MATEMATIKA, 2017, Volume 33, Number 1, 43–54

Determining radius of convergence of Newton's method using radius of curvature

[1]Ridwan Pandiya and [2]Ismail Bin Mohd
[1]Department of Informatics, ST3 Telkom,
Jalan DI. Panjaitan No. 128, Purwokerto, Jawa Tengah, Indonesia
[2]Laboratory of Statistical Computational and Operations Research,
Institute of Mathematical Research, UPM Serdang, Selangor, Malaysia
e-mail: [1]ridwanpandiya@st3telkom.ac.id, [2]ismailmohd@upm.edu.my

Abstract In this paper, we propose a method how to manage the convergence of Newton's method if its iteration process encounters a local extremum. This idea establishes the osculating circle at a local extremum. It then uses the radius of the osculating circle also known as the radius of curvature as an additional number of the local extremum. It then takes that additional number and combines it with the local extremum. This is then used as an initial guess in finding a root near to that local extremum. This paper will provide several examples which demonstrate that the proposed idea is successful and they perform to fulfill the aim of this paper.

Keywords Newton's method; convergence; curvature function; radius of curvature.

Mathematics Subject Classification 34K28

1 Introduction

One of the most frequently occuring problems in scientific work is to find the roots of the equation of the form

$$f(x) = 0. \tag{1}$$

Iterative procedures for solutions of (1) are routinely employed. Starting with the classical Newton's method, several methods for finding roots of equations have come to exist each of which has its own advantages and limitations.

The Newton's method of root finding based on the iterative formula is given by

$$x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)}. \tag{2}$$

Newton's method displays a faster quadratic convergence near the root while it requires evaluation of the function and its derivative at each step of the iteration.

However, when the derivative evaluated is zero, Newton's method stalls [1]. Newton's method will face several obstacles if it has small values of the derivative, the Newton iteration off shoots away from the current point of iteration and may possible converge to a root far away from the intended domain. For certain forms of equations, Newton's method diverges or oscillates and fails to converge to the desire root. We observe these obstacles by considering the function with expression

$$f(x) = x^3 - x^2 - x + 3,$$

where its graph is given in Figure 1 and Figure 2 [2]. If we start the iteration at x_0 where x_0 is a fixed number in the interval (1.5,1.6), then we will obtain infinite sequence like

$$x_0, \quad x_1, \quad x_0. \quad x_1, \quad \ldots$$

which does not converge to x^*, the root of f (See Figure 1).

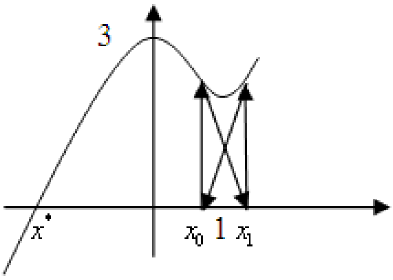


Figure 1: Newton's iteration will oscillate between x_0, x_1, x_0. x_1, ...

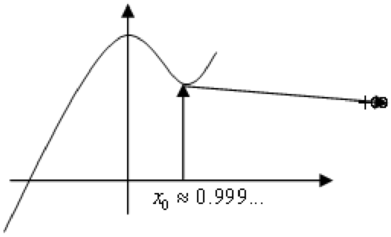


Figure 2: Starting at x_0, the Newton's iteration yield no solution ($+\infty$)

If we start at $x_0 \approx 0.999...$ as shown in Figure 2, we may get x_1 which approaches $+\infty$ (exceed the computer number). Therefore the algorithm cannot proceed.

In this paper, we would like to compute all the zeroes of a function when its graph looks like in Figure 3.

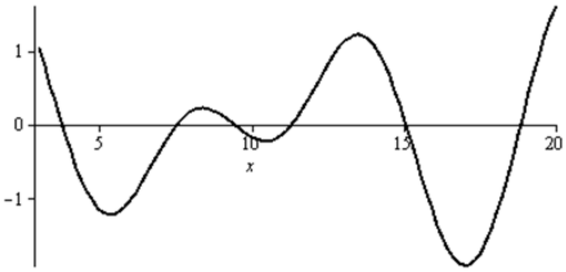


Figure 3: Graph of $f(x) = \sin(x) + \sin(2x/3), x \in [3, 20]$

Furthermore, if the derivative of a function $f(x)$ at a point is zero, then the point is a critical point of $f(x)$, and special for this case, we consider those points as minimizer or maximizer (extreme). Our idea is that how does Newton's method can still be used where its initial point as an initial is minimizer or maximizer.

