

Article

A Quartically Convergent Jarratt-Type Method for Nonlinear System of Equations

Mohammad Ghorbanzadeh ¹ and Fazlollah Soleymani ^{2,*}

¹ Department of Mathematics, Imam Reza International University, Khorasan Razavi, Mashhad, Sanabaad, Daneshgah 91735-553, Iran; E-Mail: ghorbanzadeh_imamreza@yahoo.com

² Department of Mathematics, Zahedan Branch, Islamic Azad University, Zahedan, Iran

* Author to whom correspondence should be addressed;

E-Mail: fazlollah.soleymani@gmail.com; Tel. +98-9151401695.

Academic Editor: Alicia Cordero

Received: 28 May 2015 / Accepted: 2 July 2015 / Published: 6 July 2015

Abstract: In this work, we propose a new fourth-order Jarratt-type method for solving systems of nonlinear equations. The local convergence order of the method is proven analytically. Finally, we validate our results via some numerical experiments including an application to the Chandrashekar integral equations.

Keywords: iterative methods; Fréchet; systems of nonlinear equations; Chandrashekar integral equations

1. Introduction

A system of solving nonlinear equations by iterative methods is of interest to numerical analysts [1]. One of the popular methods is the classic multi-dimensional Newton method. It has quadratic convergence close to a simple zero, *i.e.*, the number of good digits is roughly doubled at each iteration.

Higher order methods, which require the second or higher order Fréchet derivatives can mostly be costly and thus time consuming. It is consequently important to study higher order variants of Newton method, which require only one more function or first-order derivative calculation and are more robust than the Newton method. Such methods are known as multi-point Newton-like methods in the Traub's sense [2].

Accordingly, it is an efficient way of generating higher order schemes free from second or higher order derivatives for solving systems of nonlinear equations. Such methods have been developed in [2]. For

more information, one may refer to [3,4]. For application of Newton-type methods in other problems, consult the papers [5,6].

In this work, we introduce the basic preliminaries. Then, we describe a third-order Newton-like method derived from quadrature rules for systems of nonlinear equations and discuss the disadvantage of the third-order methods in terms of efficiency index. We next extend a fourth-order Jarratt-type method from a third-order method for systems of nonlinear equations. We prove the local convergence of the method. We also show that the fourth-order method is more efficient than the second order Newton and a third-order method. Finally, we check the fourth-order convergence of the method through some numerical experiments.

2. Preliminaries

In this study, we use bold font style to represent vectors, matrices and tensors. Let $\mathbf{x} = (x_1, x_2, x_3, \dots, x_n)^T$, $\mathbf{x}^* = (x_1^*, x_2^*, x_3^*, \dots, x_n^*)^T$ and $\mathbf{F}(\mathbf{x}) = (f_1(x_1, \dots, x_n), f_2(x_1, \dots, x_n), \dots, f_n(x_1, \dots, x_n))^T$ be $n \times 1$ vectors. The Jacobian $\mathbf{F}'(\mathbf{x})$, is an $n \times n$ matrix.

The matrix multiplication $\mathbf{F}'(\mathbf{x})(\mathbf{x} - \mathbf{x}^*)$ is a $n \times 1$ vector. The Hessian $\mathbf{F}''(\mathbf{x})$ is a third-order tensor ($n \times n \times n$ matrix) and the notation $\mathbf{F}''(\mathbf{x})(\mathbf{x} - \mathbf{x}^*)^2$ means the matrix multiplication $\mathbf{F}''(\mathbf{x})(\mathbf{x} - \mathbf{x}^*)(\mathbf{x} - \mathbf{x}^*)$, which results in a $n \times 1$ vector as well. The same notations are applied to the higher order derivatives. Furthermore, we let

$$\mathbf{c}_j(\mathbf{x}) = \frac{1}{j!} \mathbf{F}'(\mathbf{x})^{-1} \mathbf{F}^{(j)}(\mathbf{x}), \quad j = 2, 3, \dots \tag{1}$$

which is $\underbrace{n \times \dots \times n}_{j \text{ times}}$ tensor.

