

Article

Higher-Order Iteration Schemes for Solving Nonlinear Systems of Equations

Hessah Faihan Alqahtani ¹, Ramandeep Behl ^{2,*} and Munish Kansal ³

¹ Department of Mathematics, King Abdulaziz University, Campus Rabigh 21911, Saudi Arabia; hfalqahtani@kau.edu.sa

² Department of Mathematics, King Abdulaziz University, Jeddah 21589, Saudi Arabia

³ School of Mathematics, Thapar Institute of Engineering and Technology, Patiala 147004, India; mkmaths@gmail.com

* Correspondence: ramanbehl87@yahoo.in

Received: 23 August 2019; Accepted: 24 September 2019; Published: 10 October 2019



Abstract: We present a three-step family of iterative methods to solve systems of nonlinear equations. This family is a generalization of the well-known fourth-order King's family to the multidimensional case. The convergence analysis of the methods is provided under mild conditions. The analytical discussion of the work is upheld by performing numerical experiments on some application oriented problems. Finally, numerical results demonstrate the validity and reliability of the suggested methods.

Keywords: systems of nonlinear equations; King's family; order of convergence; multipoint iterative methods

1. Introduction

System of nonlinear equations (SNEs) finds applications to numerous phenomena in many areas of science and engineering. Given a nonlinear system, $F(X) = 0$, where F is a nonlinear map from $\mathbb{R}^k \rightarrow \mathbb{R}^k$, we are interested to compute a vector $X^* = (x_1^*, x_2^*, \dots, x_k^*)^T$ such that $F(X^*) = 0$, where $F(X) = (f_1(X), f_2(X), \dots, f_k(X))^T$ is a Fréchet differentiable function and $X = (x_1, x_2, \dots, x_k)^T \in \mathbb{R}^k$. The classical Newton's method [1] is the most famous procedure to solve SNEs. It is given by

$$X^{(k+1)} = X^{(k)} - \{F'(X^{(k)})\}^{-1}F(X^{(k)}), \quad k = 0, 1, 2, \dots \quad (1)$$

It converges quadratically if the function F is continuously differentiable and the initial approximation is close enough. In the literature, there are variety of higher-order methods that improve the convergence order of Newton's scheme. For example, several authors have proposed cubically convergent methods [2–5] requiring computation of 2- F' (2- F' stand for F' two times), 1- F (1- F stands for F one time), and two matrix inversions per step. In [6], the authors developed another family of methods of order three, one of which requires one 1- F and 3- F' , whereas the other requires 1- F and 4- F' evaluations and two matrix inversions per iteration. In [7], Darvishi and Barati utilized 2- F , 2- F' and two matrix inversions per step to propose a new third-order scheme. Similarly, several third-order methods have been proposed in [8,9] that require 2- F , 1- F' , and one matrix inversion. Babajee et al. [10] presented a method having convergence order four which consumes 1- F , 2- F' and two matrix inversions per iteration. Another fourth-order method is developed in [11] using two evaluations of the function and the Jacobian and one matrix inversion, whereas the authors of [12] propose another fourth-order method, utilizing 3- F , 1- F' , and one matrix inversion per iteration. Another fifth-order method in [13] requires three evaluations of the function and only one Jacobian evaluation, with the solution of three linear systems with the same matrix of coefficients per iteration.

In pursuit of faster algorithms, researchers have also developed fifth and sixth-order methods, for example, in [6,14–16]. In [15], Narang et al. extended the existing Babajee’s fourth-order scheme [17] to solve SNEs and developed a sixth-order convergent family of Chebyshev-Halley type methods. Their scheme requires two F , two F' evaluations, and the solution of two linear systems per iteration. One can notice that although the researchers are making an attempt to improve the order of convergence of an iterative method, it mostly leads to increase in the computational cost per iteration. The computational cost is especially high if the method involves the use of second order Fréchet derivative $F''(X)$. This is a major limitation of the higher-order methods. Thus, although developing new iterative methods, we should try to keep the computational cost low. With this intention, we have made an attempt to develop a family of three-step sixth-order family of methods requiring two F , two F' and one matrix inversion per iteration. This family of methods are compared to be more efficient than existing methods. These have been found to be effective in solving particularly large-scale nonlinear systems.

The outline of the manuscript is as follows. In Section 2, a new class of new sixth-order scheme and its convergence analysis is presented. In Section 3, we present numerous illustrative examples to validate the theoretical results. Finally, Section 4 contains some conclusions.

2. Design of the King’s Family for Multidimensional Case

In this section, we proposed a new three-point extension of King’s method [18–21] having sixth-order convergence. For this purpose, we consider the well-known fourth-order King’s method, which is given by

$$\begin{aligned}
 y_k &= x_k - \frac{f(x_k)}{f'(x_k)}, \\
 x_{k+1} &= y_k - \frac{1 + \alpha u_k}{1 + (\alpha - 2)u_k} \frac{f(y_k)}{f'(x_k)},
 \end{aligned}
 \tag{2}$$

where α is a real parameter and $u_k = \frac{f(y_k)}{f(x_k)}$. For $\alpha = 0$, one can obtain the well-known Ostrowski’s method [22–24].

Let us now modify the method (2) for SNEs by rewriting the scheme as follows,

$$\begin{aligned}
 u_k &= \frac{f(y_k) - f(x_k) + f(x_k)}{f(x_k)} \\
 &= \frac{f(y_k) - f(x_k)}{f(x_k)} + 1 \\
 &= 1 - \frac{f(y_k) - f(x_k)}{(y_k - x_k)f'(x_k)} \\
 &= 1 - f'(x_k)^{-1}[y_k, x_k; f],
 \end{aligned}$$

where $[y_k, x_k; f] = \frac{f(y_k) - f(x_k)}{y_k - x_k}$. Finally, we can rewrite the above scheme (2) for SNEs with one additional sub-step in the following manner,

$$\begin{aligned}
 y^{(k)} &= x^{(k)} - F'(x^{(k)})^{-1}F(x^{(k)}), \\
 z^{(k)} &= y^{(k)} - (I + (\alpha - 2)U^{(k)})^{-1}(I + \alpha U^{(k)})F'(x^{(k)})^{-1}F(y^{(k)}), \\
 x^{(k+1)} &= z^{(k)} - ([y^{(k)}, z^{(k)}; F])^{-1}F(z^{(k)}),
 \end{aligned}
 \tag{3}$$

where $[\cdot, \cdot; F]$ is a finite difference of order one and α is a free disposable parameter with $U^{(k)} = I - [x^{(k)}, y^{(k)}; F]F'(x^{(k)})^{-1}$. In addition, $F[Y_n, X_n]$ is a finite difference of order one.

