European Journal of Business & Social Sciences
Available at https://ejbss.org/
ISSN: 2235-767X
Volume 07 Issue 01
January 2019

# A New Variants of Newton's Method with Second-Order Derivatives

**Paul Rai [1], Dr. Sonal Bharti**

[1]Research Scholar, Department of Mathematics, Sri Satya Sai University of Technology & Medical Sciences, Sehore, MP(India)

## ABSTRACT

The study is mainly focus to develop new iterative methods for finding simple roots of scalar nonlinear equations, unconstrained optimization problems and systems of nonlinear equations. The methods are found to yield better performance than Newton's method and overcome its limitations.

**Keywords**: Newton method, Non-linear equations, Iterative method, Order of convergence,

## 1.1 INTRODUCTION

This research presents various new methods of second-order for solving scalar nonlinear equations, unconstrained optimization problems and systems of nonlinear equations. The numerical examples taken in this chapter show that, in many cases, proposed methods are effective alternative to Newton's method which may converge slowly or even fail. These are the simple expansions of Newton's method and have well-known geometric derivations. These methods remove severe conditions $f'(x_n) \neq 0$ or $f''(x_n) \neq 0$ of Newton's method for the case of nonlinear equations or for the case of unconstrained optimization problems respectively.

## 1.2 ITERATIVE METHODS FOR SOLVING SYSTEMS OF NON-LINEAR EQUATIONS

One of the most significant and challenging problems in computational mathematics is to compute efficiently the approximate roots of nonlinear equations (1.1). Generally, iterative methods are used to solve such equations. All these methods need prior knowledge of one or more initial guesses for the required root. Once an initial interval is known to contain a root, several classical procedures such as Regula-Falsi method, Bisection method, Steffensen's method, Secant method, (as shown in and Newton's method etc. are available to refine it further. Bisection and Regula-Falsi methods are globally convergent and have linear rate of convergence, on the other hand, Secant method is super linearly convergent, whereas Steffensen's and Newton's methods are quadratically convergent.

Out of these methods, Newton's method is one of the most elementary tools in computational mathematics. Newton's method is probably the simplest, flexible and most frequently used algorithm in numerical analysis, and is given by

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}. \qquad (1.1)$$

Although, Newton's method is often very efficient, but still there are many situations where it performs poorly. There is no general convergence criterion for Newton's method.

### 1.2.1 GEOMETRICALLY CONSTRUCTED FAMILIES OF NEWTON'S METHOD FOR NONLINEAR EQUATION

The aim of this work is to eliminate the defects of Newton's method by simple modifications of iteration processes. Numerical results indicate that proposed iterative methods are effective and comparable to the well-known Newton's method. Furthermore, the presented techniques have guaranteed convergence unlike Newton's method and are as simple as this known technique.

European Journal of Business & Social Sciences
Available at https://ejbss.org/
ISSN: 2235-767X
Volume 07 Issue 01
January 2019

These methods are derived by implementing approximations through a straight line, parabolic and elliptic curve in the vicinity of required root.

Let $$y = f(x), \tag{1.2}$$

represents the graph of the function $f(x)$ in equation (1.1).

**(i) Straight line approximation**

Consider an equation of a straight line having slope equal to $\alpha f(x_0)$ and passing through a point $(x_0, 0)$ in the form

$$y = \alpha f(x_0)(x - x_0), \tag{1.3}$$

where '$x_0$' is an initial guess to the required root '$r$' of the equation (1.1) and $\alpha \in \mathbb{R}$.

Let

$$x_1 = x_0 + h, |h| << 1, \tag{1.4}$$

be a better approximation to the required root. Assume that the point of intersection of straight line (2.3) with the graph (2.2) is at a point $(x_1, f(x_0+h))$. Now the straight line (2.3) while passing through the point $(x_1, f(x_0+h))$ takes the form

$$f(x_0 + h) = h\alpha f(x_0), \tag{1.5}$$

Expanding left-hand side by Taylor's series expansion about a point $x = x_0$ and retaining its terms up to $(O(h))$, one can have

$$f(x_0) + hf'(x_0) + O(h^2) = h\alpha f(x_0). \tag{1.6}$$

By solving the equation 1.6,one gets the first approximation to the required root as

$$x_1 = x_0 - \frac{f(x_0)}{f'(x_0) - \alpha f(x_0)}. \tag{1.7}$$

Now repeating this process until the straight line (2.3) becomes $x - axis$, a general formula for successive approximations is given by

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n) - \alpha f(x_n)}. \tag{1.8}$$

This is a new one-parameter family of Newton's method. In order to obtain quadratic convergence of the method, the sign of entity 'α' in denominator should be chosen so that the denominator would be the largest in magnitude. This formula is **well defined even if** $f'(x_n)$ is zero unlike Newton's formula.

Further, it can be seen that this family of Newton's method gives very good approximation to the required root when $|\alpha|$ is small. This is because that, for small values of 'α', slope or angle of inclination of straight line (2.3) with $x - axis$ becomes smaller, i.e. as $\alpha \rightarrow 0$, the straight line tends to $x - axis$. This means that our next approximation will move faster towards the desired root. However, for large values of 'α', formula (1.8) still works but takes more iterations as compared to the small values of 'α'.

One can also derive the same formula (1.10) by considering an exponentially fitted osculating straight line [2.6] in a following form:

$$y = e^{-\alpha(x-x_0)}[a(x - x_0) + b], \tag{1.9}$$

where '$x_0$' is an initial guess to the required root '$r$' of an equation (1.1), '$a$' and '$b$' are arbitrary constants to be determined by imposing the tangency conditions on $y$ at $x = x_0$:

$$y(x_0) = f(x_0) \text{ and } y'(x_0) = f'(x_0). \quad (1.10)$$

Next, Theorem 1.2.1 presents a mathematical proof for an order of convergence of the proposed iterative formula (2.10).

**Theorem 1.2.1.** *Let $f$: $I \subseteq \mathbb{R} \to \mathbb{R}$ has at least first two continuous derivatives defined on an open interval I, enclosing a simple zero of $f(x)$ (say $x = r \in I$). Assume that initial guess $x = x_0$ is sufficiently close to '$r$', $f'(r) \neq 0$ and $f'(x_n) - \alpha f(x_n) \neq 0$ in I. Then an iteration scheme defined by formula (2.10) is quadratically convergent and satisfies the following erroneous equation :*

$$e_{n+1} = (C_2 - \alpha)e_n^2 + O(e_n^3), \quad (1.11)$$

*where $e_n = x_n - r$ is an error at $n^{th}$ iteration and $C_k = \left(\frac{1}{k!}\right)\frac{f^k(r)}{f'(r)}, k = 2, 3, \ldots$*

**(ii) Parabolic approximation**

Assume that the equation (1.1) has a simple root '$r$' which is to be found and let '$x_0$' be an initial guess to this root. On the same graph of $f(x)$ given by equation (1.2), we sketch a parabola

$$y = \alpha^2 f(x_0)(x - x_0)^2, \quad (1.12)$$

where $\alpha \in \mathbb{R}$, is a scaling parameter. The parabola (3.18) widens as $|\alpha| \to 0$ and narrows down as $|\alpha|$ becomes large.

