Hindawi Publishing Corporation
International Journal of Engineering Mathematics
Volume 2016, Article ID 8565821, 12 pages
http://dx.doi.org/10.1155/2016/8565821

Research Article

An Efficient and Straightforward Numerical Technique Coupled to Classical Newton's Method for Enhancing the Accuracy of Approximate Solutions Associated with Scalar Nonlinear Equations

Grégory Antoni

Aix-Marseille Université, IFSTTAR, LBA UMR_T24, 13016 Marseille, France

Correspondence should be addressed to Grégory Antoni; antoni.gregory@yahoo.fr

Received 3 June 2016; Accepted 14 August 2016

Academic Editor: Yurong Liu

This study concerns the development of a straightforward numerical technique associated with Classical Newton's Method for providing a more accurate approximate solution of scalar nonlinear equations. The proposed procedure is based on some practical geometric rules and requires the knowledge of the local slope of the curve representing the considered nonlinear function. Therefore, this new technique uses, only as input data, the first-order derivative of the nonlinear equation in question. The relevance of this numerical procedure is tested, evaluated, and discussed through some examples.

1. Introduction

The resolution of nonlinear problems is an issue frequently encountered in several scientific fields such as mathematics, physics, or many engineering branches, for example, mechanics of solids [1–8]. In most cases, these problems are governed by nonlinear equations not having any analytical solution. In this regard, the introduction of iterative methods is therefore needed in order to provide a numerical approximate solution associated with any type of nonlinear equation [9–23]. Among these iterative algorithms, Classical Newton's Method (CNM) [24, 25] is one of the most used mainly for the following reasons: (i) the simplicity for numerical implementation in any scientific computation software; (ii) the only knowledge of the first-order derivative of the considered function; (iii) the quadratic rate of convergence. In this paper, we propose a New Numerical Technique (NNT) based on geometric considerations which enable providing a more accurate approximate solution than that obtained by CNM. The present study is organized as follows: (i) in the first part, Section 2.1, we outline the scientific framework of this study, then, in second part, Section 2.2, we recall CNM including some convergence results, and, finally, in the third part, Section 2.3, we present NNT which uses only the first-order derivative of the considered nonlinear equation in order to enhance the predictive abilities of CNM; (ii) in Section 3, the numerical relevance of the proposed procedure is addressed, assessed, and discussed on some specific examples.

2. A New Numerical Technique (NNT) Combined with Classical Newton's Method (CNM)

2.1. Problem Statement. We consider scalar-valued nonlinear equation $f : x \in \Omega \subset \mathbb{R} \to f(x) \in \mathbb{R}$ (with $f \in \mathscr{C}^{\infty}(\Omega)$) in the following form (see Figure 1):

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

where $\mathscr{C}^{\infty}(\Omega)$ denotes the class of infinitely differentiable functions in domain Ω, δ is the simple solution (so-called "simple true zero" or "simple root") on interval $I = [a, b] \in \Omega \subset \mathbb{R}$, that is, $f(\delta) = 0$ with $\delta \in I$.

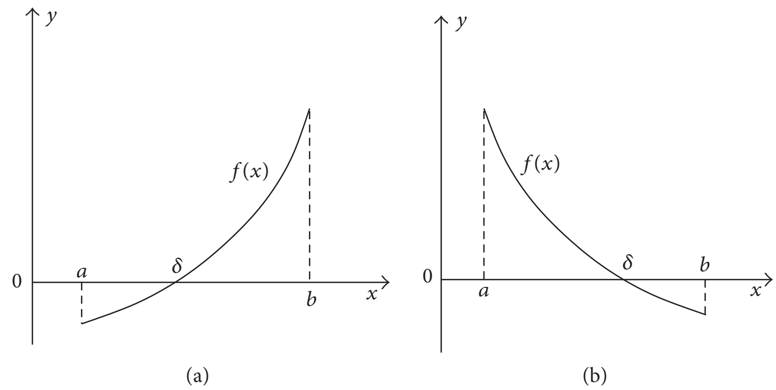

FIGURE 1: Schematic diagram associated with the problem under consideration: monotonically increasing (a) and decreasing (b) nonlinear function f with a simple root δ on interval $[a, b]$.

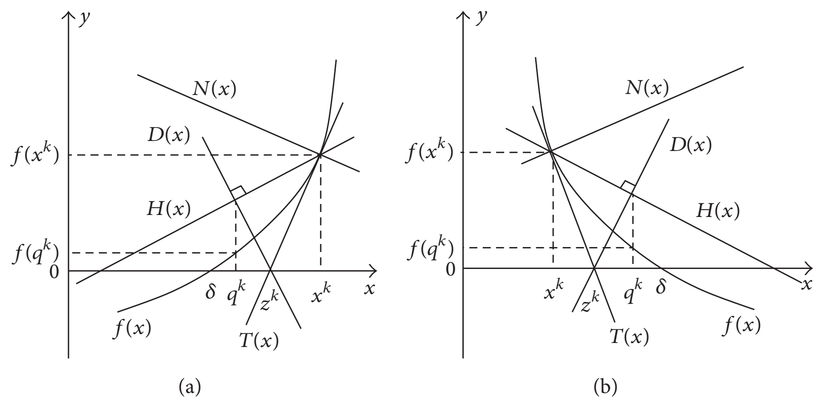

FIGURE 2: Schematic diagram with the specific entities used by NNT combined with CNM applied on monotonically increasing (a) and decreasing (b) nonlinear function f.

2.2. Classical Newton's Method (CNM)

2.2.1. Iterative Algorithm. For using Classical Newton's Method (CNM) [24, 25], we consider only the first-order term in Taylor series expansion associated with function f (i.e., the linearization of the considered function):

$$f(x) \approx f(p) + f'(p)(x - p), \quad \forall x, p \in I, \tag{2}$$

where $f'(p)$ denotes the first-order derivative of function f at point p.

The equation of tangent straight line T passing at point $p = x^k \in I$ associated with function f is ($\forall x, y \in I$; see Figure 2) as follows:

$$T(x) = f'\left(x^k\right)\left(x - x^k\right) + f\left(x^k\right), \quad \forall k = 0, \ldots, K, \tag{3}$$

where $\dagger^k$ (resp., $\dagger^{k+1}$) denotes kth (resp., $(k+1)$th) iteration associated with variable $\dagger$ (with $K \in \mathbb{N}$).

Using (3), the linearization of (1) must check the following relation ($\forall x, y \in I$ and $\forall x^k \in I$):

$$T(x) = 0 \Longleftrightarrow f'\left(x^k\right)\left(x - x^k\right) + f\left(x^k\right) = 0, \quad \forall k = 0, \ldots, K. \tag{4}$$

Based on (4), $(k+1)$th iterative point $x^{k+1} \in I$ provided by CNM is solution $x = x^{k+1}$ ($\forall x \in I$) such as

$$x^{k+1} = x^k - \frac{f\left(x^k\right)}{f'\left(x^k\right)}, \quad \forall k = 0, \ldots, K, \ \forall f'\left(x^k\right) \neq 0. \tag{5}$$

2.2.2. Convergence Results. Considering Taylor series expansion of function f and truncation error (e^k) between kth iterative approximate solution (x^k) and true zero (δ), that is, $e^k = x^k - \delta$ (with $\forall x^k \in I$ and $\delta \in I$), we have

$$\begin{aligned} f\left(x^k\right) &= f\left(\delta + e^k\right) \\ &= f(\delta) + c_1 e^k + c_2 \left(e^k\right)^2 + \cdots + \mathscr{H.O.T} \end{aligned} \tag{6}$$

with

$$c_n = \frac{f^{[n]}(\delta)}{n!}, \quad \forall k = 0, \ldots, K, \tag{7}$$

where $f^{[n]}(\delta)$ denotes n-order derivative of function f at point δ (with $n \in \mathbb{N}$), $n!$ is n-factorial, $\mathscr{H.O.T}$ represent the Higher-Order Terms which check: $\mathscr{H.O.T} \equiv o(\|e^k\|^n)$ or $\mathscr{H.O.T} \equiv \mathcal{O}(\|e^k\|^{n+1})$, $o(\cdot)$ and $\mathcal{O}(\cdot)$ are Landau notations associated with the asymptotic behavior of function f, and $\| \star \|$ is the square norm or Euclidean-norm associated with the quantity $\star$ (here, Euclidean distance reduces to absolute value $| \star |$, i.e., $\| \star \| \equiv | \star |$).