Now, it is necessary to analyze the convergence conditions of this modified King’s class of methods. In Theorem 1, we demonstrate the convergence order of the above scheme (3). We have used the following procedures [25] to prove the convergence results.

Let $F : \Omega \subseteq \mathbb{R}^k \rightarrow \mathbb{R}^k$ be sufficiently differentiable in Ω . Now, we define the q th derivative of F at $\omega \in \Omega, q \geq 1$. It can be viewed as a q -linear function $F^{(q)}(\omega) : \mathbb{R}^k \times \dots \times \mathbb{R}^k \rightarrow \mathbb{R}^k$, such that $F^{(q)}(\omega)(v_1, \dots, v_q) \in \mathbb{R}^k$. It is easy to observe that

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

Using the above relations, we can introduce the following notation,

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

Now, applying Taylor’s expansion for $\xi^* + h \in \mathbb{R}^k$ in the neighborhood of a solution ξ^* of the given linear system, one can get

$$F(\xi^* + h) = F'(\xi^*) \left[h + \sum_{q=2}^{p-1} C_q h^q \right] + O(h^p), \tag{4}$$

where $C_q = (1/q!)[F'(\xi^*)]^{-1}F^{(q)}(\xi^*), q \geq 2$. We note that $C_q h^q \in \mathbb{R}^k$ as $F^{(q)}(\xi^*) \in \mathcal{L}(\mathbb{R}^k \times \dots \times \mathbb{R}^k, \mathbb{R}^k)$, and $[F'(\bar{x})]^{-1} \in \mathcal{L}(\mathbb{R}^k)$.

Similarly, we can express F' as

$$F'(\xi^* + h) = F'(\xi^*) \left[I + \sum_{q=2}^{p-1} qC_q h^{q-1} \right] + O(h^{p-1}), \tag{5}$$

where I denotes the identity matrix. Therefore, $qC_q h^{q-1} \in \mathcal{L}(\mathbb{R}^k)$. From Equation (5), we obtain

$$[F'(\xi^* + h)]^{-1} = \left[I + X_2 h + X_3 h^2 + X_4 h^4 + \dots \right] [F'(\xi^*)]^{-1} + O(h^p), \tag{6}$$

where

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

Let us denote $e^{(k)} = x^{(k)} - \xi^*$ as the error at the k th iteration. Then, the error equation is given as follows,

$$e^{(k+1)} = M(e^{(k)})^p + O((e^{(k)})^{p+1}),$$

where, M is a p -linear function $M \in \mathcal{L}(\mathbb{R}^k \times \dots \times \mathbb{R}^k, \mathbb{R}^k)$. Here, p is the order of convergence and $(e^{(k)})^p$

is a column vector $\overbrace{(e^{(k)}, e^{(k)}, \dots, e^{(k)})^T}^p$.

Theorem 1. Let $F : \Omega \subseteq \mathbb{R}^k \rightarrow \mathbb{R}^k$ be a sufficiently differentiable function defined on a convex set Ω containing the zero ξ^* . Let us assume that $F'(x)$ is continuous and non-vanishing at ξ^* . If the initial guess $x^{(0)}$ is close enough to ξ^* , the iterative scheme (3) attains sixth-order convergence for each α .

Proof. Let $e^{(k)} = x^{(k)} - \zeta^*$ be the error at the k th-iteration. Now, expanding $F(x^{(k)})$ and $F'(x^{(k)})$ using Taylor's expansion in a neighborhood of ζ^* , we get

$$F(x^{(k)}) = F'(\zeta^*) \left[e^{(k)} + C_2(e^{(k)})^2 + C_3(e^{(k)})^3 + C_4(e^{(k)})^4 + C_5(e^{(k)})^5 + C_6(e^{(k)})^6 \right] + O((e^{(k)})^7) \tag{7}$$

and

$$F'(x^{(k)}) = F'(\zeta^*) \left[I + 2C_2e^{(k)} + 3C_3(e^{(k)})^2 + 4C_4(e^{(k)})^3 + 5C_5(e^{(k)})^4 + 6C_6(e^{(k)})^5 \right] + O((e^{(k)})^6), \tag{8}$$

where I is the identity matrix of size $n \times n$ and $C_k = \frac{1}{k!} F'(\zeta^*)^{-1} F^{(k)}(\zeta^*)$, $k \geq 2$.

With the help of above expression (8), we have

$$F'(x^{(k)})^{-1} = \left[I - 2C_2e^{(k)} + \Delta_0(e^{(k)})^2 + \Delta_1(e^{(k)})^3 + \Delta_2(e^{(k)})^4 + \Delta_3(e^{(k)})^5 + \Delta_4(e^{(k)})^6 + O((e^{(k)})^7) \right] F'(\zeta^*)^{-1} \tag{9}$$

where $\Delta_i = \Delta_i(C_2, C_3, \dots, C_6)$, for example, $\Delta_0 = 4C_2^2 - 3C_3$, $\Delta_1 = -(8C_2^3 - 6C_2C_3 - 6C_3C_2 + 4C_4)$, $\Delta_2 = 8C_2C_4 + 9C_3^2 + 8C_4C_2 - 12C_2^2C_3 - 12C_2C_3C_2 - 12C_3C_2^2 + 16C_4^2 - 5C_5$, $\Delta_3 = 10C_2C_5 + 12C_3C_4 + 12C_4C_3 + 10C_5C_2 - 16C_2^2C_4 - 18C_2C_3^2 - 16C_2C_4C_2 - 18C_3C_2C_3 - 18C_3^2C_2 - 16C_4C_2^2 + 24C_2^3C_3 + 24C_2^2C_3C_2 + 24C_2C_3C_2^2 + 24C_3C_2^3 - 32C_2^5 - 6C_6$, etc.

From expressions (7) and (9), we yield

$$F'(x^{(k)})^{-1}F(x^{(k)}) = e^{(k)} + \Theta_0(e^{(k)})^2 + \Theta_1(e^{(k)})^3 + \Theta_2(e^{(k)})^4 + \Theta_3(e^{(k)})^5 + \Theta_4(e^{(k)})^6 + O((e^{(k)})^7). \tag{10}$$

where $\Theta_j = \Theta_j(C_2, C_3, \dots, C_6)$, for example, $\Theta_0 = -C_2$, $\Theta_1 = 2C_2^2 - 2C_3$, $\Theta_2 = -(4C_2^3 - 4C_2C_3 - 3C_3C_2 + 3C_4)$, $\Theta_3 = 6C_2C_4 + 6C_3^2 + 4C_4C_2 - 8C_2^2C_3 - 6C_2C_3C_2 - 6C_3C_2^2 + 8C_4^2 - 4C_5$, $\Theta_4 = 8C_2C_5 + 9C_3C_4 + 8C_4C_3 + 5C_5C_2 - 12C_2^2C_4 - 12C_2C_3^2 - 8C_2C_4C_2 - 12C_3C_2C_3 - 9C_3^2C_2 - 8C_4C_2^2 + 16C_2^3C_3 + 12C_2^2C_3C_2 + 12C_2C_3C_2^2 + 12C_3C_2^3 - 16C_2^5 - 5C_6$, etc.