It is well-known that the Newton method ($2^{nd}NM$) in multi-dimensional space is given by

$$\mathbf{x}^{(k+1)} = \mathbf{G}_{2^{nd}NM}(\mathbf{x}^{(k)}) = \mathbf{x}^{(k)} - \mathbf{u}(\mathbf{x}^{(k)}), \quad \text{where } \mathbf{u}(\mathbf{x}^{(k)}) = \mathbf{F}'(\mathbf{x}^{(k)})^{-1} \mathbf{F}(\mathbf{x}^{(k)}) \tag{2}$$

Research on systems of nonlinear equations has widely expanded over the last few decades [7,8]. As is well known, the iteration (2) and its variants coupled with some direct solution technique such as Gaussian elimination are good solvers for challenging nonlinear systems in case one has a sufficiently good initial guess $\mathbf{x}^{(0)}$ and the dimension of system is not too large. When the Jacobian is large and sparse, inexact Newton methods or high-order methods may be used. For further reading one may refer to [9].

3. Description of a New Method

Third-order methods free from second derivatives were proposed from quadrature rule for solving systems of nonlinear equations. These methods require one function evaluation and two first order derivatives at two points. One such method is:

$3^{rd}CON$ method derived from Closed-Open quadrature formula [10]:

$$\mathbf{x}^{(k+1)} = \mathbf{G}_{3^{rd}CON}(\mathbf{x}^{(k)}) = \mathbf{x}^{(k)} - \mathbf{A}(\mathbf{x}^{(k)})^{-1} \mathbf{F}(\mathbf{x}^{(k)}) \tag{3}$$

where

$$\mathbf{A}(\mathbf{x}) = \frac{\mathbf{F}'(\mathbf{x}) + 3\mathbf{F}'\left(\mathbf{x} - \frac{2}{3}\mathbf{u}(\mathbf{x})\right)}{4} \tag{4}$$

This method is also a member of Frontini-Sormani family of third order methods derived from quadrature rules [11]. The convergence analysis of this method using point of attraction theory can be found in [12]. This method is also more efficient than Halley’s method because it does not require the evaluation of a third order tensor of n^3 values.

Let p be the order of a method and d be defined as $d = d_0n + \sum_{j=1}^q d_j n^{j+1}$, where d_0 and d_j represent the number of times \mathbf{F} and $\mathbf{F}^{(j)}$ should be evaluated, respectively. The definition of the logarithm of Informational Efficiency or Efficiency Index for nonlinear systems (IE) [12] is given by

$$IE = \frac{\ln p}{d_0n + \sum_{j=1}^q d_j n^{j+1}} \tag{5}$$

Due to that the efficiency indices for the Newton’s method ($2^{nd}NM$) and the third-order method free from second derivatives ($3^{rd}CON$) are given by

$$IE_{2^{nd}NM} = \frac{\ln 2}{n + n^2} \quad \text{and} \quad IE_{3^{rd}CON} = \frac{\ln 3}{n + 2n^2} \tag{6}$$

respectively. We observe that $IE_{3^{rd}CON} > IE_{2^{nd}NM}$, 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.

Thus, it is suitable to develop a fourth-order method from the third-order method to improve the efficiency. For the scalar case, we can suggest the following quartical iteration, which is in fact a Jarratt-type iterative method including three functional evaluations to reach the highest possible order four [13]

$$\begin{cases} y_k = x_k - \frac{2}{3} \frac{f(x_k)}{f'(x_k)}, \\ x_{k+1} = x_k - \frac{4f(x_k)}{f'(x_k) + 3f'(y_k)} \left[1 + \frac{9}{16} \left(\frac{f'(y_k)}{f'(x_k)} - 1 \right)^2 \right] \end{cases} \tag{7}$$

