Available online at www.isr-publications.com/jnsa
J. Nonlinear Sci. Appl., 10 (2017), 3834–3842
Research Article

ISSN: 2008-1898

Journal of Nonlinear Sciences and Applications

Journal Homepage: www.tjnsa.com - www.isr-publications.com/jnsa

The iterative methods with higher order convergence for solving a system of nonlinear equations

Zhongyuan Chen, Xiaofang Qiu, Songbin Lin, Baoguo Chen*

Research Center for Science Technology and Society, Fuzhou University of International Studies and Trade, Fuzhou 350202, P. R. China.

Communicated by Y.-Z. Chen

Abstract

In this paper, two variants of iterative methods with higher order convergence are developed in order to solve a system of nonlinear equations. It is proved that these two new methods have cubic convergence. Some numerical examples are given to show the efficiency and the performance of the new iterative methods, which confirm the good theoretical properties of the approach.

Keywords: System of nonlinear equations, iterative methods, higher convergence rate, numerical examples.
2010 MSC: 65H10.

1. Introduction

Consider the system of nonlinear equations

$$\begin{cases} f_1(x_1, x_2, \cdots, x_n) = 0, \\ f_2(x_1, x_2, \cdots, x_n) = 0, \\ \quad \vdots \\ f_n(x_1, x_2, \cdots, x_n) = 0, \end{cases} \tag{1.1}$$

where each function $f_i(x_1, x_2, \cdots, x_n)$ $(i = 1, 2, \cdots, n)$ can be thought of as mapping a vector $x = (x_1, x_2, \cdots, x_n)^T$ of the n-dimensional space R^n, into the real line R. The system can alternatively be represented by defining a functional F, mapping R^n into R^n by

$$F(x_1, x_2, \cdots, x_n) = (f_1(x_1, x_2, \cdots, x_n), \cdots, f_n(x_1, x_2, \cdots, x_n))^T.$$

Using vector notation to represent the variables $x_1, x_2, \cdots, x_n$, system (1.1) can be written as the following form

$$F(x) = 0. \tag{1.2}$$

Newton's method is the best procedure for finding a root of the equation (1.2) due to its formal simplicity and its fast convergence rate. It has been generalized in many ways for the solution of nonlinear

*Corresponding author
Email address: chenbg123@163.com (Baoguo Chen)

doi:10.22436/jnsa.010.07.37

Received 2017-06-05

problems (see [7, 17]). However, though the Newton's method converges very rapidly once an iteration is fairly close to the root, one cannot expect good convergence when an initial guess is not properly chosen or when the slope of $F(x)$ is extremely flat near the root. Recently, several iterative methods (see [1, 5, 6, 11, 14, 18]) based on the Newton's method for solving nonlinear equation (1.2) were developed. It has been shown (see [2–4, 9, 10, 12, 13, 15]) that the quadrature formulas have been used to develop some iterative methods for solving a system of nonlinear equations (1.2). Motivated and inspired by the on-going activities, in this paper, we construct two variants of iterative methods for solving a system of nonlinear equations with higher order convergence.

2. Iterative methods

Let $F : D \subseteq R^n \to R^n$ be k-time Fréchet differentiable in a convex set $D \subseteq R^n$ and x^* be a real zero of the nonlinear system of equations (1.2) of n equations with n variables. For any $x, x_n \in D$, we write (see [10]) Taylor's expansion for F as follows:

$$\begin{aligned} F(x) &= F(x_n) + F'(x_n)(x - x_n) + \frac{1}{2!}F''(x_n)(x - x_n)^2 \\ &\quad + \frac{1}{3!}F'''(x_n)(x - x_n)^3 + \cdots + \frac{1}{(k-1)!}F^{(k-1)}(x_n)(x - x_n)^{(k-1)} \\ &\quad + \int_0^1 \frac{(1-t)^{(k-1)}}{(k-1)!}F^{(k)}\big(x_n + t(x - x_n)\big)(x - x_n)^k dt. \end{aligned}$$

Especially, for $r = 1$ we have

$$F(x) = F(x_n) + \int_0^1 F'\big(x_n + t(x - x_n)\big)(x - x_n)dt. \tag{2.1}$$

Approximating the integral (2.1), we have

$$\int_0^1 F'\big(x_n + t(x - x_n)\big)(x - x_n)dt \cong F'(x_n)(x - x_n).$$

Then, we obtain

$$F(x) \cong F(x_n) + F'(x_n)(x - x_n).$$

In order to find $F(x) = 0$, we get

$$x_{n+1} = x_n - F'(x_n)^{-1}F(x_n),$$

which is is known as Newton's method for nonlinear system of equations (1.2) and has quadratic convergence (see [16]).

