

Research Article

A New High-Order and Efficient Family of Iterative Techniques for Nonlinear Models

Ramandeep Behl ¹ and Eulalia Martínez ²

¹Department of Mathematics, King Abdulaziz University, Jeddah 21589, Saudi Arabia

²Instituto Universitario de Matemática Multidisciplinar, Universitat Politècnica de València, València, Spain

Correspondence should be addressed to Eulalia Martínez; eumarti@mat.upv.es

Received 23 September 2019; Accepted 7 January 2020; Published 30 January 2020

Academic Editor: Carlos F. Aguilar-Ibáñez

Copyright © 2020 Ramandeep Behl and Eulalia Martínez. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

In this paper, we want to construct a new high-order and efficient iterative technique for solving a system of nonlinear equations. For this purpose, we extend the earlier scalar scheme [16] to a system of nonlinear equations preserving the same convergence order. Moreover, by adding one more additional step, we obtain minimum 5th-order convergence for every value of a free parameter, $\theta \in \mathbb{R}$, and for $\theta = -1$, the method reaches maximum 6-order convergence. We present an extensive convergence analysis of our scheme. The analytical discussion of the work is upheld by performing numerical experiments on some applied science problems and a large system of nonlinear equations. Furthermore, numerical results demonstrate the validity and reliability of the suggested methods.

1. Introduction

Solution of nonlinear equations of the following form:

$$F(x) = 0, \quad (1)$$

where $F: D \subseteq \mathbb{R}^n \rightarrow \mathbb{R}^n$, is a more challenging and broader problem as compared to the scalar equations. Several issues like neurophysiology, kinematics syntheses, transport theory, chemical equilibrium, reactor and steering combustion, and economics modeling problems can be phrased in the terms of (1) and details can be found in [1–9].

There are different ways to construct iterative schemes for nonlinear systems. We center in the natural way methods constructed for scalar equations and then extend to the multidimensional case. Some researchers like Cordero et al. [10], Abad et al. [11], Cordero et al. [12], and Wang and Zhang [13] adopted this way to construct schemes for nonlinear systems. There is no doubt that some scalar schemes can be extended to a system of nonlinear equations with the help of some transformations or substitutions. But, the most important and crucial task is to retain the same convergence order and lower computational cost.

Therefore, researchers also tried some other approaches and procedures in order to construct new and higher-order iterative methods for systems of nonlinear equations. Recently, Sharma et al. [14] proposed fourth-order and six-order iterative methods based on weighted-Newton iteration. Very recently, Artidiello et al. [15] provided fourth-order methods based on the weight function approach. Some researchers have also used the approaches like quadrature formulae, Adomian polynomial, divided difference approach for constructing iterative schemes to solve nonlinear systems. For the details of the other approaches, one can refer some standard text books [16, 17, 18].

In this paper, our main objective is to construct a new high-order family of iterative methods for a system of nonlinear equations. The development of our scheme is based on the scalar scheme proposed by Kou et al. [19]. First of all, we extend their scheme for a system of nonlinear equations retaining the same convergence order. Then, we increase the convergence order up to 6 by adding one additional substep to the same scheme. The efficiency of the suggested iterative techniques is demonstrated on several applied science problems and academic numerical examples.

We found that our proposed methods perform better than the existing ones in terms of residual error, difference between two consecutive iterations, and asymptotic error constants.

2. Higher-Order Scheme for Multidimensional Case

Here, we consider a higher-order scheme for simple zeros on univariate function proposed by Kou et al. [19], which is defined as follows:

$$\begin{aligned} y_i &= x_i - \frac{f(x_i)}{f'(x_i)}, \\ x_{i+1} &= x_i - \theta \frac{f(x_i) + f(y_i)}{f'(x_i)} \\ &\quad - (1 - \theta) \frac{f(x_i)^2}{f'(x_i)(f(x_i) - f(y_i))}, \end{aligned} \quad (2)$$

where $\theta \in \mathbb{R}$ is a free disposable parameter. Expression (2) has at least cubic convergence for every θ except $\theta \neq -1$. It has maximum fourth-order convergence for $\theta = -1$. It is a linear combination of well-known Potra-Pták and Newton-Steffensen methods. More details can be found in Kou et al.'s study [19]. We can easily obtain Potra-Pták [20] and Newton-Steffensen [21], for $\theta = 1$ and $\theta = 0$, respectively, in scheme (2). We can rewrite scheme (2) in the following way:

$$\begin{aligned} y_i &= x_i - \frac{f(x_i)}{f'(x_i)}, \\ x_{i+1} &= x_i - \theta \frac{f(x_i) + f(y_i)}{f'(x_i)} - (1 - \theta) \frac{1}{[y_i, x_i; f]} \frac{f(x_i)}{f'(x_i)^2}, \end{aligned} \quad (3)$$

or

$$\begin{aligned} y_i &= x_i - \frac{f(x_i)}{f'(x_i)}, \\ x_{i+1} &= x_i - \theta \frac{f(x_i) + f(y_i)}{f'(x_i)} - (1 - \theta) \frac{f(x_i)}{[y_i, x_i; f]}, \end{aligned} \quad (4)$$

where $[y_i, x_i; f]$ is the well-known divided difference of first order. But, scheme (2) does not work for a system of nonlinear equations. So, our foremost aim is to extend this scheme for a system of nonlinear equations and retain the same convergence order. Therefore, we rewrite expression (4) in the following way:

$$\begin{aligned} y^{(j)} &= x^{(j)} - F'(x^{(j)})^{-1} F(x^{(j)}), \\ x^{(j+1)} &= x^{(j)} - \theta F'(x^{(j)})^{-1} (F(x^{(j)}) + F(y^{(j)})) \\ &\quad - (1 - \theta) [y^{(j)}, x^{(j)}; F]^{-1} F(x^{(j)}), \end{aligned} \quad (5)$$

where θ is a free disposable parameter and $[y^{(j)}, x^{(j)}; F]$ is a finite difference of order one.

Now, by adding one extra substep, we have following higher-order scheme:

$$\begin{aligned} y^{(j)} &= x^{(j)} - F'(x^{(j)})^{-1} F(x^{(j)}), \\ z^{(j)} &= x^{(j)} - \theta F'(x^{(j)})^{-1} (F(x^{(j)}) + F(y^{(j)})) \\ &\quad - (1 - \theta) [y^{(j)}, x^{(j)}; F]^{-1} F(x^{(j)}), \\ x^{(j+1)} &= z^{(j)} - [y^{(j)}, z^{(j)}; F]^{-1} F(z^{(j)}). \end{aligned} \quad (6)$$

In Theorem 1, we demonstrate that the minimum convergence orders of methods (5) and (6) are three and five, respectively. In the proof of this result, we use the tools and procedure introduced in [10] which we recall briefly.

