Computers and Mathematics with Applications 57 (2009) 101–106

Contents lists available at ScienceDirect

Computers and Mathematics with Applications

journal homepage: www.elsevier.com/locate/camwa

Some iterative methods for solving a system of nonlinear equations

Muhammad Aslam Noor *, Muhammad Waseem

Department of Mathematics, COMSATS Institute of Information Technology, Islamabad 44000, Pakistan

ARTICLE INFO

Article history:
Received 27 March 2008
Received in revised form 4 September 2008
Accepted 16 October 2008

Keywords:
Nonlinear equations
Quadrature formulas
Iterative methods
Efficiency index
Examples

ABSTRACT

In this paper, we suggest and analyze two new two-step iterative methods for solving the system of nonlinear equations using quadrature formulas. We prove that these new methods have cubic convergence. Several numerical examples are given to illustrate the efficiency and the performance of the new iterative methods. These new iterative methods may be viewed as an extension and generalizations of the existing methods for solving the system of nonlinear equations.

1. Introduction

Consider the system of nonlinear equations

$$\begin{aligned} &f_1(x_1, x_2, \ldots, x_n) = 0,\\ &f_2(x_1, x_2, \ldots, x_n) = 0,\\ &\vdots\\ &f_n(x_1, x_2, \ldots, x_n) = 0, \end{aligned} \tag{1.1}$$

where each function f_i maps a vector $\mathbf{x} = (x_1, x_2, \ldots, x_n)^t$ of the n-dimensional space $\mathcal{R}^n$ into the real line $\mathcal{R}$. The system (1.1) of n nonlinear equations in n unknowns can also be represented by defining a function F mapping $\mathcal{R}^n$ into $\mathcal{R}^n$ as

$$\mathbf{F}(\mathbf{x}) = (f_1(\mathbf{x}), f_2(\mathbf{x}), \ldots, f_n(\mathbf{x}))^t. \tag{1.2}$$

Thus, the system (1.1) can be written in the form $\mathbf{F}(\mathbf{x}) = 0$, where the functions $f_1, f_2, \ldots, f_n$ are the coordinate functions of $\mathbf{F}$. In recent years, several iterative methods have been developed to solve the nonlinear system of equations $\mathbf{F}(\mathbf{x}) = 0$, by using essentially Taylor's polynomial [1,2], decomposition [3–6], homotopy perturbation method [7], quadrature formulas [6,8–12,19,22,23] and other techniques [2,13–21]. It is well known that the quadrature rules play an important and significant rule in the evaluation of the integrals [1,22–24]. It has been shown [6,8–12] that the quadrature formulas have been used to develop some iterative methods for solving a system of nonlinear equations. Motivated and inspired by the on-going activities in this direction, we suggest and analyze two new iterative methods for solving the nonlinear system of equations by using quadrature formulas. These methods are implicit-type methods. To implement these methods, we use Newton's method as predictor and then use the new method as a corrector. It has been shown that these two-step iterative methods are cubically convergent. A comparison between these new methods with that of Newton's method, the method of Cordero and Torregrosa [10] and the method of Darvishi and Barati [5] is given. Several numerical examples are given to illustrate the efficiency and the performance of the new iterative methods. Our results can be viewed as an improvement and refinement of the previously known results.

* Corresponding author.
E-mail addresses: noormaslam@hotmail.com (M. Aslam Noor), waseemsattar@hotmail.com (M. Waseem).

0898-1221/$ – see front matter
doi:10.1016/j.camwa.2008.10.067

2. Iterative methods

Let $\mathbf{F} : K \subseteq \mathcal{R}^n \to \mathcal{R}^n$, be r-times Fréchet differentiable function on a convex set $K \subseteq \mathcal{R}^n$ and $\boldsymbol{\alpha}$ be a real zero of the nonlinear system of equations $\mathbf{F}(\mathbf{x}) = 0$ of n equations with n variables. For any $\mathbf{x}, \mathbf{x}_k \in K$, we may write (see [2]) the Taylor's expansion for $\mathbf{F}$ as follows:

