

# A New Algorithm to Compute the Real Root of Transcendental Equation

Hajar AlZawaideh<sup>†</sup>, Feras M. AlFaqih<sup>†</sup> and Sahem Tarawneh<sup>†</sup>

Department of Mathematics, Faculty of Science, Al Hussein Bin Talal University, Ma'an,  
The Hashemite Kingdom of Jordan  
e-mail: falfaqih@ahu.edu.jo

Received 10 March 2022; Accepted 1 May 2022

## Abstract

*In this work, we construct a new algorithm for finding the real non-zero root of transcendental equation using exponential function. The new suggested algorithm based on Newton-Raphson method and Householder's method. In addition, MATLAB has been used to show the efficiency of such algorithm. In addition, to some numerical examples are listed using the proposed algorithm. We conclude that our algorithm will help to get good performance in commercial and science packages compared with some other methods.*

**Keywords:** *transcendental equations, Exponential function, Newton-Raphson method, Householder's method.*

**2020 Mathematics Subject Classification:** 65H04, 65H05, 65K05

## 1 Introduction

A root-finding algorithm of continuous functions is a method for obtaining zeroes, often known as "roots", in mathematics and computers. A zero of real valued function  $f$ , is a number  $x$  such that  $f(x) = 0$ . As generally, the roots of a function cannot be computed exactly nor expressed in closed form. Approximations to zeroes are provided by root-finding algorithms, which can be expressed as floating point numbers or on small isolating intervals.

Solving an equation  $f(x) = g(x)$  is the same as finding the zeroes of the function  $h(x) = f(x) - g(x)$ . Thus root-finding algorithms allow solving any

equation defined by continuous functions. In particular, most root-finding algorithms do not guarantee that they will find all the roots. As well, the existence of a function roots cannot be confirmed, if such used algorithm failed on finding these roots.

The root-finding methods were first conducted by Newton's method, which was written in 1669 and published in 1711 by William Jones [4]. In particular, Newton's method has been used only on root approximations of polynomials. Next, many methods were conducted under this topic, for example Bisection method, Newton-Raphson method, Secant method, Regula-Falsi and Muller's methods etc., for more about the approximation methods of roots finding one can see [2, 6–8, 10].

The main criterion to use such methods, is the rate of convergence, which is the values that represent how fast the method approaches its limit. Certainly, many iterative methods were developed to find a simple zero of a nonlinear function  $f(x)$ , also for finding multiple zeros [5, 9], these methods are called bracketing methods. Akram and Ann [3], compared the rate of convergence of several methods for solving nonlinear equations. One can look at many modified methods that convergence and efficiency in the local area should be improved as it has been suggested in recent literature [1, 11–15] and references therein. One can also see new developed methods in [16–20].

In this paper we are concentrating on developing an efficient algorithm using some ideas of Householder's method [22] and some methods concluded in [23]. Next we recall some basic concepts related to these methods:

The Householder's method for solving nonlinear equation with cubic order of convergence [21], is given as:

$$x_{n+1} = x_n \frac{f(x_n)}{f'(x_n)} - \frac{(f(x_n))^2 f''(x_n)}{2(f'(x_n))^3} \quad (1)$$

The paper "New Algorithms For Computing A Root Of Non-Linear Equations Using Exponential Series", with a rate of convergence greater than or equal two ( $p \geq 2$ ) are proposed as:

$$x_{n+1} = x_n \exp\left(\frac{-f(x_n)}{x_n f'(x_n)}\right) \quad (2)$$

## 2 The new proposed iterative algorithm

In this section, we introduce a new iterative algorithm formula (3) for obtaining the root of a nonlinear equation  $f(x) = 0$ ,

$$x_{n+1} = x_n \left[ \exp \left( \frac{-f(x_n)}{x_n f'(x_n)} \right) - \frac{(f(x_n))^2 f''(x_n)}{2(f'(x_n))^3} \right], n = 0, 1, 2, \dots \quad (3)$$

This algorithm will be investigated in Theorem 2.1.

