[Volume 2 issue 11 November 2014]
Page No.729-735 ISSN :2320-7167

INTERNATIONAL JOURNAL OF MATHEMATICS AND COMPUTER RESEARCH

Interpolation based Hybrid Algorithm for Computing Real Root of Non-Linear Transcendental Functions

Srinivasarao Thota and Vivek Kumar Srivastav

Department of Mathematics
United Institute of Technology
Allahabad, India – 211010
email: srinithota@ymail.com and vivekapril@gmail.com

Abstract

This paper present a new algorithm to find the root of non-linear transcendental functions. The new proposed algorithm is based on the combination of Regula-Falsi and Muller's Methods. It is found that Regula-Falsi method always gives guaranteed result but slow convergence. Muller is used the concept of interpolation to compute the root with faster convergence. As, it is well known that interpolation techniques are more popular and efficient for finding missing values. Therefore, the present paper used these two ideas and developed a new quadratically convergent algorithm. Error calculation has been done for real life examples using existing methods and new proposed method. The computed result shows that new proposed method provides better convergence than other methods.

keywords: Root of non-linear transcendental functions, Regula-Falsi method, Muller's method, hybrid algorithm.

1. Introduction

Most of the real life-problems are non-linear in general therefore it is the challenging task for the mathematician and engineer to find the exact solution of such problems [1, 2]. In this reference, a number of methods have been proposed/implemented in the last two decades [1, 3-7]. Analytical solution of such non-linear equations are very difficult therefore only numerical method based iterative techniques is the way to find approximate solution. From the literature, there are some numerical methods such as Bisection, Secant, Regula-Falsi, Newton-Raphson, Muller's methods etc., to find approximate root of non-linear transcendental equations. It is well known that all the iterative methods require one or more initial guesses for the initial approximations. These initial approximations are used in different ways i.e., interpolation techniques, exponential form, derivative and without derivative form etc. The following subsections describes survey of different algorithms related to Regula-Falsi and Muller's methods. Then new-proposed method is presented on the basis of these two methods.

Regula-Falsi method is popular because it gives guaranteed root and faster convergence over Bisection method. This method has adopted the concept of straight line which is obtained by connecting two guess values such that their corresponding functions have opposite sign. The intersecting point of the straight line is the new approximate root. This process is lengthy and time consuming, therefore several researchers have implemented this standard Regula-Falsi method into different hybrid models to speed up the rate of convergence [1, 3-5, 7-11].

Dowell and Jarratt (1971) has given the modified form of standard Regula-Falsi method by replacement of '*new-function*' to '*half of the some intermediate function*'. It was found better convergence than previous Regula-Falsi Method [11]. Many Researchers have used Steffensen's method with standard Regula-Falsi method to speed up the rate of convergence [1, 3, 4]. In this reference, Wu and Wu (2000) has proposed quadratic convergent algorithm without using the derivative function. This new algorithm was given by employing Steffensen's method of accelerating convergence after using standard Regula-Falsi method (RFM) [3, 4]. Kanwar et al.(2005) [5] discussed third-order iterative methods for solving non-linear equations. Using the above concept [12], they have given third-order multi-point methods without using second derivative. Sharma and Goyal (2006) studied fourth order derivative-free methods. This algorithm is developed using Steffensen's method, which use the idea of forward and backward differences [1]. It is found that few research has been done using two step methods [6, 7]. Noor and Ahmad (2006) suggested predictor-corrector method type iterative method by using standard Regula Falsi Method. This method is found better than previous methods [6]. Chen and Li (2006, 2007) [8, 9] has implemented the RFM and named as improved exponential Regula-Falsi method for solving non-linear equations. In their modified method, they used an exponential iterative methods accelerating after applying classical RFM. This new modified method has asymptotic quadratic convergence. Gottlieb and Thompson (2010) [10] presented a powerful algorithm for finding the root that use direct quadratic interpolation in place of inverse quadratic interpolation. One of the major finding of their proposed new algorithm is that it rapidly converges without checking the sign of the function. This method of root finding has named as Bisected direct quadratic Regula Falsi (BDQRF) method. They have also discussed about the Inverse Quadratic Fit-Brent's method.

