

Document downloaded from:

<http://hdl.handle.net/10251/65553>

This paper must be cited as:

Lotfi, T.; Bakhtiari, P.; Cordero Barbero, A.; Mahdiani, K.; Torregrosa Sánchez, JR. (2015). Some new efficient multipoint iterative methods for solving nonlinear systems of equations. *International Journal of Computer Mathematics*. 92(9):1921-1934. doi:10.1080/00207160.2014.946412.



The final publication is available at

<http://dx.doi.org/10.1080/00207160.2014.946412>

Copyright Taylor & Francis: STM, Behavioural Science and Public Health Titles

Additional Information

Some new efficient multipoint iterative methods for solving nonlinear systems of equations *

Taher Lotfi ^{a, †} Parisa Bakhtiari^{b, ‡} Alicia Cordero ^{c, §} Katayoun Mahdiani^{a, ¶}
Juan R. Torregrosa^{a||}

^a Department of Applied Mathematics, Hamedan branch, Islamic Azad University, Hamedan, Iran

^b Young Researchers Club, Islamic Azad University, Hamedan Branch, Hamedan, Iran

^c Instituto de Matemática Multidisciplinar, Universitat Politècnica de València, Camino de Vera, s/n, 46022 Valencia, Spain

Abstract

It is attempted to put forward a new multipoint iterative method of sixth-order convergence for approximating solutions of nonlinear systems of equations. It requires two vector-function and two Jacobian matrices per iteration. Furthermore, we use it as a predictor to derive a general multipoint method. Convergence error analysis, estimating computational complexity, numerical implementation and comparisons are given to verify applicability and validity for the proposed methods.

Key Words: Nonlinear systems; multipoint, Matrix; LU factorization; Computational complexity

1 Introduction

The solution of equations is a venerable subject in science and engineering and of particular importance in applications. In the light of this fact, there have been constructed enormous amount of iterative methods for solving scalar nonlinear equations [1]. On the other hand, it must be noted that many of this methods cannot be extended to their corresponding systems. Even if this is possible, some crucial factors must be considered. Therefore, there are few practical iterative methods in this case. Moreover, it is worth mentioning that although some of the scalar iteration can be extended, however, due to increasing in computational complexity, they have no practical interest. This matter has been discussed in [5, 14, 18] thoroughly and one can consult them.

Let the function $\mathbf{F} : \mathbf{D} \subset \mathbb{R}^n \rightarrow \mathbb{R}^n$ has at least, second-order Frechet derivatives with continuity on an open set \mathbf{D} . Suppose that the equation $\mathbf{F}(\mathbf{x}) = 0$ has a solution $\mathbf{x}^* \in \mathbf{D}$, that is $\mathbf{F}(\mathbf{x}^*) = 0$, where $\mathbf{F}(\mathbf{x}) = (f_1(\mathbf{x}), f_2(\mathbf{x}), \dots, f_n(\mathbf{x}))^T$, $\mathbf{x} = (x_1, x_2, \dots, x_n)^T$, and $f_i(\mathbf{x})$, $i = 1, 2, \dots, n$, are real-valued functions.

It is widely known that the Newton's method in several variables [2] could be written as

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} - \mathbf{F}'(\mathbf{x}^{(k)})^{-1} \mathbf{F}(\mathbf{x}^{(k)}), \quad k = 0, 1, 2, \dots, \quad (1.1)$$

where $\mathbf{x}^{(0)}$ is the initial estimate and $\mathbf{F}'(\mathbf{x}^{(k)})$ is the Jacobian matrix of the function \mathbf{F} evaluated in the k th iteration. This method has order of convergence two under certain conditions. Some other one point methods like Chebyshev and Halley [3, 4] have been extended to their corresponding system versions with order of convergence three, but they use the first and second Frechet derivatives involved n^2 and n^3 function evaluations, respectively. On the other hand, it has been considerable attempts to derive methods free from second Frechet derivative having order of convergence three which most of them are not one step methods, and can be consulted in depth in [5]-[12] and references therein. There are few other mutipoint methods for approximating system solutions of nonlinear equations.

*This research was supported by Islamic Azad University- Hamedan branch and Ministerio de Ciencia y Tecnología MTM2011-28636-C02-02.

†Corresponding author: lotfi@iauh.ac.ir, lotfitaher@yahoo.com

‡bakhtiari@iauh.ac.ir

§acordero@mat.upv.es

¶mahdiani@iauh.ac.ir

||jrtorre@mat.upv.es

Another famous scheme for solving nonlinear systems of equations is Jarratt's fourth-order method [14] which is the generalization of the scheme in the scalar case given in [13] as follows

$$\begin{cases} \mathbf{y}^{(k)} = \mathbf{x}^{(k)} - \frac{2}{3}\mathbf{F}'(\mathbf{x}^{(k)})^{-1}\mathbf{F}(\mathbf{x}^{(k)}), \\ \mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} - \frac{1}{2}\left[(3\mathbf{F}'(\mathbf{y}^{(k)}) - \mathbf{F}'(\mathbf{x}^{(k)}))^{-1}(3\mathbf{F}'(\mathbf{y}^{(k)}) + \mathbf{F}'(\mathbf{x}^{(k)}))\right] \\ \quad \times \mathbf{F}'(\mathbf{x}^{(k)})^{-1}\mathbf{F}(\mathbf{x}^{(k)}). \end{cases} \quad (1.2)$$

Sharma et al. [15] constructed, composed of two weighted Newton step, the following fourth-order method

$$\begin{cases} \mathbf{y}^{(k)} = \mathbf{x}^{(k)} - \frac{2}{3}\mathbf{F}'(\mathbf{x}^{(k)})^{-1}\mathbf{F}(\mathbf{x}^{(k)}), \\ \mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} - \frac{1}{2}\left[-\mathbf{I} + \frac{9}{4}\mathbf{F}'(\mathbf{y}^{(k)})^{-1}\mathbf{F}'(\mathbf{x}^{(k)}) + \frac{3}{4}\mathbf{F}'(\mathbf{x}^{(k)})^{-1}\mathbf{F}'(\mathbf{y}^{(k)})\right] \\ \quad \times \mathbf{F}'(\mathbf{x}^{(k)})^{-1}\mathbf{F}(\mathbf{x}^{(k)}). \end{cases} \quad (1.3)$$

Another fourth-order Jarratt-type method has been devised by Babajee et al. [16] as following

$$\begin{cases} \mathbf{y}^{(k)} = \mathbf{x}^{(k)} - \frac{2}{3}\mathbf{F}'(\mathbf{x}^{(k)})^{-1}\mathbf{F}(\mathbf{x}^{(k)}), \\ \mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} - 2\left[\mathbf{I} - \frac{1}{4}\left(\mathbf{F}'(\mathbf{x}^{(k)})^{-1}\mathbf{F}'(\mathbf{y}^{(k)}) - \mathbf{I}\right) + \frac{3}{4}\left(\mathbf{F}'(\mathbf{x}^{(k)})^{-1}\mathbf{F}'(\mathbf{y}^{(k)}) - \mathbf{I}\right)^2\right] \\ \quad \times \left(\mathbf{F}'(\mathbf{x}^{(k)}) + \mathbf{F}'(\mathbf{y}^{(k)})\right)^{-1}\mathbf{F}(\mathbf{x}^{(k)}). \end{cases} \quad (1.4)$$

Abad et al. [17], in addition to create two fourth and fifth-order methods for solving nonlinear systems, provide a conjecture relative to construction optimal multipoint methods for nonlinear systems of equations: *Given a multipoint iterative method to solve nonlinear systems of equations which requires $d = k_1 + k_2$ functional evaluations per step such that k_1 of them correspond to the functional evaluations of the Jacobian matrix and k_2 to evaluations of the nonlinear function. Then optimal order for this method is $2^{k_1+k_2-1}$ if $k_1 \leq k_2$.* It should be remarked, although this conjecture is an extension of Kung and Traub conjecture for multipoint without memory methods in the scalar case, however, computational factors for iterative methods for solving nonlinear systems of equations are of considerable differences.

Note that although most of the works emphasizes on the numerical aspects of these iterations, there are two general ways for pursuing this aim analytically. One is based on the well-known n -dimensional Taylor expansion [14, 19] and second is based on the matrix approach, which is so-called as Point of Attraction, introduced first in [2]. We here apply the first case by reminding the following:

Let $\mathbf{F} : \mathbf{D} \subseteq \mathbb{R}^n \rightarrow \mathbb{R}^n$ be sufficiently Frechet differentiable in \mathbf{D} . By using the notation introduced in [14], the q th derivative of \mathbf{F} at $\mathbf{u} \in \mathbb{R}^n$, $q \geq 1$, is the q -linear function $\mathbf{F}^{(q)}(\mathbf{u}) : \mathbb{R}^n \times \dots \times \mathbb{R}^n \rightarrow \mathbb{R}^n$ such that $\mathbf{F}^{(q)}(\mathbf{u})(v_1, \dots, v_q) \in \mathbb{R}^n$. It is well known that, for $\mathbf{x}^* + \mathbf{h} \in \mathbb{R}^n$ lying in a neighborhood of a solution \mathbf{x}^* of the nonlinear system $\mathbf{F}(\mathbf{x}) = 0$, Taylor's expansion can be applied and we have

$$\mathbf{F}(\mathbf{x}^* + \mathbf{h}) = \mathbf{F}'(\mathbf{x}^*) \left[\mathbf{h} + \sum_{q=2}^{p-1} \mathbf{C}_q \mathbf{h}^q \right] + O(\mathbf{h}^p), \quad (1.5)$$

where $\mathbf{C}_q = (1/q!)[\mathbf{F}'(\mathbf{x}^*)]^{-1}\mathbf{F}^{(q)}(\mathbf{x}^*)$, $q \geq 2$. We observe that $\mathbf{C}_q \mathbf{h}^q \in \mathbb{R}^n$ since $\mathbf{F}^{(q)}(\mathbf{x}^*) \in \mathcal{L}(\mathbb{R}^n \times \dots \times \mathbb{R}^n, \mathbb{R}^n)$ and $[\mathbf{F}'(\mathbf{x}^*)]^{-1} \in \mathcal{L}(\mathbb{R}^n)$. In addition, we can express \mathbf{F}' as

$$\mathbf{F}'(\mathbf{x}^* + \mathbf{h}) = \mathbf{F}'(\mathbf{x}^*) \left[\mathbf{I} + \sum_{q=2}^{p-1} q\mathbf{C}_q \mathbf{h}^{q-1} \right] + O(\mathbf{h}^p), \quad (1.6)$$

wherein \mathbf{I} is the identity matrix, and $q\mathbf{C}_q \mathbf{h}^{q-1} \in \mathcal{L}(\mathbb{R}^n)$. Note that in what follows, $\mathbf{e}^{(k)} = \mathbf{x}^{(k)} - \mathbf{x}^*$ is the error in the k th iteration and $\mathbf{e}^{(k+1)} = L\mathbf{e}^{(k)p} + O(\mathbf{e}^{(k)p+1})$ is the error equation, where L is a p -linear function, i.e. $L \in \mathcal{L}(\mathbb{R}^n, \mathbb{R}^n, \dots, \mathbb{R}^n)$ and p is the order of convergence. Observe that $\mathbf{e}^{(k)p} = (\mathbf{e}^{(k)}, \mathbf{e}^{(k)}, \dots, \mathbf{e}^{(k)})$.

