# A simple algorithm for high order Newton iteration formulae and some new variants

Mustafa Turkyilmazoglu [ID]

*Department of Mathematics, Hacettepe University, 06800 Beytepe, Ankara, Turkey*

## Abstract

The high order Newton iteration formulas are revisited in this paper. Translating the nonlinear root finding problem into a fixed point iteration involving an unknown general function whose root is searched, a double Taylor series is undertaken regarding the root and the root finding function. Based on the error analysis of the expansion, a simple algorithm is later proposed to construct Newton iteration formulae of any order commencing from the traditional linearly convergent fixed point iteration method and quadratically convergent Newton-Raphson method of frequently at the disposal of the scientific community. It is shown that the well-known variants like the Halley's method or Haouseholder's methods of high order can be reproduced from the general case outlined here. Some further rare single-step classes of any order are shown to be derivable from the presented algorithm. Finally, some new higher order accurate variants are also offered taking into account multi-step compositions which demand less computation of higher derivatives. The efficiency, accuracy and performance of the proposed methods and also their potential advantages over the classical ones are numerically demonstrated and discussed on some well-documented examples from the open literature.

## 1. Introduction

The open problem of searching for the best iteration formula for finding the root of a given nonlinear function in terms of its accuracy and of the least computational operation has attracted many investigators from plenty branches of science, see for instance [2,13,18, 29]. Therefore, an abundant variants of the classical Newton iteration formulae has been developed in the literature. The current paper is also devoted to the derivation of Newton iteration formulae and their well-known cousins, but from a different mathematical point of view, yielding a simple algorithm to generate the well-documented iterative schemes from which several new classes competing with the known ones are also shown to be derivable.

Nowadays, the development of this particular topic of numerical analysis is spectacular, with many researchers designing optimal higher-order iterative methods for solving nonlinear equations. In the literature there are a lot of iterative methods of any order

of convergence and many of them are optimal in some sense. After the introduction of contemporary software programs capable of implementing symbolic evaluations, last two decades witnessed many new variants of the classical Newton iteration schemes towards gaining the best performing one having the least error and high convergence order. The reason of richness of the literature on the variants of Newton's method is due to a unique constraint on the functional form of the root finding function and owing to this fact the investigation of new and novel iteration methods will keep forming a versatile field. Single-level and multi-level new iterations were hence proposed by investigators. To cite a few of recent single level techniques, quadratic convergence techniques were outlined in [5, 7, 24, 28]. The references [1, 4, 12, 22] dealt with modified and improved Newton iterations possessing cubical rate of convergence. Additionally, the new variants having quartic and higher order convergence properties were thoroughly illuminated in the publications [8, 17, 26, 27]. In order to avoid from the evaluation of higher derivatives generally appearing in the classical Newton iteration formulas, some multi-level new classes were suggested and analyzed in the references, [6, 14, 15, 19, 21, 25, 30]. Some derivative-free methods while preserving the convergence order were also developed in the recent studies [9] and [11], amongst many other earlier variants.

Although, the high order Newton iteration schemes for the nonlinear root finding problems known as the Householder's iterations, the Halley's iteration, the Schröder's iterations and the König's iterations are well-documented in the literature, a proper procedure of deriving them all in a single, simple and acute algorithm is missing, except the attempt in [20] based on some complicated notion of the inverse function theorem and differential operators. The present motivation is to fill this lacking and derive the whole classical Newton iterations from a procedural algorithm based on double series expansion at the root and also at the function whose root is targeted. An error analysis is established to construct the algorithm reflecting correctly the order of the convergence of the studied Newton method. The presented algorithm also gives a proof of why the classical Newton algorithms of desired order are optimum in their formulae. The Theorem underlying the algorithm may be thought of reinterpretation of the classical Schröder-Traubs theorem [29]. New single-step classes having any order of convergence rate are also proposed from the given algorithm. Moreover, new multi-step variants in combination with the knowns are suggested. Their error analysis were also fulfilled on the mathematical basis. Their accuracy and performance are eventually tested on some selected well-known test examples available from the open literature.

## 2. Methodology

We want to approximate the root of a nonlinear function $f(x)$, that occurs at the point $x = \alpha$, so

$$f(\alpha) \;\; = \;\; 0. \tag{2.1}$$

Dissimilar to the previous publications in the field, in place of stating a formula for the approximation and then checking out its convergence rate, we work out the Newton's high order formulas from the error analysis beforehand considering the convergence rates. The following such a methodology may be called as error elimination.

To start with, it is known that the root of $f(x)$ is also the fixed point of

$$x \;\; = \;\; g(x) \;\; = \;\; x + F(f(x), x), \tag{2.2}$$

where $F$ might be any function satisfying only the constraint

$$F(f(\alpha), \alpha) \;\; = \;\; F(0, \alpha) \;\; = \;\; 0. \tag{2.3}$$

We assume that $F$ has as many derivatives of high order as $f$. With an initial guess $x_0$ to the root $x = \alpha$, we can write the following fixed point iteration from (2.2)

$$x_{n+1} = g(x_n) = x_n + F(f(x_n), x_n), \quad n \geq 0. \tag{2.4}$$

Let us now expand $F$ near the root $\alpha$ into its Taylor's polynomial so that (2.4) turns out to be

$$
\begin{aligned}
x_{n+1} = {} & x_n + F(0, \alpha) + \frac{(x_n - \alpha)}{1!} \frac{dF}{dx}\big|_{x=\alpha} + \frac{(x_n - \alpha)^2}{2!} \frac{d^2 F}{dx^2}\big|_{x=\alpha} + \cdots \\
& + \frac{(x_n - \alpha)^n}{n!} \frac{d^n F}{dx^n}\big|_{x=\alpha} + R(\zeta),
\end{aligned} \tag{2.5}
$$

where the remainder

$$R(\zeta) = \frac{(x_n - \alpha)^{n+1}}{(n+1)!} \frac{d^{n+1} F}{dx^{n+1}}\big|_{x=\zeta}$$

will be omitted from now on with $\zeta \in (x_n, \alpha)$.

