زانكۆی سەلاحەدین - هەولێر
Salahaddin University-Erbil

# Heuristic Approaches to find the best method to solution of equation in one variable

Research Project

Submitted to the Department of Mathematics in partial fulfillment of the requirements for the degree of BSc. in MATHEMATICS.

***Prepared by:***

***Amena Najmadeen Muhammd***

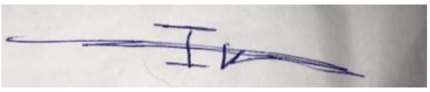
***Supervised by:***

***Assist. Prof. Dr. Ivan Subhi Latif***

**April– 2023**

# Certification of the Supervisor

I certify that this work was prepared under my supervision at the Department of Mathematics / College of Education / Salahaddin University- Erbil in partial fulfillment of the requirements for the degree of Bachelor of philosophy of Science in Mathematics.

Signature: 

Supervisor**: Dr. Ivan Subhi Latif**

Scientific grade**: Assistant Professor**

Date**: / / 2023**

In view of available recommendations, I forward this word for debate by the examining committee.

Signature: 

Name: **Dr. Rashad Rashid Haji**

Scientific grade**: Assistant Professor**

**Chairman of the Mathematics Department**

Date: **/ / 2023**

# Acknowledgment

First of all, I would like to thanks God for helping me to complete this project with success.

Secondly, I would like to express my special thanks to my supervisor Assist. Prof. Dr. Ivan Subhi Latif, it has been great honor to be here student.

It is great pleasure for me to undertake this project I have taken efforts however, it would not have been possible without the support and help of many individuals.

Also, I would like to express my gratitude towards my parents.

My thanks appreciations go to Mathematical Department and all my valuable teachers.

# Abstract

This research examines heuristic approaches to finding the best method for solving equations in one variable, specifically comparing the bisection, secant, false, and newton methods. By applying these methods to a variety of equations and evaluating their performance in terms of accuracy and efficiency, we found that the newton method was the most effective overall. The newton method consistently achieved the highest accuracy in finding the root of the equation, while also requiring fewer iterations than the other methods and minimum time required. However, we also found that the choice of method depended on the nature of the equation being solved, with certain methods being more effective for specific types of equations. Overall, this research contributes to our understanding of how heuristic approaches can be used to efficiently and effectively solve equations in one variable, and highlights the importance of choosing the appropriate method for the specific equation being solved.

# Table of Contents



# List of Table

# CHAPTER ONE

# INTRODUCTION

## 1.1 Introduction

A verity types of equation appeared in a space of mathematics, physics, chemistry and engineering is dedicated to their study. Some of these equations are simple and more of scientists and researchers have seen the linear equation $ax + b = 0$ whose solution is $x = x = \frac{-b}{a}$ where a and b are real numbers and $a \neq 0$. Indeed, it's more unlike to see the form of non-linear equation in the case of cubic and quadratic polynomial situations because if the value of $n \geq 5$ then finding the solution to these polynomial formulae is radicals (Suli & Mayers, 2003).This paper dedicated to finding the root of the nonlinear equation by using some numerical methods. In our considered problem equation contained each of the exponential and trigonometric function together within the devoted equation. Assume that the given function is continuous and bounded within the closed interval $[a, b]$ in the set real numbers $f: R \to R$. Since we have taken throughout the paper $a < b$ then it is implying that the desired closed interval is not empty. Now we want to find the real number $\mathcal{E} \epsilon [a, b]$ provided that $f(\mathcal{E}) = 0$. Consequently, if the value of $\mathcal{E}$ is exists between the interval, then it is an approximate solution to the considered equation $f(x) = 0$. In order to implement the procedure of iterations techniques we introduce each of the Bisection, Newton, False and Secant methods to determine an approximate solution and implanting its algorithms by using MATLAB program (karris, 2007; Stoer & Bulirsch, 2013). The calculation of convergence is very important and absolute of error should be computed in order to guessing the speed of each method as much as presented iteration closes to its limit, more ever calculated by number of evaluations taken in achieving an offered accuracy. The amount of convergence is calculated where the relative error decreases among sequential approximation solution. (F.Epperson, 2013) (Ali, et al., 2021)

## 1.2 History of Numerical

Numerical algorithms are almost as old as human civilization. The Rhind Papyrus (1650 BC) of ancient Egypt describes a root finding method for solving a simple equation Archimedes of Syracuse (287-212 BC) created much new mathematics, including the “method of exhaustion” for calculating lengths, areas, and volumes of geometric figures. when used as a method to find approximations, it is in much the spirit of modern numerical integration; and it was important precursor to the development of the calculus by Isaac Newton and Gottfried Leibnitz. A major impetus to developing numerical procedures was the invention of the calculus by Newton and Leibnitz, as this led to accurate mathematical models for physical reality, first in the physical sciences and eventually in the other sciences, engineering, medicine, and business. These mathematical models cannot usually be solved explicitly, and numerical method to obtain approximate solutions are needed. Another important aspect of the development of numerical method was the creation pf logarithms by Napier (1614) and others, giving a much simpler manner of carrying out the arithmetic operations of multiplication, division, and exponentiation. (Francis, 1956)