When the Newton's methods applied to a point which has zero derivative, the iterative process would not work properly since the denominator equals to zero. In order to ensure that Newton's method can still be employed, it needs to add a number to that minimizer or maximizer so that its derivative at this summand does not equal to zero. The question is how to obtain a suitable number such that no zero exists between that extreme point and the summand? Of course we cannot add any numbers, therefore, we use the theory of curvature of a function at any extreme or critical points for Newton's method works.

In Section 2, it will be explained in advance of curvature and radius of curvature.Section 3 contains the explanation how the radius of curvature can be used to satisfy the Liptschitz property for Newton's method. The illustration of the idea of curvature through an example is given in Section 4. In Section 5, it will be shown that how to improve the idea of curvature if the curve bends very sharply. 20 testing examples have been used to observe the impact of the proposed algorithm as given in Section 6. The conclusion which ends this paper, will be given in Section 7.

2 Curvature of a function

The idea of curvature is the measure of how sharply a curve bends. We would expect the curvature to be 0 for a straight line, and to be very small for curves which bend sharply. If someone moves along a curve, the direction of the tangent line will not change as long as the curve is flat. Its direction will change if the curve bends. The more the curve bends, the more the direction of the tangent line will change. As known that the movement of Newton's method searching process is depended on the tangent line of each iteration. We are thus led to the following definition and theorems which are taken from [2].

Definition 1 [2] *Let the curve C be given by the differentiable vector function $f(t) = f_1(t)\mathbf{i} + f_2(t)\mathbf{j}$. Suppose that $\phi(t)$ denotes the direction of $f'(t)$.*

(i) *The curvature of C, denoted by $\kappa(t)$, is the absolute value of the rate of change of direction with respect to arc length (s), that is,*

$$\kappa(t) = \left|\frac{d\phi}{ds}\right|$$

where $\kappa(t) \geq 0$.

(ii) *The radius of curvature $\rho(t)$ is defined by $\rho(t) = \dfrac{1}{\kappa(t)}$, if $\kappa(t) > 0$.*

Theorem 1 [2] *If $T(t)$ denotes the unit tangent vector to f, then $\kappa(t) = \left|\dfrac{dT}{ds}\right|$.*

Theorem 2 [2] *If C is a curve with equation $y = f(x)$ where f is twice differentiable,*

$$\kappa = \frac{|f"(x)|}{\left(1 + (f'(x))^2\right)^{3/2}}.$$

According to [3], when the curvature $\kappa(t) > 0$, the centre of curvature lies along the direction of $N(t)$ at distance $1/\kappa$ from the point $\alpha(t)$. When $\kappa(t) < 0$, the center of curvature lies along the direction $-N(t)$ at distance $-1/\kappa$ from $\alpha(t)$. In either case, the centre of curvature is located at

$$\alpha(t) + \frac{1}{\kappa(t)}N(t).$$

The osculating circle when $\kappa \neq 0$, is the circle at the centre of curvature with radius $1/|\kappa|$ which is called the radius of curvature. The osculating circle approximates the curve locally up to the second order (the illustration is in Figure 4).

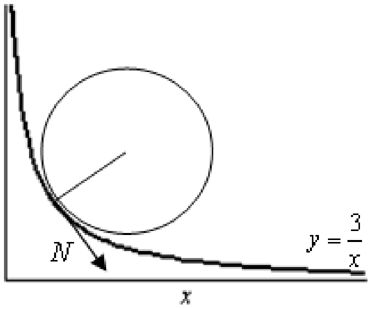


Figure 4: Approximation of the osculating circle to the curve $y = 3/x$

3 An appropriate initial guess

It has been mentioned in Section 1 that the idea is to find a promised number to be added to extreme point such that the new point will be a good initial guess for Newton's method succeeded in finding the zero point of a function.

Let consider Figure 5.