Let $F: D \subseteq \mathbb{R}^n \rightarrow \mathbb{R}^n$ be sufficiently differentiable in D . The q th derivative of F at $u \in \mathbb{R}^n$, $q \geq 1$, is the q -linear function $F^{(q)}(u): \mathbb{R}^n \times \dots \times \mathbb{R}^n \rightarrow \mathbb{R}^n$ such that $F^{(q)}(u)(v_1, \dots, v_q) \in \mathbb{R}^n$. It is easy to observe that

- (1) $F^{(q)}(u)(v_1, \dots, v_{q-1}, \cdot) \in \mathcal{L}(\mathbb{R}^n)$.
- (2) $F^{(q)}(u)(v_{\sigma(1)}, \dots, v_{\sigma(q)}) = F^{(q)}(u)(v_1, \dots, v_q)$, for all permutation σ of $\{1, 2, \dots, q\}$.

From the above properties, we can use the following notation:

- (a) $F^{(q)}(u)(v_1, \dots, v_q) = F^{(q)}(u)v_1 \dots v_q$
- (b) $F^{(q)}(u)v^{q-1}F^{(p)}v^p = F^{(q)}(u)F^{(p)}(u)v^{q+p-1}$

On the other hand, for $\xi + h \in \mathbb{R}^n$ lying in a neighborhood of a solution ξ of $F(x) = 0$, we can apply Taylor's expansion, and assuming that the Jacobian matrix $F'(\xi)$ is nonsingular, we have

$$F(\xi + h) = F'(\xi) \left[h + \sum_{q=2}^{p-1} C_q h^q \right] + O(h^p), \quad (7)$$

where $C_q = (1/q!) [F'(\xi)]^{-1} F^{(q)}(\xi)$, $q \geq 2$. We observe that $C_q h^q \in \mathbb{R}^n$ since $F^{(q)}(\xi) \in \mathcal{L}(\mathbb{R}^n \times \dots \times \mathbb{R}^n, \mathbb{R}^n)$ and $[F'(\xi)]^{-1} \in \mathcal{L}(\mathbb{R}^n)$.

In addition, we can express F' as

$$F'(\xi + h) = F'(\xi) \left[I + \sum_{q=2}^{p-2} q C_q h^{q-1} \right] + O(h^{p-1}), \quad (8)$$

where I is the identity matrix. Therefore, $q C_q h^{q-1} \in \mathcal{L}(\mathbb{R}^n)$. From (8), we obtain

$$[F'(\xi + h)]^{-1} = [I + X_2 h + X_3 h^2 + X_4 h^3 + \dots] [F'(\xi)]^{-1} + O(h^p), \quad (9)$$

where

$$\begin{aligned} X_2 &= -2C_2, \\ X_3 &= 4C_2^2 - 3C_3, \\ X_4 &= -8C_2^3 + 6C_2C_3 + 6C_3C_2 - 4C_4, \\ &\vdots \end{aligned} \quad (10)$$

We denote $e^{(j)} = x^{(j)} - \xi$ as the error in the j th-iteration in the multidimensional case. The equation $e^{(j+1)} = M e^{(j)p}$

$+O(e^{(j)p+1})$, where M is a p -linear function, $M \in \mathcal{L}(\mathbb{R}^n \times \dots \times \mathbb{R}^n, \mathbb{R}^n)$, is called the error equation and p is the order of convergence. Let us observe that $e^{(j)p}$ is $(e^{(j)}, e^{(j)}, \dots, e^{(j)})$.

Theorem 1. *Let $F: D \subseteq \mathbb{R}^n \rightarrow \mathbb{R}^n$ be a sufficiently differentiable function in an open neighborhood D of its zero ξ . Let us consider that $F'(x)$ is continuous and nonsingular in the neighborhood of ξ . In addition, we assume the initial guess $x^{(0)}$ is close enough to ξ for the guaranteed convergence. Then, the iterative schemes (5) and (6) have minimum third and fifth order of convergence, respectively.*

Proof. Let $e^{(j)} = x^{(j)} - \xi$ be the error of the j th iteration. Developing $F(x^{(j)})$ and $F'(x^{(j)})$ in a neighborhood of ξ , we write

$$F(x^{(j)}) = F'(\xi) \left[e^{(j)} + C_2 e^{(j)2} + C_3 e^{(j)3} + C_4 e^{(j)4} + C_5 e^{(j)5} + C_6 e^{(j)6} \right] + O(e^{(j)7}) \quad (11)$$

and

$$F'(x^{(j)}) = F'(\xi) \left[I + 2C_2 e^{(j)} + 3C_3 e^{(j)2} + 4C_4 e^{(j)3} + 5C_5 e^{(j)4} + 6C_6 e^{(j)5} \right] + O(e^{(j)6}), \quad (12)$$

where I is the identity matrix of size $n \times n$ and $C_i = 1/k! F^{(k)}(\xi)^{-1} F^{(k)}(\xi)$, $k \geq 2$.

With the help of expression (12), we have

$$F'(x^{(j)})^{-1} = \left[I - 2C_2 e^{(j)} + \Omega_0 e^{(j)2} + \Omega_1 e^{(j)3} + \Omega_2 e^{(j)4} + \Omega_3 e^{(j)5} + \Omega_4 e^{(j)6} + O(e^{(j)7}) \right] F'(\xi)^{-1}, \quad (13)$$

where $\Omega_i = \Omega_i(C_2, C_3, \dots, C_6)$, given by

$$\begin{aligned} \Omega_0 &= 4C_2^2 - 3C_3, \\ \Omega_1 &= -(8C_2^3 - 6C_2C_3 - 6C_3C_2 + 4C_4), \\ \Omega_2 &= 8C_2C_4 + 9C_3^2 + 8C_4C_2 - 12C_2^2C_3 \\ &\quad - 12C_2C_3C_2 - 12C_3C_2^2 + 16C_2^4 - 5C_5, \\ \Omega_3 &= 10C_2C_5 + 12C_3C_4 + 12C_4C_3 + 10C_5C_2 \\ &\quad - 16C_2^2C_4 - 18C_2C_3^2 - 16C_2C_4C_2 \\ &\quad - 18C_3C_2C_3 - 18C_3^2C_2 \\ &\quad - 16C_4C_2^2 + 24C_2^3C_3 + 24C_2^2C_3C_2 \\ &\quad + 24C_2C_3C_2^2 + 24C_3C_2^3 - 32C_2^5 - 6C_6, \\ &\quad \vdots \end{aligned} \quad (14)$$