Since that $f(\delta) = 0$, (6)-(7) can be written as follows (with $\forall x^k \in I$):

$$f\left(x^k\right) = c_1 e^k + c_2 \left(e^k\right)^2 + \cdots + \mathscr{H.O.T}, \quad \forall k = 0, \ldots, K. \tag{8}$$

The first-order derivative associated with function f at point $x^k \in I$ (i.e., $f'(x^k)$) checks:

$$f'\left(x^k\right) = c_1 + 2c_2 e^k + 3c_3 \left(e^k\right)^2 + \cdots + \mathscr{H.O.T}, \quad \forall k = 0, \ldots, K. \tag{9}$$

Combining (5), (8)-(9), and $x^k = \delta + e^k$ leads to the following:

$$e^{k+1} = x^{k+1} - \delta = \frac{c_2}{c_1}\left(e^k\right)^2 + \cdots + \mathscr{H.O.T}, \quad \forall k = 0, \ldots, K. \tag{10}$$

An iterative numerical scheme has θth order of convergence with associated convergence rate τ to number δ (i.e., "true zero") if there exists $\theta \geq 1$ and $\tau \geq 0$ [25] such as

$$\lim_{k \to +\infty} \frac{\left|e^{k+1}\right|}{\left|e^k\right|^\theta} = \tau, \quad \forall k = 0, \ldots, K, \tag{11}$$

where $|\varphi|$ is the absolute-value function of variable φ (such as $|\varphi| = -\varphi$ when $\varphi < 0$, $|\varphi| = 0$ when $\varphi = 0$, and $|\varphi| = \varphi$ when $\varphi > 0$) and "lim" is the limit operator. It is important to emphasize that (i) if $\theta = 1$ and $\tau \in]0,1[$ then the convergence is linear (or "first-order" type, e.g., bisection or false position method); (ii) if $\theta = 1$ and $\tau = 1$ then the convergence is sublinear; (iii) if $\theta = 1$ and $\tau = 0$ (or $\theta > 1$, $\forall \tau \in]0,1[$) then the convergence is superlinear (e.g., secant method); (iv) if $\theta = 2$ (with $\tau \in]0,1[$) then the convergence is quadratic (or "second-order" type, e.g., CNM); (v) if $\theta = 3$ (with $\tau \in]0,1[$) then the convergence is cubic (or "third-order" type, e.g., TMNM; see Section 3.1). θth order of convergence means that the number of significant digits is θ-fold at each iteration k, for example, in the case of CNM (resp., TMNM), where $\theta = 2$ (resp., $\theta = 3$), the number of exact decimals doubles (resp., triples) at each iteration k.

Using (10)-(11), we can see that CNM has quadratic convergence ($\theta = 2$) with a rate of convergence [24]:

$$\tau = \tau_{\mathrm{CNM}} = \left|\frac{c_2}{c_1}\right| > 0, \quad \forall c_1, c_2 \neq 0. \tag{12}$$

2.3. New Numerical Technique (NNT)

2.3.1. Proposed Iterative Procedure. In this section, we propose a New Numerical Technique (NNT) for improving the accuracy associated with the approximate solution provided by CNM (see Section 2.2, which is used for finding the roots of scalar nonlinear equations. NNT is based on some practical geometric rules and requires only the determination of the local slope of the curve representing the considered nonlinear equation taking into account the first-order derivation (see [26–28]).

Here, we present the main steps associated with the development of NNT:

(i) We consider normal straight line N associated with the curve representing nonlinear equation f at point $x^k \in I$ (see [26, 27] and Figure 2):

$$N(x) = -\frac{1}{f'\left(x^k\right)}\left(x - x^k\right) + f\left(x^k\right), \quad \forall k = 0, \ldots, K, \ \forall f'\left(x^k\right) \neq 0. \tag{13}$$

(ii) We introduce straight line H having direction vector $\vec{w}$ which depends on the sum of direction vectors $\vec{u} + \vec{v}$ associated with normal ($\vec{u}$) and tangent ($\vec{v}$) straight lines passing by point $(x^k, f(x^k))$ (with $\forall x^k \in I$; see Figure 2), that is:

$$H(x) = \frac{1}{2}\left(f'\left(x^k\right) - \frac{1}{f'\left(x^k\right)}\right)\left(x - x^k\right) + f\left(x^k\right), \quad \forall k = 0, \ldots, K, \ \forall f'\left(x^k\right) \neq 0, \ \forall \left|f'\left(x^k\right)\right| \neq 1 \tag{14}$$

with

$$w_1 = u_1 + v_1 = \left(f'\left(x^k\right) - \frac{1}{f'\left(x^k\right)}\right), \qquad w_2 = u_2 + v_2 = 2, \tag{15}$$

where $w_1 = u_1 + v_1$ and $w_2 = u_2 + v_2$ are the components of direction vector $\vec{w}$ associated with straight line H in any orthonormal basis $\{\vec{e}_1, \vec{e}_2\}$.

(iii) Using CNM, we define kth iterative point $z^k \in I$ (with $\forall x \in I$; see (5) and also Figure 2):

$$z^k = x^k - \frac{f\left(x^k\right)}{f'\left(x^k\right)}, \quad \forall k = 0, \ldots, K, \ \forall f'\left(x^k\right) \neq 0. \tag{16}$$

(iv) We introduce also straight line D in the following form (with $\forall x \in I$; see Figure 2):

$$D(x) = -\frac{1}{\mathscr{M}^k}\left(x - z^k\right) \tag{17}$$

with

$$\mathscr{M}^k = \frac{1}{2}\left(f'\left(x^k\right) - \frac{1}{f'\left(x^k\right)}\right), \quad \forall k = 0, \ldots, K, \ \forall f'\left(x^k\right) \neq 0, \ \forall \left|f'\left(x^k\right)\right| \neq 1. \tag{18}$$

(v) Combining (14)-(15) and (17)-(18), we adopt the solution $x = q^k \in I$ of the equation $H(x) = D(x)$ (with $\forall x \in I$; see Figure 2), that is:

$$q^k = \frac{1}{\mathscr{M}^k + 1/\mathscr{M}^k}\left(\frac{z^k}{\mathscr{M}^k} + \mathscr{M}^k x^k - f\left(x^k\right)\right), \quad \forall k = 0, \ldots, K, \ \forall \mathscr{M}^k \neq 0. \tag{19}$$

According to (16) and (19), kth iterative point $q^k \in I$ associated with NNT (see Figure 2) can be rewritten (with $\forall x^k \in I$):

$$q^k = x^k - \frac{f\left(x^k\right)}{\mathscr{M}^k + 1/\mathscr{M}^k}\left(1 + \frac{1}{\mathscr{M}^k f'\left(x^k\right)}\right), \quad \forall k = 0, \ldots, K, \ \forall f'\left(x^k\right) \neq 0, \ \forall \mathscr{M}^k \neq 0. \tag{20}$$

The iterative numerical scheme associated with NNT is coupled with CNM and therefore $(k+1)$th iterative solution $x^{k+1} \in I$ can be written as follows (with, $\forall q^k, z^k \in I$; see Figure 2):

$$x^{k+1} = q^k \quad \text{if } [\mathrm{A}], \tag{21a}$$

$$x^{k+1} = z^k \quad \text{else}, \quad \forall k = 0, \ldots, K \tag{21b}$$

with the different conditions associated with the iterative solution [A] (with $\forall q^k, z^k, x^k \in I$):

(i) First condition [A1]: $\operatorname{sgn}(q^k) = \operatorname{sgn}(x^k)$ and $\operatorname{sgn}(f(q^k)) = \operatorname{sgn}(f(x^k))$ and $|f(q^k)| < |f(z^k)|$.

(ii) Second condition [A2]: $|f(q^k)| < |f(z^k)|$.

2.3.2. Convergence Analysis. Similar to that in Section 2.2.2, we analyze the convergence associated with NNT which is presented in Section 2.2.

