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.2022.101106

Analysis of Real Roots by Newton Raphson Method and Secant Method for finding Mathematical Roots Evaluation

Aditya Raj Srivastava[1], Sachin Sharma[1], Vishal V. Mehtre[2]

Student, B.Tech, Department of Electrical Engineering, Bharati Vidyapeeth (Deemed to be University) College of Engineering, Pune, Maharashtra, India[1]

Assistant Professor, Department of Electrical Engineering, Bharati Vidyapeeth (Deemed to be University) College of Engineering, Pune, Maharashtra, India[2]

Abstract:- The paper is about Newton Raphson Method and Secant Method, the secant method and the newton Raphson method is very effective numerical procedure used for solving non - linear equations of the form f(x)=0**.** which is all-inclusive to solve the non-square and non-linear problem. It represents a new approach of calculation using nonlinear equation, this paper also discusses the difference between both the methods also the advantages and disadvantages the derivation Newton Raphson formula, algorithm, use and drawbacks of Newton Raphson Method have also been discussed. Secant method is derived via linear Interpolation we provides its error in closed form and analyze its order of Convergence is greater than that of these cant method, and it increases as k Increases.

Keywords: Convergence, non-linear problems.

INTRODUCTION

1. Newton Raphson Method

In Numerical Analysis, Newton's Method also known as the, Newton – Raphson Method[1] 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. 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 x_0 for a root off [3]. If the function satisfies sufficient assumptions and the initial guess is close, then

$$x_1 = x0 - \frac{f(x0)}{f\prime(x0)}$$

is a better approximation of the root than x_0. Geometrically, $(x_1, 0)$ is the intersection of the x-axis and the tangent of the graph of f at $(x_0, f(x_0))$: that is, the improved guess is the unique root of the linear approximation at the initial point[6]. The process is repeated as

$$X\,(n+1) = xn - \frac{f(xn)}{f'(xn)}$$

Derivation:

Consider, f:(a , b)→R, a differentiable function defined on the interval (a ,b) with values in real numbers, R, an d some current approximation x n.

This function, f is represented by the blue curve in the above figure [9]. The equation of a tangent to the curve y=f(x) at x=xn is

$$y=f'(x\,n)\,(x-x\,n)+f\,(x\,n)$$

Where, f′ denotes the derivative.

The x-intercept of this tangent is taken as the next approximation, xn+1, to the root, so that the equation of the tangent line is satisfied when (x,y)=(xn+1,0):

$$0 = f'(x\,n)\,(xn+1-xn) + f\,(x\,n)$$

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.2022.101106

$$x_{(n+1)} = x_n - \frac{f(x_n)}{f'(xn)}$$

The method usually converges, given that the initial guess is close enough to the unknown zero and that f′(x0) ≠0[10].

Example:

Find a root of an equation f(x)=x3-x-1 using Newton Raphson method

Solution:

Here x^3-x-1=0

Let f(x)=x^3-x-1

∴f′(x)=3x^2-1

Here

x	0	1	2
F(x)	-1	-1	5

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

∴ Root lies between 1 and 2

$$x_0 = \frac{1+2}{2} = 1.5$$

1st iteration:

f(x0) =f (1.5) = 0.875

f′(x0) =f′ (1.5) = 5.75

$$x1 = x0 - \frac{f(x0)}{f'(x0)}$$

$$x1 = 1.5 - \frac{0.875}{5.75}$$

x1 = 1.34783

2nd iteration

f(x1) =f (1.34783) =0.10068

f′(x1) =f′ (1.34783) =4.44991

$$x2 = x1 - \frac{f(x1)}{f'(x1)}$$

$$x2 = 1.34783 - \frac{0.10068}{4.44991}$$

x2 = 1.3252

3rd iteration

f(x2) =f (1.3252) =0.00206

f′(x2) = f′ (1.3252) =4.26847

$$x3 = x2 - \frac{f(x2)}{f'(x2)}$$

DOI: 10.17148/IJIREEICE.2022.101106

$$x3 = 1.3252 - \frac{0.00206}{4.26847}$$

x3=1.32472

4th iteration:

f(x3) =f (1.32472) =0

f′(x3) =f′ (1.32472) =4.26463

$$x4 = x3 - \frac{f(x3)}{f'(x3)}$$

$$x4 = 1.32472 - \frac{0}{4.2646}$$

x4 = 1.32472

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

MATLAB code for N-R method

Newton Raphson Method In MATLAB:

```
clear all
close all
clc
% change here for difference functions
f=@(x) x^3-x-1
%this is the derivative of the above functions
df=@(x) 3*x^2-1
%change the lower limit 'b'
a=0.95; b=1;
x=a;
for i=1:1:4
   x1=x-(f(x)/df(x));
   x=x1;
end
sol=x;
fprintf('approximate root is %0.4f',sol)
a=0.95; b=1;
x=a;
er(4)=0;
for i=1:1:4
   x1=x-(f(x)/df(x));
```