We developed algorithm (4) in [23] which has an order of convergence  $p \geq 2$  :

$$x_{n+1} = x_n \exp \left( \frac{-f(x_n)}{x_n f'(x_n)} \right), n = 0, 1, 2, \dots \quad (4)$$

By expanding this formula, we got many methods based on Series truncation, the first one is the standard Newton- Raphson method [?], that has an order of convergence  $p = 2$ .

These formulae:

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}, n = 0, 1, 2, \dots \quad (5)$$

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} + \frac{1}{2x_n} \left( \frac{f(x_n)}{f'(x_n)} \right)^2 \quad (6)$$

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} + \frac{1}{2x_n} \left( \frac{f(x_n)}{f'(x_n)} \right)^2 - \frac{1}{6x_n^2} \left( \frac{f(x_n)}{f'(x_n)} \right)^3 \quad (7)$$

By these formulae the new algorithm (4) that proved in [23] has an order of convergence  $p \geq 2$ .

Now, we will show the next theorem on which we develop our new algorithm.

**Theorem 2.1.** *Let  $D$  be an open interval of real number and  $f : D \rightarrow R$  be a differentiable nonlinear function and  $f(x) = 0$  with a simple root  $\alpha \in D$  . If  $x_0$  close to  $\alpha$ , then formula (3) has at least convergence more than  $p \geq 2$ .*

*Proof:* The new iterative formula (3) is

$$x_{n+1} = x_n \left[ \exp \left( \frac{-f(x_n)}{x_n f'(x_n)} \right) - \frac{(f(x_n))^2 f''(x_n)}{2(f'(x_n))^3} \right], n = 0, 1, 2, \dots$$

Since

$$\lim_{x_n \rightarrow \alpha} \exp \left( \frac{-f(x_n)}{x_n f'(x_n)} \right) = 1 \quad \& \quad \lim_{x_n \rightarrow \alpha} \frac{(f(x_n))^2 f''(x_n)}{2(f'(x_n))^3} = 0$$

then  $x_{n+1} = \alpha$

Using the expansion of the exponential function and from eq.(4) we have

$$\begin{aligned}
 x_{n+1} &= x_n \left[ \exp \left( \frac{-f(x_n)}{x_n f'(x_n)} \right) - \frac{(f(x_n))^2 f''(x_n)}{2(f'(x_n))^3} \right] \\
 &= x_n \left( 1 + \left( \frac{-f(x_n)}{x_n f'(x_n)} \right) + \frac{1}{2} \left( \frac{-f(x_n)}{x_n f'(x_n)} \right)^2 + \frac{1}{6} \left( \frac{-f(x_n)}{x_n f'(x_n)} \right)^3 \right. \\
 &\quad \left. + \frac{1}{24} \left( \frac{-f(x_n)}{x_n f'(x_n)} \right)^4 + \dots + \frac{(-f(x_n))^2 f''(x_n)}{2(f'(x_n))^3} \right) \\
 &= \left( x_n - \frac{f(x_n)}{f'(x_n)} + \frac{1}{2x_n} \left( \frac{f(x_n)}{f'(x_n)} \right)^2 - \frac{1}{6x_n^2} \left( \frac{f(x_n)}{f'(x_n)} \right)^3 \right. \\
 &\quad \left. + o \left( \frac{1}{24x_n^3} \left( \frac{-f(x_n)}{x_n f'(x_n)} \right)^4 \right) - \frac{x_n (f(x_n))^2 f''(x_n)}{2(f'(x_n))^3} \right)
 \end{aligned}$$

since  $f(x_n) \approx 0$ , and neglect some terms which have a higher order, then we have the following algorithms:

- **Algorithm (2.1)**

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} - \frac{x_n (f(x_n))^2 f''(x_n)}{2(f'(x_n))^3}, \quad n = 0, 1, 2, \dots$$

- **Algorithm (2.2)**

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} + \frac{1}{2x_n} \left( \frac{f(x_n)}{f'(x_n)} \right)^2 - \frac{x_n (f(x_n))^2 f''(x_n)}{2(f'(x_n))^3}, \quad n = 0, 1, 2, \dots$$