Let

$$x_1 = x_0 + h, |h| << 1, \quad (1.13)$$

be a better approximation to the required root. Assume that one of the points of intersections of this graph with parabola (3.18) is at a point ($x_1$, $f(x_0+h)$). Now the parabola (3.18) while passing through point ($x_1$, $f(x_0+h)$) takes a form of

$$f(x_0 + h) = \alpha^2 f(x_0)h^2. \quad (1.14)$$

Expanding left-hand side by Taylor's series expansion about a point $x = x_0$ and ignoring its terms with second and higher-order derivatives, one can have

$$f(x_0) + hf'(x_0) + O(h^2) = \alpha^2 f(x_0)h^2. \quad (1.15)$$

By solving the equation 1.21 ,one gets first approximation to the required root as

$$x_1 = x_0 - \frac{2f(x_0)}{f'(x_0) \pm \sqrt{\{f'(x_0)\}^2 + 4\alpha^2\{f(x_0)\}^2}}. \quad (1.16)$$

Now repeating this process until parabola (3.18) becomes $x - axis$, a general formula for successive approximations is given by

$$x_{n+1} = x_0 - \frac{2f(x_n)}{f'(x_n) \pm \sqrt{\{f'(x_n)\}^2 + 4\alpha^2\{f(x_n)\}^2}}, \quad (1.17)$$

in which the sign should be chosen so as to make the denominator largest in magnitude. This is a new parabolic version of Newton's method [1]. The advantage of this method is that, it **converges**

European Journal of Business & Social Sciences
Available at https://ejbss.org/
ISSN: 2235-767X
Volume 07 Issue 01
January 2019

**quadratically and has a same erroneous equation** as Newton's method. Moreover, this method **does not fail** even if $f'(x_n) = 0$ in the vicinity of the required root unlike Newton's method.

Further, it can be observed that a family of parabolic methods (1.17) gives a very good approximation to the required root when |α| (scaling parameter) is small. This is because that for small values of 'α', parabola widens along with horizontal direction. This means that our next approximation will move faster towards the desired root. For large values of 'α', the formula (1.17) still works but takes a more number of iterations as compared to the small values of 'α'.

A mathematical proof for an order of convergence of the proposed iterative formula (1.17) has been presented in the following theorem:

**Theorem 1.2.2.** *Let f: $I \subseteq \mathbb{R} \to \mathbb{R}$ has at least first two continuous derivatives defined on an open interval I, enclosing a simple zero of f (x) (say x = r ∈ I). Assume that initial guess x = $x_0$ is sufficiently close to 'r' and $f'(r) \neq 0$ in I. Then an iteration scheme defined by formula (1.18) is quadratically convergent and satisfies the following erroneous equation :*

$$e_{n+1} = C_2 e_n^2 + O(e_n^3). \tag{1.18}$$

**(iii) Ellipse Approximation**

Let '$x_0$' be an initial guess to a simple root '*r*' of an equation (1.1). For some $\alpha' \in \mathbb{R} - \{0\}$, let us sketch an ellipse

$$\frac{(x - x_0)^2}{\alpha'^2} + \frac{\{y - f(x_0)\}^2}{\{f(x_0)\}^2} = 1, \tag{1.19}$$

on graph (2.2) of *f* (*x*).

Let

$$x_1 = x_0 + h, |h| << 1, \tag{1.20}$$

be a better approximation to the required root. Assume that one of the points of intersections of this graph with ellipse (3.30) is at a point ($x_1$, *f* ($x_0$ +*h*)). Now an ellipse (3.30) while passing through point ($x_1$, *f* ($x_0$+*h*)) takes a form

$$\alpha^2 h^2 \{f(x_0)\}^2 + \{f(x_0 + h)\}^2 - 2f(x_0 + h)f(x_0) = 0, \tag{1.21}$$

where $\alpha' = \frac{1}{\alpha}$.

Expanding left-hand side by Taylor's series expansion about a point *x* = $x_0$ and neglecting second and higher-order derivatives, one can have

$$\alpha^2 h^2 \{f(x_0)\}^2 + \{f(x_0) + hf'(x_0) + O(h^2)\}^2 - 2\{f(x_0) + hf'(x_0) + O(h^2)\} f(x_0) = 0 \tag{1.22}$$

By solving the equation 1.22, one can get the first approximation to the required root as

$$x_1 = x_0 \pm \frac{f(x_0)}{\sqrt{\{f'(x_0)\}^2 + \alpha^2 \{f(x_0)\}^2}}, \tag{1.23}$$

where the positive sign is taken if $x_0$ < *r* and the negative sign is taken if $x_0$ > *r*. Geometrically, if slope of curve $\{f'(x_0)\}$ at a point $\{x_0, f(x_0)\}$ is negative, then we take positive sign otherwise negative.

European Journal of Business & Social Sciences
Available at https://ejbss.org/
ISSN: 2235-767X
Volume 07 Issue 01
January 2019

Now repeating this process until an ellipse becomes "point ellipse" on $x - axis$, a general formula of successive approximations is given by

$$x_{n+1} = x_n \pm \frac{f(x_n)}{\sqrt{\{f'(x_n)\}^2 + \alpha^2\{f(x_n)\}^2}}. \tag{1.24}$$

This is an elliptic version of Newton's method [204]. Again the advantage of this method is that, it is **quadratically convergent and has a same erroneous equation** as Newton's method. Moreover, this method **does not fail** even if $f'(x_n) = 0$ or in the vicinity of the required root unlike Newton's method.

Further, it is found that a family of ellipse methods (1.24) gives a very good approximation to the required root when 'α' is small. This is because for small values of 'α', the ellipse shrinks in vertical direction and extends along with horizontal direction. This means that our next approximation will move faster towards the desired root. For large values of 'α', formula (1.24) still works but needs more number of iterations as compared to the small values of 'α'.

**Special case of ellipse method: circle method**

At $\alpha = \frac{1}{f(x_n)}$, an equation (1.25) yields

$$x_{n+1} = x_n \pm \frac{f(x_n)}{\sqrt{1 + \{f'(x_n)\}^2}}. \tag{1.25}$$

This formula can be named as the circle method, as an ellipse given by equation (1.25), now becomes a circle having center at $\{x_0, f(x_0)\}$ and radius $f(x_0)$. Therefore, the circle method is a special case of ellipse method (1.25). Apparently, the formula (1.25) has a scaling problem. As a result of this, an efficiency index of the circle method tangibly decreases as compared to Newton's method [3, 4-9], parabolic method (1.23) and ellipse method (1.30) respectively. However, the circle method has guaranteed convergence unlike Newton's method.