Muller's proposed a new algorithm based on three initial guesses. Muller's method gives rapid convergence without good initial values [13]. These three initial guesses are interpolated to compute the next value. One of the main advantages of Muller's method that it is useful to find the real as well as complex roots. An improved version of Bisection method and Muller’s with global and asymptotic convergence of non-linear equations was given by Xinyuan Wu (2005). Their new proposed algorithm is more efficient than other previous methods. The modification in Muller's method by incorporating Bisection property, they obtained nice property of convergence [2].

Thus previously published works implemented Regula-Falsi method in several ways to obtain better convergence. It is found that if standard Regula-Falsi method is hybrid with interpolation based Muller's method then the resultant root is more appropriate than these two methods. Therefore, in the present work Regual-Falsi method is hybrid with interpolation based Muller's method to compute the real root.

2. New algorithm

Consider a continuous function $f(x) = 0$. Suppose that x_0, x_1 and x_2 are the initial approximations for a solution of $f(x) = 0$ such that $f(x_i)\, f(x_j) < 0$ for any i, j in $\{0, 1, 2\}$ $i \neq j$. Using Muller's method, the next approximation x_3 begins by considering the quadratic polynomial

$$P(x) = a(x - x_2)^2 + b(x - x_2) + c \tag{1}$$

that passes through $(x_0, p(x_0))$, $(x_1, p(x_1))$ and $(x_2, p(x_2))$. The constants a, b and c can be determined from the conditions

$$\begin{aligned} f(x_0) &= a(x_0 - x_2)^2 + b(x_0 - x_2) + c \\ f(x_1) &= a(x_1 - x_2)^2 + b(x_1 - x_2) + c \\ f(x_2) &= c \end{aligned}$$

To determine x_3, the root of $P(x) = 0$, we apply the quadratic formula to $P(x)$, i.e.

$$x_3 - x_2 = \frac{-2c}{b \pm \sqrt{b^2 - 4ac}} \tag{2}$$

This gives two possibilities for x_3, depending on the sign preceding the radical term. In Muller's method, the sign is chosen to agree with the sign of b. Chosen in this manner, the denominator will be the largest in magnitude, which avoids the possibility of subtracting nearly equal numbers and results in x_3 being selected as the closest root of $P(x) = 0$ to x_2.

If the points x_0, x_1 and x_2 are such that any two of them having their function values in opposite sign, say x_0 and x_1 are such that $f(x_0)\, f(x_1) < 0$, then from Regula-Falsi method, the first approximate root, x_3, can be calculated by using the formula

$$x_3 = \frac{x_0 f(x_1) - x_1 f(x_0)}{f(x_1) - f(x_0)} \tag{3}$$

and, the first approximate root by using Muller's method as given in equation (2) is

$$x_3 = x_2 - \frac{2c}{b \pm \sqrt{b^2 - 4ac}}, \tag{4}$$

where,

$$a = \frac{(x_1 - x_2)[f(x_0) - f(x_2)] - (x_0 - x_2)[f(x_1) - f(x_2)]}{(x_0 - x_2)(x_1 - x_2)(x_0 - x_1)}$$

$$b = \frac{(x_1 - x_2)^2[f(x_1) - f(x_2)] - (x_1 - x_2)^2[f(x_0) - f(x_2)]}{(x_0 - x_2)(x_1 - x_2)(x_0 - x_1)}$$

$$c = f(x_2)$$

Now in the present proposed algorithm, we take the average of the iterations in equations (3) and (4) as our first approximate root $\hat{x}$. If the functional signs of initial approximations are having same sign, then repeat Muller method.

The generalization of this process is described in the following section.

2.1 Formulation of the proposed Algorithm

Recall the equations (3) and (4) in terms of iteration formulae in terms of x_{n-1}, x_n and x_{n+1} such that $f(x_{n-1})$, $f(x_n)$ and $f(x_{n+1})$ any pair of them having opposite signs. Without loss of generality, suppose $f(x_n)$ and $f(x_{n+1})$ are in opposite signs. Now by Muller's method

$$x_{n+2} = x_{n+1} - \frac{2c}{b \pm \sqrt{b^2 - 4ac}} \tag{5}$$

where a, b and c are as in equation (4); and by Regula-Falsi method

$$x_{n+2} = \frac{x_n f(x_{n+1}) - x_{n+1} f(x_n)}{f(x_{n+1}) - f(x_n)} \tag{6}$$

Now the average of equations (5) and (6) is

$$x_{n+2} = \frac{1}{2}\left[x_{n+1} - \frac{2c}{b \pm \sqrt{b^2 - 4ac}} + \frac{x_n f(x_{n+1}) - x_{n+1} f(x_n)}{f(x_{n+1}) - f(x_n)} \right] \tag{7}$$

Now by replacing the unused approximate point in Regula-Falsi method of three initial approximations i.e., x_{n-1} by the new approximation x_{n+2} in initial approximations and repeat the process until we get desired accuracy.