- **New Proposed Iterative Algorithm (PIA)**

$$\begin{aligned}
 x_{n+1} &= x_n - \frac{f(x_n)}{f'(x_n)} + \frac{1}{2x_n} \left( \frac{f(x_n)}{f'(x_n)} \right)^2 - \frac{1}{6x_n^2} \left( \frac{f(x_n)}{f'(x_n)} \right)^3 + o \left( \frac{1}{24x_n^3} \left( \frac{-f(x_n)}{x_n f'(x_n)} \right)^4 \right) \\
 &\quad - \frac{x_n (f(x_n))^2 f''(x_n)}{2(f'(x_n))^3}
 \end{aligned}$$

$$\equiv x_{n+1} = x_n \left[ \exp \left( \frac{-f(x_n)}{x_n f'(x_n)} \right) - \frac{(f(x_n))^2 f''(x_n)}{2(f'(x_n))^3} \right], n = 0, 1, 2, \dots$$

In the above algorithms, the first algorithm is Newton-Raphson method in the first two terms which have a quadratic convergence and equation (4) has an order of convergence  $p \geq 2$  see [23].

Therefore, the order of convergence for new proposed algorithm is more than the order of convergence in equation (4).

This result will be showed by numerical examples in the next section.

### 3 Enforcement the new proposed algorithm

Now we will show how to represent our new proposed algorithm on MATLAB and steps for computing real valued function root.

#### 3.1 The steps for computing root

1. Pick up an initial approximation  $x_0 \neq 0$  and  $f'(x_0) \neq 0$ .
2. The new iterative formula which is given in equation ( 3) should be applied.
3. Follow Step 2 until the approximate root is obtained , for  $n = 0, 1, 2, \dots$

the next figure is a flow chart of this algorithm.

Figure 1: Flow chart of the proposed algorithm

#### 3.2 The new proposed algorithm in MATLAB

Here we describe the enforcement of the new proposed algorithm in MATLAB, where  $f$  is given a nonlinear equation,  $x$  is the initial guess of the root,  $iter$  is the number of iterations needed and  $eps$  is the relative error (let it equal  $1 \times 10^{-8}$ ).

```
clc;clear;
```

```

syms x
f=input('f(x)=');
df=diff(f,x);
ddf=diff(f,2);
f=inline(f);
df=inline(df);
ddf=inline(ddf);
x=input('x=');
eps=10^(-8);
iter=1;
xn=x*exp(-(f(x)/(df(x)*x)))-(x*f(x)^2*ddf(x)/(2*df(x)^3));
tol=abs(xn-x);
while tol>eps
iter=iter+1;
x=xn;
xn=x*exp(-(f(x)/(df(x)*x)))-(x*f(x)^2*ddf(x)/(2*df(x)^3));
tol=abs(xn-x);
end
disp([xn , iter])

```

## 4 Numerical examples

In this section, we will apply our new proposed iterative algorithm (PIA) on many non-linear equations and compare it with many well-known methods such as Newton-Raphson method (NR), Regula Falsi method (RFM), Steffensen's method (STM), Householder's method (HHM) and new algorithm for computing a root of non-linear equations using exponential series (EXPM). Table 1, shows the number of iterations to obtain the required approximate root in different methods and showmen the efficiency of the our proposed method. The following examples are of non-liner equations with given initial approximations and tolerance error  $1 \times 10^{-8}$  for simplicity.

1.  $f_1(x) = x^2 - (1 - x)^5$ , with initial approximation 5.
2.  $f_2(x) = x^3 - e^{-x}$ , with initial approximation 6.
3.  $f_3(x) = -20x^5 - \frac{x}{2} + \frac{1}{2}$ , with initial approximation 1.5.
4.  $f_4(x) = \ln(x - 1) + \cos(x - 1)$ , with initial approximation 1.3.
5.  $f_5(x) = (e^x + x - 20)^3$ , with initial approximation 4.

6.  $f_6(x) = x - 3 \ln x$ , with initial approximation 0.5.

7.  $f_7(x) = 11x^{11} - 1$ , with initial approximation 1.

8.  $f_8(x) = xe^{-x} - 0.1$ , with initial approximation 0.1.

