# An Efficient Two-step Iterative Method for Solving System of Nonlinear Equations

M. A. Hafiz<sup>1</sup> & Mohamed S. M. Bahgat<sup>1,2</sup>

<sup>1</sup> Department of Mathematics, Faculty of Science and Arts, Najran University, Najran, Saudi Arabia

<sup>2</sup> Department of Mathematics, Faculty of Science, Minia University, Minia, Egypt

Correspondence: M. A. Hafiz, Department of Mathematics, Faculty of Science and Arts, Najran University, Najran, Saudi Arabia. Tel: 96-656-303-4182. E-mail: admin@mhafiz.net; Mohamed S. M. Bahgat, Department of Mathematics, Faculty of Science, Minia University, Minia, Egypt & Department of Mathematics, Faculty of Science and Arts, Najran University, Najran, Saudi Arabia. Tel: 96-650-891-7215. E-mail: msmbahgat66@hotmail.com

Received: May 17, 2012 Accepted: June 2, 2012 Online Published: July 4, 2012 doi:10.5539/jmr.v4n4p28 URL: http://dx.doi.org/10.5539/jmr.v4n4p28

## Abstract

In this paper, we propose and analyze a new predictor-corrector method for solving a nonlinear system of equations using the weight combination of mid-point, Trapezoidal and quadrature formulas. A new algorithm based on this method has been presented. The proposed method has been tested on a series of examples published in the literature and shows good results. Comparison with other similar methods is given.

Keywords: system of non-linear equations, predictor-corrector methods, quadrature formulas

Recently, several iterative methods have been developed to solve nonlinear equations and the system of nonlinear equations. These methods have been improved using Taylor interpolating polynomials quadrature formula, homotopy perturbation method and decomposition techniques (Abbasbandy, 2005; Jafari & Gejji, 2006; He, 1999; Awawdeh, 2009; Vahidi et al., 2012). Noor et al. (2006) have considered an alternative decomposition technique which does not involve the derivative of the domain polynomial. Furthermore, by improving Newton method, Chun (2006) has presented a new iterative method to solve nonlinear equations. By using Adomain decomposition, Darvishi et al. (2007a; b) have constructed new methods, and Golbabai & Javidi (2007) have applied the homotopy perturbation method to build a new family of Newton-like iterative methods for solving system of nonlinear equations. Ozel (2010) has considered a new decomposition method for solving the system of nonlinear equations. Hafiz & Bahgat (2012) modified Householder iterative method for solving system of nonlinear equations. He also show that this new method includes famous two step Newton method as a special case. Soheili et al. (2008) have developed a new iterative method by using the weight combination of the midpoint, and Trapezoidal quadrature formulas to solve nonlinear equation. In this paper, we aim to generalize and apply this method to solve a system of nonlinear equations. Finally, some numerical examples have fulfilled with Maple software to demonstrate our method and comparison of our results with those derived from previous methods. All test problems reveal accuracy and fast convergence of our method.

## 2. Iterative Method

The general form of a system of non-linear equations is

$$f_1(x_1, x_2, \cdots, x_n) = 0, \quad f_2(x_1, x_2, \cdots, x_n) = 0, \quad \dots, \quad f_n(x_1, x_2, \cdots, x_n) = 0,$$
 (1)

Where each function  $f_i$  can be thought of as mapping a vector  $x = (x_1, x_2, ..., x_n)$  of the *n*-dimensional space  $\mathbb{R}^n$ , into the real line  $\mathbb{R}$ . The system can alternatively be represented by defining a functional F, mapping  $\mathbb{R}^n$  into  $\mathbb{R}^n$  by:

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

Using vector notation to represent the variables  $x_1, x_2, \ldots, x_n$ , a system (1) then assumes the form:

$$F(\mathbf{x}) = 0 \tag{2}$$

The functions  $f_1, f_2, \ldots, f_n$  are called the coordinate functions of *F*.

Before discussing the numerical solution of a system given in the form (1) or (2), we need to consider some results concerning continuity, differentiability and limits of functions from  $R^n$  to  $R^n$ , see (Burden & Farires, 1985).