Flow chat of the proposed algorithm is presented in Figure 1.

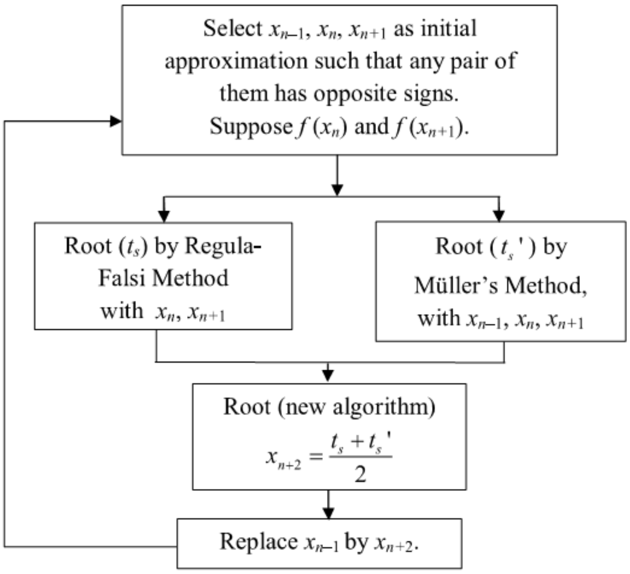

Figure 1: Flow Chart for Proposed Algorithm

2.2 Steps for Calculating Root

The working steps for the proposed algorithm.

1. Choose any three values x_0, x_1 and x_2 as initial approximations such that any pair of them has opposite signs, say x_1 and x_2.
2. Calculate the approximate root by Muller's method as in equation (5) and find approximate root between x_1 and x_2 by Regula-Falsi method as in equation (6).
3. Take the average of these two approximate roots as our first approximation, say x_3.
4. Replace the unused point (in Regula-Falsi method) x_0 by x_3 in Step-1 and repeat the process until, we get desired approximation.

2.3 Implementation of proposed method in Matlab

In this section, we provide the implementation of the proposed method in Matlab code similar to Regula-Falsi method in [14] by creating a function **`MullerFalsePosition`** (*f*, *p0*, *p1*, *p2*, *esp*, *n*)}, as given below, where *f* is given transcendental function, *p0*, *p1*, and *p2* are the initial approximation of the root, *esp* is the relative error and n is the number of iterations required.

```
function root = MullerFalsePosition(f,p0,p1,p2,esp,n)
iter = 0;ea = 0;pn = 0;
disp('  ------------------------------------');
disp('  No            Root          f(Root)   ');
disp('  ------------------------------------');
while (1)
       a=p0;b=p1;c=p2;
       if f(p0)*f(p1)<0
              p0=p2;p1=a;p2=b;
              p0=p0;p1=p1;p2=p2;
              elseif f(p2)*f(p0)<0
              p0=p1;p1=a;p2=c;
       end
       pnold = pn;
f0 = f(p0);f1 = f(p1); f2 = f(p2);
c = f2;
b =((p0-p2)^2*(f1-f2)-(p1-p2)^2*(f0-f2))/((p0-p2)*(p1-p2)*(p0-p1));
```

```
a = ((p1-p2)*(f0-f2)-(p0-p2)*(f1-f2))/((p0-p2)*(p1-p2)*(p0-p1));
pnM = p2-(2*c)/(b+sign(b)*sqrt(b$^2$ 2-4*a*c));
pnRF =(p1*f2-p2*f1)/(f2-f1);
pn = (pnM + pnRF)/2;
disp(sprintf('\%3d \%10.5f \%15.5e', iter+1, pn, f(pn)));
iter = iter + 1;
pnnew = pn;
ea = abs((pnnew - pnold)/pnnew) * 100;
p0 = p1; p1 = pn; p2 = p2;
if ea <= esp | iter >= n, break, end
end
disp('-----------------------');
disp(['Given function f(x) = ' char(f)]);
disp(sprintf('Approximate root = \%f',pn));
```

Section -3 has been included to verify the proposed algorithm.

