Yet Another Modified Newton Method *

M. Imran

mimran@unri.ac.id

Laboratorium Matematika Terapan Jurusan Matematika
Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Riau
Kampus Binawidya Pekanbaru (28293)

Abstract

We discuss modified Newton methods, by looking at an approximation error of a numerical approximation on an integral, and propose a modification using midpoint rule. Comparison among the discussed methods is also given by considering computational costs.

Keywords: *Modified Newton's Method, Midpoint rule, trapeziodal rule*

1 Introduction

Finding a root of a nonlinear equation in one variable, $f(x) = 0$, is a familiar topic included in a numerical analysis course, since these types of problems raised in many field. Many methods have been developed to tackle this type of problem. The developments were done by modifying existing methods, such as [3] [5] [6] [8], or by introducing a new method which have the same characteristics with the old methods, such as [1]. The aim of all developments is to find a method which shall convergent faster than the old ones, and is also reliable.

Among the found methods, Newton's method is the most fomous one, which is quadraticly convergent. Many authors are interested in modifying this method to obtain a higher order method. The first third order method resulted on the modifying this method appeared in Wall [9].

Hasanov et al. [5] have suggested an improvement to the iteration of Newton's method. They have approximated the indefinite integral using Simpson formula instead of rectangle of the left Reimann sum. Their idea follows what have been done by Weerakoon and Fernando [8], where they have approximated the indefinite integral using trapezoidal rule. Both of these new improvements have a third order convergence.

In this study we suggest a modification of the iteration of Newton's method by approximating the indefinite integral using a midpoint rule. The modified method need one functional and two first derivative evaluations.

*Presented at SEMIRATA BKS PTN Wilayah Barat, held at the University of Syarif Hidayatullah in Jakarta, July 9-10, 2007

2 Some Modified Newton's Methods

Newton's method (NM) for computing the root α of the nonlinear equation $f(x) = 0$ is to start with initial estimate x_0 sufficiently close to the root α and to use the one point iteration

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

where x_n is the n-th aproximation of α. We may also view x_{n+1} as the root of the two-term Taylor expansion or linear model of f about x_n,[7],

$$M(x) = f(x_n) + f'(x_n)(x - x_n). \tag{2}$$

By integrating by part this local model can be viewed as the following obivious identity,[4],

$$f(x) = f(x_n) + \int_{x_n}^{x} f'(s)ds \tag{3}$$

Newton approximates $\int_{x_n}^{x} f'(s)ds$ in (3) using the left Reimann-sum for one interval, resulted in

$$\int_{x_n}^{x} f'(s)ds \approx f'(x_n)(x - x_n), \tag{4}$$

which can be visualized as Figure 1(a). On substituting (4) into (3), setting $f(x) = 0$ rearranging the terms of the resulting equation, we end up with the equation (1).

Weerakoon and Fernando [8] approximate the indefinite integral involved in (3) by the trapezoidal rule, see Figure 1.(b),

$$\int_{x_n}^{x} f'(s)ds \approx \Big(\frac{f'(x_n) + f'(x)}{2}\Big)(x - x_n), \tag{5}$$

and then by some algebra they end up with the following scheme (TNM)

$$x_{n+1} = x_n - \frac{2f(x_n)}{f'(x_n) + f'(x^*_{n+1})} \tag{6}$$

$$x^*_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} \tag{7}$$

They prove that the scheme is a third order convergence. The way they choose x^*_{n+1} as in (7) was introduce for the first time by Wall in [9].

Following the Weerakoon's and Fernando's idea, Hasanov et al. [5] approximate the indefinite integral involved in (3) by Simpson rule, see Figure 1(c). They obtain the following scheme(SNM)

$$x_{n+1} = x_n - \frac{6f(x_n)}{f'(x^*_{n+1}) + 4f'(x^{**}_{n+1}) + f'(x_n)} \tag{8}$$

$$x^*_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} \tag{9}$$

$$x^{**}_{n+1} = x_n - \frac{f(x_n)}{2f'(x_n)} \tag{10}$$

They prove that this method is third order convergence.