By inserting the expression (10) in the first substep of (3), we obtain

$$y^{(k)} - \zeta^* = -\Theta_0(e^{(k)})^2 - \Theta_1(e^{(k)})^3 - \Theta_2(e^{(k)})^4 - \Theta_3(e^{(k)})^5 - \Theta_4(e^{(k)})^6 + O((e^{(k)})^7). \tag{11}$$

which further produces

$$F(y^{(k)}) = F'(\zeta^*) \left[-\Theta_0(e^{(k)})^2 - \Theta_1(e^{(k)})^3 + (C_2\Theta_0^2 - \Theta_2)(e^{(k)})^4 + (2C_2\Theta_0\Theta_1 - \Theta_3)(e^{(k)})^5 + \right. \\ \left. - (C_3\Theta_0^3 - C_2(\Theta_1^2 + 2\Theta_0\Theta_2) + \Theta_4)(e^{(k)})^6 + O((e^{(k)})^7) \right] \tag{12}$$

and

$$U^{(k)} = I - F'(x^{(k)})^{-1}[x^{(k)}, y^{(k)}; F] = C_2e^{(k)} + (C_2\Theta_0 - 2C_2^2 + 2C_3)(e^{(k)})^2 + \left(-2C_2^2\Theta_0 + C_2(\Theta_1 - 7C_3) + C_3\Theta_0 + 4C_3^2 \right. \\ \left. + 3C_4 \right) (e^{(k)})^3 + \left(4C_2^3\Theta_0 + C_2^2(20C_3 - 2\Theta_1) + C_2(-5C_3\Theta_0 - 10C_4 + \Theta_2) + C_4\Theta_0 \right. \\ \left. + C_3(\Theta_1 - \Theta_0^2) - 8C_2^4 - 6C_3^2 + 4C_5 \right) (e^{(k)})^4 + \left[-8C_2^4\Theta_0 + C_2^3(4\Theta_1 - 52C_3) \right. \\ \left. + 2C_2^2(8C_3\Theta_0 + 14C_4 - \Theta_2) + C_2(C_3(2\Theta_0^2 - 5\Theta_1) - 6C_4\Theta_0 + 33C_3^2 - 13C_5 + \Theta_3) \right. \\ \left. - C_4\Theta_0^2 - 3C_3^2\Theta_0 + C_5\Theta_0 + C_4\Theta_1 + C_3(-17C_4 - 2\Theta_0\Theta_1 + \Theta_2) + 16C_2^5 + 5C_6 \right] (e^{(k)})^5 \\ + O((e^{(k)})^6). \tag{13}$$

By using expressions (9), (12), and (13), we obtain

$$\left(I + (\alpha - 2)U^{(k)} \right)^{-1} \left(I + \alpha U^{(k)} \right) F'(x^{(k)})^{-1} F(y^{(k)}) = -\Theta_0(e^{(k)})^2 - \Theta_1(e^{(k)})^3 + (2\alpha C_2^2\Theta_0 - C_2\Theta_0^2 \\ - C_3\Theta_0 - \Theta_2)(e^{(k)})^4 + O((e^{(k)})^5) \tag{14}$$

By adopting expressions (11)–(14) in the scheme (3), we have

$$z^{(k)} - \zeta^* = \tau_1(e^{(k)})^4 + \tau_2(e^{(k)})^5 + \tau_3(e^{(k)})^6 + O((e^{(k)})^7). \tag{15}$$

where $\tau_j = \tau_j(\Theta_0, \Theta_1, \dots, \Theta_4, C_2, C_3, \dots, C_6, \alpha), j = 1, 2, 3$, for example, $\tau_1 = \Theta_0(-2\alpha C_2^2 + C_2\Theta_0 + C_3)$. Now, expanding the $F(z^{(k)})$ in a neighborhood of ζ^* , we have

$$F(z^{(k)}) = F'(\zeta^*) [\tau_1(e^{(k)})^4 + \tau_2(e^{(k)})^5 + \tau_3(e^{(k)})^6 + O((e^{(k)})^7)], \tag{16}$$

which further produces with the help of expression (12)

$$[z^{(k)}, y^{(k)}; F]^{-1} F(z^{(k)}) = \tau_1(e^{(k)})^4 + \tau_2(e^{(k)})^5 + (C_2\Theta_0\tau_1 + \tau_3)(e^{(k)})^6 + O((e^{(k)})^7). \tag{17}$$

By using equation (17) in the final substep of (3), we have

$$\begin{aligned} e^{(j+1)} &= (\tau_1(e^{(k)})^4 + \tau_2(e^{(k)})^5 + \tau_3(e^{(k)})^6) - (\tau_1(e^{(k)})^4 + \tau_2(e^{(k)})^5 + (C_2\Theta_0\tau_1 + \tau_3)(e^{(k)})^6) + O((e^{(k)})^7) \\ &= -(C_2\Theta_0\tau_1)(e^{(k)})^6 + O((e^{(k)})^7) \\ &= C_2^3((2\alpha + 1)c_2^2 - c_3)(e^{(k)})^6 + O((e^{(k)})^7). \end{aligned} \tag{18}$$

Therefore, the scheme (3) has sixth-order convergence. \square

3. Numerical Experiments

Here, we checked the efficiency and effectiveness of our scheme on real-life and standard academic test problems. Therefore, we consider five number of the examples' details, as seen in the examples (1)–(5). Further, we also depicted the starting points and zeros of the considered nonlinear system in the examples (1)–(5). Now, we employ our sixth-order scheme (3), called (PM), to verify the computational performance of them with existing methods considered in the previous section.

Now, we compare (3) with a sixth-order family proposed by Abbasbandy et al. [26] and Hueso et al. [27]. We choose their best expressions (8) and (14–15) (for $t_1 = -\frac{9}{4}$ and $s_2 = \frac{9}{8}$), respectively denoted by (AM₆) and (HM₆). Moreover, a comparison of a newly proposed scheme has been done with the sixth-order family of iterative method proposed by Sharma and Arora [28] and Wang and Li [29], out these works we choose two methods, namely, (13) and (6), respectively, called (SM₆) and (WM₆). Finally, we compare (3) with sixth-order methods suggested by Mona et al. [15] and Lotfi et al. [16], we consider methods (3.1) (for $\lambda = 1, \beta = 2, p = 1$ and $q = \frac{3}{2}$) and (5), respectively, called by (MM₆) and (LM₆). All the iterative expressions are mentioned below.