In view of (2.3) and defining the point-wise error

$$e_{n+1} = x_{n+1} - \alpha,$$

(2.5) can be rewritten as

$$e_{n+1} = e_n \Big[ 1 + \frac{dF}{dx}\big|_{x=\alpha} + \frac{e_n}{2!} \frac{d^2 F}{dx^2}\big|_{x=\alpha} + \cdots + \frac{e_n^{n-1}}{n!} \frac{d^n F}{dx^n}\big|_{x=\alpha} \Big]. \tag{2.6}$$

As mentioned, in general we do not have the knowledge of function $F$, except the constraint (2.3). However, this constraint helps us derive the Taylor expansion of $F$ near the point $f(\alpha) = 0$ in the format

$$F(f(x), x) = F(0, x) + \frac{f(x)}{1!} \frac{\partial F}{\partial f}\big|_{f=0} + \frac{f(x)^2}{2!} \frac{\partial^2 F}{\partial f^2}\big|_{f=0} + \cdots + \frac{f(x)^n}{n!} \frac{\partial^n F}{\partial f^n}\big|_{f=0}, \tag{2.7}$$

in which the remainder is omitted again.

Since the partial derivatives in (2.7) are depending on $x$, we may write (2.7) in a more convenient form as

$$
\begin{aligned}
F(f(x), x) = {} & -A_1(x)\Big(\frac{f(x)}{f'(x)}\Big) - A_2(x)\Big(\frac{f(x)}{f'(x)}\Big)^2 - A_3(x)\Big(\frac{f(x)}{f'(x)}\Big)^3 - \cdots \\
& - A_n(x)\Big(\frac{f(x)}{f'(x)}\Big)^n,
\end{aligned} \tag{2.8}
$$

where $A_n(x)$ are to be determined.

The accuracy of the iterative scheme (2.4) is measured by

$$\lim_{n \to \infty} \frac{e_{n+1}}{e_n^p} = C, \tag{2.9}$$

where $p$ is the order of accuracy of the method and $C$ is the asymptotic error. We observe from (2.6) that

$$\frac{e_{n+1}}{e_n} = \Big[ 1 + \frac{dF}{dx}\big|_{x=\alpha} + \frac{e_n}{2!} \frac{d^2 F}{dx^2}\big|_{x=\alpha} + \cdots + \frac{e_n^{n-1}}{n!} \frac{d^n F}{dx^n}\big|_{x=\alpha} \Big]. \tag{2.10}$$

Hence, if we require the iterative scheme in (2.5) to be linear with $p = 1$, then from (2.8), (2.9) and (2.10) we have

$$C = 1 - A_1(x) \tag{2.11}$$

and the corresponding iterative scheme in (2.5) will be simply the linear fixed point iteration (Euler) with $A_1(x)$ being still arbitrary.

Now, if we require that the iterative method in (2.5) is of second order or quadratic with $p = 2$ in (2.9), then the error in (2.11) should be eliminated resulting in

$$A_1(x) = 1. \tag{2.12}$$

So, $F$ must be now

$$F(f(x), x) = -\left(\frac{f(x)}{f'(x)}\right) - A_2(x)\left(\frac{f(x)}{f'(x)}\right)^2 - A_3(x)\left(\frac{f(x)}{f'(x)}\right)^3 - \cdots$$
$$-A_n(x)\left(\frac{f(x)}{f'(x)}\right)^n. \tag{2.13}$$

Therefore, at the quadratic level of approximation, if we demand the least evaluations the remaining coefficients in (2.13) must be disregarded leading to the classical Newton's iteration from (2.4) or (2.5)

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}. \tag{2.14}$$

The above methodology is also a procedural proof that the classical Newton's method is the most optimum one in terms of quadratic convergence and least computational demands. Otherwise, recalling that $F$ in (2.4) can be any arbitrary function, all other forms of quadratic iterative schemes must have the term $-\frac{f(x_n)}{f'(x_n)}$ in their leading-order series expansion, therefore it may be presumed that

$$F = F\left(\frac{f(x)}{f'(x)}, x\right)$$

for the quadratic convergence to hold. Indeed, we may have numerous variants, amongst many others, such as the following one-parameter families

$$x_{n+1} = x_n - \frac{f(x_n)}{hf(x_n) + f'(x_n)} = x_n - \frac{f(x_n)}{f'(x_n)}\left(\frac{1}{1 + h\frac{f(x_n)}{f'(x_n)}}\right),$$
$$x_{n+1} = x_n + \frac{1}{2h}\ln\left[\left(1 - h\frac{f(x_n)}{f'(x_n)}\right)^2\right], \tag{2.15}$$
$$x_{n+1} = x_n - \frac{1}{h}x_n\left(1 - e^{-h\frac{f(x_n)}{x_n f'(x_n)}}\right),$$
$$\vdots$$

in which $h$ is a real parameter. We see that the quadratic iterative schemes in (2.15) are not optimum in the sense that either they require more functional evaluations or more arithmetic operations to pass to the next iterate, as compared with (2.14). On the other hand, it should be alerted that the iterative schemes proposed in (2.15) might have some explicit advantages where the classical Newton's method (2.14) fails. For instance, at the turning points or for the initial guess taken near the extreme points of $f(x)$, these might be useful, even better with a convergence control parameter $h$, see Section 4 for their performances.