Using (9) leads to the following:

$$\begin{aligned}\mathscr{M}^k &= \frac{1}{2}\left(f'\left(x^k\right) - \frac{1}{f'\left(x^k\right)}\right) \\ &= \frac{1}{2}\left(c_1 - \frac{1}{c_1}\right) + c_2 e^k + \frac{3}{2}c_3\left(e^k\right)^2 + \cdots \\ &\quad + \mathscr{H.O.T}, \\ &\quad \forall k = 0, \ldots, K, \ \forall c_1 \neq 0, \ \forall \left|c_1\right| \neq 1.\end{aligned} \tag{22}$$

In line with (22), we have the following:

$$\begin{aligned}\mathscr{M}^k + \frac{1}{\mathscr{M}^k} &= \left[\frac{1}{2}\left(c_1 - \frac{1}{c_1}\right) + \frac{2}{c_1 - 1/c_1}\right] + c_2 e^k \\ &\quad + \frac{3}{2}c_3\left(e^k\right)^2 + \cdots + \mathscr{H.O.T}, \\ &\quad \forall k = 0, \ldots, K, \ \forall c_1 \neq 0, \ \forall \left|c_1\right| \neq 1.\end{aligned} \tag{23}$$

On the other hand, we have the following:

$$\begin{aligned}\frac{z^k}{\mathscr{M}^k} &= \frac{2}{c_1 - 1/c_1}\left(\delta + \frac{c_2}{c_1}\left(e^k\right)^2\right) + \mathscr{H.O.T}, \\ &\quad \forall c_1 \neq 0, \ \forall \left|c_1\right| \neq 1, \\ \mathscr{M}^k x^k &= \frac{1}{2}\left(c_1 - \frac{1}{c_1}\right)\delta + \left(\frac{1}{2}\left(c_1 - \frac{1}{c_1}\right) + c_2\delta\right)e^k \\ &\quad + \cdots + \mathscr{H.O.T}, \\ &\quad \forall k = 0, \ldots, K, \ \forall c_1 \neq 0, \ \forall \left|c_1\right| \neq 1.\end{aligned} \tag{24}$$

Combining (8) and (24) leads to the following:

$$\begin{aligned}&\frac{z^k}{\mathscr{M}^k} + \mathscr{M}^k x^k - f\left(x^k\right) \\ &\quad = \left[\frac{2}{c_1 - 1/c_1} + \frac{1}{2}\left(c_1 - \frac{1}{c_1}\right)\right]\delta \\ &\quad + \left[\frac{1}{2}\left(c_1 - \frac{1}{c_1}\right) + c_2\delta - c_1\right]e^k + \cdots + \mathscr{H.O.T}, \\ &\quad \forall k = 0, \ldots, K, \ \forall c_1 \neq 0, \ \forall \left|c_1\right| \neq 1.\end{aligned} \tag{25}$$

According to (19) and using (23) and (25), it holds that

$$\begin{aligned}e^{k+1} &= x^{k+1} - \delta \\ &= \frac{1}{(1/2)\left(c_1 - 1/c_1\right) + 2/\left(c_1 - 1/c_1\right)}\left[\frac{1}{2}\left(c_1 - \frac{1}{c_1}\right)\right. \\ &\quad \left. + c_2\delta - c_1\right]e^k + \cdots + \mathscr{H.O.T}, \\ &\quad \forall k = 0, \ldots, K, \ \forall c_1 \neq 0, \ \forall \left|c_1\right| \neq 1.\end{aligned} \tag{26}$$

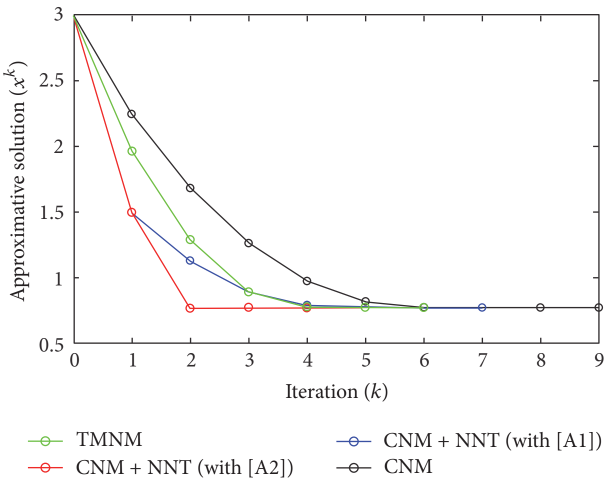

Figure 3: Evolution of approximate solution x^k compared to kth iteration for scalar nonlinear equation f_1 (when guest point $x^0 = 3$) obtained by CNM (black solid line with circles), TMNM (green solid line with circles), and CNM + NNT with the condition [A1] (blue solid line with circles) and condition [A2] (red solid line with circles).

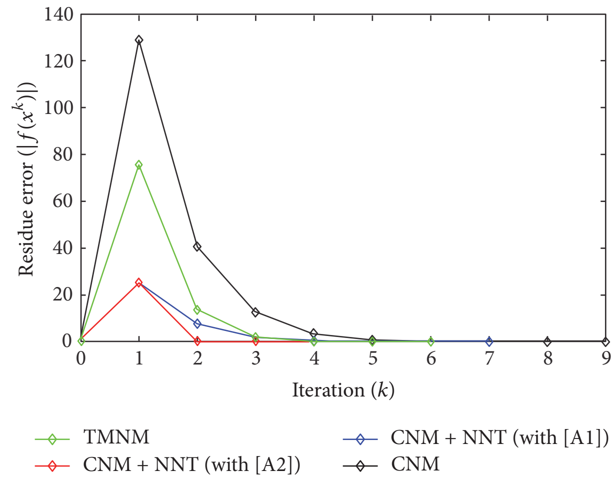

Figure 4: Evolution of residue error $|f(x^k)|$ compared to kth iteration for scalar nonlinear equation f_1 (when guest point $x^0 = 3$) obtained by CNM (black solid line with diamonds), TMNM (green solid line with diamonds), and CNM + NNT with condition [A1] (blue solid line with diamonds) and condition [A2] (red solid line with diamonds).

In line with (11) and (26), we can see that the rate of convergence τ associated with NNT is

$$\tau = \tau_{\text{NNT}} = \left| \frac{(1/2)(c_1 - 1/c_1) + c_2\delta - c_1}{(1/2)(c_1 - 1/c_1) + 2/(c_1 - 1/c_1)} \right|, \quad \forall c_1 \neq 0,\ \forall |c_1| \neq 1 \tag{27}$$

and the order of convergence is of linear-type (i.e., $\theta = 1$) if $0 < \tau_{\text{NNT}} \leq 1$ and quadratic-type (i.e., $\theta = 2$) if $\tau_{\text{NNT}} = 0$.

By taking (12) and (27), the rate of convergence $\tau_{\text{NNT-CNM}}$ of NNT combined with CNM is

$$\tau_{\text{NNT-CNM}} = \begin{cases} \tau_{\text{NNT}} & \text{if } [A] \\ \tau_{\text{CNM}} & \text{else.} \end{cases} \tag{28}$$

It is important to emphasize that the associated convergence order is linear-type ($\theta = 1$) if $0 < \tau_{\text{NNT}} \leq 1$ with condition [A] (i.e., [A1] or [A2]) and quadratic-type ($\theta = 2$) if $\tau_{\text{NNT}} = 0$ with condition [A] (i.e., [A1] or [A2]) and elsewhere with $\tau_{\text{CNM}} \neq 0$.

3. Numerical Examples

3.1. Some Preliminary Remarks. In this section, we propose to test, evaluate, and analyze the iterative numerical procedure presented in Section 2.3 on some particular examples. This New Numerical Technique (NNT) is coupled with Classical Newton's Method (CNM) in order to provide a more accurate approximate solution associated with scalar nonlinear equations. The numerical predictions obtained by combining both NNT and CNM are compared with those provided by Third-order Modified Newton's Method (TMNM) [22, 26]. All numerical results presented here have been made with Matlab software (see [25, 29–32]).