In this work, based on results in the scalar case in [20], we extend a new variant of Weerakoon and Fernando's method with sixth-order convergence in the multidimensional case. It uses two vector-function and two Jacobian matrices per iteration. Furthermore, we drive a general procedure in such a way that it uses this sixth-order as its predictor using only new one vector-function evaluation in each step. This is the most advantageous of this paper. In other words, when we add another step, the order of convergence increases by three units.

The paper is organized as follows. In the next section, extension and development of Weerakoon and Fernando's method, [20], for solving nonlinear systems of equations are derived. Furthermore, analysis of convergence order is established. In Section 3, our primary goal is to develop the extended method in Section 2. Indeed, we derive the ninth- and twelfth- order methods and then the general multipoint iteration is provided along with some relevant discussion. In Section 4, computational complexities are discussed. Numerical test problems and comparisons are illustrated in Section 5. The last section includes some conclusions.

2 Extension and Development

We here extend and develop a new method for solving nonlinear systems of equations described by

$$\begin{cases} \mathbf{y}^{(k)} = \mathbf{x}^{(k)} - \mathbf{F}'(\mathbf{x}^{(k)})^{-1} \mathbf{F}(\mathbf{x}^{(k)}), \\ \mathbf{z}^{(k)} = \mathbf{x}^{(k)} - 2(\mathbf{F}'(\mathbf{x}^{(k)}) + \mathbf{F}'(\mathbf{y}^{(k)}))^{-1} \mathbf{F}(\mathbf{x}^{(k)}), \\ \mathbf{x}^{(k+1)} = \mathbf{z}^{(k)} - \left(\frac{7}{2} \mathbf{I} - 4\mathbf{F}'(\mathbf{x}^{(k)})^{-1} \mathbf{F}'(\mathbf{y}^{(k)}) + \frac{3}{2} (\mathbf{F}'(\mathbf{x}^{(k)})^{-1} \mathbf{F}'(\mathbf{y}^{(k)}))^2 \right) \\ \quad \mathbf{F}'(\mathbf{x}^{(k)})^{-1} \mathbf{F}(\mathbf{z}^{(k)}). \end{cases} \quad (2.1)$$

Needless to say, that its first two steps are extensions for Weerakoon and Fernando's method [20] for solving nonlinear systems of equations. Furthermore, the third step has been built in such a way that no additional computations of any Jacobian matrices is required so that it speeds up the convergence order from three to six. In order to analyze the convergence order of the proposed method (2.1) we address the following theorem.

Theorem 2.1. *Let $\mathbf{F} : \mathbf{D} \subseteq \mathbb{R}^n \rightarrow \mathbb{R}^n$ be sufficiently Frechet differentiable at each point of an open convex neighborhood \mathbf{D} of $\mathbf{x}^* \in \mathbb{R}^n$, that is a solution of the system $\mathbf{F}(\mathbf{x}) = 0$. Let us suppose that $\mathbf{F}'(\mathbf{x})$ is continuous and nonsingular in \mathbf{x}^* . Then, the sequence $\{\mathbf{x}^{(k)}\}_{k \geq 0}$ obtained using the iterative method (2.1) converges to \mathbf{x}^* with convergence rate six.*

Proof. By (1.5) and (1.6), we get

$$\mathbf{F}(\mathbf{x}^{(k)}) = \mathbf{F}'(\mathbf{x}^*)[\mathbf{e}^{(k)} + \mathbf{C}_2 \mathbf{e}^{(k)2} + \mathbf{C}_3 \mathbf{e}^{(k)3} + \mathbf{C}_4 \mathbf{e}^{(k)4} + \mathbf{C}_5 \mathbf{e}^{(k)5} + \mathbf{C}_6 \mathbf{e}^{(k)6}] + O(\mathbf{e}^{(k)7}), \quad (2.2)$$

and

$$\mathbf{F}'(\mathbf{x}^{(k)}) = \mathbf{F}'(\mathbf{x}^*)[\mathbf{I} + 2\mathbf{C}_2 \mathbf{e}^{(k)} + 3\mathbf{C}_3 \mathbf{e}^{(k)2} + 4\mathbf{C}_4 \mathbf{e}^{(k)3} + 5\mathbf{C}_5 \mathbf{e}^{(k)4} + 6\mathbf{C}_6 \mathbf{e}^{(k)5}] + O(\mathbf{e}^{(k)6}), \quad (2.3)$$

where $\mathbf{C}_k = (1/k!)[\mathbf{F}'(\mathbf{x}^*)]^{-1} \mathbf{F}^{(k)}(\mathbf{x}^*)$, $k = 2, 3, \dots$. From (2.3), we have

$$[\mathbf{F}'(\mathbf{x}^{(k)})]^{-1} = \left[\mathbf{I} + \mathbf{X}_2 \mathbf{e}^{(k)} + \mathbf{X}_3 \mathbf{e}^{(k)2} + \mathbf{X}_4 \mathbf{e}^{(k)3} + \dots + \mathbf{X}_7 \mathbf{e}^{(k)6} \right] [\mathbf{F}'(\mathbf{x}^*)]^{-1} + O(\mathbf{e}^{(k)7}), \quad (2.4)$$

where $\mathbf{X}_1 = \mathbf{I}$, $\mathbf{X}_2 = -2\mathbf{C}_2$, $\mathbf{X}_3 = 4\mathbf{C}_2^2 - 3\mathbf{C}_3$, $\mathbf{X}_4 = -8\mathbf{C}_2^3 + 6\mathbf{C}_2\mathbf{C}_3 + 6\mathbf{C}_3\mathbf{C}_2 - 4\mathbf{C}_4$; in general, $\mathbf{X}_s = \sum_{j=2}^s j \mathbf{X}_{s-j+1} \mathbf{C}_j$, for $s = 1, 2, \dots$.

Then, by (2.2) and (2.4), we have

$$\begin{aligned} \mathbf{F}(\mathbf{x}^{(k)})^{-1} \mathbf{F}(\mathbf{x}^{(k)}) &= \mathbf{e}^{(k)} - \mathbf{C}_2 \mathbf{e}^{(k)2} + 2(\mathbf{C}_2^2 - \mathbf{C}_3) \mathbf{e}^{(k)3} \\ &\quad + (-4\mathbf{C}_2^3 + 4\mathbf{C}_2\mathbf{C}_3 + 3\mathbf{C}_3\mathbf{C}_2 - 3\mathbf{C}_4) \mathbf{e}^{(k)4} \\ &\quad + (8\mathbf{C}_2^4 + 5\mathbf{C}_2\mathbf{C}_4 - 10\mathbf{C}_2^2\mathbf{C}_3 + 6\mathbf{C}_3^2 - 10\mathbf{C}_3\mathbf{C}_2^2 + 5\mathbf{C}_4\mathbf{C}_2 - 4\mathbf{C}_5) \mathbf{e}^{(k)5} \\ &\quad + \left(-16\mathbf{C}_2^5 + 26\mathbf{C}_2^3\mathbf{C}_3 - 14\mathbf{C}_2^2\mathbf{C}_4 - 17\mathbf{C}_2\mathbf{C}_3^2 + 9\mathbf{C}_3\mathbf{C}_4 + 8\mathbf{C}_4\mathbf{C}_3 \right. \\ &\quad \left. + 7\mathbf{C}_5\mathbf{C}_2 + 26\mathbf{C}_3\mathbf{C}_2^3 - 14\mathbf{C}_4\mathbf{C}_2^2 - 16\mathbf{C}_3^2\mathbf{C}_2 + 6\mathbf{C}_2\mathbf{C}_5 - 5\mathbf{C}_6 \right) \mathbf{e}^{(k)6} + O(\mathbf{e}^{(k)7}). \end{aligned} \quad (2.5)$$