The following theorem presents a proof for an order of convergence of the proposed iterative formula (1.30):

**Theorem 1.2.3.** *Let $f$: $I \subseteq \mathbb{R} \to \mathbb{R}$ has at least first two continuous derivatives defined on an open interval I, enclosing a simple zero of $f(x)$ (say $x = r \in I$). Assume that initial guess $x = x_0$ is sufficiently close to 'r' and $f'(r) \neq 0$ in I. Then an iteration scheme defined by formula (3.30) is quadratically convergent and satisfies the following erroneous equation :*

$$e_{n+1} = C_2 e_n^2 + O(e_n^3). \tag{1.26}$$

**(iv) Exponential iterative methods**

The proposed methods can also be extended further to exponentially quadratically convergent iterative formulae for solving nonlinear equations numerically. Let us consider $x_{n+1} = x_n \exp\left(-\frac{h}{x_n}\right)$ as a better approximation to the required root 'r', then from equations (1.8), (1.23) and (1.30), one can derive the following exponential iterative formulae :

$$x_{n+1} = x_n \exp\left(\frac{f(x_n)}{f'(x_n) - \alpha f(x_n)}\right), \tag{1.27}$$

$$x_{n+1} = x_n \exp\left(\frac{2f(x_n)}{f'(x_n) \pm \sqrt{\{f'(x_n)\}^2 + 4\alpha^2\{f(x_n)\}^2}}\right), \tag{1.28}$$

and

$$x_{n+1} = x_n \exp\left(\pm\frac{f(x_n)}{\sqrt{\{f'(x_n)\}^2 + \alpha^2\{f(x_n)\}^2}}\right), \tag{1.29}$$

respectively.

Letting $\alpha \to 0$ in (1.27), (1.28) and (1.29), one can obtain another exponential iterative method given by

$$x_{n+1} = x_n \exp\left(-\frac{f(x_n)}{x_n f'(x_n)}\right). \tag{1.30}$$

Note that by taking the first-order Taylor's series expansion of exp $\left(-\frac{f(x_n)}{x_n f'(x_n)}\right)$ in equation (1.30), Newton's formula can be achieved. Chen and Li [10] have also derived some new classes of exponentially quadratically convergent iterative methods by using different approach based on the main idea of Mamta et al. [1].

1.2.2 TESTED EXAMPLES

In this section, an attempt has been made to check the effectiveness of newly developed methods. Here, let us consider some examples to compare the number of iterations, total number of function evaluations (**TNOFE**) and computation order of convergence (**COC**) needed in the conventional Newton's method and its modifications namely, modified Newton's method (1.18) (***MNM***), parabolic method (1.23) (***PM***) and ellipse method (1.30) (***EM***) respectively, for solving nonlinear equations. Here, for simplicity, the formulae are tested for $|\alpha| = 1$. The results are summarized in Table 1.1 and Table 1.2 respectively. Computations have been performed by using ***MATLAB***® version 7.5 (***R***2007*b*) in double precision arithmetic. We use $\varepsilon = 10^{-15}$ as a tolerance error. The following stopping criteria are used for computer programmes :

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

**Example 1.2.1.** $\sin(x) = 0.$

This equation has infinite number of roots. It can be seen that Newton's method does not necessarily converge to a root that is nearest to the starting value. The proposed methods **do not exhibit** this type of behaviour.

**Example 1.2.2.** $e^{-x} - \sin(x) = 0.$

Again this equation has infinite number of roots lying close to π, 2π, 3π........... Newton's method [11] with starting value $x_0$ =5 converges to the root closest to 3π whereas proposed methods converge to **the closest root** 6.285049273382587.

**Example 1.2.3.** $xe^{-x} = 0.$

The derivative of function $xe^{-x}$ is zero at $x = 1$ and negative for $x > 1$. For any initial guess $x_0 < 0$, Newton's method converges to the root very efficiently. For example, $x_0 = 2$ then $x_1 = 4$, $x_2 = 5.3333$ and so on. On the other hand, proposed methods **can give the required root** if sign in the proposed methods is chosen suitably.

**Example 1.2.4.** $4x^4 - 4x^2 = 0.$

In applying, Newton's method to solve this equation, problems arise if starting points give horizontal tangents or tangents cycle back and forth from one to another. **The proposed methods** (1.23) **(*PM*)** and (1.30) **(*EM*) do not exhibit this behaviour.**

European Journal of Business & Social Sciences
Available at https://ejbss.org/
ISSN: 2235-767X
Volume 07 Issue 01
January 2019

## TABLE 1.1

**Test**

| Example No. | I | Initial guesses | Exact root ($r$) | NM | MNM (2.10) | PM (2.25) | EM (2.37) |
|---|---|---|---|---|---|---|---|
| | | 1.5 | 0.000000000000000 | $-4\pi$* | 7 | 5 | 4 |
| 1.2.1 | [−2, 2] | 1.51 | 0.000000000000000 | $-5\pi$* | 7 | 5 | 4 |
| | | 1.52 | 0.000000000000000 | $-6\pi$* | 7 | 5 | 4 |
| | | 1.53 | 0.000000000000000 | $-7\pi$* | 7 | 5 | 4 |
| 1.2.2 | [5, 7] | 5.0 | 6.285049273382587 | 9.42467254738522* | 7 | 5 | 4 |
| | | 6.0 | 6.285049273382587 | 3 | 5 | 4 | 3 |
| 1.2.3 | [0, 5] | 2.0 | 0.000000000000000 | Divergent | 1 | 6 | 6 |
| | | $\frac{\sqrt{2}}{2}$ | 1.000000000000000 | −1.000000000000000* | 7 | 9 | 8 |
| 1.2.4 | | $-\frac{\sqrt{2}}{2}$ | −1.000000000000000 | 1.000000000000000* | 7 | 9 | 8 |
| | [−.8,.8] | $\frac{\sqrt{21}}{7}$ | 1.000000000000000 | 0.000000000000000* | 0.000000000000000* | 7 | 8 |

**examples, intervals** ($I$), **their initial guesses** ($x_0$), **exact root** ($r$) **and number of iterations**

*Converges to undesired root

**TABLE 1.2 : Computational order of convergence and Total number of function evaluations**

| (COC) | | | | | (TONFE) | | | |
|---|---|---|---|---|---|---|---|---|
| Example No. | NM | MNM (2.10) | PM (2.25) | EM (2.37) | NM | MNM (2.10) | PM (2.25) | EM (2.37) |
| 1.2.1 | — | 2.00 | 3.00 | 3.00 | ⌒ | 14 | 10 | 8 |
| | — | 2.00 | 3.00 | 3.00 | ⌒ | 14 | 10 | 8 |
| | — | 2.00 | 3.00 | 3.00 | ⌒ | 14 | 10 | 8 |
| | — | 2.00 | 3.00 | 3.00 | ⌒ | 14 | 10 | 8 |
| 1.2.2 | — | 2.00 | 2.40 | 2.74 | ⌒ | 14 | 10 | 8 |
| | 2.85 | 2.00 | 2.20 | 2.70 | 6 | 10 | 8 | 6 |
| 1.2.3 | ⋯ | ≡≡ | 2.00 | 2.00 | ∼ | 2 | 12 | 12 |
| | — | 2.00 | 1.99 | 2.00 | ⌒ | 14 | 18 | 16 |
| 1.2.4 | — | 2.00 | 1.99 | 2.00 | ⌒ | 14 | 18 | 16 |
| | — | — | 2.00 | 2.00 | ⌒ | ⌒ | 14 | 16 |
| | — | — | 2.00 | 2.00 | ⌒ | ⌒ | 14 | 16 |

Here, the symbols namely

(i) '—' and ' ⌒ ' represent **COC** and **TNOFE** for a case of undesired root respectively.

