PERGAMON

Applied Mathematics Letters 13 (2000) 87–93

Applied Mathematics Letters

www.elsevier.nl/locate/aml

A Variant of Newton's Method with Accelerated Third-Order Convergence

S. WEERAKOON AND T. G. I. FERNANDO
Department of Mathematics, University of Sri Jayewardenepura
Gangodawila, Nugegoda, Sri Lanka
sweera@sjp.ac.lk

(Received July 1998; revised and accepted November 1998)

Abstract—In the given method, we suggest an improvement to the iteration of Newton's method. Derivation of Newton's method involves an indefinite integral of the derivative of the function, and the relevant area is approximated by a rectangle. In the proposed scheme, we approximate this indefinite integral by a trapezoid instead of a rectangle, thereby reducing the error in the approximation. It is shown that the order of convergence of the new method is three, and computed results support this theory. Even though we have shown that the order of convergence is three, in several cases, computational order of convergence is even higher. For most of the functions we tested, the order of convergence in Newton's method was less than two and for our method, it was always close to three.

Keywords—Newton's formula, Nonlinear equations, Iterative methods, Order of convergence, Function evaluations.

1. INTRODUCTION

Newton's method that approximates the root of a nonlinear equation in one variable using the value of the function and its derivative, in an iterative fashion, is probably the best known and most widely used algorithm, and it converges to the root quadratically. In other words, after some iterations, the process doubles the number of correct decimal places or significant digits at each iteration.

In this study, we suggest an improvement to the iteration of Newton's method at the expense of one additional first derivative evaluation. Derivation of Newton's method involves an indefinite integral of the derivative of the function, and the relevant area is approximated by a rectangle. Here, we approximate this indefinite integral by a trapezoid instead of a rectangle, and the result is a method with third-order convergence.

It is shown that the suggested method converges to the root, and the order of convergence is at least three in a neighbourhood of the root, whenever the first and higher order derivatives of the function exist in a neighbourhood of the root; i.e., our method approximately triples the number of significant digits after some iterations. Computed results overwhelmingly support this theory, and the computational order of convergence is even more than three for certain functions.

PII: S0893-9659(00)00100-2

2. PRELIMINARY RESULTS

DEFINITION 2.1. *(See [1].) Let $\alpha \in \Re$, $x_n \in \Re$, $n = 0, 1, 2, \dots$. Then, the sequence $\{x_n\}$ is said to converge to α if*

$$\lim_{n \to \infty} |x_n - \alpha| = 0.$$

If, in addition, there exists a constant $c \geq 0$, an integer $n_0 \geq 0$, and $p \geq 0$ such that for all $n > n_0$,

$$|x_{n+1} - \alpha| \leq c|x_n - \alpha|^p, \tag{2.1}$$

then $\{x_n\}$ is said to converge to α with q-order at least p. If $p = 2$ or 3, the convergence is said to be q-quadratic or q-cubic, respectively.

When $e_n = x_n - \alpha$ is the error in the n^{th} iterate, the relation

$$e_{n+1} = ce_n^p + O\left(e_n^{p+1}\right) \tag{2.2}$$

is called the error equation. By substituting $e_n = x_n - \alpha$ for all n in any iterative method and simplifying, we obtain the error equation for that method. The value of p thus obtained is called the order of this method.

DEFINITION 2.2. *Let α be a root of the function $f(x)$ and suppose that x_{n+1}, x_n, and x_{n-1} are three consecutive iterations closer to the root α. Then, the computational order of convergence ρ can be approximated using the formula*

$$\rho \approx \frac{\ln |(x_{n+1} - \alpha) / (x_n - \alpha)|}{\ln |(x_n - \alpha) / (x_{n-1} - \alpha)|}.$$

STOPPING CRITERIA. We have to accept an approximate solution rather than the exact root, depending on the precision (ε) of the computer. So, we use the following stopping criteria for computer programs:

(i) $|x_{n+1} - x_n| < \sqrt{\varepsilon}$;
(ii) $|f(x_{n+1})| < \sqrt{\varepsilon}$.