$$\begin{aligned}\mathbf{F}(\mathbf{x}) &= \mathbf{F}(\mathbf{x}_k) + \mathbf{F}'(\mathbf{x}_k)(\mathbf{x}-\mathbf{x}_k) + \frac{1}{2!}\mathbf{F}''(\mathbf{x}_k)(\mathbf{x}-\mathbf{x}_k)^2 + \frac{1}{3!}\mathbf{F}'''(\mathbf{x}_k)(\mathbf{x}-\mathbf{x}_k)^3 + \cdots \\ &\quad + \frac{1}{(r-1)!}\mathbf{F}^{(r-1)}(\mathbf{x}_k)(\mathbf{x}-\mathbf{x}_k)^{r-1} + \int_0^1 \frac{(1-t)^{r-1}}{(r-1)!}\mathbf{F}^{(r)}(\mathbf{x}_k + t(\mathbf{x}-\mathbf{x}_k))(\mathbf{x}-\mathbf{x}_k)^r \, \mathrm{d}t. \end{aligned} \tag{2.1}$$

For $r = 1$, we have

$$\mathbf{F}(\mathbf{x}) = \mathbf{F}(\mathbf{x}_k) + \int_0^1 \mathbf{F}'(\mathbf{x}_k + t(\mathbf{x}-\mathbf{x}_k))(\mathbf{x}-\mathbf{x}_k)\,\mathrm{d}t. \tag{2.2}$$

Approximating the integral in (2.2), we have

$$\int_0^1 \mathbf{F}'(\mathbf{x}_k + t(\mathbf{x}-\mathbf{x}_k))(\mathbf{x}-\mathbf{x}_k)\,\mathrm{d}t \cong \mathbf{F}'(\mathbf{x}_k)(\mathbf{x}-\mathbf{x}_k). \tag{2.3}$$

By using (2.3) in (2.2) and $\mathbf{F}(\mathbf{x}) = 0$, we have

$$\mathbf{x} = \mathbf{x}_k - \mathbf{F}'(\mathbf{x}_k)^{-1}\mathbf{F}(\mathbf{x}_k).$$

This allows us to suggest the following one-step iterative method for solving the system of nonlinear equations (1.1).

Algorithm 2.1. For a given $\mathbf{x}_0$, compute the approximate solution $\mathbf{x}_{k+1}$ by the iterative scheme

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

where $\mathbf{F}'(\mathbf{x}_k)$ is the Jacobian matrix at the point $\mathbf{x}_k$. Algorithm 2.1 is known as Newton's method for nonlinear system of equations $\mathbf{F}(\mathbf{x}) = 0$ and has quadratic convergence [2].

If we approximate the integral in (2.2) by using the Closed-Open quadrature formula [24], then

$$\int_0^1 \mathbf{F}'(\mathbf{x}_k + t(\mathbf{x}-\mathbf{x}_k))(\mathbf{x}-\mathbf{x}_k)\,\mathrm{d}t \cong \frac{1}{4}\left[\mathbf{F}'(\mathbf{x}_k) + 3\mathbf{F}'\left(\frac{\mathbf{x}_k + 2\mathbf{x}}{3}\right)\right](\mathbf{x}-\mathbf{x}_k). \tag{2.4}$$

By using (2.4) in (2.2) and $\mathbf{F}(\mathbf{x}) = 0$, we obtain

$$\mathbf{x} = \mathbf{x}_k - 4\left[\mathbf{F}'(\mathbf{x}_k) + 3\mathbf{F}'\left(\frac{\mathbf{x}_k + 2\mathbf{x}}{3}\right)\right]^{-1}\mathbf{F}(\mathbf{x}_k).$$

Using this relation, we can suggest the following two-step iterative method for solving the nonlinear system of Eq. (1.1) as:

Algorithm 2.2. For a given $\mathbf{x}_0$, compute the approximate solution $\mathbf{x}_{k+1}$ by the iterative schemes

Predictor step:

$$\mathbf{y}_k = \mathbf{x}_k - \mathbf{F}'(\mathbf{x}_k)^{-1}\mathbf{F}(\mathbf{x}_k).$$

Corrector step:

$$\mathbf{x}_{k+1} = \mathbf{x}_k - 4\left[\mathbf{F}'(\mathbf{x}_k) + 3\mathbf{F}'\left(\frac{\mathbf{x}_k + 2\mathbf{y}_k}{3}\right)\right]^{-1}\mathbf{F}(\mathbf{x}_k), \quad k = 0, 1, 2, \ldots.$$

Algorithm (2.2) is our newly derived method in this paper for solving the system of nonlinear equations and is one of the main motivations of this paper.