If we approximate the integral in (2.1) by using the four points Newton-Cotes, open formula (see [17]), then

$$\int_a^b f(x)dx \approx \frac{b-a}{24}\Big[11f(\frac{4a+b}{5}) + f(\frac{3a+2b}{5}) + f(\frac{2a+3b}{5}) + 11f(\frac{a+4b}{5})\Big]. \tag{2.2}$$

If we approximate the integral in (2.1) by using (2.2), then

$$\begin{aligned} \int_0^1 F'\big(x_n + t(x - x_n)\big)(x - x_n)dt \approx \Big[&\frac{11}{24}F'(\frac{4x_n + x}{5}) + \frac{1}{24}F'(\frac{3x_n + 2x}{5}) \\ &+ \frac{1}{24}F'(\frac{2x_n + 3x}{5}) + \frac{11}{24}F'(\frac{x_n + 4x}{5})\Big]. \end{aligned} \tag{2.3}$$

By using (2.3) in (2.1) and (1.2), we obtain

$$x = x_n - \Big[\frac{11}{24}F'(\frac{4x_n + x}{5}) + \frac{1}{24}F'(\frac{3x_n + 2x}{5}) + \frac{1}{24}F'(\frac{2x_n + 3x}{5}) + \frac{11}{24}F'(\frac{x_n + 4x}{5})\Big]^{-1} F(x_n). \tag{2.4}$$

Eq. (2.4) is an implicit method. To overcome this drawback, we use the prediction and correction technique. Using (2.4), we can obtain the following two-step iterative method for solving the nonlinear equation (1.2) as:

Algorithm 2.1. For a given x_0, compute the approximate solution x_{n+1} by iterative scheme

$$\begin{cases} y_n = x_n - F'(x_n)^{-1}F(x_n), \\ x_{n+1} = x_n - \Big[\frac{11}{24}F'(\frac{4x_n+y_n}{5}) + \frac{1}{24}F'(\frac{3x_n+2y_n}{5}) + \frac{1}{24}F'(\frac{2x_n+3y_n}{5}) \\ \qquad + \frac{11}{24}F'(\frac{x_n+4y_n}{5})\Big]^{-1} F(x_n). \end{cases} \tag{2.5}$$

Algorithm 2.1 is another iterative method for solving a system of nonlinear equations (1.2).

Cordero et al. [6] developed an improved two-step Newton's method for solving the nonlinear system (1.2)

$$\begin{cases} z_n = \phi(x_n, y_n), \\ w_n = z_n - F'(y_n)^{-1}F(z_n), \end{cases} \tag{2.6}$$

where $y_n = x_n - F'(x_n)^{-1}F(x_n)$ is the Newton's method with quadratic convergence. If $z_n = \phi(x_n, y_n)$ has order of convergence p, then (2.6) has order of convergence $p+2$.

Replacing $z_n = \phi(x_n, y_n)$ with approximation in (2.5), we can construct a new method as follows:

Algorithm 2.2. For a given x_0, compute the approximate solution x_{n+1} by iterative schemes

$$\begin{cases} y_n = x_n - F'(x_n)^{-1}F(x_n), \\ z_n = x_n - \Big[\frac{11}{24}F'(\frac{4x_n+y_n}{5}) + \frac{1}{24}F'(\frac{3x_n+2y_n}{5}) + \frac{1}{24}F'(\frac{2x_n+3y_n}{5}) \\ \qquad + \frac{11}{24}F'(\frac{x_n+4y_n}{5})\Big]^{-1} F(x_n), \\ x_{n+1} = z_n - F'(y_n)^{-1}F(z_n). \end{cases} \tag{2.7}$$

Algorithm 2.2 is another iterative method for solving a system of nonlinear equations (1.2).

3. Convergence of the method

In this section, we turn to analyze the convergence of Algorithms 2.1 and 2.2 using the Taylor's series technique.

Theorem 3.1. *Let $F : D \subseteq R^n \to R^n$ be k-time Fréchet differentiable in a convex set $D \subseteq R^n$ and x^* be a real simple zero of the nonlinear system of equations (1.2). The new iterative method defined by Algorithm 2.1 has order of convergence three.*

Proof. Defining $e_n = x_n - x^*$, then we have

$$e_{n+1} - e_n = -24\Big[11F'(\frac{4x_n+y_n}{5}) + F'(\frac{3x_n+2y_n}{5}) + F'(\frac{2x_n+3y_n}{5}) + 11F'(\frac{x_n+4y_n}{5})\Big]^{-1} F(x_n). \tag{3.1}$$