We here build our efficient high order method according to (7) to reach the highest possible order using a fixed and the smallest possible number of functional evaluations. The improved fourth-order method ($4^{th}CON$) to systems of nonlinear equations can be constructed and suggested as follows

$$\mathbf{x}^{(k+1)} = \mathbf{G}_{4^{th}CON}(\mathbf{x}^{(k)}) = \mathbf{x}^{(k)} - \mathbf{H}(\mathbf{x}^{(k)})\mathbf{A}(\mathbf{x}^{(k)})^{-1}\mathbf{F}(\mathbf{x}^{(k)}) \tag{8}$$

where

$$\mathbf{H}(\tau(\mathbf{x})) = \mathbf{I} + \frac{9}{16}(\tau(\mathbf{x}) - \mathbf{I})^2, \quad \tau(\mathbf{x}) = \mathbf{F}'(\mathbf{x})^{-1}\mathbf{F}'\left(\mathbf{x} - \frac{2}{3}\mathbf{u}(\mathbf{x})\right)$$

and \mathbf{I} is the $n \times n$ identity matrix. We also note that this method is an improvement of the $3^{th}CON$ method because it can also be written as:

$$\mathbf{G}_{4^{th}CON}(\mathbf{x}^{(k)}) = \mathbf{x}^{(k)} + \mathbf{H}(\mathbf{x}^{(k)}) \left(\mathbf{G}_{3^{th}CON}(\mathbf{x}^{(k)}) - \mathbf{x}^{(k)} \right) \tag{9}$$

We describe the algorithm of $4^{th}CON$ method, which requires the evaluations of one function with n values, two Jacobians of n^2 values each and the inversion of two matrices as comes next:

Algorithm 4thCON

$$\begin{cases} \mathbf{F}'(\mathbf{x}^{(k)})\mathbf{u}(\mathbf{x}^{(k)}) = \mathbf{F}(\mathbf{x}^{(k)}) \\ \mathbf{y}^{(k)} = \mathbf{x}^{(k)} - \frac{2}{3}\mathbf{u}(\mathbf{x}^{(k)}) \\ \mathbf{A}(\mathbf{x}^{(k)}) = \frac{\mathbf{F}'(\mathbf{x}^{(k)})+3\mathbf{F}'(\mathbf{y}^{(k)})}{4} \\ \mathbf{F}'(\mathbf{x}^{(k)})\tau(\mathbf{x}^{(k)}) = \mathbf{F}'(\mathbf{y}^{(k)}) \\ \mathbf{A}(\mathbf{x}^{(k)})\mathbf{h}(\mathbf{x}^{(k)}) = \mathbf{F}(\mathbf{x}^{(k)}) \\ \mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} - \left(\mathbf{I} + \frac{9}{16}(\tau(\mathbf{x}^{(k)}) - \mathbf{I})^2\right) \mathbf{h}(\mathbf{x}^{(k)}) \end{cases} \tag{10}$$

Note that the efficiency index of the fourth order method free from second derivatives (4thCON) is given by

$$IE_{4^{th}CON} = \frac{\ln 4}{n + 2n^2}, \quad \frac{IE_{4^{th}CON}}{IE_{2^{nd}NM}} = \frac{2n + 2}{2n + 1} > 1, \quad n \geq 1 \tag{11}$$

This shows that the 4thCON method is better than 2ndNM and 3rdCON methods.

4. Convergence Analysis

The local convergence of the 4thCON is proved in Theorem 1 and it is as follows.

Theorem 1. Let $\mathbf{F} : \mathbf{D} \subset \mathbb{R}^n \rightarrow \mathbb{R}^n$, be four times Fréchet differentiable in a convex set \mathbf{D} of $\mathbf{F}(\mathbf{x}) = \mathbf{0}$. Then the scheme defined by Equation (10) has order of convergence 4.

