

Research Article

On a Novel Fourth-Order Algorithm for Solving Systems of Nonlinear Equations

Diyashvir K. R. Babajee,¹ Alicia Cordero,² Fazlollah Soleymani,³ and Juan R. Torregrosa²

¹ Department of Applied Mathematical Sciences, School of Innovative Technologies and Engineering, University of Technology, Mauritius, La Tour Koenig, Pointe aux Sables, Mauritius

² Instituto Universitario de Matemática Multidisciplinar, Universitat Politècnica de València, Camino de Vera, s/n, 40022 Valencia, Spain

³ Department of Mathematics, Islamic Azad University, Zahedan Branch, P.O. Box 987-98138 Zahedan, Iran

Correspondence should be addressed to Juan R. Torregrosa, jr Torre@mat.upv.es

Received 23 July 2012; Revised 9 October 2012; Accepted 10 October 2012

Academic Editor: Changbum Chun

Copyright © 2012 Diyashvir K. R. Babajee et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

This paper focuses on solving systems of nonlinear equations numerically. We propose an efficient iterative scheme including two steps and fourth order of convergence. The proposed method does not require the evaluation of second or higher order Frechet derivatives per iteration to proceed and reach fourth order of convergence. Finally, numerical results illustrate the efficiency of the method.

1. Introduction

Many relationships in nature are inherently nonlinear, which according to these effects are not in direct proportion to their cause. In fact, a large number of such real-world applications are reduced to solve a system of nonlinear equations numerically. Solving such systems has become one of the most challenging problems in numerical analysis, (see [1–4]), for example in Chaos (chaos theory studies the behavior of dynamical systems that are highly sensitive to initial conditions), or in climate studies (the weather is famously chaotic, where simple changes in one part of the system produce complex effects throughout).

Some robust and efficient methods for solving nonlinear systems are brought forward (e.g., see [5–7]). The Newton's iteration for nonlinear equations is a basic method, which converges quadratically under some conditions; this method has also been extended for systems of nonlinear equations. In this method, if the derivative of the multivariate function

$F(x)$ at an iterate point is singular or almost singular, namely $|F'(x)| \approx 0$, the iteration can be aborted due to the overflow or leads to divergence, which restrict its applications to some extent. Recently, there has been some modified Newton's methods with cubic convergence for systems of nonlinear equations in order to overcome on this flaw; see for example [8]. In fact, there exist fourth-order schemes that avoid the condition of nonsingular Jacobian matrix. For example, Noor et al. in [9] design different algorithms for solving nonlinear systems of equations, obtained by means of variational techniques. One of the iterative schemes proposed in this paper, that we use in the numerical section, has the following expression:

$$\begin{aligned} y^{(k)} &= x^{(k)} - \left[F'(x^{(k)}) - \text{diag}(\beta_1 f_1, \beta_2 f_2, \dots, \beta_n f_n)(x^{(k)}) \right]^{-1} F(x^{(k)}), \\ x^{(k+1)} &= y^{(k)} - \left[F'(y^{(k)}) - \text{diag}(\beta_1 f_1, \beta_2 f_2, \dots, \beta_n f_n)(y^{(k)}) \right]^{-1} F(y^{(k)}), \end{aligned} \quad (1.1)$$

denoted by NR, where f_i , $i = 1, \dots, n$ are the coordinate functions of F .

However, the condition of $|F'(x)| \neq 0$ cannot be always avoided, though as discussed in [10], the generalized inverse could be used instead of $F'(x)^{-1}$, and the convergence still remains quadratically. Moreover, the efficiency index of these solvers is not satisfactory for practical problems, which provide us a way to suggest high order iterative methods with high efficiency index.

In this paper, we suggest a novel two-step iterative scheme for finding the solution of nonlinear systems whose Jacobian matrix is required to be nonsingular. The main merit of our proposed scheme is that per iteration it does not require the evaluation of the second, or higher order Frechet derivatives, while the fourth-order of convergence is attainable using only one function and two first-order Frechet derivatives per computing step.

The paper is organized as follows. In the next section, we give a short review on basic notes in this field of study, while Section 3 proposes the new iterative algorithm to solve systems of nonlinear equations and analyze its local convergence. In Section 4, numerical examples will be further considered to show that the new method performs well. The detailed results for numerical examples are analyzed and discussed in this section. And finally, some conclusions are drawn in Section 5.