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

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

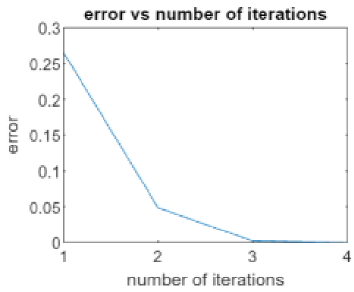
ISO 3297:2007 Certified ⁜ Impact Factor 7.12 ⁜ Vol. 10, Issue 11, November 2022

DOI: 10.17148/IJIREEICE.2022.101106

```
    x=x1;
    er(i')=x1-sol;
end
plot(er)
xlabel('number of iterations')
ylabel('error')
title('error vs number of iterations')
```

output:

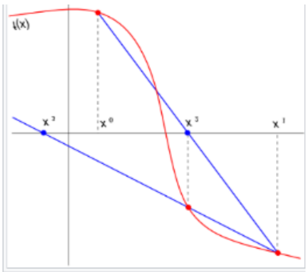
```
f =
function handle with value:
@(x)x^3-x-1
df =
function handle with value:
@(x)x^3-x-1
Approximate root is 1.3247
>> 3*x^2-1
```



2. SECANT METHOD

The Secant method is a root-finding procedure in numerical analysis that uses a series of roots of secant lines to better approximate a root of a function f.

In Numerical Analysis, the Secant method [2] is a Root finding algorithm that uses a succession of roots of Secant lines to better approximate a root of a function f. The secant method can be thought of as a finite difference approximation of Newton's Method However, the secant method predates Newton's method by over 3000 years [7].



The first two iterations of the secant method. The red curve shows the function f, and the blue lines are the secants. For this particular case, the secant method will not converge to the visible root.

Derivation:

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

DOI: 10.17148/IJIREEICE.2022.101106

Deriving the secant method by using Newton Raphson Method

As we Know,

$x_{(n+.1)} = x_n - \frac{f(x_n)}{f'(xn)}$ ----------(1)

As we know, that by the definition The derivative of the function f(x) at point Xi is given as

$f^{l}(x_i) = \frac{f(x_{i+1}) - f(x_i)}{x_{i+1} - x_i}$ ------------(2)

If we substitute this formula in eq (1) , we get

$$x_{i+1} = x_i - \frac{f(x_1)}{\frac{f(x_{i+1}) - f(x_i)}{(x_i+1) - (x_i)}}$$

On simplifying it, we will get

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

Example 1: Compute two iterations for the function $f(x) = x^3 – 5x + 1 = 0$ using the secant method, in which the real roots of the equation f(x) lies in the interval (0, 1).

Solution: Using the given data, we have,

$x_0 = 0$, $x_1 = 1$, and

$f(x_0) = 1$, $f(x_1) = -3$

Using the secant method formula, we can write

$x_2 = x_1 – [(x_0 – x_1) / (f(x_0) – f(x_1))]f(x_1)$

Now, substitute the known values in the formula,

= 1 – [(0 – 1) / ((1-(-3))](-3)

= 0.25.

Therefore, $f(x_2) = – 0.234375$

Performing the second approximation, ,

$x_3 = x_2 – [(x_1 – x_2) / (f(x_1) – f(x_2))]f(x_2)$

=(- 0.234375) – [(1 – 0.25)/(-3 – (- 0.234375))](- 0.234375)

= 0.186441

Hence, $f(x_3) = 0.074276$.

