Mehran University Research Journal of Engineering and Technology

https://doi.org/10.22581/muet1982.2301.15

2023, 42(1) 165-176

A new family of twentieth order convergent methods with applications to nonlinear systems in engineering

Hameer Akhtar Abro [a], Muhammad Mujtaba Shaikh [a,b,*]

[a] *Department of Basic Sciences and Related Studies, Mehran University of Engineering and Technology, Jamshoro, Pakistan*

[b] *Supply Chain and Operations Management Research Group, Mehran University of Engineering and Technology, Jamshoro, Pakistan*

[*] Corresponding author: Muhammad Mujtaba Shaikh, Email: mujtaba.shaikh@faculty.muet.edu.pk

Received: 21 October 2019, Accepted: 19 September 2022, Published: 01 January 2023

KEYWORDS

Iterative Methods

Convergence Analysis

Computational Efficiency

NASA's "Wind" Equation

Tank-Reactor Problem

Neurophysiology Application

Combustion Problem

Kinematic Synthesis

Boundary-Value Problem

ABSTRACT

A new family of iterative methods with a strong converging order of twenty to solve nonlinear equations and systems is presented in this study. A simple strategy of blending some existing methods is used to develop the proposed family. The theoretical order of convergence is derived by employing Taylor's series. The performance of the iterative methods in the proposed family is examined by applying the methods on real-world engineering problems. A nonlinear equation modeled by NASA for launching "Wind" satellite and some other complex applied systems, such as combustion problem, tank-reactor problem, kinematic synthesis mechanism, neurophysiology application and one boundary-value problem, have been solved to check the performance of the proposed family against other methods under similar test conditions. All the numerical results show that the proposed family converges very fast in complex and difficult problems as compared to other well-known methods. The methods in the proposed family have an efficiency improvement of 11.99% over the classical Newton method for scalar nonlinear equations.

1. Introduction

Let $\underset{\sim}{\mathrm{F}}(\mathrm{x}) = \left(f_1(\mathrm{x}), f_2(\mathrm{x}), \ldots, f_n(\mathrm{x})\right)^t$ be a nonlinear system of n algebraic equations in n unknowns which can also be written as $\mathrm{F}(\mathrm{x}) = 0$, where $\mathrm{x} = (x_1, x_2, \ldots, x_n)^T$. Finding the solutions to such systems is very important and challenging task in computational mathematics. The task is more complicated when nonlinear equations are involved in systems. The systems of nonlinear equations are used in many mathematical and engineering problems. Some case study nonlinear systems like neurophysiology application, chemical equilibrium problem, kinematics problem, combustion application, economics modeling problem, arithmetic benchmark problem were studied in [1]. Awawdeh [2] highlighted the applications iterative methods on nonlinear systems for non-adiabatic stirred tank reactors and steering problem.

The systems of nonlinear equations also result from the discretization of nonlinear differential equations by using well-known finite element, finite volume and finite difference methods. Some nonlinear systems appearing after the discretization of nonlinear elliptic partial differential equation by means of finite element method were discussed in [3]. Similarly, nonlinear systems arise from the discretization of second order nonlinear ordinary differential equation by employing finite difference method [4], nonlinear ordinary and partial differential equations by using Chebyshev pseudo-spectral collocation method [5], and nonlinear one-dimensional heat conduction equation by using finite difference method [6]. The discretization process of nonlinear integral equations also require nonlinear systems to be solved such as Chandrasekhar's integral equation related to radiative transfer theory, gas kinetic theory and transport of neutrons in [4], [7-10] and Fredholm–Volterra Hammerstein integral equation in [11-12].

In situation where important variables, representing important properties of physical systems, are related in form of a nonlinear system, the numerical methods play a vital role to acquire robust, accurate and stable solution. Due to this importance, many iterative methods have been developed to solve such systems. The well-known classical Newton's method (NM) [13] is a very basic technique to find the solutions of systems of nonlinear equations. The method is defined by the well-known scheme:

$$\mathrm{x}_{i+1} = \mathrm{x}_i - \mathrm{F}'(\mathrm{x}_i)^{-1}\mathrm{F}(\mathrm{x}_i),\ i = 0,1,2,\ldots \quad (1)$$

The NM requires one evaluation of F and one evaluation of its Jacobian which comprises of first order partial derivatives of F. The NM finds solution with second order convergence. In recent years, many modifications to NM have been proposed in order to accelerate the order of convergence, and to get the solution in fewer numbers of iterations. It is obvious that the new modifications with higher order convergence need extra evaluations of function and higher order partial derivatives. Noor [14] proposed an iterative method of third order convergence in 2006 by using first and second order derivatives. Noor [15] presented third and fourth order iterative methods which require five and eight new functions and derivatives respectively in 2010. Waseem [8] developed a fifth order method in 2016 which require five functional evaluations. Srivastava [3] proposed fifteenth order convergent iterative method to solve nonlinear systems in 2016 requiring seven evaluations of functional and first order partial derivatives per iteration. Zhong Yong presented an iterative method of ninth order convergence for nonlinear systems in [16]. Raza [17] proposed an eleventh order convergent method for nonlinear equations and systems.

Different techniques have been used in past to develop higher order convergent methods such as Adomian decomposition method [18], rules of quadrature [19], method of variational iteration [20] and homotopy perturbation method [21]. Shaikh et al. [22,23] attempted to solve the nonlinear Colebrook's equation for Darcy friction factor in rough pipes under highly turbulent flow by utilizing an iterative scheme based on the use of fixed-point method in a 1000 by 1000 mesh of Reynolds number and relative roughness values. The developed database was designed for further studies on explicit equations for the Colebrook's equation with soft computing techniques.

The order of convergence of methods may also be accelerated by combining two different iterative methods [24]. Two methods of converging orders q_1 and q_2 respectively may be combined to achieve a new iterative method of order $q_1 q_2$. This technique may require new evaluations of functions and derivatives but results in rapid acceleration of order of convergence. The technique of [24] is referred here as the blending technique. The main objective of this paper is to accelerate the convergence order of iterative methods to obtain the desirable solution in lesser number of iterations. To do so, we use the same blending technique, i.e. combining two different iterative methods to develop some higher order convergent methods, particularly, a family of twentieth order convergent methods.

The paper is arranged as follows: Section-1 highlights importance of nonlinear solvers in mathematics, science and engineering. Related literature on the proposition of nonlinear solvers and their performance is added, and the main objectives of this study are described. Section-2 contains some important results which are required to prove the order of convergence. The development of the proposed family and theorems regarding its theoretical order of convergence using Taylor's expansion are discussed. The efficiency indices of the methods in proposed family and some other well-known methods are also discussed in Section-2. The numerical setup to verify performance of proposed family and the some applied case study nonlinear systems of equations from engineering and science are presented in Section-3. The application of the proposed methods is also discussed with comments on their performance in Section-3. Finally, the main contributions of this paper are presented in the conclusion, Section-4.

2. Material and Methods

In this section, we start with some basic definitions and concepts. The general form of the proposed family is discussed and theorems concerning the order of convergence of methods in the proposed family are proved. We also compare the efficiency indices of proposed methods against some other existing methods.

2.1 Some Basic Concepts

Here, a few important definitions and a lemma are presented to support the main contributions of this study.

Definition 1. [4]

Let $\{x_i\}$, $i \geq 1$ be a sequence in R^n that converges to s. Then the sequence is said to be of converging order p, p>1, if there exists $M, M > 0$, and i_0 such that

$$\|e_{i+1}\| \leq M\|e_i\|^p \quad \forall i \geq i_0, \tag{2}$$

where $e_i = x_i - s$

Definition 2. [4]

For multidimensional case, if $e_i = x_i - s$ be the error at i^th iteration, then the error equation is defined as,

$$e_{i+1} = L(e_i)^p + O(e_i)^{p+1} \tag{3}$$

where, p represents the converging order and L is a p-linear function, i.e. $L \in £(R^n \times \ldots \times R^n, R^n)$, £ is the set of linear functions and $e_i = (e_i, e_i, \ldots, e_i)$.