From (3.1), we obtain

$$\begin{aligned} &\Big[11F'(\frac{4x_n+y_n}{5}) + F'(\frac{3x_n+2y_n}{5}) + F'(\frac{2x_n+3y_n}{5}) + 11F'(\frac{x_n+4y_n}{5})\Big] e_{n+1} \\ &= \Big[11F'(\frac{4x_n+y_n}{5}) + F'(\frac{3x_n+2y_n}{5}) + F'(\frac{2x_n+3y_n}{5}) + 11F'(\frac{x_n+4y_n}{5})\Big] e_n - 24F(x_n). \end{aligned} \tag{3.2}$$

Let $x = x^*$ be a simple zero of the nonlinear system of equations (1.2). Since F is sufficiently differentiable, by expanding $F(x_n)$ and $F'(x_n)$ about $x = x^*$, we get

$$F(x^*) = F(x_n) + F'(x_n)(x^* - x_n) + \frac{1}{2!}F''(x_n)(x^* - x_n)^2 + \frac{1}{3!}F'''(x_n)(x^* - x_n)^3 + O(\|x^* - x_n\|^4).$$

But $F(x^*) = 0$, then

$$F(x_n) = F'(x_n)e_n - \frac{1}{2!}F''(x_n)e_n^2 + \frac{1}{3!}F'''(x_n)e_n^3 + O(\|e_n\|^4).$$

Then we obtain

$$F'(x_n)^{-1}F(x_n) = e_n - \frac{1}{2!}F'(x_n)^{-1}F''(x_n)e_n^2 + \frac{1}{3!}F'(x_n)^{-1}F'''(x_n)e_n^3 + O(\|e_n\|^4). \tag{3.3}$$

Applying Taylor's expansion to

$$F'\Big(\frac{4x_n + y_n}{5}\Big),\ F'\Big(\frac{3x_n + 2y_n}{5}\Big),\ F'\Big(\frac{2x_n + 3y_n}{5}\Big),\ F'\Big(\frac{x_n + 4y_n}{5}\Big)$$

at the point x_n, and using (3.3), we have

$$\begin{aligned}
F'\Big(\frac{4x_n + y_n}{5}\Big) &= F'(x_n) + F''(x_n)\Big(\frac{4x_n + y_n}{5} - x_n\Big) + \frac{F'''(x_n)}{2!}\Big(\frac{4x_n + y_n}{5} - x_n\Big)^2 + \cdots \\
&= F'(x_n) - \frac{1}{5}F''(x_n)\Big(F'(x_n)^{-1}F(x_n)\Big) + \frac{F'''(x_n)}{50}\Big(F'(x_n)^{-1}F(x_n)\Big)^2 + \cdots \\
&= F'(x_n) - \frac{1}{5}F''(x_n)e_n + \frac{1}{10}F''(x_n)F'(x_n)^{-1}F''(x_n)e_n^2 + \frac{1}{50}F'''(x_n)e_n^2 + O(\|e_n\|^3), \\
F'\Big(\frac{3x_n + 2y_n}{5}\Big) &= F'(x_n) + F''(x_n)\Big(\frac{3x_n + 2y_n}{5} - x_n\Big) + \frac{F'''(x_n)}{2!}\Big(\frac{3x_n + 2y_n}{5} - x_n\Big)^2 + \cdots \\
&= F'(x_n) - \frac{2}{5}F''(x_n)\Big(F'(x_n)^{-1}F(x_n)\Big) + \frac{2}{25}F'''(x_n)\Big(F'(x_n)^{-1}F(x_n)\Big)^2 + \cdots \\
&= F'(x_n) - \frac{2}{5}F''(x_n)e_n + \frac{1}{5}F''(x_n)F'(x_n)^{-1}F''(x_n)e_n^2 + \frac{2}{25}F'''(x_n)e_n^2 + O(\|e_n\|^3), \\
F'\Big(\frac{2x_n + 3y_n}{5}\Big) &= F'(x_n) + F''(x_n)\Big(\frac{2x_n + 3y_n}{5} - x_n\Big) + \frac{F'''(x_n)}{2!}\Big(\frac{2x_n + 3y_n}{5} - x_n\Big)^2 + \cdots \\
&= F'(x_n) - \frac{3}{5}F''(x_n)\Big(F'(x_n)^{-1}F(x_n)\Big) + \frac{9}{50}F'''(x_n)\Big(F'(x_n)^{-1}F(x_n)\Big)^2 + \cdots \\
&\quad + \frac{9}{50}F'''(x_n)\Big[e_n^2 - F'(x_n)^{-1}F''(x_n)e_n^3 + O(\|e_n\|^4)\Big] \\
&= F'(x_n) - \frac{3}{5}F''(x_n)e_n + \frac{3}{10}F''(x_n)F'(x_n)^{-1}F''(x_n)e_n^2 + \frac{9}{50}F'''(x_n)e_n^2 + O(\|e_n\|^3), \\
F'\Big(\frac{x_n + 4y_n}{5}\Big) &= F'(x_n) + F''(x_n)\Big(\frac{x_n + 4y_n}{5} - x_n\Big) + \frac{F'''(x_n)}{2!}\Big(\frac{x_n + 4y_n}{5} - x_n\Big)^2 + \cdots \\
&= F'(x_n) - \frac{4}{5}F''(x_n)\Big(F'(x_n)^{-1}F(x_n)\Big) + \frac{8}{25}F'''(x_n)\Big(F'(x_n)^{-1}F(x_n)\Big)^2 + \cdots \\
&= F'(x_n) - \frac{4}{5}F''(x_n)e_n + \frac{2}{5}F''(x_n)F'(x_n)^{-1}F''(x_n)e_n^2 + \frac{8}{25}F'''(x_n)e_n^2 + O(\|e_n\|^3).
\end{aligned} \tag{3.4}$$