In a similar way, approximating the integral in (2.2) by using the Open-Closed quadrature formula [24], we have

$$\int_0^1 \mathbf{F}'(\mathbf{x}_k + t(\mathbf{x}-\mathbf{x}_k))(\mathbf{x}-\mathbf{x}_k)\,\mathrm{d}t \cong \frac{1}{4}\left[3\mathbf{F}'\left(\frac{2\mathbf{x}_k + \mathbf{x}}{3}\right) + \mathbf{F}'(\mathbf{x})\right](\mathbf{x}-\mathbf{x}_k). \tag{2.5}$$

By using (2.5) in (2.2) and $\mathbf{F}(\mathbf{x}) = 0$, we get

$$\mathbf{x} = \mathbf{x}_k - 4\left[3\mathbf{F}'\left(\frac{2\mathbf{x}_k + \mathbf{x}}{3}\right) + \mathbf{F}'(\mathbf{x})\right]^{-1}\mathbf{F}(\mathbf{x}_k).$$

Using this relation, we can suggest the following two-step iterative method for solving the nonlinear system of Eq. (1.1) as:

Algorithm 2.3. For a given $\mathbf{x}_0$, compute the approximate solution $\mathbf{x}_{k+1}$ by the iterative schemes
Predictor step:

$$\mathbf{y}_k = \mathbf{x}_k - \mathbf{F}'(\mathbf{x}_k)^{-1}\mathbf{F}(\mathbf{x}_k).$$

Corrector step:

$$\mathbf{x}_{k+1} = \mathbf{x}_k - 4\left[3\mathbf{F}'\left(\frac{2\mathbf{x}_k+\mathbf{y}_k}{3}\right)+\mathbf{F}'(\mathbf{y}_k)\right]^{-1}\mathbf{F}(\mathbf{x}_k), \quad k = 0, 1, 2, \ldots.$$

Algorithm (2.3) is another iterative method for solving a system of nonlinear equations (1.1).

3. Convergence analysis

In this section, we consider the convergence criteria of the Algorithm 2.2 using the Taylor series technique, see [5,6,11,12].

Theorem 3.1. *Let $\mathbf{F}: K \subseteq \mathcal{R}^n \to \mathcal{R}^n$, be r-times Fréchet differentiable function on a convex set K containing the root $\boldsymbol{\alpha}$ of $\mathbf{F}(\mathbf{x}) = 0$. The iterative method defined by Algorithm 2.2 has cubic convergence and satisfies the error equation*

$$\left[\mathbf{F}'(\mathbf{x}_k)+3\mathbf{F}'\left(\frac{\mathbf{x}_k+2\mathbf{y}_k}{3}\right)\right]\boldsymbol{e}_{k+1} = \left[\mathbf{F}''(\mathbf{x}_k)\mathbf{F}'(\mathbf{x}_k)^{-1}\mathbf{F}''(\mathbf{x}_k)\right]\boldsymbol{e}_k^3 + \mathbf{O}\left(\|\boldsymbol{e}_k^4\|\right).$$

Proof. The technique is given by

$$\mathbf{y}_k = \mathbf{x}_k - \mathbf{F}'(\mathbf{x}_k)^{-1}\mathbf{F}(\mathbf{x}_k) \tag{3.1}$$

$$\mathbf{x}_{k+1} = \mathbf{x}_k - 4\left[\mathbf{F}'(\mathbf{x}_k)+3\mathbf{F}'\left(\frac{\mathbf{x}_k+2\mathbf{y}_k}{3}\right)\right]^{-1}\mathbf{F}(\mathbf{x}_k). \tag{3.2}$$