Lemma 1. [26]

Let $F: Q \subseteq R^n \to R^n$ be a p-times Frechet differentiable in a convex set $Q \subseteq R^n$, then for any x, $h \in R^n$ the following expression holds true:

$$F(x+h) = F(x) + F'(x)h + \frac{1}{2!}F''(x)h^2$$

$$+\frac{1}{3!}F'''(x)h^3 + \ldots + +\frac{1}{u!}F^{(u-1)}(x)h^{(u-1)} + R_u \tag{4}$$

where $\|R_u\| \leq \frac{1}{u!}\sup_{0<t<1}\left\|F^{(u)}(x+th)\right\|\|h\|^u$ and

$h^u = (h, h, \overset{u}{\ldots}, h)$.

2.2 Development of Proposed Family of Nonlinear Iterative Methods

Following the blending strategy suggested in [24], we present a new family of iterative methods having twentieth order of convergence by blending a fourth order method in [25] and the fifth order composite Newton-Traub method in [4]. The general form of the proposed family of five-step iterative methods for nonlinear equations and systems can be defined as:

$$\begin{cases} a_i = x_i - \frac{2}{3}F'(x_i)^{-1}F(x_i) \\ y_i = x_i - \begin{bmatrix} \left(-\frac{1}{2}\right)I + \frac{9}{8}F'(a_i)^{-1}F'(x_i) \\ +\frac{3}{8}F'(x_i)^{-1}F'(a_i) \end{bmatrix} F'(x_i)^{-1}F(x_i) \\ z_i = y_i - \psi F'(y_i)^{-1}F(y_i) \\ w_i = y_i - \left[\left(1+\frac{1}{2\psi}\right)I - \frac{1}{2\psi}F'(y_i)^{-1}F'(z_i)\right]F'(y_i)^{-1}F(y_i) \\ x_{i+1} = w_i - [AI + BF'(y_i)^{-1}F'(z_i)]F'(y_i)^{-1}F(w_i) \\ i = 0,1,2,\ldots \end{cases} \tag{5}$$

where, I is the identity matrix of order n, $\psi \neq 0$ is a fixed real number, and the parameters *A*, *B*, depending on ψ, are chosen so that the order of convergence of the proposed methods is twenty.

2.3 Convergence Analysis of the Proposed Family

We first prove order of convergence of methods in the proposed family for any fixed real number $\psi \neq 0$. The conditions on the parameters A and B so that the proposed methods have twentieth order of convergence are also outlined. Then, some special cases of the proposed methods will be discussed in the nest section.

To prove the order of convergence for proposed family, we use the Taylor's expansion for vector functions defined in Lemma 1 from [26].

Theorem 1.

Let the function $F: Q \subseteq R^n \to R^n$be sufficiently differentiable in a convex set Q containing the zeros of F(x). Let us consider that $F'(x)$ is continuous and nonsingular in s, then, the solution x obtained by using (5) converges to s with twentieth order convergence, if x_0 close to s is taken as initial guess.

Proof of Theorem 1.

Let $e_i = x_i - s$. Using Taylor's expansion for $F(x_i)$,

$$F(x_i) = F'(\alpha)\left[e_i + c_2e_i^2 + c_3e_i^3 + c_4e_i^4 + O\left(e_i^5\right)\right] \tag{6}$$

where, $c_j = \frac{F^j(\alpha)}{j!F'(\alpha)}, j = 2,3,\ldots$

Using Taylor's expansion for $F'(x_i)$,

$$F'(x_i) = F'(\alpha)[I + 2c_2e_i + 3c_3e_i^2 + 4c_4e_i^3 + O(e_i^4)] \tag{7}$$

The inversion of $F'(x_i)$ becomes,

$$F'(x_i)^{-1} = F'(\alpha)^{-1}\begin{bmatrix} I - 2c_2e_i + (4c_2^2 - 3c_3)e_i^2 \\ +(12c_2c_3 - 8c_2^3 - 4c_4)e_i^3 \\ +O(e_i^4) \end{bmatrix} \tag{8}$$

The multiplication of (8) and (6) becomes,

$$F'(x_i)^{-1}F(x_i) = e_i - c_2e_i^2 + (2c_2^2 - 2c_3)e_i^3$$

$$+(7c_2c_3 - 4c_2^3 - 3c_4)e_i^4 + O\left(e_i^5\right) \tag{9}$$

Using (9) in first step of (5), we have:

$$\hat{e}_i = a_i - s = \frac{1}{3}\begin{bmatrix} e_i + 2c_2e_i^2 + (4c_3 - 4c_2^2)e_i^3 \\ +(6c_4 + 8c_2^3 - 14c_2c_3)e_i^4 + O(e_i^5) \end{bmatrix} \quad (10)$$

Using Taylor's expansion for $F'(a_i)$,

$$F'(a_i) = F'(\alpha)\begin{bmatrix} I + 2c_2\hat{e}_i + 3c_3\hat{e}_i^2 \\ +4c_4\hat{e}_i^3 + O(\hat{e}_i^4) \end{bmatrix} \quad (11)$$

The inversion of $F'(a_i)$ becomes,

$$F'(a_i)^{-1} = F'(\alpha)^{-1}\begin{bmatrix} I - 2c_2\hat{e}_i + (4c_2^2 - 3c_3)\hat{e}_i^2 \\ +(12c_2c_3 - 8c_2^3 - 4c_4)\hat{e}_i^3 \\ +O(\hat{e}_i^4) \end{bmatrix} \quad (12)$$

The multiplication of (12) and (7) becomes,

$$\begin{aligned} F'(a_i)^{-1}F'(x_i) &= 2c_2e_i + 3c_3e_i^2 + 4c_4e_i^3 \\ &-2c_2\hat{e}_i - 4c_2^2e_i\hat{e}_i - 6c_2c_3e_i^2\hat{e}_i + (4c_2^2 - 3c_3)\hat{e}_i^2 \\ &+(8c_2^3 - 6c_2c_3)e_i\hat{e}_i^2 + (12c_2c_3 - 8c_2^3 - 4c_4)\hat{e}_i^3 \end{aligned} \quad (13)$$

Also the multiplication of (8) and (11) gives:

$$\begin{aligned} F'(x_i)^{-1}F'(a_i) &= -2c_2e_i + (4c_2^2 - 3c_3)e_i^2 \\ &+(12c_2c_3 - 8c_2^3 - 4c_4)e_i^3 + 2c_2\hat{e}_i - 4c_2^2e_i\hat{e}_i \\ &+(8c_2^3 - 6c_2c_3)e_i^2\hat{e}_i + 3c_3\hat{e}_i^2 - 6c_2c_3e_i\hat{e}_i^2 + 4c_4\hat{e}_i^3 \end{aligned} \quad (14)$$

Using (9), (10), (13) and (14) in second step of (5):

$$\tilde{e}_i = y_i - s = \left(\frac{7}{3}c_2^3 - c_2c_3 + \frac{c_4}{9}\right)e_i^4 + O(e_i^5) \quad (15)$$

Using Taylor's expansion for $F(y_i)$,

$$F(y_i) = F'(\alpha)\left[\tilde{e}_i + c_2\tilde{e}_i^2 + c_3\tilde{e}_i^3 + O(\tilde{e}_i^4)\right] \quad (16)$$

$$F'(y_i)^{-1} = F'(\alpha)^{-1}\begin{bmatrix} I - 2c_2\tilde{e}_i + (4c_2^2 - 3c_3)\tilde{e}_i^2 \\ +(12c_2c_3 - 8c_2^3 - 4c_4)\tilde{e}_i^3 \\ +O(\tilde{e}_i^4) \end{bmatrix} \quad (17)$$