# CHAPTER TWO

# BACKGROUND

**Definition (Bisection method)2.1:** (Richard & J.Douglas, 2011)

The Bisection method is used to find the roots of a polynomial equation. It separates the interval and subdivides the interval in which the root of the equation lies. The principle be hiding this method is the intermediate theorem for continuous functions. It works by narrowing the gap between the positive and negative intervals until it closes in on the correct answer. This method narrows the gap by taking the average of the positive and negative intervals. It is a simple method and it is relatively slow. The Bisection method is also known as interval halving method, root finding method, binary search method or dichotomy method.

**Definition of Newton Raphson method 2.2:** (Richard & J.Douglas, 2011)

The most basic version starts with a single-variable function $f$ defined for a real variable $x$, the function's derivative $f\,'$, and an interval guess $x_\circ$ for a root of $f$. If the function satisfies sufficient assumptions and the initial guess is close, then $x_1 = x_\circ - \frac{f(x_\circ)}{f'(x_\circ)}$ is a better approximation of the root than $x_\circ$, Geometrically, $(x_1, 0)$ is the intersection of $f$ at $(x_\circ, f(x_\circ))$: that is, the improved guess is the $_n$unique root of the linear approximation at the initial point. The process is repeated as

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

until a sufficiently precise value is reached. This algorithm is first in the class of Householder's method.

**Definition of False position method 2.3:** (Richard & J.Douglas, 2011)

The poor convergence of the bisection method as well as its poor adaptability to higher dimensions (i.e. systems of two or more non-linear equation) motivate the use of better techniques one such method is the method of false position. Here, we start with an initial interval $[x_1, x_2]$, and we assume that the function changes sign only once in this interval. Now we find an $x_3$in this interval, which is given by the intersection of the x axis and the straight line passing through $(x_1, f\ (x_1))$ and $(x_2, f\ (x_2))$. It is easy to verify that $x_3$ is given by

$$x_3 = x_1 - \frac{(x_2 - x_1)\ * f(x_1)}{f(x_2) - f(x_{1)}}$$

Now, we choose the new interval from the two choices $[x_1, x_3]$ or $[x_2, x_1]$ depending on in which interval the function changes sign.

**Definition of Secant method 2.4**: (Richard & J.Douglas, 2011)

A potential problem in implementing the Newton-Raphson method is the evaluation of the derivative. Although this is not inconvenient for polynomials and many other functions, there are certain function whose derivative may be extremely difficult or inconvenient to evaluate. For these cases, the derivative can be approximated by a backward finite divided difference,

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

This approximation can be substituted into Eq. to yield the following iterative equation:

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

# CHAPTER THREE

# ADVANTAGES AND DISADVANTAGES OF NUMERICAL METHOD TO SOLUTION OF EQUATION IN ONE VARIABLE

## 3.1 Advantages of Numerical Method to solution of equation in one variable

**Bisection Method 3.1.1:** (Richard & J.Douglas, 2011) (Reza, et al., 2021)

I. Convergence is guaranteed Bisection method is bracketing method and it is always convergent.
II. Error can be controlled: In Bisection method, increasing number of iterations always yields more accurate root.
III. Does not involve complex calculations: Bisection method does not require any complex calculations. To perform Bisection method, all we need is to calculate average of two numbers.
IV. Guaranteed error bound: in this method, there is a guaranteed error bound, and it decreases with each successive iteration the error bound deceases by ½ with iteration.
V. Bisection method is very simple and easy to program in computer.
VI. Bisection method is fast in case of multiple roots.

**Secant Method 3.1.2:** (Reza, et al., 2021) (Richard & J.Douglas, 2011)

I. It converges at faster than a linear rate, so that it is more rapidly convergent than the bisection method.
II. It does not require use of the derivative of the function, something that is a variable in a number of applications
III. It requires only one function evaluation per iteration, as compared with Newton’s method which requires two.

**Newton Raphson Method 3.1.3:** (Reza, et al., 2021) (Richard & J.Douglas, 2011)

I. One of the fastest methods which converges to root quickly.
II. Converges on the root quadratically.
III. As we go near to root, number of significate digits approximately doubles with each step.
IV. It makes this method useful to get precise results for a root which was previously obtained from some other convergence method.
V. Easy to convert to multiple dimension.