2. Preliminary Notes

Let us consider the problem of finding a real zero of a multivariate nonlinear function $F(x) = (f_1(x), f_2(x), \dots, f_n(x))^T$, $x = (x_1, x_2, \dots, x_n)^T$, $f_i : \mathbb{R}^n \rightarrow \mathbb{R}$, for all $i = 1, 2, \dots, n$ and $F : D \subset \mathbb{R}^n \rightarrow \mathbb{R}^n$ is a smooth map and D is an open and convex set. We assume that $\alpha = (\alpha_1, \alpha_2, \dots, \alpha_n)^T$ is a zero of the system, and $y = (y_1, y_2, \dots, y_n)^T$ is an initial guess sufficiently close to α .

Consider Taylor expansion of $F(x)$ about y as follows:

$$F(x) = F(y) + F'(y)(x - y) + G(x), \quad (2.1)$$

where $G(x) = O((x - y)^2)$. Replace y by $x^{(k)}$ into (2.1) to get

$$F(x) \approx F(x^{(k)}) + F'(x^{(k)})(x - x^{(k)}). \quad (2.2)$$

Solve (2.2) for x to get

$$x \approx x^{(k)} - [F'(x^{(k)})]^{-1} F(x^{(k)}) + [F'(x^{(k)})]^{-1} F(x). \quad (2.3)$$

If x is an approximation of α , we denote it by $x^{(k+1)}$ and get

$$x^{(k+1)} \approx x^{(k)} - [F'(x^{(k)})]^{-1} F(x^{(k)}) + [F'(x^{(k)})]^{-1} F(x^{(k+1)}), \quad (2.4)$$

that is,

$$x^{(k+1)} - x^{(k)} \approx [F'(x^{(k)})]^{-1} (F(x^{(k+1)}) - F(x^{(k)})). \quad (2.5)$$

Waziri et al. [11] proposed a technique to approximate the Jacobian inverse $[F'(x^{(k)})]^{-1}$ by a diagonal matrix $D_k = \text{diag}(d_i^{(k)})$, which is updated at each iteration, and where the i th diagonal entry of the matrix D_k is defined to be

$$d_i^{(k)} = \frac{x_i^{(k)} - x_i^{(k-1)}}{f_i(x^{(k)}) - f_i(x^{(k-1)})}, \quad (2.6)$$

where $x_i^{(k)}$ is the i th component of the vector $x^{(k)}$, for all $i = 1, 2, \dots, n$ and $k = 0, 1, 2, \dots$. We here remark that this procedure is similar to the Secant method in one-dimensional case. The approximation is valid only if $f_i(x^{(k)}) - f_i(x^{(k-1)})$ is not equal to zero or $|f_i(x^{(k)}) - f_i(x^{(k-1)})| > \delta$, where δ is a positive real number for all $i = 1, 2, \dots, n$. If else, then set $d_i^{(k)} = d_i^{(k-1)}$, for more details, please refer to [11].

The order of convergence and efficiency index of such solvers are indeed not good for practical problems in which iterations with high efficiencies are required. This was just the motivation of producing higher-order iterative nonlinear systems solvers. Third-order methods free from second derivatives were proposed from Quadrature rules for solving systems of nonlinear equations. These methods require one functional evaluation and two evaluations of first derivatives at two different points. An example of such iterations is the Arithmetic Mean Newton method (3rd AM) derived from Trapezoidal rule [12]

$$x^{(k+1)} = G_{3\text{rd AM}}(x^{(k)}) = x^{(k)} - A_1(x^{(k)})^{-1} F(x^{(k)}), \quad (2.7)$$

where

$$A_1(x) = \frac{F'(x) + F'(y)}{2}, \quad (2.8)$$

wherein y is the Newton's iteration.

The convergence analysis of these methods using point of attraction theory can be found in [13]. This third-order Newton-like method is more efficient than Halley's method because it does not require the evaluation of a third-order tensor of n^3 values.

In the next section, we present our new algorithm which reaches the highest possible order four by using only three functional evaluations of the multivariate function for solving systems of nonlinear equations.