At the quadratic convergence level, the error by the leading order term in (2.13) penetrated to the next order can be evaluated from (2.9) and (2.10) as

$$C = -\frac{1}{2!}\left(\frac{f(x)}{f'(x)}\right)''\Big|_{f=0} = \frac{1}{2!}\frac{f''(x)}{f'(x)}. \tag{2.16}$$

Therefore, for a cubical rate of convergence level with $p = 3$ in (2.9), $F$ in (2.13) must compensate for the error in (2.16), dictating the form of $A_2(x)$ as

$$A_2(x) = \frac{1}{2!}\frac{f''(x)}{f'(x)}. \tag{2.17}$$

Hence, at the cubical convergence order of approximation, (2.13) must now take the form

$$F(f(x), x) = -\Big(\frac{f(x)}{f'(x)}\Big) - \frac{f''(x)}{2f'(x)}\Big(\frac{f(x)}{f'(x)}\Big)^2 - A_3(x)\Big(\frac{f(x)}{f'(x)}\Big)^3 - \cdots$$
$$-A_n(x)\Big(\frac{f(x)}{f'(x)}\Big)^n, \qquad (2.18)$$

It should be anticipated here that there is no restriction on the coefficients $A_n(x)$ for $n \geq 3$, provided that a cubic rate iterative scheme is aimed at. Again, for the sake of the least computations, of course the coefficients $A_n(x)$; $n \geq 3$ must be dropped from (2.18), thus from (2.5) leading to

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} - \frac{f''(x_n)}{2f'(x_n)}\Big(\frac{f(x_n)}{f'(x_n)}\Big)^2, \qquad (2.19)$$

which is know as the Householder's iteration or Schröder's iteration (see the book by Householder [16]).

We also remind that cubical accuracy of an approximation imposes for all other variants to obey their Taylor series expansions evaluated at $f = 0$ as in equation (2.18), actually

$$F = F(\frac{f(x)}{f'(x)}, \frac{f''(x)}{f'(x)}, x).$$

For instance, anticipating that $f$ is small we may rewrite $F$ in (2.18) in the form

$$F(f(x), x) = -\frac{f(x)}{f'(x)}\Big(\frac{1}{1 - A_2(x)\frac{f(x)}{f'(x)}}\Big) = -\frac{f(x)}{f'(x)}\Big(1 + \frac{f''(x)}{2f'(x)}\frac{f(x)}{f'(x)} + \cdots\Big). \qquad (2.20)$$

Hence, retaining the first two terms in (2.20), we simply get the cubical iteration known as Halley's iteration or König's iteration (see the book by Householder [16])

$$x_{n+1} = x_n - \frac{\frac{f(x_n)}{f'(x_n)}}{1 - \frac{f''(x_n)}{2f'(x_n)}\frac{f(x_n)}{f'(x_n)}} = x_n + 2\frac{\Big(\frac{1}{f(x_n)}\Big)'}{\Big(\frac{1}{f(x_n)}\Big)''}, \qquad (2.21)$$

possessing the error

$$C = \frac{3f''(x)^2 - 2f'(x)f^{(3)}(x)}{12f'(x)^2}, \qquad (2.22)$$

which can also be constructed from a geometrical point of view or from another mathematical treatment of quadratic Newton-Raphson iteration in (2.14) [2].

Amongst the many other variants, we may propose yet a further one-parameter family of cubical iterations from rearranging $F$ in (2.18) as

$$F(f(x), x) = -\frac{f(x_n)}{f'(x_n)}\Big(1 - h + \frac{h}{1 - \frac{A_2(x)}{h}\frac{f(x_n)}{f'(x_n)}}\Big), \qquad (2.23)$$

giving rise to the cubical iterative scheme

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}\Big(1 + \frac{hf(x_n)f''(x_n)}{2hf'(x_n)^2 - f(x_n)f''(x_n)}\Big), \qquad (2.24)$$

with the error

$$C = \frac{(-3 + 6h)f''(x)^2 - 2hf'(x)f^{(3)}(x)}{12hf'(x)^2}, \qquad (2.25)$$

where $h$ is a real parameter. It is noticed that when $h$ is set to unity, our one parameter family (2.24-2.25) evolves into the classical Halley's iteration (2.21-2.22). To get the least

error in absolute value sense, it is better choose $h = 1/2$ in (2.24-2.25) leading to the cubical iteration

$$x_{n+1} = x_n - \frac{1}{2}\frac{f(x_n)}{f'(x_n)}\Big(1 + \frac{1}{1 - \frac{f(x_n)f''(x)}{f'(x_n)^2}}\Big),$$ (2.26)

having the error

$$C = -\frac{f^{(3)}(x)}{6f'(x)}.$$ (2.27)

The accuracy and performance of new proposal in (2.26) and (2.27), which is a super-Halley scheme, are discussed in Section 4. We now generalize the problem of finding coefficients $A_n(x)$ in (2.8) in the following Theorem.