**False Position Method 3.1.4:** (Kinnari & Janki, 2020) (Reza, et al., 2021) (Richard & J.Douglas, 2011)

I. It does not require the derivative calculation.
II. This method has first order rate of convergence.

## 3.2 Disadvantages of Numerical Method to solution of equation in one variable

**Disadvantages Bisection Method3.2.1:** (Reza, et al., 2021) (Richard & J.Douglas, 2011)

I. Slow rate of convergence: All through convergence of Bisection method is guaranteed, it is generally slow.
II. Choosing one guess dose to root has no advantage: choosing one guess close to the root may result in requiring many iterations to converge.
III. Cannot find root of some equations. For example, $f(x) = x^2$as there are no bracketing values.
IV. It has linear rate of convergence.
V. It falls to determine complex roots.
VI. It cannot be applied if there are discontinuities in the guess interval.

VII. It cannot be applied over an interval where their function takes values of the same sign.

**Disadvantages Secant Method 3.2.2:** (Richard & J.Douglas, 2011) (Reza, et al., 2021)

I. It may not converge.
II. There is no guaranteed error bound for the computed iterates.
III. It is likely to have difficulty if $f\circ(a) = 0$ this means the $x$-axis is tangent to the graph of $Y = f(x)$ at $x = a$.
IV. Newton's method generalizes more easily to new methods for solving simultaneous systems of non-linear equations.

**Disadvantages Newton Raphson Method 3.2.3:** (Kinnari & Janki, 2020) (Reza, et al., 2021) (Richard & J.Douglas, 2011)

I. Its convergence is not guaranteed, so some times for given equation and for given guess we may not get solution.
II. Division by zero problem can occur.
III. Root jumping might take place there by not getting intended solution.
IV. Inflection point issue might occur.
V. Symbolic derivative is required.
VI. In case of multiple roots. This method converges slowly.
VII. Near local maxima and local minima due to oscillation, its convergence is slow.

**Disadvantages False Position Method 3.2.4:** (Reza, et al., 2021) (Kinnari & Janki, 2020) (Richard & J.Douglas, 2011)

I. As it is trial and error method in some case it may take large time span to calculate the correct root and there by slowing down the process.
II. It is used to calculate only a single unknow in the equation.

## 3.3 Algorithm of the numerical method in one variable

**Bisection Method Algorithm:**

Step 1: Read $x_1$, $x_2$, $e$

*Here $x_1$and $x_2$initial guesses $e$ is the absolute error.

Step 2: Compute: $f_1 = f(x_1)$ and $f_2 = f(x_2)$

Step 3: If $(f_1 * f_2) > 0$, then display initial guesses are wrong and go to step (9), otherwise, continue

Step 4: $X = (x_1 + x_2)/2$

Step 5: If ($[(x_1 - x_2)/x] < e$), then display initial guesses are wrong and go to step (9). * Here [] refers to the modulus sign.

Step 6: Else, $f = f\ (x)$

Step 7: If ($(f * f_1) > 0$), then $x_1 = x$ and $f_1 = f$

Step 8: Else, $x_2 = x$ and $f_2 = f$

Step 9: Go to step (4). *Now the loop continues with new values and stop.

**MATLAB:** (George & John, 2019)

$a = input$ (‘$Enter\ function\ with\ right\ hand\ side\ zero$: ‘);

$f = inline(a)$;

$x1 = input$ (‘$Enter\ the\ first\ value\ of\ guess\ interval$:’);

$xu = input$ (‘$Enter\ the\ end\ value\ of\ \ guess\ interval$: ’);

$tol = input$ (‘$Enter\ the\ allowed\ error$: ’);

$if \quad f(xu) * f(x1) \ < 0$

$else$

$f\ print\ f$ (‘$The\ guess\ is\ incorrect$! $Enter\ new\ guesses$ \n ‘);

$x1 = input$ (‘$Enter\ the\ first\ value\ of\ \ guess\ interval$: \n’);

$xu = input$ (‘$Enter\ the\ end\ value\ of\ \ guess\ interval$:\n’);

$end$

$for \ \ i = 2{:}\ 1000$

$xr = \ (xu \ + x1)/2;$

$if\ f(xu) * f(xr) \ < 0$

$xu = \ xr;$

$else$

$x1 = xr;$

$end$

$x\ new\ (1) \ = 0;$

$x\ new\ (i) \ = \ xr;$

$if\ abs((x\ new\ (i) – \ x\ new(i-1))/x\ new(i)) \ < tol,$

$break$

$end$

$end$

$str = [$‘$Therequiredrootoftheequationis$: ‘, $num2str(xr)$, “]

**Secant Method Algorithm:**

Step 1: Define function as $f(x)$

Step 2: Input initial guesses ($x_\circ$ and $x_1$), tolerable error ($e$) and maximum iteration ($N$)

Step 3: Initialize iteration counter $i = 1$

Step 4: If $f(x_\circ) = f(x_1)$ then print "Mathematical Error "and go to step (9) otherwise go to step (5)

Step 5: Calculate $x_2 = x_1 - (x_1 - x_\circ) * f(x_1) / f(x_1) - f(x_\circ)$

Step 6: Increment iteration counter $i = i + 1$

Step 7: If $i >= N$ then print "Not convergent "and go to step (9) otherwise go to step (8)