Method AM₆:

$$\begin{aligned} y^{(k)} &= x^{(k)} - \frac{2}{3}F'(x^{(k)})^{-1}F(x^{(k)}), \\ z^{(k)} &= x^{(k)} - \left[I + \frac{21}{8}F'(x^{(k)})^{-1}F'(y^{(k)}) - \frac{9}{2}(F'(x^{(k)})^{-1}F'(y^{(k)}))^2 + \frac{15}{8}(F'(x^{(k)})^{-1}F'(y^{(k)}))^3 \right] F'(x^{(k)})^{-1}F(x^{(k)}), \\ x^{(k+1)} &= z^{(k)} - \left[3I - \frac{5}{2}F'(x^{(k)})^{-1}F'(y^{(k)}) + \frac{1}{2}(F'(x^{(k)})^{-1}F'(y^{(k)}))^2 \right] F'(x^{(k)})^{-1}F(z^{(k)}). \end{aligned} \tag{19}$$

Method HM₆:

$$\begin{aligned} y^{(k)} &= x^{(k)} - F'(x^{(k)})^{-1}F(x^{(k)}), \\ H(x^{(k)}, y^{(k)}) &= F'(x^{(k)})^{-1}F(y^{(k)}), \\ G_s(x^{(k)}, y^{(k)}) &= s_1I + s_2H(y^{(k)}, x^{(k)}) + s_3H(x^{(k)}, y^{(k)}) + s_4H(y^{(k)}, x^{(k)})^2, \\ z^{(k)} &= x^{(k)} - G_s(x^{(k)}, y^{(k)})F'(x^{(k)})^{-1}F(x^{(k)}), \\ x^{(k+1)} &= z^{(k)}. \end{aligned} \tag{20}$$

where $s_1, s_2, s_3,$ and s_4 are real numbers.

Method SM_6 :

$$\begin{aligned} y^{(k)} &= x^{(k)} - \frac{2}{3}F'(x^{(k)})^{-1}F(x^{(k)}), \\ z^{(k)} &= \phi_4^{(4)}(x^k, y^k), \\ x^{(k+1)} &= z^{(k)} - [sI + tF'(x^{(k)})^{-1}F'(y^{(k)})]F'(x^{(k)})^{-1}F(z^{(k)}), \end{aligned} \tag{21}$$

where s and t are real parameters.

Method WM_6 :

$$\begin{aligned} y^{(k)} &= x^{(k)} - F'(x^{(k)})^{-1}F(x^{(k)}), \\ z^{(k)} &= y^{(k)} - [2I - F'(x^{(k)})^{-1}F'(y^{(k)})]F'(x^{(k)})^{-1}F(y^{(k)}), \\ x^{(k+1)} &= z^{(k)} - [2I - F'(x^{(k)})^{-1}F'(y^{(k)})]F'(x^{(k)})^{-1}F(z^{(k)}). \end{aligned} \tag{22}$$

Method MM_6 :

$$\begin{aligned} y^{(k)} &= x^{(k)} - \frac{2}{3}F'(x^{(k)})^{-1}F(x^{(k)}), \\ z^{(k)} &= x^{(k)} - [I + \frac{1}{2\beta}(I - \frac{\lambda}{\beta}G(x^{(k)}))H(G(x^{(k)}))u(x^{(k)})], \\ x^{(k+1)} &= z^{(k)} - [pI + qG(x^{(k)})]F'(x^{(k)})^{-1}F(z^{(k)}), \end{aligned} \tag{23}$$

where $\lambda, \beta, p,$ and q are real numbers.

Method LM_6 :

$$\begin{aligned} y^{(k)} &= x^{(k)} - F'(x^{(k)})^{-1}F(x^{(k)}), \\ z^{(k)} &= x^{(k)} - 2(F'(x^{(k)}) + F'(y^{(k)})^{-1}F(x^{(k)})), \\ x^{(k+1)} &= z^{(k)} - [\frac{7}{2}I - 4F'(x^{(k)})^{-1}F'(y^{(k)}) + \frac{3}{2}F'(x^{(k)})^{-1}F'(y^{(k)})^2]F'(x^{(k)})^{-1}F(z^{(k)}). \end{aligned} \tag{24}$$

The abscissas t_j and the weights w_j are known and depicted in the Table 1 when $t = 8$ (for the more details please have a look at Example 1). In Tables 2–6, we mention the number of iteration indexes (n), residual errors ($\|F(x^{(k)})\|$), error $\|x^{(k+1)} - x^{(k)}\|$ and computational convergence order $\rho^* \approx \frac{\log [\|x^{(k+1)} - x^{(k)}\| / \|x^{(k)} - x^{(k-1)}\|]}{\log [\|x^{(k)} - x^{(k-1)}\| / \|x^{(k-1)} - x^{(k-2)}\|]}$. In addition, the value of η is the last calculated value of $\frac{\|x^{(k+1)} - x^{(k)}\|}{\|x^{(k)} - x^{(k-1)}\|^6}$. Finally, the comparison on the basis of number of iterations taken by different methods on numerical Examples 1–5 is also depicted in Table 7.

All the computations have been done with multiple precision arithmetic with 1000 digits of mantissa, which minimize round-off errors in Mathematica-9. Here, $a (\pm b)$ is $a \times 10^{(\pm b)}$ in all the tables. The stopping criteria for the programming is defined as follows,

$$(i) \|F(x^{(k)})\| < 10^{-100} \text{ and } (ii) \|x^{(k+1)} - x^{(k)}\| < 10^{-100}.$$

Example 1. Let us consider the Hammerstein integral equation (see [1], pp. 19–20) given as follows,

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

where $y \in C[0, 1], s, t \in [0, 1],$ and the kernel F is

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

Now, using the Gauss Legendre formula, we transform the above equation into a finite-dimensional problem, which is given as $\int_0^1 f(t)dt \simeq \sum_{j=1}^8 w_j f(t_j)$, where t_j and w_j denote abscissas and weights, respectively. Now t_j^s and w_j^s are determined for $t = 8$ by Gauss Legendre quadrature formula. Let us call $y(t_i)$ by $y_i (i = 1, 2, \dots, 8)$, then we get the following nonlinear system,

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

where $a_{ij} = \begin{cases} w_j t_j (1 - t_i), j \leq i, \\ w_j t_i (1 - t_j), i < j. \end{cases}$

Here, the abscissas t_j and the weights w_j are known and depicted in the Table 1 when $t = 8$. The convergence of the methods towards the root

$$\xi^* = (1.00209 \dots, 1.00990 \dots, 1.01972 \dots, 1.02643 \dots, 1.02643 \dots, 1.01972 \dots, 1.00990 \dots, 1.00209 \dots)^T,$$

is tested in the Table 2 on the choice of the initial guess $x^{(0)} = (\frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2})^T$. The numerical results show that the proposed methods $PM1_6$ and $PM2_6$ have better residual errors in comparison with the existing ones. In addition, smaller asymptotic error constants also belong to our methods $PM1_6$ and $PM2_6$.