Table 1: Comparison of iterations by different methods

	IT	$x_n$	$f(x_n)$
$f_1, x_0 = 5$			
NRM	12	0.345954815848242	-6.938893903907228e-17
STM	failure		
RFM	22	0.345954821403951	8.927287131754724e-09
EXPM	13	0.345954815848242	-6.938893903907228e-17
HHM	8	0.345954815848242	-6.938893903907228e-17
PIA	7	0.345954815848242	-6.938893903907228e-17
$f_2, x_0 = 6$			
NRM	10	0.772882959149210	-2.220446049250313e-16
STM	390	0.007728829591492	-0.992300499333549
RFM	45	0.772882955215741	-8.864955047815926e-09
EXPM	11	0.772882959149210	-2.220446049250313e-16
HHM	7	0.772882959149210	-2.220446049250313e-16
PIA	8	0.772882959149210	-2.220446049250313e-16
$f_3, x_0 = 1.5$			
NRM	10	0.427677296931004	-1.332267629550188e-15
STM	failure		
RFM	15	0.427677295853494	4.143594356520453e-09
EXPM	11	0.427677296931004	-1.332267629550188e-15
HHM	7	0.427677296931004	-1.332267629550188e-15
PIA	9	0.427677296931004	-1.332267629550188e-15
$f_4, x_0 = 1.3$			
NRM	5	1.397748475958747	2.220446049250313e-16
STM	6	1.397748475958747	2.220446049250313e-16
RFM	11	1.397748477721991	3.750081800646399e-09
EXPM	5	1.397748475958747	2.220446049250313e-16
HHM	4	1.397748475958747	2.220446049250313e-16
PIA	4	1.397748475958747	2.220446049250313e-16
$f_5, x_0 = 4$			
NRM	47	2.842438968413151	1.874088523482121e-20
STM	failure		
RFM	divergent		
EXPM	47	2.842438969577919	2.358334350437148e-20
HHM	33	2.842438963549335	5.574094632581084e-21
PIA	19	2.842438957769491	3.788539833477978e-22

$f_6, x_0 = 0.5$			
NRM	7	1.857183860207835	2.220446049250313e-16
STM	10	4.536403654973539	4.440892098500626e-15
RFM	22	1.857183870520140	-6.345665770624009e-09
EXPM	6	1.857183860207836	-4.440892098500626e-16
HHM	5	1.857183860207835	2.220446049250313e-16
PIA	6	1.857183860207835	2.220446049250313e-16
$f_7, x_0 = 1$			
NRM	7	0.804133097503664	-4.218847493575595e-15
STM	failure		
RFM	75	0.804133096915643	-8.043735566509724e-09
EXPM	7	0.804133097503664	-4.218847493575595e-15
HHM	5	0.804133097503664	-4.218847493575595e-15
PIA	6	0.804133097503664	-4.218847493575595e-15
$f_8, x_0 = 0.1$			
NRM	4	0.111832559158963	1.387778780781446e-17
STM	4	0.111832559158963	1.387778780781446e-17
RFM	9	0.111832563450532	3.408338222765650e-09
EXPM	4	0.111832559158963	1.387778780781446e-17
HHM	3	0.111832559158963	1.387778780781446e-17
PIA	4	0.111832559158963	1.387778780781446e-17

Iterations at different methods of  $f_1(x)$     Iterations at different methods of  $f_2(x)$

Iterations at different methods of  $f_3(x)$     Iterations at different methods of  $f_4(x)$

Iterations at different methods of  $f_5(x)$     Iterations at different method of  $f_6(x)$

Iterations at different methods of  $f_7(x)$     Iterations at different methods of  $f_8(x)$

As we see in the previous figures , the number of iterations for finding on approximate solution of functions ( $f_1(x) : f_8(x)$ ) using our new proposed

iterative algorithm (PIA) are less than the other methods. We conclude that our proposed method is converge faster than many other methods.

## 5 Conclusions

In this study, we proposed a new iterative algorithm approximation root of transcendental one variable equation which is given using Newton-Raphson method with exponential series and the Householder method.

We have compared our new proposed algorithm with many other methods as shown in the table, which reveals the efficiency of the proposed method.

From the table it's clear that our new algorithm has less numbers of iterations and sometimes the same number with some methods.

So, based on the above analysis, our proposed method can be used as a substitute for methods mentioned in table (1).

## 6 Open problem

This paper presents a new algorithm to compute the real root of a transcendental equation. As an open problem we suggest extending the algorithm to approximate the solution of a system of nonlinear equations.