3. NUMERICAL SCHEMES

3.1. Newton's Method (NM)

Newton's algorithm to approximate the root α of the nonlinear equation $f(x) = 0$ is to start with an initial approximation x_0^* sufficiently close to α and to use the one point iteration scheme

$$x_{n+1}^* = x_n^* - \frac{f(x_n^*)}{f'(x_n^*)}, \tag{3.1}$$

where x_n^* is the n^{th} iterate. It is well known that Newton's method as given above is quadratically convergent.

It is important to understand how Newton's method is constructed. At each iterative step we construct a local linear model of our function $f(x)$ at the point x_n^* and solve for the root (x_{n+1}^*) of the local model. In Newton's method (Figure 1), this local linear model is the tangent drawn to the function $f(x)$ at the current point x_n^*.

The local linear model at x_n^* is

$$M_n(x) = f(x_n^*) + f'(x_n^*)(x - x_n^*). \tag{3.2}$$

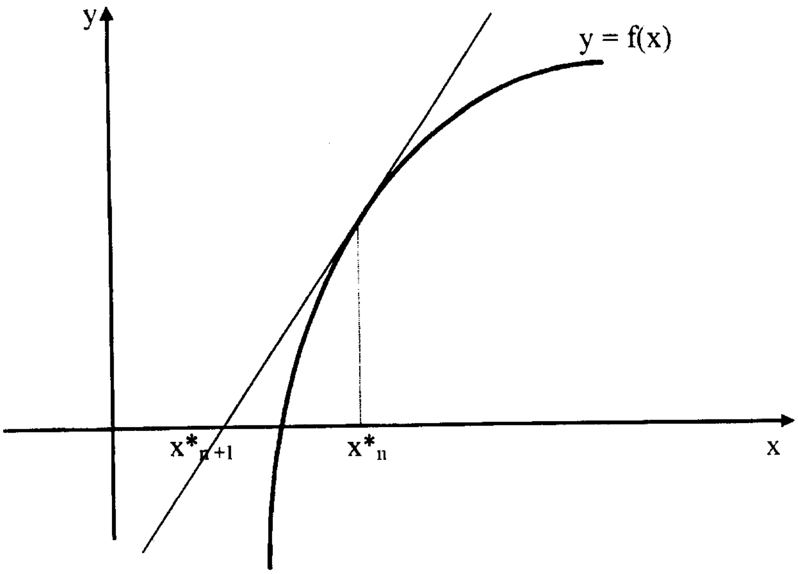

Figure 1. Newton's iterative step.

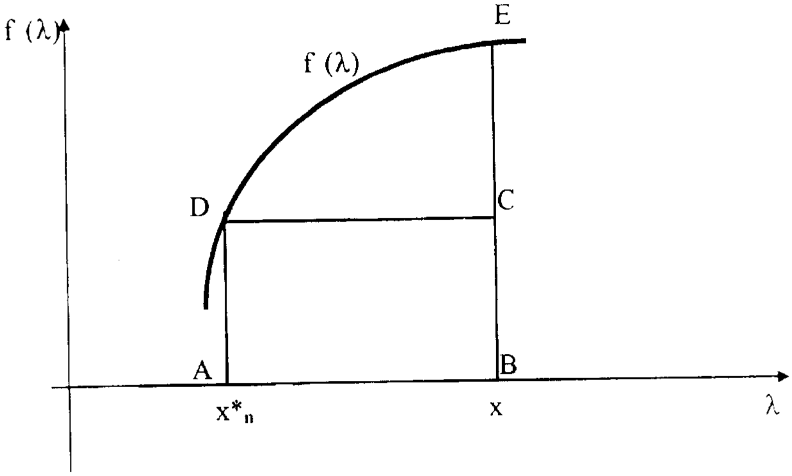

Figure 2. Approximating the area by the rectangle $ABCD$.