From expressions (11) and (2), we yield

$$F'(x^{(j)})^{-1} F(x^{(j)}) = e^{(j)} + \Lambda_0 e^{(j)2} + \Lambda_1 e^{(j)3} + \Lambda_2 e^{(j)4} + \Lambda_3 e^{(j)5} + \Lambda_4 e^{(j)6} + O(e^{(j)7}), \quad (15)$$

where $\Lambda_j = \Lambda_j(C_2, C_3, \dots, C_6)$, i.e.,

$$\begin{aligned} \Lambda_0 &= -C_2, \\ \Lambda_1 &= 2C_2^2 - 2C_3, \\ \Lambda_2 &= -(4C_2^3 - 4C_2C_3 - 3C_3C_2 + 3C_4), \\ \Lambda_3 &= 6C_2C_4 + 6C_3^2 + 4C_4C_2 - 8C_2^2C_3 - 6C_2C_3C_2 - 6C_3C_2^2 \\ &\quad + 8C_2^4 - 4C_5, \\ \Lambda_4 &= 8C_2C_5 + 9C_3C_4 + 8C_4C_3 + 5C_5C_2 - 12C_2^2C_4 - 12C_2C_3^2 \\ &\quad - 8C_2C_4C_2 - 12C_3C_2C_3 - 9C_3^2C_2 - 8C_4C_2^2 + 16C_2^3C_3 \\ &\quad + 12C_2^2C_3C_2 + 12C_2C_3C_2^2 \\ &\quad + 12C_3C_2^3 - 16C_2^5 - 5C_6, \\ &\quad \vdots \end{aligned} \quad (16)$$

By inserting expression (15) in the first substep of (5), we obtain

$$y^{(j)} - \xi = -\Lambda_0 e^{(j)2} - \Lambda_1 e^{(j)3} - \Lambda_2 e^{(j)4} - \Lambda_3 e^{(j)5} - \Lambda_4 e^{(j)6} + O(e^{(j)7}), \quad (17)$$

which further produces

$$F(y^{(j)}) = F'(\xi) \left[-\Lambda_0 e^{(j)2} - \Lambda_1 e^{(j)3} + (C_2\Lambda_0^2 - \Lambda_2) e^{(j)4} + (2C_2\Lambda_0\Lambda_1 - \Lambda_3) e^{(j)5} - (C_3\Lambda_0^3 - C_2(\Lambda_1^2 + 2\Lambda_0\Lambda_2) + \Lambda_4) e^{(j)6} + O(e^{(j)7}) \right], \quad (18)$$

and

$$\begin{aligned} [y^{(j)}, x^{(j)}; F]^{-1} &= 1 - C_2 e^{(j)} + (C_2\Lambda_0 + C_2^2 - C_3) e^{(j)2} \\ &\quad + (-2C_2^2\Lambda_0 + C_2(2C_3 + \Lambda_1) \\ &\quad + C_3\Lambda_0 - C_2^3 - C_4) e^{(j)3} \\ &\quad + (3C_2^3\Lambda_0 + C_2^2(-3C_3 + \Lambda_0^2 - 2\Lambda_1) \\ &\quad + C_2(-4C_3\Lambda_0 + 2C_4 + \Lambda_2) + C_4\Lambda_0 \\ &\quad + C_3(\Lambda_1 - \Lambda_0^2) + C_2^4 + C_3^2 - C_5) e^{(j)4} + O(e^{(j)5}). \end{aligned} \quad (19)$$

By using expressions (11)–(19) in scheme (5), we have

$$e^{(j+1)} = \Gamma_0 e^{(j)2} + \Gamma_1 e^{(j)3} + \Gamma_2 e^{(j)4} + \Gamma_3 e^{(j)5} + \Gamma_4 e^{(j)6} + O(e^{(j)7}), \quad (20)$$

where $\Gamma_i = \Gamma_i(\Lambda_0, \Lambda_1, \dots, \Lambda_4, C_2, C_3, \dots, C_6, \theta)$, i.e.,

$$\begin{aligned} \Gamma_0 &= \theta(C_2 + \Lambda_0), \\ \Gamma_1 &= -C_2(\theta + 1)\Lambda_0 + \theta(2C_3 + \Lambda_1) - 2C_2^2\theta, \\ &\quad \vdots \end{aligned} \quad (21)$$

By using the values of $\Lambda_0 = -C_2$ and $\Lambda_1 = 2C_2^2 - 2C_3$ from (15) in Γ_0 and Γ_1 , we have

$$\begin{aligned}\Gamma_0 &= 0, \\ \Gamma_1 &= (\theta + 1)C_2^2,\end{aligned}\quad (22)$$

and adopting (22) in (21), we yield

$$\begin{aligned}e^{(j+1)} &= z^{(j)} - \xi \\ &= (\theta + 1)C_2^2 e^{(j)3} + \Gamma_2 e^{(j)4} + \Gamma_3 e^{(j)5} + \Gamma_4 e^{(j)6} + O(e^{(j)7}).\end{aligned}\quad (23)$$

From expression (23), it clear that scheme (5) has the least third-order convergence for every θ .

Now, expanding the $F(z^{(j)})$ in a neighborhood of ξ , we have

$$\begin{aligned}F(z^{(j)}) &= F'(\xi) [(\theta + 1)C_2^2 e^{(j)3} + \Gamma_2 e^{(j)4} + \Gamma_3 e^{(j)5} \\ &\quad + (C_2^5 (\theta + 1)^2 + \Gamma_4) e^{(j)6} + O(e^{(j)7})],\end{aligned}\quad (24)$$

which further produces, with the help of expression (18),

$$\begin{aligned}[z^{(j)}, y^{(j)}; F]^{-1} &= 1 - C_2^2 e^{(j)2} - ((\theta - 1)C_2^3 + 2C_3 C_2) e^{(j)3} \\ &\quad + (C_2^4 - C_3 C_2^2 - \Gamma_2 C_2 + \Lambda_2 C_2) e^{(j)4} \\ &\quad + O(e^{(j)5}).\end{aligned}\quad (25)$$

By using equations (24) and (25) in the final substep of (6), we obtain

$$e^{(j+1)} = (\theta + 1)C_2^4 e^{(j)5} + \bar{\Gamma}_4 e^{(j)6} + O(e^{(j)7}). \quad (26) \quad \square$$

It is clear from expression (26) that we have minimum fifth-order convergence for every $\theta \in \mathbb{R} - \{-1\}$. In addition, we have maximum sixth order of convergence for $\theta = -1$. Hence, schemes (5) and (6) have minimum third and fifth order convergence, respectively, for every $\theta \neq -1 \in \mathbb{R}$.

3. Numerical Experiments

Here, we checked the efficiency and effectiveness of our methods on 4 real life problems. Some of them have been taken from the paper of R. Behl et al. (see [22]). Additionally, we also consider a large system of equations of 150×150 to verify the theoretical results. The details of all numerical problems can be seen in examples (1–5). Furthermore, we also mentioned the starting points and solutions of the considered problems in the examples. Now, we employ our fourth and sixth-order schemes (5) and (6), called $OS1_4$ and $OS2_6$, respectively, for $\theta = -1$ in order to verify the computational performance of them with the existing methods.

Now, we compare (2) with sixth-order methods suggested by Mona et al. [23] and Lotfi et al. [24]. We consider methods (27) (for $\lambda = 1$, $\beta = 2$, $p = 1$, and $q = 3/2$) and (5), respectively, called MS and LS. In addition, we also compared our scheme with sixth-order family of iterative methods proposed by Sharma and Arora [25]. From them, we choose method (13), called SS. Finally, we compare (2) with a sixth-order family of iterative method that is recently designed by Abbasbandy et al. [26] and Hueso et al. [27].