Defining $\boldsymbol{e}_k = \mathbf{x}_k - \boldsymbol{\alpha}$ and from Eq. (3.2), we have

$$\boldsymbol{e}_{k+1} - \boldsymbol{e}_k = -4\left[\mathbf{F}'(\mathbf{x}_k)+3\mathbf{F}'\left(\frac{\mathbf{x}_k+2\mathbf{y}_k}{3}\right)\right]^{-1}\mathbf{F}(\mathbf{x}_k),$$

from which, we have

$$\left[\mathbf{F}'(\mathbf{x}_k)+3\mathbf{F}'\left(\frac{\mathbf{x}_k+2\mathbf{y}_k}{3}\right)\right]\boldsymbol{e}_{k+1} = \left[\mathbf{F}'(\mathbf{x}_k)+3\mathbf{F}'\left(\frac{\mathbf{x}_k+2\mathbf{y}_k}{3}\right)\right]\boldsymbol{e}_k - 4\mathbf{F}(\mathbf{x}_k). \tag{3.3}$$

From Eq. (2.1) with $\mathbf{x} = \boldsymbol{\alpha}$, we obtain

$$\mathbf{F}(\boldsymbol{\alpha}) = \mathbf{F}(\mathbf{x}_k) + \mathbf{F}'(\mathbf{x}_k)(\boldsymbol{\alpha}-\mathbf{x}_k) + \frac{1}{2!}\mathbf{F}''(\mathbf{x}_k)(\boldsymbol{\alpha}-\mathbf{x}_k)^2 + \frac{1}{3!}\mathbf{F}'''(\mathbf{x}_k)(\boldsymbol{\alpha}-\mathbf{x}_k)^3 + \mathbf{O}\left(\|\boldsymbol{\alpha}-\mathbf{x}_k\|^4\right).$$

If $\boldsymbol{\alpha}$ be the zero of the nonlinear system of equations $\mathbf{F}(\mathbf{x}) = 0$, then we have

$$\mathbf{F}(\mathbf{x}_k) = \mathbf{F}'(\mathbf{x}_k)\boldsymbol{e}_k - \frac{1}{2!}\mathbf{F}''(\mathbf{x}_k)\boldsymbol{e}_k^2 + \frac{1}{3!}\mathbf{F}'''(\mathbf{x}_k)\boldsymbol{e}_k^3 + \mathbf{O}\left(\|\boldsymbol{e}_k^4\|\right). \tag{3.4}$$

Pre-multiplying Eq. (3.4) by $\mathbf{F}'(\mathbf{x}_k)^{-1}$, we obtain

$$\mathbf{F}'(\mathbf{x}_k)^{-1}\mathbf{F}(\mathbf{x}_k) = \boldsymbol{e}_k - \frac{1}{2!}\mathbf{F}'(\mathbf{x}_k)^{-1}\mathbf{F}''(\mathbf{x}_k)\boldsymbol{e}_k^2 + \frac{1}{3!}\mathbf{F}'(\mathbf{x}_k)^{-1}\mathbf{F}'''(\mathbf{x}_k)\boldsymbol{e}_k^3 + \mathbf{O}\left(\|\boldsymbol{e}_k^4\|\right). \tag{3.5}$$

Now, applying Taylor's expansion for $\mathbf{F}'\left(\frac{\mathbf{x}_k+2\mathbf{y}_k}{3}\right)$ at the point $\mathbf{x}_k$, we have

$$\mathbf{F}'\left(\frac{\mathbf{x}_k+2\mathbf{y}_k}{3}\right) = \mathbf{F}'(\mathbf{x}_k) - \frac{2}{3}\mathbf{F}''(\mathbf{x}_k)\left(\mathbf{F}'(\mathbf{x}_k)^{-1}\mathbf{F}(\mathbf{x}_k)\right) + \frac{2}{9}\mathbf{F}'''(\mathbf{x}_k)\left(\mathbf{F}'(\mathbf{x}_k)^{-1}\mathbf{F}(\mathbf{x}_k)\right)^2 + \cdots. \tag{3.6}$$