Table 1. t_j^s and w_j^s of Gauss–Legendre 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...

Table 2. Comparison of methods on Hammerstein integral equation in Example 1.

Methods	k	$\ F(x^{(k)})\ $	$\ x^{(k+1)} - x^{(k)}\ $	ρ^*	$\frac{\ x^{(k+1)} - x^{(k)}\ }{\ x^{(k)} - x^{(k-1)}\ ^6}$
AM_6	1	1.1(−5)	2.4(−6)	5.9991	6.596956919(−6)
	2	5.4(−39)	1.2(−39)		
	3	8.0(−239)	1.7(−239)		
HM_6	1	3.0(−5)	6.5(−6)	4.9991	1.994799598
	2	6.9(−31)	1.5(−31)		
	3	4.4(−159)	9.4(−160)		
SM_6	1	9.4(−6)	2.0(−6)	5.9991	4.964844066(−6)
	2	1.5(−39)	3.2(−40)		
	3	2.7(−242)	5.7(−243)		

Table 2. Cont.

Methods	k	$\ F(x^{(k)})\ $	$\ x^{(k+1)} - x^{(k)}\ $	ρ^*	$\frac{\ x^{(k+1)} - x^{(k)}\ }{\ x^{(k)} - x^{(k-1)}\ ^6}$
WM ₆	1	1.6(−6)	3.3(−7)		
	2	9.0(−45)	1.9(−45)		1.377597868(−6)
	3	3.3(−274)	7.1(−275)	5.9996	1.431074607(−6)
MM ₆	1	3.9(−6)	8.2(−7)		
	2	5.8(−36)	1.2(−36)		3.943508142
	3	4.6(−185)	9.9(−186)	4.9992	2.773373608(+30)
LM ₆	1	8.6(−6)	1.8(−6)		
	2	8.4(−37)	1.7(−37)		4.445532921(−3)
	3	1.4(−189)	3.0(−190)	4.9224	1.232404905(+31)
PM1 ₆	1	1.7(−7)	1.5(−7)		
	2	1.9(−47)	4.1(−48)		3.291248720(−7)
	3	7.6(−291)	1.6(−291)	5.9998	3.358306951(−7)
PM2 ₆	1	6.6(−7)	1.4(−7)		
	2	1.1(−47)	2.3(−48)		2.820724919(−7)
	3	2.0(−292)	4.3(−293)	5.9998	2.880288646(−7)

Example 2. Let us consider the Van der Pol equation [30], which governs the flow of current in a vacuum tube, defined as follows,

$$y'' - \mu(y^2 - 1)y' + y = 0, \mu > 0. \tag{25}$$

Here, boundary conditions are given by $y(0) = 0, y(2) = 1$. Further, we divide the given interval $[0, 2]$ as follows,

$$x_0 = 0 < x_1 < x_2 < x_3 < \dots < x_n, \text{ where } x_i = x_0 + ih, h = \frac{2}{n}.$$

Moreover, we assume that

$$y_i = y(x_i), \quad i = 0, 1, \dots, n.$$

Now, if we discretize the above problem (25) by using the finite-difference formula for the first and second derivatives, which are given by

$$y'_k = \frac{y_{k+1} - y_{k-1}}{2h}, \quad y''_k = \frac{y_{k-1} - 2y_k + y_{k+1}}{h^2}, \quad k = 1, 2, \dots, n - 1,$$

then, we obtain a SNEs of order $(n - 1) \times (n - 1)$.

$$2h^2y_k - h\mu \left(y_k^2 - 1 \right) (y_{k+1} - y_{k-1}) + 2(y_{k-1} + y_{k+1} - 2y_k) = 0, \quad k = 1, 2, \dots, n - 1.$$

Let us consider $\mu = \frac{1}{2}$ and initial approximation $y_k^{(0)} = \log\left(\frac{1}{k^2}\right), k = 1, 2, \dots, n - 1$. In this example, we solve 9×9 SNEs by taking $n = 10$. The solution of this problem is

$$\xi^* = (-0.4795\dots, -0.9050\dots, -1.287\dots, -1.641\dots, -1.990\dots, -2.366\dots, -2.845\dots, -3.673\dots, -6.867\dots)^T.$$

The numerical results are displayed in Table 3. It is found that the newly proposed methods perform better in all aspects, whereas the existing methods show larger residual errors.

Table 3. Comparisons of methods on Van der Pol equation in Example 2.

Methods	k	$\ F(x^{(k)})\ $	$\ x^{(k+1)} - x^{(k)}\ $	ρ^*	$\frac{\ x^{(k+1)} - x^{(k)}\ }{\ x^{(k)} - x^{(k-1)}\ ^6}$
AM ₆	1	9.8(+16)	1.2(+6)	1.0000	1.115653208(−29)
	2	9.3(+15)	5.3(+5)		
	3	8.9(+14)	2.4(+5)		
HM ₆	1	2.0(+2)	3.4(+1)	0.13787	1.044481693(−16)
	2	2.6(+0)	1.4(+3)		
	3	6.9(+7)	8.5(+2)		
SM ₆	1	4.7(+10)	9.4(+3)	0.99993	3.757585542(−19)
	2	3.9(+9)	4.1(+3)		
	3	3.3(+8)	1.8(+3)		
WM ₆	1	3.8(+10)	8.7(+3)	0.99992	5.216543771(−19)
	2	3.3(+9)	3.9(+3)		
	3	2.9(+8)	1.7(+3)		
MM ₆	1	1.8(+9)	3.2(+3)	0.99947	8.435924608(−17)
	2	1.4(+8)	1.4(+3)		
	3	1.1(+7)	6.0(+2)		
LM ₆	1	7.8(+3)	5.1(+1)	1.3904	9.718708834(−8)
	2	6.0(+2)	1.9(+1)		
	3	3.9(+1)	5.0(0)		
PM1 ₆	1	2.6(0)	7.2(−1)	5.0546	3.049589936
	2	1.5(−4)	2.8(−5)		
	3	6.7(−27)	1.6(−27)		
PM2 ₆	1	2.6(−1)	1.2(−1)	4.9654	7.413843150(+4)
	2	8.3(−9)	1.5(−9)		
	3	2.8(−48)	7.4(−49)		