By (2.5), the first step of (2.1) induces

$$\begin{aligned} \mathbf{y}^{(k)} - \mathbf{x}^* &= \mathbf{C}_2 \mathbf{e}^{(k)2} + (-2\mathbf{C}_2^2 + 2\mathbf{C}_3) \mathbf{e}^{(k)3} \\ &\quad + (4\mathbf{C}_2^3 - 4\mathbf{C}_2\mathbf{C}_3 - 3\mathbf{C}_3\mathbf{C}_2 + 3\mathbf{C}_4) \mathbf{e}^{(k)4} \\ &\quad + (-8\mathbf{C}_2^4 + 10\mathbf{C}_2^2\mathbf{C}_3 - 5\mathbf{C}_2\mathbf{C}_4 - 6\mathbf{C}_3^2 + 10\mathbf{C}_3\mathbf{C}_2^2 - 5\mathbf{C}_4\mathbf{C}_2 + 4\mathbf{C}_5) \mathbf{e}^{(k)5} \\ &\quad + \left(-16\mathbf{C}_2^5 + 26\mathbf{C}_2^3\mathbf{C}_3 - 14\mathbf{C}_2^2\mathbf{C}_4 - 17\mathbf{C}_2\mathbf{C}_3^2 + 9\mathbf{C}_3\mathbf{C}_4 + 8\mathbf{C}_4\mathbf{C}_3 \right. \\ &\quad \left. + 7\mathbf{C}_5\mathbf{C}_2 + 26\mathbf{C}_3\mathbf{C}_2^3 - 14\mathbf{C}_4\mathbf{C}_2^2 - 16\mathbf{C}_3^2\mathbf{C}_2 + 6\mathbf{C}_2\mathbf{C}_5 - 5\mathbf{C}_6 \right) \mathbf{e}^{(k)6} + O(\mathbf{e}^{(k)7}). \end{aligned} \quad (2.6)$$

The Taylor expansion of the Jacobian matrix $\mathbf{F}'(\mathbf{y}^{(k)})$ results

$$\begin{aligned} \mathbf{F}'(\mathbf{y}^{(k)}) &= \mathbf{F}'(\mathbf{x}^*) \left\{ \mathbf{I} + 2\mathbf{C}_2(\mathbf{y}^{(k)} - \mathbf{x}^*) + 3\mathbf{C}_3(\mathbf{y}^{(k)} - \mathbf{x}^*)^2 + 4\mathbf{C}_4(\mathbf{y}^{(k)} - \mathbf{x}^*)^3 \right. \\ &\quad \left. + 5\mathbf{C}_5(\mathbf{y}^{(k)} - \mathbf{x}^*)^4 + 6\mathbf{C}_6(\mathbf{y}^{(k)} - \mathbf{x}^*)^6 \right\} + O((\mathbf{y}^{(k)} - \mathbf{x}^*)^7). \end{aligned} \quad (2.7)$$

Taking into account (2.4), (2.6) and (2.7), we obtain

$$\begin{aligned} \mathbf{F}'(\mathbf{x}^{(k)}) + \mathbf{F}'(\mathbf{y}^{(k)}) &= \mathbf{F}'(\mathbf{x}^*) \left\{ 2\mathbf{I} + 2\mathbf{C}_2\mathbf{e}^{(k)} + (2\mathbf{C}_2^2 + 3\mathbf{C}_3)\mathbf{e}^{(k)2} \right. \\ &\quad + 4(-\mathbf{C}_2^3 + \mathbf{C}_2\mathbf{C}_3 + \mathbf{C}_4)\mathbf{e}^{(k)3} \\ &\quad + (8\mathbf{C}_2^4 - 11\mathbf{C}_2^2\mathbf{C}_3 + 3\mathbf{C}_2\mathbf{C}_4 + 3\mathbf{C}_4\mathbf{C}_2 + 5\mathbf{C}_5)\mathbf{e}^{(k)4} \\ &\quad + 2(-8\mathbf{C}_2^5 + 14\mathbf{C}_2^3\mathbf{C}_3 - 10\mathbf{C}_2^2\mathbf{C}_4 + 2\mathbf{C}_2\mathbf{C}_5 + 2\mathbf{C}_5\mathbf{C}_2 + 3\mathbf{C}_6)\mathbf{e}^{(k)5} \\ &\quad + 2(16\mathbf{C}_2^6 - 34\mathbf{C}_2^4\mathbf{C}_3 + 6\mathbf{C}_3^3 + 30\mathbf{C}_2^3\mathbf{C}_4 \\ &\quad \left. - 13\mathbf{C}_2^2\mathbf{C}_5 - 8\mathbf{C}_2\mathbf{C}_3\mathbf{C}_4 + 3\mathbf{C}_2\mathbf{C}_6 + 2\mathbf{C}_6\mathbf{C}_2) \right\} \mathbf{e}^{(k)6} + O(\mathbf{e}^{(k)7}). \end{aligned} \quad (2.8)$$

Therefore,

$$\begin{aligned} (\mathbf{F}'(\mathbf{x}^{(k)}) + \mathbf{F}'(\mathbf{y}^{(k)}))^{-1} &= \left\{ \frac{1}{2}\mathbf{I} - \frac{1}{2}\mathbf{C}_2\mathbf{e}^{(k)} - \frac{3}{4}\mathbf{C}_3\mathbf{e}^{(k)2} \right. \\ &\quad + \frac{1}{2}(3\mathbf{C}_2^3 + \mathbf{C}_2\mathbf{C}_3 - 2\mathbf{C}_4)\mathbf{e}^{(k)3} \\ &\quad + \frac{1}{8}(-36\mathbf{C}_2^4 + 32\mathbf{C}_2^2\mathbf{C}_3 + 9\mathbf{C}_3^2 + 2\mathbf{C}_2\mathbf{C}_4 + 2\mathbf{C}_4\mathbf{C}_2 - 10\mathbf{C}_5)\mathbf{e}^{(k)4} \\ &\quad + \frac{1}{2} \left(18\mathbf{C}_2^5 - 36\mathbf{C}_2^3\mathbf{C}_3 + 14\mathbf{C}_2^2\mathbf{C}_4 + 6\mathbf{C}_3\mathbf{C}_4 - \frac{3}{4}\mathbf{C}_2\mathbf{C}_3^2 + \mathbf{C}_2\mathbf{C}_5 - 3\mathbf{C}_6 \right) \mathbf{e}^{(k)5} \\ &\quad + \frac{1}{18} \left[-216\mathbf{C}_2^6 + 732\mathbf{C}_2^4\mathbf{C}_3 - 75\mathbf{C}_3^3 - 520\mathbf{C}_2^3\mathbf{C}_4 + 32\mathbf{C}_4^2 \right. \\ &\quad \left. - 2\mathbf{C}_2^2(95\mathbf{C}_3^2 - 74\mathbf{C}_5) + 30\mathbf{C}_3\mathbf{C}_5 + 30\mathbf{C}_5\mathbf{C}_3 \right. \\ &\quad \left. + 56\mathbf{C}_2\mathbf{C}_3\mathbf{C}_4 + 3\mathbf{C}_2\mathbf{C}_6 + 3\mathbf{C}_6\mathbf{C}_2 \right] \mathbf{e}^{(k)6} \Big\} \mathbf{F}'(\mathbf{x}^*)^{-1} + O(\mathbf{e}^{(k)7}). \end{aligned} \quad (2.9)$$

Consequently, by (2.2) and (2.9), we have

$$\begin{aligned} 2(\mathbf{F}'(\mathbf{x}^{(k)}) + \mathbf{F}'(\mathbf{y}^{(k)}))^{-1} \mathbf{F}'(\mathbf{x}^{(k)}) &= \mathbf{e}^{(k)} + \frac{1}{2}(-2\mathbf{C}_2^2 - \mathbf{C}_3)\mathbf{e}^{(k)3} \\ &\quad + \left(3\mathbf{C}_2^3 - \frac{3\mathbf{C}_2\mathbf{C}_3}{4} - \frac{3\mathbf{C}_3\mathbf{C}_2}{4} - \mathbf{C}_4 \right) \mathbf{e}^{(k)4} \\ &\quad + \left(-6\mathbf{C}_2^4 + 9\mathbf{C}_2^2\mathbf{C}_3 - \mathbf{C}_2\mathbf{C}_4 - \mathbf{C}_4\mathbf{C}_2 + \frac{3}{4}(\mathbf{C}_3^2 - 2\mathbf{C}_5) \right) \mathbf{e}^{(k)5} \\ &\quad + \frac{1}{2} \left(18\mathbf{C}_2^5 - 50\mathbf{C}_2^3\mathbf{C}_3 + 30\mathbf{C}_2^2\mathbf{C}_4 + 3\mathbf{C}_3\mathbf{C}_4 + 2\mathbf{C}_4\mathbf{C}_3 \right. \\ &\quad \left. + 5\mathbf{C}_2\mathbf{C}_3^2 - 3\mathbf{C}_2\mathbf{C}_5 - 2\mathbf{C}_5\mathbf{C}_2 - 4\mathbf{C}_6 \right) \mathbf{e}^{(k)6} + O(\mathbf{e}^{(k)7}) \end{aligned} \quad (2.10)$$

Substituting (2.10) into the second step of (2.1) results

$$\begin{aligned} \mathbf{z}^{(k)} - \mathbf{x}^* &= \mathbf{x}^{(k)} - \mathbf{x}^* - 2(\mathbf{F}'(\mathbf{x}^{(k)}) + \mathbf{F}'(\mathbf{y}^{(k)}))^{-1} \mathbf{F}'(\mathbf{x}^{(k)}) \\ &= \left(\mathbf{C}_2^2 + \frac{\mathbf{C}_3}{2} \right) \mathbf{e}^{(k)3} \\ &\quad + \left(-3\mathbf{C}_2^3 + \frac{3\mathbf{C}_2\mathbf{C}_3}{4} + \frac{3\mathbf{C}_3\mathbf{C}_2}{4} + \mathbf{C}_4 \right) \mathbf{e}^{(k)4} \\ &\quad + \left(6\mathbf{C}_2^4 - 9\mathbf{C}_2^2\mathbf{C}_3 + \mathbf{C}_2\mathbf{C}_4 + \mathbf{C}_4\mathbf{C}_2 - \frac{3}{4}(\mathbf{C}_3^2 - 2\mathbf{C}_5) \right) \mathbf{e}^{(k)5} \\ &\quad + \frac{1}{2} \left(-18\mathbf{C}_2^5 + 50\mathbf{C}_2^3\mathbf{C}_3 - 30\mathbf{C}_2^2\mathbf{C}_4 - 3\mathbf{C}_3\mathbf{C}_4 - 2\mathbf{C}_4\mathbf{C}_3 \right. \\ &\quad \left. - 5\mathbf{C}_2\mathbf{C}_3^2 - 3\mathbf{C}_2\mathbf{C}_5 - 2\mathbf{C}_5\mathbf{C}_2) + 4\mathbf{C}_6 \right) \mathbf{e}^{(k)6} + O(\mathbf{e}^{(k)7}). \end{aligned} \quad (2.11)$$