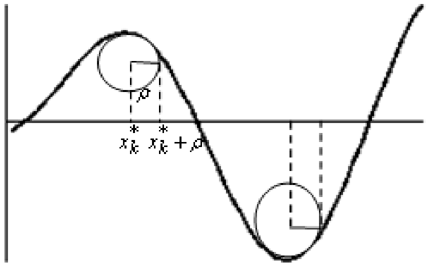


Figure 5: Finding a suitable initial point for Newton's iteration

Basically, in order to make Newton's method converges to x^*, the zero of a function f, an initial estimation that is closed enough to x^*, is needed. Therefore, for obtaining a suitable number named ρ must be searched such that $x_k^* + \rho$ becomes the best estimation as an initial point for Newton's iteration. In the above explanation, the radius of curvature of $f\left(x\right)$ can be employed as ρ. Therefore $x_k^* + \rho$ is one of the initial best estimation when the Newton's method is used to find the root of $f\left(x\right)$. Therefore, it can be proved that $\eta = \left|x_k^* + \rho - x^*\right|$ is the radius of the largest interval around x^* in which the application of Newton's method to any point in $\left(x^* - \eta, x^* + \eta\right)$ will converge to x^*. However, the following definition is needed.

Definition 2 [4] *The function $f : D \subset R \rightarrow R$ is Lipschitz continuous function with constant γ in an interval D, written $f \in Lip_\gamma\left(D\right)$, if for every $x, y \in D$,*

$$\left|f\left(x\right) - f\left(y\right)\right| \leq \gamma \left|x - y\right|.$$

For the convergence of Newton's method, we need to show that $f' \in Lip_\gamma\left(D\right)$. This condition has been shown in [4] through the following Lemma.

Lemma 1 [4] *If* (i) *$f : D \subset R \rightarrow R$ for an open interval D,* (ii) *$f' \in Lip_\gamma\left(D\right)$, then for any $x, y \in D$,*

$$|f(y) - f(x) - f'(x)(y - x)| \leq \frac{\gamma(y - x)^2}{2}.$$

For most problems, Newton's method will converge q-quadratically to the root of one nonlinear equation with one variable. We shall now state the fundamental theorem of numerical mathematics.

Theorem 3 [4] *If* (i) *$f : D \subset R \rightarrow R$ for an open interval D,* (ii) *$f' \in Lip_\gamma\left(D\right)$,* (iii) *for some $\beta > 0$, $|f'(x)| \geq \beta$ for every $x \in D$,* (iv) *$f(x) = 0$ has a solution $x^* \in D$, for some $\eta > 0$ such that if $|x_0 - x^*| < \eta$, then the sequence $\{x_n\}$ generated by*

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

exists and converges to x^*. Furthermore , for $(n = 0, 1, 2, ...)$

$$|x_{n+1} - x^*| \leq \frac{\gamma}{2\beta}|x_n - x^*|^2. \tag{3}$$

Now, we will prove that $\hat{\eta} = |x_1^* + \hat{\rho} - x^*|$ is a radius of the largest interval around the solution of $f(x) = 0$ as shown in the following theorem.

Theorem 4 *If* (i) *$f : D \subset R \rightarrow R$ is an objective function with x_1^* is a local minima of $f(x)$,* (ii) *$f' \in Lip_\gamma\left(X\right)$,* (iii) *for some $\rho > 0$, $|f'(x)| \geq \rho$ for every $x \in D$,* (iv) *$f(x) = 0$ has a solution $x^* \in D$, for some $\eta > 0$ such that if $|x_0 - x^*| < \eta$, then the sequence $\{x_n\}$ generated by*

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

exists and converges to x^*. Furthermore , for $(n = 0, 1, 2, ...)$

$$|x_{n+1} - x^*| \leq \frac{\gamma}{2\rho}|x_n - x^*|^2. \tag{4}$$

Proof Let $\hat{\rho}$ be the radius of curvature of $f(x)$ at x_1^*. Let $\hat{\eta}$ be the radius of the largest interval around x^*, that is contained in D and define

$$\eta = \min\{\hat{\eta}, (2\rho/\gamma)\}.$$

We will show by induction that for $n = 0, 1, 2, ...,$ the equation (4) holds, and

$$|x_{n+1} - x^*| \leq |x_n - x^*| < \eta.$$

Take $\hat{\eta} = |x_k^* + \hat{\rho} - x^*|$ as the radius of the largest interval around $x^* \in D$, and let $x_0 = x_k^* + \hat{\rho}$ be an initial point which is a lower bound or an upper bound of $[x^* - \hat{\eta}\,, x^* + \hat{\eta}]$. The proof simply shows at each iteration that the new error $|x_{n+1} - x^*|$ is bounded by a constant times the error the affine model makes, in approximating f at x^*.