Now, from (3.2) and (3.4), we obtain

$$\begin{aligned}
&\Big[11F'(\frac{4x_n + y_n}{5}) + F'(\frac{3x_n + 2y_n}{5}) + F'(\frac{2x_n + 3y_n}{5}) + F'(\frac{x_n + 4y_n}{5})\Big]e_{n+1} \\
&= \Big[11F'(\frac{4x_n + y_n}{5}) + F'(\frac{3x_n + 2y_n}{5}) + F'(\frac{2x_n + 3y_n}{5}) + F'(\frac{x_n + 4y_n}{5})\Big]e_n - 24F(x_n)
\end{aligned}$$

$$
\begin{aligned}
&= 11\Big[F'(x_n)e_n - \frac{1}{5}F''(x_n)e_n^2 + \frac{1}{10}F''(x_n)F'(x_n)^{-1}F''(x_n)e_n^3 + O(\|e_n\|^4)\Big] \\
&\quad + \Big[F'(x_n)e_n - \frac{2}{5}F''(x_n)e_n^2 + \frac{1}{5}F''(x_n)F'(x_n)^{-1}F''(x_n)e_n^3 + O(\|e_n\|^4)\Big] \\
&\quad + \Big[F'(x_n)e_n - \frac{3}{5}F''(x_n)e_n^2 + \frac{3}{10}F''(x_n)F'(x_n)^{-1}F''(x_n)e_n^3 + O(\|e_n\|^4)\Big] \\
&\quad + 11\Big[F'(x_n)e_n - \frac{4}{5}F''(x_n)e_n^2 + \frac{2}{5}F''(x_n)F'(x_n)^{-1}F''(x_n)e_n^3 + O(\|e_n\|^4)\Big] \\
&\quad - 24\Big[F'(x_n)e_n - \frac{1}{2!}F''(x_n)e_n^2 + \frac{1}{3!}F'''(x_n)e_n^3 + O(\|e_n\|^4)\Big] \\
&= 6F''(x_n)F'(x_n)^{-1}F''(x_n)e_n^3 + O(\|e_n\|^4)\Big].
\end{aligned}
$$

This shows the third order convergence of the method. Hence, the proof is completed. □

Theorem 3.2. *Let $F : D \subseteq R^n \to R^n$ be k-time Fréchet differentiable in a convex set $D \subseteq R^n$ and x^* be a real simple zero of the nonlinear system of equations (1.2). The new iterative method defined by Algorithm 2.2 has order of convergence five and satisfies the error equation*

$$e^{n+1} = 2C_2^4 e_n^5 + O(\|e_n^6\|).$$

Proof. Defining $e_n = x_n - x^*, \tilde{e}_n = y_n - x^*, d_n = z_n - x^*, C_k = \frac{1}{k!}F'(x^*)^{-1}F(x^*)^k, k = 2, 3, \ldots,$ and applying Taylor's expansion to $F(x_n), F'(x_n), F'(y_n)$ at the point x_n, we have

$$F(x_n) = F'(x^*)\big[e_n + C_2e_n^2 + C_3e_n^3 + C_4e_n^4 + C_5e_n^5 + O(\|e_n^6\|)\big], \tag{3.5}$$

$$F'(x_n) = F'(x^*)\big[I + 2C_2e_n + 3C_3e_n^2 + 4C_4e_n^3 + 5C_5e_n^4 + O(\|e_n^5\|)\big]. \tag{3.6}$$

It follows from (3.6) that

$$F'(x_n)^{-1} = \big[I - 2C_2e_n + (4C_2^2 - 3C_3)e_n^2 - (8C_2^3e_n^3 - 6C_2C_3 - 6C_3C_2 + 4C_4)e_n^3\big]F'(x^*)^{-1} + O(\|e_n^4\|). \tag{3.7}$$