(ii) '⋯' and '∼' represent **COC** and **TNOFE** for a case of divergence respectively.

(iii) '==' and '=' represent **COC** and **TNOFE** for a case of for a case of failure respectively.

(iv) '≡≡' represents **COC** when the number of iterations are not sufficient.

1.3 ITERATIVE METHODS FOR UNCONSTAINED OPTIMIZATION PROBLEMS

Optimization problems with or without constraints arise in various fields such as science, engineering, economics especially in management sciences, engineering design, operation research, computer science, financial management etc. where numerical information is processed. In recent years, many problems in business situations and engineering designs have been modelled as an optimization problem for taking optimal decisions. In fact, numerical optimization techniques play a significant role in almost all branches of engineering and mathematics.

Several classical methods such as Golden search method, Fibonacci search method, Quadratic fitting method, Cubic interpolation method and Newton's method etc. are available in literature for solving unconstrained minimization problems (1.2). For detailed survey of these most important methods, many excellent text books are also available in literature. All unconstrained minimization methods are iterative in nature and hence they start from an initial trial solution and proceed towards the minimum point in a sequential manner.

Again, Newton's method (3.39) plays a vital role in operation research, optimization and control theory. It has many applications in management science, industrial and financial research, chaos and fractals, dynamical systems, variational inequalities and equilibrium-type problems, stability analysis, data mining and even to random operator equations. Its role in optimization theory cannot be underestimated as this method is a basis for the most effective procedures in linear and nonlinear programming. For a more detailed survey, one can refer to paper by Ployak and the references cited therein. In short, for solving nonlinear, univariate optimization problems, Newton's method is an important and basic method which converges quadratically. The idea behind Newton's method is to approximate an objective function locally by a quadratic function which at $x = x_n$ agrees with function $f(x)$ up to second-order derivative. The process can be repeated at a point that optimizes an approximate function. Again the condition $f''(x_n) \neq 0$ in the neighbourhood of the root is required for the success of the Newton's method. Many new modified Newton type methods for solving unconstrained optimization problems (1.2) have been suggested in the literature.

The purpose of this study is to develop iterative methods to solve nonlinear, univariate unconstrained optimization problems to find an extremum of the given function $f(x)$ under no constraints and is to eliminate the defects of Newton's method by simple modifications of iteration processes.

1.3.1 EXPANSION OF PROPOSED ITERATIVE METHODS

An attempt has been made to expand the proposed iterative formulae namely, modified Newton's method (1.8) (***MNM***), parabolic method (1.23) (***PM***) and ellipse method (1.30) (***EM***) respectively to unconstrained optimization problems.

**(i) Expansion of the method (1.8)**

Assume that $f(x)$ is sufficiently smooth function and has an extremum (maxima or minima) at a point $x = \beta$. From (3.5), consider an auxiliary function with parameter 'α' as

$$q(x) = f(x) - \alpha h f(x_n). \qquad (1.31)$$

It is possible to construct a quadratic function $q(x)$ from (2.47) which agrees with $f(x)$ up to second-order derivative in the neighbourhood of a point $x = x_n$, i.e.

$$q(x) = f(x_n) + (x - x_n)f'(x_n) + \frac{1}{2}(x - x_n)^2 f''^{(x_n)} - \alpha(x - x_n)f(x_n). \qquad (1.32)$$

One may calculate an estimate of $f(x)$ at $x = x_{n+1}$ by finding a point where derivative of $q(x)$ vanishes [12, 13], i.e. $q'(x_{n+1}) = 0$. This gives

$$f'(x_n) + (x_{n+1} - x_n)f''(x_n) - \alpha(x_{n+1} - x_n)f'(x_n) = 0. \tag{1.33}$$

Solving this equation for $x_{n+1} - x_n$, one gets

$$x_{n+1} = x_n - \frac{f'(x_n)}{f''(x_n) - \alpha f'(x_n)}. \tag{1.34}$$

This is a one parameter family of Newton's method for solving unconstrained optimization problems. To obtain a quadratic convergence of this method, the sign of entity 'α' in denominator should be chosen so that the denominator would be the largest in magnitude. This method is **well defined even if** $f''(x_n) = 0$ or very small in the vicinity of the required optimum point unlike Newton's method.

Following is a mathematical proof for an order of convergence of the proposed iterative method (1.34) :

**Theorem 1.3.1.** *Let* $f: I \subseteq \mathbb{R} \to \mathbb{R}$ *be a sufficiently differentiable function defined on an open interval I and* $x = \beta \in I$ *be an optimum point of* $f(x)$*. Assume that initial guess* $x = x_0$ *is sufficiently close to* 'β' *and* $f''(x_n) - \alpha f'(x_n) \neq 0$ in *I. Then an iteration scheme defined by formula* (1.34) *is quadratically convergent and satisfies the following erroneous equation :*

$$e_{n+1} = (3A_3 - \alpha)e_n^2 + O(e_n^3), \tag{1.35}$$

where $e_n = x_n - \beta$ *is an error at* $n^{th}$ *iteration and* $A_k = \left(\frac{1}{k!}\right)\frac{f^k(\beta)}{f''(\beta)}$, $k = 3,4,.........$

**(i) Expansion of the method (1.23)**

Assume that $f(x)$ is sufficiently smooth function and has an extremum (maxima or minima) at a point $x = \beta$. From (2.20), consider an auxiliary function with parameter 'α' as

$$q(x) = f(x) - \alpha^2 h^2 f(x_n). \tag{1.36}$$

Again, it is possible to construct a quadratic function $q(x)$ from (2.56) which agrees with $f(x)$ up to second-order derivative in the neighbourhood of a point $x = x_n$, i.e.

$$q(x) = f(x_n) + (x - x_n)f'(x_n) + \frac{1}{2}(x - x_n)^2 f''(x_n) - \alpha^2(x - x_n)^2 f(x_n). \tag{1.37}$$

One may calculate an estimate of $f(x)$ at $x = x_{n+1}$ by finding a point where derivative of $q(x)$ vanishes [12, 13], i.e. $q'(x_{n+1}) = 0$. This leads to

$$\alpha^2(x_{n+1} - x_n)^2 f'(x_n) - (x_{n+1} - x_n)f''(x_n) - f'(x_n) = 0. \tag{1.38}$$

Solving this quadratic equation for $x_{n+1} - x_n$, one gets

$$x_{n+1} = x_n - \frac{2f'(x_n)}{f''(x_n) \pm \sqrt{\{f''(x_n)\}^2 + 4\alpha^2\{f'(x_n)\}^2}}. \tag{1.39}$$

This is a modification over the method (1.23) for solving unconstrained optimization problems. In formula (1.38), a sign in denominator should be chosen so that the denominator would be the largest in magnitude. The advantage of this method is that, it converges quadratically and has a same erroneous equation as that of Newton's method. Moreover, this method **does not fail even if** $f''(x_n) = 0$ (or very small in the vicinity of the required optimum point unlike Newton's method for unconstrained optimization problems.