For $n = 0$, we have

$$\begin{aligned}
x_1 - x^* &= x_0 - x^* - \frac{f(x_0)}{f'(x_0)} = x_0 - x^* - \frac{f(x_0) - f(x^*)}{f'(x_0)} \\
&= x_k^* + \hat{\rho} - x^* - \frac{f(x_k^* + \hat{\rho}) - f(x^*)}{f'(x_k^* + \hat{\rho})} \\
&= \frac{x_k^* f'(x_k^* + \hat{\rho})}{f'(x_k^* + \hat{\rho})} + \frac{\hat{\rho} f'(x_k^* + \hat{\rho})}{f'(x_k^* + \hat{\rho})} - \frac{x^* f'(x_k^* + \hat{\rho})}{f'(x_k^* + \hat{\rho})} - \frac{f(x_k^* + \hat{\rho}) - f(x^*)}{f'(x_k^* + \hat{\rho})} \\
&= \frac{1}{f'(x_k^* + \hat{\rho})} \left[f(x^*) - f(x_k^* + \hat{\rho}) - x^* f'(x_k^* + \hat{\rho}) + x_k^* f'(x_k^* + \hat{\rho}) + \hat{\rho} f'(x_k^* + \hat{\rho}) \right] \\
&= \frac{1}{f'(x_k^* + \hat{\rho})} \left[f(x^*) - f(x_k^* + \hat{\rho}) - f'(x_k^* + \hat{\rho})\ (x^* - x_k^* - \hat{\rho}) \right]
\end{aligned}$$

By taking the absolute value of the above, we have

$$|x_1 - x^*| = \left| \frac{1}{f'(x_k^* + \hat{\rho})} \left[f(x^*) - f(x_k^* + \hat{\rho}) - f'(x_k^* + \hat{\rho})\ (x^* - x_k^* - \hat{\rho}) \right] \right|.$$

Thus by Lemma 1, we have

$$|x_1 - x^*| \leq \frac{\gamma}{2\,|f'(x_k^* + \hat{\rho})|} \, |x_k^* + \hat{\rho} - x^*|^2$$

and by assumptions on $f'(x)$, we obtain

$$|x_1 - x^*| \leq \frac{\gamma}{2\rho} \, |x_k^* + \hat{\rho} - x^*|^2 .$$

Since

$$|x_k^* + \hat{\rho} - x^*| \leq \eta \leq \frac{2\rho}{\gamma},$$

we have

$$|x_1 - x^*| \leq \frac{\gamma}{2\rho} \, |x_k^* + \hat{\rho} - x^*|^2 \leq \frac{\gamma}{2\rho} \, |x_k^* + \hat{\rho} - x^*| \, \frac{2\rho}{\gamma} \leq |x_k^* + \hat{\rho} - x^*| < \eta.$$

The proof of the induction step then proceeds identically. □

Based on the above discussion, an experiment can be tried on several example problems to demonstrate that the use of radius of curvature is quite effective to make Newton's method converges to the desire solutions.

4 Numerical examples

In this section, for obtaining the nearest root closed enough to an extreme point, the initial guesses $x_k^* + \rho$, and $x_k^* - \rho$ are tried where x_k^* is an extreme point, and ρ is the radius of curvature at this extreme point. Five examples (Exp.) given in Table 1, have been tried to obtain a root to the right hand side of extreme point, and a root to the left extreme point of each function.

Table 1: Data for the testing

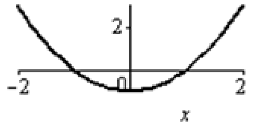
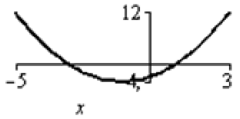
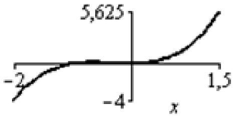
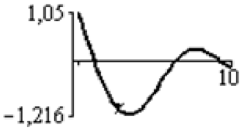
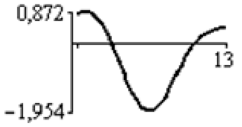
Exp	Function	Domain	Graph
1	$f_1(x) = x^2 - 1$	$[-2, 2]$	
2	$f_2(x) = x^2 + 2x - 3$	$[-5, 3]$	
3	$f_3(x) = x^3 + x^2$	$[-1.5, 1]$	
4	$f_4(x) = \sin x + \sin\left(\frac{2}{3}x\right)$	$[3, 10]$	
5	$f_5(x) = \cos\left(\frac{3}{5}x\right)\cos(2x) + \sin x$	$[3, 10]$	