From (2.7), (3.6), and (3.7), we obtain

$$
\begin{aligned}
\tilde{e}_n = x_n - x^* - F'(x_n)^{-1}F(x_n) &= C_2e_n^2 + 2(C_3 - C_2^2)e_n^3 + (8C_2^3 - 6C_3C_2 - 4C_2C_3 + 3C_4)e_n^4 + O(\|e_n^5\|), \\
F'(y_n) &= F'(x^*)\big[I + 2C_2\tilde{e}_n + 3C_3\tilde{e}_n^2 + O(\|\tilde{e}_n^3\|)\big].
\end{aligned}
$$

Then we have

$$
\begin{aligned}
F'(y_n) = F'(x^*)\big[&I + 2C_2^2e_n^2 + 4(C_2C_3 - C_2^3)e_n^3 \\
&+ (16C_2^4 - 12C_2C_3C_2 - 8C_2^2C_3 + 3C_3C_2^2 + 6C_2C_4)e_n^4 + O(\|e_n^5\|)\big].
\end{aligned} \tag{3.8}
$$

It follows from (3.8) that

$$F'(y_n)^{-1} = F'(x^*)\big[I - 2C_2^2e_n^2 + 4C_2(C_2^2 - C_3)e_n^3\big]F'(x^*)^{-1} + O(\|e_n^4\|). \tag{3.9}$$

Furthermore, we have

$$
\begin{aligned}
\frac{4x_n + y_n}{5} - x^* &= \frac{4}{5}(x_n - x^*) + \frac{1}{5}(y_n - x^*) = \frac{4}{5}e_n + \frac{1}{5}C_2e_n^2 + \frac{2}{5}(C_3 - C_2^2)e_n^3 + O(\|e_n^4\|), \\
\frac{3x_n + 2y_n}{5} - x^* &= \frac{3}{5}(x_n - x^*) + \frac{2}{5}(y_n - x^*) = \frac{3}{5}e_n + \frac{2}{5}C_2e_n^2 + \frac{4}{5}(C_3 - C_2^2)e_n^3 + O(\|e_n^4\|), \\
\frac{2x_n + 3y_n}{5} - x^* &= \frac{2}{5}(x_n - x^*) + \frac{3}{5}(y_n - x^*) = \frac{2}{5}e_n + \frac{3}{5}C_2e_n^2 + \frac{6}{5}(C_3 - C_2^2)e_n^3 + O(\|e_n^4\|), \\
\frac{x_n + 4y_n}{5} - x^* &= \frac{1}{5}(x_n - x^*) + \frac{4}{5}(y_n - x^*) = \frac{1}{5}e_n + \frac{4}{5}C_2e_n^2 + \frac{8}{5}(C_3 - C_2^2)e_n^3 + O(\|e_n^4\|).
\end{aligned}
$$