To obtain the general error equation of (2.1) we need

$$\begin{aligned}
\mathbf{F}'(\mathbf{x}^{(k)})^{-1}\mathbf{F}'(\mathbf{y}^{(k)}) &= \mathbf{I} - 2\mathbf{C}_2\mathbf{e}^{(k)} + (6\mathbf{C}_2^2 - 3\mathbf{C}_3)\mathbf{e}^{(k)2} \\
&\quad - 4(4\mathbf{C}_2^3 - 2\mathbf{C}_2\mathbf{C}_3 - 2\mathbf{C}_3\mathbf{C}_2 + \mathbf{C}_4)\mathbf{e}^{(k)3} \\
&\quad + (40\mathbf{C}_2^4 - 61\mathbf{C}_2^2\mathbf{C}_3 + 9\mathbf{C}_3^2 + 11\mathbf{C}_2\mathbf{C}_4 + 11\mathbf{C}_4\mathbf{C}_2 - 5\mathbf{C}_5)\mathbf{e}^{(k)4} \\
&\quad + \left(-96\mathbf{C}_2^5 + 198\mathbf{C}_2^3\mathbf{C}_3 - 88\mathbf{C}_2^2\mathbf{C}_4 + 24\mathbf{C}_3\mathbf{C}_4 \right. \\
&\quad \left. - 66\mathbf{C}_2\mathbf{C}_3^2 + 14\mathbf{C}_2\mathbf{C}_5 + 14\mathbf{C}_5\mathbf{C}_2 - 6\mathbf{C}_6\right)\mathbf{e}^{(k)5} \\
&\quad + \left(224\mathbf{C}_2^6 - 584\mathbf{C}_2^4\mathbf{C}_3 - 15\mathbf{C}_3^3 + 300\mathbf{C}_2^2\mathbf{C}_4 + 16\mathbf{C}_4^2 \right. \\
&\quad + 315\mathbf{C}_2^2\mathbf{C}_3^2 - 56\mathbf{C}_2\mathbf{C}_5 - 56\mathbf{C}_5\mathbf{C}_2 + 15\mathbf{C}_3\mathbf{C}_5 + 15\mathbf{C}_5\mathbf{C}_3 \\
&\quad \left. - 194\mathbf{C}_2\mathbf{C}_3\mathbf{C}_4 + 17\mathbf{C}_2\mathbf{C}_6 + 17\mathbf{C}_6\mathbf{C}_2\right)\mathbf{e}^{(k)6} + O(\mathbf{e}^{(k)7}). \tag{2.12}
\end{aligned}$$

Thus,

$$\begin{aligned}
\frac{7}{2}\mathbf{I} - 4\mathbf{F}'(\mathbf{x}^{(k)})^{-1}\mathbf{F}'(\mathbf{y}^{(k)}) + \frac{2}{3}(\mathbf{F}'(\mathbf{x}^{(k)})^{-1}\mathbf{F}'(\mathbf{y}^{(k)}))^2 &= \mathbf{I} + 2\mathbf{C}_2\mathbf{e}^{(k)} + 3\mathbf{C}_3\mathbf{e}^{(k)2} \\
&\quad + (-20\mathbf{C}_2^3 + \mathbf{C}_2\mathbf{C}_3 + \mathbf{C}_3\mathbf{C}_2 + 4\mathbf{C}_4)\mathbf{e}^{(k)3} \\
&\quad + \left(110\mathbf{C}_2^4 - 89\mathbf{C}_2^2\mathbf{C}_3 + \frac{9\mathbf{C}_3^2}{2} + \mathbf{C}_2\mathbf{C}_4 + \mathbf{C}_4\mathbf{C}_2 + 5\mathbf{C}_5\right)\mathbf{e}^{(k)4} \\
&\quad + 2\left(-216\mathbf{C}_2^5 + 300\mathbf{C}_2^3\mathbf{C}_3 - 58\mathbf{C}_2^2\mathbf{C}_4 - 66\mathbf{C}_2\mathbf{C}_3^2 \right. \\
&\quad \left. + \mathbf{C}_2\mathbf{C}_5 + 3\mathbf{C}_3\mathbf{C}_4 + 3\mathbf{C}_4\mathbf{C}_3 + \mathbf{C}_6\right)\mathbf{e}^{(k)5} \\
&\quad + \left(1456\mathbf{C}_2^6 - 2830\mathbf{C}_2^4\mathbf{C}_3 - 66\mathbf{C}_3^3 + 816\mathbf{C}_2^2\mathbf{C}_4 + 8\mathbf{C}_4^2 + 1176\mathbf{C}_2^2\mathbf{C}_3^2 - 146\mathbf{C}_2^2\mathbf{C}_5 \right. \\
&\quad \left. + 15\mathbf{C}_3\mathbf{C}_5 - 340\mathbf{C}_2\mathbf{C}_3\mathbf{C}_4 + \mathbf{C}_2\mathbf{C}_6 + \mathbf{C}_6\mathbf{C}_2\right)\mathbf{e}^{(k)6} + O(\mathbf{e}^{(k)7}). \tag{2.13}
\end{aligned}$$

In addition, we need

$$\begin{aligned}
\mathbf{F}'(\mathbf{x}^{(k)})^{-1}\mathbf{F}'(\mathbf{z}^{(k)}) &= \left(\mathbf{C}_2^2 + \frac{\mathbf{C}_3}{2}\right)\mathbf{e}^{(k)3} \\
&\quad + \left(-5\mathbf{C}_2^3 + \frac{\mathbf{C}_2\mathbf{C}_3}{4} + \frac{\mathbf{C}_3\mathbf{C}_2}{4} + \mathbf{C}_4\right)\mathbf{e}^{(k)4} \\
&\quad + \frac{1}{4}(64\mathbf{C}_2^4 - 52\mathbf{C}_2^2\mathbf{C}_3 - 9\mathbf{C}_3^2 + 6\mathbf{C}_5)\mathbf{e}^{(k)5} \\
&\quad + \frac{1}{4}\left(-160\mathbf{C}_2^5 + 268\mathbf{C}_2^3\mathbf{C}_3 - 76\mathbf{C}_2^2\mathbf{C}_4 - 15\mathbf{C}_3\mathbf{C}_4 - 15\mathbf{C}_4\mathbf{C}_3 \right. \\
&\quad \left. + 3\mathbf{C}_2\mathbf{C}_3^2 - 3\mathbf{C}_2\mathbf{C}_5 - 3\mathbf{C}_5\mathbf{C}_2 + 8\mathbf{C}_6\right)\mathbf{e}^{(k)6} + O(\mathbf{e}^{(k)7}). \tag{2.14}
\end{aligned}$$

Considering (2.13) and (2.14) and the third step of (2.1), the final error equation follows

$$\begin{aligned}
\mathbf{e}^{(k+1)} &= \mathbf{z}^{(k)} - \mathbf{x}^* \\
&\quad - \left(\frac{7}{2}\mathbf{I} - 4\mathbf{F}'(\mathbf{x}^{(k)})^{-1}\mathbf{F}'(\mathbf{y}^{(k)}) + \frac{2}{3}(\mathbf{F}'(\mathbf{x}^{(k)})^{-1}\mathbf{F}'(\mathbf{y}^{(k)}))^2\right)\mathbf{F}'(\mathbf{x}^{(k)})^{-1}\mathbf{F}'(\mathbf{z}^{(k)}) \\
&= \left(19\mathbf{C}_2^5 + 7\mathbf{C}_2^3\mathbf{C}_3 - \frac{5}{4}\mathbf{C}_2\mathbf{C}_3^2\right)\mathbf{e}^{(k)6} + O(\mathbf{e}^{(k)7}). \tag{2.15}
\end{aligned}$$

This establishes that the new method has sixth order of convergence for solving systems of nonlinear equations. \square

In practice, instead of using (2.1) directly, to avoid computing involved inverses, the following approach is

applied

$$\begin{cases} \boldsymbol{\nu}_1^{(k)} = \boldsymbol{x}^{(k)} - \boldsymbol{\gamma}^{(k)} \\ \boldsymbol{\nu}_2^{(k)} = \boldsymbol{x}^{(k)} - 2\boldsymbol{\zeta}^{(k)}, \\ \boldsymbol{x}^{(k+1)} = \boldsymbol{\nu}_2^{(k)} - \left(\frac{7}{2}\boldsymbol{I} - 4\boldsymbol{\xi}^{(k)} + \frac{3}{2}(\boldsymbol{\xi}^{(k)})^2 \right) \boldsymbol{\rho}^{(k)}, \end{cases} \quad (2.16)$$

where

$$\begin{cases} \boldsymbol{F}'(\boldsymbol{x}^{(k)})\boldsymbol{\gamma}^{(k)} = \boldsymbol{F}(\boldsymbol{x}^{(k)}), \\ \left(\boldsymbol{F}'(\boldsymbol{x}^{(k)}) + \boldsymbol{F}'(\boldsymbol{\nu}_1^{(k)}) \right) \boldsymbol{\zeta}^{(k)} = \boldsymbol{F}(\boldsymbol{x}^{(k)}), \\ \boldsymbol{F}'(\boldsymbol{x}^{(k)})\boldsymbol{\xi}^{(k)} = \boldsymbol{F}'(\boldsymbol{\nu}_1^{(k)}), \\ \boldsymbol{F}'(\boldsymbol{x}^{(k)})\boldsymbol{\rho}^{(k)} = \boldsymbol{F}(\boldsymbol{\nu}_2^{(k)}). \end{cases} \quad (2.17)$$

By (2.17), it can be elucidated that the proposed method (2.1) consumes two vector-function, $\boldsymbol{F}(\boldsymbol{x}^{(k)})$ and $\boldsymbol{F}(\boldsymbol{y}^{(k)})$, and two Jacobian matrices, $\boldsymbol{F}'(\boldsymbol{x}^{(k)})$ and $\boldsymbol{F}'(\boldsymbol{x}^{(k)}) + \boldsymbol{F}'(\boldsymbol{y}^{(k)})$ per iteration k . Finally, We need to solve four linear systems of equations per iteration in which merely one of them applies a different LU factorization.