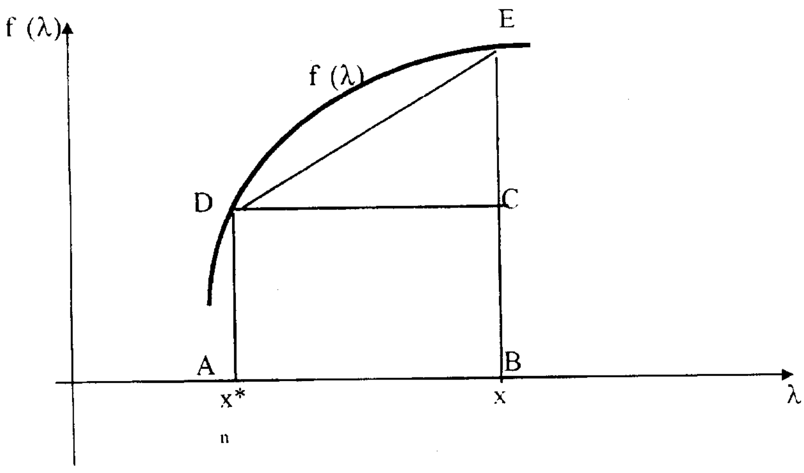

Figure 3. Approximating the area by the trapezoid $ABED$.

This local linear model can be interpreted [1] in another way. From Newton's theorem,

$$f(x) = f\left(x_n^*\right) + \int_{x_n^*}^{x} f'(\lambda)\, d\lambda. \tag{3.3}$$

In Newton's method, the indefinite integral involved in (3.3) is approximated by the rectangle $ABCD$ (Figure 2), i.e.,

$$\int_{x_n^*}^{x} f'(\lambda)\, d\lambda \approx f'\left(x_n^*\right)\left(x - x_n^*\right), \tag{3.4}$$

which will result in the model given in (3.2).

3.2. A Variant of Newton's Method (VNM)

From Newton's theorem,

$$f(x) = f\left(x_n\right) + \int_{x_n}^{x} f'(\lambda)\, d\lambda. \tag{3.5}$$

In the proposed scheme, we approximate the indefinite integral involved in (3.5) by the trapezium $ABED$ (Figure 3), i.e.,

$$\int_{x_n}^{x} f'(\lambda)\, d\lambda \approx \left(\frac{1}{2}\right)\left(x - x_n\right)\left[f'\left(x_n\right) + f'(x)\right]. \tag{3.6}$$

Thus, the local model equivalent to (3.2) is

$$\hat{M}_n(x) = f(x_n) + \left(\frac{1}{2}\right)(x - x_n)[f'(x_n) + f'(x)]. \tag{3.7}$$

Note that not only the model and the derivative of the model agree with the function $f(x)$ and the derivative of the function $f'(x)$, respectively, but also the second derivative of the model and the second derivative of the function agree at the current iterate $x = x_n$. Even though the model for Newton's method matches with the values of the slope $f'(x_n)$ of the function, it does not match with its curvature in terms of $f''(x_n)$.

We take the next iterative point as the root of the local model (3.7)

$$\begin{aligned} \hat{M}_n(x_{n+1}) &= 0, \qquad \text{i.e.,} \\ &\Rightarrow f(x_n) + \left(\frac{1}{2}\right)(x_{n+1} - x_n)\left[f'(x_n) + f'(x_{n+1})\right] = 0 \\ &\Rightarrow x_{n+1} = x_n - \frac{2f(x_n)}{[f'(x_n) + f'(x_{n+1})]}. \end{aligned}$$

Obviously, this is an implicit scheme, which requires having the derivative of the function at the $(n+1)^{\text{th}}$ iterative step to calculate the $(n+1)^{\text{th}}$ iterate itself. We could overcome this difficulty by making use of Newton's iterative step to compute the $(n+1)^{\text{th}}$ iterate on the right-hand side.