Example 3. The 2D Bratu problem [31,32] is defined as

$$\begin{aligned}
 &u_{xx} + u_{tt} + Ce^u = 0, \text{ on} \\
 &\Omega : (x, t) \in 0 \leq x \leq 1, 0 \leq t \leq 1, \\
 &\text{with boundary conditions } u = 0 \text{ on } \Omega.
 \end{aligned}
 \tag{26}$$

Using finite difference discretization, a given nonlinear PDE can be reduced to a SNEs. Let $\Theta_{i,j} = u(x_i, t_j)$ be the numerical solution at the grid points of the mesh. Let M_1 and M_2 be the number of steps in x and t directions, respectively, and h and k be the respective step sizes. To solve the given PDE, we apply the central difference formula to u_{xx} and u_{tt} in the following way,

$$u_{xx}(x_i, t_j) = \frac{\Theta_{i+1,j} - 2\Theta_{i,j} + \Theta_{i-1,j}}{h^2}, \quad C = 0.1, \quad t \in [0, 1]
 \tag{27}$$

By using expression (27) in (26) and after some simplification, we have

$$\Theta_{i,j+1} + \Theta_{i,j-1} - \Theta_{i,j} + \Theta_{i+1,j} + \Theta_{i-1,j} + h^2 C \exp(\Theta_{i,j}) \quad i = 1, 2, 3, \dots, M_1, j = 1, 2, 3, \dots, M_2
 \tag{28}$$

By choosing $M_1 = M_2 = 11$, $C = 0.1$, and $h = \frac{1}{11}$, we obtain a system of nonlinear equations of size 10×10 , with the initial guess $x_0 = 0.1(\sin(\pi h)\sin(\pi k), \sin(2\pi h)\sin(2\pi k), \dots, \sin(10\pi h)\sin(10\pi k))^T$. Numerical estimations are given in Table 4. Numerical results demonstrate that the new methods have much improved error estimations and computational order of convergence in comparison to its competitors.

Table 4. Comparisons of different methods on 2D Bratu problem in Example 3.

Methods	k	$\ F(x^{(k)})\ $	$\ x^{(k+1)} - x^{(k)}\ $	ρ^*	$\frac{\ x^{(k+1)} - x^{(k)}\ }{\ x^{(k)} - x^{(k-1)}\ ^6}$
AM ₆	1	4.4(−15)	2.4(−14)	5.9994	1.428095547(−12)
	2	6.9(−95)	3.5(−94)		
	3	7.9(−574)	3.9(−573)		
HM ₆	1	2.1(−13)	1.2(−12)	4.9997	7.368055345(−11)
	2	2.1(−71)	1.2(−70)		
	3	1.7(−361)	9.3(−361)		
SM ₆	1	4.4(−15)	2.4(−14)	5.9994	1.433541371(−12)
	2	7.1(−95)	3.6(−94)		
	3	9.2(−574)	4.5(−573)		
WM ₆	1	8.1(−2)	5.0(−1)	5.9999	1.754949400(−16)
	2	5.0(−19)	2.9(−18)		
	3	1.7(−122)	1.0(−121)		
MM ₆	1	7.5(−15)	4.1(−14)	4.9997	1.375781477(+1)
	2	1.2(−80)	6.4(−80)		
	3	1.2(−409)	6.3(−409)		
LM ₆	1	6.4(−16)	2.5(−15)	4.9853	1.447342836(−13)
	2	2.9(−87)	5.3(−87)		
	3	9.1(−445)	3.3(−444)		
PM1 ₆	1	5.7(−18)	7.0(−18)	5.9999	5.229619528(−14)
	2	4.9(−117)	5.9(−117)		
	3	1.9(−711)	2.3(−711)		
PM2 ₆	1	5.7(−18)	6.9(−18)	5.9999	1.698178527(−6)
	2	4.7(−117)	5.6(−117)		
	3	1.4(−711)	1.6(−711)		

Example 4. Consider another typical nonlinear problem, that is, Fisher’s equation [33] with homogeneous Neumann’s BC’s, the diffusion coefficient H is

$$\begin{aligned}
 u_t &= Hu_{xx} + u(1 - u) = 0, \\
 u(x, 0) &= 1.5 + 0.5\cos(\pi x), \quad 0 \leq x \leq 1, \\
 u_x(0, t) &= 0, \quad \forall t \geq 0, \\
 u_x(1, t) &= 0, \quad \forall t \geq 0.
 \end{aligned}
 \tag{29}$$

Again using finite difference discretization, the equation (29) reduces to a SNEs. Consider $\Theta_{i,j} = u(x_i, t_j)$ be its approximate solution at the grid points of the mesh. Let M_1 and M_2 be the number of steps in x and t directions, and h and k be the respective step size. Applying central difference formula to u_{xx} , backward difference for $u_t(x_i, t_j)$, and forward difference for $u_x(x_i, t_j)$, respectively, in the following way,

$$\begin{aligned}
 u_{xx}(x_i, t_j) &= \frac{\Theta_{i+1,j} - 2\Theta_{i,j} + \Theta_{i-1,j}}{h^2}, \\
 u_t(x_i, t_j) &= \frac{\Theta_{i,j} - \Theta_{i,j-1}}{k} \\
 \text{and} \\
 u_x(x_i, t_j) &= \frac{\Theta_{i+1,j} - \Theta_{i,j}}{h},
 \end{aligned}
 \tag{30}$$

where $h = \frac{1}{M_1}, k = \frac{1}{M_2}, t \in [0, 1]$.

By adopting expression (30) in (29), after some simplification, we get

$$\frac{\Theta_{1,j} - \Theta_{i,j-1}}{k} - \Theta_{i,j} (1 - \Theta_{i,j}) - H \frac{\Theta_{i+1,j} - 2\Theta_{i,j} + \Theta_{i-1,j}}{h^2}, \quad i = 1, 2, 3, \dots, M_1, j = 1, 2, 3, \dots, M_2 \quad (31)$$

For the system of nonlinear equations, we considered $M_1 = M_2 = 5, h = \frac{1}{5}, k = \frac{1}{5}$ and $H = 1$, which reduces to nonlinear system of size 5×5 , with the initial guess $x_0 = (1 + \frac{i}{25})^T, i = 1, 2, \dots, 25$. All the numerical results are shown in Table 5. Numerical results show that our methods have better computational efficiency than the already existing schemes, in terms of residual errors and difference between consecutive approximations.

