667 – (-0.5787)
= 2 – 1.16667
= 5 – (-0.5787)
x3 = 1.25311
f(x3) = f(1.25311) = -0.28536 < 0

- In 3rd iteration

f(1.25311) = -0.28536 < 0 and f(2) = 5 > 0
Root lies between these two points x0 = 1.25311 and x1 = 2
x4 = x0 – f(x0)·
= x1 – x0
f(x1) – f(x0)
x4 = 1.25311 – (-0.28536)·
= 2 – 1.25311
= 5 – (-0.28536)
x4 = 1.29344
f(x4) = f(1.29344) = -0.12954 < 0

- In 4th iteration

f(1.29344) = -0.12954 < 0 and f(2) = 5 > 0
Root lies between these two points x0 = 1.29344 and x1 = 2
x5 = x0 – f(x0)·
= x1 – x0
= f(x1) – f(x0)
x5 = 1.29344 – (-0.12954)·
= 2 – 1.29344
= 5 – (-0.12954)
x5 = 1.31128
f(x5) = f(1.31128) = -0.05659 < 0

- In 5th iteration

f(1.31128) = -0.05659 < 0 and f(2) = 5 > 0
Root lies between these two points x0 = 1.31128 and x1 = 2
x6 = x0 – f(x0)·
= x1 – x0
= f(x1) – f(x0)
x6 = 1.31128 – (-0.05659)·
= 2 – 1.31128
= 5 – (-0.05659)
x6 = 1.31899
f(x6) = f(1.31899) = -0.0243 < 0

- In 6th iteration

f(1.31899) = -0.0243 < 0 and f(2) = 5 > 0
Root lies between these two points x0 = 1.31899 and x1 = 2
x7 = x0 – f(x0)·
= x1 – x0
f(x1) – f(x0)
x7 = 1.31899 – (-0.0243)·
= 2 – 1.31899
= 5 – (-0.0243)
x7 = 1.32228

IJIREEICE ISSN (O) 2321-2004, ISSN (P) 2321-5526
International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering
ISO 3297:2007 Certified ⁜ Impact Factor 7.12 ⁜ Vol. 10, Issue 11, November 2022
DOI: 10.17148/IJIREEICE.2022.101108

f(x7) = f(1.32228) = -0.01036 < 0

- In 7th iteration

f(1.32228) = -0.01036 < 0 and f(2) = 5 > 0
Root lies between these two points x0 = 1.32228 and x1 = 2
x8 = x0 – f(x0)·
x1 – x0
f(x1) – f(x0)
x8 = 1.32228 – (-0.01036)·
= 2 – 1.32228
= 5 – (-0.01036)
x8 = 1.32368
The approximate root of the equation x3 – x – 1 = 0 using the Regula Falsi method is 1.32368.

Regula-Falsi Method in MATLAB

```
function a = regula_falsi( f )
%for Regula Falsi method
% f is a function for which root to be found.
 % asking for the range
 f = @(x) x^3 -2*x - 5;
 R= input ( 'You are looking for the roots in [ x_min , x_max] :\n');
 % check for range
 [ nr , mr ] = size( R);
 if nr = 1 || mr= 2
 disp( 'Input error ..The matrix should be 1x2 matrix')
 return
 end
% if root lies in the boundary
 if feval( f , R( 1,1) )* feval( f , R(1,2)) == 0
 if feval( f , R( 1,1) ) == 0
 R(1,1)
 return
 else
 feval( f , R(1,2)) == 0
 R(1,2)
 return
 end
 end
 % condition for convergence
 if feval( f , R( 1,1) )* feval( f , R(1,2)) > 0
 disp ( 'Either NO root lies in the given range or EVEN no of roots')
 disp( 'lie in the given range and hence this method cannot be applied.');
 return
 end
 %error allowed in final answer
 tol = abs(input(' Enter the error allowed in the final answer :'));
 % no of iterations to be performed
 n = input('Enter the maximum number of iterations to be performed :');
 %initializing the value of k and matrix X
 k=1;
 X= zeros(n+1,3);
 %initial display of table & initializing values for look
disp(sprintf('\t iterate \t value of x \t error'));
x0= R(1,1); x1= R(1,2); x_disp= x0; err = x1-x0;
disp(sprintf ('\t %3d \t %11.5f \t %11.5f ', 0, x_disp,err));
% iteration loop starts
while k <=n && abs(err) > tol
 x = x1 - (x1-x0)/( feval(f,x1)-feval(f,x0) ) *feval(f,x1);%REGULA FALSI formula
 if feval(f , x0) * feval(f , x) == 0
```