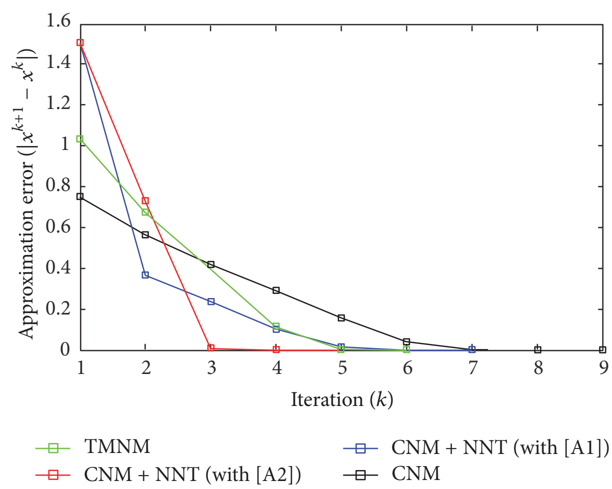

Figure 5: Evolution of approximation error $|x^{k+1} - x^k|$ compared to kth iteration for scalar nonlinear equation f_1 (when guest point $x^0 = 3$) obtained by CNM (black solid line with squares), TMNM (green solid line with squares), and CNM + NNT with condition [A1] (blue solid line with squares) and condition [A2] (red solid line with squares).

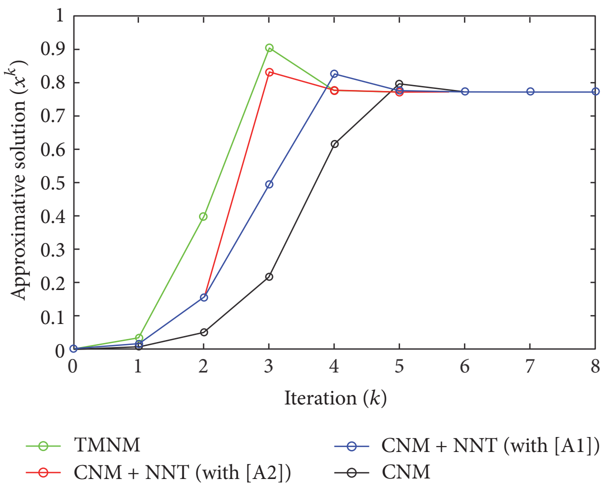

Figure 6: Evolution of approximate solution x^k compared to kth iteration for scalar nonlinear equation f_1 (when guest point $x^0 = 10^{-3}$) obtained by CNM (black solid line with circles), TMNM (green solid line with circles), and CNM + NNT with condition [A1] (blue solid line with circles) and condition [A2] (red solid line with circles).

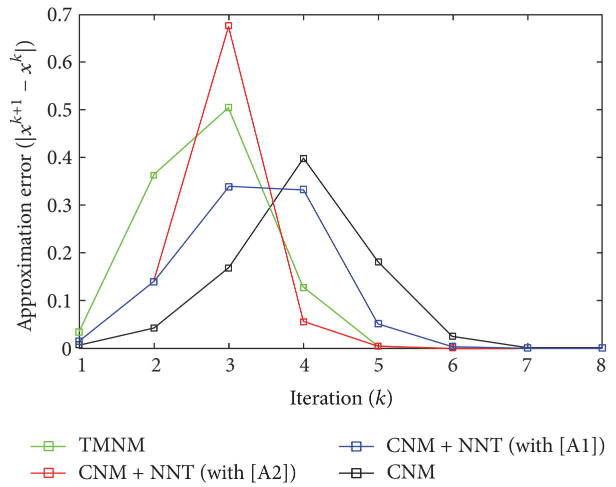

Figure 8: Evolution of approximation error $|x^{k+1} - x^k|$ compared to kth iteration for scalar nonlinear equation f_1 (when guest point $x^0 = 10^{-3}$) obtained by CNM (black solid line with squares), TMNM (green solid line with squares), and CNM + NNT with condition [A1] (blue solid line with squares) and condition [A2] (red solid line with squares).

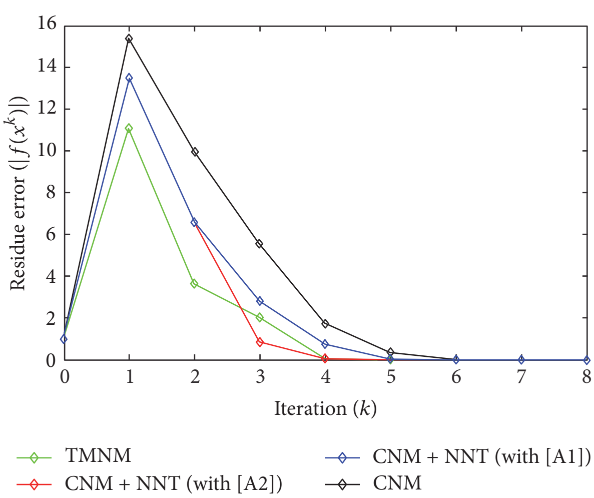

Figure 7: Evolution of residue error $|f(x^k)|$ compared to kth iteration for scalar nonlinear equation f_1 (when guest point $x^0 = 10^{-3}$) obtained by CNM (black solid line with diamonds), TMNM (green solid line with diamonds), and CNM + NNT with condition [A1] (blue solid line with diamonds) and condition [A2] (red solid line with diamonds).

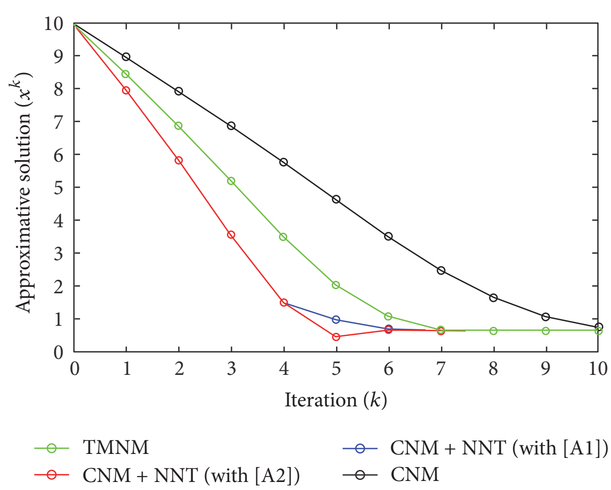

Figure 9: Evolution of approximate solution x^k compared to kth iteration for scalar nonlinear equation f_2 (when guest point $x^0 = 10$) obtained by CNM (black solid line with circles), TMNM (green solid line with circles), and CNM + NNT with condition [A1] (blue solid line with circles) and condition [A2] (red solid line with circles).

For stopping the iterative process associated with each considered algorithm, we adopt three coupled types of Convergence Criterion (CC):

$$\begin{aligned} &(\text{CC1}): m \leq K_{\max}, \\ &(\text{CC2}): \left|f\left(x^{k+1}\right)\right| \leq \epsilon_{\text{re}}, \\ &(\text{CC3}): \left|x^{k+1} - x^k\right| \leq \epsilon_{\text{ae}}, \\ &\qquad \forall k = 0, \ldots, K, \end{aligned} \tag{29}$$

where $K_{\max}$ denotes the maximum number of iterations and ϵ_{re} (resp., ϵ_{ae}) is the tolerance parameter associated with the residue (resp., approximation) error criterion. Here, the considered values for each CC are $K_{\max} = 10$, $\epsilon_{\text{re}} = 10^{-10}$, and $\epsilon_{\text{ae}} = 10^{-10}$.

The iterative numerical scheme associated with TMNM (see [22, 26]) is

$$x^{k+1} = x^k - \frac{f\left(x^k\right)}{f'\left(x^k\right)} - \frac{f''\left(x^k\right)\left(f\left(x^k\right)\right)^2}{2\left(f'\left(x^k\right)\right)^3}, \qquad \forall k = 0, \ldots, K, \ \forall f'\left(x^k\right) \neq 0, \tag{30}$$

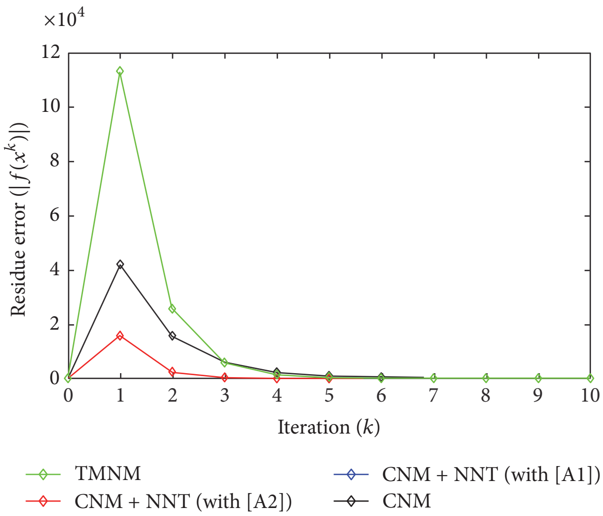

FIGURE 10: Evolution of residue error $|f(x^k)|$ compared to kth iteration for scalar nonlinear equation f_2 (when guest point $x^0 = 10$) obtained by CNM (black solid line with diamonds), TMNM (green solid line with diamonds), and CNM + NNT with condition [A1] (blue solid line with diamonds) and condition [A2] (red solid line with diamonds).