The product of (17) and (16) becomes,

$$\begin{aligned} F'(y_i)^{-1}F(y_i) &= \tilde{e}_i - c_2\tilde{e}_i^2 + (2c_2^2 - 2c_3)\tilde{e}_i^3 \\ &+(7c_2c_3 - 4c_2^3 - 3c_4)\tilde{e}_i^4 + O(\tilde{e}_i^5) \end{aligned} \quad (18)$$

Using (18) in third step of (5), we have:

$$\begin{aligned} \bar{e}_i = z_i - s &= (1-\psi)\tilde{e}_i + \psi c_2\tilde{e}_i^2 + \psi(2c_3 - 2c_2^2)\tilde{e}_i^3 \\ &+\psi(3c_4 + 4c_2^3 - 7c_2c_3)\tilde{e}_i^4 + O(\tilde{e}_i^5) \end{aligned} \quad (19)$$

Using Taylor's expansion for $F'(z_i)$,

$$F'(z_i) = F'(\alpha)[I + 2c_2\bar{e}_i + 3c_3\bar{e}_i^2 + 4c_4\bar{e}_i^3 + O(\bar{e}_i^4)] \quad (20)$$

The multiplication of (17) and (20) becomes,

$$\begin{aligned} F'(y_i)^{-1}F'(z_i) &= I - 2c_2\tilde{e}_i + (4c_2^2 - 3c_3)\tilde{e}_i^2 \\ &+(12c_2c_3 - 8c_2^3 - 4c_4)\tilde{e}_i^3 + 2c_2\bar{e}_i - 4c_2^2\tilde{e}_i\bar{e}_i \\ &+(8c_2^3 - 6c_2c_3)\tilde{e}_i^2\bar{e}_i + 3c_3\bar{e}_i^2 - 6c_2c_3\tilde{e}_i\bar{e}_i^2 + 4c_4\bar{e}_i^3 \end{aligned} \quad (21)$$

Using (18) and (21) in fourth step of (5), we have:

$$\begin{aligned} \breve{e}_i = w_i - s &= \left(2c_2^2 + \left(\frac{3}{2}\psi - 1\right)c_3\right)\tilde{e}_i^3 \\ &-\left[9c_2^3 + \begin{pmatrix} \frac{15}{2}\psi \\ -12 \end{pmatrix}c_2c_3 + \begin{pmatrix} 2\psi^2 \\ -6\psi + 3 \end{pmatrix}c_4\right]\tilde{e}_i^4 + O(\tilde{e}_i^5) \end{aligned} \quad (22)$$

Using Taylor's expansion for $F(w_i)$,

$$F(w_i) = F'(\alpha)\left[\breve{e}_i + O(\breve{e}_i^2)\right] \quad (23)$$

Using (17), (21) and (23) in fifth step of (5):

$$\begin{aligned} e_{i+1} &= (1 - A - B)\breve{e}_i + 2(A + (1+\psi)B)\tilde{e}_i\breve{e}_i \\ &-\begin{pmatrix} (4c_2^2 - 3c_3)(A+B) + 10B\psi c_2^2 \\ +3\psi(\psi - 2)Bc_3 \end{pmatrix}\tilde{e}_i^2\breve{e}_i \end{aligned} \quad (24)$$

For a family of twentieth order convergence, we take:

$$A + B = 1 \text{ and } B = -\frac{1}{\psi}. \quad (25)$$

Using (25) in (24), we get:

$$e_{i+1} = (6c_2^2 + 3(\psi - 1)c_3)\tilde{e}_i^2\breve{e}_i \quad (26)$$

Using (15) and (22) in (26), leads to:

$$e_{i+1} = \begin{bmatrix} 6c_2^2 \\ +3(\psi - 1)c_3 \end{bmatrix}\begin{bmatrix} 2c_2^2 \\ +\left(\frac{3\psi}{2} - 1\right)c_3 \end{bmatrix}\begin{pmatrix} \frac{7}{3}c_2^3 \\ -c_2c_3 \\ +\frac{c_4}{9} \end{pmatrix}^5 e_i^{20} + O(e_i^{21}) \quad (27)$$

Equation (27) shows twentieth order of convergence of all methods of the proposed family (5) under imposed conditions on the parameters *A* and *B*, i.e. (25), and for a fixed real number $\psi \neq 0$.

2.4 The Proposed Family with Some Particular Cases

Using the impositions (25) in the general form of the proposed family (5), the final version of the iterative schemes of methods in proposed family take the form:

$$\begin{cases} \mathbf{a}_i = \mathbf{x}_i - \frac{2}{3}\mathbf{F}'(\mathbf{x}_i)^{-1}\mathbf{F}(\mathbf{x}_i) \\ \mathbf{y}_i = \mathbf{x}_i - \begin{bmatrix} \left(-\frac{1}{2}\right)\mathbf{I} + \frac{9}{8}\mathbf{F}'(\mathbf{a}_i)^{-1}\mathbf{F}'(\mathbf{x}_i) \\ +\frac{3}{8}\mathbf{F}'(\mathbf{x}_i)^{-1}\mathbf{F}'(\mathbf{a}_i) \end{bmatrix}\mathbf{F}'(\mathbf{x}_i)^{-1}\mathbf{F}(\mathbf{x}_i) \\ \mathbf{z}_i = \mathbf{y}_i - \psi\mathbf{F}'(\mathbf{y}_i)^{-1}\mathbf{F}(\mathbf{y}_i) \\ \mathbf{w}_i = \mathbf{y}_i - \left[\left(1 + \frac{1}{2\psi}\right)\mathbf{I} - \frac{1}{2\psi}\mathbf{F}'(\mathbf{y}_i)^{-1}\mathbf{F}'(\mathbf{z}_i)\right]\mathbf{F}'(\mathbf{y}_i)^{-1}\mathbf{F}(\mathbf{y}_i) \\ \mathbf{x}_{i+1} = \mathbf{w}_i - \left[\left(1 + \frac{1}{\psi}\right)\mathbf{I} - \frac{1}{\psi}\mathbf{F}'(\mathbf{y}_i)^{-1}\mathbf{F}'(\mathbf{z}_i)\right]\mathbf{F}'(\mathbf{y}_i)^{-1}\mathbf{F}(\mathbf{w}_i) \\ i = 0,1,2,\ldots \end{cases} \quad (28)$$

For some particular cases, we take $\psi = 1, -1, -\frac{1}{2}$. The algorithms of proposed methods A1, A2, A3 with these values are defined in schemes (29), (30) and (31).

$$\begin{cases} a_i = x_i - \frac{2}{3}F'(x_i)^{-1}F(x_i) \\ y_i = x_i - \begin{bmatrix} \left(-\frac{1}{2}\right)I + \frac{9}{8}F'(a_i)^{-1}F'(x_i) \\ +\frac{3}{8}F'(x_i)^{-1}F'(a_i) \end{bmatrix} F'(x_i)^{-1}F(x_i) \\ z_i = y_i - F'(y_i)^{-1}F(y_i) \\ w_i = y_i - \left[\frac{3}{2}I - \frac{1}{2}F'(y_i)^{-1}F'(z_i)\right]F'(y_i)^{-1}F(y_i) \\ x_{i+1} = w_i - [2I - F'(y_i)^{-1}F'(z_i)]F'(y_i)^{-1}F(w_i) \\ i = 0,1,2,\ldots \end{cases} \tag{29}$$

$$\begin{cases} a_i = x_i - \frac{2}{3}F'(x_i)^{-1}F(x_i) \\ y_i = x_i - \begin{bmatrix} \left(-\frac{1}{2}\right)I + \frac{9}{8}F'(a_i)^{-1}F'(x_i) \\ +\frac{3}{8}F'(x_i)^{-1}F'(a_i) \end{bmatrix} F'(x_i)^{-1}F(x_i) \\ z_i = y_i + F'(y_i)^{-1}F(y_i) \\ w_i = y_i - \left[\frac{1}{2}I + \frac{1}{2}F'(y_i)^{-1}F'(z_i)\right]F'(y_i)^{-1}F(y_i) \\ x_{i+1} = w_i - [F'(y_i)^{-1}F'(z_i)]F'(y_i)^{-1}F(w_i) \\ i = 0,1,2,\ldots \end{cases} \tag{30}$$