3. A Novel Computational Iterative Algorithm

Let p be the order of a method and d be defined as

$$d = d_0 n + \sum_{j=1}^q d_j n^{j+1}, \quad (3.1)$$

where d_0 and d_j represent the number of times F and $F^{(j)}$ are to be evaluated, respectively. The definition of the Logarithms Informational Efficiency or Efficiency Index for nonlinear systems (IE) [1] is given by

$$\text{IE} = \frac{\ln p}{d_0 n + \sum_{j=1}^q d_j n^{j+1}}. \quad (3.2)$$

The efficiency indices of the Newton method (2nd NM) and the third-order methods free from second derivatives (3rd AM) are given by

$$\text{IE}_{2\text{nd NM}} = \frac{\ln 2}{n + n^2}, \quad \text{IE}_{3\text{rd AM}} = \frac{\ln 3}{n + 2n^2}, \quad (3.3)$$

respectively. We observe that $\text{IE}_{3\text{rd AM}} > \text{IE}_{2\text{nd NM}}$, if $n = 1$. That is, the third-order methods free from second derivatives are less efficient than Newton's method for systems of nonlinear equations. Let us also note that NR scheme has the same efficiency index as Newton's method ($\text{IE}_{4\text{th NR}} = \ln 4 / (2n + 2n^2) = \text{IE}_{2\text{nd NR}}$). Thus, it is important to develop fourth-order methods from these third-order methods to improve the efficiency. Soleymani et al. [5] have recently improved the 3rd AM method to get a class of fourth-order Jarratt-type methods for solving the nonlinear equation $f(x) = 0$. A simplified example is

$$y_k = x_k - \frac{2 f(x_k)}{3 f'(x_k)}, \quad (3.4)$$

$$x_{k+1} = x_k - \frac{2f(x_k)}{f'(x_k) + f'(y_k)} \left[1 - \frac{1}{4} \left(\frac{f'(y_k)}{f'(x_k)} - 1 \right) + \frac{3}{4} \left(\frac{f'(y_k)}{f'(x_k)} - 1 \right)^2 \right].$$

We extend this method to the multivariate case. The improved fourth-order Arithmetic Mean Newton method (4th AM) for systems of nonlinear equations can now be suggested as follows:

$$\begin{aligned} x^{(k+1)} &= G_{4\text{th AM}}(x^{(k)}) \\ &= x^{(k)} - H_1(x^{(k)}) \left[A_1(x^{(k)}) \right]^{-1} F(x), \end{aligned} \quad (3.5)$$

where $H_1(x) = (I - (1/4)(\tau(x) - I) + (3/4)(\tau(x) - I)^2)$, $\tau(x) = [F'(x)]^{-1}F'(x - (2/3)u(x))$, and I is the $n \times n$ identity matrix and $u(x) = [F'(x)]^{-1}F(x)$. Thus, by simplifying, we can produce the following fourth-order convergent algorithm.

Proposed Algorithm

Step 1. Solve the linear system $F'(x^{(k)})u(x^{(k)}) = F(x^{(k)})$,

Step 2. Calculate

- (i) $y^{(k)} = x^{(k)} - (2/3)u(x^{(k)})$,
- (ii) $\tau(x^{(k)}) = [F'(x^{(k)})]^{-1}F'(y^{(k)})$,

Step 3. Solve the linear system $A_1(x^{(k)})h(x^{(k)}) = F(x^{(k)})$, where $A_1(x^{(k)}) = (1/2)(F'(x^{(k)}) + F'(y^{(k)}))$,

Step 4. Calculate $x^{(k+1)} = x^{(k)} - (I - (1/4)(\tau(x^{(k)}) - I) + (3/4)(\tau(x^{(k)}) - I)^2)h(x^{(k)})$.