**Theorem 2.1.** *The coefficients $A_n(x)$ in (2.8) for the higher order Newton iteration schemes (Schröder's iterations) can be generated from the algorithm*

$$A_1(x) = 1,$$

$$A_n(x) = -\frac{1}{n!}\Big[\sum_{k=1}^{n-1} A_k(x)\Big(\frac{f(x)}{f'(x)}\Big)^k\Big]^{(n)}\Big|_{f=0}, \quad n \geq 2,$$ (2.28)

*where the superscript $n$ is to mean derivative of order $n$ with respect to $x$ and $A_{n+1}(x)$ is simply the error of the $n$th order convergent iterative scheme.*

**Proof.** $A_1(x)$ is obtained as in (2.12) and $A_2(x)$ is obtained as in (2.17). Following the above methodology, whenever for instance, the quartic convergence is desired with $p = 4$ in (2.9-2.10), then the accumulated error due to (2.20), that is

$$C = -\frac{1}{3!}\frac{-3f''(x)^2 + f'(x)f^{(3)}(x)}{f'(x)^2}$$ (2.29)

must be compensated which yields

$$A_3(x) = C = -\frac{1}{3!}\Big[A_1(x)\Big(\frac{f(x)}{f'(x)}\Big) + A_2(x)\Big(\frac{f(x)}{f'(x)}\Big)^2\Big]^{(3)}\Big|_{f=0}$$ (2.30)

corresponding to $n = 3$ in (2.28). So, the rest of the proof follows. ⋄ □

**Corollary 2.2.** *At the $(n+1)$th convergence rate of an approximation, the shape of $F$ must be in the form*

$$F = F\Big(\frac{f(x)}{f'(x)}, \frac{f''(x)}{f'(x)}, \frac{f'''(x)}{f'(x)}, \cdots, \frac{f^{(n)}(x)}{f'(x)}, x\Big).$$ (2.31)

**Corollary 2.3.** *Two distinct variants of quartic convergent iteration may be derived from (2.28) by considering only the three terms in the fashion*

$$x_{n+1} = x_n - A_1(x_n)\frac{f(x_n)}{f'(x_n)}\Big[\frac{1}{1 - A_2(x_n)\frac{f(x_n)}{f'(x_n)} - \overline{A_3(x_n)}\Big(\frac{f(x_n)}{f'(x_n)}\Big)^2}\Big],$$ (2.32)

*where $\overline{A_3(x)} = \frac{A_3(x)}{2} - \frac{1}{12}\frac{f^{(3)}(x)}{f'(x)}$, having the error*

$$C = \frac{6f''(x)^3 - 6f'(x)f''(x)f^{(3)}(x) + f'(x)^2 f^{(4)}(x)}{24f'(x)^3},$$ (2.33)

*and*

$$x_{n+1} = x_n - A_1(x_n)\frac{f(x_n)}{f'(x_n)}\Big[1 + A_2(x_n)\frac{f(x_n)}{f'(x_n)}\Big(\frac{1}{1 - \frac{A_3(x_n)}{A_2(x_n)}\frac{f(x_n)}{f'(x_n)}}\Big)\Big],$$ (2.34)

*having the error*

$$C = \frac{9f''(x)^4 - 6f'(x)f''(x)^2f^{(3)}(x) - 4f'(x)^2f^{(3)}(x)^2 + 3f'(x)^2f''(x)f^{(4)}(x)}{72f'(x)^3f''(x)}, \quad (2.35)$$

*both competing with the classical Newton iteration of quartic convergence rate (Schröder) obtained from (2.28) with the error*

$$C = \frac{15f''(x)^3 - 10f'(x)f''(x)f^{(3)}(x) + f'(x)^2f^{(4)}(x)}{24f'(x)^3}, \quad (2.36)$$

*and also with the Householder's fourth order iteration (König)*

$$x_{n+1} = x_n + 3\frac{\left(\frac{1}{f(x)}\right)''}{\left(\frac{1}{f(x)}\right)'''}, \quad (2.37)$$

*with the error*

$$C = \frac{3f''(x)^3 - 4f'(x)f''(x)f^{(3)}(x) + f'(x)^2f^{(4)}(x)}{24f'(x)^3}. \quad (2.38)$$

The performance of (2.32) and (2.34) can be found in Section 4.

**Corollary 2.4.** *Yet, another variant of one-parameter family of quartic convergent iteration may be derived from (2.28) or accounting for the form in (2.31) by rearranging the terms in the fashion*

$$x_{n+1} = x_n - \frac{f[x]}{f'[x]}\left(\frac{1 - \frac{f''[x]}{2f'[x]}\frac{f[x]}{f'[x]} + (-1+h)\frac{f^{(3)}[x]}{6f'[x]}\left(\frac{f[x]}{f'[x]}\right)^2}{1 - \frac{f''[x]}{f'[x]}\frac{f[x]}{f'[x]} + h\frac{f^{(3)}[x]}{6f'[x]}\left(\frac{f[x]}{f'[x]}\right)^2}\right), \quad (2.39)$$

*having the error*

$$C = \frac{3f''[x]^3 + 2(-3+h)f'[x]f''[x]f^{(3)}[x] + f'[x]^2f^{(4)}[x]}{24f'[x]^3}. \quad (2.40)$$

*Notice that $h = 1$ yields the König's iteration of fourth-order (2.37-2.38). Taking the advantage of having less error in absolute value sense, however, we must choose $h = 3$ leading to*

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}\left(\frac{1 - \frac{f''(x_n)}{2f'(x_n)}\frac{f(x_n)}{f'(x_n)} + \frac{f^{(3)}(x_n)}{3f'(x_n)}\left(\frac{f(x_n)}{f'(x_n)}\right)^2}{1 - \frac{f''(x_n)}{f'(x_n)}\frac{f(x_n)}{f'(x_n)} + \frac{f^{(3)}(x_n)}{2f'(x_n)}\left(\frac{f(x_n)}{f'(x_n)}\right)^2}\right), \quad (2.41)$$

*having the error*

$$C = \frac{3f''(x)^3 + f'(x)^2f^{(4)}(x)}{24f'(x)^3}. \quad (2.42)$$

Section 4 can be referred for the performance of (2.41).