Proof. Considering the same notation and definitions given above, and by Taylor’s series around \mathbf{x} ,

$$\begin{aligned} \mathbf{F}(\mathbf{x}^*) &= \mathbf{F}(\mathbf{x}) + \mathbf{F}'(\mathbf{x})(\mathbf{x}^* - \mathbf{x}) + \frac{1}{2}\mathbf{F}^{(2)}(\mathbf{x})(\mathbf{x}^* - \mathbf{x})^2 \\ &\quad + \frac{1}{3!}\mathbf{F}^{(3)}(\mathbf{x})(\mathbf{x}^* - \mathbf{x})^3 + \frac{1}{4!}\mathbf{F}^{(4)}(\mathbf{x})(\mathbf{x}^* - \mathbf{x})^4 + O(\|\mathbf{x}^* - \mathbf{x}\|^5) \end{aligned} \tag{12}$$

Since $\mathbf{F}(\mathbf{x}^*) = \mathbf{0}$ and $\mathbf{e} = \mathbf{x} - \mathbf{x}^*$, Equation (12) can be simplified to

$$\mathbf{F}(\mathbf{x}) = \mathbf{F}'(\mathbf{x}) (\mathbf{e} - \mathbf{c}_2(\mathbf{x})\mathbf{e}^2 + \mathbf{c}_3(\mathbf{x})\mathbf{e}^3 - \mathbf{c}_4(\mathbf{x})\mathbf{e}^4) + O(\|\mathbf{e}\|^5) \tag{13}$$

where $\mathbf{c}_i = \frac{1}{i!}\mathbf{F}'(\mathbf{x}^*)^{-1}\mathbf{F}^{(i)}(\mathbf{x}^*)$, $\mathbf{F}^{(i)}(\mathbf{x}^*) \in \mathcal{L}(\mathbb{R}^n, \dots, \mathbb{R}^n)$, $\mathbf{e}^p = \left(\overbrace{\mathbf{e}, \dots, \mathbf{e}}^p\right)$ and $\mathbf{e} \in \mathbb{R}$. Note that \mathbb{L} denotes the set of bounded linear functions. Therefore,

$$\mathbf{u}(\mathbf{x}) = \mathbf{F}'(\mathbf{x})^{-1}\mathbf{F}(\mathbf{x}) = \mathbf{e} - \mathbf{c}_2(\mathbf{x})\mathbf{e}^2 + \mathbf{c}_3(\mathbf{x})\mathbf{e}^3 - \mathbf{c}_4(\mathbf{x})\mathbf{e}^4 + O(\|\mathbf{e}\|^5) \tag{14}$$

Applying Equation (14), we come by

$$\begin{aligned} \mathbf{F}'(\mathbf{x} - \frac{2}{3}\mathbf{u}(\mathbf{x})) &= \mathbf{F}'(\mathbf{x}) \left(\mathbf{I} - \frac{4}{3}\mathbf{c}_2(\mathbf{x})\mathbf{e} + \frac{4}{3}(\mathbf{c}_2(\mathbf{x})^2 + \mathbf{c}_3(\mathbf{x}))\mathbf{e}^2 \right. \\ &\quad \left. - (4\mathbf{c}_2(\mathbf{x})\mathbf{c}_3(\mathbf{x}) + \frac{32}{27}\mathbf{c}_4(\mathbf{x}))\mathbf{e}^3 \right) + O(\|\mathbf{e}\|^4) \end{aligned} \tag{15}$$

and

$$\mathbf{A}(\mathbf{x}) = \mathbf{F}'(\mathbf{x}) \left(\mathbf{I} - \mathbf{c}_2(\mathbf{x})\mathbf{e} + (\mathbf{c}_2(\mathbf{x})^2 + \mathbf{c}_3(\mathbf{x}))\mathbf{e}^2 - \left(3\mathbf{c}_2(\mathbf{x})\mathbf{c}_3(\mathbf{x}) + \frac{8}{9}\mathbf{c}_4(\mathbf{x})\right)\mathbf{e}^3 \right) + O(\|\mathbf{e}\|^4) \tag{16}$$

Using Equations (15) and (16), we can obtain the error equation of 3thCON.

$$\mathbf{A}(\mathbf{x})(\mathbf{G}_{3^{th}CON}(\mathbf{x}) - \mathbf{x}^*) = \mathbf{F}'(\mathbf{x})\left(\mathbf{c}_2(\mathbf{x})^2\mathbf{e}^3 + \left(-3\mathbf{c}_2(\mathbf{x})\mathbf{c}_3(\mathbf{x}) + \frac{1}{9}\mathbf{c}_4(\mathbf{x})\right)\mathbf{e}^4\right) + O(\|\mathbf{e}\|^5) \quad (17)$$