Thus, we have

$$
\begin{aligned}
F'\Big(\frac{4x_n+y_n}{5}\Big) &= F'(x^*)\Big[I+2C_2(\frac{4x_n+y_n}{5}-x^*)+3C_3(\frac{4x_n+y_n}{5}-x^*)^2\\
&\quad +4C_4(\frac{4x_n+y_n}{5}-x^*)^3+O(\|(\frac{4x_n+y_n}{5}-x^*)^4\|)\Big]\\
&= F'(x^*)\Big[I+\frac{8}{5}C_2e_n+\Big(\frac{2}{5}C_2^2+\frac{48}{25}C_3\Big)e_n^2+\Big(\frac{4}{5}(C_2C_3-C_2^3)\\
&\quad +\frac{24}{25}C_3C_2+\frac{256}{125}C_4\Big)e_n^3+O(\|e_n^4\|)\Big],\\
F'\Big(\frac{3x_n+2y_n}{5}\Big) &= F'(x^*)\Big[I+2C_2(\frac{3x_n+2y_n}{5}-x^*)+3C_3(\frac{3x_n+2y_n}{5}-x^*)^2\\
&\quad +4C_4\Big(\frac{3x_n+2y_n}{5}-x^*\Big)^3+O(\|(\frac{3x_n+2y_n}{5}-x^*)^4\|)\Big]\\
&= F'(x^*)\Big[I+\frac{6}{5}C_2e_n+\Big(\frac{4}{5}C_2^2+\frac{27}{25}C_3\Big)e_n^2+\Big(\frac{8}{5}(C_2C_3-C_2^3)\\
&\quad +\frac{36}{25}C_3C_2+\frac{108}{125}C_4\Big)e_n^3+O(\|e_n^4\|)\Big],\\
F'\Big(\frac{2x_n+3y_n}{5}\Big) &= F'(x^*)\Big[I+2C_2(\frac{2x_n+3y_n}{5}-x^*)+3C_3(\frac{2x_n+3y_n}{5}-x^*)^2\\
&\quad +4C_4(\frac{2x_n+3y_n}{5}-x^*)^3+O(\|(\frac{2x_n+3y_n}{5}-x^*)^4\|)\Big]\\
&= F'(x^*)\Big[I+\frac{4}{5}C_2e_n+\Big(\frac{6}{5}C_2^2+\frac{12}{25}C_3\Big)e_n^2+\Big(\frac{12}{5}(C_2C_3-C_2^3)\\
&\quad +\frac{36}{25}C_3C_2+\frac{32}{125}C_4\Big)e_n^3+O(\|e_n^4\|)\Big],\\
F'\Big(\frac{x_n+4y_n}{5}\Big) &= F'(x^*)\Big[I+2C_2(\frac{x_n+4y_n}{5}-x^*)+3C_3(\frac{x_n+4y_n}{5}-x^*)^2\\
&\quad +4C_4(\frac{x_n+4y_n}{5}-x^*)^3+O(\|(\frac{x_n+4y_n}{5}-x^*)^4\|)\Big]\\
&= F'(x^*)\Big[I+\frac{2}{5}C_2e_n+\Big(\frac{8}{5}C_2^2+\frac{3}{25}C_3\Big)e_n^2+\Big(\frac{16}{5}(C_2C_3-C_2^3)\\
&\quad +\frac{24}{25}C_3C_2+\frac{4}{125}C_4\Big)e_n^3+O(\|e_n^4\|)\Big].
\end{aligned}
\tag{3.10}
$$

From (3.10), we have

$$
\begin{aligned}
&\frac{11}{24}F'\Big(\frac{4x_n+y_n}{5}\Big)+\frac{1}{24}F'\Big(\frac{3x_n+2y_n}{5}\Big)+\frac{1}{24}F'\Big(\frac{2x_n+3y_n}{5}\Big)+\frac{11}{24}F'\Big(\frac{x_n+4y_n}{5}\Big)\\
&\quad = F'(x^*)\Big[I+C_2e_n+(C_2^2+C_3)e_n^2+(2C_2C_3+C_3C_2-2C_2^3+\frac{74}{75}C_4)e_n^3+O(\|e_n^4\|)\Big].
\end{aligned}
\tag{3.11}
$$

We obtain from (3.11) that

$$
\begin{aligned}
&\Big[\frac{11}{24}F'(\frac{4x_n+y_n}{5})+\frac{1}{24}F'(\frac{3x_n+2y_n}{5})+\frac{1}{24}F'(\frac{2x_n+3y_n}{5})+\frac{11}{24}F'(\frac{x_n+4y_n}{5})\Big]^{-1}\\
&\quad = \Big[I-C_2e_n-C_3e_n^2-(C_2C_3-3C_2^2+\frac{74}{75}C_4)e_n^3+O(\|e_n^4\|)\Big]F'(x^*)^{-1}.
\end{aligned}
\tag{3.12}
$$

From (3.5) and (3.12) we have

$$
\begin{aligned}
d_n = z_n-x^* = x_n-x^* - \Big[\frac{11}{24}F'(\frac{4x_n+y_n}{5})+\frac{1}{24}F'(\frac{3x_n+2y_n}{5})\\
+\frac{1}{24}F'(\frac{2x_n+3y_n}{5})+\frac{11}{24}F'(\frac{x_n+4y_n}{5})\Big]^{-1}F(x_n)
\end{aligned}
$$

$$= C_2^2 e_n^3 + (2C_2C_3 + C_3C_2 - 3C_2^3 - \frac{1}{75}C_4)e_n^4 + O(\|e_n^5\|).$$

Applying Taylor's expansion for $F(z_n)$ at the point x_n, we obtain

$$\begin{aligned} F(z_n) &= F(x^*)\left[d_n + C_2 d_n^2 + C_3 d_n^3 + C_4 d_n^4 + O(\|d_n^5\|)\right] \\ &= F(x^*)\left[C_2^2 e_n^3 + (2C_2C_3 + C_3C_2 - 3C_2^3 - \frac{1}{75}C_4)\right] e_n^4 + O(\|e_n^5\|). \end{aligned} \tag{3.13}$$

From (3.9) and (3.13) we have

$$F'(y_n)^{-1}F(z_n) = C_2^2 e_n^3 + (2C_2C_3 - 3C_2^3 - \frac{1}{75}C_4)e_n^4 - 2C_2^4 e_n^5 + O(\|e_n^6\|). \tag{3.14}$$