Table 5. Comparisons of different methods on Fisher’s equation in Example 4.

Methods	k	$\ F(x^{(k)})\ $	$\ x^{(k+1)} - x^{(k)}\ $	ρ^*	$\frac{\ x^{(k+1)} - x^{(k)}\ }{\ x^{(k)} - x^{(k-1)}\ ^6}$
AM ₆	1	6.9(−3)	1.5(−3)		
	2	4.2(−21)	6.5(−22)		7.021836209(−5)
	3	4.5(−131)	7.0(−132)	5.9941	9.022505886(−5)
HM ₆	1	6.0(−3)	1.3(−3)		
	2	1.3(−18)	2.0(−19)		5.236602433(−2)
	3	1.8(−97)	2.8(−98)	4.9940	4.016099735(+14)
SM ₆	1	4.8(−3)	1.0(−3)		
	2	2.9(−22)	4.5(−23)		4.391559871(−5)
	3	3.0(−138)	4.7(−139)	5.9945	5.610610203(−5)
WM ₆	1	4.8(−3)	1.0(−3)		
	2	2.9(−22)	4.5(−23)		4.391559871(−5)
	3	3.0(−138)	4.7(−139)	5.9945	5.610610203(−5)
MM ₆	1	2.7(−3)	5.5(−4)		
	2	9.5(−21)	1.5(−21)		5.507288873(−2)
	3	1.6(−108)	2.5(−109)	4.9964	2.350947999(+16)
LM ₆	1	4.4(−3)	1.0(−3)		
	2	4.6(−20)	7.4(−21)		5.815894917(−3)
	3	2.2(−107)	1.2(−108)	5.1204	7.024389352(+12)
PM1 ₆	1	5.7(−18)	7.0(−18)		
	2	4.9(−117)	5.9(−117)		5.229619528(−14)
	3	1.9(−711)	2.3(−711)	5.9999	5.367043263(−14)
PM2 ₆	1	5.7(−18)	6.9(−18)		
	2	4.7(−117)	5.6(−117)		5.194329469(−14)
	3	1.4(−711)	1.6(−711)	5.9999	5.331099808(−14)

Example 5. Let us consider the following nonlinear system,

$$F(X) = \begin{cases} x_j^2 x_{j+1} - 1 = 0, & 1 \leq j \leq n - 1, \\ x_n^2 x_1 - 1 = 0. \end{cases} \quad (32)$$

To obtain a large SNEs 200×200 , we choose $n = 200$ and the initial approximation $x^{(0)} = (1.25, 1.25, 1.25, \dots, 1.25(200\text{times}))^T$ for this problem. The required solution of this problem is $\xi^* = (1, 1, 1, \dots, 1(200\text{times}))^T$. The obtained results can be observed in Table 6. It can be easily seen that the proposed scheme performs well; in this case, in terms of error estimates as compared to the available methods of the same nature.

Table 6. Comparisons of different methods on Example 5.

Methods	k	$\ F(x^{(k)})\ $	$\ x^{(k+1)} - x^{(k)}\ $	ρ^*	$\frac{\ x^{(k+1)} - x^{(k)}\ }{\ x^{(k)} - x^{(k-1)}\ ^6}$
AM_6	1	5.5(−2)	1.8(−2)	5.9993	7.599151595(−5)
	2	8.2(−15)	2.7(−15)		
	3	9.5(−92)	3.2(−92)		
HM_6	1	3.0(−2)	1.0(−2)	4.9997	6.625126396(−3)
	2	2.0(−14)	6.7(−15)		
	3	2.7(−75)	8.9(−76)		
SM_6	1	3.9(−2)	1.3(−2)	6.0000	4.636076578(−5)
	2	6.4(−16)	2.1(−16)		
	3	1.3(−98)	4.3(−99)		
WM	1	4.2(−2)	1.4(−2)	5.9995	5.254428593(−5)
	2	1.2(−15)	3.9(−16)		
	3	5.7(−97)	1.9(−97)		
MM_6	1	2.7(−2)	8.9(−3)	4.9993	3.463613165(−3)
	2	5.3(−15)	1.8(−15)		
	3	1.6(−78)	5.4(−79)		
LM_6	1	3.4(−2)	1.1(−2)	5.9996	3.721130070(−5)
	2	2.3(−16)	7.7(−17)		
	3	2.3(−101)	7.6(−102)		
$PM1_6$	1	5.5(−3)	1.8(−3)	6.0000	2.944381059(−6)
	2	3.4(−22)	1.1(−22)		
	3	1.8(−137)	5.9(−138)		
$PM2_6$	1	2.8(−3)	9.5(−4)	6.0000	1.178221976(−6)
	2	2.5(−24)	8.5(−25)		
	3	1.3(−150)	4.4(−151)		

Table 7. Number of iterations taken by different methods on Examples 1–5.

Methods	Ex.1	Ex.2	Ex.3	Ex.4	Ex. 5	Total	Average
AM_6	3	20	3	3	4	33	6.6
HM_6	3	D	3	3	4	13 *	3.25 *
SM_6	3	13	3	3	4	26	5.2
WM_6	3	13	2	3	4	25	5
MM_6	3	12	3	3	4	25	5
LM_6	3	7	3	3	3	19	3.8
$PM1_6$	3	4	2	2	3	14	2.8
$PM2_6$	3	4	2	2	3	14	2.8

* means, the total number of iteration calculated on all examples except Example 2, because HM_6 is divergent in Example 2.

4. Conclusions

In this work, we have developed new family of sixth-order iterative methods for solving SNEs, numerically. To check their effectiveness, the proposed scheme is applied on some large-scale systems arising from various academic problems. Further, the numerical results show that the proposed methods perform better than already existing schemes in the scientific literature.

Author Contributions: R.B. and M.K.: Conceptualization; Methodology; Validation; Writing–Original Draft Preparation; Writing–Review & Editing. H.F.A.: Review & Editing.

Funding: Deanship of Scientific Research (DSR), King Abdulaziz University, Jeddah, under grant No. G-1349-665-1440.