From Eqs. (3.5) and (3.6), we have

$$\begin{aligned}\mathbf{F}'\left(\frac{\mathbf{x}_k+2\mathbf{y}_k}{3}\right) = {} & \mathbf{F}'(\mathbf{x}_k) - \frac{2}{3}\mathbf{F}''(\mathbf{x}_k)\left[\boldsymbol{e}_k - \frac{1}{2!}\mathbf{F}'(\mathbf{x}_k)^{-1}\mathbf{F}''(\mathbf{x}_k)\boldsymbol{e}_k^2 + \frac{1}{3!}\mathbf{F}'(\mathbf{x}_k)^{-1}\mathbf{F}'''(\mathbf{x}_k)\boldsymbol{e}_k^3 + \mathbf{O}\left(\|\boldsymbol{e}_k^4\|\right)\right] \\ & + \frac{2}{9}\mathbf{F}'''(\mathbf{x}_k)\left[\boldsymbol{e}_k - \frac{1}{2!}\mathbf{F}'(\mathbf{x}_k)^{-1}\mathbf{F}''(\mathbf{x}_k)\boldsymbol{e}_k^2 + \frac{1}{3!}\mathbf{F}'(\mathbf{x}_k)^{-1}\mathbf{F}'''(\mathbf{x}_k)\boldsymbol{e}_k^3 + \mathbf{O}\left(\|\boldsymbol{e}_k^4\|\right)\right]^2 + \cdots.\end{aligned}$$

Thus

$$\mathbf{F}'\left(\frac{\mathbf{x}_k+2\mathbf{y}_k}{3}\right)=\mathbf{F}'(\mathbf{x}_k)-\frac{2}{3}\mathbf{F}''(\mathbf{x}_k)\,\boldsymbol{e}_k+\frac{1}{3}\mathbf{F}''(\mathbf{x}_k)\,\mathbf{F}'(\mathbf{x}_k)^{-1}\,\mathbf{F}''(\mathbf{x}_k)\,\boldsymbol{e}_k^2+\frac{2}{9}\mathbf{F}'''(\mathbf{x}_k)\,\boldsymbol{e}_k^2+\mathbf{O}\left(\|\boldsymbol{e}_k^3\|\right). \tag{3.7}$$

Now, from Eqs. (3.3), (3.4) and (3.7), we have

$$\begin{aligned}\left[\mathbf{F}'(\mathbf{x}_k)+3\mathbf{F}'\left(\frac{\mathbf{x}_k+2\mathbf{y}_k}{3}\right)\right]\boldsymbol{e}_{k+1}&=\mathbf{F}'(\mathbf{x}_k)\,\boldsymbol{e}_k+3\left[\mathbf{F}'(\mathbf{x}_k)-\frac{2}{3}\mathbf{F}''(\mathbf{x}_k)\,\boldsymbol{e}_k\right.\\&\quad\left.+\frac{1}{3}\mathbf{F}''(\mathbf{x}_k)\,\mathbf{F}'(\mathbf{x}_k)^{-1}\,\mathbf{F}''(\mathbf{x}_k)\,\boldsymbol{e}_k^2+\frac{2}{9}\mathbf{F}'''(\mathbf{x}_k)\,\boldsymbol{e}_k^2+\mathbf{O}\left(\|\boldsymbol{e}_k^3\|\right)\right]\boldsymbol{e}_k\\&\quad-4\left[\mathbf{F}'(\mathbf{x}_k)\,\boldsymbol{e}_k-\frac{1}{2!}\mathbf{F}''(\mathbf{x}_k)\,\boldsymbol{e}_k^2+\frac{1}{3!}\mathbf{F}'''(\mathbf{x}_k)\,\boldsymbol{e}_k^3+\mathbf{O}\left(\|\boldsymbol{e}_k^4\|\right)\right].\end{aligned}$$

Thus, from above equation, we have

$$\left[\mathbf{F}'(\mathbf{x}_k)+3\mathbf{F}'\left(\frac{\mathbf{x}_k+2\mathbf{y}_k}{3}\right)\right]\boldsymbol{e}_{k+1}=\left[\mathbf{F}''(\mathbf{x}_k)\,\mathbf{F}'(\mathbf{x}_k)^{-1}\,\mathbf{F}''(\mathbf{x}_k)\right]\boldsymbol{e}_k^3+\mathbf{O}\left(\|\boldsymbol{e}_k^4\|\right). \tag{3.8}$$

Eq. (3.8) shows that the Algorithm 2.2 has cubic convergence. In a similar way, one can prove the cubic convergence of the Algorithm 2.3. □