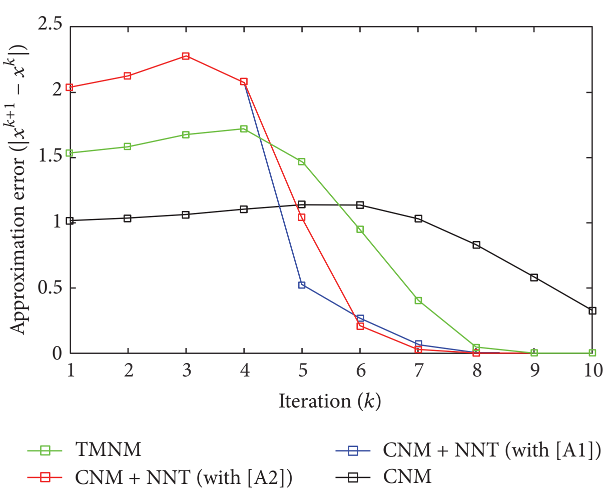

FIGURE 11: Evolution of approximation error $|x^{k+1} - x^k|$ compared to kth iteration for scalar nonlinear equation f_2 (when guest point $x^0 = 10$) obtained by CNM (black solid line with squares), TMNM (green solid line with squares), and CNM + NNT with condition [A1] (blue solid line with squares) and condition [A2] (red solid line with squares).

where $f''(x^k)$ denotes the second-order derivative of function f at point x^k. It should be noted that order of convergence θ is cubic (i.e., $\theta = 3$) and rate of convergence τ is (see [22, 26])

$$\tau = \tau_{\text{TMNM}} = \left| \frac{2(c_2)^2 - c_3}{c_1} \right| > 0, \quad \forall c_1 \neq 0. \tag{31}$$

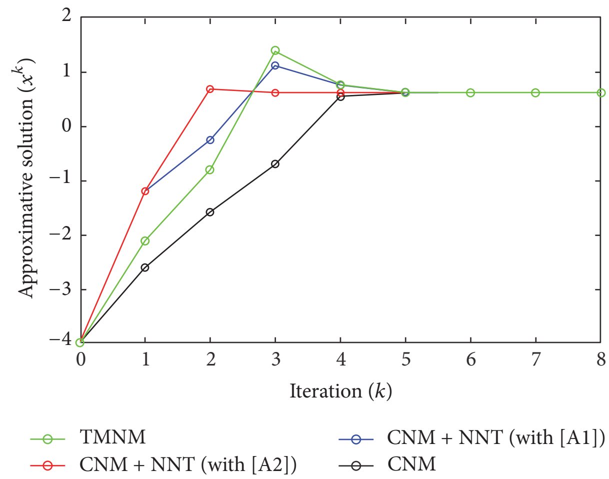

FIGURE 12: Evolution of approximate solution x^k compared to kth iteration for scalar nonlinear equation f_2 (when guest point $x^0 = -4$) obtained by CNM (black solid line with circles), TMNM (green solid line with circles), and CNM + NNT with condition [A1] (blue solid line with circles) and condition [A2] (red solid line with circles).

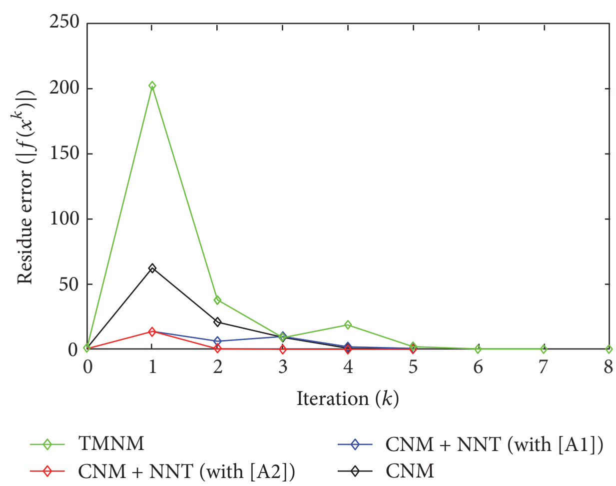

FIGURE 13: Evolution of residue error $|f(x^k)|$ compared to kth iteration for scalar nonlinear equation f_2 (when guest point $x^0 = -4$) obtained by CNM (black solid line with diamonds), TMNM (green solid line with diamonds), amd CNM + NNT with condition [A1] (blue solid line with diamonds) and condition [A2] (red solid line with diamonds).

3.2. Examples. We consider the following scalar nonlinear equations:

$$\begin{aligned} f_1(x) &= 5x^4 + 3\ln(x) - 1 = 0, \\ f_2(x) &= 3x^3 + 5\exp(x) - 10 = 0, \\ f_3(x) &= 5\exp(x) + x^3\cos(x) - 20 = 0, \end{aligned} \tag{32}$$

where $\exp(\cdot)$, $\ln(\cdot)$, and $\cos(\cdot)$ represent exponential, natural logarithm, and cosinus functions.

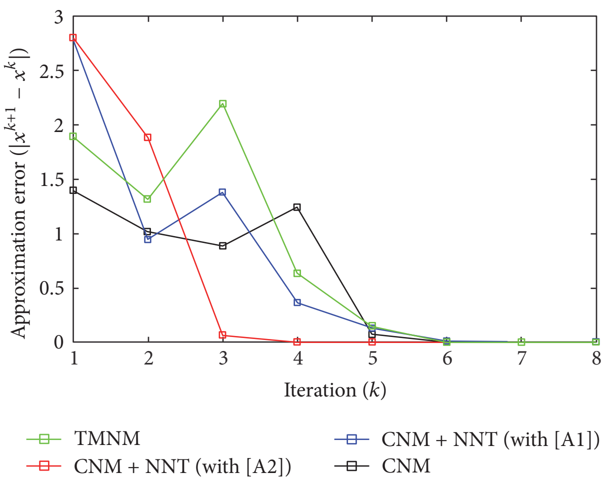

Figure 14: Evolution of approximation error $|x^{k+1} - x^k|$ compared to kth iteration for scalar nonlinear equation f_2 (when guest point $x^0 = -4$) obtained by CNM (black solid line with squares), TMNM (green solid line with squares), and CNM + NNT with condition [A1] (blue solid line with squares) and condition [A2] (red solid line with squares).

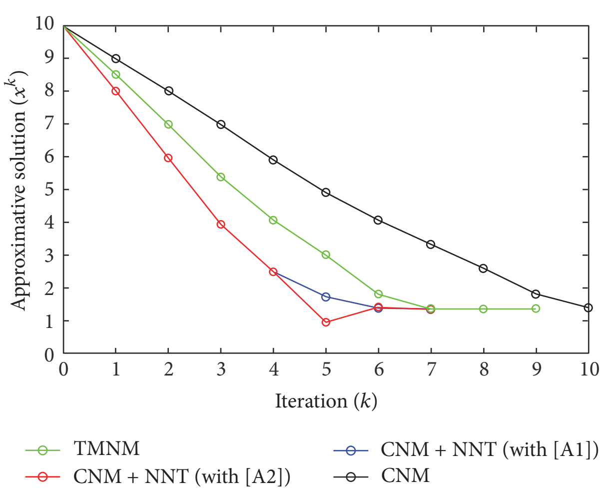

Figure 15: Evolution of approximate solution x^k compared to kth iteration for scalar nonlinear equation f_3 (when guest point $x^0 = 10$) obtained by CNM (black solid line with circles), TMNM (green solid line with circles), and CNM + NNT with condition [A1] (blue solid line with circles) and condition [A2] (red solid line with circles).