$$\begin{cases} a_i = x_i - \frac{2}{3}F'(x_i)^{-1}F(x_i) \\ y_i = x_i - \begin{bmatrix} \left(-\frac{1}{2}\right)I + \frac{9}{8}F'(a_i)^{-1}F'(x_i) \\ +\frac{3}{8}F'(x_i)^{-1}F'(a_i) \end{bmatrix} F'(x_i)^{-1}F(x_i) \\ z_i = y_i + \frac{1}{2}F'(y_i)^{-1}F(y_i) \\ w_i = y_i - [F'(y_i)^{-1}F'(z_i)]F'(y_i)^{-1}F(y_i) \\ x_{i+1} = w_i + [I - 2F'(y_i)^{-1}F'(z_i)]F'(y_i)^{-1}F(w_i) \\ i = 0,1,2,\ldots \end{cases} \tag{31}$$

The error equations of methods A1-A3 are defined in (32)-(34), respectively,

$$e_{i+1} = 3c_2^2(4c_2^2 + c_3)\left(\frac{7}{3}c_2^3 - c_2c_3 + \frac{c_4}{9}\right)^5 e_i^{20} + O(e_i^{21}) \tag{32}$$

$$e_{i+1} = 3(c_2^2 - c_3)(4c_2^2 - 5c_3)\begin{pmatrix} \frac{7}{3}c_2^3 - c_2c_3 \\ +\frac{c_4}{9} \end{pmatrix}^5 e_i^{20} + O(e_i^{21}) \tag{33}$$

$$e_{i+1} = \frac{3}{8}(4c_2^2 - 3c_3)(8c_2^2 - 7c_3)\begin{pmatrix} \frac{7}{3}c_2^3 \\ -c_2c_3 \\ +\frac{c_4}{9} \end{pmatrix}^5 e_i^{20} + O(e_i^{21}) \tag{34}$$

The coefficients of the error equations defined in (32)-(34) for methods A1-A3 depend on derivatives of the nonlinear function F(x) near the solution. While all the proposed methods: A1-A3 are twentieth order convergent, their numerical performance to achieve some pre-specified error tolerance may slightly differ due to different coefficients in (32)-(34).

2.5 Efficiency Index

The efficiency index E of an iterative method is calculated by the formula $E=p^{1/m}$, also used by [8] where p denotes the converging order of method and m is the total computational cost (functional and derivative evaluations) taken by that method per iteration. For n dimensional case, the total computational cost m depends on "n" evaluations for new functions and "n^2" evaluations for new first order partial derivatives. We compare the efficiency index of proposed family with well-known second order convergent classical NM and fifteenth order convergent method of Srivastava [3] denoted by M15 in the following discussion. Thus, the efficiency index of proposed family requiring three new functional evaluations and four new first order derivatives per iteration is $20^{1/(3n+4n^2)}$, whereas the efficiency indices of NM and M15 are $2^{1/(n+n^2)}$ and $15^{1/(4n+3n^2)}$ respectively. For n= 1, i.e. for the case of scalar nonlinear equations the efficiency indices of the proposed methods A1-A3, the NM and the M15 methods are 1.53412740, 1.41421356 and 1.47235670. The percentage improvement in efficiency index of the proposed methods over the NM and M15 methods are 11.99% and 6.177%, respectively. The proposed methods are better in efficiency than NM and M15 methods, and many other methods in literature. The comparison of efficiency indices of proposed family, NM and M15 versus order of nonlinear system ($n > 1$) is shown in Fig. 1. Further, it can be observed from Fig. 1, as the size of system increase the efficiency indices of all methods decrease and there is no significant difference in the efficiency indices.

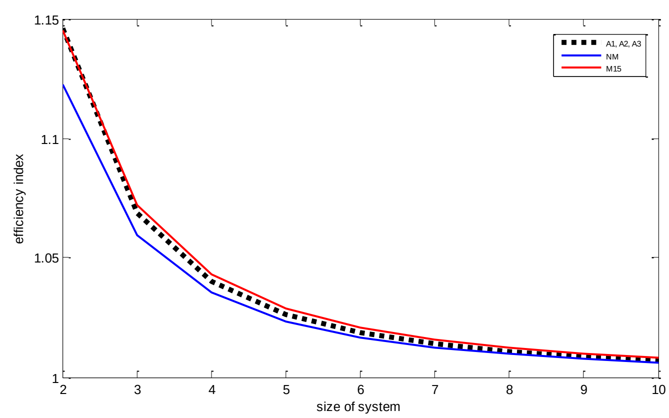

Fig. 1. Efficiency index comparison for different size systems

3. Numerical Setup, Results and Discussion

In this section, we use the proposed algorithms A1, A2 and A3 to solve some systems of nonlinear equations, including complicated case studies, as included here. The numerical test conditions are also explained. Finally, the results are discussed for all examples to demonstrate the numerical performance, accuracy and efficiency of proposed methods.

Example 1. A nonlinear model for the distance "r" of satellite "Wind" launched by NASA from earth discussed in [27] is given as.

$$G\frac{M_S m}{r^2}=G\frac{M_e m}{(R-r)^2}+mr\omega^2$$

Where, $G=6.67\times 10^{-11}$, $M_S=1.98\times 10^{30}[kg]$, $M_e=5.98\times 10^{24}[kg]$, m=the mass of satellite $[kg]$, $R=1.49\times 10^{11}[m]$, $\omega=\frac{2\pi}{T}$, $T=3.15576\times 10^{7}$[s].

We begin numerical proceedure with initial guess

$r_0=2$, to hit the required solution as given using schemes: $r=147617750096.1504621902$.

Example 2. A system nonlinear equation from [7].

$$\begin{cases} x^3-3xy^2+a_1(2x^2+xy)+b_1y^2+c_1x+a_2y=0 \\ 3x^2y-y^3-a_1(4xy-y^2)+b_2x^2+c_2=0 \end{cases}$$

Where, a_1=25, b_1=1, c_1=2, a_2=3, b_2=4, c_2=5. Taking the initial guesses $x_0=[2, 12]^T$ to find the required solutions of form: x=[1.6359717996. . . , 13.8476653258. . .].

Example 3. The nonlinear system is taken from [5]:

$$\begin{cases} x_2x_3+x_4(x_2+x_3)=0 \\ x_1x_3+x_4(x_1+x_3)=0 \\ x_1x_2+x_4(x_1+x_2)=0 \\ x_1x_2+x_1x_3+x_2x_3-1=0 \end{cases}$$

The initial approximations are $x_0=[0.5,0.5,0.5, -0.2]^T$ and the solutions using numerical processes to attain are: x=[0.5773502692…, 0.5773502692…,

0.5773502692…, -0.2886751346…].

Example 4. A nonlinear system representing non-adiabatic stirred tank reactors taken from [2] is.

$$\begin{cases} (1-R)\left[\frac{D}{10(1+\beta_1)}-\phi_1\right]e^{\left(\frac{10\phi_1}{1+\frac{10\phi_1}{\gamma}}\right)}-\phi_1 \\ \phi_1-(1+\beta_1)\phi_2+(1-R)\left[\begin{matrix}\frac{D}{10}-\beta_1\phi_1 \\ -(1+\beta_2)\phi_2\end{matrix}\right]e^{\left(\frac{10\phi_2}{1+\frac{10\phi_2}{\gamma}}\right)} \end{cases}$$

in two unknowns ϕ_1 and ϕ_2. Where, the parameters R, γ, D, β_1, β_2 are 0.94, 1000, 22, 2, and 2, respectively.