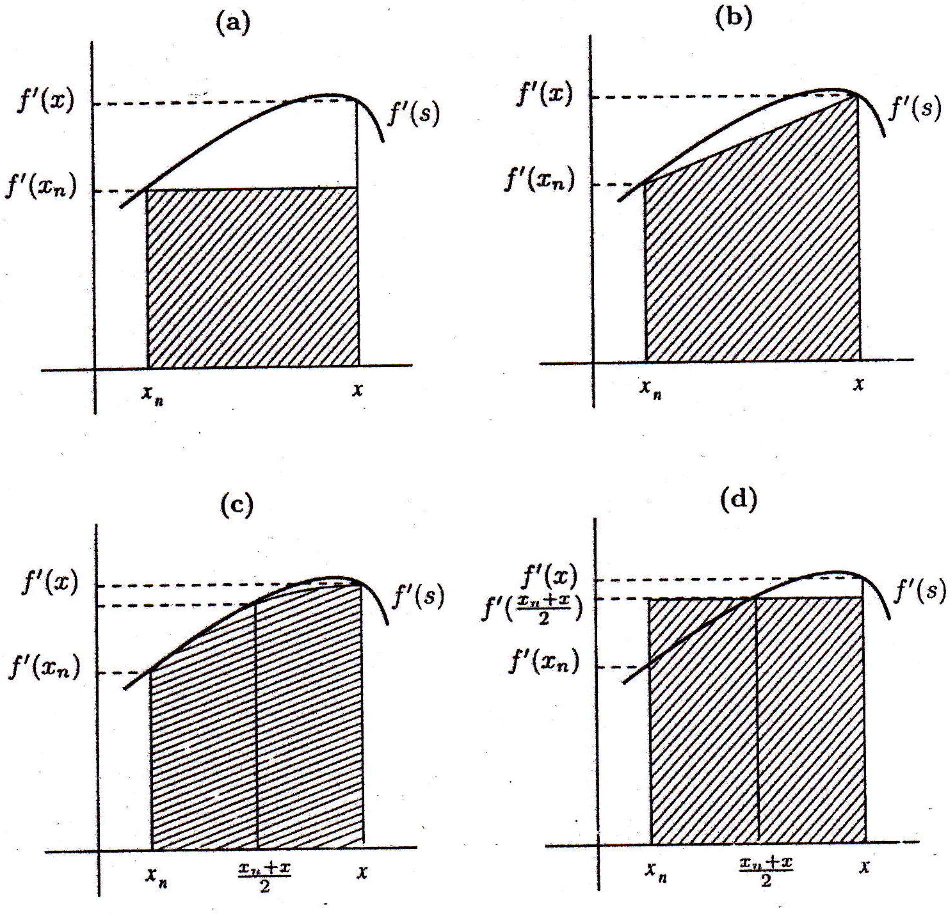

Figure 1: Graph of approximation of $\int_{x_n}^{x} f'(s)ds$ using (a) left Reimann-sum, (b) Trapezoidal rule, (c) Simpson rule, and (d) Midpoint rule

Now we look at the approximation error of numerical methods for an approximation of an integral of a function f which is smooth enough. The approximation error of Trapezoidal rule is given by

$$E_T := \int_a^b f(x)dx - \frac{h}{2}(f(a) + f(b))) = -\frac{(b-a)^3}{12} f''(\xi), \tag{11}$$

where $\xi \in (a,b)$ and $f \in C^2[a,b]$. The approximation error of Simpson rule is given by

$$E_S := \int_a^b f(x)dx = \frac{h}{3}(f(a) + 4f(\frac{a+b}{2}) + f(b)) = -\frac{(b-a)^5}{90} f^{(4)}(\xi), \tag{12}$$

where $\xi \in (a,b)$ and $f \in C^4[a,b]$, and the approximation error of midpoint rule

$$E_M := \int_a^b f(x)dx - (b-a)f(\frac{a+b}{2}) = \frac{(b-a)^3}{24} f''(\xi), \tag{13}$$

where $\xi \in (a,b)$ and $f \in C^2[a,b]$,[2]

We see that the approximation error of the midpoint rule needs the same smoothness as the trapezoidal rule. We only require the second derivative of f instead of the fourth derivative of f for Simsons rule. The absolute value of midpoint approximation error is slightly smaller than the trapezoidal rule, by comparing the constant in front of the derivative of f. From this view we may use the midpoint rule to approximate the indefinite integral involved (3), see Figure 1(d), that is

$$\int_{x_n}^{n} f'(s)ds \approx (x - x_n) f\Big(\frac{x + x_n}{2}\Big) \tag{14}$$

On substituting (14) into (3) we obtain the local model

$$\widehat{M}(x) = f(x_n) + (x - x_n) f\Big(\frac{x + x_n}{2}\Big). \tag{15}$$

We take the next iterative point as the root of local model (15),

$$\widehat{M}(x_{n+1}) = 0,$$

and by arranging resulting equation we have

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(\frac{x + x_n}{2})}. \tag{16}$$

Now, let $x^*_{n+1} = \frac{x_{n+1} + x_n}{2}$ in (16), then approximate x_{n+1} using Newton's method (1) , i.e.

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

Hence

$$x^*_{n+1} = \frac{1}{2}\left[x_n - \frac{f(x_n)}{f'(x_n)} + x_n\right]$$

$$x^*_{n+1} = x_n - \frac{f(x_n)}{2f'(x_n)}. \tag{17}$$

Then we propose the scheme (MNM)

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x^*_{n+1})} \tag{18}$$

where x^*_{n+1} is given by (17).