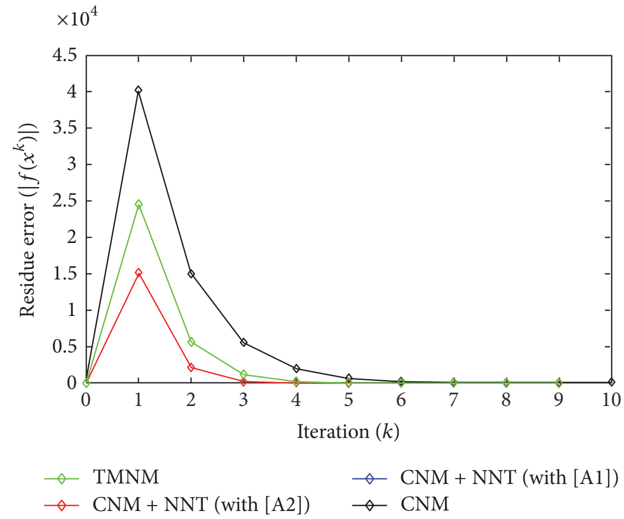

Figure 16: Evolution of residue error $|f(x^k)|$ compared to kth iteration for scalar nonlinear equation f_3 (when guest point $x^0 = 10$) obtained by CNM (black solid line with diamonds), TMNM (green solid line with diamonds), and CNM + NNT with condition [A1] (blue solid line with diamonds) and condition [A2] (red solid line with diamonds).

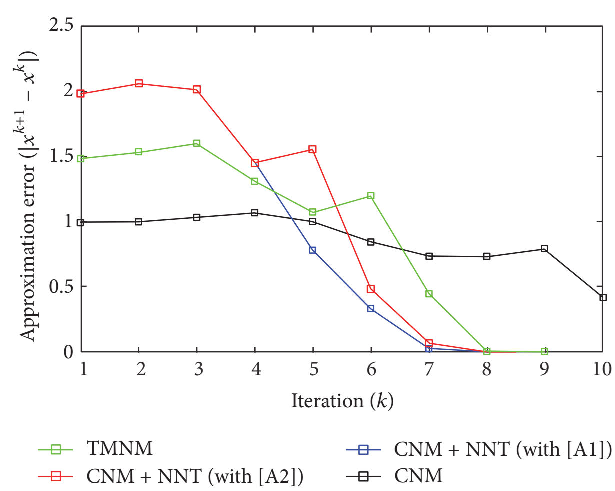

Figure 17: Evolution of approximation error $|x^{k+1} - x^k|$ compared to kth iteration for scalar nonlinear equation f_3 (when guest point $x^0 = 10$) obtained by CNM (black solid line with squares), TMNM (green solid line with squares), and CNM + NNT with condition [A1] (blue solid line with squares) and condition [A2] (red solid line with squares).

The approximate numerical solution of roots associated with functions f_1, f_2, and f_3 are, respectively,

$$\begin{aligned} \delta_1 &\approx 0.7720250256\cdots, \\ \delta_2 &\approx 0.6192449540\cdots, \\ \delta_3 &\approx 1.3595986211\cdots. \end{aligned} \tag{33}$$

3.3. Results and Discussion. All the numerical results associated with scalar nonlinear functions f_1 to f_3 are presented in Tables 1–6 and also in Figures 3–20. For each scalar nonlinear function f_j (with $j = 1, 2, 3$), we test two guest points x^0 for starting iterative procedure: (i) in the case of function f_1, we adopt the first-guest point $x^0 = 3$ (resp., second-guest point $x^0 = 10^{-3}$) and we show different approximate solutions x^k (with $k = 1, \ldots, 10$) and also the evolution of both residue error $|f(x^{k+1})|$ and approximation error $|x^{k+1} - x^k|$ provided by Classical Newton's Method (CNM; see Section 2.2), CNM coupled to the proposed New Numerical Technique (NNT) with both conditions [A1] and [A2] (CNM + NNT; see Section 2.3), and Third-order Modified Newton's Method (TMNM; see Section 3.1) in Table 1 (resp., Table 2) and

TABLE 1: Approximate solution x^k of scalar nonlinear equation f_1 (when guest point $x^0 = 3$) obtained by Classical Newton's Method (CNM) (cf. Section 2.2), Classical Newton's Method coupled with New Numerical Technique (CNM + NNT) for conditions [A1] and [A2] (cf. Section 2.3), and Third-order Modified Newton's Method (TMNM) (cf. Section 3.1). Notations: $(**)$ (resp., $(*)$) denotes that approximate solution x^k is provided using (21a) (resp., (21b)) in Section 2.3.1.

Iteration (k)	CNM	CNM + NNT with condition [A1]	CNM + NNT with condition [A2]	TMNM
0	3.0000000000	3.0000000000	3.0000000000	3.0000000000
1	2.2471426305	1.4942904056 $(**)$	1.4942904056 $(**)$	1.9644439737
2	1.6823826593	1.1286469200 $(*)$	0.7631581663 $(**)$	1.2884809024
3	1.2337388302	0.8906377962 $(*)$	0.7721173336 $(*)$	0.8906149334
4	0.9723200229	0.7878458749 $(*)$	0.7720250356 $(*)$	0.7746164630
5	0.8146707520	0.7723183427 $(*)$	0.7720250256 $(*)$	0.7720250503
6	0.7741430851	0.7720251266 $(*)$		0.7720250256
7	0.7720302909	0.7720250256 $(*)$		
8	0.7720250256			
9	0.7720250256			
10				

TABLE 2: Approximate solution x^k of scalar nonlinear equation f_2 (when guest point $x^0 = 10^{-3}$) obtained by Classical Newton's Method (CNM) (cf. Section 2.2), Classical Newton's Method coupled with New Numerical Technique (CNM + NNT) for conditions [A1] and [A2] (cf. Section 2.3), and Third-order Modified Newton's Method (TMNM) (cf. Section 3.1). Notations: $(**)$ (resp., $(*)$) denotes that approximate solution x^k is provided using (21a) (resp., (21b)) in Section 2.3.1.

Iteration (k)	CNM	CNM + NNT with condition [A1]	CNM + NNT with condition [A2]	TMNM
0	0.0010000000	0.0010000000	0.0010000000	0.0010000000
1	0.0082410886	0.0154821756 $(**)$	0.0154821756 $(**)$	0.0344577770
2	0.0505339927	0.1548613133 $(**)$	0.1548613133 $(**)$	0.3980194000
3	0.2182202963	0.4938846484 $(*)$	0.8311197179 $(**)$	0.9031891893
4	0.6163033790	0.8261543800 $(*)$	0.7760693275 $(*)$	0.7755098327
5	0.7975366151	0.7754244611 $(*)$	0.7720442199 $(*)$	0.7720250858
6	0.7727863680	0.7720385875 $(*)$	0.7720250260 $(*)$	0.7720250256
7	0.7720257060	0.7720250258 $(*)$	0.7720250256 $(*)$	
8	0.7720250256	0.7720250256 $(*)$		
9				
10				

TABLE 3: Approximate solution x^k of scalar nonlinear equation f_2 (when guest point $x^0 = 10$) obtained by Classical Newton's Method (CNM) (cf. Section 2.2), Classical Newton's Method coupled with New Numerical Technique (CNM + NNT) for conditions [A1] and [A2] (cf. Section 2.3), and Third-order Modified Newton's Method (TMNM) (cf. Section 3.1). Notations: $(**)$ (resp., $(*)$) denotes that approximate solution x^k is provided using (21a) (resp., (21b)) in Section 2.3.1.

Iteration (k)	CNM	CNM + NNT with condition [A1]	CNM + NNT with condition [A2]	TMNM
0	10.000000000	10.000000000	10.000000000	10.000000000
1	8.9811766535	7.9623533071 $(**)$	7.9623533071 $(**)$	8.4655416573
2	7.9456740785	5.8372149905 $(**)$	5.8372149905 $(**)$	6.8799380466
3	6.8825672128	3.5600762341 $(**)$	3.5600762341 $(**)$	5.2048513181
4	5.7803883041	1.4821734803 $(**)$	1.4821734803 $(**)$	3.4865001565
5	4.6404493806	0.9609028311 $(*)$	0.4402289741 $(**)$	2.0197617679
6	3.5056070182	0.6928062817 $(*)$	0.6483081663 $(*)$	1.0685736902
7	2.4744259573	0.6233620739 $(*)$	0.6199084714 $(*)$	0.6632122684
8	1.6461067325	0.6192585092 $(*)$	0.6192453069 $(*)$	0.6193191836
9	1.0635449602	0.6192449541 $(*)$	0.6192449540 $(*)$	0.6192449540
10	0.7354406002	0.6192449510 $(*)$		0.6192449540

TABLE 4: Approximate solution x^k of scalar nonlinear equation f_2 (when guest point $x^0 = -4$) obtained by Classical Newton's Method (CNM) (cf. Section 2.2), Classical Newton's Method coupled with New Numerical Technique (CNM + NNT) for conditions [A1] and [A2] (cf. Section 2.3), and Third-order Modified Newton's Method (TMNM) (cf. Section 3.1). Notations: $(**)$ (resp., $(*)$) denotes that approximate solution x^k is provided using (21a) (resp., (21b)) in Section 2.3.1.