The initial guesses are $x_0=[5, 5]^T$ and the solution set is x=[0.7206169356…, 0.2454153706…].

Example 5. The sixth order nonlinear system related to Neurophysiology application is taken from [1].

$$\begin{aligned} &x_1^2+x_3^2=1 \\ &x_2^2+x_4^2=1 \\ &x_5x_3^3+x_6x_4^3=c_1 \\ &x_5x_1^3+x_6x_2^3=c_2 \\ &x_5x_1x_3^2+x_6x_4^2x_2=c_3 \\ &x_5x_1^2x_3+x_6x_2^2x_4=c_4 \end{aligned}$$

Here, $c_i=0$, $i=1,2,3,4$.

The initial values of unknowns are taken as $\mathbf{x}_0=[0.1, 0.2, 0.3, 0.4, 0.5, 0.6]^T$ and the solutions obtained are **x**=[0.3162277660…, 0.4472135955…, 0.9486832980…, 0.8944271909…, 1.42E-12225, 5.42E-19336].

Example 6. A 10×10 system is related to combustion for a temperature of 3000^0C considered in [28]:

$$\begin{cases} x_2+2x_6+x_9+2x_{10}-10^{-5}=0, \\ x_3+x_8-3.10^{-5}=0, \\ x_1+x_3+2x_5+2x_8+x_9+x_{10}-5.10^{-5}=0, \\ x_4+2x_7-10^{-5}=0, \\ x_1^2-0.5140437.10^{-7}x_5=0, \\ 2x_2^2-0.1006932.10^{-6}x_6=0, \\ x_4^2-0.7816278.10^{-15}x_7=0, \\ x_1x_3-0.1496236.10^{-6}x_8=0, \\ x_1x_2-0.6914411.10^{-7}x_9=0, \\ x_1x_2^2-0.2089296.10^{-14}x_{10}=0, \end{cases}$$

$x_0=[0.1, 0.4, 0.2, 0.3, 0.1, 0.6, 0.7, 0.5, 0.1, 0.4]^T$ are the initial guesses, and the solutions to first few digits are

x=[0.00000014709013277155…,
0.00000022619636102493…,
0.00001512807633833404…,
0.00000000006251491477…,
0.00000042088848007963…,
0.00000101625122135197…,
0.00000499996874254262…,
0.00001487192366166596…,
0.00000053711729453530…,
0.00000360209195086792...]^T.

Example 7. The nonlinear system considered is an application of kinematic synthesis mechanism given in [8] for i= 1, 2, 3.

$$\begin{aligned} &[E_i(x_2\sin\psi_i-x_3)-F_i(x_2\sin\phi_i-x_3)]^2 \\ &+[F_i(1+x_2\cos\phi_i)-E_i(x_2\cos\psi_i-1)]^2 \\ &-\left[\begin{matrix}(1+x_2\cos\phi_i)(x_2\sin\psi_i-x_3)x_1 \\ -(x_2\sin\phi_i-x_3)(x_2\cos\psi_i-x_3)x_1\end{matrix}\right]^2=0 \end{aligned}$$

Where, $E_i=x_2(\cos\phi_i-\cos\phi_0)-x_2x_3(\sin\phi_i-\sin\phi_0)$

$-(x_2\sin\phi_i-x_3)x_1$

and, $F_i = -x_2\cos\psi_i - x_2x_3\sin\psi_i + x_2\cos\psi_0$
$+x_1x_3 + (x_3 - x_1)x_2\sin\psi_0$

The values of angles in radians are displayed in Table 1.

Table 1

Angular values of ψ_i and ϕ_i for Example 7

i	ψ_i	ϕ_i
0	1.3954170041747090114	1.7461756494150842271
1	1.7444828545735749268	2.0364691127919609051
2	2.0656234369405315689	2.2390977868265978920
3	2.4600678478912500533	2.4600678409809344550

We use x_0=[0.7, 0.7, 0.7]^T as initial values and the solutions found are x=[0.9051567633…, 0.6977417789…, 0.6508335928…].

Example 8. As an application, we have solved one boundary value problem of the form of nonlinear ordinary differential equation taken from [4] written as

$$y'' + y^3 = 0, \quad y(0) = 0, \quad y(1) = 1,$$

The domain [0,1] is divided as follows:

$$a_0 = 0 < a_1 < a_2 < \cdots < a_{m-1} < a_m = 1,$$

$a_{j+1} = a_j + i, \quad i = 1/m.$ Let:

$$y_0 = y(a_0) = 0, \qquad y_1 = y(a_1), \quad \ldots, \quad y_{m-1} = y(a_{m-1}),$$

$$y_m = y(a_m) = 1.$$

Now applying finite difference discretization on problem by using the following approximation

$$y''_k \cong \frac{y_{k-1} - 2y_k + y_{k+1}}{i^2}, \quad k = 1,2,3,\ldots,m-1,$$

We get *m-1* nonlinear equations in *m-1* unknowns:

$$y_{k-1} - 2y_k + y_{k+1} + i^2y_k^3, \quad k = 1,2,3,\ldots,m-1.$$

Particularly, we use $m = 5$ and solve 4 nonlinear equations. The solution is **x**= [0.21026611204934614470…, 0.42049503909366502639…, 0.62774995479438586769…, 0.82510977336343305744…]^T.

The numerical results by proposed A1, A2 and A3 methods are compared with NM and M15. All the numerical calculations are carried out by MATLAB R2013a installed in Intel(R) Core (TM) i3 hp laptop with RAM of 4GB and operating at a processing speed of 2.4GHz up to 12000 digits. We use the following stopping criterion to note all the results:

$$\|e_i\|_\infty \leq 10^{-299} \tag{35}$$

where, $\boldsymbol{e_i}$ is absolute error [29-30] obtained as:

$$\mathbf{e}_i = |\mathbf{x}_{i+1} - \mathbf{x}_i| \qquad i = 1,2,3,\ldots \tag{36}$$

The computational order of convergence (p) is computed to verify theoretical one by using formula [8]:

$$p \approx \frac{\ln\left(\frac{\|e_{i+1}\|_\infty}{\|e_i\|_\infty}\right)}{\ln\left(\frac{\|e_i\|_\infty}{\|e_{i-1}\|_\infty}\right)}, i = 2,3,\ldots \tag{37}$$

where, $\|.\|_\infty$ is the *L*-infinity norm which means that we have displayed the maximum error among all variables at the end of iteration when pre-specified tolerance is achieved. The total computational cost (*COC*) taken by a method in each problem to reach at the pre-specified stopping criterion (35) can be found by multiplying the number of iterations '*I*' with the total number of new functions and derivative used, say '*m*', by that method. The formula for *COC* is defined in (38).

$$COC = I \times m \tag{38}$$

4. Results and Discussion

We use the notations like $\mathbf{x_0}$ for initial approximations and div if the solution cannot be found or a method diverges. In Fig. 2, the required number of iterations to achieve the pre-specified error tolerance (35) is shown for examples 1-7 by all methods. All methods converged to the expected solutions in all cases, except M15 method which diverged in Examples 5 and 6. It appears from Fig. 2 that the number of iterations for the proposed methods is fewer than those by NM in all cases. All proposed methods use fewer number of iterations than the M15 method in example 3 and equal number of iterations in example 2. For examples 5 and 6, where M15 method diverges, the number of iterations required for the proposed methods are smallest of all. Due to different coefficients of error terms in equations (32)-(34) of the proposed methods A1-A3, as discussed before, the number of iterations and other indicators for the proposed methods are warranted to be same always, except the order of convergence. One or more of the proposed methods are best of all in cases where all discussed methods are applicable. For instance, in example 1 and 4, A2 is best of all, whereas in examples 6 and 7 A1 is best of all.