3 Further development: General multipoint method

This section concerns with construction of a general multipoint for solving nonlinear systems of equations in which it uses the proposed method (2.1) as its first three steps as predictor. In fact, by adding one step it increases the order of convergence by three units and only one vector-function evaluation is required. It contains a economic solid frozen factor, too.

3.1 First development: ninth-order of convergence

Following we present a new ninth-order iterative method for solving nonlinear systems of equations:

$$\begin{cases} \boldsymbol{y}^{(k)} = \boldsymbol{x}^{(k)} - \boldsymbol{F}'(\boldsymbol{x}^{(k)})^{-1} \boldsymbol{F}(\boldsymbol{x}^{(k)}), \\ \boldsymbol{z}^{(k)} = \boldsymbol{x}^{(k)} - 2(\boldsymbol{F}'(\boldsymbol{x}^{(k)}) + \boldsymbol{F}'(\boldsymbol{y}^{(k)}))^{-1} \boldsymbol{F}(\boldsymbol{x}^{(k)}), \\ \boldsymbol{w}^{(k)} = \boldsymbol{z}^{(k)} - \left(\frac{7}{2}\boldsymbol{I} - 4\boldsymbol{F}'(\boldsymbol{x}^{(k)})^{-1} \boldsymbol{F}'(\boldsymbol{y}^{(k)}) + \frac{3}{2}(\boldsymbol{F}'(\boldsymbol{x}^{(k)})^{-1} \boldsymbol{F}'(\boldsymbol{y}^{(k)}))^2 \right) \\ \quad \boldsymbol{F}'(\boldsymbol{x}^{(k)})^{-1} \boldsymbol{F}(\boldsymbol{z}^{(k)}), \\ \boldsymbol{x}^{(k+1)} = \boldsymbol{w}^{(k)} - \left(\frac{7}{2}\boldsymbol{I} - 4\boldsymbol{F}'(\boldsymbol{x}^{(k)})^{-1} \boldsymbol{F}'(\boldsymbol{y}^{(k)}) + \frac{3}{2}(\boldsymbol{F}'(\boldsymbol{x}^{(k)})^{-1} \boldsymbol{F}'(\boldsymbol{y}^{(k)}))^2 \right) \\ \quad \boldsymbol{F}'(\boldsymbol{x}^{(k)})^{-1} \boldsymbol{F}(\boldsymbol{w}^{(k)}). \end{cases} \quad (3.1)$$

Similar to the proof the Theorem (2.1), we can prove the following.

Theorem 3.1. *Let $\boldsymbol{F} : \boldsymbol{D} \subseteq \mathbb{R}^n \rightarrow \mathbb{R}^n$ be sufficiently Frechet differentiable at each point of an open convex neighborhood \boldsymbol{D} of $\boldsymbol{x}^* \in \mathbb{R}^n$, that is a solution of the system $\boldsymbol{F}(\boldsymbol{x}) = \mathbf{0}$. Let us suppose that $\boldsymbol{F}'(\boldsymbol{x})$ is continuous and nonsingular in \boldsymbol{x}^* . Then, the sequence $\{\boldsymbol{x}^{(k)}\}_{k \geq 0}$ obtained using the iterative method (3.1) converges to \boldsymbol{x}^* with convergence rate 9, and its error equation is given by*

$$\boldsymbol{e}^{(k+1)} = \boldsymbol{C}\boldsymbol{e}^{(k)9} + O(\boldsymbol{e}^{(k)10}), \quad (3.2)$$

where

$$\boldsymbol{C} = \left(308\boldsymbol{C}_2^8 + 51\boldsymbol{C}_2^6\boldsymbol{C}_3 + 51\boldsymbol{C}_3\boldsymbol{C}_2^6 - 20\boldsymbol{C}_2^4\boldsymbol{C}_3^2 - 19\boldsymbol{C}_3^2\boldsymbol{C}_2^4 + \frac{5}{4}\boldsymbol{C}_2^2\boldsymbol{C}_3^3 + \frac{5}{4}\boldsymbol{C}_3^3\boldsymbol{C}_2^2 \right).$$

Notice that (3.1) requires three vector-function and two Jacobian matrices evaluations per iteration. It also needs solving five linear systems of equations per cycle in which one of them uses a different LU factorization.

3.2 Second development: twelfth-order of convergence

Following we present a new twelfth-order iterative method for solving nonlinear systems of equations:

$$\left\{ \begin{array}{l} \mathbf{y}^{(k)} = \mathbf{x}^{(k)} - \mathbf{F}'(\mathbf{x}^{(k)})^{-1} \mathbf{F}(\mathbf{x}^{(k)}), \\ \mathbf{z}^{(k)} = \mathbf{x}^{(k)} - 2(\mathbf{F}'(\mathbf{x}^{(k)}) + \mathbf{F}'(\mathbf{y}^{(k)}))^{-1} \mathbf{F}(\mathbf{x}^{(k)}), \\ \mathbf{w}^{(k)} = \mathbf{z}^{(k)} - \left(\frac{7}{2} \mathbf{I} - 4\mathbf{F}'(\mathbf{x}^{(k)})^{-1} \mathbf{F}'(\mathbf{y}^{(k)}) + \frac{3}{2} (\mathbf{F}'(\mathbf{x}^{(k)})^{-1} \mathbf{F}'(\mathbf{y}^{(k)}))^2 \right) \\ \quad \mathbf{F}'(\mathbf{x}^{(k)})^{-1} \mathbf{F}(\mathbf{z}^{(k)}), \\ \mathbf{u}^{(k)} = \mathbf{w}^{(k)} - \left(\frac{7}{2} \mathbf{I} - 4\mathbf{F}'(\mathbf{x}^{(k)})^{-1} \mathbf{F}'(\mathbf{y}^{(k)}) + \frac{3}{2} (\mathbf{F}'(\mathbf{x}^{(k)})^{-1} \mathbf{F}'(\mathbf{y}^{(k)}))^2 \right) \\ \quad \mathbf{F}'(\mathbf{x}^{(k)})^{-1} \mathbf{F}(\mathbf{w}^{(k)}), \\ \mathbf{x}^{(k+1)} = \mathbf{u}^{(k)} - \left(\frac{7}{2} \mathbf{I} - 4\mathbf{F}'(\mathbf{x}^{(k)})^{-1} \mathbf{F}'(\mathbf{y}^{(k)}) + \frac{3}{2} (\mathbf{F}'(\mathbf{x}^{(k)})^{-1} \mathbf{F}'(\mathbf{y}^{(k)}))^2 \right) \\ \quad \mathbf{F}'(\mathbf{x}^{(k)})^{-1} \mathbf{F}(\mathbf{u}^{(k)}). \end{array} \right. \quad (3.3)$$

Notice that (3.3) requires four vector-function and two Jacobian matrices evaluations per iteration. Furthermore, it needs solving six linear systems of equations per cycle in which only one of them uses a different LU factorization. Similar to the proof of Theorem (2.1), we can prove the following.

Theorem 3.2. *Let $\mathbf{F} : \mathbf{D} \subseteq \mathbb{R}^n \rightarrow \mathbb{R}^n$ be sufficiently Frechet differentiable at each point of an open convex neighborhood \mathbf{D} of $\mathbf{x}^* \in \mathbb{R}^n$, that is a solution of the system $\mathbf{F}(\mathbf{x}) = 0$. Let us suppose that $\mathbf{F}'(\mathbf{x})$ is continuous and nonsingular in \mathbf{x}^* . Then, the sequence $\{\mathbf{x}^{(k)}\}_{k \geq 0}$ obtained using the iterative method (3.3) converges to \mathbf{x}^* with convergence rate 12, and its error equation is given by*

$$\mathbf{e}^{(k+1)} = \mathbf{C} \mathbf{e}^{(k)12} + O(\mathbf{e}^{(k)13}), \quad (3.4)$$

where

$$\mathbf{C} = \mathbf{C}_2^3(-10\mathbf{C}_2^3 + \mathbf{C}_3^3) (76\mathbf{C}_2^4 + 14\mathbf{C}_2^2\mathbf{C}_3 + 14\mathbf{C}_3\mathbf{C}_2^2 - 5\mathbf{C}_3^2) (-10\mathbf{C}_2^3 + \mathbf{C}_3^3).$$

3.3 General m -point method $m \geq 4$

We now deal with developing a new general multipoint method. Indeed, based on what have been mentioned above, the method (2.1) can be developed as follows:

$$\left\{ \begin{array}{l} \nu_1^{(k)} = \mathbf{x}^{(k)} - \gamma^{(k)} \\ \nu_2^{(k)} = \mathbf{x}^{(k)} - 2\zeta^{(k)}, \\ \nu_3^{(k)} = \nu_2^{(k)} - \left(\frac{7}{2} \mathbf{I} - 4\xi^{(k)} + \frac{3}{2} (\xi^{(k)})^2 \right) \rho_3^{(k)}, \\ \nu_4^{(k)} = \nu_3^{(k)} - \left(\frac{7}{2} \mathbf{I} - 4\xi^{(k)} + \frac{3}{2} (\xi^{(k)})^2 \right) \rho_4^{(k)}, \\ \vdots \\ \mathbf{x}^{(k+1)} = \nu_m = \nu_{m-1}^{(k)} - \left(\frac{7}{2} \mathbf{I} - 4\xi^{(k)} + \frac{3}{2} (\xi^{(k)})^2 \right) \rho_m^{(k)}, \end{array} \right. \quad (3.5)$$

where

$$\left\{ \begin{array}{l} \mathbf{F}'(\mathbf{x}^{(k)}) \gamma^{(k)} = \mathbf{F}(\mathbf{x}^{(k)}), \\ \left(\mathbf{F}'(\mathbf{x}^{(k)}) + \mathbf{F}'(\nu_1^{(k)}) \right) \zeta^{(k)} = \mathbf{F}(\mathbf{x}^{(k)}), \\ \mathbf{F}'(\mathbf{x}^{(k)}) \xi^{(k)} = \mathbf{F}'(\nu_1^{(k)}), \\ \mathbf{F}'(\mathbf{x}^{(k)}) \rho_i^{(k)} = \mathbf{F}(\nu_{i-1}^{(k)}), \quad i = 3, 4, \dots, m. \end{array} \right. \quad (3.6)$$

Notice that m -step (3.5) requires $m - 1$ vector-function and two Jacobian matrices evaluations per iteration. In addition, it needs to solve $m + 1$ linear systems of equations per cycle in which one of them use a different Lu factorization.