Now, from (2.7) and (3.14), we obtain

$$e^{n+1} = x^{n+1} - x^* = d_n - F'(y_n)^{-1}F(z_n) = 2C_2^4 e_n^5 + O(\|e_n^6\|).$$

This shows the fifth-order convergence of the method. Hence, the proof is completed. □

4. Numerical results

In this section, some numerical comparisons are made with several other existing methods to illustrate the efficiency and the performance of the newly developed method. We compare the classical Newton's method (NM), the method of Darvishi and Barati (DV, [10]), the method of Aslam Noor and Waseem (AN, [2]), Algorithm 2.1 (MY1) and Algorithm 2.2 (MY2) introduced in this paper. All numeric computations have been carried out in MATLAB 2010b environment, rounding to 16 significant decimal digits. The stopping criterion used is $\|x_{n+1} - x_n\| < 10^{-16}$ and the maximum number of iterations is 101. What is more, the effect to efficiency and the performance of the new iterative methods is considered, which includes the iteration times "Iter", the "CPU" run time and the "ERR" errors. For every method, we analyze plenty of iterations needed to converge to the computational order of convergence p, approximated by means of

$$p \approx \frac{\ln(\|(x_{n+1} - x_n\|_\infty \|x_n - x_{n-1})\|_\infty^{-1})}{\ln(\|(x_n - x_{n-1}\|_\infty \|x_{n-1} - x_{n-2}\|_\infty^{-1})},$$

which is defined by Cordero and Torregrosa (see [8]).

Example 4.1. Consider the following system of nonlinear equations:

$$F(x) = \begin{cases} f_1(x) = x_1^4 x_2 - x_1 x_2 + 2x_1 - x_2 = 1, \\ f_2(x) = x_2 e^{-x_1} + x_1 - x_2 - e^{-1} = 0. \end{cases}$$

Table 1: Numerical results for Example 4.1.

Method	Exact solution	Initial point	Iter	CPU	Err	p
NM			9	0. 0015	1. 1102e-016	-Inf
DV			5	0. 1141	2. 2204e-016	0. 7160
AN	$x^* = (1,1)^T$	$x_0 = (1.5, 1.5)^T$	4	0. 0010	4. 4409e-016	3. 0806
MY1			4	0. 0008	6. 6613e-016	3. 0192
MY2			2	0. 0006	6. 7354e-016	3. 5883

Example 4.2. Consider the following system of nonlinear equations, which have two exact solutions at $x_1^* = \left(-\frac{1}{\sqrt{3}}, -\frac{1}{\sqrt{3}}, -\frac{1}{\sqrt{3}}, \frac{1}{2\sqrt{3}}\right)^T$ using initial approximation at $x_{01} = (-2,-2,-2,0)^T$ and $x_2^* = \left(\frac{1}{\sqrt{3}}, \frac{1}{\sqrt{3}}, \frac{1}{\sqrt{3}}, -\frac{1}{2\sqrt{3}}\right)^T$ using initial approximation at $x_{02} = (2,2,2,2)^T$:

$$F(x) = \begin{cases} f_1(x) = x_2x_3 + x_2x_4 + x_3x_4 = 0, \\ f_2(x) = x_1x_3 + x_1x_4 + x_3x_4 = 0, \\ f_3(x) = x_1x_2 + x_1x_4 + x_2x_4 = 0, \\ f_4(x) = x_1x_2 + x_1x_3 + x_2x_3 = 1. \end{cases}$$

Table 2: Numerical results for Example 4.2.

Method	Exact solution	Initial point	Iter	CPU	Err	p
NM			6	0. 0066	7. 2164e-016	2. 0833
DV			4	0. 0013	4. 7184e-015	3. 3702
AN	x_1^*	x_{01}	3	0. 0072	9. 9466e-007	3. 2770
MY1			3	0. 0018	9. 9466e-007	3. 2770
MY2			2	0. 0016	2. 1091e-005	2. 9962
NM			6	0. 0021	2. 1094e-015	2. 0815
DV			4	0. 0034	1. 4266e-014	3. 3214
AN	x_2^*	x_{02}	3	0. 0010	3. 1083e-006	3. 5079
MY1			3	0. 0019	3. 1083e-006	3. 5079
MY2			2	0. 0008	6. 6438e-005	4. 5467

Example 4.3. Consider the following system of nonlinear equations:

$$F(x) = \begin{cases} f_1(x) = x_1^3 + x_1^2 - x_2 + x_3 - 13 = 0, \\ f_2(x) = x_2^3 + x_2^2 + x_1 + 2x_3 - 20 = 0, \\ f_3(x) = x_3^3 - x_3^2 + 2x_1 - 2x_2 - 18 = 0. \end{cases}$$

Table 3: Numerical results for Example 4.3.