From them, we choose their best expressions (8) and (14, 15) (for $t_1 = -9/4$ and $s_2 = 9/8$), respectively, denoted by AS and HS.

In Tables 1–5, we depicted the iteration indexes (k), residual error of the corresponding function ($\|F(x^{(j)})\|$), error between the two consecutive iterations $\|x^{(j+1)} - x^{(j)}\|$, approximate computational order of convergence $\rho = \log[\|x^{(j+1)} - x^{(j)}\| / \|x^{(j)} - x^{(j-1)}\|] / \log[\|x^{(j)} - x^{(j-1)}\| / \|x^{(j-1)} - x^{(j-2)}\|]$, $i = 2, 3, \dots$, $\|x^{(j+1)} - x^{(j)}\| / \|x^{(j)} - x^{(j-1)}\|^p$ (where pp is either 4 or 6), and η is the last calculated value.

During the current numerical experiments with programming language Mathematica (Version 9), all computations have been performed with multiple precision arithmetic with 1000 digits of mantissa, which minimize round-off errors. The meaning of $a(\pm b)$ is $a \times 10^{(\pm b)}$ in all tables. We adopted the command ‘‘AbsoluteTiming’’ in order to calculate the CPU time. We run our programs three times and depicted the average CPU time in Table 6. Also, one can observe the times used for each iterative method and each problem in Figure 1, where we want to point out that for big size problems, the method $OS2_6$ uses the minimum time, so it is being very competitive. The configuration of the used computer is given as follows:

Processor: Intel(R) Core(TM) i7-4790 CPU @ 3.60 GHz
 Made of: HP
 RAM: 8:00 GB
 System type: 64-bit-Operating System, x64-based processor

Example 1. Let us consider a boundary value problem described in the book of Ortega and Rheinbolt [16], which is defined as follows:

$$y'' = \frac{1}{2}y^3 + 3y' - \frac{3}{2-x} + \frac{1}{2}, \quad y(0) = 0, y(1) = 1. \quad (27)$$

In addition, we consider the following partition of the interval $[0, 1]$, which is given by

$$x_0 = 0 < x_1 < x_2 < x_3 < \dots < x_n,$$

$$\text{where } i = 0, 1, \dots, n-1, x_{i+1} = x_i + h, h = \frac{1}{n}. \quad (28)$$

Moreover, we also consider the notation, $y_i = y(x_i)$, for $i = 0, 1, \dots, n$. We discretize problem (27) with the help of following numerical approximation for first and second derivatives

$$\begin{aligned}y'_j &\approx \frac{y_{j+1} - y_{j-1}}{2h}, \\ y''_j &\approx \frac{y_{j-1} - 2y_j + y_{j+1}}{h^2}, \quad j = 1, 2, \dots, n-1.\end{aligned}\quad (29)$$

Hence, we yield a nonlinear system of $(n-1) \times (n-1)$, which is given as follows:

TABLE 1: Convergence behavior of different methods on differential equation (27).

Methods	j	$\ F(x^{(j)})\ $	$\ x^{(j+1)} - x^{(j)}\ $	ρ^*	$\ x^{(j+1)} - x^{(j)}\ /\ x^{(j)} - x^{(j-1)}\ ^6$	η
MS	1	1.4 (-4)	5.0 (-4)			
	2	6.5 (-24)	2.1 (-23)		1.270833506 (-3)	3.431307196 (+16)
	3	9.3 (-121)	2.9 (-120)	4.9974	3.431307196 (+16)	
LS	1	1.3 (-3)	4.7 (-3)			
	2	9.6 (-20)	8.2 (-20)		7.809583095 (-6)	1.069128721 (+12)
	3	2.1 (-103)	3.3 (-103)	4.9773	1.069128721 (+12)	
AS	1	1.3 (-3)	4.6 (-3)			
	2	1.5 (-21)	4.9 (-21)		5.332292596 (-7)	6.184712903 (-7)
	3	2.7 (-129)	8.7 (-129)	5.9964	6.184712903 (-7)	
HS	1	2.2 (-4)	4.2 (-4)			
	2	9.0 (-24)	3.5 (-23)		6.562626596 (-3)	1.536785689 (+17)
	3	8.4 (-119)	2.9 (-118)	4.9846	1.536785689 (+17)	
SS	1	6.3 (-4)	2.4 (-3)			
	2	4.5 (-23)	1.5 (-22)		8.028226594 (-7)	9.186167408 (-7)
	3	3.5 (-138)	1.1 (-137)	5.9970	9.186167408 (-7)	
OS1 ₄	1	4.7 (-3)	1.0 (-2)			
	2	7.5 (-13)	2.1 (-12)		1.806107555 (-4)	1761209601 (-4)
	3	1.1 (-51)	3.3 (-51)	4.0011	1.761209601 (-4)	
OS2 ₆	1	5.4 (-5)	1.7 (-4)			
	2	7.4 (-31)	2.1 (-30)		9.979976566 (-8)	9.855399505 (-8)
	3	2.9 (-186)	8.1 (-186)	6.0002	9.855399505 (-8)	

The lowest residual errors, lowest difference between two consecutive iterations, and lowest asymptotic error constants belong to our scheme OS2₆. So, we can say that OS2₆ has better convergence behavior than other mentioned methods.

TABLE 2: Convergence behavior of different methods on Fisher's equation (32).

Methods	j	$\ F(x^{(j)})\ $	$\ x^{(j+1)} - x^{(j)}\ $	ρ^*	$\ x^{(j+1)} - x^{(j)}\ /\ x^{(j)} - x^{(j-1)}\ ^6$	η
MS	1	2.7 (-3)	5.5 (-4)			
	2	9.5 (-21)	1.5 (-21)		5.507288873 (-2)	2.350947999 (+16)
	3	1.6 (-108)	2.5 (-109)	4.9964	2.350947999 (+16)	
LS	1	4.4 (-3)	1.0 (-3)			
	2	4.6 (-20)	7.4 (-21)		5.815894917 (-3)	7.024389352 (+12)
	3	2.2 (-107)	1.2 (-108)	5.1204	7.024389352 (+12)	
AS	1	6.9 (-3)	1.5 (-3)			
	2	4.2 (-21)	6.5 (-22)		7.021836209 (-5)	9.022505886 (-5)
	3	4.5 (-131)	7.0 (-132)	5.9941	9.022505886 (-5)	
HS	1	6.0 (-3)	1.3 (-3)			
	2	1.3 (-18)	2.0 (-19)		5.236602433 (-2)	4.016099735 (+14)
	3	1.8 (-97)	2.8 (-98)	4.9940	4.016099735 (+14)	
SS	1	4.8 (-3)	1.0 (-3)			
	2	2.9 (-22)	4.5 (-23)		4.391559871 (-5)	5.610610203 (-5)
	3	3.0 (-138)	4.7 (-139)	5.9945	5.610610203 (-5)	
OS1 ₄	1	4.8 (-2)	1.1 (-2)			
	2	9.3 (-11)	1.6 (-11)		9.250513914 (-4)	1.088252704 (-3)
	3	4.1 (-46)	6.4 (-47)	3.9920	1.088252704 (-3)	
OS2 ₆	1	9.9 (-4)	2.2 (-4)			
	2	3.5 (-27)	5.5 (-28)		4.830577428 (-6)	5.569289669 (-6)
	3	9.9 (-169)	1.6 (-169)	5.9974	5.569289669 (-6)	