Theorem 3.3. *The m -step ($m \geq 4$) iterative method (3.5) has the local convergence order $3(m - 1)$ using $m - 1$ vector-function evaluations of the sufficiently smooth function \mathbf{F} and two frozen Jacobian matrices per iteration.*

Proof. The proof is by induction on m . Theorem (3.1) accounts for $m = 4$. Assume that the result holds for m ; we will prove it for $m + 1$. To this end, we take advantage of the Traub's acceleration theorem [1] and corrector frozen Jacobian matrix together. In other words, Traub's theorem states that if iterative method $\mathbf{G}(\mathbf{x}^{(k)})$ has convergence order p , then $\mathbf{x}^{(k+1)} = \mathbf{G}(\mathbf{x}^{(k)}) - \mathbf{F}'(\mathbf{x}^{(k)})^{-1} \mathbf{F}(\mathbf{G}(\mathbf{x}^{(k)}))$ is of convergence order $p + 1$. On the other hand, the frozen Jacobian matrix factor, $\frac{7}{2} \mathbf{I} - 4 \mathbf{F}'(\mathbf{x}^{(k)})^{-1} \mathbf{F}'(\mathbf{y}^{(k)}) + \frac{3}{2} (\mathbf{F}'(\mathbf{x}^{(k)})^{-1} \mathbf{F}'(\mathbf{y}^{(k)}))^2$, based on the induction hypothesis plays a key role and improves the convergence order by three units per iterate. Therefore, the $(m + 1)$ -step of (3.5) has convergence order $3m + 3$, where 3 is the convergence improving factor for (3.5). Hence, the $(m + 1)$ -step has the convergence order $3(m + 1)$. \square

4 Computational complexities

In the iterative method (2.1), we solve four linear systems of equations, while just two LU factorizations must be done due to multiple right hand sides. In these situations, one could compute a factorization of the (Jacobian) matrix and use it repeatedly.

The computational efficiency of some algorithms are now computed by a practical efficiency index, defined by

$$FEI = p^{\frac{1}{CO}}, \quad (4.1)$$

also known as flops-like efficiency index (FEI) [18], wherein C stands for the total computational cost per iteration in terms of the number of functional evaluations along with cost of LU decompositions and solving two triangular systems (based on the flops), to observe the competence of distinctive methods. For solving r linear systems of equations with the frozen LU factorization, the following theorem is generally used to estimate the computational operations [14]

Theorem 4.1. *The number of products and quotients required for solving r linear systems of equations with the same matrix of coefficients, by using LU factorization, is*

$$o(n, r) = \frac{1}{3}n^3 + rn^2 - \frac{1}{3}n, \quad (4.2)$$

where n is the size of each system.

We also denote by $o(\mathbf{F})$ and $o(\mathbf{F}')$, the number of function and Jacobian matrices, respectively. Thus, for our purposes, it is more convenient to redefine (4.1) as

$$FEI = p^{\frac{1}{CO}}, \quad (4.3)$$

where $CO = o(\mathbf{F}) + o(\mathbf{F}') + o(n, r)$.

Theorem 4.2. *For (2.1), we have $FEI_{(2.1)} = 6^{\frac{1}{\frac{5n^3}{3} + 5n^2 + \frac{4}{3}n}}$*

Proof. The iterative method (2.1) has the following cost: n evaluations of scalar functions for $\mathbf{F}(\mathbf{x})$, n evaluations of scalar functions for $\mathbf{F}(\mathbf{z})$, then, $o(\mathbf{F}) = 2n$. It also needs n^2 evaluations of scalar functions for $\mathbf{F}'(\mathbf{x})$, again n^2 evaluations of scalar functions for $\mathbf{F}'(\mathbf{y})$, that is $o(\mathbf{F}') = 2n^2$. Furthermore, we must solve two linear and one matrix systems with the same LU factorization and another linear system with a different LU factorization. In other words, $o(n, (n + 2)) + o(n, 1) = \frac{1}{3}n^3 + (n + 2)n^2 - \frac{1}{3}n + \frac{1}{3}n^3 + n^2 - \frac{1}{3}n = \frac{5}{3}n^3 + 3n^2 - \frac{2}{3}n$. Consequently, $CO = \frac{5n^3}{3} + 5n^2 + \frac{4}{3}n$. \square

The following theorem provides flops-like efficiency index for (3.5).

Theorem 4.3. *For (3.5), $m \geq 4$, we have $FEI_{(3.5)} = 3(m - 1)^{\frac{1}{\frac{5n^3}{3} + (m+2)n^2 + \frac{3m-5}{3}n}}$.*

Proof. The iterative method (3.5) has the following cost: $(m - 1)n$ evaluations of scalar functions. It also needs n^2 evaluations of scalar functions for $\mathbf{F}'(\mathbf{x})$, again n^2 evaluations of scalar functions for $\mathbf{F}'(\mathbf{y})$, that is $o(\mathbf{F}') = 2n^2$. Furthermore, we must solve $m - 1$ linear systems and one matrix system with the same LU factorization and another linear system with a different LU factorization. In other words, $o(n, (n + m - 1)) + o(n, 1) = \frac{1}{3}n^3 + (n + m - 1)n^2 - \frac{1}{3}n + \frac{1}{3}n^3 + n^2 - \frac{1}{3}n = \frac{5}{3}n^3 + mn^2 - \frac{2}{3}n$. Consequently, $CO = \frac{5n^3}{3} + (m + 2)n^2 + \frac{3m-5}{3}n$. \square

Therefore we can easily obtain the following:

Corollary 4.1. For (3.1), $m = 4$, we have $FEI_{(3.5)} = 9 \frac{1}{\frac{5n^3}{3} + 6n^2 + \frac{7}{3}n}$.

Corollary 4.2. For (3.3), $m = 5$, we have $FEI_{(3.5)} = 12 \frac{1}{\frac{5n^3}{3} + 7n^2 + \frac{10}{3}n}$.

In the same manner we can prove that:

Theorem 4.4. $FEI_{(1.1)} = 2 \frac{1}{\frac{n^3}{3} + 2n^2 + \frac{2}{3}n}$, $FEI_{(1.2)} = 4 \frac{1}{\frac{5n^3}{3} + 3n^2 + \frac{1}{3}n}$, $FEI_{(1.3)} = 4 \frac{1}{3n^3 + 3n^2}$, and $FEI_{(1.4)} = \frac{1}{4 \frac{1}{2n^3 + 5n^2}}$.

Proof. First we examine Newton's FEI for (1.1). It uses n and n^2 evaluations for $\mathbf{F}(\mathbf{x}^{(k)})$ and $\mathbf{F}'(\mathbf{x}^{(k)})$, respectively. We must solve a linear system for $\mathbf{F}'(\mathbf{x}^{(k)})\Gamma = F(\mathbf{x}^{(k)})$ for Γ with $\frac{1}{3}n^3 + n^2 - \frac{1}{3}n$. Therefore, $CO = \frac{1}{3}n^3 + 2n^2 + \frac{2}{3}n$.

We deal with Jarratt's FEI for (1.2). It computes $\mathbf{F}(\mathbf{x}^{(k)})$, $\mathbf{F}'(\mathbf{x}^{(k)})$ and $\mathbf{F}'(\mathbf{y}^{(k)})$ with n , n^2 and n^2 evaluations, respectively. Furthermore, in the first step we must solve $\mathbf{F}'(\mathbf{x}^{(k)})\Gamma = F(\mathbf{x}^{(k)})$ for Γ with $\frac{1}{3}n^3 + n^2 - \frac{1}{3}n$ evaluations. And in the second step we must solve $(3\mathbf{F}'(\mathbf{y}^{(k)}) - \mathbf{F}'(\mathbf{x}^{(k)}))\Theta = (3\mathbf{F}'(\mathbf{y}^{(k)}) + \mathbf{F}'(\mathbf{x}^{(k)}))$ for Θ with $\frac{1}{3}n^3 + n^3 - \frac{1}{3}n$ evaluations. Consequently, $CO = \frac{5}{3}n^3 + 3n^2 + \frac{1}{3}n$.

We now turn to Sharma et al. FEI for (1.3). It computes $\mathbf{F}(\mathbf{x}^{(k)})$, $\mathbf{F}'(\mathbf{x}^{(k)})$ and $\mathbf{F}'(\mathbf{y}^{(k)})$ with n , n^2 and n^2 evaluations, respectively. Furthermore, in the first step we must solve $\mathbf{F}'(\mathbf{x}^{(k)})\Gamma = F(\mathbf{x}^{(k)})$ for Γ with $\frac{1}{3}n^3 + n^2 - \frac{1}{3}n$ evaluations. In the second step, we must solve two systems of matrix of equations, namely $\mathbf{F}'(\mathbf{x}^{(k)})\Phi = F'(\mathbf{y}^{(k)})$ and $\mathbf{F}'(\mathbf{y}^{(k)})\Psi = F'(\mathbf{x}^{(k)})$ with $2(\frac{1}{3}n^3 + n^3 - \frac{1}{3}n)$ evaluations. So, we have $CO = 3n^3 + 3n^2$.