3 Comparisons

3.1 Analytical Comparisons

For analytical comparisons we look into computational costs of the methods. From Table 1, we see that all methods need only one function evaluation. The modified methods, TNM and MNM, need an expense in the derivarive of f. From this point, we can say that TNM and MNM are comparable in the functional evaluation. However if the cost of an addition is comparable with a multiplication the computational cost of MNM is fewer than TNM.

Table 1: Comparisons of the computational costs of the modified methods

Method	Cost of			
	addition/subtraction	multiplication/division	$f(x)$	$f'(x)$
NM	1	1	1	1
TNM	3	3	1	2
SNM	4	5	1	3
MNM	2	3	1	2

Table 2: Comparisons of the number of iterations of the modified methods

$f(x)$	x_0	Number of Iterations				$Time_{30000}$			
		NM	TNM	SNM	MNM	NM	TNM	SNM	MNM
$\cos(x) - x$	2.00	5	5	4	4	0.886	0.942	1.006	0.820
	3.00	8	10	5	5	0.982	1.592	1.207	0.847
	1.70	6	4	4	4	0.899	0.894	1.002	0.777
	-1.00	9	4	5	7	1.069	0.961	1.229	0.966
	4.00	ND	8	5	5	ND	1.316	1.22	0.804
$(x - 1.0)^3 - 1$	2.50	7	5	5	5	0.554	0.648	0.553	0.541
	4.00	9	6	6	6	0.637	0.600	0.57	0.601
	-0.50	17	17	7	6	0.791	0.775	0.642	0.553
	-1.00	11	9	6	7	0.591	0.677	0.609	0.598
	-2.00	12	9	8	8	0.625	0.651	0.616	0.582
$xe^{(x^2)} - \sin^2(x) + 3\cos(x) + 5.0$	-3.00	15	10	10	10	2.963	3.594	5.714	3.368
	1.20	ND	21	8	39	ND	6.516	4.795	10.861
$e^{(x^2+7*x-30)} - 1$	3.30	10	7	7	7	0.982	1.001	1.261	0.936
	3.50	13	9	9	8	1.196	1.316	1.496	1.038

3.2 Numerical Experiments

The modified methods and Newton's method are tested using some functions and initial points, which have used [8] and [5]. We compare the CPU time, by running the program for 30000 times. We stop the program using the following criteria

$$\frac{|x_{n+1} - x_n|}{|x_{n+1}|} < \epsilon,$$
$$|f(x_{n+1})| < \epsilon,$$

where $\epsilon = 1.0842e - 19$, machine epsilon. All programs are writen on C-languange and run on Windows PC with Intel Processor at 2.4 GHz. The computational resuls are given in Table 2. From the Table we see that the results or our propose method (MNM) are comparable with those of SNM, except at the starting point $x_0 = 1.2$ for $f(x) = x\exp(x^2) - \sin^2(x) + 3\cos(x) + 5.0$, where MNM method needs 39 number of iterations. Almost in all computations, MNM method requires fewer time than all the method we compare here. This matchs what we expect from Table 1.

In this presentation, we do not touch the convergence of the propose method, and this subjects will be presented in the future articles.

References

[1] Abu-Alshaikh, I. 2005. *A New Iterative Method for Solving Nonlinear Equations.* Enformatika. **5**:190–193.

[2] Atkinson, K. E. 1989. **An Introduction to Numerical Analysis**. Second Edition. John Wiley & Son, New York.

[3] Gerlach, J. 1994. *Accelerate Convergence in Newton's Method.* Siam Review. **36**(2): 272–276.

[4] Hamming, R. H. 1973. **Numerical Method for Scientists and Engineers**. McGraw-Hill Inc. New York. Republished by Dover, New York.

[5] Hasanov, V.I., Ivanov, I. G. & Nedjibov, G. 2005. *A new modification of Newton's Method.* preprint laboratorium of Mathematical Modelling, Shoumen university, Bulgaria.

[6] Kanwar, V. Sharma, J. R. & Mamta. 2005. *A new family of Secant-like method with superlinear convergence.* Applied Mathematics and Computation. **171**:104–107.

[7] Kelley, C. T. 1995. **Iterative Methods for Linear and Nonlinear Equations**. Frontier in Applied Mathematics 16. SIAM, Philadelphia.

[8] Weerakoon, S. & Fernando, T. G. I. 2000. *Avariant if Newton's Method With Accelerated Third-Order Convergence.* Applied Mathematics Letters. **13**: 87–93.

[9] Wall, H. S. 1948. *A Modification of Newton's Method.* Amer. Math. Monthly. **55**: 90–94.