From the table, it is clear that there is big difference of residual errors, among our method OS2₆ and other existing methods. In addition, OS2₆ has the lowest difference between two consecutive iterations and asymptotic error constants.

$$\begin{aligned}
& y_{j+1} - 2y_j + y_{j-1} - \frac{h^2}{2}y_j^3 - \frac{3}{2-x_j}h^2 \\
& - 3\frac{y_{j+1} - y_{j-1}}{2}h - \frac{1}{h^2} = 0, \quad j = 1, 2, \dots, n-1.
\end{aligned} \tag{30}$$

Let us consider initial approximation $y_j^{(0)} = (1.8, \dots, 1.8, 1.8)^T$. In particular, we will solve this problem for $n = 7$ so that we can obtain a 6×6 system of nonlinear equations. The solution of this problem is

TABLE 3: Convergence behavior of different methods on 2D Bratu problem (34).

Methods	j	$\ F(x^{(j)})\ $	$\ x^{(j+1)} - x^{(j)}\ $	ρ^*	$\ x^{(j+1)} - x^{(j)}\ /\ x^{(j)} - x^{(j-1)}\ ^6$	η
MS	1	7.5 (-15)	4.1 (-14)			
	2	1.2 (-80)	6.4 (-80)		1.375781477 (+1)	9.148606524 (+66)
	3	1.2 (-409)	6.3 (-409)	4.9997	9.148606524 (+66)	
LS	1	6.4 (-16)	2.5 (-15)			
	2	2.9 (-87)	5.3 (-87)		1.447342836 (-13)	2.808772449 (+1)
	3	9.1 (-445)	3.3 (-444)	4.9853	2.808772449 (+1)	
AS	1	4.4 (-15)	2.4 (-14)			
	2	6.9 (-95)	3.5 (-94)		1.428095547 (-12)	1.973434769 (-12)
	3	7.9 (-574)	3.9 (-573)	5.9994	1.973434769 (-12)	
HS	1	2.1 (-13)	1.2 (-12)			
	2	2.1 (-71)	1.2 (-70)		7.368055345 (-11)	3.495510769 (+1)
	3	1.7 (-361)	9.3 (-361)	4.9997	3.495510769 (+1)	
SS	1	4.4 (-15)	2.4 (-14)			
	2	7.1 (-95)	3.6 (-94)		1.433541371 (-12)	1.433541371 (-12)
	3	9.2 (-574)	4.5 (-573)	5.9994	1.433541371 (-12)	
OS1 ₄	1	7.5 (-11)	4.1 (-10)			
	2	3.8 (-47)	2.0 (-46)		6.870113499 (-9)	7.171191212 (-9)
	3	2.1 (-192)	1.1 (-191)	3.9995	7.171191212 (-9)	
OS2 ₆	1	2.3 (-18)	1.3 (-17)			
	2	8.2 (-118)	4.7 (-117)		8.429675254 (-16)	8.564522080 (-16)
	3	1.6 (-714)	9.0 (-714)	5.9999	8.564522080 (-16)	

The scheme OS2₆ has the lowest residual error and asymptotic error constant as compared to the other mentioned methods. In addition, lowest difference between two consecutive iterations also belongs to OS2₆.

TABLE 4: Convergence behavior of different methods on Hammerstein integral equation (38).

Methods	j	$\ F(x^{(j)})\ $	$\ x^{(j+1)} - x^{(j)}\ $	ρ^*	$\ x^{(j+1)} - x^{(j)}\ /\ x^{(j)} - x^{(j-1)}\ ^6$	η
MS	1	3.9 (-6)	8.2 (-7)			
	2	5.8 (-36)	1.2 (-36)		3.943508142	2.773373608 (+30)
	3	4.6 (-185)	9.9 (-186)	4.9992	2.773373608 (+30)	
LS	1	8.6 (-6)	1.8 (-6)			
	2	8.4 (-37)	1.7 (-37)		4.445532921 (-3)	1.232404905 (+31)
	3	1.4 (-189)	3.0 (-190)	4.9224	1.232404905 (+31)	
AS	1	1.1 (-5)	2.4 (-6)			
	2	5.4 (-39)	1.2 (-39)		6.596956919 (-6)	7.072478176 (-6)
	3	8.0 (-239)	1.7 (-239)	5.9991	7.072478176 (-6)	
HS	1	3.0 (-5)	6.5 (-6)			
	2	6.9 (-31)	1.5 (-31)		1.994799598	9.220736175 (+25)
	3	4.4 (-159)	9.4 (-160)	4.9991	9.220736175 (+25)	
SS	1	9.4 (-6)	2.0 (-6)			
	2	1.5 (-39)	3.2 (-40)		4.964844066 (-6)	5.324312398 (-6)
	3	2.7 (-242)	5.7 (-243)	5.9991	5.324312398 (-6)	
OS1 ₄	1	9.0 (-4)	1.9 (-4)			
	2	2.0 (-18)	4.2 (-19)		3.089659514 (-4)	3.268514485 (-4)
	3	4.8 (-77)	1.0 (-77)	3.9983	3.268514485 (-4)	
OS2 ₆	1	7.6 (-7)	1.6 (-7)			
	2	3.2 (-47)	6.9 (-48)		3.761775905 (-7)	3.836358838 (-7)
	3	2.0 (-289)	4.2 (-290)	5.9998	3.836358838 (-7)	

From this table, it is clear that there is a big difference of residual errors among OS2₆ and other existing methods. In addition, OS2₆ has the lowest difference between two consecutive iterations and asymptotic error constants.

$$y(x_i) = (0.076543 \dots, 0.165873 \dots, 0.271521 \dots, 0.398454 \dots, 0.553886 \dots, 0.748687 \dots)^T. \quad (31)$$

The numerical results appeared in Table 1. Moreover, we represent in Figure 2 the difference between two consecutive

iterations but in logarithmic scale and taking absolute value, $\|\log(\|x^{(j+1)} - x^{(j)}\|)\|$, so the biggest value corresponds to the minimum distance error, by the properties of logarithmic function, that is, the OS2₆ iterative method presents the best results.

TABLE 5: Convergence behavior of different methods on example 5.