A mathematical proof for order of convergence of proposed iterative method (1.39) is presented below.

**Theorem 1.3.2.** *Let* $f: I \subseteq \mathbb{R} \to \mathbb{R}$ *be a sufficiently differentiable function defined on an open interval I and* $x = \beta \in I$ *be an optimum point of* $f(x)$. *Assume that initial guess* $x = x_0$ *is sufficiently close to* 'β' in *I*. *Then an iteration scheme defined by formula* (3.59) *is quadratically convergent and satisfies the following erroneous equation :*

$$e_{n+1} = 3A_3 e_n^2 + O(e_n^3). \tag{1.40}$$

**(ii) Expansion of the method (1.30)**

Consider the function

$$G(x) = x - \frac{g(x)}{g'(x)}, \text{where } g(x) = f'(x). \tag{1.41}$$

Here $f(x)$ is function to be *minimized*. $G'(x)$ is defined around a critical point 'β' of $f(x)$, if $g'(\beta) = f''(\beta) \neq 0$ and is given by $G'(x) = \frac{g(x)g''(x)}{\{g'(x)\}^2}$.

If we assume that $g''(x) \neq 0$, we have $G'(x) = 0$ if $g(\beta) = 0$. Consider an equation $g(x) = 0$, whose one or more roots are to be found.

Let $y = g(x)$ represents the graph of the function $g(x)$ and assume that an initial estimate '$x_0$' is known for an optimum point 'β' of an equation $g(x) = 0$.

Let us sketch an ellipse

$$\frac{(x-x_0)^2}{\alpha'^2} + \frac{\{y-g(x_0)\}^2}{\{g(x_0)\}^2} = 1, \tag{1.42}$$

for some $\alpha' \in \mathbb{R} - \{0\}$.

Let

$$x_1 = x_0 + h, |h| << 1, \tag{1.43}$$

be a better approximation to the required root. Assume that one of the points of intersections of this graph with an ellipse (2.65) is at a point $(x_1, g(x_0+h))$. Now the ellipse (2.65) while passing through the point $(x_1, g(x_0+h))$ takes a form

$$\alpha^2 h^2 \{g(x_0)\}^2 + \{g(x_0+h)\}^2 - 2g(x_0+h)g(x_0) = 0, \tag{1.44}$$

where $\alpha' = \frac{1}{\alpha}$.

Expanding left-hand side by Taylor's series expansion about a point $x = x_0$ and neglecting second and higher-order derivatives, one can have

$$\alpha^2 h^2 \{g(x_0)\}^2 + \{g(x_0) + hg'(x_0) + O(h^2)\}^2 - 2\{g(x_0) + hg'(x_0) + O(h^2)\}\, g(x_0) = 0. \tag{1.45}$$

Simplifying (1.45), one gets first approximation to the required root as

$$x_1 = x_0 \pm \frac{g(x_0)}{\sqrt{\{g'(x_0)\}^2 + \alpha^2\{g(x_0)\}^2}}, \tag{1.46}$$

where the positive sign is taken if $x_0 < \beta$ and the negative sign is taken if $x_0 > \beta$. Geometrically, if slope of curve $\{g'(x_0)\}$ at a point $\{x_0, g(x_0)\}$ is negative, then we take positive sign otherwise negative.

Now repeating this process until an ellipse becomes a point ellipse on $x - axis$, a general formula of successive approximations is given by

$$x_{n+1} = x_n \pm \frac{g(x_n)}{\sqrt{\{g'(x_n)\}^2 + \alpha^2\{g(x_n)\}^2}}. \tag{1.47}$$

Since $g(x_n) = f'(x_n)$, therefore successive iterative process becomes

$$x_{n+1} = x_n \pm \frac{f'(x_n)}{\sqrt{\{f''(x_n)\}^2 + \alpha^2\{f'(x_n)\}^2}}. \tag{1.48}$$

This is a modification over method (1.30) for solving unconstrained optimization problems. The advantage of this method is that, it converges quadratically and has a same erroneous equation as that of Newton's method. Moreover, this method **does not fail even if** $f''(x_n) = 0$ or very small in the vicinity of the required optimum point unlike Newton's method for solving unconstrained optimization problems.

Now, a mathematical proof is presented for an order of convergence of the proposed iterative method (1.48).

**Theorem 1.3.3.** *Let* $f : I \subseteq \mathbb{R} \to \mathbb{R}$ *be a sufficiently differentiable function defined on an open interval I and* $x = \beta \in I$ *be an optimum point of* $f(x)$. *Assume that initial guess* $x = x_0$ *is sufficiently close to* 'β' *in I. Then an iteration scheme defined by formula* (1.50) *is quadratically convergent and satisfies the following erroneous equation :*

$$e_{n+1} = 3A_3 e_n^2 + O(e_n^3). \tag{1.49}$$

**(iv) Exponential iterative methods**

One can also expand the proposed methods to exponentially quadratically convergent iterative formulae for solving unconstrained optimization problems.

If one consider $x_{n+1} = x_n \exp\left(-\frac{h}{x_n}\right)$ be a better approximation to exact optimum point 'β', then from equations (1.49), (1.50) and (1.51) one can obtain the following exponential iteration formulae:

$$x_{n+1} = x_n \exp\left(\frac{f'(x_n)}{f''(x_n) - \alpha f'(x_n)}\right), \tag{1.50}$$

$$x_{n+1} = x_n \exp\left(\frac{2f'(x_n)}{f''(x_n) \pm \sqrt{\{f''(x_n)\}^2 + 4\alpha^2\{f'(x_n)\}^2}}\right), \tag{1.51}$$

and

$$x_{n+1} = x_n \exp\left(\pm\frac{f'(x_n)}{\sqrt{\{f''(x_n)\}^2 + \alpha^2\{f'(x_n)\}^2}}\right), \tag{1.52}$$

respectively.

Letting $\alpha \to 0$ in these formulae, one can derive another exponential iterative formula given by

European Journal of Business & Social Sciences
Available at https://ejbss.org/
ISSN: 2235-767X
Volume 07 Issue 01
January 2019

$$x_{n+1} = x_n \exp\left(-\frac{f'(x_n)}{x_n f''(x_n)}\right). \tag{1.53}$$

Note that by taking the first-order Taylor's series expansion of $\exp\left(-\frac{f'(x_n)}{x_n f''(x_n)}\right)$ in equation (1.53), Newton's method for solving unconstrained optimization problems can be achieved. Recently, Kahya [12] also derived similar methods namely (1.40) and (1.45) by using different approach based on the ideas of Mamta et. al.