Thus, the resulting new scheme is

$$x_{n+1} = x_n - \frac{2f(x_n)}{\left[f'(x_n) + f'\left(x_{n+1}^*\right)\right]}, \quad n = 0, 1, 2, \ldots, \qquad \text{where } x_{n+1}^* = x_n - \frac{f(x_n)}{f'(x_n)}. \tag{3.8}$$

4. ANALYSIS OF CONVERGENCE

THEOREM 4.1. *Let $f : D \to \Re$ for an open interval D. Assume that f has first, second, and third derivatives in the interval D. If $f(x)$ has a simple root at $\alpha \in D$ and x_0 is sufficiently close to α, then the new method defined by (3.8) satisfies the following error equation:*

$$\mathbf{e}_{n+1} = \left(\mathbf{C}_2^2 + \frac{1}{2}\mathbf{C}_3\right)\mathbf{e}_n^3 + \mathbf{O}\left(\mathbf{e}_n^4\right), \tag{4.1}$$

where $e_n = x_n - \alpha$ and $C_j = (1/j!) f^{(j)}(\alpha)/f^{(1)}(\alpha)$, $j = 1, 2, 3, \ldots$.

PROOF. The suggested variant of Newton's method (VNM) is

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

where $x_{n+1}^* = x_n - f(x_n)/f'(x_n)$. Let α be a simple root of $f(x)$ (i.e., $f(\alpha) = 0$ and $f'(\alpha) \neq 0$) and $x_n = \alpha + e_n$. We use the following Taylor expansions:

$$\begin{aligned} f(x_n) = f(\alpha + e_n) &= f(\alpha) + f^{(1)}(\alpha)e_n + \frac{1}{2!}f^{(2)}(\alpha)e_n^2 + \frac{1}{3!}f^{(3)}(\alpha)e_n^3 + O\left(e_n^4\right) \\ &= f^{(1)}(\alpha)\left[e_n + \frac{1}{2!}\frac{f^{(2)}(\alpha)e_n^2}{f^{(1)}(\alpha)} + \frac{1}{3!}\frac{f^{(3)}(\alpha)e_n^3}{f^{(1)}(\alpha)} + O\left(e_n^4\right)\right] \\ &= f^{(1)}(\alpha)\left[e_n + C_2e_n^2 + C_3e_n^3 + O\left(e_n^4\right)\right], \end{aligned} \tag{4.2}$$

where $C_j = (1/j!)f^{(j)}(\alpha)/f^{(1)}(\alpha)$. Furthermore, we have

$$\begin{aligned} f^{(1)}(x_n) = f^{(1)}(\alpha + e_n) &= f^{(1)}(\alpha) + f^{(2)}(\alpha)e_n + \frac{1}{2!}f^{(3)}(\alpha)e_n^2 + O\left(e_n^3\right) \\ &= f^{(1)}(\alpha)\left[1 + \frac{f^{(2)}(\alpha)e_n}{f^{(1)}(\alpha)} + \frac{1}{2!}\frac{f^{(3)}(\alpha)e_n^2}{f^{(1)}(\alpha)} + O\left(e_n^3\right)\right] \\ &= f^{(1)}(\alpha)\left[1 + 2C_2e_n + 3C_3e_n^2 + O\left(e_n^3\right)\right]. \end{aligned} \tag{4.3}$$

Dividing (4.2) by (4.3),

$$\begin{aligned} \frac{f(x_n)}{f^{(1)}(x_n)} &= \left[e_n + C_2e_n^2 + C_3e_n^3 + O\left(e_n^4\right)\right]\left[1 + 2C_2e_n + 3C_3e_n^2 + O\left(e_n^3\right)\right]^{-1} \\ &= \left[e_n + C_2e_n^2 + C_3e_n^3 + O\left(e_n^4\right)\right] \\ &\quad \times \left\{1 - \left[2C_2e_n + 3C_3e_n^2 + O\left(e_n^3\right)\right] + \left[2C_2e_n + 3C_3e_n^2 + O\left(e_n^3\right)\right]^2 - \cdots\right\} \\ &= \left[e_n + C_2e_n^2 + C_3e_n^3 + O\left(e_n^4\right)\right]\left\{1 - \left[2C_2e_n + 3C_3e_n^2 + O\left(e_n^3\right)\right] + 4C_2^2e_n^2 + \cdots\right\} \\ &= \left[e_n + C_2e_n^2 + C_3e_n^3 + O\left(e_n^4\right)\right]\left[1 - 2C_2e_n + \left(4C_2^2 - 3C_3\right)e_n^2 + O\left(e_n^3\right)\right] \\ &= e_n - 2C_2e_n^2 + \left(4C_2^2 - 3C_3\right)e_n^3 + C_2e_n^2 - 2C_2^2e_n^3 + C_3e_n^3 + O\left(e_n^4\right) \\ &= e_n - C_2e_n^2 + \left(2C_2^2 - 2C_3\right)e_n^3 + O\left(e_n^4\right), \end{aligned} \tag{4.4}$$