Finally, the task is now to find FEI for Babajee et al. method (1.4). It computes $\mathbf{F}(\mathbf{x}^{(k)})$, $\mathbf{F}'(\mathbf{x}^{(k)})$ and $\mathbf{F}'(\mathbf{y}^{(k)})$ with n , n^2 and n^2 evaluations, respectively. Furthermore, in the first step we must solve $\mathbf{F}'(\mathbf{x}^{(k)})\Gamma = F(\mathbf{x}^{(k)})$ for Γ with $\frac{1}{3}n^3 + n^2 - \frac{1}{3}n$ evaluations. In the second step, we must solve one system of matrix of equations, namely $\mathbf{F}'(\mathbf{x}^{(k)})\Phi = F'(\mathbf{y}^{(k)})$ and with $\frac{1}{3}n^3 + n^3 - \frac{1}{3}n$ evaluations. We also solve another system, namely $(\mathbf{F}'(\mathbf{x}^{(k)}) + \mathbf{F}'(\mathbf{y}^{(k)}))\Upsilon = \mathbf{F}(\mathbf{x}^{(k)})$ with $\frac{1}{3}n^3 + n^2 - \frac{1}{3}n$ evaluations. Hence, $CO = 2n^3 + 5n^2$ \square

5 Numerical implementation

In this section, we want to apply our methods to solve four examples, taken from [14, 18]. Computations have been carried out using variable precision arithmetic with 600 digits of mantissa in Mathematica 8. Besides, to verify that our algorithms support their given theories, the approximated computational order of convergence (ACOC), see [14], has been computed by the formula

$$ACOC \approx \frac{\ln \left(\|\mathbf{x}^{(k+1)} - \mathbf{x}^{(k)}\| \|\mathbf{x}^{(k)} - \mathbf{x}^{(k-1)}\|^{-1} \right)}{\ln \left(\|\mathbf{x}^{(k)} - \mathbf{x}^{(k-1)}\| \|\mathbf{x}^{(k-1)} - \mathbf{x}^{(k-2)}\|^{-1} \right)} \quad (5.1)$$

In addition to the methods (1.2)-(1.4), (2.1), and (3.1)-(3.3), we consider the following methods for comparison purpose and illustrations

The sixth-order method by Cordero et al. in [14]:

$$\begin{cases} \mathbf{y}^{(k)} = \mathbf{x}^{(k)} - \frac{2}{3}\mathbf{F}'(\mathbf{x}^{(k)})^{-1}\mathbf{F}(\mathbf{x}^{(k)}), \\ \mathbf{z}^{(k)} = \mathbf{x}^{(k)} - \frac{1}{2}(3\mathbf{F}'(\mathbf{y}^{(k)}) - \mathbf{F}'(\mathbf{x}^{(k)}))^{-1}(3\mathbf{F}'(\mathbf{y}^{(k)}) + \mathbf{F}'(\mathbf{x}^{(k)}))\mathbf{F}'(\mathbf{x}^{(k)})^{-1}\mathbf{F}(\mathbf{x}^{(k)}), \\ \mathbf{x}^{(k+1)} = \mathbf{z}^{(k)} - 2(3\mathbf{F}'(\mathbf{y}^{(k)}) - \mathbf{F}'(\mathbf{x}^{(k)}))^{-1}\mathbf{F}(\mathbf{z}^{(k)}). \end{cases} \quad (5.2)$$

The sixth-order method by Cordero et al. in [21]:

$$\begin{cases} \mathbf{y}^{(k)} = \mathbf{x}^{(k)} - \frac{1}{2}\mathbf{F}'(\mathbf{x}^{(k)})^{-1}\mathbf{F}(\mathbf{x}^{(k)}), \\ \mathbf{z}^{(k)} = \frac{1}{3}(4\mathbf{y}^{(k)} - \mathbf{x}^{(k)}), \\ \mathbf{u}^{(k)} = \mathbf{y}^{(k)} + (\mathbf{F}'(\mathbf{x}^{(k)}) - 3\mathbf{F}'(\mathbf{z}^{(k)}))^{-1}\mathbf{F}(\mathbf{x}^{(k)}), \\ \mathbf{x}^{(k+1)} = \mathbf{u}^{(k)} + 2(\mathbf{F}'(\mathbf{x}^{(k)}) - 3\mathbf{F}'(\mathbf{z}^{(k)}))^{-1}\mathbf{F}(\mathbf{z}^{(k)}). \end{cases} \quad (5.3)$$

The sixth-order method by Soleymani et al. in [18]:

$$\begin{cases} \mathbf{y}^{(k)} = \mathbf{x}^{(k)} - \frac{2}{3}\mathbf{F}'(\mathbf{x}^{(k)})^{-1}\mathbf{F}(\mathbf{x}^{(k)}), \\ \mathbf{z}^{(k)} = \mathbf{x}^{(k)} - \frac{1}{2}(3\mathbf{F}'(\mathbf{y}^{(k)}) - \mathbf{F}'(\mathbf{x}^{(k)}))^{-1}(3\mathbf{F}'(\mathbf{y}^{(k)}) + \mathbf{F}'(\mathbf{x}^{(k)}))\mathbf{F}'(\mathbf{x}^{(k)})^{-1}\mathbf{F}(\mathbf{x}^{(k)}), \\ \mathbf{x}^{(k+1)} = \mathbf{z}^{(k)} - [(3\mathbf{F}'(\mathbf{y}^{(k)}) - \mathbf{F}'(\mathbf{x}^{(k)}))^{-1}(3\mathbf{F}'(\mathbf{y}^{(k)}) + \mathbf{F}'(\mathbf{x}^{(k)}))\mathbf{F}'(\mathbf{x}^{(k)})^{-1}\mathbf{F}(\mathbf{x}^{(k)})]^2 \\ \quad \mathbf{F}'(\mathbf{x}^{(k)})^{-1}\mathbf{F}(\mathbf{z}^{(k)}). \end{cases} \quad (5.4)$$

We compare the computed results and justify the accuracy and applicability of the mentioned algorithm and theorems. In fact, we want to estimate the zeros of the following nonlinear systems. In tables A_i means $\|\mathbf{F}(\mathbf{x}^{(i)})\|_\infty$ and $a \times 10^{-b}$ was shown with $a(-b)$.

We consider the following nonlinear systems of equations:

- (1) $\mathbf{F}(\mathbf{x}) = (e^{x_1^2} - e^{\sqrt{2}x_1}, x_1 - x_2)^T$ with $\mathbf{x}^{(0)} = (2, 2)^T$ and $\mathbf{x}^* = (\sqrt{2}, \sqrt{2})^T$.
- (2) $\mathbf{F}(\mathbf{x}) = (x_1^2 + x_2^2 + x_3^2 - 1, 2x_1^2 + x_2^2 - 4x_3, 3x_1^2 - 4x_2^2 + x_3^2)^T$ with $\mathbf{x}^{(0)} = (0.5, 0.5, 0.5)^T$ and $\mathbf{x}^* \approx (0.698288\dots, 0.628524\dots, 0.342561\dots)^T$.
- (3) In order to tackle with large-scale nonlinear systems, we have included this example in this work

$$\begin{cases} x_i x_{i+1} = 1, & i = 1, 2, \dots, n-1 \\ x_n x_1 = 1 \end{cases} \quad (5.5)$$

where a solution is the vector $\mathbf{x}^* = (1, 1, \dots, 1)$ for odd n , and its first Frechet derivative has the following sparse pattern :

$$J(x) = \begin{pmatrix} x_2 & x_1 & 0 & \cdots & 0 \\ 0 & x_3 & x_2 & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \ddots & 0 & x_{n-2} & x_{n-1} \\ x_n & 0 & \cdots & 0 & x_1 \end{pmatrix}, \quad (5.6)$$

and the initial value $\mathbf{x}^{(0)} = (2, 2, \dots, 2)^T$, also the case for 99×99 is considered.

- (4) In order to tackle with large-scale nonlinear systems , we have included this example in this work

$$\begin{cases} (x_i x_{i+1})^2 - 3 = 0, & i = 1, 2, \dots, n-1 \\ x_n (x_1)^2 - 1 = 0 \end{cases} \quad (5.7)$$

where its solution is the vector $\mathbf{x}^* \approx (0.57735, 3.0000, 0.57735, 3.0000, \dots, 0.57735, 3.0000)^T$ and note that its first Frechet derivative has the following sparse pattern

$$J(x) = \begin{pmatrix} 2x_1 x_2^2 & 2x_1^2 x_2 & 0 & \cdots & 0 \\ 0 & 2x_2 x_3^2 & 2x_2^2 x_3 & \cdots & \vdots \\ \vdots & \cdots & 2x_3 x_4^2 & 2x_3^2 x_4 & \vdots \\ 0 & \ddots & \ddots & \ddots & 0 \\ 2x_1 x_n & 0 & \cdots & 0 & x_1^2 \end{pmatrix}, \quad (5.8)$$

and the initial value $\mathbf{x}^{(0)} = Table [2.0 \{i, 1, 250\}]$, also the case for 250×250 is considered .

It can be seen from Tables 1-4 that numerical results demonstrate all selected methods converge with the expected rates. In fact, it is clear that methods (1.2)-(1.4) have convergence order four while methods (2.1) and (5.2)-(5.4) have sixth-order convergence. Moreover, methods (3.1) and (3.3) have ninth- and twelfth-order convergence. Last columns show that it can be concluded that approximated computational order of convergence (ACOC), calculated by (5.1), supports the very well the theoretical order.