The observed computational orders of convergence, using (37), by all methods are displayed in Fig. 3. The theoretical orders of convergence are also mentioned against names of all methods in Fig. 3 for ready reference. In examples 1-5, the theoretical orders of convergence for the NM and the proposed A1-A3 methods have been verified, and are numerically approaching to atleast 2 and 20, respectively. While the M15 method diverges in example 5 and 6, it also compromises on the order of convergence in examples 2 and 3 where the observed orders are less than the expected – 15. However, the theoretical order of convergence for M15 has been verified in examples 1 and 4. The examples 6 and 7 are complicated ones among all other examples. It is know that most of the methods, when go outside the asymptotic error regimes, which can be due to initial guess or irregularities in the nonlinear functions, tend to compromise the theoretical properties and converge linearly or super linearly inspite of being higher order accurate in the regular problems. This is the case in examples 6 and 7, where all methods compromise on the theoretical order of convergence. The order of convergence of NM, the proposed A1-A3 and the M15 are approaching 1 in examples 6 and 7; the M15 method in fact diverges in example 6. However, even after compromising on order of convergence, the proposed methods are better than others on the basis of fewer number of iterations, see Fig. 2.

The normed absolute error distributions for all methods in examples 1-7 are shown in Figs. 4-11 versus number of iterations. Due to limitations of MATLAB for axes marks and display in the range (10^{-300}, 10^{300}), errors fewer than 10^{-300} could not be displayed. It is clear from Figs. 4-11 that the proposed twentieth order methods exhibit rapid decrease in errors as the number of iterations advance. The results concerning the number of iterations from Fig. 2 can be verified in detail from Figs. 4-11. If the error curve for a method disappears earlier than others, then it means that the error has decreases sufficiently, i.e. had become lower than 10^{-300}. It is evident from Fig. 4-11 that one or more of the proposed methods achieve the specified error tolerance earlier than NM and M15 methods, and in cases where tie for number of iterations exists, the errors for the proposed methods are lower in magnitude.

Since higher order iterative methods use more information per iteration to find the solution, it is important to compare the performance of methods in view of computational burden to reach a pre-specified error tolerance. There are two ways in which we have calculated the computational overhead of the discussed methods for examples 1-7. The COC, as defined in (38) counts only the total number of evaluations of the function and its derivatives – the partial derivatives in the case of systems of equations – to achieve specified error. However, the complexity due to arithmetic, vector and matrix operations in the formula of a method are not counted in COC. For this reason, the execution time in the form of CPU time (in seconds) required to achieve the specified error is preferred. The CPU time, however doesn't show the breakup of total time into operations, evaluations etc, but it is considered more comprehensive than the traditional COC. We have computed both COC and CPU time to compare the performance of discussed methods. After the usual theoretical preference due to order of convergence of the methods, the next parameter can be the COC and CPU times under similar conditions. The NM used only one functional and one derivative evaluation per iteration, so it's COC for scalar nonlinear equation per iteration is 2, whereas for an n^{th} order nonlinear system the COC for NM is $n + n^2$ (n functional and n^2 first order partial derivatives). Similarly, for the M15 method the COC for scalar nonlinear equation is 7, whereas for n^{th} order system the COC per iteration for M15 method is $4n + 3n^2$. For the proposed methods A1-A3, the COC per iteration for an n^{th} order nonlinear system is $3n + 4n^2$, and the COC for scalar nonlinear equations is same as that of M15 per iteration. The comparison of efficiency indices has been discussed in Fig. 1.

The computational cost, COC, for examples 1-5, 7-8 and examples 6 are shown in Figs. 12 and 13, respectively. The CPU times (in seconds) are shown in Figs. 14 for examples 1-8. Fig. 12 shows that the COC for the proposed A2 method is smaller than all others in examples 1 and 4, whereas the execution time in Fig. 14 for example 1 and example 4 for M15 and A2 methods are closer and lower than all other methods. The COC for the NM is smallest of all in example 2 (Fig. 12) and the CPU times are methods are almost same (Fig. 14). Since the theoretical preference in terms of order of convergence of the proposed methods A1-A3 and M15 over the NM is well developed, the errors at the stopping iterations of the A1, A2, A3 and M15 methods, which are 5.26E-5469, 2.06E-5634, 6.44E-5582 and 6.99E-995 are respectively 15.99%, 16.42%, 16.27% and 2.90% smaller than the same for NM, which is 6.88E-343. The COC for the M15 method is highest of all and for the NM is smallest of all in example 3 (Fig. 12, where as in terms of CPU time from Fig. 14 it is clear that all proposed methods take lesser time, particularly A2 and A3 methods are quicker than others. Fig. 12 further show that, for example 5, the NM uses lesser evaluations than other applicable methods (M15 diverges here), whereas the CPU times of the proposed methods and the NM, via Fig. 14, are closer and approaching 43 seconds. Similarly, in example 6, the M15 example fails, the NM method is better in COC (Fig. 13), and the proposed method A1, via Fig. 14, is best of all applicable methods in execution time. Fig. 14 further shows that the CPU time of A3 method is also smaller than that of NM method. For examples 6 and 7, as the theoretical performance of all methods from view-point of order of convergence could not be observed, however, all methods managed to converge to the solution with order of convergence closer to 1 only. This is the reason that in the expected performance in form of COC and CPU times for higher order methods could not be guaranteed. The M15 method in examples 1-4 was applicable and resulted comparable performance with other methods and in examples 5-6 diverged. Surprisingly, in example 7, the M15 method takes smallest COC (Fig. 12) and CPU time (Fig. 14) as compared to other methods with a tie in number of iterations with A1 method (Fig. 2).The main advantage of the methods A1-A3 in the proposed family is their twentieth order convergence against lower order methods. The challenge of developing new, efficient and higher-order iterative methods has been addressed as some existing methods failed to obtain the solutions in complex problems, whereas the proposed methods were shown to be applicable in similar situations for the complex problems describing dynamics of complex systems in Engineering.

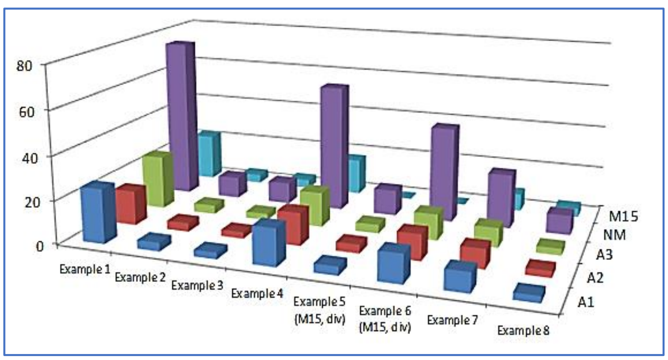

Fig. 2. Comparison of number of iterations to achieve specified error tolerance in all methods for examples 1-8

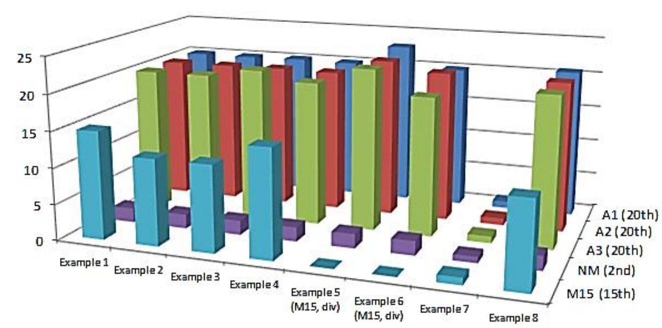

Fig. 3. Observed orders of convergence to achieve specified error tolerance in all methods for examples 1-8