The main theorem is going to be demonstrated by means of the n -dimensional Taylor expansion of the functions involved. Let $F : D \subseteq \mathbb{R}^n \rightarrow \mathbb{R}^n$ be sufficiently Frechet differentiable in D . By using the notation introduced in [14], the q th derivative of F at $u \in \mathbb{R}^n$, $q \geq 1$, is the q -linear function $F^{(q)}(u) : \mathbb{R}^n \times \cdots \times \mathbb{R}^n \rightarrow \mathbb{R}^n$ such that $F^{(q)}(u)(v_1, \dots, v_q) \in \mathbb{R}^n$. It is easy to observe that

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

So, in the following, we will denote:

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

It is well known that for $\alpha + h \in \mathbb{R}^n$ lying in a neighborhood of a solution α of the nonlinear system $F(x) = 0$, Taylor's expansion can be applied (assuming that the Jacobian matrix $F'(\alpha)$ is nonsingular), and

$$F(\alpha + h) = F'(\alpha) \left[h + \sum_{q=2}^{p-1} C_q h^q \right] + O(h^p), \quad (3.6)$$

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

In addition, we can express F' as

$$F'(\alpha + h) = F'(\alpha) \left[I + \sum_{q=2}^{p-1} qC_q h^{q-1} \right] + O(h^p), \quad (3.7)$$

where I is the identity matrix. Therefore, $qC_q h^{q-1} \in \mathcal{L}(\mathbb{R}^n)$. From (3.7), we obtain

$$[F'(\alpha + h)]^{-1} = \left[I + X_1 h + X_2 h^2 + X_3 h^3 + \dots \right] [F'(\alpha)]^{-1} + O(h^p), \quad (3.8)$$

where

$$\begin{aligned} X_1 &= -2C_2, \\ X_2 &= 4C_2^2 - 3C_3, \\ X_3 &= -8C_2^3 + 6C_2C_3 + 6C_3C_2 - 4C_4, \\ &\vdots \end{aligned} \quad (3.9)$$

We denote $e^{(k)} = x^{(k)} - \alpha$ as the error in the k th iteration. The equation

$$e^{(k+1)} = L e^{(k)p} + O(e^{(k)p+1}), \quad (3.10)$$

where L is a p -linear function $L \in \mathcal{L}(\mathbb{R}^n \times \dots \times \mathbb{R}^n, \mathbb{R}^n)$, is called the *error equation*, and p is the *order of convergence*. Observe that $e^{(k)p}$ is $(e^{(k)}, e^{(k)}, \dots, e^{(k)})$.

Theorem 3.1. *Let $F : D \subseteq \mathbb{R}^n \rightarrow \mathbb{R}^n$ be sufficiently Frechet differentiable at each point of an open convex neighborhood D of $\alpha \in \mathbb{R}^n$, that is, a solution of the system $F(x) = 0$. Let us suppose that $F'(x)$ is continuous and nonsingular in α , and $x^{(0)}$ close enough to α . Then the sequence $\{x^{(k)}\}_{k \geq 0}$ obtained using the iterative expression (3.5) converges to α with order 4, being the error equation*

$$e^{(k+1)} = \left(\frac{1}{9}C_4 + \frac{11}{3}C_2^3 - C_3C_2 \right) e^{(k)4} + O(e^{(k)5}). \quad (3.11)$$

Proof. From (3.6) and (3.7), we obtain

$$\begin{aligned} F(x^{(k)}) &= F'(\alpha) \left[e^{(k)} + C_2 e^{(k)2} + C_3 e^{(k)3} + C_4 e^{(k)4} \right] + O(e^{(k)5}), \\ F'(x^{(k)}) &= F'(\alpha) \left[I + 2C_2 e^{(k)} + 3C_3 e^{(k)2} + 4C_4 e^{(k)3} + 5C_5 e^{(k)4} \right] + O(e^{(k)5}), \end{aligned} \quad (3.12)$$

where $C_k = (1/k!) [F'(\alpha)]^{-1} F^{(k)}(\alpha)$, $k = 2, 3, \dots$, and $e^{(k)} = x^{(k)} - \alpha$.

From (3.8), we have

$$[F'(x^{(k)})]^{-1} = \left[I + X_1 e^{(k)} + X_2 e^{(k)2} + X_3 e^{(k)3} \right] [F'(\alpha)]^{-1} + O(e^{(k)4}), \quad (3.13)$$

where $X_1 = -2C_2$, $X_2 = 4C_2^2 - 3C_3$ and $X_3 = -8C_2^3 + 6C_2C_3 + 6C_3C_2 - 4C_4$.