Methods	j	$\ F(x^{(j)})\ $	$\ x^{(j+1)} - x^{(j)}\ $	ρ^*	$\ x^{(j+1)} - x^{(j)}\ /\ x^{(j)} - x^{(j-1)}\ ^6$	η
MS	1	3.4 (-6)	1.5 (-8)			
	2	6.6 (-45)	2.8 (-47)		3.031863687	1.549161749 (+39)
	3	1.9 (-238)	8.2 (-241)	5.0000	1.549161749 (+39)	
LS	1	1.4 (-6)	6.1 (-9)			
	2	2.3 (-55)	1.0 (-57)		1.946645858 (-8)	1.946645858 (-8)
	3	4.5 (-348)	1.9 (-350)	6.0000	1.946645858 (-8)	
AS	1	1.4 (-6)	5.9 (-9)			
	2	1.8 (-55)	7.8 (-58)		1.814507829 (-8)	1.814507829 (-8)
	3	9.7 (-349)	4.1 (-351)	6.0000	1.814507829 (-8)	
HS	1	4.6 (-6)	2.0 (-8)			
	2	3.5 (-44)	1.5 (-46)		2.485682422	3.301595366 (+38)
	3	8.4 (-235)	3.6 (-237)	5.0000	3.301595366 (+38)	
SS	1	7.7 (-7)	3.3 (-9)			
	2	3.1 (-57)	1.3 (-59)		1.007050020 (-8)	1.007050020 (-8)
	3	1.3 (-359)	5.4 (-362)	6.0000	1.007050020 (-8)	
OS1 ₄	1	4.3 (-4)	1.8 (-6)			
	2	9.9 (-27)	4.2 (-29)		3.734034356 (-6)	3.734033747 (-6)
	3	2.8 (-117)	1.2 (-119)	4.0000	3.734033747 (-6)	
OS2 ₆	1	6.5 (-8)	2.8 (-10)			
	2	8.5 (-65)	3.6 (-67)		8.150704909 (-10)	8.150704909 (-10)
	3	4.3 (-406)	1.8 (-408)	6.0000	8.150704909 (-10)	

Our scheme OS2₆ has minimum residual error constant among other mentioned methods. It also has the lowest $\|x^{(4)} - x^{(3)}\|$ and asymptotic error constants as compared to other existing methods.

TABLE 6: CPU time taken by different methods on examples 1–5.

Methods	Ex. 1	Ex. 2	Ex. 3	Ex. 4	Ex. 5	Total time	Average time
MS	0.141097	1.658164	636.148568	49.370628	1418.257989	2105.576446	421.1152892
LS	0.109075	1.031724	323.672644	24.845433	973.952124	1323.611	264.7222
AS	0.586411	1.590115	634.605760	49.738892	1492.151852	2178.67303	435.734606
HS	0.646454	2.371663	611.849108	50.585468	2207.404740	2872.857433	574.571866
SS	1.262873	1.570112	376.072834	148.621206	1018.385809	1545.912834	309.1825668
OS1 ₄	0.260183	0.609427	251.443292	25.600944	779.653091	1057.566937	211.5133874
OS2 ₆	0.283189	0.764537	364.446516	25.026559	723.418750	1113.939551	222.7879102

Our scheme OS1₄ has the lowest average CPU time. Since, OS1₄ has convergence order fourth and others have sixth. Then, it is quite obvious that OS1₄ has lower convergence behavior as compared to other mentioned methods in terms of residual error, asymptotic error constant, etc. However, it is better than the others in regard of CPU timing. On the other hand, the second lowest average CPU time belongs to our sixth-order convergent scheme OS2₆ and also illustrates better convergence performance as compared to other mentioned methods.

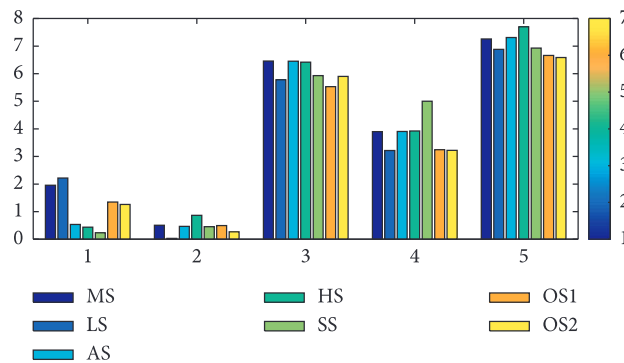


FIGURE 1: CPU times for each example with the corresponding method.

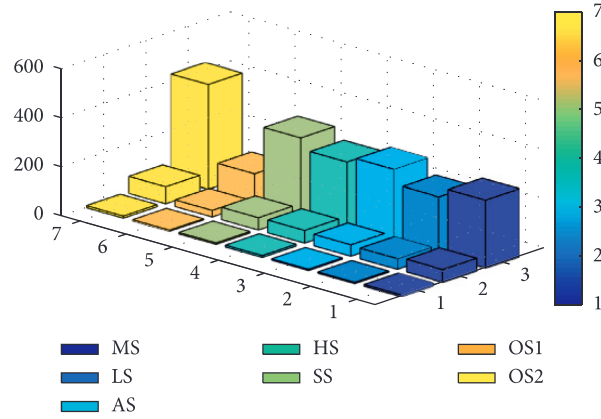


FIGURE 2: Difference between consecutive iterations in logarithmic scale for example 1.

Example 2. Consider another typical nonlinear problem, that is, Fisher's equation [28] with homogeneous Neumann's boundary conditions and D the diffusion coefficient:

$$\begin{aligned} u_t &= Du_{xx} + u(1 - u), \\ u(x, 0) &= 1.5 + 0.5 \cos(\pi x), \quad 0 \leq x \leq 1, \\ u_x(0, t) &= 0, \quad \forall t \geq 0, \\ u_x(1, t) &= 0, \quad \forall t \geq 0. \end{aligned} \quad (32)$$

Again using a finite difference discretization, equation (32) reduces to a system of nonlinear equations. Consider

$u_{i,j} = u(x_i, t_j)$ to be its approximate solution at the grid points of the mesh. Let M and N be the number of steps in x and t directions, and h and k be the respective step size. Apply central difference to $u_{xx}(x_i, t_j) = (u_{i+1,j} - 2u_{i,j} + u_{i-1,j})/h^2$, backward difference for $u_t(x_i, t_j) = (u_{i,j} - u_{i,j-1})/k$, and forward difference for $u_x(x_i, t_j) = (u_{i+1,j} - u_{i,j})/h$, $t \in [0, 1]$. For the solution of the system, we consider $M = 5$ and $N = 5$ which reduces to a nonlinear system of size 25, with the initial vector $u_0(x_i, t_j) = (1 + i + j/M^2)^T$, $i, j = 1, 2, \dots, M$ convergence towards the following solution:

$$u(x_i, t_j) = \begin{pmatrix} 1.4606 \dots, 1.2959 \dots, 1.2166 \dots, 1.1688 \dots, 1.1355 \dots \\ 1.3988 \dots, 1.2783 \dots, 1.2113 \dots, 1.1672 \dots, 1.1349 \dots \\ 1.3081 \dots, 1.2509 \dots, 1.2028 \dots, 1.1645 \dots, 1.1341 \dots \\ 1.2260 \dots, 1.2245 \dots, 1.1944 \dots, 1.1619 \dots, 1.1333 \dots \\ 1.1812 \dots, 1.2089 \dots, 1.1893 \dots, 1.1602 \dots, 1.1327 \dots \end{pmatrix}, \quad (33)$$

and the numerical results of comparing the behavior different methods are shown in Table 2.