Acknowledgments: This project was supported by the Deanship of Scientific Research (DSR), King Abdulaziz University, Jeddah, under grant No. G-1349-665-1440. The authors, therefore, gratefully acknowledge with thanks DSR for technical and financial support.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Ortega J.M.; Rheinboldt, W.C. *Iterative Solution of Nonlinear Equations in Several Variables*; Academic Press: New York, NY, USA, 1970.
2. Cordero, A.; Torregrosa, J.R. Variants of Newton's method for functions of several variables. *Appl. Math. Comput.* **2006**, *183*, 199–208. [[CrossRef](#)]
3. Frontini M.; Sormani, E. Third-order methods from quadrature formulae for solving systems of nonlinear equations. *Appl. Math. Comput.* **2004**, *149*, 771–782. [[CrossRef](#)]
4. Grau-Sánchez, M.; Grau, À; Noguera, M. On the computational efficiency index and some iterative methods for solving systems of non-linear equations. *Comput. Appl. Math.* **2011**, *236*, 1259–1266.
5. Homeier, H.H.H. A modified Newton method with cubic convergence: The multivariable case. *Comput. Appl. Math.* **2004**, *169*, 161–169. [[CrossRef](#)]
6. Cordero, A.; Torregrosa, J.R. Variants of Newton's method using fifth-order quadrature formulas. *Appl. Math. Comput.* **2007**, *190*, 686–698. [[CrossRef](#)]
7. Darvishi, M.T.; Barati, A. Super cubic iterative methods to solve systems of nonlinear equations. *Appl. Math. Comput.* **2007**, *188*, 1678–1685. [[CrossRef](#)]
8. Darvishi, M.T.; Barati, A. A third-order Newton-type method to solve systems of non-linear equations. *Appl. Math. Comput.* **2007**, *187*, 630–635.
9. Potra, F.A.; Pták, V. *Nondiscrete Induction and Iterative Processes*; Pitman Publishing: Boston, MA, USA, 1984.
10. Babajee, D.K.R.; Madhu, K.; Jayaraman, J. On some improved harmonic mean Newton-like methods for solving systems of nonlinear equations. *Algorithms* **2015**, *8*, 895–909. [[CrossRef](#)]
11. Cordero, A.; Martínez, E.; Torregrosa, J.R. Iterative methods of order four and five for systems of nonlinear equations. *Comput. Appl. Math.* **2009**, *231*, 541–551. [[CrossRef](#)]
12. Cordero, A.; Hueso, J.L.; Martínez, E.; Torregrosa, J.R. Efficient high-order methods based on golden ratio for nonlinear systems. *Appl. Math. Comput.* **2011**, *217*, 4548–4556. [[CrossRef](#)]
13. Arroyo, V.; Cordero, A.; Torregrosa, J.R. Approximation of artificial satellites' preliminary orbits: The efficiency challenge. *Math. Comput. Modell.* **2011**, *54*, 1802–1807. [[CrossRef](#)]
14. Alzahrani, A.K.H.; Behl, R.; Alshomrani, A. Some higher-order iteration functions for solving nonlinear models. *Appl. Math. Comput.* **2018**, *334*, 80–93. [[CrossRef](#)]
15. Narang, M.; Bhatia, S.; Kanwar, V. New two parameter Chebyshev-Halley like family of fourth and sixth-order methods for systems of nonlinear equations. *Appl. Math. Comput.* **2016**, *275*, 394–403. [[CrossRef](#)]
16. Lotfi, T.; Bakhtiari, P.; Cordero, A.; Mahdiani, K.; Torregrosa, J.R. Some new efficient multipoint iterative methods for solving nonlinear systems of equations. *Int. J. Comput. Math.* **2015**, *92*, 1921–1934. [[CrossRef](#)]
17. Babajee, D.K.R. On a two-parameter Chebyshev-Halley like family of optimal two-point fourth order methods free from second derivatives. *Afrika Matematika* **2015**, *26*, 689–695. [[CrossRef](#)]
18. Campos, B.; Cordero, A.; Torregrosa, J.R.; Vindel, P. Stability of King's family of iterative methods with memory. *J. Comput. Appl. Math.* **2017**, *318*, 504–514. [[CrossRef](#)]
19. Cordero, A.; Maimó, J.G.; Torregrosa, J.R.; Vassileva, M.P.; Vindel, P. Chaos in King's iterative family. *Appl. Math. Lett.* **2013**, *26*, 842–848. [[CrossRef](#)]
20. Chicharro, F.; Cordero, A.; Torregrosa, J.R. Dynamics of iterative families with memory based on weight functions procedure. *J. Comput. Appl. Math.* **2019**, *354*, 286–298. [[CrossRef](#)]
21. King, R.F. A family of fourth-order methods for nonlinear equations. *SIAM J. Numer. Anal.* **1973**, *10*, 876–879. [[CrossRef](#)]
22. Argyros, I.K. *Convergence and Application of Newton-Type Iterations*; Springer: Berlin/Heidelberg, Germany, 2008.
23. Petković, M.S.; Neta, B.; Petković, L.D.; Džunić, J. *Multipoint Methods for Solving Nonlinear Equations*; Academic Press: Amsterdam, The Netherlands, 2012.
24. Traub, J.F. *Iterative Methods for the Solution of Equations*; Prentice-Hall: Englewood Cliffs, NJ, USA, 1964.

25. Cordero, A.; Hueso, J.L.; Martínez, E.; Torregrosa, J.R. A modified Newton-Jarratt's composition. *Numer. Algor.* **2010**, *55*, 87–99. [[CrossRef](#)]
26. Abbasbandy, S.; Bakhtiari, P.; Cordero, A.; Torregrosa, J.R.; Lotfi, T. New efficient methods for solving nonlinear systems of equations with arbitrary even order. *Appl. Math. Comput.* **2016**, *287*, 287–288. [[CrossRef](#)]
27. Hueso, J.L.; Martínez, E.; Teruel, C. Convergence, efficiency and dynamics of new fourth and sixth order families of iterative methods for nonlinear systems. *J. Comput. Appl. Math.* **2015**, *275*, 412–420. [[CrossRef](#)]
28. Sharma, J.R.; Arora, H. Efficient Jarratt-like methods for solving systems of nonlinear equations. *Calcolo* **2014**, *51*, 193–210. [[CrossRef](#)]
29. Wang, X.; Li, Y. An Efficient Sixth Order Newton Type Method for Solving Nonlinear Systems. *Algorithms* **2017**, *10*, 45. [[CrossRef](#)]
30. Burden, R.L.; Faires, J.D. *Numerical Analysis*; PWS Publishing Company: Boston, MA, USA, 2001.
31. Simpson, R.B. A method for the numerical determination of bifurcation states of nonlinear systems of equations. *SIAM J. Numer. Anal.* **1975**, *12*, 439–451. [[CrossRef](#)]
32. Kapania, R.K. A pseudo-spectral solution of 2-parameter Bratu's equation. *Comput. Mech.* **1990**, *6*, 55–63. [[CrossRef](#)]
33. Sauer, T. *Numerical Analysis*, 2nd ed.; Pearson: Upper Saddle River, NJ, USA, 2012.



© 2019 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).