**Corollary 2.5.** *Similar variants may be produced from (2.28) competing with higher order Schröder's iterations and $(d+2)$th order convergence rate Householder's (Köni's) iterations*

$$x_{n+1} = x_n + (d+1)\frac{\left(\frac{1}{f(x)}\right)^{(d)}}{\left(\frac{1}{f(x)}\right)^{(d+1)}}. \quad (2.43)$$

*At least we may propose the following variant produced from (2.28)*

$$x_{n+1} = x_n - A_1(x_n)\frac{f(x_n)}{f'(x_n)}\left(\frac{1}{1 - \sum_{n=2}^d A_n(x_n)\left(\frac{f(x_n)}{f'(x_n)}\right)^n}\right), \quad (2.44)$$

*where $\overline{A_n(x)}$ is appropriately modified form of $A_n(x)$.*

**Corollary 2.6.** *The iterative scheme proposed in the reference [1] as a result of the Adomian decomposition is simply a subclass of cubically convergent Newton iteration method obtained from (2.28). Indeed, the error of the method in [1] is given by*

$$C = -\frac{f'''(x)}{6f'(x)}, \tag{2.45}$$

*which coincides with the error of the proposed scheme here in (2.27).*

## 3. Multi-step iterations

In this section, by combining the introduced methods in Section 2, using predictor-corrector technique multi-step iterations requiring less derivative computations are derived. The composition of known methods and the approximation of the derivative of the last step, in order to preserve the optimality of the method, is an standard procedure for designing optimal new schemes, refer to the books [3] and [23].

### 3.1. Some two-step iterations

We may provide in the sequel some new two-step iterations possessing cubical order of convergence. For instance,

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

with the pairs

$$(\alpha, \beta) = (-\frac{3+\sqrt{5}}{2}, -\frac{1-\sqrt{5}}{2}) \tag{3.2}$$

and

$$(\alpha, \beta) = (\frac{3-\sqrt{5}}{2}, \frac{1+\sqrt{5}}{2}) \tag{3.3}$$

both having the error of order

$$C = \frac{3f''(x_n) - \alpha^{-1}f'(x_n)f'''(x_n)}{6f'(x_n)^2}. \tag{3.4}$$

Similar designs were also suggested in [10].

Moreover, combining the classical quadratic Newton iteration with the cubical Householder's yields

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} - \frac{f\left(x_n - \frac{f(x_n)}{f'(x_n)}\right)}{f'\left(x_n - \frac{f(x_n)}{f'(x_n)}\right)} - \frac{f\left(x_n - \frac{f(x_n)}{f'(x_n)}\right)^2 f''\left(x_n - \frac{f(x_n)}{f'(x_n)}\right)}{2f'\left(x_n - \frac{f(x_n)}{f'(x_n)}\right)^3}, \tag{3.5}$$

having the sixth order convergence rate with the error

$$C = \frac{3f''(x)^5 - f'(x)f''(x)^3 f^{(3)}(x)}{48f'(x)^5}. \tag{3.6}$$

Additionally, combining the classical quadratic Newton iteration with the cubical Halley's method yields

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} + \frac{2f\left(x_n - \frac{f(x_n)}{f'(x_n)}\right) f'\left(x_n - \frac{f(x_n)}{f'(x_n)}\right)}{-2f'\left(x_n - \frac{f(x_n)}{f'(x_n)}\right)^2 + f\left(x_n - \frac{f(x_n)}{f'(x_n)}\right) f''\left(x_n - \frac{f(x_n)}{f'(x_n)}\right)}, \tag{3.7}$$

**Table 1.** The performance of new algorithms in (2.15) against the classical quadratic Newton algorithm in (2.14) on the test function (4.1). Adjacent to the number of iterations, the corresponding values of $h$ are given in parenthesis.

| Initial guess | NQ | NQ1 | NQ2 | NQ3 |
|---|---|---|---|---|
| $x_0 = 2$ | Divergent | 3 ($h = 0.319$) | 3 ($h = -0.890$) | 20 ($h = 0.922$) |
| $x_0 = 5$ | Divergent | 4 ($h = 0.314$) | 5 ($h = -0.844$) | 21 ($h = 0.785$) |

having the sixth order convergence rate with the error

$$C \;=\; \frac{3f''(x)^5 - 2f'(x)f''(x)^3 f^{(3)}(x)}{96 f'(x)^5}. \tag{3.8}$$

## 3.2. A three-step iteration

Finally, a three-level of classical quadratic Newton method is presented as

$$y_n \;=\; x_n - \frac{f(x_n)}{f'(x_n)},$$

$$z_n \;=\; y_n - \frac{f(y_n)}{f'(y_n)},$$

$$x_{n+1} \;=\; z_n - \frac{f(z_n)}{f'(z_n)}, \tag{3.9}$$

which has eight-order accuracy with the error

$$C \;=\; \frac{f''(x)^7}{128 f'(x)^7}. \tag{3.10}$$

## 4. Illustrations and comparisons

In this section, the new variants as presented in Section 2 are applied to a variety of worked examples chosen from the recent open literature in an aim to measure their efficiency, accuracy and performance against the classical iterative schemes. The dynamic basins of attraction analysis in order to determine the appropriate length of interval of convergence is out of the scope of the present work.