1.3.2 TEST EXAMPLES

Here, let us consider some examples to compare number of iterations, computation order of convergence **(COC)** and total number of function evaluations **(TNOFE)** needed in the traditional Newton's method and its modifications namely, modified version of Newton's method (1.40) (***MVNM***), modified version of parabolic method (1.45) (***PVNM***) and modified version of ellipse method (1.54) (***EVNM***) respectively, for solving unconstrained optimization problems. Here for simplicity, the formulae are tested for $|\alpha| = 1$. The results are summarized in Table 1.3 and Table 1.4 respectively. Computations have been performed by using ***MATLAB***® version 7.5 (***R***2007*b*) in double precision arithmetic. We use $\varepsilon = 10^{-15}$ as a tolerance error. The following stopping criteria are used for computer programmes:

$$(i)\ |x_{n+1} - x_n| < \varepsilon, \qquad (ii)\ |f'(x_{n+1})| < \varepsilon$$

**TABLE 3.3:Test initial guesses** ($x_0$)**, optimal point** (β)**, number of iterations and optimal value.**

| Example No. | Examples | Initial guesses | Optimal point (β) | NM | MVNM (2.50) | MVPM (2.59) | MVEM (2.73) | Optimal value |
|---|---|---|---|---|---|---|---|---|
| 3.3.1 | $x3 - 6x2 + 9x - 8 = 0.$ | 2 | 3 | Fails | 1 | 1 | 1 | −8 |
| | | 3.5 | | 5 | 6 | 5 | 5 | |
| | | 0 | | Fails | 8 | 10 | 8 | |
| 3.3.2 | $x4 - x - 10 = 0.$ | 1 | 0.62996052 49474366 | 6 | 7 | 6 | 6 | −10.47247 039371058 |
| | | 2 | | 8 | 10 | 8 | 8 | |
| | | −2 | | Fails | 1 | 1 | 1 | |
| 3.3.3 | $x \exp(x) - 1 = 0.$ | −1.5 | −1 | 7 | 1 | 6 | 6 | - 1.36787944 11714423 |
| | | 0 | | 7 | 8 | 7 | 7 | |
| 3.3.4 | $\frac{3774.522}{x} + 2.27x - 181.529 = 0.$ | 32 | 40.7772610 9029923 | 5 | 17 | 14 | 13 | 3.59971630 30155524 |
| | | 45 | | 5 | 11 | 9 | 8 | |
| 3.3.5 | $(x - 2)2 + \cos(x) = 0$ | 1 | 2.35424275 8222781 | 5 | 7 | 5 | 5 | - 0.58023742 06231672 |
| | | 3 | | 4 | 6 | 5 | 5 | |
| 3.3.6 | $\exp(x) - 3x2 = 0.$ | −1 | 0.20448144 933399151 | 4 | 7 | 6 | 6 | 1.10145070 6667036 |
| | | 1 | | 5 | 6 | 5 | 5 | |
| 3.3.7 | $\frac{10.2}{x} + 6.2x^3 = 0, x > 0.$ | 0.5 | 0.86054147 55706750 | 6 | 6 | 6 | 6 | 15.8040029 2848297 |
| | | 2 | | 5 | 7 | 6 | 6 | |

**TABLE 3.4 : Computational order of convergence and Total number of function evaluations**

| | (COC) | | | | (TONFE) | | | |
|---|---|---|---|---|---|---|---|---|
| **Example No.** | **NM** | **MVNM** (2.50) | **MVPM** (2.59) | **MVEM** (2.73) | **NM** | **MVNM** (2.50) | **MVPM** (2.59) | **MVEM** (2.73) |
| 2.3.1 | == | ≡≡ | ≡≡ | ≡≡ | = | 2 | 2 | 2 |
| | 2.00 | 2.00 | 2.00 | 2.00 | 10 | 12 | 10 | 10 |
| | == | 2.01 | 2.00 | 2.00 | = | 16 | 20 | 16 |
| 2.3.2 | 2.00 | 2.01 | 2.00 | 2.00 | 12 | 14 | 12 | 12 |
| | 2.00 | 2.01 | 2.00 | 2.00 | 16 | 20 | 16 | 16 |
| | == | ≡≡ | ≡≡ | ≡≡ | = | 2 | 2 | 2 |
| 2.3.3 | 2.00 | ≡≡ | 2.01 | 2.00 | 14 | 2 | 12 | 12 |
| | 2.00 | 2.00 | 2.00 | 2.00 | 14 | 16 | 14 | 14 |
| 2.3.4 | 2.00 | 2.00 | 2.14 | 2.05 | 10 | 34 | 28 | 26 |
| | 2.00 | 2.00 | 1.92 | 1.81 | 10 | 22 | 18 | 16 |
| 2.3.5 | 2.00 | 2.00 | 1.83 | 1.99 | 10 | 14 | 10 | 10 |
| | 2.00 | 2.00 | 2.01 | 2.00 | 8 | 12 | 10 | 10 |
| 2.3.6 | 2.01 | 2.00 | 2.00 | 2.04 | 8 | 14 | 12 | 12 |
| | 2.00 | 2.00 | 2.00 | 2.00 | 10 | 12 | 10 | 10 |
| 2.3.7 | 2.00 | 2.00 | 2.00 | 2.00 | 12 | 12 | 12 | 12 |
| | 2.01 | 2.00 | 2.00 | 2.00 | 10 | 14 | 12 | 12 |

Here, the symbols namely

(i) '==' and '=' represent **COC** and **TNOFE** for a case of failure respectively.

(ii) '≡≡' represents **COC** when the number of iterations is not sufficient.

## 1.4 A NEW MODIFIED FAMILY OF NEWTON'S METHOD FOR SYSTEMS OF NONLINEAR EQUATIONS

Nonlinearity is the interest of the physicists and mathematicians, since more physical systems are inherently nonlinear in nature. One of the most important problems in optimization and computational mathematics is to solve systems of nonlinear equations (1.3). Systems of nonlinear equations are difficult to solve in general. The best way to solve these equations is by using iterative methods. Many robust and efficient methods are brought forward. One of the classical methods to solve systems of nonlinear equations is Newton's method [3], which is given by

$$x^{(n+1)} = x^{(n)} - [J_F(x^{(n)})]^{-1} F\left(x^{(n)}\right), \tag{1.54}$$

where $J_F(x)$ is the Jacobian matrix of the function $F_{(x)}$ and $[J_F(x)]^{-1}$ is its inverse.

This basic method converges quadratically. Although, Newton's method for simple roots is generalized to a system of nonlinear equations, convergence of iteration is a serious problem. Unless a good initial guess is known, it is extremely unlikely that an iteration will converge. Moreover, convergence of Newton's method must require a condition that the Jacobian is non-singular in a neighborhood of the root. If this condition is not provided, then Newton's method may be divergent and even fail. Many attempts have been made to develop new techniques for solving systems of nonlinear equations.