Example 3. The 2D Bratu problem [29, 30] is defined in $\Omega = \{(x, t): 0 \leq x \leq 1, 0 \leq t \leq 1\}$, as

$$\begin{aligned} u_{xx} + u_{tt} + Ce^u &= 0, \text{ on} \\ \text{with boundary conditions } u &= 0 \text{ on } \partial\Omega. \end{aligned} \quad (34)$$

The approximated solution of a nonlinear partial differential equation can be found using finite difference discretization which reduces to solving a system of nonlinear equations. Let $u_{i,j} = u(x_i, t_j)$ be its approximate solution at the grid points of the mesh. Let M and N be the number of steps in x and t directions and h and k be the respective step size. To solve the given PDE, apply central difference to u_{xx} and u_{tt} , i.e., $u_{xx}(x_i, t_j) = (u_{i+1,j} - 2u_{i,j} + u_{i-1,j})/h^2$, $C = 0.1, t \in [0, 1]$. We look for the solution of the system for $M = 11$ and $N = 11$ of size 100, with the initial sinusoidal approximation $u_0(x_i, t_j) = 0.1(\sin(\pi ih)\sin(\pi jk))$; by using different methods, one can see the obtained

result in Table 3. The solution of this problem can be seen in Figure 3.

Example 4. In this example, we consider the following Hammerstein integral equation (see [pp. 19, 20]) Ortega to check the effectiveness and applicability of our proposed methods, and compared with other existing methods, the nonlinear integral equation is given by

$$x(s) = 1 + \frac{1}{5} \int_0^1 F(s, t)x(t)^3 dt, \quad (35)$$

where $x(s) \in C[0, 1]$, $s, t \in [0, 1]$ and the kernel F can be written as

$$F(s, t) = \begin{cases} (1-s)t, & \text{if } t \leq s, \\ s(1-t), & \text{if } s \leq t. \end{cases} \quad (36)$$

To transform the above equation into a finite-dimensional problem by using Gauss Legendre quadrature formula given as $\int_0^1 f(t)dt \approx \sum_{j=1}^n w_j f(t_j)$, where the abscissas t_j and the weights w_j are determined for $n = 8$. Denoting the

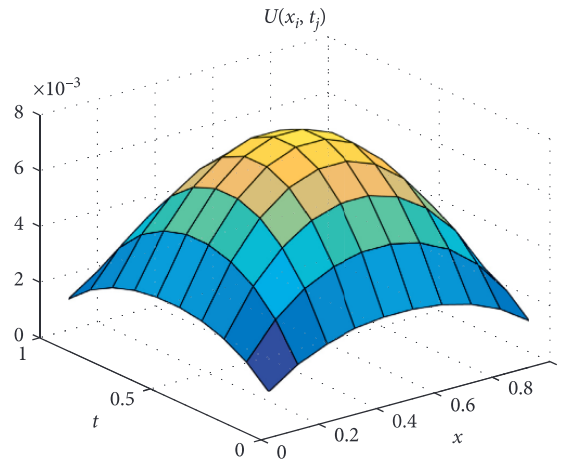


FIGURE 3: Approximated solution for Bratu equation with $C = 0.1$.

TABLE 7: Abscissas and weights of Gauss Legendre quadrature formula for $t = 8$.

j	t_j	w_j
1	0.01985507175123188415821957...	0.05061426814518812957626567...
2	0.10166676129318663020422303...	0.11119051722668723527217800...
3	0.23723379504183550709113047...	0.15685332293894364366898110...
4	0.40828267875217509753026193...	0.18134189168918099148257522...
5	0.59171732124782490246973807...	0.18134189168918099148257522...
6	0.76276620495816449290886952...	0.15685332293894364366898110...
7	0.89833323870681336979577696...	0.11119051722668723527217800...
8	0.98014492824876811584178043...	0.05061426814518812957626567...

approximations of $x(t_i)$ by x_i , one gets the system of nonlinear equations:

$$5x_i - 5 - \sum_{j=1}^8 a_{ij}x_j^3 = 0, \quad \text{where } i = 1, 2, \dots, 8,$$

$$a_{ij} = \begin{cases} w_j t_j (1 - t_i), & \text{if } j \leq i, \\ w_j t_i (1 - t_j), & \text{if } i < j. \end{cases} \quad (37)$$

where the abscissas t_j and the weights w_j are known and depicted in Table 7 when $n = 8$.

The different methods converge towards the solution $x(t_i) = (1.00209 \dots, 1.00990 \dots, 1.01972 \dots, 1.02643 \dots, 1.02643 \dots, 1.01972 \dots, 1.00990 \dots, 1.00209 \dots)^T$, and different conditions of convergence can be checked in Table 1. The initial guess $x^{(0)} = (1/2, \dots, 1/2)^T$.

Example 5. Finally, let us consider the following system of nonlinear equations (chosen from Grau-Sánchez et al. [31]):

$$F(x_1, x_2, \dots, x_n) = \sum_{j=1, j \neq i}^n x_j - e^{-x_i}, \quad 1 \leq i \leq n. \quad (38)$$

We choose $n = 150$ in order to check the theoretical results mentioned with a large size system. We take the initial guess $x^{(0)} = (1/2, \dots, 1/2)^T$ for this problem. The obtained solution of this problem is

$$\xi = (0.5671433 \dots, 0.5671433 \dots, 0.5671433 \dots, \dots, 0.5671433 \dots (150 \text{ times}))^T. \quad (39)$$

The obtained results can be observed in Table 5.

4. Concluding Remarks

In this paper, we proposed a new high-order and efficient family of iterative techniques for a system of nonlinear equations. The suggested methods are the extension of the study by Kou et al. [19] for multidimensional case. In addition, our scheme has minimum 5th-order convergence by adding additional substep and for particular value $\theta = -1$, it reached the maximum 6th order. From the obtained numerical results in Tables 1–5, it is confirmed that the lowest residual error, difference between two consecutive iterations and asymptotic error constant, belongs to our method OS2₆. Finally, our schemes OS1₆ and OS2₆, respectively, also consume the first and second lowest average CPU time as compared to the other existing methods in examples 1–5. In the future work, we will try to propose higher-order version of the same or new methods with lower computational cost and that employ more accurate solution.

Data Availability

In this paper, we want to construct a new high-order and efficient iterative technique for solving a system of nonlinear

equations. For this purpose, we extend an earlier scalar scheme to a system of nonlinear equations preserving the same convergence order. Moreover, by adding one additional step, we obtain minimum 5th-order convergence for every value of a free parameter, and in some specific cases, the method reaches maximum 6-order convergence. We present an extensive convergence analysis of our scheme. The analytical discussion of the work is upheld by performing numerical experiments on some applied science problems and a large system of nonlinear equations. Further, numerical results demonstrate the validity and reliability of the suggested methods.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was funded by the Deanship of Scientific Research (DSR), King Abdulaziz University, Jeddah, under grant no. (D-349-130-1441). The authors, therefore, acknowledge with thanks, DSR technical and financial support.