3: Numerical Examples

This section provides two real life examples to show that the proposed algorithm is more efficient than other existing methods. In the following examples, MM, RF and PM denote Muller's, Regula-Falsi and Proposed methods respectively.

Example 3.1: Consider the polynomial equation of the form

$$f(x)=16x^4-40x^3+5x^2+20x+6 \tag{8}$$

In [13, p.71] author has considered the polynomial in equation (8) for x_0 = 0.5, x_1 = 1.0, x_2 = 1.5 with accuracy tolerance 10^{-5}. We use the same values using proposed method to show the efficiency of proposed method.

Table 1: Comparison of PM with MM and RF

Ite. No.	PM	$\|f(x_n)\|$	MM	$\|f(x_n)\|$	RF	$\|f(x_n)\|$
1	1.27120	8.83×10^{-1}	1.28785	1.370	1.16250	2.387
2	1.23990	5.34×10^{-2}	1.23746	0.126	1.25068	2.70×10^{-1}
3	1.24169	3.65×10^{-4}	1.24160	0.22×10^{-2}	1.24171	9.45×10^{-4}
4	1.24168	2.97×10^{-8}	1.24168	2.58×10^{-5}	1.24168	2.44×10^{-6}
5	1.24168	3.45×10^{-12}	1.24168	2.57×10^{-5}	1.24168	6.31×10^{-9}

In fourth iteration (Table 1), the proposed method computes its approximate root and corresponding function as 1.24168 and 2.97×10^{-8} respectively. This corresponding function value is more closure to zero than 2.58×10^{-5} and 2.44×10^{-6} obtained from Muller's [13] and Regula-Falsi methods respectively. Therefore proposed method is more accurate than the standard Muller's [13] and Regula-Falsi methods. Hence, the proposed method is not only compute guaranteed root but also take less computational efforts for solving the real life problems.

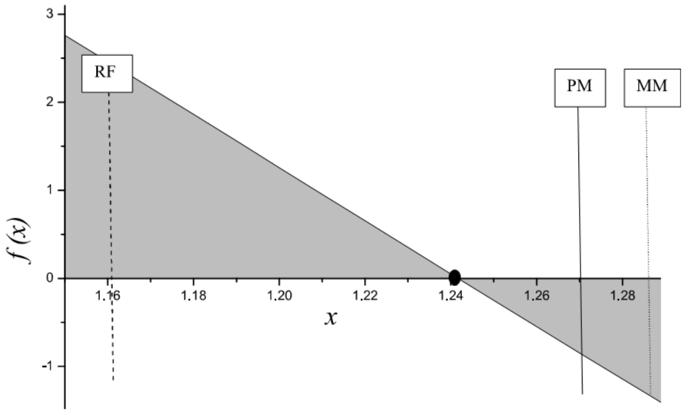

Figure 2: Graphical representation of first iteration

In Figure-2, it is found that the first approximate root computed by proposed method (PM) is closer to exact root that other two methods (RF and MM).

Example 3.2: In this example, we compare different functions and their real roots occur as given in Table-2. These functions are taken from existing literature [15].

Table 2: Comparison of different methods for non-linear transcendental functions

Function f **(x)**	**Initial Approx.**	**Approx.** $\lvert f(x_n)\rvert$	**Exact Root**	**MM No. Ite.**	**RF No. Ite.**	**PM No. Ite.**
$e^x - 2x - 1$	1,2,3	10^{-6}	1.25643	5	17	4
ln (1 + x)	-0.5,0,1	10^{-6}	0	Fails	9	6
sin(x) – cos (x)	0,1,2	10^{-6}	0.785398	4	3	3

In the first function i.e., $e^x - 2x - 1$ is converged in fifth, seventeen and fourth iterations using Muller's [13], Regula-Falsi methods and Proposed algorithm respectively. Which indicate that the proposed method involves less computational efforts.

For the second function ln(1+x), Muller's [13] fails to compute the root with initial approximation -0.5,0 and 1. However the proposed method are able to compute the root in the sixth iteration.