$$\begin{aligned} x_{n+1}^* &= x_n - \frac{f(x_n)}{f^{(1)}(x_n)} \\ &= \alpha + e_n - \left[e_n - C_2e_n^2 + \left(2C_2^2 - 2C_3\right)e_n^3 + O\left(e_n^4\right)\right], \qquad \text{(by (4.4))} \\ &= \alpha + C_2e_n^2 + \left(2C_3 - 2C_2^2\right)e_n^3 + O\left(e_n^4\right)]. \end{aligned} \tag{4.5}$$

Again by (4.5) and the Taylor's expansion,

$$\begin{aligned} f^{(1)}\left(x_{n+1}^*\right) &= f^{(1)}(\alpha) + \left[C_2e_n^2 + \left(2C_3 - 2C_2^2\right)e_n^3 + O\left(e_n^4\right)\right]f^{(2)}(\alpha) + O\left(e_n^4\right) \\ &= f^{(1)}(\alpha)\left\{1 + \left[2C_2e_n^2 + 4\left(C_3 - C_2^2\right)e_n^3 + O\left(e_n^4\right)\right]\left[\frac{f^{(2)}(\alpha)}{2f^{(1)}(\alpha)}\right]\right\} \\ &= f^{(1)}(\alpha)\left[1 + 2C_2^2e_n^2 + 4C_2\left(C_3 - C_2^2\right)e_n^3 + O\left(e_n^4\right)\right]. \end{aligned} \tag{4.6}$$

Adding (4.3) and (4.6),

$$f^{(1)}(x_n) + f^{(1)}\left(x_{n+1}^*\right) = 2f^{(1)}(\alpha)\left[1 + C_2 e_n + \left(C_2^2 + \frac{3}{2}C_3\right)e_n^2 + O\left(e_n^3\right)\right]. \tag{4.7}$$

From equations (4.2) and (4.7),

$$\begin{aligned}
\frac{2f(x_n)}{\left[f^{(1)}(x_n)+f^{(1)}\left(x_{n+1}^*\right)\right]} &= \left[e_n + C_2e_n^2 + C_3e_n^3 + O\left(e_n^4\right)\right] \\
&\quad \times \left[1 + C_2e_n + \left(C_2^2 + \frac{3}{2}C_3\right)e_n^2 + O\left(e_n^3\right)\right]^{-1} \\
&= \left[e_n + C_2e_n^2 + C_3e_n^3 + O\left(e_n^4\right)\right]\left\{1 - \left[C_2e_n + \left(C_2^2 + \frac{3}{2}C_3\right)e_n^2 + O\left(e_n^3\right)\right]\right. \\
&\quad \left. + \left[C_2e_n + \left(C_2^2 + \frac{3}{2}C_3\right)e_n^2 + O\left(e_n^3\right)\right]^2 - \cdots\right\} \\
&= \left[e_n + C_2e_n^2 + C_3e_n^3 + O\left(e_n^4\right)\right]\left[1 - C_2e_n - \frac{3}{2}C_3e_n^2 + O\left(e_n^3\right)\right] \\
&= e_n - C_2e_n^2 - \frac{3}{2}C_3e_n^3 + C_2e_n^2 - C_2^2e_n^3 + C_3e_n^3 + O\left(e_n^4\right) \\
&= e_n - \left(C_2^2 + \frac{1}{2}C_3\right)e_n^3 + O\left(e_n^4\right).
\end{aligned} \tag{4.8}$$

Table 1.