In Tables 5-7, we demonstrate and compare the flop-like efficiency indices of the proposed methods together some other methods expressed in this paper. It can be concluded that our methods show good tendency as opposed the other methods.

| Methods | A_1 | A_2 | A_3 | ACOC |
|---------|--------|--------|-------------|--------|
| (1.2) | 2.8562 | 0.0470 | 4.3625(-8) | 3.9950 |
| (1.3) | 4.9170 | 0.3057 | 0.0001 | 3.9229 |
| (1.4) | 5.9382 | 0.5403 | 0.0020 | 3.8285 |
| (2.1) | 4.3234 | 0.1598 | 3.1611(-7) | 5.9998 |
| (3.1) | 2.9217 | 0.0179 | 2.1353(-18) | 9.0000 |
| (3.3) | 2.1491 | 0.0012 | 4.5650(-38) | 11.999 |
| (5.2) | 1.7595 | 0.0017 | 2.5856(-20) | 5.9999 |
| (5.3) | 1.7595 | 0.0017 | 2.5856(-20) | 5.9999 |
| (5.4) | 2.2600 | 0.0087 | 1.6334(-15) | 5.9552 |

Table 1: Numerical results for Example 1

| Methods | A_1 | A_2 | A_3 | ACOC |
|---------|--------|-------------|--------------|--------|
| (1.2) | 0.0084 | 2.0142(-11) | 4.2577(-46) | 4.0213 |
| (1.3) | 0.0228 | 2.3487(-9) | 1.8332(-37) | 4.0223 |
| (1.4) | 0.0415 | 3.8243(-8) | 2.0232(-32) | 4.0217 |
| (2.1) | 0.0085 | 4.3218(-16) | 5.9810(-96) | 6.0000 |
| (3.1) | 0.0019 | 2.1717(-29) | 5.0746(-263) | 9.0013 |
| (3.3) | 0.0004 | 1.2046(-46) | 2.2679(-557) | 12.000 |
| (5.2) | 0.0006 | 6.5609(-14) | 7.7067(-89) | 6.0071 |
| (5.3) | 0.0006 | 6.5609(-24) | 7.7067(-144) | 6.0070 |
| (5.4) | 0.0027 | 1.4796(-19) | 3.0421(-117) | 6.0062 |

Table 2: Numerical results for Example 2

| Methods | A_1 | A_2 | A_3 | ACOC |
|---------|--------|-------------|--------------|--------|
| (1.2) | 0.5037 | 9.2456(-7) | 1.1590(-29) | 3.9924 |
| (1.3) | 0.7925 | 0.0001 | 8.0715(-25) | 3.9748 |
| (1.4) | 1.0012 | 0.0004 | 2.4850(-22) | 3.9999 |
| (2.1) | 0.2720 | 6.8908(-11) | 2.0370(-68) | 5.9948 |
| (3.1) | 0.0545 | 2.4936(-22) | 2.2500(-205) | 9.0000 |
| (3.3) | 0.0112 | 7.5839(-38) | 6.9320(-460) | 11.999 |
| (5.2) | 0.0947 | 1.4083(-10) | 1.5626(-61) | 5.9999 |
| (5.3) | 0.0947 | 1.4083(-12) | 1.5626(-74) | 5.9999 |
| (5.4) | 0.1517 | 6.9449(-13) | 6.7420(-81) | 6.0000 |

Table 3: Numerical results for Example 3

| Methods | A_1 | A_2 | A_3 | ACOC |
|---------|---------|------------|-------------|--------|
| (1.2) | 0.5879 | 0.0088 | 1.2817(-15) | 4.0250 |
| (1.3) | 2.7267 | 0.1053 | 1.1451(-10) | 4.0817 |
| (1.4) | 5.9382 | 0.5403 | 0.0020 | 3.8658 |
| (2.1) | 4.7399 | 0.0001 | 1.9969(-38) | 5.9988 |
| (3.1) | 11.1123 | 6.0790(-6) | 1.3395(-54) | 8.9888 |
| (3.3) | 9.8499 | 2.0244(-8) | 3.4518(-87) | 8.8889 |
| (5.2) | 3.5087 | 2.6227(-6) | 1.5340(-45) | 5.8861 |
| (5.3) | 3.5087 | 2.6227(-6) | 1.5340(-45) | 5.8899 |
| (5.4) | 5.3508 | 0.0007 | 4.3403(-35) | 5.9977 |

Table 4: Numerical results for Example 4

| n | Method (2.1) | Method (1.1) | Method (1.2) | Method (1.3) | Method (1.4) |
|-----|--------------|--------------|--------------|--------------|--------------|
| 2 | 1.051 | 1.050 | 1.055 | 1.039 | 1.039 |
| 3 | 1.019 | 1.021 | 1.019 | 1.013 | 1.014 |
| 4 | 1.009 | 1.011 | 1.009 | 1.006 | 1.007 |
| 5 | 1.005 | 1.005 | 1.005 | 1.003 | 1.004 |
| 6 | 1.003 | 1.003 | 1.003 | 1.002 | 1.002 |

Table 5: EFI of the different methods for different sizes of the systems

| n | Method (2.1) | Method (3.1) | Method (3.3) |
|-----|--------------|--------------|--------------|
| 2 | 1.051 | 1.060 | 1.053 |
| 3 | 1.019 | 1.023 | 1.021 |
| 4 | 1.009 | 1.011 | 1.011 |
| 5 | 1.005 | 1.006 | 1.006 |
| 6 | 1.003 | 1.004 | 1.004 |

Table 6: EFI of the different methods for different sizes of the systems

6 Conclusion

First, we extended and developed Weerakoon and Fernando's method for solving nonlinear systems of equations with sixth-order convergence. Its convergence analysis was discussed. To develop a general multipoint iteration, we used this extended method as a predictor with two frozen Jacobian matrices. Some concrete examples of the general method have been given. In addition, we dealt with computational complexities. Numerical examples verified that these new methods can compete very good with the existing ones.

Acknowledgements

The first author is greatly indebted to Dr. Fazlollah Soleymani for several helpful comments.

References

- [1] J.F. Traub, Iterative Methods for the Solution of Equations, Prentice Hall, New York, 1964.
- [2] J. M. Ortega, W. C. Rheinboldt, Iterative Solution of Nonlinear Equations in Several Variables, Academic Press, New York, 1970.
- [3] J.M. Gutierrez, M.A. Hernandez, A family of Chebyshev-Halley type methods in Banach spaces, Bull. Austral. Math. Soc. 55 (1997) 113–130.
- [4] S. Amat, S. Busquier, J.M. Gutierrez, Geometric constructions of iterative methods to solve nonlinear equations, Comput. Appl. Math. 157 (2003) 197–205.

| n | Method (2.1) | Method (5.2) | Method (5.3) | Method (5.4) |
|-----|--------------|--------------|--------------|--------------|
| 2 | 1.051 | 1.036 | 1.062 | 1.053 |
| 3 | 1.019 | 1.012 | 1.024 | 1.019 |
| 4 | 1.009 | 1.006 | 1.012 | 1.009 |
| 5 | 1.005 | 1.003 | 1.007 | 1.005 |
| 6 | 1.003 | 1.002 | 1.004 | 1.003 |

Table 7: EFI of the different methods for different sizes of the systems

- [5] D. K. R. Babajee, M.Z. Dauhoo, M.T. Darvishi, A. Karami, A. Barati, Analysis of two Chebyshev-like third order methods free from second derivatives for solving systems of nonlinear equations, *J. Comput. Appl. Math.* 233 (8) (2010) 2002-2012.
- [6] D.K.R. Babajee, M.Z. Dauhoo, An analysis of the properties of the variants of Newton's method with third order convergence, *Appl. Math. Comput.* 183 (2006) 659-684.
- [7] A. Cordero, J.R. Torregrosa, Variants of Newton's method for functions of several functions, *Appl. Math. Comput.* 183 (2006) 199-208.
- [8] J.A. Ezquerro, M.A. Hernandez, On Halley-type iterations with free second derivative, *J. Comput. Appl. Math.* 170 (2004) 455-459.
- [9] M. Frontini, E. Sormani, Third-order methods from quadrature formulae for solving systems of nonlinear equations, *Appl. Math. Comput.* 149 (2004) 771-782.
- [10] H.H.H. Homeier, Modified Newton method with cubic convergence: The multivariate case, *J. Comput. Appl. Math.* 169 (2004) 161-169.
- [11] W. Haijun, New third-order method for solving systems of nonlinear equations, *Numer Algor* (2009) 50:271-282 DOI 10.1007/s11075-008-9227-2
- [12] M. Grau-Sanchez, Improvements of the efficiency of some three-step iterative like-Newton methods, *Numer. Math.* 107 (2007) 131-146.
- [13] P. Jarratt, Some fourth order multipoint iterative methods for solving equations, *Math. Comput.*, 20 (1966), 434-437.
- [14] A. Cordero, J.L. Hueso, E. Martinez, J.R. Torregrosa, A modified Newton-Jarratt's composition, *Numer. Algor.* 55 (2010), 87-99.
- [15] J.R. Sharma, R.K. Guha, R. Sharma, An efficient fourth order weighted-Newton method for systems of nonlinear equations, *Numer, Algor.* 62 (2013) 307-323.
- [16] D.K.R. Babajee, A. Cordero, F. Soleymani, J.R. Torregrosa, On a novel fourth-order algorithm for solving systems of nonlinear equations, Volume 2012, Article ID 165452, 12 pages, doi:10.1155/2012/165452
- [17] M.F. Abad, A. Cordero, J.R. Torregrosa, Fourth- and fifth-order methods for solving nonlinear systems of equations: An application to the global positioning system, *Abstract and Applied Analysis* Volume 2013, Article ID 586708, 10 pages <http://dx.doi.org/10.1155/2013/586708>.
- [18] F. Soleymani, T. Lotfi, P. Bakhtiari, A multi-step class of iterative methods for nonlinear systems, *Optim. Lett.*, 7 2013, DOI 10.1007/s11590-013-0617-6
- [19] H. Ren, Q. Wu, W. Bi, New variants of Jarratts method with sixth-order convergence, *Numer. Algor.* 52 (2009) 585-603 DOI 10.1007/s11075-009-9302-3
- [20] S. Weerakoon, T. C. I. Fernando, A variant of Newton's method with accelerated third-order convergence, *Appl. Math. Lett.* 13 (8), (2000), 87-93.
- [21] A. Cordero, J.R. Torregrosa, M.P. Vassileva, Increasing the order of convergence of iterative schemes for solving nonlinear systems, *J. Comput. Appl. Math.* (2012), doi: 10.1016/j.cam.2012.11.024