Then,

$$\begin{aligned} \left[F'(x^{(k)}) \right]^{-1} F(x^{(k)}) &= e^{(k)} - C_2 e^{(k)2} + 2(C_2^2 - C_3) e^{(k)3} \\ &+ \left(-4C_2^3 + 4C_2C_3 + 3C_3C_2 - 3C_4 \right) e^{(k)4} + O(e^{(k)5}), \end{aligned} \quad (3.14)$$

and the expression for $y^{(k)}$ is

$$\begin{aligned} y^{(k)} &= \alpha + \frac{1}{3} e^{(k)} + \frac{2}{3} C_2 e^{(k)2} - \frac{4}{3} (C_2^2 - C_3) e^{(k)3} \\ &+ \left(2C_4 - \frac{8}{3} C_2C_3 - 2C_3C_2 + 8C_2^3 \right) e^{(k)4} + O(e^{(k)5}). \end{aligned} \quad (3.15)$$

The Taylor expansion of Jacobian matrix $F'(y^{(k)})$ is

$$\begin{aligned} F'(y^{(k)}) &= F'(\alpha) \left[I + 2C_2(y^{(k)} - \alpha) + 3C_3(y^{(k)} - \alpha)^2 + 4C_4(y^{(k)} - \alpha)^3 \right. \\ &\quad \left. + 5C_5(y^{(k)} - \alpha)^4 \right] + O(e^{(k)5}) \\ &= F'(\alpha) \left[I + N_1 e^{(k)} + N_2 e^{(k)2} + N_3 e^{(k)3} \right] + O(e^{(k)4}), \end{aligned} \quad (3.16)$$

where

$$\begin{aligned} N_1 &= \frac{2}{3} C_2, \\ N_2 &= \frac{4}{3} C_2^2 + \frac{1}{3} C_3, \\ N_3 &= -\frac{8}{3} C_2(C_2^2 - C_3) + \frac{4}{3} C_3C_2 + \frac{4}{27} C_4. \end{aligned} \quad (3.17)$$

Therefore,

$$\begin{aligned} \tau(x^{(k)}) &= \left[F'(x^{(k)}) \right]^{-1} F'(y^{(k)}) \\ &= I + (N_1 + X_1) e^{(k)} + (N_2 + X_1N_1 + X_2) e^{(k)2} \\ &\quad + (N_3 + X_1N_2 + X_2N_1 + X_3) e^{(k)3} + O(e^{(k)4}) \\ &= I - \frac{4}{3} C_2 e^{(k)} + \left(4C_2^2 - \frac{8}{3} C_3 \right) e^{(k)2} \\ &\quad + \left(-\frac{32}{3} C_2^3 + 8C_2C_3 + \frac{16}{3} C_3C_2 - \frac{104}{27} C_4 \right) e^{(k)3} + O(e^{(k)4}), \end{aligned} \quad (3.18)$$

and then,

$$\begin{aligned}
 H_1(x^{(k)}) - I &= -\frac{1}{4}(\tau(x^{(k)}) - I) + \frac{3}{4}(\tau(x^{(k)}) - I)^2 \\
 &= \frac{1}{3}C_2e^{(k)} + \left(\frac{1}{3}C_2^2 + \frac{2}{3}C_3\right)e^{(k)^2} \\
 &\quad + \left(-\frac{16}{3}C_2^3 + \frac{2}{3}C_2C_3 + \frac{4}{3}C_3C_2 + \frac{26}{27}C_4\right)e^{(k)^3} + O(e^{(k)^4}).
 \end{aligned} \tag{3.19}$$

On the other hand, the arithmetic mean of the Jacobian matrices can be expressed as

$$\begin{aligned}
 A_1(x^{(k)}) &= \frac{1}{2}(F'(x^{(k)}) + F'(y^{(k)})) \\
 &= F'(\alpha) \left[I + \frac{4}{3}C_2e^{(k)} + \left(\frac{2}{3}C_2^2 + \frac{5}{3}C_3\right)e^{(k)^2} \right. \\
 &\quad \left. + \left(-\frac{4}{3}C_2^3 + \frac{4}{3}C_2C_3 + \frac{2}{3}C_3C_2 + \frac{56}{27}C_4\right)e^{(k)^3} \right] + O(e^{(k)^4}).
 \end{aligned} \tag{3.20}$$