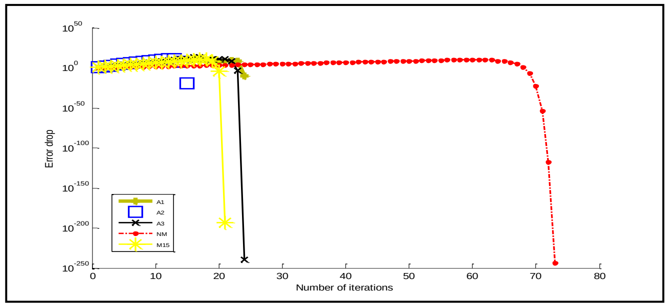

Fig. 4. Error drops for example 1

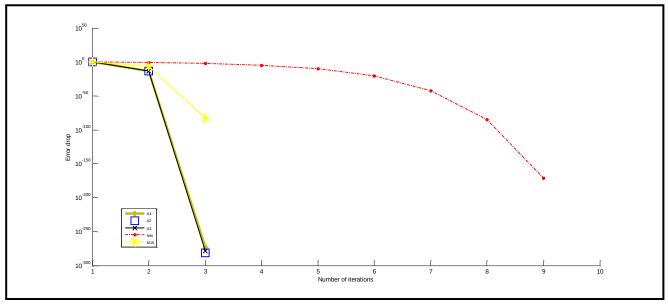

Fig. 5. Error drops for example 2

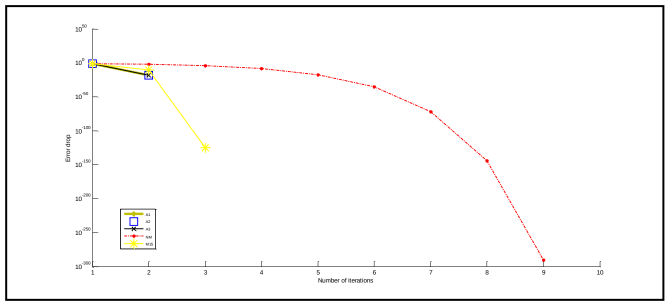

Fig. 6. Error drops for example 3

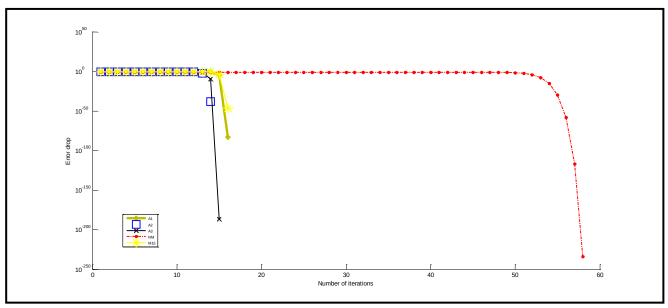

Fig. 7. Error drops for example 4

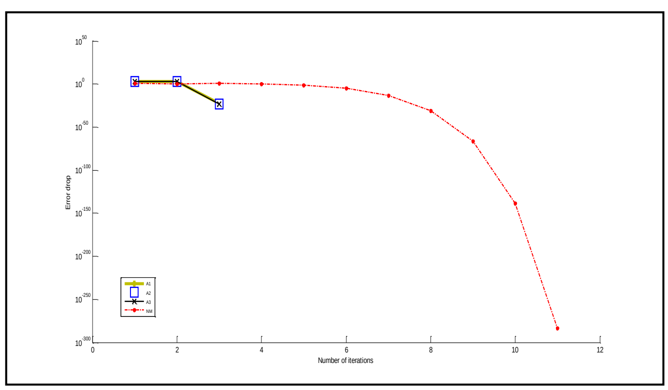

Fig. 8. Error drops for example 5

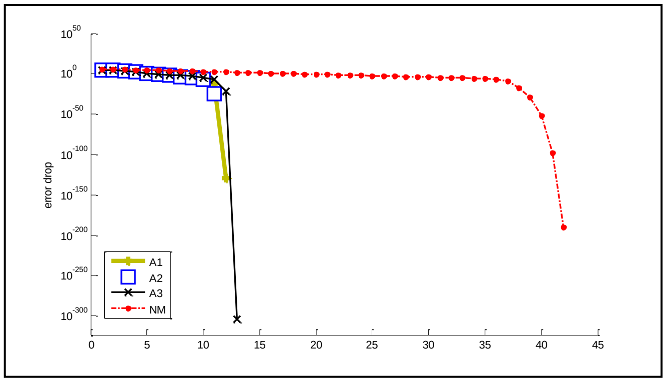

Fig. 9. Error drops for Example 6

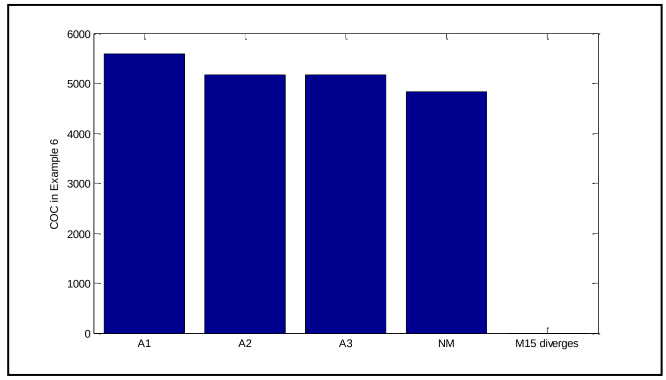

Fig. 13. COC in Example 6

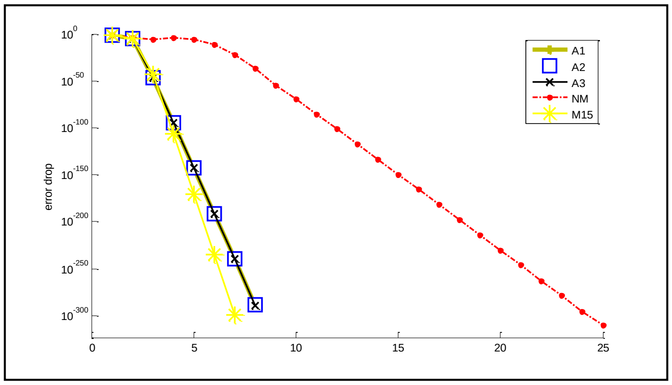

Fig. 10. Error drops for Example 7

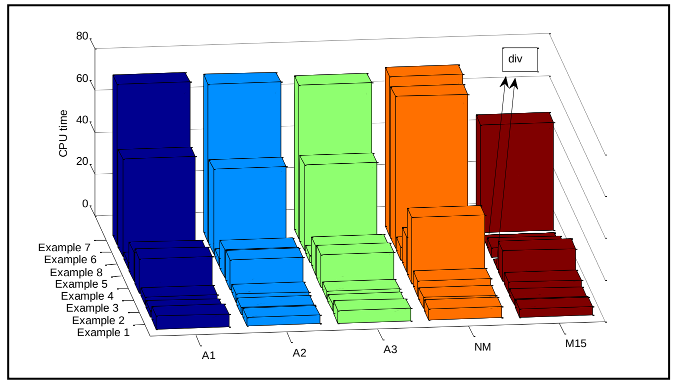

Fig. 14. CPU times in Examples 1-8

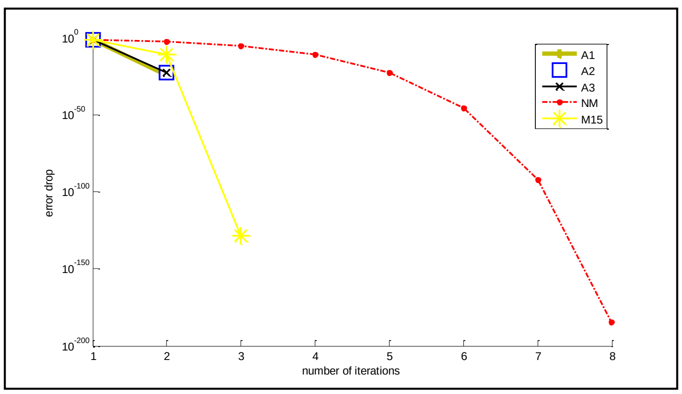

Fig. 11. Error drops for example 8

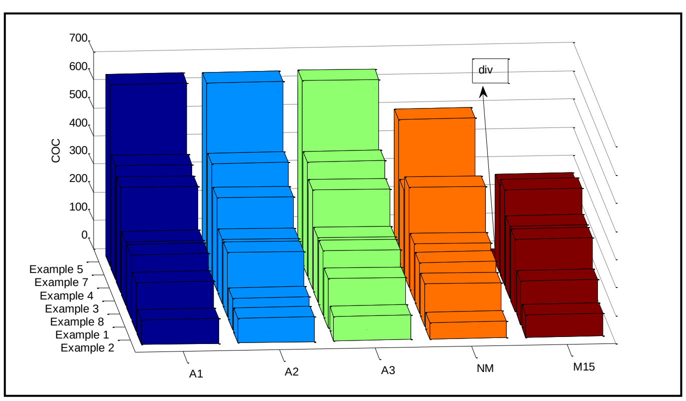

Fig. 12. COC in examples 1-5 and 7-8

4. Conclusion

In this research paper, an efficient family of iterative methods with twentieth order convergence to solve complex models described in terms of nonlinear equations and systems. We have proved the order of convergence theoretically and verified numerically by using the methods to solve a nonlinear scalar equation and some complex nonlinear systems from engineering applications. The performance of the proposed methods in comparison with other methods used in this work under similar conditions shows that the convergence of proposed methods is faster as compared to other well-known existing methods. For testing the performance of propped family and other existing methods, error distributions, number of iterations, computational order of convergence, number of evaluations and execution times needed to achieve pre-specified tolerance are examined on different case study real-world problems. The proposed family of methods results in higher degree of accuracy in fewer number of iteration than other discussed methods.

5. Acknowledgement

The authors would like to thank Mehran University of Engineering and Technology, Jamshoro for providing support to perform this research work conveniently. The efforts of anonymous reviewers for suggesting several improvements in the present version of paper are highly appreciated.

6. References

[1] C. Grosan, and A. Abraham, "A new approach for solving nonlinear equations systems", IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans, Vol. 38, no. 3, pp. 698-714, 2008.

[2] F. Awawdeh, "On new iterative method for solving systems of nonlinear equations", Numerical Algorithms, Vol. 54, no. 3, pp. 395-409, 2010.

[3] A. Srivastava, "An iterative method with fifteenth-order convergence to solve systems of nonlinear equations", Computational Mathematics and Modeling, Vol. 27, no. 4, pp. 497-510, 2016.

[4] J. R. Sharma, J. R. Rajni, and N. Kalra, "A novel family of composite Newton–Traub methods for solving systems of nonlinear equations", Applied Mathematics and Computation, Vol. 269, pp. 520-535, 2015.

[5] Z. Malik, et al., "Higher order multi-step iterative method for computing the numerical solution of systems of nonlinear equations: Application to nonlinear PDEs and ODEs", Applied Mathematics and Computation, Vol. 269, pp. 972-987, 2015.

[6] A. Cordero, E. Gómez, and J. R. Torregrosa, "Efficient high-order iterative methods for solving nonlinear systems and their application on heat conduction problems", Complexity, 2017.

[7] X. Xiao, and H. Yin, "A simple and efficient method with high order convergence for solving systems of nonlinear equations", Computers and Mathematics with Applications, Vol. 69, no. 10, pp. 1220-1231, 2015.

[8] M. Waseem, M. A. Noor, and K. I. Noor, "Efficient method for solving a system of nonlinear equations", Applied Mathematics and Computation, Vol. 275, no. 2016, pp. 134-146, 2016.