Using Equation (15). we have

$$\tau(\mathbf{x}) = \mathbf{I} - \frac{4}{3}\mathbf{c}_2(\mathbf{x})\mathbf{e} + \frac{4}{3}(\mathbf{c}_2(\mathbf{x})^2 + \mathbf{c}_3(\mathbf{x}))\mathbf{e}^2 - \left(4\mathbf{c}_2(\mathbf{x})\mathbf{c}_3(\mathbf{x}) + \frac{32}{27}\mathbf{c}_4(\mathbf{x})\right)\mathbf{e}^3 + O(\|\mathbf{e}\|^4) \quad (18)$$

and subsequently

$$\mathbf{H}(\tau(\mathbf{x})) = \mathbf{I} + \frac{9}{16}(\tau(\mathbf{x}) - \mathbf{I})^2 = \mathbf{I} + \mathbf{c}_2(\mathbf{x})^2\mathbf{e}^2 - 2(\mathbf{c}_2(\mathbf{x})^3 + \mathbf{c}_2(\mathbf{x})\mathbf{c}_3(\mathbf{x}))\mathbf{e}^3 + O(\|\mathbf{e}\|^4) \quad (19)$$

Now,

$$\begin{aligned} \mathbf{G}_{4^{th}CON}(\mathbf{x}) - \mathbf{x}^* &= \mathbf{x} - \mathbf{x}^* + \mathbf{H}(\tau(\mathbf{x}))(\mathbf{G}_{3^{th}CON}(\mathbf{x}) - \mathbf{x}) \\ &= (\mathbf{I} - \mathbf{H}(\tau(\mathbf{x}))\mathbf{e} + \mathbf{H}(\tau(\mathbf{x}))(\mathbf{G}_{3^{th}CON}(\mathbf{x}) - \mathbf{x}^*)) \end{aligned} \quad (20)$$

since $\mathbf{G}_{3^{th}CON}(\mathbf{x}) - \mathbf{x} = \mathbf{G}_{3^{th}CON}(\mathbf{x}) - \mathbf{x}^* - \mathbf{e}$. Using Equations (19) and (20), we have

$$\begin{aligned} \mathbf{H}(\tau(\mathbf{x}))(\mathbf{G}_{3^{th}CON}(\mathbf{x}) - \mathbf{x}^*) &= (\mathbf{G}_{3^{th}CON}(\mathbf{x}) - \mathbf{x}^*) + O(\|\mathbf{e}\|^2 \times \|\mathbf{G}_{3^{th}CON}(\mathbf{x}) - \mathbf{x}^*\|) \\ &= \mathbf{A}(\mathbf{x})^{-1}\mathbf{F}'(\mathbf{x})\left(\mathbf{c}_2(\mathbf{x})^2\mathbf{e}^3 + \left(-3\mathbf{c}_2(\mathbf{x})\mathbf{c}_3(\mathbf{x}) + \frac{1}{9}\mathbf{c}_4(\mathbf{x})\right)\mathbf{e}^4\right) + O(\|\mathbf{e}\|^5) \end{aligned} \quad (21)$$

Furthermore, we acquire

$$(\mathbf{I} - \mathbf{H}(\tau(\mathbf{x}))\mathbf{e} + \mathbf{H}(\tau(\mathbf{x}))(\mathbf{G}_{3^{th}CON}(\mathbf{x}) - \mathbf{x}^*)) = \mathbf{A}(\mathbf{x})^{-1}\left(\mathbf{A}(\mathbf{x})(\mathbf{I} - \mathbf{H}(\tau(\mathbf{x}))\mathbf{e})\mathbf{F}'(\mathbf{x})\left(\mathbf{c}_2(\mathbf{x})^2\mathbf{e}^3 + \left(-3\mathbf{c}_2(\mathbf{x})\mathbf{c}_3(\mathbf{x}) + \frac{1}{9}\mathbf{c}_4(\mathbf{x})\right)\mathbf{e}^4\right) + O(\|\mathbf{e}\|^5)\right) \quad (22)$$