Step 8: If $| f(x_2) | > e$ then set $x_\circ = x_1, x_1 = x_2$and go to step (4) otherwise go to step (9)

Step 9: Print root as $x_2$ and stop

**MATLAB:** (George & John, 2019)

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

$f = inline(a)$

$x1 = input$ ('$Enter\ first\ point\ of\ guess\ interval$:');

$x2 = input$ ($Enter\ second\ point\ of\ guess\ interval$:');

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

$iteration = 0$;

$for\ i = 3: 1000$

$x(i) = x(i-1) - (f(x(i-1))) * x(i-2)/(f(x(i-1)) - f(x(i-2)));$

$iteration = iteration + 1;$

$if\ abs((x(i)) - x(i-1))/x(i) * 100 < n$

$root = x(i)$

$iteration = iteration$

$break$

$end$

$end$

**Newton Raphson Method Algorithm:**

step 1: Define function as $f(x)$

Step 2: Define first derivative of $f(x)$ as $g(x)$

Step 3: Input initial guess ($x_\circ$), tolerable error ($e$) and maximum iteration ($N$)

Step 4: Initialize iteration counter $i = 1$

Step 5: If $g(x_\circ) = 0$ then print "Mathematical Error "and go to step (10) otherwise go to step (7)

Step 6: Calculate $x_1 = x_\circ - f(x_\circ)/ g(x_\circ)$

Step 7: Increment iteration counter $i = i + 1$

Step 8: If $i >= N$ then print "Not convergent" and go to step (10) otherwise go to step (9)

Step 9: If $f(x_1)$ then set $x_\circ = x_1$and go to step (5) otherwise go to step (10)

Step 10: Print root as $x_1$ and stop

**MATLAB:** (George & John, 2019)

$a = input$ (‘$Enter\ the\ function\ in\ the\ form\ of\ variable\ x$:’,’$s$’);

$x(1) = input$ (‘$Enter\ initial\ guess$:’);

$error = input$ (‘$Enter\ allowed\ error$:’);

$f = inline(a)$

$dif = diff(sym(a))$;

$d = inline(dif)$;

$for\ i = 1{:}\,100$

$x(i + 1) = x(i) - ((f(x(i))/d(x(i)))$;

$err(i) = abs((x(i + 1) - x(i))/x(i))$;

$if\ err(i) < error$

$break$

$end$

$end$

$root = x(i)$

**False Position Method:**

Step 1: Read values of $x_\circ$, $x_1$and $e$.*Here x◦ and $x_1$are the two initial guesses $e$ is the degree of accuracy or the absolute error

Step 2: Computer function values $f(x_\circ)$ and $f(x_1)$

Step 3: Check whether the product of $f(x_\circ)$ and $f(x_1)$ is negative or not, if it is positive take another initial guess, if it is negative then go to step (4)

Step 4: Determine: $x = [x_\circ * f(x_1) - x_1 * f(x_1)]/(f(x_1) - f(x_\circ)$

Step 5: Check whether the product of $f(x_1)$ and $f(x)$ is negative or not if it is negative, assign $x_1 = x$;

Step 6: Check whether the value of $f(x)$ is greater than 0.00001 or not. If yes, go to step (4). If no, go to step (7). *Here the value 0.00001 is the desired degree of accuracy, and hence the stepping criteria. *

Step 7: Display the root as $x$ and stop

**MATLAB:** (George & John, 2019)

$a = input$ (‘$Enter\ function$: ’ ,’$s$’);

$x0 = input$ (‘$Enter\ the\ value\ of\ x0$: ’);

$x1 = input$ (‘$Enter\ the\ value\ of\ x1$: ’);

$e = input$ (‘$Enter\ the\ tolerable$: ’);

$fa = eval\ (subs\ (y, x, a))$;

$fb = eval\ (subs\ (y, x, b))$;

$if\quad fa * fb > o$

$disp$(‘$given\ initial\ values\ do\ not\ bracket\ the\ root$ .’);

$else$

$c = a - (a - b)\ * fa/(fa - fb)$;

$fc = eval\ (subs\ (y, x, c))$;

$print(fc)$;

$while\ abs\ (fc) > e$

$print\ (a, b, c, fc)$;

$if\ fa * fc < 0$

$b = c;$

$fb = eval\ (subs\ (y, x, b));$

$else$

$a = c;$

$fa = eval\ (subs\ (y, x, a));$

$end$

$c = a - (a - b)\ * fa/(fa - fb);$

$fc = eval\ (subs\ (y, x, c));$

$end$

$print\ (root\ is\ c);$

$end$

# CHAPTER FOUR

## NUMERICAL OF ANALYSIS

**Example 4.1:** Use the bisection method, secant method, false position method and newton Raphson method to determine the root of the equation $f(x) = x^3 + 4x^2 - 10$ .Take the initial approximations as $a_1 = 1$, $b_1 = 2$.