From an analogous reasoning as in (3.8), we obtain

$$[A_1(x^{(k)})]^{-1} = [I + Y_1e^{(k)} + Y_2e^{(k)^2} + Y_3e^{(k)^3}] [F'(\alpha)]^{-1} + O(e^{(k)^4}), \tag{3.21}$$

where

$$\begin{aligned}
 Y_1 &= -\frac{4}{3}C_2, \\
 Y_2 &= \frac{10}{9}C_2^2 - \frac{5}{3}C_3, \\
 Y_3 &= -\frac{56}{27}C_4 + \frac{20}{27}C_2^3 + \frac{8}{9}C_2C_3 + \frac{14}{9}C_3C_2.
 \end{aligned} \tag{3.22}$$

So,

$$\begin{aligned}
 h(x^{(k)}) &= A_1(x^{(k)})^{-1} F(x^{(k)}) \\
 &= e^{(k)} - \frac{1}{3}C_2e^{(k)^2} + \left(-\frac{2}{9}C_2^2 - \frac{2}{3}C_3\right)e^{(k)^3} \\
 &\quad + \left(\frac{50}{27}C_2^3 - \frac{4}{9}C_2C_3 - \frac{1}{9}C_3C_2 - \frac{29}{27}C_4\right)e^{(k)^4} + O(e^{(k)^5}),
 \end{aligned} \tag{3.23}$$

and finally, by using (3.19) and (3.23), the error equation can be expressed as

$$\begin{aligned} e^{(k+1)} &= x^{(k)} - H_1(x^{(k)})A_1(x^{(k)})^{-1}F(x^{(k)}) - \alpha \\ &= \left(\frac{1}{9}C_4 + \frac{11}{3}C_2^3 - C_3C_2\right)e^{(k)4} + O(e^{(k)5}). \end{aligned} \quad (3.24)$$

So, taking into account (3.24), it can be concluded that the order of convergence of the proposed method is four. \square

The efficiency index of the fourth-order method free from second derivative (4th AM) is given by

$$\text{IE}_{4\text{th AM}} = \frac{\ln 4}{n + 2n^2}, \quad \frac{\text{IE}_{4\text{th AM}}}{\text{IE}_{2\text{nd NM}}} = \frac{2n + 2}{2n + 1} > 1, \quad n \geq 1. \quad (3.25)$$

This shows that the 4th AM method is more efficient than 2nd NM and 3rd AM methods. We next conduct some numerical experiments to compare the methods.

4. Numerical Examples

In this section, we compare the performance of contributed method with Newton's scheme and Arithmetic Mean Newton method (2.7). The algorithms have been written in MATLAB 7.6 and tested for the examples given below. For the following test problems, the approximate solutions are calculated correct to 500 digits by using variable arithmetic precision. We use the following stopping criterium:

$$\text{err} = \|F(x^{(k)})\| + \|x^{(k+1)} - x^{(k)}\| < 1 \cdot e - 100. \quad (4.1)$$

We have used the approximated computational order of convergence p given by (see [15])

$$p_c \approx \frac{\log(\|x^{(k+1)} - x^{(k)}\|) / (\|x^{(k)} - x^{(k-1)}\|)}{\log(\|x^{(k)} - x^{(k-1)}\|) / (\|x^{(k-1)} - x^{(k-2)}\|)}. \quad (4.2)$$

Let Iter_{\max} be the number of iterations required before convergence is reached and err_{\min} be the minimum residual.

We check the mentioned method by solving the following test problems.

Test Problem 1 (TP1). Consider $F(x_1, x_2) = 0$, where $F : (4, 6) \times (5, 7) \rightarrow \mathbb{R}^2$, and

$$F(x_1, x_2) = \left(x_1^2 - x_2 - 19, \frac{x_2^3}{6} - x_1^2 + x_2 - 17\right). \quad (4.3)$$