[9] D. K. R. Babajee "On the kung-Traub conjecture for iterative methods for solving quadratic equations", Algorithms, Vol. 9, no. 1, 2015.

[10] J. R. Sharma, and P. Gupta. "An efficient fifth order method for solving systems of nonlinear equations", Computers and Mathematics with Applications, Vol. 67, no. 3, pp. 591-601, 2014.

[11] J. L. Hueso, M. Eulalia, and T. Carles, "Derivative free iterative methods for nonlinear systems", Applied Mathematics and Computation, Vol. 259, pp. 955-966, 2015.

[12] G-S, Miquel, G. Àngela, and M. Noguera. "On the computational efficiency index and some iterative methods for solving systems of nonlinear equations", Journal of Computational and Applied Mathematics, Vol. 236, no. 6, pp. 1259-1266, 2011.

[13] J. F. Traub, "Iterative methods for the solution of equations", American Mathematical Soc. Vol. 312, 1982.

[14] M. A. Noor et al., "An iterative method with cubic convergence for nonlinear equations", Applied Mathematics and Computation, Vol. 183, no. 2, pp. 1249-1255, 2006.

[15] M. A. Noor et al., "Some new iterative methods for nonlinear equations", Mathematical Problems in Engineering, 2010.

[16] H., Zhongyong, L. Guocai, and L. Tian, "An iterative method with ninth-order convergence for solving nonlinear equations", International Journal of Contemporary Mathematical Sciences, Vol. 6, no. 1, pp. 17-23, 2011.

[17] M. Raza, "Eleventh-order convergent iterative method for solving nonlinear equations", Int. J. Appl. Math, Vol. 25, pp. 365-371, 2012.

[18] A. Cordero, M. Eulalia, and J. R. Torregrosa, "Iterative methods of order four and five for systems of nonlinear equations", Journal of Computational and Applied Mathematics, Vol. 231, no. 2, pp. 541-551, 2009.

[19] M. A. Noor, and M. Wasim, "Some iterative methods for solving a system of nonlinear

equations", Computers and Mathematics with Applications, Vol. 57, no. 1, pp. 101-106, 2009.

[20] M. A. Noor, "New classes of iterative methods for nonlinear equations", Applied Mathematics and Computation, Vol. 191, no. 1, pp. 128-131, 2007.

[21] M. Rafiullah, "A fifth-order iterative method for solving nonlinear equations", Numerical Analysis and Applications, Vol. 4, no. 3, pp. 239, 2011.

[22] M. M. Shaikh, S-u-R. Massan, and A. I. Wagan, “A new explicit approximation to Colebrook’s friction factor in rough pipes under highly turbulent cases”, International Journal of Heat and Mass Transfer, Vol. 88, pp. 538-543, 2015.

[23] M. M. Shaikh, S-u-R. Massan, and A. I. Wagan, “A sixteen decimal places’ accurate Darcy friction factor database using non-linear Colebrook’s equation with a million nodes: a way forward to the soft computing techniques”, Data in brief, Vol. 27, no. Dec. 2019, 104733, 2019.

[24] A. Cordero, J. L. Hueso, E. Martínez, and J. R. Torregrosa, “A modified Newton-Jarratt’s composition”, Numerical Algorithms, Vol. 55, no. 1, pp.87-99, 2010.

[25] R. Sharma, and B. Ashu, "An optimal fourth order iterative method for solving nonlinear equations and its dynamics", Journal of Complex Analysis, Vol. 2015, 2015, 259167.

[26] J. M. Ortega, and C. R. Werner, “Iterative solution of nonlinear equations in several variables”, SIAM, Vol. 30, 1970.

[27] W. Y. Yang et al., ”Applied numerical methods using MATLAB”, John Wiley and Sons, 2020.

[28] H. A. Abro, M. M. Shaikh, "A new time-efficient and convergent nonlinear solver", Applied Mathematics and Computation, Vol. 355, no. 2019, pp. 516-536, 2019.

[29] A. A. Mastoi, M. M. Shaikh, and A. W. Shaikh, “A new third-order derivative-based iterative method for nonlinear equations”, J. Mech. Cont. and Math. Sci, Vol. 15, no. 10, pp.110-123, 2020.

[30] S. Umar, M. M. Shaikh, and A. W. Shaikh, “A new quadrature-based iterative method for scalar nonlinear equations”, J. Mech. Cont. and Math. Sci, Vol. 15, no. 10, pp.79-93, 2020.