Method	Exact solution	Initial point	Iter	CPU	Err	p
NM			6	0. 0010	8. 8818e-016	1. 9669
DV			25	0. 0056	8. 6331e-011	2. 9681
AN	$x^* = (2, 2, 3)^T$	$x_0 = (1, 1, 2)^T$	3	0. 0067	1. 1472e-006	3. 1071
MY1			3	0. 0042	1. 1472e-006	3. 1071
MY2			2	0. 0020	8. 4229e-005	5. 5413

Example 4.4. Consider the following system of nonlinear equations:

$$F(x) = \begin{cases} f_1(x) = x_1^2 - 10x_1 + x_2^2 + 8 = 0, \\ f_2(x) = x_1x_2^2 + x_1 - 10x_2 + 8 = 0. \end{cases}$$

Table 4: Numerical results for Example 4.4.

Method	Exact solution	Initial point	Iter	CPU	Err	p
NM			101	0. 0165	2. 2204e-016	-1
DV			4	0. 1012	1. 1102e-016	0. 6834
AN	$x^* = (1, 1)^T$	$x_0 = (-1, 2)^T$	3	0. 0012	4. 6629e-015	2. 8862
MY1			3	0. 0007	4. 6629e-015	2. 8862
MY2			2	0. 0007	1. 3700e-013	4. 9308

5. Conclusions

In this paper, two variants of iterative methods are developed in order to solve a system of nonlinear equations with higher order convergence. The numerical examples in tables show that our method is very effective and provides highly accurate results in a less number of iterations as compared with other existing methods.

Acknowledgment

The authors thank the anonymous referee for helping to improve the original manuscript by valuable suggestions. The research was supported by National Social Science Foundation of China (Grant No.16BKS132).

References

[1] S. Abbasbandy, *Extended Newton's method for a system of nonlinear equations by modified Adomian decomposition method*, Appl. Math. Comput., **170** (2005), 648–656. 1

[2] M. Aslam Noor, M. Waseem, *Some iterative methods for solving a system of nonlinear equations*, Comput. Math. Appl., **57** (2009), 101–106. 1, 4

[3] 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.

[4] E. Babolian, J. Biazar, A. R. Vahidi, *Solution of a system of nonlinear equations by Adomian decomposition method*, Appl. Math. Comput., **150** (2004), 847–854. 1

[5] C.-B. Chun, *A new iterative method for solving nonlinear equations*, Appl. Math. Comput., **178** (2006), 415–422. 1

[6] A. Cordero, J. L. Hueso, E. Martínez, J. R. Torregrosa, *Increasing the convergence order of an iterative method for nonlinear systems*, Appl. Math. Lett., **25** (2012), 2369–2374. 1, 2

[7] A. Cordero, J. R. Torregrosa, *Variants of Newton's method for functions of several variables*, Appl. Math. Comput., **183** (2006), 199–208. 1

[8] A. Cordero, J. R. Torregrosa, *Variants of Newton's method using fifth-order quadrature formulas*, Appl. Math. Comput., **190** (2007), 686–698. 4

[9] 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. 1

[10] M. T. Darvishi, A. Barati, *A third-order Newton-type method to solve systems of nonlinear equations*, Appl. Math. Comput., **187** (2007), 630–635. 1, 2, 4

[11] M. T. Darvishi, A. Barati, *Super cubic iterative methods to solve systems of nonlinear equations*, Appl. Math. Comput., **188** (2007), 1678–1685. 1

[12] M. Frontini, E. Sormani, *Some variant of Newton's method with third-order convergence*, Appl. Math. Comput., **140** (2003), 419–426. 1

[13] M. Frontini, E. Sormani, *Third-order methods from quadrature formulae for solving systems of nonlinear equations*, Appl. Math. Comput., **149** (2004), 771–782. 1

[14] C. T. Kelley, *Solving nonlinear equations with Newton's method*, Fundamentals of Algorithms, Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, (2003). 1

[15] M. A. Noor, *Fifth-order convergent iterative method for solving nonlinear equations using quadrature formula*, J. Math. Control Sci. Appl., **1** (2007), 241–249. 1

[16] J. M. Ortega, W. C. Rheinboldt, *Iterative solution of nonlinear equations in several variables*, Academic Press, New York-London, (1970). 2

[17] 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. 1, 2

[18] J. R. Sharma, R. Sharma, *Some third order methods for solving systems of nonlinear equations*, World Acad. Sci. Eng. Technol., Int. J. Math. Comput. Phys. Electr. Comput. Eng., **5** (2011), 1864–1871. 1