We consider the test functions

$$f(x) \;=\; \tan^{-1} x, \tag{4.1}$$

$$f(x) \;=\; x^3 - 2x + 2, \tag{4.2}$$

$$f(x) \;=\; \pi - 2x \sin\left(\frac{\pi}{x}\right), \tag{4.3}$$

$$f(x) \;=\; x^2 - (1 - x)^5, \tag{4.4}$$

$$f(x) \;=\; x^3 + 4x^2 - 10, \tag{4.5}$$

$$f(x) \;=\; -1 + \frac{x^4}{4} \sin x, \tag{4.6}$$

$$f(x) \;=\; xe^{x^2} - \sin^2 x + 3 \cos x + 5, \tag{4.7}$$

$$f(x) \;=\; x^2 - e^x - 3x + 2, \tag{4.8}$$

which are taken, respectively from the references [15], [29], [24], [1], [8], [4], [15] and [14].

The initial check is presented in Table 1 on the classical quadratic Newton iteration (NQ) given by equation (2.14) against the newly proposed quadratic iterations (NQ1,NQ2,NQ3), respectively given by the expressions in equation (2.15). It is of no surprise to see in Table 1 that the classical Newton iteration algorithm diverges away from the root, since zero is

**Table 2.** The performance of new algorithms in (2.15) against the classical quadratic Newton algorithm in (2.14) on the test function (4.2) possessing the root $x = -1.769292354238631$. Adjacent to the number of iterations, the corresponding values of $h$ are given in parenthesis.

| Initial guess | NQ | NQ1 | NQ2 | NQ3 |
|---|---|---|---|---|
| $x_0 = -2$ | 5 | 3 ($h = 0.7190$) | 3 ($h = -1.269$) | 3 ($h = -1.269$) |
| $x_0 = 0$ | Divergent | 5 ($h = -0.107$) | 5 ($h = 0.2160$) | Divergent |
| $x_0 = 1$ | Divergent | 6 ($h = -0.103$) | 6 ($h = 0.2120$) | 18 ($h = 0.0040$) |

a turning point of $f(x)$. On the other hand, NQ1 and NQ2 successfully resolve the root within a very small number of iterations. NQ3 also catches the root within a reasonable number of iterations, though the algorithm has a singular character at zero. Near the maxima of the function, the failure of the classical Newton iteration by entering into infinite loop (2-cycle) is seen in Table 2. However, we observe from both Tables that the parameter $h$ contributes to the success of the proposed algorithms by means of adjusting the rate of convergence. It is recalled that in the case of failure of the classical Newton iterations near the points of inflexion or extremum, either higher order Newton methods or successive over-relaxation is generally suggested. Besides, the quadratic algorithms in (2.15) constitute powerful alternatives. Another curious example of failure of the classical Newton's quadratic method was presented in [15], see equation (4.3). Taking the initial guess $x_0 = 1/2$, the Newton's method converges to the point zero which is not the actual root of the function in (4.3), but the real root is $x = 1.657400240258006$. Keeping in mind the geometric meaning of the classical Newton's method, figure 1 fairly explains the tendency of failure. On the other hand, with the same initial approximation NQ1 converges to the root within 5 iterations assigning $h = 0.901$, NQ2 converges to the root within 8 iterations assigning $h = 5.005$ and NQ3 converges to the root within 4 iterations assigning $h = 0.862$. All these algorithms converge faster than the recently presented one in [15] based on a vector form which took 18 iterations.
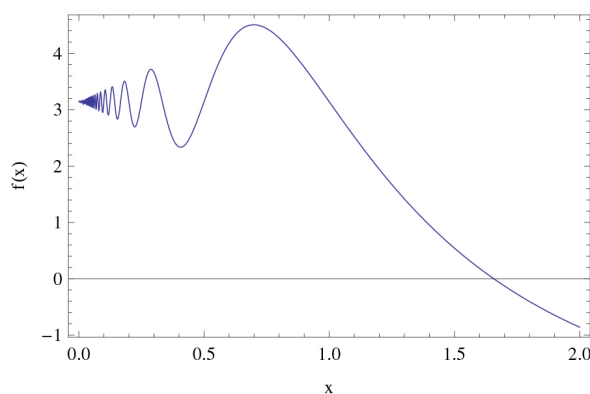


**Figure 1.** Graph of the function in (4.3).

Next, the accuracy and performance of newly proposed single-step cubical iteration (NC) given by equation (2.26) is tested against the classical quadratic Householder's iteration (Schröder) (NS) in (2.19), the classical Halley's or Könnig's iteration (NHK) in (2.21) as well as the algorithm of the reference [1] (AB) on the test function (4.4) in Table 3. The best performance on the test function is observed for our proposed cubic iteration method as compared to the others. Table also justifies our assertion in Corollary 5 that

**Table 3.** The performance and accuracy of the new algorithm in (2.26) against the classical cubical Newton algorithm in (2.19), the classical Halley's iteration in (2.21), the algorithm of the reference [1] on the test function (4.4) possessing the root $x = 0.345954815848242$. Initial approximation is $x_0 = 3$.

| Iter | NS | NHK | AB | NC |
|------|----|-----|----|----|
| 1 | 2.309185040310916 | 2.134723926380368 | 2.116938139481553 | 0.423431620504596 |
| 2 | 1.559422964223167 | 0.721648446504665 | 0.837748863864384 | 0.344873683670437 |
| 3 | 0.508379803859623 | 0.287962091869351 | 0.272745463769937 | 0.345954819218795 |
| 4 | 0.337284989965960 | 0.346136448288485 | 0.346827109618553 | 0.345954815848242 |
| 5 | 0.345954823310806 | 0.345954815839783 | 0.345954814080809 | |
| 6 | 0.345954815848242 | 0.345954815848242 | 0.345954815848242 | |