Iteration (k)	CNM	CNM + NNT with condition [A1]	CNM + NNT with condition [A2]	TMNM
0	−4.0000000000	−4.0000000000	−4.0000000000	−4.0000000000
1	−2.5987493208	−1.1976336149 $(**)$	−1.1976336149 $(**)$	−2.1088093909
2	−1.5803235380	−0.2513579056 $(*)$	0.6858578709 $(**)$	−0.7930763263
3	−0.6950023819	1.1303884977 $(*)$	0.6226379112 $(*)$	1.4022343921
4	0.5489290102	0.7665675348 $(*)$	0.6192541648 $(*)$	0.7698587570
5	0.6234126286	0.6349082108 $(*)$	0.6192449540 $(*)$	0.6217981372
6	0.6192588436	0.6194395336 $(*)$	0.6192449540 $(*)$	0.6192449694
7	0.6192449541	0.6192449843 $(*)$		0.6192449540
8	0.6192449540	0.6192449540 $(*)$		0.6192449540
9				
10				

TABLE 5: Approximate solution x^k of scalar nonlinear equation f_3 (when guest point $x^0 = 10$) obtained by Classical Newton's Method (CNM) (cf. Section 2.2), Classical Newton's Method coupled with New Numerical Technique (CNM + NNT) for conditions [A1] and [A2] (cf. Section 2.3), and Third-order Modified Newton's Method (TMNM) (cf. Section 3.1). Notations: $(**)$ (resp., $(*)$) denotes that approximate solution x^k is provided using (21a) (resp., (21b)) in Section 2.3.1.

Iteration (k)	CNM	CNM + NNT with condition [A1]	CNM + NNT with condition [A2]	TMNM
0	10.000000000	10.000000000	10.000000000	10.000000000
1	9.0104267607	8.0208535216 $(**)$	8.0208535216 $(**)$	8.5171506533
2	8.0146859130	5.9615544966 $(**)$	5.9615544966 $(**)$	6.9859301990
3	6.9847503720	3.9493415483 $(**)$	3.9493415483 $(**)$	5.3851997390
4	5.9189110435	2.4996523082 $(**)$	2.4996523082 $(**)$	4.0778432768
5	4.9180876708	1.7226271036 $(*)$	0.9467658294 $(**)$	3.0067861343
6	4.0738875311	1.3881795157 $(*)$	1.4273487664 $(*)$	1.8098899052
7	3.3404588095	1.3598140013 $(*)$	1.3607723321 $(*)$	1.3648044141
8	2.6091363141	1.3595986336 $(*)$	1.3595989925 $(*)$	1.3595986468
9	1.8194405679	1.3595986211 $(*)$	1.3595986211 $(*)$	1.3595986211
10	1.4045600984			

TABLE 6: Approximate solution x^k of scalar nonlinear equation f_3 (when guest point $x^0 = -1$) obtained by Classical Newton's Method (CNM) (cf. Section 2.2), Classical Newton's Method coupled with New Numerical Technique (CNM + NNT) for conditions [A1] and [A2] (cf. Section 2.3), and Third-order Modified Newton's Method (TMNM) (cf. Section 3.1). Notations: $(**)$ (resp., $(*)$) denotes that approximate solution x^k is provided using (21a) (resp., (21b)) in Section 2.3.1.

Iteration (k)	CNM	CNM + NNT with condition [A1]	CNM + NNT with condition [A2]	TMNM
0	−1.0000000000	−1.0000000000	−1.0000000000	−1.0000000000
1	6.1409303723	6.1409303723 $(*)$	6.1409303723 $(*)$	14.5343055115
2	5.1148208484	4.0887116616 $(**)$	4.0887116616 $(**)$	13.0337347682
3	4.2379703013	2.6194618995 $(**)$	2.6194618995 $(**)$	11.5318478151
4	3.4892766889	1.8291595323 $(*)$	1.0397597354 $(**)$	10.0370297219
5	2.7676801225	1.4064394475 $(*)$	1.3966765848 $(*)$	8.5556896133
6	1.9776897481	1.3601688047 $(*)$	1.3599586541 $(*)$	7.0203746257
7	1.4417009128	1.3595987088 $(*)$	1.3595986561 $(*)$	5.3974911835
8	1.3613036994	1.3595986211 $(*)$	1.3595986211 $(*)$	4.1321703085
9	1.3595994046			3.1258556869
10	1.3595986211			1.9485648236

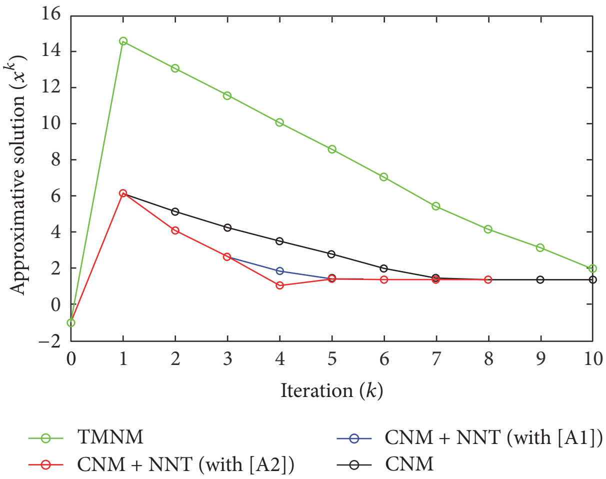

FIGURE 18: Evolution of approximate solutions x^k compared to kth iteration for scalar nonlinear equation f_2 (when guest point $x^0 = -1$) obtained by CNM (black solid line with circles), TMNM (green solid line with circles), and CNM + NNT with condition [A1] (blue solid line with circles) and condition [A2] (red solid line with circles).

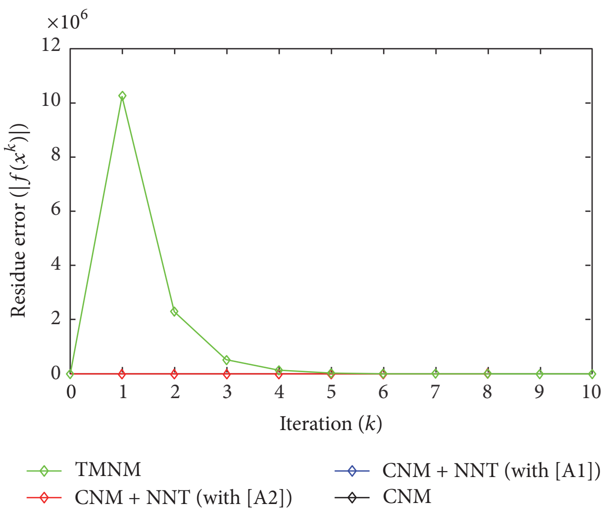

FIGURE 19: Evolution of residue error $|f(x^k)|$ compared to kth iteration for scalar nonlinear equation f_3 (when guest point $x^0 = -1$) obtained by CNM (black solid line with diamonds), TMNM (green solid line with diamonds), and CNM + NNT with condition [A1] (blue solid line with diamonds) and condition [A2] (red solid line with diamonds).