4. Efficiency index

It should be noted that our newly derived methods (Algorithms 2.2 and 2.3) for solving the system of nonlinear equations have a better efficiency index than the methods of Cordero and Torregrosa [10] for solving the system of nonlinear equations. We consider the definition of efficiency index (see [25]) as $p^{1/m}$, where p is the order of the method and m is the number of functional evaluations per iteration required by the method. If we consider the 2-dimensional case, then Algorithm 2.2 requires two functional evaluations and eight of its derivatives, whereas the Newton–Simpson's method of Cordero and Torregrosa [10] requires two functional evaluations and twelve of its derivatives, so the efficiency index of Algorithm 2.2 is $3^{1/10}\approx 1.11612$, which is better than the efficiency index $3^{1/14}\approx 1.08163$, of Newton–Simpson's method of Cordero and Torregrosa [10]. Similarly, Algorithm 2.3 requires two functional evaluations and twelve of its derivatives, so the efficiency index of Algorithm 2.3 is $3^{1/14}\approx 1.08163$, which is better than the efficiency index $3^{1/18}\approx 1.06294$, of the Open Newton's method of Cordero and Torregrosa [10] which requires two functional evaluations and eighteen of its derivatives. In general, for n-dimensional case, the efficiency index of Algorithm 2.2 is $3^{1/(n+2n^2)}$, for $n\geq 2$, which is better than the efficiency index $3^{1/(n+3n^2)}$ for $n\geq 2$, of the Newton–Simpson's method [10] and the efficiency index of Algorithm 2.3 is $3^{1/(n+3n^2)}$, for $n\geq 2$, which is better than the efficiency index $3^{1/(n+4n^2)}$ for $n\geq 2$, of the Open Newton's method [10].

5. Numerical examples and conclusions

We now present some examples to illustrate the efficiency and the comparison of the newly developed methods (Algorithms 2.2 and 2.3), see Table 5.1 We compare Newton's method (NM), the methods of Cordero and Torregrosa (Newton–Simpson's method) (CT[10]), the methods of Darvishi and Barati (DV [5]), Algorithm 2.2 (NR1) and Algorithm 2.3 (NR2) introduced in this paper. All computations were done using MAPLE, using 30 digit floating point arithmetic (Digits :=30). We used $\varepsilon=10^{-14}$. The following stopping criteria is used for computer programs:

(i) $\|\mathbf{x}_{k+1}-\mathbf{x}_k\|_\infty<\varepsilon$ (ii) $\|\mathbf{F}(\mathbf{x}_k)\|_\infty<\varepsilon$.

For every method, we analyze the number of iterations needed to converge to the solution. The computational order of convergence p approximated by means of

$$p\approx\frac{\ln\left(\|\mathbf{x}_{k+1}-\mathbf{x}_k\|_\infty/\|\mathbf{x}_k-\mathbf{x}_{k-1}\|_\infty\right)}{\ln\left(\|\mathbf{x}_k-\mathbf{x}_{k-1}\|_\infty/\|\mathbf{x}_{k-1}-\mathbf{x}_{k-2}\|_\infty\right)},$$

which was used by Cordero and Torregrosa [10].

Example 5.1 (*[10]*). Consider the following system of nonlinear equations

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

Table 5.1
Numerical examples and comparison.