**Table 4.** The performance of new algorithms in (2.32), (2.34) and (2.41) against the classical quartic Householder's iteration in (2.36) and the classical quartic Könnig's iteration in (2.37) on the test function (4.5) having the root $x = 1.365230013414097$.

| Initial guess | HQ | KQ | PQ1 | PQ2 | PQ3 |
|---------------|----|----|-----|-----|-----|
| $x_0 = 1$ | 3 | 3 | 3 | 3 | 3 |
| $x_0 = 10$ | 5 | 4 | 5 | 4 | 5 |
| $x_0 = 100$ | 8 | 7 | 7 | 7 | 7 |

**Table 5.** The performance and accuracy of the new algorithms in (3.1,3.2) and (3.1,3.3) against the classical cubical Newton algorithm in (2.19) and the classical Halley's iteration in (2.21) on the test function (4.6) for its root $x = 1.418344180662527$. Initial approximation is $x_0 = 2.1$.

| Iter | NS | NHK | MQ1 | MQ2 |
|------|----|-----|-----|-----|
| 1 | 1.586846277446990 | 1.580466213654476 | 1.497483028980226 | 2.011823215688909 |
| 2 | 1.423519529811290 | 1.421244951596745 | 1.419042687095258 | 1.725367747206929 |
| 3 | 1.418344423339663 | 1.418344200579770 | 1.418344181297605 | 1.453308741122478 |
| 4 | 1.418344180662527 | 1.418344180662527 | 1.418344180662527 | 1.418408301922735 |
| 5 | | | | 1.418344180662936 |
| 6 | | | | 1.418344180662527 |

even though NC and AB has the same errors, NC is a better cubically convergent iterative scheme.

It is later demonstrated the history of iterations in Table 4 to tabulate the performance of the newly proposed single-step quartic iterations, namely (PQ1,PQ2,PQ3) in equations (2.32), (2.34) and (2.41) against those of the classical quartic Householder's iteration (Schröder) (HQ) in (2.36) and the classical quartic Könnig's iteration (KQ) in (2.37). The competing performance of our newly proposed quartic convergent iterative methods with the existing ones is witnessed. We should also state that the present algorithms performs better as compared to the quadratic methods of [8].

As for the multi-step methods, the accuracy and performance of cubical iterative methods (3.1,3.2) (MQ1) and those of (3.1,3.3) (MQ2) is given over the classical iterations (NS) and (NHK) in Table 5 on the test function (4.6). Although both the new proposals compete with the well-documented ones, the third order MQ1 seems to perform better for the present problem.

**Table 6.** The performance of new algorithms in (3.5) and (3.7) against the classical sixth-order Schröder's iteration (S6) in (2.28) and Konig's iteration (K6) in (43) on the test function (4.7) for its root $x = -1.207647827130919$.

| Initial guess | S6 | K6 | NH6 | NK6 |
|---|---|---|---|---|
| $x_0 = -5$ | 13 | 7 | 12 | 10 |
| $x_0 = -3$ | 7 | 4 | 6 | 5 |
| $x_0 = -1$ | 2 | 2 | 2 | 2 |
| $x_0 = 1$ | Divergent | 5 | 3 | 3 |
| $x_0 = 2$ | 7 | 16 | Divergent | 5 |

**Table 7.** The performance and accuracy of the new algorithm in (3.7) against the classical eighth-order Schröder's iteration (SE) in (2.28) and Konig's iteration (KE) in (43) on the test function (4.8) possessing the root $x = 0.2575302854398608$. Initial approximation is $x_0 = 5$.

| Iter | SE | KE | NE |
|---|---|---|---|
| 1 | 2.4642661690426630 | 5.4063466209445250 | 2.0050680443844201 |
| 2 | -0.069507913507796 | 14.380905931142920 | 0.2575294164543645 |
| 3 | 0.2575300355441590 | 7.4001957390452030 | 0.2575302854398608 |
| 4 | 0.2575302854398608 | 0.7631615414212887 | |
| 5 | | 0.2575309154917721 | |
| 6 | | 0.2575302854398608 | |

Sixth-order accurate multi-step methods proposed in (3.5) (NH6) and in (3.7) (NK6) are then tested in Table 6 on the test function 4.7 against the classical sixth-order Schröder's iteration (S6) in (2.28) and Konig's iteration (K6) in (43). In terms of converged iteration numbers, again good performance of the newly offered two-step techniques are anticipated from the Table.

Finally, the performance of multi-step eighth-order accurate method proposed in (3.9) (NE) is investigated versus the performance of classical eighth-order Schröder's iteration (SE) in (2.28) and Konig's iteration (KE) in (43) on the test function (4.8) in Table 7, which reveals the good performance of the newly proposed three-step method.

## 5. Concluding remarks

The classical Newton iteration formulas of any order for finding the root of a nonlinear equation are revisited in an aim to design and construct them from an easily pursuable procedure. A simple methodology of derivation of the iterative schemes is hence pursued on the ground of error elimination. Making use of a general fixed-point representation of the equation and double expansion of the representation around the root and the nonlinear function whose root is searched for, a simple algorithm is presented based on the error tracking.

Starting from the conventional linearly convergent fixed point iteration method and quadratically convergent Newton-Raphson method, the algorithm is shown to produce the well-formulated high order Newton iteration formulae as well as their well-documented variants like the Schröder's and the König's iterations all known as Householder's iterations. Taking into account of the algorithm, some rare classes of new variants of any convergence rate are shown to be derivable with alternative errors so that they noticeably compete with the known formulas with a detectable gain. The well-accepted deficiencies of the classical Newton-Raphson method such as the divergence near a point of inflection,

the oscillatory behavior near the local extremum and the divergence near-zero slope are exhibited to be defeated by the newly proposed beneficial quadratic alternative iterative schemes.