The Jacobian matrix is given by $F'(x) = \begin{pmatrix} 2x_1 & -1 \\ -2x_1 & (1/2)x_2^2+1 \end{pmatrix}$. The starting vector is $x^{(0)} = (5.1, 6.1)^T$ and the exact solution is $\alpha = (5, 6)^T$. In this case, the comparison of efficiencies is

$$\begin{aligned} \text{IE}_{2\text{nd NM}} = \frac{\ln 2}{2+4} = 0.1155 = \text{IE}_{4\text{th NR}}, \quad \text{IE}_{3\text{rd AM}} = \frac{\ln 3}{2+2(4)} = 0.1099, \\ \text{IE}_{4\text{th AM}} = \frac{\ln 4}{2+2(4)} = 0.1386. \end{aligned} \quad (4.4)$$

Test Problem 2 (TP2). Consider

$$\begin{aligned} \cos x_2 - \sin x_1 &= 0, \\ x_3^{x_1} - \frac{1}{x_2} &= 0, \\ \exp x_1 - x_3^2 &= 0. \end{aligned} \quad (4.5)$$

The solution is

$$\alpha \approx (0.909569, 0.661227, 1.575834)^T. \quad (4.6)$$

We choose the starting vector $x^{(0)} = (1, 0.5, 1.5)^T$. The Jacobian matrix is given by

$$F'(x) = \begin{pmatrix} -\cos x_1 & -\sin x_2 & 0 \\ x_3^{x_1} \ln x_3 & \frac{1}{x_2^2} & x_3^{x_1} \frac{x_1}{x_3} \\ \exp x_1 & 0 & -2x_3 \end{pmatrix}, \quad (4.7)$$

and has 7 nonzero elements. In this case, the comparison of efficiencies is: $\text{IE}_{2\text{nd NM}} = \ln 2 / (3+7) = 0.0693 = \text{IE}_{4\text{th NR}}$, $\text{IE}_{3\text{rd AM}} = \ln 3 / (3+2(7)) = 0.0646$, $\text{IE}_{4\text{th AM}} = \ln 4 / (3+2(7)) = 0.0815$.

Test Problem 3 (TP3). Consider

$$\begin{aligned} x_2 x_3 + x_4(x_2 + x_3) &= 0, \\ x_1 x_3 + x_4(x_1 + x_3) &= 0, \\ x_1 x_2 + x_4(x_1 + x_2) &= 0, \\ x_1 x_2 + x_1 x_3 + x_2 x_3 &= 1. \end{aligned} \quad (4.8)$$

We solve this system using the initial approximation $x^{(0)} = (0.5, 0.5, 0.5, -0.2)^T$. The solution is

$$\alpha \approx (0.577350, 0.577350, 0.577350, -0.288675)^T. \quad (4.9)$$

Table 1: Comparison of different methods for systems of nonlinear equations.

Methods	TP1			TP2			TP3		
	Iter _{max}	err _{min}	p_c	Iter _{max}	err _{min}	p_c	Iter _{max}	err _{min}	p_c
2nd NM	7	$8.8e-113$	2	9	$1.7e-107$	2	8	$1.7e-144$	2.02
3rd AM	5	$1.9e-140$	3.02	7	$5.9e-255$	3	6	$6.3e-276$	3.02
4th NR	4	$8.8e-113$	4.02	5	$1.7e-107$	4	5	$8.9e-291$	4.11
4th AM	4	$4.9e-102$	4.02	6	$2.7e-299$	4	5	$1.0e-237$	4.14

The Jacobian is a matrix given by

$$F'(x) = \begin{pmatrix} 0 & x_3 + x_4 & x_2 + x_4 & x_2 + x_3 \\ x_3 + x_4 & 0 & x_1 + x_4 & x_1 + x_3 \\ x_2 + x_4 & x_1 + x_4 & 0 & x_1 + x_2 \\ x_2 + x_3 & x_1 + x_3 & x_1 + x_2 & 0 \end{pmatrix}, \quad (4.10)$$

with 12 nonzero elements. In this case, the comparison of efficiencies is: $IE_{2nd\ NM} = \ln 2 / (4 + 12) = 0.0433 = IE_{4th\ NR}$, $IE_{3rd\ AM} = \ln 3 / (4 + 2(12)) = 0.0392$, $IE_{4th\ AM} = \ln 4 / (2 + 2(12)) = 0.0495$.