Function	x_0	i		COC		NOFE		Root
$f(x)$		NM	VNM	NM	VNM	NM	VNM	
(1) $x^3 + 4x^2 - 10$	-0.5	109	6	1.98	**2.96**	218	18	1.36523001341448
	1	5	3	1.98	**ND**	10	9	-Do-
	2	5	3	1.99	**ND**	10	9	-Do-
	-0.3	113	6	1.99	**3.05**	226	18	-Do-
(2) $\sin^2(x) - x^2 + 1$	1	5	4	1.98	**3.04**	10	12	1.40449164821621
	3	6	3	1.98	**ND**	12	9	-Do-
(3) $x^2 - e^x - 3x + 2$	2	4	4	1.56	**3.01**	8	12	0.257530285439771
	3	6	4	1.66	**3.04**	12	12	-Do-
(4) $\cos(x) - x$	1	4	2	1.99	**ND**	8	6	0.739085133214758
	1.7	4	3	1.99	**ND**	8	9	-Do-
	-0.3	5	3	1.98	**ND**	10	9	-Do-
(5) $(x-1)^3 - 1$	3.5	7	4	1.98	**2.67**	14	12	2
	2.5	5	4	1.99	**2.99**	10	12	-Do-
(6) $x^3 - 10$	1.5	5	4	1.99	**3.01**	10	12	2.15443469003367
(7) $x\exp\left(x^2\right) - \sin^2(x) + 3\cos(x) + 5$	-2	8	5	1.99	**2.92**	16	15	-1.20764782713013
(8) $x^2\sin^2(x) + \exp\left[x^2\cos(x)\sin(x)\right] - 28$	5	9	5	1.99	**2.87**	18	15	4.82458931731526
(9) $\exp\left(x^2 + 7x - 30\right) - 1$	3.5	11	8	1.99	**2.96**	22	24	3
	3.25	8	5	1.99	**2.81**	16	15	-Do-

NM – Newton's method
VNM – Variant of Newton's method
ND – Not defined
COC – Computational order of convergence
NOFE – Number of function evaluations
i – Number of iterations to approximate the root to 15 decimal places

Thus,

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

$$e_{n+1} + \alpha = e_n + \alpha - \left[e_n - \left(C_2^2 + \frac{1}{2}C_3\right)e_n^3 + O\left(e_n^4\right)\right], \qquad \text{(by (4.8))}. \tag{4.9}$$

$$e_{n+1} = \left(C_2^2 + \frac{1}{2}C_3\right)e_n^3 + O\left(e_n^4\right).$$

Equation (4.9) establishes the third-order convergence of the VNM. ∎

5. CONCLUSIONS

We have shown that VNM is at least third-order convergent provided the first, second, and third derivatives of the function exist. Computed results (Table 1) overwhelmingly support the third-order convergence, and for some functions the Computational Order of Convergence (COC) is even more than three. The most important characteristic of the VNM is that unlike all other third-order or higher order methods, it is not required to compute second or higher derivatives of the function to carry out iterations.

Apparently, the VNM needs one more function evaluation at each iteration, when compared to Newton's method. However, it is evident by the computed results (Table 1) that the total number of function evaluations required is less than that of Newton's method.

REFERENCES

1. J.E. Dennis and R.B. Schnable, *Numerical Methods for Unconstrained Optimisation and Nonlinear Equations*, Prentice Hall, (1983).
2. R.L. Burden and D.J. Faires, *Numerical Analysis. PWS*, Kent Publishing, Boston, (1993).
3. T.G.I. Fernando and S. Weerakoon, Improved Newton's method for finding roots of a nonlinear equation, *Proceedings of the 53rd Annual Sessions of Sri Lanka Association for the Advancement of Science (SLAAS)* **E1–22**, 309, (1997).