From Equations (21) and (22), we come by

$$\mathbf{A}(\mathbf{x})(\mathbf{I} - \mathbf{H}(\tau(\mathbf{x}))\mathbf{e}) = \mathbf{F}'(\mathbf{x})\left(-\mathbf{c}_2(\mathbf{x})^2\mathbf{e}^3 + (3\mathbf{c}_2(\mathbf{x})^3 + 2\mathbf{c}_2(\mathbf{x})\mathbf{c}_3(\mathbf{x}))\mathbf{e}^4\right) + O(\|\mathbf{e}\|^5) \quad (23)$$

Substituting Equation (23) into Equation (8), we obtain

$$\mathbf{A}(\mathbf{x})(\mathbf{G}_{4^{th}CON}(\mathbf{x}) - \mathbf{x}^*) = \mathbf{F}'(\mathbf{x})\left(3\mathbf{c}_2(\mathbf{x})^3 - \mathbf{c}_2(\mathbf{x})\mathbf{c}_3(\mathbf{x}) + \frac{1}{9}\mathbf{c}_4(\mathbf{x})\right)\mathbf{e}^4 + O(\|\mathbf{e}\|^5),$$

which establishes the fourth order convergence of the method. The proof is now complete.

5. Numerical Examples

In this section, we compare the performance of the contributed method with that of (2), and (3). The algorithms were written in Matlab 7.6 and tested for the examples given below.

We start with small systems of nonlinear equations. For the following test problems, the approximate solutions are calculated up to 500 digits using the variable arithmetic precision in Matlab 7.6. We define

$$err = \|F(\mathbf{x}^{(k)})\|_2 + \|\mathbf{x}^{(k+1)} - \mathbf{x}^{(k)}\|_2 < 1e - 100 \quad (24)$$

We use the approximate computational order of convergence p (see [14]) given by

$$p_c \approx \frac{\log(\|\mathbf{x}^{(k+1)} - \mathbf{x}^{(k)}\|)/(\|\mathbf{x}^{(k)} - \mathbf{x}^{(k-1)}\|)}{\log(\|\mathbf{x}^{(k)} - \mathbf{x}^{(k-1)}\|)/(\|\mathbf{x}^{(k-1)} - \mathbf{x}^{(k-2)}\|)} \tag{25}$$

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

Test Problem 1 (TP1) [5] $\mathbf{F}(x_1, x_2) = \mathbf{0}$, where $\mathbf{F} : (4, 6) \times (5, 7) \rightarrow \mathbb{R}^2$, and $\mathbf{F}(x_1, x_2) = (x_1^2 - x_2 - 19, x_2^3/6 - x_1^2 + x_2 - 17)$. The starting vector is $\mathbf{x}^{(0)} = (5.1, 6.1)^T$ and the exact solution is $\mathbf{x}^* = (5, 6)^T$. In addition, it is easy to see that $IE_{2^{nd}NM} = \frac{\ln 2}{2+4} = 0.1155$, $IE_{3^{th}CON} = \frac{\ln 3}{2+2(4)} = 0.1099$, $IE_{4^{th}CON} = \frac{\ln 4}{2+2(4)} = 0.1386$.

Test Problem 2 (TP2) [12] The test is as follows

$$\begin{aligned} \cos x_2 - \sin x_1 &= 0 \\ x_3^{x_1} - \frac{1}{x_2} &= 0 \\ \exp x_1 - x_3^2 &= 0 \end{aligned} \tag{26}$$

with the following solution $x_1^* = 0.9095694945200\dots$, $x_2^* = 0.6612268322748\dots$, $x_3^* = 1.5758341439070\dots$. We choose the starting vector $\mathbf{x}^{(0)} = (1, 0.5, 1.5)^T$. Here, we have

$$IE_{2^{nd}NM} = \frac{\ln 2}{12} = 0.0577, \quad IE_{3^{th}CON} = \frac{\ln 3}{21} = 0.0523, \quad IE_{4^{th}CON} = \frac{\ln 4}{21} = 0.0660$$