Method	Initial value	IT	Approximate solution	p
Exp. 5.1				
NM	$(0.6, 0.6, 0.6, -0.2)^t$	5	$(0.57735, 0.57735, 0.57735, -0.28868)^t$	2.0
CT		3	$(0.57735, 0.57735, 0.57735, -0.28868)^t$	3.3
DV		3	$(0.57735, 0.57735, 0.57735, -0.28868)^t$	3.3
NR1		3	$(0.57735, 0.57735, 0.57735, -0.28868)^t$	3.3
NR2		3	$(0.57735, 0.57735, 0.57735, -0.28868)^t$	3.3
Exp. 5.2				
NM	$(0.2, 0.8)^t$	5	$(-0.14028501081, 0.1402850108)^t$	2.0
CT		4	$(-0.14028501081, 0.1402850108)^t$	3.0
DV		4	$(-0.14028501081, 0.1402850108)^t$	3.0
NR1		4	$(-0.14028501081, 0.1402850108)^t$	3.0
NR2		4	$(-0.14028501081, 0.1402850108)^t$	3.0
Exp. 5.3				
NM	$(0.5, 0.5, 0.5)^t$	6	$(0.69828861, 0.62852430, 0.34256419)^t$	2.0
CT		4	$(0.69828861, 0.62852430, 0.34256419)^t$	3.0
DV		4	$(0.69828861, 0.62852430, 0.34256419)^t$	3.0
NR1		4	$(0.69828861, 0.62852430, 0.34256419)^t$	3.0
NR2		4	$(0.69828861, 0.62852430, 0.34256419)^t$	3.0
Exp. 5.4				
NM	$(0.5, 0.5)^t$	7	$(-0.2222145551, 0.9938084186)^t$	2.0
CT		5	$(-0.2222145551, 0.9938084186)^t$	3.0
DV		Fails	–	–
NR1		5	$(-0.2222145551, 0.9938084186)^t$	3.0
NR2		5	$(-0.2222145551, 0.9938084186)^t$	3.0
NM	$(-0.3, 1)^t$	5	$(-0.2222145551, 0.9938084186)^t$	2.0
CT		3	$(-0.2222145551, 0.9938084186)^t$	3.0
DV		3	$(-0.2222145551, 0.9938084186)^t$	3.0
NR1		3	$(-0.2222145551, 0.9938084186)^t$	3.0
NR2		3	$(-0.2222145551, 0.9938084186)^t$	3.0
Exp. 5.5				
NM	$(-0.5, 0.5)^t$	5	$(0, 0)^t$	3.0
CT		3	$(0, 0)^t$	5.1
DV		3	$(0, 0)^t$	5.0
NR1		3	$(0, 0)^t$	4.5
NR2		3	$(0, 0)^t$	4.5
Exp. 4.5				
NM	$(0.8, 0.8)^t$	4	$(0, 0)^t$	3.0
CT		3	$(0, 0)^t$	4.5
DV		3	$(0, 0)^t$	4.3
NR1		3	$(0, 0)^t$	4.3
NR2		3	$(0, 0)^t$	4.2
Exp. 5.6				
NM	$(2.5, 0.5, 1.5)^t$	5	$(2.49137570, 0.24274588, 1.65351794)^t$	2.0
CT		4	$(2.49137570, 0.24274588, 1.65351794)^t$	3.0
DV		4	$(2.49137570, 0.24274588, 1.65351794)^t$	3.0
NR1		4	$(2.49137570, 0.24274588, 1.65351794)^t$	3.0
NR2		4	$(2.49137570, 0.24274588, 1.65351794)^t$	3.0

Example 5.2 ([10]). Consider the following system of nonlinear equations

$$e^{x_1^2} + 8x_1 \sin(x_2) = 0,$$
$$x_1 + x_2 = 1.$$

Example 5.3 ([5]). Consider the following system of nonlinear equations

$$x_1^2 + x_2^2 + x_3^2 = 1,$$
$$2x_1^2 + x_2^2 - 4x_3 = 0,$$
$$3x_1^2 - 4x_2^2 + x_3^2 = 0.$$

Example 5.4 ([10]). Consider the following system of nonlinear equations

$$x_1^2 - 2x_1 - x_2 + 0.5 = 0,$$
$$x_1^2 + 4x_2^2 - 4 = 0.$$

Example 5.5 (*[10]*). Consider the following system of nonlinear equations

$$e^{x_1^2} - e^{\sqrt{2}x_1} = 0,$$
$$x_1 - x_2 = 0.$$

Example 5.6 (*[11]*). Consider the following system of nonlinear equations

$$x_1^2 + x_2^2 + x_3^2 = 9,$$
$$x_1 x_2 x_3 = 1,$$
$$x_1 + x_2 - x_3^2 = 0.$$

Remark 5.1. We would like to mention that the CPU time of all the methods is 0.1 s. From the Table 5.1, we see that the Algorithms 2.2 and 2.3 are comparable with the other methods. Our methods can be considered as an alternative to Newton's method and other similar methods [3,5,10,12] for solving systems of nonlinear equations.

Acknowledgements

The authors would like to express their gratitude to Prof. Khalida Inayat Noor and the referees for their constructive and valuable suggestions.