The purpose of this attempt is to develop modified family of Newton's method for solving systems of nonlinear equations. The developed family is found to yield better performance than Newton's method.

### 1.4.1 NEW MODIFIED FAMILY OF NEWTON'S METHOD

European Journal of Business & Social Sciences
Available at https://ejbss.org/
ISSN: 2235-767X
Volume 07 Issue 01
January 2019

The proposed scheme given by formula (2.10) can further be expanded for the case of system of nonlinear equations. In general, we can usually find the solutions to a system of equations when the number of unknowns matches the number of equations.

To illustrate the extension of proposed scheme (3.10), consider the following system of nonlinear equations in '$x$' and '$y$':

$$\left.\begin{aligned} f(x,y) &= 0, \\ g(x,y) &= 0. \end{aligned}\right\}. \tag{1.55}$$

Consider the auxiliary equations

$$\left.\begin{aligned} e^{-\alpha(x-x_0)}\ f(x,y) &= 0, \\ e^{-\alpha(y-y_0)}\ g(x,y) &= 0. \end{aligned}\right\}, \tag{1.56}$$

where $\alpha \in \mathbb{R}$. Root of the system of equations (3.83) is also the root of the system (3.84) and vice-versa.

Let ($x_0$, $y_0$) be an initial guess to the required root of a system (3.84). If ($x_0 + h$, $y_0 + s$) is a root of the system (3.84), then one must have

$$\left.\begin{aligned} e^{-\alpha h} f(x_0+h, y_0+s) &= 0, \\ e^{-\alpha k} g(x_0+h, y_0+s) &= 0. \end{aligned}\right\}. \tag{1.57}$$

Assuming that $f$ ($x$, $y$) and $g$ ($x$, $y$) are sufficiently differentiable, one can expand (1.63) by Taylor's series expansion about a point ($x_0$, $y_0$) to obtain

$$\left.\begin{aligned} h\left(\frac{\partial f}{\partial x_0} - \alpha f_0\right) + s\frac{\partial f}{\partial y_0} + f_0 \ldots\ldots\ldots\ldots &= 0, \\ h\frac{\partial g}{\partial x_0} + s\left(\frac{\partial g}{\partial y_0} - \alpha g_0\right) + g_0 \ldots\ldots\ldots\ldots &= 0. \end{aligned}\right\}, \tag{1.58}$$

where

$$\frac{\partial f}{\partial x_0} = \left[\frac{\partial f}{\partial x}\right]_{x=x_0}, \ \frac{\partial f}{\partial y_0} = \left[\frac{\partial f}{\partial y}\right]_{y=y_0}, \ f_0 = f(x_0, y_0),$$

$$\frac{\partial g}{\partial x_0} = \left[\frac{\partial g}{\partial x}\right]_{x=x_0}, \ \frac{\partial g}{\partial y_0} = \left[\frac{\partial g}{\partial y}\right]_{y=y_0}, \ g_0 = g(x_0, y_0) \text{ etc.}$$

Neglecting second and higher-order terms in (1.64), one gets a following system of linear equations :

$$\left.\begin{aligned} h\left(\frac{\partial f}{\partial x_0} - \alpha f_0\right) + s\frac{\partial f}{\partial y_0} + f_0 &= 0, \\ h\frac{\partial g}{\partial x_0} + s\left(\frac{\partial g}{\partial y_0} - \alpha g_0\right) + g_0 &= 0. \end{aligned}\right\}. \tag{1.59}$$

If the Jacobian given by

$$J(f,g) = \begin{vmatrix} \left(\frac{\partial f}{\partial x_0} - \alpha f_0\right) & \frac{\partial f}{\partial y_0} \\ \frac{\partial g}{\partial x_0} & \left(\frac{\partial g}{\partial y_0} - \alpha g_0\right) \end{vmatrix}, \tag{1.60}$$

does not vanish, then linear system of equations (2.65) possesses a unique solution given by

European Journal of Business & Social Sciences
Available at https://ejbss.org/
ISSN: 2235-767X
Volume 07 Issue 01
January 2019

$$\left.\begin{aligned} h &= \frac{g_0\frac{\partial f}{\partial y_0} - f_0\left(\frac{\partial g}{\partial y_0} - \alpha g_0\right)}{J(f,g)}, \\ s &= \frac{f_0\frac{\partial g}{\partial x_0} - g_0\left(\frac{\partial f}{\partial x_0} - \alpha f_0\right)}{J(f,g)}. \end{aligned}\right\}. \tag{1.61}$$

Therefore, new approximations are given by

$$\left.\begin{aligned} x_1 &= x_0 + h, \\ y_1 &= y_0 + s. \end{aligned}\right\}. \tag{1.62}$$

The process is to be repeated till one obtains the roots of desired accuracy. The parameter 'α' in (1.62) is chosen so as to give largest value of denominator. Method (1.63) will work even if $\frac{\partial f}{\partial x_0}\frac{\partial g}{\partial y_0} - \frac{\partial f}{\partial y_0}\frac{\partial g}{\partial x_0} = 0$ unlike Newton's method.

Again, it is interesting to note that by ignoring the term in 'α', the equation (1.63) reduces to Newton's method for solving systems of nonlinear equations.

**Vector notation**

Let us consider a general system of nonlinear equations.

$$F(x) = 0, \tag{1.63}$$

where $(x) = (f_1(x), f_2(x), \dots \dots, f_n(x))^T$ , $f_i : \mathbb{R}^n \to \mathbb{R}^n$and $x = (x_1, x_2, \dots \dots, x_n)^T$ is a vector.

Then the modified Newton's method (1.63) (***MNM***) may be written as

$$x^{n+1} = x^{(n)} - \left[F'\left(x^{(n)}\right) - diag\left(\alpha_i^{(n)} f_i\left(x^{(n)}\right)\right)\right]^{-1} F\left(x^{(n)}\right), \tag{1.64}$$

where $i = 1,2,\dots, n$. In case when Jacobian $F'\left(x^{(n)}\right)$ is numerically singular, then parameter $\alpha_i^{(n)}$ is chosen such that $\left[F'\left(x^{(n)}\right) - diag\left(\alpha_i^{(n)} f_i\left(x^{(n)}\right)\right)\right]$ is non-singular then the modified Newton's method (1.65) can continue computation.

Order of convergence of the proposed iterative formula (1.65) is established in the following theorem :

**Theorem 1.4.1.** *Let* $F : D \subset \mathbb{R}^n \to \mathbb{R}^n$ *for an open convex set D. Assume that F(x) has a root* $x^{(*)} \in D$, *where the Jacobian* $F'\left(x^{(*)}\right)$ *is non-singular and F(x) is sufficiently smooth in some neighbourhood S of the root. If for all* $x \in S$, $\left[F'(x) - diag\left(\alpha_i f_i(x)\right)\right]$ *is non-singular, then proposed method 1.70) has quadratic order of convergence.*

1.4.2 TESTED EXAMPLES

Now, some numerical results are presented for method (1.65) in Table 1.5. Compared the Newton's method (***NM***) and the modified Newton's method (1.70) (***MNM***) for solving systems of nonlinear equations, computations have been performed by using ***MATLAB***® version 7.5 (***R***2007*b*) in double precision arithmetic. Here, the formulae are tested for |α| = 1. We use $\varepsilon = 10^{-15}$ as a tolerance error. The following stopping criteria are used for computer programmes:

$$(i)\ \left\|x^{(n+1)} - x^{(n)}\right\| < \varepsilon, \qquad (ii)\ \left\|F\left(x^{(n)}\right)\right\| < \varepsilon.$$

The numbers of iterations needed to converge to the required solution have been analyzed.