## References

- [1] Popovski D. B., *Some sixth-order variants of ostrowski root-finding methods*, Applied Mathematics and Computation, 193 (2007)no.2, 389–394.
- [2] J. Chen and W. Li, *An exponential regula falsi method for solving nonlinear equations*, Applied Mathematics and Computation, 41(2006),327–338.
- [3] Saba Akram and Qurrat Ul Ann, *Newton raphson method*, International Journal of Scientific and Engineering Research,6(2015)no.7, 1748–1752.
- [4] Hald Anders, *A history of probability and statistics and their applications before 1750*,John Wiley & Sons, 574 (2005).
- [5] Popovski D. B., *A note on king's method f for finding a bracketed root*, Computing, 29(1982)no.4, 355–359.
- [6] J. Chen and W. Li, *An improved exponential regula falsi methods with quadratic convergence of both diameter and point for solving nonlinear equations*, Applied Mathematics and Computation, 57(2007), 80–88.
- [7] J.S. Chitode, *Numerical Techniques*, Technical Publications, (2021).

- [8] Dowell-Jarrat, *A modified regula-falsi method for computing the real root of an equation*, BIT Numerical Mathematics, 11 (1971), 168–174.
- [9] King R. F., *Methods without secant steps for finding a bracketed root*, Computing, 17 (1976)no.1, 49–57.
- [10] C. Gutierrez, F. Gutierrez, and M.-C. Rivara, *Complexity of the bisection method*, Theoretical Computer Science, 382 (2007),131–138.
- [11] Kou Jisheng, Li Yitian, and Wang Xiuhua, *Some variants of ostrowski's method with seventh-order convergence*, Journal of Computational and Applied Mathematics, 209(2007)no.2, 153–159.
- [12] Miodrag S Petkovic and Beong In Yun, *Sigmoid-like functions and root finding methods*, Applied mathematics and computation, 204(2008) no.2,784–793.
- [13] Bi Weihong, Ren Hongmin, and Wu Qingbiao, *Three-step iterative methods with eighth-order convergence for solving nonlinear equations*, Journal of Computational and Applied Mathematics, 255(2009)no.1, 105–112.
- [14] Bi Weihong, Wu Qingbiao, and Ren Hongmin, *A new family of eighth-order iterative methods for solving nonlinear equations*, Applied Mathematics and Computation, 214(2009)no.1, 236–245.
- [15] Wang Xia and Liu Liping, *Two new families of sixth-order methods for solving non-linear equations*, Applied Mathematics and Computation, 213(2009) no.1,73–78.
- [16] SA Ibrahim , *An improved exponential regula falsi methods with cubic convergence for solving nonlinear equations*, Journal of applied mathematics & informatics, 28(2010)no.5\_6, 1467–1476.
- [17] Gustavo Fernández Torres, and Francisco Rubén Castillo Soria, *Some New Derivative Free Methods For Solving Nonlinear Equations*, Academic Research International, 2 (2012)no.1.
- [18] Srinivasarao Thota, *A numerical algorithm to find a root of non-linear equations using householder's method*, Int J Adv Appl Sci, 10(2021) no.2,141–148.
- [19] Sunethra Weerakoon, and TGI Fernando, *A variant of Newton's method with accelerated third-order convergence*, Applied mathematics letters, 13(2000), 700-713.

- [20] Ababu Teklemariam Tiruneh, William N Ndlela and Stanley J Nkambule, *A Two-Point Newton Method Suitable for Nonconvergent Cases and with Super-Quadratic Convergence*, *Advances in Numerical Analysis*, (2013).
- [21] Alston Scott Householder, *The numerical treatment of a single nonlinear equation*, McGraw-Hill, (1970).
- [22] Khalida Inayat Noor, Muhammad Aslam Noor, and Shaher Momani, *Modified Householder iterative method for nonlinear equations*, *Applied mathematics and computation*, 190 (2007)no.2, 1534–1539.
- [23] Srinivasarao Thota, Tekle Gemechu, and P Shanmugasundaram, *New algorithms for computing a root of non-linear equations using exponential series.*, *Palestine Journal of Mathematics*, 10 (2021)no.1.