Systems of four nonlinear equations:

Test Problem 3 (TP3) (see [12]) This example is in what follows:

$$\begin{aligned} x_2x_3 + x_4(x_2 + x_3) &= 0 \\ x_1x_3 + x_4(x_1 + x_3) &= 0 \\ x_1x_2 + x_4(x_1 + x_2) &= 0 \\ x_1x_2 + x_1x_3 + x_2x_3 &= 1 \end{aligned} \tag{27}$$

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

$$\begin{aligned} x_1^* &= 0.57735026918963\dots & x_2^* &= 0.57735026918963\dots \\ x_3^* &= 0.57735026918963\dots & x_4^* &= -0.28867513459482\dots \end{aligned}$$

Note that

$$IE_{2^{nd}NM} = \frac{\ln 2}{4 + 16} = 0.0346, \quad IE_{3^{th}CON} = \frac{\ln 3}{4 + 2(16)} = 0.0305, \quad IE_{4^{th}CON} = \frac{\ln 4}{2 + 2(16)} = 0.0385$$

Table 1 gives the results for the TP1-3. It is observed that for all problems the fourth-order method converges in the least number of iterations. The computational order of convergence agrees with the theory. $4^{th}CON$ gives the best results in terms of least residual and is the most efficient method.

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

Methods	TP1			TP2			TP3		
	<i>Iter_{max}</i>	<i>err_{min}</i>	<i>p_c</i>	<i>Iter_{max}</i>	<i>err_{min}</i>	<i>p_c</i>	<i>Iter_{max}</i>	<i>err_{min}</i>	<i>p_c</i>
2 nd NM	7	8.8e−113	2	9	1.7e−107	2	8	1.7e−144	2.02
3 rd CON	5	3.2e−143	3.02	7	7.3e−285	3	6	6.3e−276	3.04
4 th CON	4	5.0e−104	4.02	6	2.7e−322	4	5	5.0e−246	4.14

Application in Integral Equations

The Chandrasekhar-*H* equation arising in Radiative Heat Transfer theory (see [15]) is a nonlinear integral equation which gives a full nonlinear system of equations if discretized. The Chandrasekhar-*H* equation is given as

$$\mathcal{J}(H, c) = 0, \quad H : [0, 1] \rightarrow \mathbb{R} \tag{28}$$

with parameter *c* and the operator \mathcal{J} as

$$\mathcal{J}(H, c)(U) = H(U) - \left(1 - \frac{c}{2} \int_0^1 \frac{U H(v)}{U + v} dv \right)^{-1} \tag{29}$$

If we discretize the integral given in Equation (29) using the Mid-point Integration Rule with *n* grid points

$$\int_0^1 f(t) dt = \frac{1}{n} \sum_{j=1}^n f(t_j), \quad t_j = (j - 0.5)h, \quad h = \frac{1}{n}, \quad 1 \leq j \leq n \tag{30}$$

we obtain the resulting system of non-linear equations as follows:

$$\mathbf{F}_i(\mathbf{U}, c) = \mathbf{U}_i - \left(1 - \frac{c}{2n} \sum_{j=1}^n \frac{t_i \mathbf{U}_i}{t_i + t_j} \right)^{-1}, \quad 1 \leq i \leq n \tag{31}$$

When starting with $(1, 1, \dots, 1)^T$ vector, the system (31) has a solution for all $c \in (0, 1)$. The *c* are equally spaced with $\Delta c = 0.01$ in the interval $c \in (0, 1)$ and we choose $n = 100$. The approximate solutions are correct to 14 digits. We note that in this case the Jacobian is a full matrix. The stopping criterion is when, $\|\mathbf{U}^{(k+1)} - \mathbf{U}^{(k)}\|_2 + \|\mathbf{F}(\mathbf{U}^{(k+1)})\|_2 < 1(-11)$. Table 2 shows the result for the Chandrasekhar *H*-equation by implementation of our codes in double precision arithmetic.

Table 2. Key results for the Chandrasekhar *H*-equation.