Finally, some new higher order accurate competitive variants by forming the composition of the derived variants are proposed giving rise to novel multi-step approximations which demand less computation of higher derivatives. The utility and performance of the proposed methods and their potential advantages over the classical ones of the same order are discussed on some well-documented examples from the open literature.

# References

[1] S. Abbasbandy, *Improving Newton-Raphson method for nonlinear equations modified Adomian decomposition method*, Appl. Math. Comput. **145**, 887–893, 2003.

[2] G. Alefeld, *On the convergence of Halley's method*, Amer. Math. Monthly, **8**, 530–536, 1981.

[3] S. Amat and S. Busquier, *Advances in iterative methods for nonlinear equations*, Springer, 2016.

[4] G. Ardelean, *A new third-order Newton-type iterative method for solving nonlinear equations*, Appl. Math. Comput. **219**, 9856–9864, 2013.

[5] J. Chen, *A new family of exponential iteration methods with quadratic convergence of both diameters and points for enclosing zeros of nonlinear equations*, Math. Comput. Model. **45**, 544–552, 2007.

[6] C. Chun, *A geometric construction of iterative formulas of order three*, Appl. Math. Lett. **23**, 512–516, 2010.

[7] C. Chun and Y. Ham, *A one-parameter fourth-order family of iterative methods for nonlinear equations*, Appl. Math. Comput. **189**, 610–614, 2007.

[8] C. Chun and Y. Ham, *Some fourth-order modifications of Newtons method*, Appl. Math. Comput. **197**, 654–658, 2008.

[9] A. Cordero and J.R. Torregrosa, *Low-complexity root-finding iteration functions with no derivatives of any order of convergence*, J. Comput. Appl. Math. **275**, 502–515, 2015.

[10] A. Cordero, J.L. Hueso, E. Martinez and J.R. Torregrosa, *Efficient high-order methods based on golden ratio for nonlinear systems*, Appl. Math. Comput. **217**, 4548–4556, 2011.

[11] M. Dehghan and M. Hajarian, *Some derivative free quadratic and cubic convergence iterative formulaa for solving nonlinear equations*, Comp. & App. Math. **29**, 19–30, 2010.

[12] T. Fang, F. Guo and C.F.F. Lee, *A new iteration method with cubic convergence to solve nonlinear algebraic equations*, Appl. Math. Comput. **175**, 1147–1155, 2006.

[13] W.J. Gilbert, *Generalizations of Newton's method*, Fractals, **9**, 251–262, 2001.

[14] Y.M. Hama, C. Chunb and S.G. Leeb, *Some higher-order modifications of Newton's method for solving nonlinear equations*, J. Comput. Appl. Math. **222**, 477–486, 2008.

[15] D. Herceg and D. Herceg, *Sixth-order modifications of Newton's method based on Stolarsky and Gini means*, J. Comput. Appl. Math. **267**, 244–253, 2014.

[16] A.S. Householder, *Principles of numerical analysis*, Dover, New-York, 1974.

[17] P. Jarratt, *Some efficient fourth order multipoint methods for solving equations*, BIT, **9**, 119–124, 1969.

[18] R. Kalaba and A. Tishler, *A generalized Newton algorithm using higher-order derivatives*, J. Optim. Theory Appl. **39**, 1–16, 1983.

[19] J. Kou, *The improvements of modified Newton's method*, Appl. Math. Comput. **189**, 602–609, 2007.

[20] G. Labelle, *On extensions of the Newton-Raphson iterative scheme to arbitrary orders*, Discrete Math. Theoret. Comput. Sci. proc. AN, 713–724, 2010.

[21] T.J. McDougalla and S.J. Wotherspoon, *A simple modification of Newton's method to achieve convergence of order* $1 + \sqrt{2}$, Appl. Math. Lett. **29**, 20–25, 2014.

[22] M. Pakdemirli and H. Boyaci, *Generation of root finding algorithms via perturbation theory and some formulas*, Appl. Math. Comput. **184**, 783–788, 2007.

[23] M.S. Petković, B. Neta, L.D. Petković, and J. Džunić, *Multipoint methods for solving nonlinear equations*, Elsevier, 2013.

[24] H. Ramosa and J.V. Aguiar, *The application of Newton's method in vector form for solving nonlinear scalar equations where the classical Newton method fails*, J. Comput. Appl. Math. **275**, 228–237, 2015.

[25] F.A. Shah and M.A. Noor, *Some numerical methods for solving nonlinear equations by using decomposition technique*, Appl. Math. Comput. **251**, 378–386, 2015.

[26] M. Sharifi, D.K.R. Babajee and F. Soleymani, *Finding the solution of nonlinear equations by a class of optimal methods*, Computers and Mathematics with Applications, **63**, 764–774, 2012.

[27] M.K. Singh, *A six-order variant of Newtons method for solving nonlinear equations*, Computational Methods in Science and Technology, **15**, 185–193, 2009.

[28] H. Susanto and N. Karjanto, *Newtons methods basins of attraction revisited*, Appl. Math. Comput. **215**, 1084–1090, 2009.

[29] J.F. Traub, *Iterative methods for the solution of equations*, Chelsea Publishing Company, New York, 1997.

[30] P. Wang, *A third-order family of Newton-like iteration methods for solving nonlinear equations*, J. Numer. Math. Stoch. **3**, 13–19, 2011.