Figures 3–5 (resp., Figures 6–8); (ii) in the case of function f_2, we adopt first-guest point $x^0 = 10$ (resp., second-guest point $x^0 = -4$) and we show different approximate solutions x^k (with $k = 1, \ldots, 10$) provided by CNM, CNM + NNT with both conditions [A1] and [A2], and TMNM in Table 3 (resp., Table 4) and Figures 9–11 (resp., Figures 12–14); (iii) in the case of function f_3, we adopt first-guest point $x^0 = 10$ (resp., second-guest point $x^0 = -1$) and we show different approximate solutions x^k (with $k = 1, \ldots, 10$) provided by CNM, CNM + NNT with both conditions [A1] and [A2],

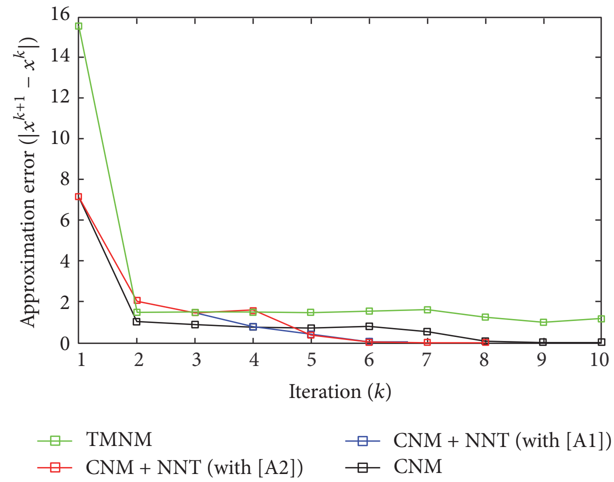

FIGURE 20: Evolution of approximation error $|x^{k+1} - x^k|$ compared to kth iteration for scalar nonlinear equation f_3 (when guest point $x^0 = -1$) obtained by CNM (black solid line with squares), TMNM (green solid line with squares), and CNM + NNT with condition [A1] (blue solid line with squares) and condition [A2] (red solid line with squares).

and TMNM in Table 5 (resp., Table 6) and Figures 15–17 (resp., Figures 18–20). The obtained numerical results emphasize clearly that NNT combined with CNM is able to provide in the vast majority of cases a better approximate solution than that supplied only by CNM. In addition, on all results presented, condition [A2] seems to be more convenient than condition [A1]. For summary, NNT presents the main advantages of being a procedure: (i) "accurate" in terms of achieved approximate solution; (ii) "straightforward" to implement in computation software.

4. Concluding Comments

This study is devoted to a New Numerical Technique (NNT) to improve the accuracy of approximate solution provided by Classical Newton's Method (CNM) and afford to have better numerical evaluation of the roots associated with the scalar nonlinear equations. As in CNM, this NNT requires only the determination of the first-order derivative of the nonlinear function under consideration. The predictive capabilities associated with NNT are shown on some examples.

Competing Interests

The author declares that they have no competing interests.

References

[1] Q. S. Nguyen, *Stability and Nonlinear Solid Mechanics*, John Wiley & Sons, New York, NY, USA, 2000.

[2] I. Doghri, *Mechanics of Deformable Solids: linear, Nonlinear, Analytical And Computational Aspects*, Springer, Berlin, Germany, 2000.

[3] A. Curnier, *Méthodes Numériques en Mécanique des Solides*, Presses Polytechniques et Universitaires Romandes, 2000.

[4] R. Parnes, *Solid Mechanics in Engineering*, Wiley, Hoboken, NJ, USA, 2001.

[5] J. Besson, G. Cailletaud, J.-L. Chaboche, and S. Forest, *Non-Linear Mechanics of Materials*, vol. 167 of *Solid Mechanics and Its Applications*, Springer, 2010.

[6] R. De Borst, M. A. Crisfield, J. J. C. Remmers, and C. V. Verhoosel, *Non-Linear Finite Element Analysis of Solids and Structures. Computational Mechanics*, Wiley-Blackwell, Hoboken, NJ, USA, 2012.

[7] M. Bonnet, A. Frangi, and C. Rey, *The Finite Element Method in Solid Mechanics*, McGraw-Hill Education, 2014.

[8] M. Kojić and K. J. Bathe, *Inelastic Analysis of Solids and Structures*, Computational Fluid and Solid Mechanics, Springer, 2005.

[9] S. Abbasbandy, "Improving Newton-Raphson method for nonlinear equations by modified Adomian decomposition method," *Applied Mathematics and Computation*, vol. 145, no. 2-3, pp. 887–893, 2003.

[10] M. A. Noor, "New family of iterative methods for nonlinear equations," *Applied Mathematics and Computation*, vol. 190, no. 1, pp. 553–558, 2007.

[11] K. I. Noor, M. A. Noor, and S. Momani, "Modified Householder iterative method for nonlinear equations," *Applied Mathematics and Computation*, vol. 190, no. 2, pp. 1534–1539, 2007.

[12] M. A. Noor, "Some iterative methods free from second derivatives for nonlinear equations," *Applied Mathematics and Computation*, vol. 192, no. 1, pp. 101–106, 2007.

[13] M. A. Noor, "New classes of iterative methods for nonlinear equations," *Applied Mathematics and Computation*, vol. 191, no. 1, pp. 128–131, 2007.

[14] C. Chun, "A family of composite fourth-order iterative methods for solving nonlinear equations," *Applied Mathematics and Computation*, vol. 187, no. 2, pp. 951–956, 2007.

[15] B. Neta and A. N. Johnson, "High-order nonlinear solver for multiple roots," *Computers & Mathematics with Applications*, vol. 55, no. 9, pp. 2012–2017, 2008.

[16] A. Cordero and J. R. Torregrosa, "A class of multi-point iterative methods for nonlinear equations," *Applied Mathematics and Computation*, vol. 197, no. 1, pp. 337–344, 2008.

[17] T.-F. Li, D.-S. Li, Z.-D. Xu, and Y.-l. Fang, "New iterative methods for non-linear equations," *Applied Mathematics and Computation*, vol. 197, no. 2, pp. 755–759, 2008.

[18] C. Chun, "A simply constructed third-order modifications of Newton's method," *Journal of Computational and Applied Mathematics*, vol. 219, no. 1, pp. 81–89, 2008.

[19] Y. M. Ham, C. Chun, and S.-G. Lee, "Some higher-order modifications of Newton's method for solving nonlinear equations," *Journal of Computational and Applied Mathematics*, vol. 222, no. 2, pp. 477–486, 2008.

[20] L. Fang and G. He, "Some modifications of Newton's method with higher-order convergence for solving nonlinear equations," *Journal of Computational and Applied Mathematics*, vol. 228, no. 1, pp. 296–303, 2009.

[21] C. Chun and B. Neta, "Some modification of Newton's method by the method of undetermined coefficients," *Computers & Mathematics with Applications*, vol. 56, no. 10, pp. 2528–2538, 2008.

[22] G. Antoni, "A new Newton-type method for solving both nonlinear equations and systems," *Asian Journal of Mathematics and Computer Research*, vol. 6, no. 3, pp. 193–212, 2015.

[23] G. Antoni, "A new class of two-step iterative algorithms for finding roots of nonlinear equations," *Asian Journal of Mathematics and Computer Research*, vol. 7, no. 3, pp. 175–189, 2015.

[24] C. T. Kelley, *Solving Nonlinear Equations with Newton's Method*, vol. 1 of *Fundamental Algorithms for Numerical Calculations*, Society for Industrial and Applied Mathematics, Philadelphia, Pa, USA, 2003.

[25] A. Quarteroni, R. Sacco, and F. Saleri, *Méthodes Numériques pour le Calcul Scientifique: Programmes en MATLAB*, Springer, Berlin, Germany, 2000.

[26] G. Antoni, "A new iterative algorithm for approximating zeros of nonlinear scalar equations: geometry-based solving procedure," *Asian Journal of Mathematics and Computer Research*, vol. 10, no. 2, pp. 78–97, 2015.

[27] G. Antoni, "A new accurate and efficient iterative numerical method for solving the scalar and vector nonlinear equations: approach based on geometric considerations," *International Journal of Engineering Mathematics*, vol. 2016, Article ID 6390367, 18 pages, 2016.

[28] G. Antoni, "A simple and efficient root-finding algorithm for dealing with scalar nonlinear equations: iterative procedure based on geometric considerations," *Asian Journal of Mathematics and Computer Research*, In press.

[29] J. H. Mathews and K. K. Fink, *Numerical Methods Using Matlab*, Pearson, 2004.

[30] W. Y. Yang, W. Cao, T. S. Chung, and J. Morris, *Applied Numerical Methods Using MATLAB*, Wiley-Interscience, New York, NY, USA, 2005.

[31] J. Kiusalaas, *Numerical Methods in Engineering with MATLAB*, Cambridge University Press, Cambridge, UK, 2009.

[32] S. C. Chapra, *Applied Numerical Methods with MATLAB for Engineers and Scientists*, McGraw-Hill Higher Education, 2011.