Table 2 shows that the use of initial guess $x_k^* + \rho$, with x_k^* is a local extremum of a function, and ρ is a radius of curvature at that local extremum, will make Newton's method (N) converges (C) to a root of a function closest to the local extremum. However, for Exp. 5 when the radius of curvature is too small, Newton's method iterative process fails (F) to carry the job. To overcome this obstacle, it has been made a modification to the radius of curvature which will be discussed in the next section.

5 An improved starting point

Before we go through about this special case, now consider Figure 6.

Table 2: Results of five testing examples

Exp.	x_k^*	ρ	x^{*+}	$f(x^{*+})$	x^{*-}	$f(x^{*-})$	N
1	0	0.5	1	0	-1	0	C
2	-1	0.5	1	0	-3	0	C
3	$-2/3$	0.5	$-1.75962\text{e}{-162}$	0	-1	0	C
4	5.36225	1.01755	7.53982	$-1.01481\text{e}{-016}$	3.76991	$1.82363\text{e}{-016}$	C
5	10.9598	0.191837	12.1839	$4.40999\text{e}{-016}$	6.71118	-1.89519	F

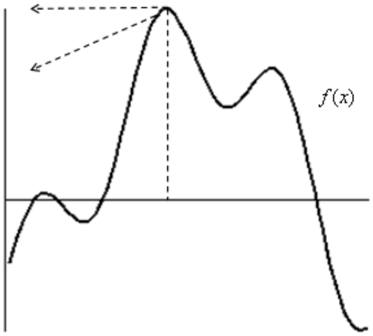


Figure 6: The behaviour of Newton's method

Unfortunate case when Newton's method encounters a trial guess near such a local extremum, then Newton's method will send its solution far away from the desired solution (see Figure 6). This situation happened in Exp. 5 of Table 2 where the size of radius of curvature to be added to the minima point is not enough to bring that point to the expected root.

For details, in Exp. 5, $x_k^* = 10.9598$ is a minimizer of $f_5(x)$, $\rho = 0.191837$ is radius of curvature at x_k^*, then $x_k^* + \rho$ is an initial guess in finding the nearest root from x_k^* on the right hand side, and $x_k^* - \rho$ on the left hand side. In Table 2, it has been given a sign that Exp. 5 failed in getting the root on the left hand side of x_k^*. Now we try to double up the radius of curvatureto become 2ρ, and use $x_k^* + 2\rho$ as the new initial guess, then with that new initial guess, we will obtain $x^{*-} = 9.93822$ which is the nearest root of x_k^* on the left. So we assume that the failure due to Exp. 5 caused by the small radius of curvature. To overcome this obstacle, we decide to restrict the radius of curvature as

$$\rho \geq r \text{ for } r \in (0, 1).$$

The algorithm of modified radius of curvature can be described in Algorithm M.

Algorithm M

This simple algorithm computes $\rho \in R$ using data $(x_0, \varepsilon, r \in R, m \in N)$ where x_0 is local extremum of a function, ε is a tolerance, r is a real number, and m is the maximum number of iteration.

```
1. $\rho_j = \frac{\left(1+\left(f'(x_0)\right)^2\right)^{3/2}}{|f"(x_0)|}$
2. $\rho := \rho_j$
3. $i := 1$
4. while $\rho_i < r$ do
     4.1 $\rho_i := i\rho_j$
     4.2 $i := i + 1$
     4.3 $\rho := \rho_i$
5. return.
```

6 Numerical results

In this section, the proposed method is employed to solve several nonlinear equations. All experiments were performed on a personal computer with AMD Dual-Core Processor E-350 1.6 GHz and 2 GB memory. The operating system was Windows 7 Starter (32-bit) and the implementations were done in Microsoft Visual C++ 6.0.