Methods	<i>Iter_{Total}</i>	<i>Iter_{mean}</i>	CPU Time (s)
2 nd NM	414	4.18	23.28
3 rd CON	361	3.65	27.08
4 th CON	293	2.99	23.01

6. Concluding Remarks

We have extended a fourth-order Jarratt-type method from a third-order method for systems of nonlinear equations. The local convergence of the fourth-order method has been proved. We have shown that the quartical iterative method is more efficient than the second order Newton and a third-order method.

As future works, we would like to improve the order of the fourth-order methods through an additional function evaluation in a three-step cycle to achieve higher orders and better efficiencies. Finally, we state that the question: “can the idea of with memorization, (see e.g., [16]), be incorporated into the new schemes?” would also be considered for future studies.

Author Contributions

The contributions of all of the authors have been similar. All of them have worked together to develop the present manuscript.

Conflicts of Interest

The authors declare no conflict of interest.

References

1. Cordero, A.; Franques, A.; Torregrosa, J.R. Numerical Solution of Turbulence Problems by Solving Burgers' Equation. *Algorithms* **2015**, *8*, 224–233.
2. Traub, J.F. *Iterative Methods for the Solution of Equations*; Publishing Company: New York, NY, USA, 1982.
3. Soleymani, F. A three-step iterative method for nonlinear systems with sixth order of convergence. *Int. J. Comput. Sci. Math.* **2013**, *4*, 363–373.
4. Ullah, M.Z.; Soleymani, F.; Al-Fhaid, A.S. Numerical solution of nonlinear systems by a general class of iterative methods with application to nonlinear PDEs. *Numer. Algor.* **2014**, *67*, 223–242.
5. Soleymani, F. An efficient and stable Newton-type iterative method for computing generalized inverse $A_{T,S}^{(2)}$. *Numer. Algor.* **2015**, *69*, 569–578.
6. Soleymani, F.; Salmani, H.; Rasouli, M. Finding the Moore-Penrose inverse by a new matrix iteration. *J. Appl. Math. Comput.* **2015**, *47*, 33–48.
7. Babajee, D.K.R.; Dauhoo, M.Z.; Darvishi, M.T.; Bharati, A. A Karami, Analysis of two Chebyshev-like third order methods free from second derivatives for solving systems of nonlinear equations. *J. Comp. Appl. Math.* **2010**, *233*, 2002–2012.
8. Cordero, A.; Hueso, J.L.; Martinez, E.; Torregrosa, J.R. Accelerated methods of order $2p$ for systems of nonlinear equations. *J. Comput. Appl. Math.* **2010**, *233*, 2696–2702.
9. Cordero, A.; Soleymani, F.; Torregrosa, J.R. Dynamical analysis of iterative methods for nonlinear systems or how to deal with the dimension? *Appl. Math. Comput.* **2014**, *244*, 398–412.
10. Noor, M.A.; Waseem, M. Some iterative methods for solving a system of nonlinear equations. *Comput. Math. Appl.* **2009**, *57*, 101–106.

11. Frontini, M.; Sormani, E. Third-order methods from quadrature formulae for solving systems of nonlinear equations. *Appl. Math. Comput.* **2004**, *140*, 771–782.
12. Babajee, D.K.R. 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, Réduit, Moka, Mauritius, December 2010.
13. Sharifi, M.; Babajee, D.K.R.; Soleymani, F. Finding the solution of nonlinear equations by a class of optimal methods. *Comput. Math. Appl.* **2012**, *63*, 764–774.
14. Cordero, A.; Torregrosa, J.R. Variants of Newton's Method using fifth-order quadrature formulas. *Appl. Math. Comput.* **2007**, *190*, 686–698.
15. Kelley, C.T. Solution of the Chandrasekhar H -equation by Newton's method. *J. Math. Phys.* **1980**, *21*, 1625–1628.
16. Lotfi, T.; Mahdiani, K.; Bakhtiari, P.; Soleymani, F. Constructing two step iterative methods with and without memory. *Comput. Math. Math. Phys.* **2015**, *55*, 183–193.

© 2015 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 license (<http://creativecommons.org/licenses/by/4.0/>).