In the last function sin(x) – cos (x) [13], Regula-Falsi methods and Proposed algorithm are computed the roots in 4th, 3rd and 3rd iteration respectively.

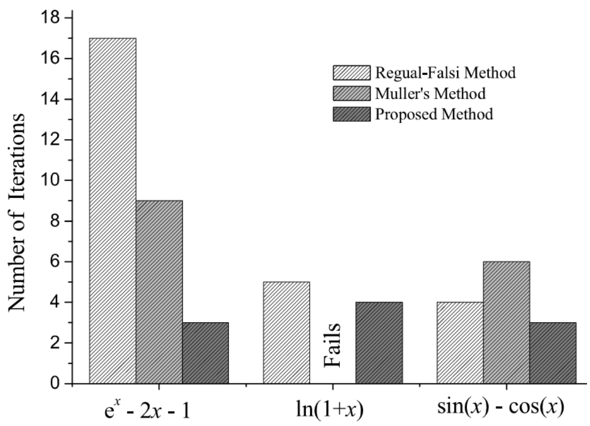

Figure 3: Graphical representation of first iteration

Figure 3 shows the number of iterations for different methods. It is found the proposed methods takes minimum number of iterations in comparison of other methods for each case.

Thus, From Table-2 and Fig.3, it is clear that proposed method is more efficient than other two methods.

Conclusion

The aim of the present work is to provide new algorithms which compute the approximate root of non-linear transcendental function. In this paper, Interpolation technique is hybrid with Standard Regula-Falsi method and obtained better convergence with guaranteed results over existing methods. Thus, the proposed algorithms will help to compute the root of real life problem with less computational efforts.

[1]. Sharma J.R. and Goyal R.K. Fourth order derivative methods for solving non-linear equations. International Journal of Computer Mathematics. Vol.83. No.1, 101-106, 2006.

[2]. Gottlieb R.G. and Thompson B.F. Bisected direct quadratic Regula-Falsi.Applied Mathematical Sciences. Vol.4,No.15,709-718, 2010.

[3]. Wu Xinyuan, Wu Hongwei. On a class of quadratic convergence iteration formula without derivatives. Applied Mathematics and Computation Vol.107, 77-80, 2000.

[4]. Wu Xinyuan, Shen Zuhe, Xia Jianlin. An improved regula falsi method with quadratic convergence of both diameter and point for enclosing simple zeros of non-linear equations. Vol.144, 381-388, 2003.

[5]. Mamta V. K.,Kukreja V.K., Singh S.On some third-order iterative methods for solving nonlinear equations. Applied Mathematics and Computation Vol.171, 272-280, 2005.

[6]. Noor M.A. and Ahmad F. Numerical comparison of iterative methods for solving nonlinear equations. Applied Mathematics and Computation Vol.180, 167-172, 2006.

[7]. Noor M.A., Noor K.I., Khan W.A.,Ahmad F. On iterative methods for nonlinear equations.Applied Mathematics and Computation Vol.183, 128-133, 2006.

[8]. Chen J. and Li W. An exponential regula falsi method for solving nonlinear equations. Numerical Algorithms, Vol.41, 327-338, 2006.

[9]. Chen J. and Li W.An improved exponential regula falsi methods with quadratic convergence of both diameter and point for solving nonlinear equations.Applied Numerical Mathematics Vol.57, 80-88, 2007.

[10]. Gottlieb R.G. and Thompson B.F. Bisected direct quadratic Regula-Falsi.Applied Mathematical Sciences. Vol.4,No.15,709-718, 2010.

[11]. Dowell-Jarratt. A modified Regula-Falsi method for computing the real root of an equation. BIT Numerical Mathematics. Vol.11, 168-174, 1971.

[12]. Mamta V. K.,Kukreja V.K., Singh S.On a class of quadratically convergent iteration formulae. Applied Mathematics and Computation Vol.166, 633-637, 2005.

[13]. Faires and Burden: Numerical methods, Third edition, Books cole publications; 2002.

[14]. Steven C. Chapra: Solutions Manual to accompany, Applied Numerical Methods With MATLAB for Engineers and Scientists, Mc Graw Hill Higher education.

[15]. Wu Xinyuan. Improved Muller method and Bisection method with global and asymptotic superlinear convergence of both point and interval for solving nonlinear equations. Applied Mathematics and Computation Vol. 166, 299-311, 2005.