**Example 4.2:** Use the bisection method, secant method, false position method and newton Raphson method to determine the root of the equation $f(x) = \cos(x) - xe^x$ .Take the initial approximations as $x_0 = 0$, $x_1 = 1$.

**Example 4.3:** Use the bisection method, secant method, false position method and newton Raphson method to determine the root of the equation $f(x) = 2x + 3x - 5$ .Take the initial approximations as $x_0 = -1$, $x_1 = 2$

**Example 4.4:** Use the bisection method, secant method, false position method and newton Raphson method to determine the root of the equation $f(x) = x^3 - x - 1$ .Take the initial approximations as $x_0 = 1$, $x_1 = 3$.

**Example 4.5:** Use the bisection method, secant method, false position method and newton Raphson method to determine the root of the equation $f(x) = e^x + x$ .Take the initial approximations as $x_0 = -1$, $x_1 = 0$.

**Example 4.6:** Use the bisection method, secant method, false position method and newton Raphson method to determine the root of the equation $f(x) = \cos(x) + 2\sin(x) + x^2$ .Take the initial approximations as $x_0 = -0.9$, $x_1 = 2$.

**Example 4.7:** Use the bisection method, secant method, false position method and newton Raphson method to determine the root of the equation $f(x) = \cos(x) - x$ .Take the initial approximations as $x_0 = 0, x_1 = 2$.

**Example 4.8:** Use the bisection method, secant method, false position method and newton Raphson method to determine the root of the equation $f(x) = 2e^{x^{-2}} \tan(x\pi) - 3$ .Take the initial approximations as $x_0 = 1$, $x_1 = 2.5$

**Example 4.9:** Use the bisection method, secant method, false position method and newton Raphson method to determine the root of the equation $f(x) = e^{-x}\,(3.2\sin(x) - 0.5\cos(x))$ .Take the initial approximations as $x_0 = 0$, $x_1 = 2$.

**Example 4.10:** Use the bisection method, secant method, false position method and newton Raphson method to determine the root of the equation $f(x) = (x-4)^2(1-5x)$ .Take the initial approximations as $x_0 = 0$, $x_1 = 1$.

**Example 4.11:** Use the bisection method, secant method, false position method and newton Raphson method to determine the root of the equation $f(x) = \frac{-2}{(x tan(9\pi x))}$.Take the initial approximations as $x_0 = 0.75$, $x_1 = 2.34$

**Example 4.12:** Use the bisection method, secant method, false position method and newton Raphson method to determine the root of the equation $f(x) = x^2 - 4x + 4 - Inx$ .Take the initial approximations as $x_0 = 0.99$, $x_1 = 3$

**Example 4.13:** Use the bisection method, secant method, false position method and newton Raphson method to determine the root of the equation $f(x) = 3xe^x$ .Take the initial approximations as $x_0 = 0, x_1 = 2$.

**Example 4.14:** Use the bisection method, secant method, false position method and newton Raphson method to determine the root of the equation $f(x) = 2x + 3\cos(x) - e^{-x}$ .Take the initial approximations as $x_0 = -2, x_1 = 1$.

**Example 4.15:** Use the bisection method, secant method, false position method and newton Raphson method to determine the root of the equation $f(x) = (x-2)^2 - Inx$ .Take the initial approximations as $x_0 = 0, x_1 = 1.5$

**Example 4.16:** Use the bisection method, secant method, false position method and newton Raphson method to determine the root of the equation $f(x) = 2x\cos(2\pi x) - (x-4)^{-2}$ .Take the initial approximations as $x_0 = -1, x_1 = 1$.

**Example 4.17:** Use the bisection method, secant method, false position method and newton Raphson method to determine the root of the equation $f(x) = In(x-1) + \cos(x)$ .Take the initial approximations as $x_0 = 3, x_1 = 4$.

**Example 4.18:** Use the bisection method, secant method, false position method and newton Raphson method to determine the root of the equation $f(x) = \mathrm{x} - 0.8 - 0.2\mathrm{sinx}$ .Take the initial approximations as $x_0 = -0.66, x_1 = 1$.

**Example 4.19:** Use the bisection method, secant method, false position method and newton Raphson method to determine the root of the equation $f(x) = -x^3 + \cos(x)$ .Take the initial approximations as $x_0 = 0, x_1 = 3$.

**Example 4.20:** Use the bisection method, secant method, false position method and newton Raphson method to determine the root of the equation $f(x) = \frac{1}{x+1} - \frac{3}{4}$ .Take the initial approximations as $x_0 = 0$, $x_1 = 2$.

**Example 4.21:** Use the bisection method, secant method, false position method and newton Raphson method to determine the root of the equation $f(x) = \mathrm{x} + \frac{1}{x}$ -3sin(x) Take the initial approximations as $x_0 = -1$, $x_1 = 2$.

**Example 4.22:** Use the bisection method, secant method, false position method and newton Raphson method to determine the root of the equation $f(x) = \sqrt{x-1}$ .Take the initial approximations as $x_0 = -1$, $x_1 = 0$.