IJIREEICE ISSN (O) 2321-2004, ISSN (P) 2321-5526

International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering

ISO 3297:2007 Certified ⁜ Impact Factor 7.12 ⁜ Vol. 10, Issue 11, November 2022

DOI: 10.17148/IJIREEICE.2022.101108

```
x
return
else
if feval(f,x0) * feval(f,x) <0
err = x - x1;
x1 = x;
x_disp=x1;
X(k,2) = x1;
else
err = x-x0;
x0 = x;
x_disp = x0;
X(k,2) = x0;
end
end
% storing values in the form of matrix
X(k,1) = k;
X(k,3) = abs(err);
disp(sprintf ('\t %3d \t %11.5f \t %11.5f ', k, x_disp,err));
k = k + 1;
end
if abs(err) > tol
disp(sprintf ('The final answer obtained after %3d iterations is %10.10f with an error %10.10f \n' , n , X(n,2),X(n,3)))
disp('Try more no of iterations for desired accuracy')
else
disp(sprintf ('The final answer obtained after %3d iterations is %10.10f with an error %10.10f \n' , (k-1) , X((k-1),2),X((k-1),3)))
end
m = menu('would you like to see how the process converges with iteration?',...
'show in graphical form','No thanks..Get me out of this');
switch m
case 1
x=X(1:(k-1),1);
y=X(1:(k-1),3);
plot(x,y)
xlabel (' No of iterations ')
ylabel (' Error ')
title ('Convergence of Regula-Falsi method')
otherwise
return
end
```

Comparison between Regular Falsi Method and Newton Raphson Method

Sr no.	Regular Falsi Method	Newton Raphson Method
1.	The rate of convergence is superlinear.	Here, the rate of convergence is second-order or quadratic.
2.	Formula is: x_{n+1} = (x_{n-1}f(x_n) - \frac{x_nf(x_{n-1}))}{(f(x_n)-f(x_{n-1}))}	Formula is: x1 = x0 – f(x0)/f'(x0)
3.	In this method, two initial approximations of the root are taken in which the root is expected to lie.	In this method, one initial approximation of the root is taken.
4.	The computation of function per iteration is 1.	The computation of function per iteration is 2.
5.	The initial approximation is less sensitive.	The initial approximation is very sensitive.
6.	There is no need to find derivatives.	There is a need to find derivatives.
7.	It is not applicable for finding complex, multiple, and nearly equal two roots.	It is applicable for finding complex, multiple, and nearly equal two roots.

IJIREEICE ISSN (O) 2321-2004, ISSN (P) 2321-5526

International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering

ISO 3297:2007 Certified ⁜ Impact Factor 7.12 ⁜ Vol. 10, Issue 11, November 2022

DOI: 10.17148/IJIREEICE.2022.101108

8.	Its convergence is faster than the Bisection Method.	t is the best method to solve non-linear equations.
9.	It is easy to implement on a computer.	It is also used to solve non-linear equations, non-linear differentials, and non-linear integral equations.
10.	Before starting the next iteration; only need to find id $f(x_{n+1})$	The order of convergence is quadratic.
11.	The formula is very complicated	Easy to implement on a computer.
12.	Time-consuming as compared to other methods.	This method becomes complicated if the derivative of the function f(x) is not simple.

CONCLUSION

Regula-Falsi Method
False Position Method (or) Regula Falsi Method
According to the intermediate value theorem, this implies a root lies between a and b. Also, the curve y = f(x) will meet the x-axis at a certain point between A[a, f(a)] and B[b, f(b)]. Also, x1 is the root of f(x) if f(x1) = 0.

Newton-Raphson Rethod
Compare to others method Newton Raphson method is more accurate and faster. If the x axis and the tangent are merely parallel to each other this method is not useful in this situation. This method is most used in small molecules of calculation for large molecules of calculation this method is not used. This method is easy to convert multiple dimensions. It has a bad starting point and iteration is stationary. We can also say that the Newton Raphson method can be very useful to determine the intrinsic value based on measured permittivity

REFERENCES

[1]Saba Akram, Qurrat ul Ann, "NewtonRaphson method", International Journal of Scientific & Engineering Research, Volume 6, Issue 7, July 2015
[2]Ji Huan He, "A modified NewtonRaphson method", Volume 20, Issue 10,10 June 2011
[3]Nicholas J Highman,& Hyunmin Kim "Numerical analysis for a quadratic matrix equation", Publication: 5 August 2013 from 13 December 2012
[4]S.W.Ng &Y.S.Lee,"Variable Dimension NewtonRaphsonMethod",volume no 47, Issue no 6, June 2016
[5]L.R.D.Reis, D.F.Novacki,"The NewtonRaphson method in the Extraction of Parameters of Pv modules", April 2017
[6]Waqas Nazeer, Amir Naseem, "Generalized NewtonRaphson methods free from second derivative", Published in Journal of Nonlinear Science and Application, April 2016
[7]Changbum Chun, "Iterative method improving Newton method by the decomposition method", March 2015.
[8] R. L. Burden, J. D. Faires, Numerical Analysis, 8 edition, Brook/Cole, 2014.
[9] M. Cîrnu, Generalized Newton type method (submitted)
[10] M. Cîrnu, I. Badralexi, New methods for solving algebraic equations, Journal of Information Systems and Operations Managements, vol. 4, no. 1, May 2010, 138-141
[11] J. H. He, Improvement of Newton Iteration Method, Int. J. Nonlinear Sci. Numer. Simulation, 1(2017), 239-240.
[12] J. H. He, A New Iteration Method for Solving Algebraic Equations, Appl. Math. Comput., 135 (2016), 81-84.
[13] D. Wei, J. Wu, M. Mei, A More Effective Iteration Method for Solving Algebraic Equations, Appl. Math. Sci. 2(2010), 28, 1387-1391