We used the following 20 test functions and display the approximate zeros x^*.

$$f_1(x) = \sin(x) + \sin\left(\tfrac{2}{3}x\right) \qquad f_2(x) = \cos\left(\tfrac{3}{5}x\right)\cos(2x) + \sin(x)$$

$$f_3(x) = \cos\left(\tfrac{2}{5}x\right)\sin\left(\tfrac{1}{10}x\right) + \cos(x) \qquad f_4(x) = \sin\left(\tfrac{4}{9}x\right)\sin(x)$$

$$f_5(x) = \tfrac{1}{2}\cos(x) + \tfrac{1}{3}\sin(2x) \qquad f_6(x) = \sin(2x)\sin(x) + \sin\left(\tfrac{2}{3}x\right)$$

$$f_7(x) = -\sum_{i=1}^{5} i\cos\left\{(i+1)\,x + i\right\} \qquad f_8(x) = \sin(x) + \sin\left(\tfrac{10}{3}x\right) + \ln x - 0.84x + 3$$

$$f_9(x) = -\exp(0.1x)\sum_{i=1}^{5}\left\{(i+1)\,x + i\right\} \qquad f_{10}(x) = \cos(x) + \tfrac{10}{3}\cos\left(\tfrac{10}{3}x\right) + \tfrac{1}{x} - 0.84$$

$$f_{11}(x) = -\sum_{i=1}^{5} i\sin\left\{(i+1)\,x + i\right\} \qquad f_{12}(x) = \sin(x) + \sin\left(\tfrac{10}{3}x\right) + \ln x - 0.84x$$

$$f_{13}(x) = -\sum_{i=1}^{5} \sin\left\{(i+1)\,x + i\right\} \qquad f_{14}(x) = x^2 - 1$$

$$f_{15}(x) = x^2 + 2x - 3 \qquad f_{16}(x) = x^3 + x^2$$

$$f_{17}(x) = 2x^2 - 1.05x^4 + \tfrac{1}{6}x^6 - x \qquad f_{18}(x) = x^4 + 4x^3 + 4x^2 - 0.5$$

$$f_{19}(x) = 3x - x^3 \qquad f_{20}(x) = x^6 - 22x^4 + 9x^2 + 102$$

Table 3 shows that the use of radius of curvatute at the extreme point will make Newton's method always converges to the roots closed to this extreme point. Nonzero value of r indicate that the functions have the small radius of curvature at their extreme points.

7 Conclusion

In this paper, we have presented that the radius of curvature at maximizer or minimizer points can be used as an increment to those extremum points in the attempt to find the radius of convergence of Newton's method near to the said maximizer or minimizer of a function. Numerical results show that our valuable method succeed in finding the desire solutions.