Table 1 gives the results for Test Problems 1, 2, and 3. It is observed for all problems that the fourth-order method converges in the least number of iterations. The computational order of convergence agrees with the theory. 4th AM gives the best results in terms of least residual, and it is the most efficient method compared to 2nd NM and 3rd AM. It can compete with the 4th NR method and has the advantage of having a higher efficiency index.

5. Concluding Remarks

The efficiency of the quadratically and cubically multidimensional methods is not satisfactory in most practical problems. Thus in this paper, we have extended a fourth-order method from a third-order method for systems of nonlinear equations. We have shown that the fourth-order iterative method is more efficient than the second order Newton and third-order methods. Numerical experiments have also shown that the fourth-order method is efficient.

Acknowledgments

The authors would like to thank the referees for the valuable comments and for the suggestions to improve the readability of the paper. This research was supported by Ministerio de Ciencia y Tecnología MTM2011-28636-C02-02.

References

- [1] D. K. R. Babajee, M. Z. Dauhoo, M. T. Darvishi, A. Karami, and A. Barati, "Analysis of two Chebyshev-like third order methods free from second derivatives for solving systems of nonlinear equations," *Journal of Computational and Applied Mathematics*, vol. 233, no. 8, pp. 2002–2012, 2010.
- [2] M. T. Darvishi and A. Barati, "A fourth-order method from quadrature formulae to solve systems of nonlinear equations," *Applied Mathematics and Computation*, vol. 188, no. 1, pp. 257–261, 2007.

- [3] J. A. Ezquerro, J. M. Gutiérrez, M. A. Hernández, and M. A. Salanova, "A biparametric family of inverse-free multipoint iterations," *Computational & Applied Mathematics*, vol. 19, no. 1, pp. 109–124, 2000.
- [4] J. F. Traub, *Iterative Methods for the Solution of Equations*, Prentice Hall, Englewood Cliffs, NJ, USA, 1964.
- [5] F. Soleymani, S. K. Khattri, and S. Karimi Vanani, "Two new classes of optimal Jarratt-type fourth-order methods," *Applied Mathematics Letters*, vol. 25, no. 5, pp. 847–853, 2012.
- [6] A. Cordero, J. L. Hueso, E. Martínez, and J. R. Torregrosa, "Accelerated methods for order $2p$ for systems of nonlinear equations," *Journal of Computational and Applied Mathematics*, vol. 233, no. 10, pp. 2696–2702, 2010.
- [7] B. H. Dayton, T.-Y. Li, and Z. Zeng, "Multiple zeros of nonlinear systems," *Mathematics of Computation*, vol. 80, no. 276, pp. 2143–2168, 2011.
- [8] W. Haijun, "New third-order method for solving systems of nonlinear equations," *Numerical Algorithms*, vol. 50, no. 3, pp. 271–282, 2009.
- [9] M. A. Noor, M. Waseem, K. I. Noor, and E. Al-Said, "Variational iteration technique for solving a system of nonlinear equations," *Optimization Letters*. In press.
- [10] A. Ben-Israel and T. N. E. Greville, *Generalized Inverses*, Springer, New York, NY, USA, 2nd edition, 2003.
- [11] M. Y. Waziri, W. J. Leong, M. A. Hassan, and M. Monsi, "An efficient solver for systems of nonlinear equations with singular Jacobian via diagonal updating," *Applied Mathematical Sciences*, vol. 4, no. 69–72, pp. 3403–3412, 2010.
- [12] M. Frontini and E. Sormani, "Third-order methods from quadrature formulae for solving systems of nonlinear equations," *Applied Mathematics and Computation*, vol. 149, no. 3, pp. 771–782, 2004.
- [13] D. K. R. Babajee, *Analysis of higher order variants of Newton's method and their applications to differential and integral equations and in ocean acidification [Ph.D. thesis]*, University of Mauritius, 2010.
- [14] A. Cordero, J. L. Hueso, E. Martínez, and J. R. Torregrosa, "A modified Newton-Jarratt's composition," *Numerical Algorithms*, vol. 55, no. 1, pp. 87–99, 2010.
- [15] A. Cordero and J. R. Torregrosa, "Variants of Newton's method using fifth-order quadrature formulas," *Applied Mathematics and Computation*, vol. 190, no. 1, pp. 686–698, 2007.