**Example 4.23:** Use the bisection method, secant method, false position method and newton Raphson method to determine the root of the equation $f(x) = 0.5\mathrm{x} - \frac{1}{x}$ .Take the initial approximations as $x_0 = -1$, $x_1 = 1$.

**Example 4.24:** Use the bisection method, secant method, false position method and newton Raphson method to determine the root of the equation $f(x) = 3\mathrm{xe}^{\mathrm{x}}$ .Take the initial approximations as $x_0 = -1$, $x_1 = 0.5$.

**Example 4.25:** Use the bisection method, secant method, false position method and newton Raphson method to determine the root of the equation $f(x) = \frac{7}{x-2}$ .Take the initial approximations as $x_0 = 0, x_1 = 2$.

**Example 4.26:** Use the bisection method, secant method, false position method and newton Raphson method to determine the root of the equation $f(x) = 1 - 2xe^{1-x}$ .Take the initial approximations as $x_0 = 1, x_1 = 2.7$.

**Example 4.27:** Use the bisection method, secant method, false position method and newton Raphson method to determine the root of the equation $f(x) = \sqrt{x+1}$-$6x^2$Take the initial approximations as $x_0 = -0.8, x_1 = 0$.

**Example 4.28:** Use the bisection method, secant method, false position method and newton Raphson method to determine the root of the equation $f(x) = -xe^x$ .Take the initial approximations as $x_0 = -2, x_1 = 1$.

**Example 4.29:** Use the bisection method, secant method, false position method and newton Raphson method to determine the root of the equation $f(x) = x^3 + 2x^2 - x + 1$ .Take the initial approximations as $x_0 = 0, x_1 = 1$.

**Example 4.30:** Use the bisection method, secant method, false position method and newton Raphson method to determine the root of the equation $f(x) = \sqrt{x - 4x} - e^{8x}$ .Take the initial approximations as $x_0 = -0.8, x_1 = 1$.

(Table 4.1) Solution and time of the four methods by using Matlab.

| $f(x)$ | $x_0$ | $x_1$ | $S_1$ | $T_1$ | $S_2$ | $T_2$ | $S_3$ | $T_3$ | $S_4$ | $T_4$ |
|---|---|---|---|---|---|---|---|---|---|---|
| $x^3 + 4x^2 - 10$ | 1 | 2 | 1.3652 | 9.981491 sec | 1.647 | 5.847129 sec | 1.3652 | 9.494748 sec | 1.5568 | 0.204452 sec |
| $\cos(x) - xe^x$ | 0 | 1 | 0.5178 | 8.447121 sec | 0.593 | 5.829059 sec | 0.5178 | 7.624136 sec | 0.5179 | 0.005861 sec |
| $2x + 3x - 5$ | -1 | 2 | 1.0000 | 5.913741 sec | 1.250 | 14.611352 sec | c is exact root | — | 1.000 | 0.004428 sec |
| $x^3 - x - 1$ | 1 | 3 | 1.3247 | 4.468315 sec | 2.021 | 3.914922 sec | 1.3247 | 3.667681 sec | -0.5671 | 0.004597 sec |
| $e^x + x$ | -1 | 0 | -0.5671 | 8.316807 sec | -0.380 | 5.223344 sec | -0.5671 | 5.522496 sec | 1.3252 | 0.004480 sec |
| $\cos(x) + 2\sin(x) + x^2$ | -0.9 | 2 | -0.6593 | 7.524585 sec | 0.568 | 12.565224 sec | -0.6593 | 7.0101799 sec | -0.6530 | 0.002660 sec |
| $\cos(x) - x$ | 0 | 2 | 0.7391 | 3.610845 sec | 1.171 | 5.260147 sec | 0.7391 | 4.112576 sec | 0.7391 | 0.002809 sec |
| $2e^{-x^{-2}} \tan(x\pi) - 3$ | 1 | 2.5 | 2.2838 | 18.536834 sec | 2.024 | 14.551809 sec | 1.5708 | 6.398835 sec | 1.2056 | 0.007875 sec |
| $e^{-x}(3.2\sin(x) - 0.5\cos(x))$ | 0 | 2 | 0.1550 | 3.252839 sec | 1.372 | 5.369799 sec | 1.550 | 4.117060 sec | 0.1550 | 0.005519 sec |
| $(x - 4)^2(1 - 5x)$ | 0 | 1 | 0.2000 | 4.887500 sec | 0.591 | 5.391672 sec | 0.2000 | 3.508162 sec | 2.000 | 0.252365 sec |

| | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| $\frac{-2}{(x\tan(9\pi x))}$ | 0.75 | 2.34 | 1.8889 | 14.572227 sec | 1.727 | 10.523627 sec | 1.222 | 8.885395 sec | 0.7222 | 0.010221 sec |
| $x^2-4x+4-\text{In}x$ | 0.99 | 3 | 1.4124 | 12.116748 sec | 2.839 | 10.630322 sec | 1.4142_007 | 12.636021 sec | 1.4142 | 0.001822 sec |
| $3xe^x$ | -2 | 1 | 2.3842e_007 | 5.751083 sec | -0.429 | 10.693879 sec | 1.7381e_007 | 4.899973 sec | -6.5641 | 0.001729 sec |
| $2x+3\cos(x)-e^{-x}$ | 0 | 1.5 | 1.2397 | 11.955341 sec | 1.080 | 7.558156 sec | 1.2397 | 7.467100 sec | 0.000 | 0.001790 sec |
| $(x-2)^2-\text{In}x$ | 0 | 2 | 1.4124 | 4.110579 sec | NaN | 4.854609 sec | NaN | 3.571974 sec | NaN | 0.001500sec |
| $2x\cos(2x\pi)-(x-4)^{-2}$ | -1 | 1 | 0.7600 | 9.540087 sec | 0.351 | 4.176187 sec | -0.7453 | 3.781864 sec | 0.0324 | 0.067359 sec |
| $\text{In}(x-1)+\cos(x)$ | 3 | 4 | 3.5265 | 7.246917 sec | 3.625 | 6.306251 sec | 3.5265 | 9.458928 sec | 3.5270 | 0.01061 sec |
| $x-0.8-0.2\sin x$ | -0.66 | 1 | 0.9693 | 16.916829 sec | 0.962 | 7.323086 sec | 0.9643 | 7.824625 sec | 0.9645 | 0.001588 sec |
| $-x^3+\cos(x)$ | 0 | 3 | 0.8655 | 7.611832 sec | 1.526 | 3.589746 sec | 0.8655 | 4.392338 sec | 0.8655 | 0.001536 sec |
| $\frac{1}{x+1}-\frac{3}{4}$ | 0 | 2 | 0.3333 | 12.124310 sec | 1.231 | 4.696770 sec | 0.3333 | 3.805115 sec | 0.3333 | 0.001825 sec |
| $x+\frac{1}{x}-3\sin(x)$ | -1 | 2 | -2.3842e_007 | 6.155229 sec | 0.929 | 7.997775 sec | -9.9688e | 4.662585 sec | NaN | 0.004886 sec |

| | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| $\sqrt{x-1}$ | -1 | 0 | -1.000 | 7.343616 sec | inf | 5.6588785 sec | Inf | 6.482939 sec | -7.000 | 0.005885 sec |
| $0.5-\frac{1}{x}$ | -1 | 1 | -9.5367 | 5.732517 sec | 0.333 | 4.388861 sec | 9.5367e_007 | 5.946039 sec | -1.4142 | 0.004346 sec |
| $4\cos(x)-2^{x}$ | -1 | 0.5 | -2.3842e_007 | 6.352793 sec | -0.113 | 8.313664 sec | 3.0175e_007 | 6.667721 sec | 5.5371e_004 | 0.003701 sec |
| $\frac{7}{x-2}$ | 0 | 2 | 2.000 | 9.919260 sec | NAN | 4.547014 sec | NaN | 4.077080 sec | -14.000 | 0.003075 sec |
| $1-2xe^{1-x}$ | 1 | 2.7 | 2.6783 | 22.476104 sec | 2.678 | 7.023069 sec | 2.6783 | 6.205261 sec | 2.6783 | 0.002701 sec |
| $\sqrt{x+1}-6x^{2}$ | -0.8 | 0 | -3.3645 | 32.139657 sec | -0.148 | 9.778977 sec | -0.3645 | 6.593837 sec | -0.3643 | 0.002722 sec |
| $-xe^{x}$ | -2 | 1 | 2.3842e_007 | 6.350894 sec | -0.429 | 5.134635 sec | 1.7381e-007 | 6.425653 sec | -6.5641 | 0.001160 sec |
| $x^{3}+2x^{2}-x+1$ | 0 | 1 | 0.5550 | 4.240331 sec | 0.667 | 3.916249 sec | 0.5550 | 3.654052 sec | -0.1429 | 0.009114 sec |
| $\sqrt{x-4x}-e^{8x}$ | -0.8 | 1 | -0.852 | 6.404304 sec | 1.000 | 5.660364 sec | -0.0852 | 32.4496601 sec | 1.3949 | 0.00830 sec |
| **General TOTAL of 30 function** | | | $\sum S1$=11.2469 | $\sum T1$=258.4785439 sec | $\sum S2$=28.686 | $\sum T2$=216.063914 | $\sum S3$=23.71277 | $\sum T3$=199.5612215 | $\sum S4$=-10.6478 | $\sum T4$=0.637318 |

Eventually we found that the newton method was the most effective for finding solution and minimum time required.

Note:

$S1$: solution of the bisection method by using Matlab.

$T1$: time of the solution in bisection method.

$S2$: solution of the secant method by using Matlab.

$T2$: time of the solution in secant method.

$S3$: solution of the false position method by using Matlab.