References

- [1] J. J. Rubio, D. R. Cruz, I. Elias, G. Ochoa, R. Balcazar, and A. Aguilar, "ANFIS system for classification of brain signals," *Journal of Intelligent and Fuzzy Systems*, vol. 37, no. 3, pp. 4033–4041, 2019.
- [2] J. J. Rubio, "SOFMLS: online self-organizing fuzzy modified least-squares network," *IEEE Transactions on Fuzzy Systems*, vol. 17, no. 6, pp. 1296–1309, 2009.
- [3] C. N. Giap, L. H. Son, and F. Chiclana, "Dynamic structural neural network," *Journal of Intelligent & Fuzzy Systems*, vol. 34, no. 4, pp. 2479–2490, 2018.
- [4] J. J. Rubio, E. Garcia, G. Ochoa et al., "Unscented Kalman filter for learning of a solar dryer and a greenhouse," *Journal of Intelligent and Fuzzy Systems*, vol. 37, no. 5, pp. 6731–6741, 2019.
- [5] F. Awawdeh, "On new iterative method for solving systems of nonlinear equations," *Numerical Algorithms*, vol. 54, no. 3, pp. 395–409, 2010.
- [6] 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.
- [7] Y. Lin, L. Bao, and X. Jia, "Convergence analysis of a variant of the Newton method for solving nonlinear equations," *Computers & Mathematics with Applications*, vol. 59, no. 6, pp. 2121–2127, 2010.
- [8] J. J. Moré, "A collection of nonlinear model problems," in *Computational Solution of Nonlinear Systems of Equations, Lectures in Applied Mathematics*, E. L. Allgower and K. Georg, Eds., Vol. 26, American Mathematical Society, Providence, RI, USA, 1990.
- [9] I. G. Tsoulos and A. Stavrakoudis, "On locating all roots of systems of nonlinear equations inside bounded domain using global optimization methods," *Nonlinear Analysis: Real World Applications*, vol. 11, no. 4, pp. 2465–2471, 2010.
- [10] 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.
- [11] M. F. Abad, A. Cordero, and J. R. Torregrosa, "A family of seventh-order schemes for solving nonlinear systems," *Bulletin Mathématique*, vol. 57, no. 105, pp. 133–145, 2014.
- [12] A. Cordero, J. G. Maimó, J. R. Torregrosa, and M. P. Vassileva, "Solving nonlinear problems by Ostrowski-Chun type parametric families," *Journal of Mathematical Chemistry*, vol. 53, no. 1, pp. 430–449, 2014.
- [13] X. Wang and T. Zhang, "A family of Steffensen type methods with seventh-order convergence," *Numerical Algorithms*, vol. 62, no. 3, pp. 429–444, 2013.
- [14] J. R. Sharma, R. K. Guha, and R. Sharma, "An efficient fourth order weighted-Newton method for systems of nonlinear equations," *Numerical Algorithms*, vol. 62, no. 2, pp. 307–323, 2013.
- [15] S. Artidiello, A. Cordero, J. R. Torregrosa, and M. P. Vassileva, "Multidimensional generalization of iterative methods for solving nonlinear problems by means of weight-function procedure," *Applied Mathematics and Computation*, vol. 268, pp. 1064–1071, 2015.
- [16] J. M. Ortega and W. C. Rheinboldt, *Iterative Solution of Nonlinear Equations in Several Variables*, Academic Press, New-York, NY, USA, 1970.
- [17] I. K. Argyros, *Convergence and Application of Newton-type Iterations*, Springer, Berlin, Germany, 2008.
- [18] J. F. Traub, *Iterative Methods for the Solution of Equations*, Prentice-Hall, Upper Saddle River, NJ, USA, 1964.
- [19] J. Kou, Y. Li, and X. Wang, "A composite fourth-order iterative method for solving non-linear equations," *Applied Mathematics and Computation*, vol. 184, no. 2, pp. 471–475, 2007.
- [20] F. A. Potra and V. Pták, "Nondiscrete induction and iterative processes," *Research Notes in Mathematics*, vol. 103, Pitman, Boston, MA, USA, 1984.
- [21] J. R. Sharma, "A composite third order Newton-Steffensen method for solving nonlinear equations," *Applied Mathematics and Computation*, vol. 169, no. 1, pp. 242–246, 2005.
- [22] A. K. H. Alzahrani, R. Behl, and A. S. Alshomrani, "Some higher-order iteration functions for solving nonlinear models," *Applied Mathematics and Computation*, vol. 334, pp. 80–93, 2018.
- [23] M. Narang, S. Bhatia, and V. Kanwar, "New two-parameter Chebyshev-Halley-like family of fourth and sixth-order methods for systems of nonlinear equations," *Applied Mathematics and Computation*, vol. 275, pp. 394–403, 2016.
- [24] T. Lotfi, P. Bakhtiari, A. Cordero, K. Mahdiani, and J. R. Torregrosa, "Some new efficient multipoint iterative methods for solving nonlinear systems of equations," *International Journal of Computer Mathematics*, vol. 92, no. 9, pp. 1921–1934, 2015.
- [25] J. R. Sharma and H. Arora, "Efficient Jarratt-like methods for solving systems of nonlinear equations," *Calcolo*, vol. 51, no. 1, pp. 193–210, 2014.
- [26] S. Abbasbandy, P. Bakhtiari, A. Cordero, J. R. Torregrosa, and T. Lotfi, "New efficient methods for solving nonlinear systems of equations with arbitrary even order," *Applied Mathematics and Computation*, vol. 287–288, pp. 94–103, 2016.
- [27] J. L. Hueso, E. Martínez, and C. Teruel, "Convergence, efficiency and dynamics of new fourth and sixth order families of iterative methods for nonlinear systems," *Journal of Computational and Applied Mathematics*, vol. 275, pp. 412–420, 2015.
- [28] T. Sauer, *Numerical Analysis*, Pearson, Columbus, OH, USA, 2nd edition, 2012.

- [29] R. B. Simpson, "A method for the numerical determination of bifurcation states of nonlinear systems of equations," *SIAM Journal on Numerical Analysis*, vol. 12, no. 3, pp. 439–451, 1975.
- [30] R. K. Kapania, "A pseudo-spectral solution of 2-parameter Bratu's equation," *Computational Mechanics*, vol. 6, no. 1, pp. 55–63, 1990.
- [31] M. Grau-Sánchez, M. Noguera, and S. Amat, "On the approximation of derivatives using divided difference operators preserving the local convergence order of iterative methods," *Journal of Computational and Applied Mathematics*, vol. 237, no. 1, pp. 363–372, 2013.