Table 3: Computational results of twenty testing examples

Exp	x_k^*	ρ	r	x^{*+}	$f(x^{*+})$	r	x^{*-}	$f(x^{*-})$
1	5.36225	1.01755	0	7.53982	−1.01481e−016	0	3.76991	1.82363e−016
	8.39609	1.73864	0	9.42478	1.22461e−016	0	7.53982	−1.01481e−016
	10.4535	1.73861	0	11.3097	−2.71918e−016	0	9.42478	1.22461e−016
	13.4873	1.01755	0	15.0796	7.69079e−016	0	11.3097	−2.71918e−016
	17.0392	0.721093	0	18.8496	−1.22461e−015	0	15.0796	7.69079e−016
2	10.9598	0.191837	0	12.1839	4.40999e−016	0.2	9.93822	−1.28993e−015
3	−15.708	0.854701	0	−13.4781	$-6.18754e-016$	0	−17.9378	1.50541e−016
	−12.1808	1.07705	0	−10.5816	−6.62719e−017	0	−13.4781	$-6.18754e-016$
	−9.5911	1.17664	0	−8.66132	−6.2244e−016	0	−10.5816	$-6.62719e-017$
	−6.4786	0.966118	0	−4.59502	−9.98415e−017	0	−8.66132	−6.2244e−016
	−3.06054	0.920927	0	−1.44966	6.19554e−017	0	−4.59502	−9.98415e−017
	0.0999216	1.00009	0	1.70283	3.91522e−017	0	−1.44966	6.19554e−017
	3.23845	1.10178	0	4.88978	2.5247e−016	0	1.70283	3.91533e-017
	6.07014	1.05587	0	7.16913	−2.27954e−016	0	4.88978	2.5247e−016
	9.29931	0.88308	0	11.2033	8.2974e−016	0	7.16913	−2.27954e−016
4	−8.42662	1.03325	0	−7.06858	8.65927e−017	0	−9.42478	−3.18162e−016
	−6.66794	1.12298	0	−6.28319	−8.3768e−017	0	−7.06858	8.65927e−017
	−4.50953	0.877408	0	−3.14159	1.206e−016	0	−6.28319	−8.3768e−017
	−1.93595	0.947863	0	−1.08393e−162	0	0	−3.14159	1.206e−016
	1.93595	0.947863	0	3.14159	1.206e−016	0	1.08393e−016	0
	4.50953	0.877408	0	6.28319	−8.3768e−017	0	3.14159	1.206e−016
	6.66794	1.12298	0	7.06858	8.65927e−017	0	6.28319	−8.3768e−017
	8.42662	1.03325	0	9.42478	−3.18162e−016	0	7.06858	8.65927e−017
5	2.56634	0.61094	0	3.98965	5.55112e−017	0	1.5708	7.14354e−017
6	4.231	0.219526	0	4.71239	-2.44921e-016	0	3.78726	−3.99529e−016
	5.19377	0.219526	0	5.63752	−3.18051e−016	0	4.71239	−2.44921e−016
7	−9.28634	0.104062	0.1	−9.03415	−9.76996e−015	0.1	−9.55476	8.43769e−015
	-8.79406	0.104051	0.1	−8.57612	5.77316e−015	0.1	−9.03415	−9.76996e−015
	−8.29039	0.101354	0.1	−8.05487	3.55271e−015	0.1	-8.57612	5.77316e-015
	−7.70831	0.102709	0.1	−7.40542	−7.21645e−015	0.1	−8.05487	3.55271e−015
	−7.08351	0.101927	0.1	−6.73964	−3.19744e−014	0.1	−7.40542	−7.21645e−015
	−6.47857	0.103276	0.1	−6.15885	−1.77636e−015	0.1	−6.73964	−3.19744e-014
	−5.94894	0.10492	0.1	−5.70985	−1.5099e−014	0.1	−6.15885	−1.77636e−015
	−5.4614	0.1052	0.1	−5.19666	−8.88178e−015	0.1	−5.70985	−1.5099e−014
	−4.96318	0.104799	0.1	−4.71693	−1.11011e−015	0.1	−5.19666	−8.88178e−015
	−4.47753	0.104027	0.1	−4.23649	4.76789e−015	0.1	−4.71693	−1.11011e−015
	−3.98396	0.105011	0.1	−3.73129	2.88658e−015	0.1	−4.23649	4.76789e-015
	−3.49725	0.100153	0.1	−3.27157	−2.22045e−015	0.1	−3.73129	2.88658e−015
	−3.00316	0.104062	0.1	−2.75097	−1.33227e−015	0.1	−3.27157	−2.22045e−015
	−2.51088	0.104052	0.1	−2.29294	4.44089e−016	0.1	−2.75097	−1.33227e−015
8	5.19978	0.334964	0.3	5.83696	8.88178e−016	0.3	4.60891	8.88178e−016
	6.15443	0.364976	0.3	6.48083	8.88178e−016	0.3	5.83696	8.88178e−016
	7.06776	0.385188	0.3	7.67077	8.88178e−016	0.3	6.48083	8.88178e−016
	7.9385	0.338984	0.3	8.18818	−1.77636e−015	0.3	7.67077	8.88178e−016

Table 3: (Continued)