References

[1] R.L. Burden, J.D. Faires, Numerical Analysis, 7th ed., PWS Publishing Company, Boston, 2001.
[2] J.M. Ortega, W.C. Rheinboldt, Iterative Solution of Nonlinear Equations in Several variables, Academic Press, 1970.
[3] S. Abbasbandy, Extended Newton's method for a system of nonlinear equations by modified Adomian decomposition method, Appl. Math. Comput. 170 (2005) 648–656.
[4] E. Babolian, J. Biazar, A.R. Vahidi, Solution of a system of nonlinear equations by Adimian decomposition method, Appl. Math. Comput. 150 (2004) 847–854.
[5] M.T. Darvishi, A. Barati, A third-order Newton-type method to solve systems of nonlinear equations, Appl. Math. Comput. 187 (2007) 630–635.
[6] M.T. Darvishi, A. Barati, Super cubic iterative methods to solve systems of nonlinear equations, Appl. Math. Comput. 188 (2007) 1678–1685.
[7] A. Golbabai, M. Javidi, A new family of iterative methods for solving system of nonlinear algebric equations, Appl. Math. Comput. 190 (2007) 1717–1722.
[8] D.K.R. Babajee, M.Z. Dauhoo, M.T. Darvishi, A. Barati, A note on the local convergence of iterative methods based on Adomian decomposition method and 3-node quadrature rule, Appl. Math. Comput. 200 (2008) 452–458.
[9] A. Cordero, J.R. Torregrosa, Variants of Newton's method for functions of several variables, Appl. Math. Comput. 183 (2006) 199–208.
[10] A. Cordero, J.R. Torregrosa, Variants of Newton's method using fifth-order quadrature formulas, Appl. Math. Comput. 190 (2007) 686–698.
[11] M.T. Darvishi, A. Barati, A fourth-order method from quadrature formulae to solve systems of nonlinear equations, Appl. Math. Comput. 188 (2007) 257–261.
[12] M. Frontini, E. Sormani, Third-order methods from quadrature formulae for solving systems of nonlinear equations, Appl. Math. Comput. 149 (2004) 771–782.
[13] K.E. Atkinson, An Introduction to Numerical Analysis, 2nd ed., John Wiley and Sons, New York, 1987.
[14] D.K.R. Babajee, M.Z. Dauhoo, Analysis of the properties of the variants of Newton's method with third order convergence, Appl. Math. Comput. 183 (2006) 659–684.
[15] F. Freudensten, B. Roth, Numerical solution of systems of nonlinear equations, J. ACM 10 (1963) 550–556.
[16] M. Grau-Sánchez, J.M. Peris, J.M. Gutiérrez, Accelerated iterative methods for finding solutions of a system of nonlinear equations, Appl. Math. Comput. 190 (2007) 1815–1823.
[17] H.H.H. Homeier, A modified Newton method with cubic convergence: The multivariate case, J. Comput. Appl. Math. 169 (2004) 161–169.
[18] J. Kou, A third-order modification of Newton method for systems of nonlinear equations, Appl. Math. Comput. 191 (2007) 117–121.
[19] M. Aslam Noor, Numerical Analysis and Optimization, Lecture Notes, Mathematics Department, COMSATS Institute of Information Technology, Islamabad, Pakistan, 2007.
[20] L.F. Shampine, R.C. Allen, S. Pruess, Fundamentals of Numerical Computing, John Wiley and Sons, New York, 1997.
[21] R.S. Varga, Matrix Iterative Analysis, Springer, Berlin, 2000.
[22] M. Aslam Noor, Some applications of quadrature formulas for solving nonlinear equations, Nonlinear Anal. Forum 12 (1) (2007) 91–96.
[23] M. Aslam Noor, Fifth-order convergent iterative method for solving nonlinear equations using quadrature formula, J. Math. Control Sci. Appl. 1 (2007) 241–249.
[24] M. Podisuk, U. Chundang, W. Sanprasert, Single-step formulas and multi-step formulas of the integration method for solving the initial value problem of ordinary differential equation, Appl. Math. Comput. 190 (2007) 1438–1444.
[25] W. Gautschi, Numerical Analysis: An Introduction, Birkhäuser, 1997.