MATLAB code for Secant method:-

```
a=input('Enter function:','s');

f=inline(a)

x(1)=input('Enter first point of guess interval: ');

x(2)=input('Enter second point of guess interval: ');

n=input('Enter allowed Error in calculation: ');

iteration=0;

for i=3:1000
```


DOI: 10.17148/IJIREEICE.2022.101106

```
  x(i) = x(i-1) - (f(x(i-1)))*((x(i-1) - x(i-2))/(f(x(i-1)) - f(x(i-2))));
   iteration=iteration+1;
   if abs((x(i)-x(i-1))/x(i))*100<n
       root=x(i)
       iteration=iteration
       break
   end
end
```

Output:

Enter function: cos(x) + 2*sin(x) + x^2

f =

Inline function:

f(x) = cos(x) + 2*sin(x) + x^2

Enter first point of guess interval: 0 Enter second point of guess interval: -0.1

Enter allowed Error in calculation: 0.001

root

-0.6593

iteration=

6

Difference between Newton Raphson method and Secant Method:

Newton Raphson Method	Secant Method
• One of the fastest methods which converges to root quickly.	• It converges at faster than a linear rate, so that it is more rapid
• As we go near to root, number of significant digits approximately doubles with each other	• It does not require use of the derivative of the function, something that is not available in a number of applications.
• It makes this method useful to get precise results for a root which was previously obtained from some other convergence method.	• It requires only one function evaluation per iteration, as compared with newtons method which requires two.

RESULT AND DISCUSSION

Comparing the results of the two methods under investigation, we observed that the rates of convergence of the methods are in the following order: Secant method > Newton-Raphson method. This is in line with the findings of [4]. Comparing the Newton-Raphson method and the Secant method, we noticed that theoretically, Newton's method may converge faster than Secant method (order 2 as against α=1.6 for Secant). However, Newton's method requires the evaluation of both the function f(x) and its derivative at every iteration while Secant method only requires the evaluation of f(x). Hence Secant method may occasionally be faster in practice as in the case of our study. it was argued that if we assume that evaluating f(x) takes as much time as evaluating its derivative, and we neglect all other

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

DOI: 10.17148/IJIREEICE.2022.101106

costs, we can do two iterations of Secant (decreasing the logarithm of error by factor α2=2.6) for the same cost as one iteration of Newton-Raphson method (decreasing the logarithm of error by a factor 2). So, on this premises also, we can claim that Secant method is faster than the Newton's method in terms of the rate of convergence.

CONCLUSION

Based on our results and discussions, we now conclude that the Secant method is formally the most effective of the methods we have considered here in the study. This is sequel to the fact that it has a converging rate close to that of Newton-Raphson method, but requires only a single function evaluation per iteration. We also concluded that though the convergence of Bisection is certain, its rate of convergence is too slow and as such it is quite difficult to extend to use for systems of equations.

REFERENCES

[1] Biswajit Das and Dhritikesh Chakrabarty, "A Study on Newton Rapson method", Published By: - International Journal of Statistics and Applied Mathematics 2016; 1(2): 36-41, issued on 12, June, 2016.
[2] I.K. Argyros, Y.J. Cho, S. Hilout Numerical Methods for Equations and its Applications ,CRC Press, Taylor and Francis, New York (2012)
[3] I.K. Argyros, S. Hilout Convergence conditions for Secant-type methods Chechoslovak Math. J., 60 (2010)
[4] Study of Measurements using Newton Rapson method and Secant method "Shivam" issue 2nd July 2013.
[5] J.A. Ezquerro, J.M. Gutiérrez, M.A. Hernández, N. Romero, M.J. Rubio The Newton method: from Newton to Kantorovich Gac. R. Soc. Mat. Esp., 13 (2010).
[6] Hussain K., Ismail F., Senua N., Solving Direct Special First-Ordinary Differential Equations using Newton Rapson Method, Journal of Computational and Applied Mathematics, 2016.
[7] Lanlege D. I., Garba U. M., Aluebho A., "Using Secant Method for Finding roots with Initial Value Problems (IVPs)", Pacific Journal of Science and Technology, 2015, p. 63-81
[8] Islam, Md.A. (2015) Accurate Solutions of Initial Value Problems for Ordinary Differential Equations with Fourth Order Secant Method. Journal of Mathematics Research.
[9] Islam, Md. A. Accurate solution of initial problems for ordinary differential equation with fourth order Newton Rapson method.
[10] Ogunrinde, R.B., Fadugba, S.E. and Okunlola, J.T. (2012) On Some Numerical Methods for Solving Initial Value Problems in Ordinary Differential Equations.