Exp	x_k^*	ρ	r	x^{*+}	$f(x^{*+})$	r	x^{*-}	$f(x^{*-})$
9	4.48968	0.205434	0.2	4.74779	−3.65861e−015	0.2	4.02655	5.14902e−016
	5.16194	0.200876	0.2	5.53319	8.04363e−015	0.2	4.74779	−3.65861e−015
	5.85113	0.207187	0.2	6.31858	−7.46456e−016	0.2	5.53319	8.04363e−015
10	12.2438	0.22176	0.2	12.7028	1.66533e−015	0.2	11.7764	1.11022e−016
	13.179	0.21162	0.2	13.6398	2.44249e−015	0.2	12.7028	1.66533e−015
	14.1643	0.216723	0.2	14.7318	6.99441e−015	0.2	13.6398	2.44249e−015
	15.0633	0.221103	0.2	15.3878	8.65974e−015	0.2	14.7318	6.99441e−015
	16.0141	0.210649	0.2	16.6224	−1.02141e−014	0.2	15.3878	8.65974e−015
	16.9904	0.218488	0.2	17.3735	−1.18794e−014	0.2	16.6224	−1.02141e−014
	17.885	0.219978	0.2	18.3692	4.21885e−015	0.2	17.3735	7.77156e−015
	18.8495	0.210323	0.2	19.3297	2.78666e−014	0.2	18.3692	4.21885e−015
11	−9.03744	0.102605	0.1	−8.79418	−1.02141e−014	0.1	−9.29755	−2.35367e−014
	−8.54977	0.100094	0.1	−8.33823	3.55271e−015	0.1	−8.79418	−1.02141e−014
	−8.00868	0.101775	0.1	−7.73795	4.44089e−016	0.1	−8.33823	3.55271e−015
	−7.39728	0.100627	0.1	−7.06931	−3.61933e−014	0.1	−7.73795	4.44089e−016
	−6.77458	0.100028	0.1	−6.42988	−2.66454e−015	0.1	−7.06931	−3.61933e−014
	−6.20297	0.102166	0.1	−5.93072	1.02141e−014	0.1	−6.42988	−2.66454e−015
	−5.70624	0.101291	0.1	−5.4615	−1.19904e−014	0.1	−5.93072	1.02141e−014
12	0.62006	0.231592	0.2	0.925759	−1.11022e−016	0.2	0.369959	2.22045e−016
	1.41124	0.207872	0.2	1.88437	0	0.2	0.925759	1.11022e−016
	2.26149	0.260227	0.1	2.59313	4.44089e−016	0.2	1.88437	0
13	−8.08035	0.216855	0.2	−7.81858	4.79548e−015	0.2	−8.53982	6.58544e−016
	−7.40995	0.202454	0.2	−7.03319	−4.67072e−015	0.2	−7.81858	4.79548e−015
	−6.72004	0.212611	0.2	−6.24779	1.85789e−015	0.2	−7.03319	−4.67072e−015
	−6.14385	0.214841	0.2	−6.02655	−7.90844e−016	0.2	−6.24779	1.85789e−015
	−5.72898	0.208749	0.2	−5.46239	−1.98916e−015	0.2	−6.02655	−7.90844e−016
14	0	0.5	0	1	0	0	−1	0
15	−1	0.5	0	1	0	0	−3	0
16	−0.66667	0.5	0	−1.68131e−0162	0	0	−1	0
17	0.2704	0.322013	0.1	0.614543	0	0.1	0	0
	0.94168	0.308503	0.1	1.2004	0	0.1	0.614543	0
	1.75767	0.156304	0.1	2.023	−4.44089e−016	0.1	1.2004	0
18	−2	0.125	0	−1.5412	0	0	−2.30656	0
	−1	0.25	0	−0.458804	0	0	−1.5412	0
	0	0.125	0	0.306563	0	0	−0.458804	0
19	−1	0.333333	0.2	0	0	0.2	−1.73205	−8.88178e−016
	1	0.333333	0.2	1.73205	8.88178e−016	0.2	0	0
20	−3.80252	0.300101	0.3	−1.59114	0	0.3	−4.62113	−8.2423e−013
	3.80252	0.300101	0.3	4.62113	−8.2423e−013	0.3	1.59114	0

Acknowledgments

This research was supported by Jurnal Karya Asli Lorekan Ahli Matematik (KALAM), Lot 3116, Jalan Pantai, Kampung Pengkalan Maras, Mengabang Telipot, 21030 Kuala Terengganu, Malaysia during working as Research Fellow at Laboratory of Statistical Computational and Operations Research, Institute of Mathematical Research, UPM Serdang, Selangor, Malaysia

References

[1] Ababu, T. T. A two-point Newton method suitable for nonconvergent cases and with super-quadratic convergence. *Advance in Numerical Analysis.* Hindawi Publishing Corporation, Article ID 687383, http://dx.doi.org/10.1155/2013/687382. 2013.

[2] Mohd., I. B. The width is unreachable, the travel is at the speed of light. *Siri Syarahan Inaugral KUSTEM: 6 (2002), Inaugral Lecture of Prof. Dr. Ismail Bin Mohd.* 14th September 2002.

[3] Grossman, S. I. *Calculus.* 3rd Edition. London: Academic Press. 1984.

[4] Dennis, J. E. and Schnabel, R. B. *Numerical Methods for Unconstrained Optimization and Nonlinear Equations.* New Jursey: Prentice-Hall. 1983.