European Journal of Business & Social Sciences
Available at https://ejbss.org/
ISSN: 2235-767X
Volume 07 Issue 01
January 2019

Consider the following systems of nonlinear equations

**Example 1.4.1**

$$\left.\begin{aligned} x_1 - \cos(x_2) = 0, \\ \sin(x_1) + 0.5x_2 = 0. \end{aligned}\right\}.$$

Solution is $(0.5303886895389944, -1.1011737334182012)^T$ .

**Example 1.4.2**

$$\left.\begin{aligned} x_1x_2 - 1 = 0, \\ \ln(x_1) - x_2 = 0. \end{aligned}\right\}.$$

Solution is $(1.763222834351897, 0.5671432904097838)^T$ .

**Example 1.4.3**

$$\left.\begin{aligned} x_1^2 - x_2^2 - 1 = 0, \\ x_1^3x_2^2 - 1 = 0. \end{aligned}\right\}.$$

Solution is $(1.236505703391499, 0.7272869822289587)^T$.

**Example 1.4.4**

$$\left.\begin{aligned} x_1^2 - x_2^2 - 4 = 0, \\ x_1^2 + x_2^2 - 16 = 0. \end{aligned}\right\}.$$

Solution is $(3.162277660168379, 2.449489742783178)^T$ .

**TABLE 1.5 : Test examples, their initial guesses and number of iterations**

| Example No. | Initial guesses | NM | MNM (2.92) |
|---|---|---|---|
| 1.4.1 | $(x_1, x_2)^T = (0, 0)^T$ | 6 | 5 |
| | $(x_1, x_2)^T = (2, 2)^T$ | Divergent | 10 |
| 1.4.2 | $(x_1, x_2)^T = (1, 1)^T$ | 5 | 6 |
| | $(x_1, x_2)^T = (6, -1)^T$ | Fails | 13 |
| 2.4.3 | $(x_1, x_2)^T = (0, 0)^T$ | Fails | 6 |
| | $(x_1, x_2)^T = (1, 2)^T$ | 6 | 6 |
| 2.4.4 | $(x_1, x_2)^T = (0, 0)^T$ | Fails | 9 |
| | $(x_1, x_2)^T = (3, 2)^T$ | 5 | 6 |

## 1.5 DISCUSSION AND CONCLUSIONS

The behaviors of Newton's method and proposed modifications can be compared by their correction factors. For example, Newton's method correction factor $\frac{f(x_n)}{f'(x_n)}$ is now modified to $\frac{f(x_n)}{f'(x_n)-\alpha f(x_n)}$, where parameter 'α' is chosen such that corresponding function values $f'(x_n)$ and $\alpha f(x_n)$ have opposite signs. However, for $\alpha = 0$ and if derivatives of function $f(x_n)$ are singular or almost singular, Newton's method will either fail or diverge. Therefore, these modifications have two remarkable advantages over Newton's method namely : (i) if $\alpha \neq 0$, the modified denominator of proposed modifications is well-defined and is never zero, provided $x_n$ is not accepted as an approximation to the required root or optima respectively and hence they are well-defined even if $f'(x_n) = 0$ or $f''(x_n) = 0$ happens, (ii) the absolute value of the modified denominator of modified techniques is always greater than denominator of Newton's method i.e.

$|f'(x_n)|$ provided $x_n$ is not accepted as an approximation to the required root or optima respectively. This means that proposed methods are numerically more stable unlike Newton's method. Further, numerical results demonstrate that parabolic and ellipse methods outperform Newton's method and one-parameter family of Newton's method for solving nonlinear equations as well as unconstrained optimization problems.

Furthermore, for systems of nonlinear equations, a new family of Newton's method has been proposed for finding a zero of vector function. The benefits of the proposed family is that it works even if Jacobian is singular at some points. From numerical results, one can see that proposed method is much superior in global convergence to Newton's method and this method is very efficient when Jacobian is singular. However, proposed methods for solving nonlinear equations, unconstrained optimization problems and systems of nonlinear equations could not be expanded for a case of complex roots.

## REFERENCES


1. Fletcher, F., Practical Methods of Optimization, *John Wiley and Sons Ltd.*, 1987.
2. Wait, R., The Numerical Solution of Algebraic Equations, John Willey and Sons, 1979.
3. Amat, S., Busquier, S., Candela, V.F. and Potra, F.A, Convergence of third-order iterative methods in Banach spaces, *Preprint U.P. Caratgena*, **16** (2001).
4. Aslamnoor, M. and Waseem, M., Some iterative methods for solving a system of nonlinear equations, *Computer Mathematics with Applications,* **57**(1) (2009), 101-106.
5. Atkinson, K.E., An Introduction to Numerical Analysis, *John Willey and Sons Inc*, second edition, 1989.
6. Awaweh, F., On new iterative method for solving systems of nonlinear equations, *Numerical Algorithm,* **54**(3) (2009), 395-409.
7. Babajee, D.K.R., Several improvements of the two-point third-order midpoint iterative method using weight function, *Applied Mathematics and Computation*, **218**(15) (2012), 7958-7966.
8. Babajee, D.K.R., Dauhoo, M.Z., Darvishi, M.T., Karami, A. and Barati, A., Analysis of two Chebyshev-like third-order methods free from second derivatives for solving system of nonlinear equations, *Journal of Computation and Applied Mathematics*, **233** (2010), 2002-2012.
9. Basu, D., Family of cubically convergent schemes using inverse Newton function for multiple roots, *International Journal for Computational Methods in Engineering Science and Mechanics,* **9**(3) (2008), 189-200.
10. Wang, X. and Wei, L., The Julia set of Newton's method for multiple roots, *Applied Mathematics and Computation*, **172**(1) (2006), 101-110.
11. Weerakoon, S. and Fernando, T.G.I., A variant of Newton's method with accelerated third-order convergence, *Applied Mathematics Letters*, **13**(8) (2000), 87-93. *Computation* **215**(3) (2009), 1288-1292.
12. Li, S. G., Cheng, L.Z. and Neta, B., Some fourth-order nonlinear solvers with closed formulae for multiple roots, *Computer and Mathematics with Applications*, **59**(1) (2010), 126-135.
13. Li, X., Mu, C., Ma, J. and Wang, C., Sixteen-order method for nonlinear equations, *Journal of Computational and Applied Mathematics*, **215**(10) (2010), 3754-3758.