$T3$: time of the solution in false position method.

S4: solution of the newton Raphson method by using matlab.

$T4$: time of the solution in newton Raphson method.

With the error $\varepsilon = 10^{\wedge} - 6$ and n=3

(Table 4.2) compart between the four methods efficient for Time cost

| **Methods** | **Time Cost 100%** |
|---|---|
| Compart between bisection and secant method | 16.4% |
| Compart between bisection and false position method | 22.8% |
| Compart between bisection and newton Raphson method | 99.8% |
| Compart between secant and false position method | 7.6% |
| Compart between secant and newton Raphson method | 99.7% |
| Compart between false position and newton Raphson method | 99.7% |

# References

Ali, W. N., Hemia, M. K., Ribwar, B. I. & Sundws, M. M., 2021. Comparison Between Bisection,Newton and Secant Methods for determining the root of the Non-Linear equation using MATLAB. *Turkish Journal of Computer and Mathematics Education,* 12(14), pp. 1115-1122.
F.Epperson, J., 2013. *An Introduction to Numerical Methods and Analysis.* 2nd ed. canada: Wiley.

Francis, B., 1956. *Introduction to Numerical Analysis.* 2nd ed. New York: Dover Publications,Inc..

George, L. & John, P., 2019. *Numerical Methods.* 4th ed. London: Acadimic Press.

Kinnari, C. & Janki, M., 2020. Analogical Study of Newton-Raphson Method & False Position Method. *International Journal of Creative Research Thoughts(IJCRT),* 8(4), pp. 3508-3511.

Reza, E., Mohsen, M., Mashaallah, M. & Reza, A. G., 2021. Anew method for rooting nonlinear equations based on the Bisection method. *National Library of Medicine,* 8(2215-0161), p. 101502.

Richard, L. & J.Douglas, F., 2011. *Numerical Analisis.* 9th ed. Boston: Richard Stratton.

# پوختە

ئەم توێژینەوەیە لێکۆڵینەوە لە شێوازە هێوریستییەکان دەکات بۆ دۆزینەوەی باشترین ڕێگە بۆ چارەسەرکردنی هاوکێشەکان لە یەک گۆڕاودا، بە تایبەتی بەراوردکردنی bisection ، secant، false , newton methods . بە بەکارهێنانی ئەم شێوازانە بۆ هاوکێشە جۆراوجۆرەکان و هەڵسەنگاندنی ئەدای کارکردنیان لە ڕووی وردبینی و کاراییەوە، بۆمان دەرکەوت کە شێوازی نیوتن بە گشتی کاریگەرترینە. شێوازی نیوتن بە بەردەوامی بەرزترین وردبینی لە دۆزینەوەی ڕەگی هاوکێشەیەکدا بەدەستهێناوە، لە هەمان کاتدا پێویستی بە دووبارەکردنەوەی کەمتر هەیە لە چاو شێوازەکانی تر و کەمترین کاتی پێویستە. هەروەها بۆمان دەرکەوت کە هەڵبژاردنی شێوازەکە بەندە بە سروشتی ئەو هاوکێشەیەی کە چارەسەر دەکرێت، لەگەڵ هەندێک شێواز کە کاریگەرترن بۆ هەندێک جۆری هاوکێشە. بە گشتی ئەم توێژینەوەیە بە شدارە لە تێگەیشتنمان لەوەی کە چۆن دەتوانرێت شێوازە ئەزموونییەکان بەکاربهێنرێت بۆ چارەسەرکردنی هاوکێشەکان بە شێوەیەکی کارا و کاریگەر لە یەک گۆڕاودا، و گرنگی هەڵبژاردنی شێوازی گونجاو بۆ ئەو هاوکێشە تایبەتەی کە چارەسەر دەکرێت، دەردەخات.

# الخلاصة


يبحث هذا البحث في الأساليب الاستكشافية لإيجاد أفضل طريقة لحل المعادلات في متغير واحد ، وتحديداً مقارنة طرق bisection، secant، false ، newton methods . من خلال تطبيق هذه الطرق على مجموعة متنوعة من المعادلات وتقييم أدائها من حيث الدقة والكفاءة ، وجدنا أن طريقة نيوتن كانت الأكثر فعالية بشكل عام. حققت طريقة نيوتن باستمرار أعلى دقة في إيجاد جذر المعادلة ، بينما تتطلب أيضًا تكرارات أقل من الطرق الأخرى و أقل الوقت. ومع ذلك ، وجدنا أيضًا أن اختيار الطريقة يعتمد على طبيعة المعادلة التي يتم حلها ، حيث تكون بعض الطرق أكثر فاعلية لأنواع معينة من المعادلات. بشكل عام ، يساهم هذا البحث في فهمنا لكيفية استخدام الأساليب التجريبية لحل المعادلات بكفاءة وفعالية في متغير واحد ، ويسلط الضوء على أهمية اختيار الطريقة المناسبة للمعادلة المحددة التي يتم حلها.