**Definition 1** Let *F* be a function from  $D \subset R^n$  into  $R^n$  and suppose *F* has the representation

$$F(x_1, x_2, \dots, x_n) = (f_1(x), f_2(x), \dots, f_n(x))^T$$

Where  $f_i$  for each *i* is a mapping from  $R^n$  into *R*, we define

$$\lim_{x \to x_0} F(x) = L = (L_1, L_2, \dots, L_n)^T$$

If and only if  $\lim_{x \to x_0} f_i(x) = L_i$  for each i = 1, 2, ..., n.

**Definition 2** Let *F* be a function from  $D \subset \mathbb{R}^n$  into  $\mathbb{R}^n$  with the representation  $F(x) = (f_1(x), f_2(x), \dots, f_n(x))^T$ . The function *F* is said to be continuous at  $x_0 \in D$  provided  $\lim_{x \to x_0} F(x)$  exists and  $\lim_{x \to x_0} F(x) = F(x_0)$ . *F* is said to be continuous on the set *D* if *F* is continuous at each *x* into *D*.

**Definition 3** Let *F* be a function from  $D \subset \mathbb{R}^n$  into  $\mathbb{R}$  and  $x_0 \in D$ . If constants  $\delta > 0$  and K > 0 exist with  $\left|\frac{\partial f(x)}{\partial x_j}\right| \leq K$  for each j = 1, 2, ..., n whenever  $||x - x_0|| < \delta$  and  $x \in D$ , then *f* is continuous at  $x_0$ .

#### 3. Two-step Iterative Algorithm Description

Suppose that *x* be the simple zero of sufficiently differentiable functions and consider the numerical solution of the system of equations F(x) = 0, where  $F : D \subseteq \mathbb{R}^n \to \mathbb{R}^n$  is a smooth mapping that has continuous second order partial derivatives on a convex open set *D*, and that has a locally unique root *x* in *D*,  $F(x) = (f_1(x), f_2(x), \dots, f_n(x))^T$ ,  $x = (x_1, x_2, \dots, x_n)^T$  and  $f_i : \mathbb{R}^n \to \mathbb{R}$  is a nonlinear function, then,

$$F(x) = F(x_i) + \int_{x_i}^{x} F'(T) dT,$$
(3)

The matrix of partial derivatives appearing in equation (3) is the Jacobian J where  $\int_{x_i}^x F'(t)dt$  is multiple integrals as follows

$$\int_{x_i}^x F'(t)dt = \int_{x_{i,1}}^{x_1} \int_{x_{i,2}}^{x_2} \dots \int_{x_{i,n}}^{x_n} F'(x_1, x_2, \dots, x_n)dx_n dx_{n-1} \dots dx_1$$

The most obvious approach is to treat the multiple integral as a nested sequence of one-dimensional integrals, and to use one-dimensional quadrature with respect to each argument in turn. So we can approximate  $\int_{x_i}^x F'(t)dt$  with average of midpoint and Simpson quadrature formulas then we have

$$\int_{x_i}^x F'(t)dt = \left(\frac{x - x_i}{2}\right)F'(\frac{x_i + x}{2}) + \frac{x - x_i}{12}[F'(x_i) + 4F'(\frac{x_i + x}{2}) + F'(x)]$$
(4)

From (3) and (4), we have,

$$F(x) = F(x_i) + (\frac{x - x_i}{2})F'(\frac{x_i + x}{2}) + \frac{x - x_i}{12}[F'(x_i) + 4F'(\frac{x_i + x}{2}) + F'(x)]$$

Since, F(x) = 0 then

$$x = x_i - 12 \left[ J(x_i) + 10J(w_i) + J(x) \right]^{-1} \left[ F(x) \right]$$

where

$$w_{i} = \begin{bmatrix} w_{1} \\ w_{2} \\ \vdots \\ w_{n} \end{bmatrix} = \frac{1}{2} \begin{bmatrix} x_{1,i} + x_{1} \\ x_{2,i} + x_{2} \\ \vdots \\ x_{n,i} + x_{n} \end{bmatrix}, \quad x_{i} = \begin{bmatrix} x_{1,i} \\ x_{2,i} \\ \vdots \\ x_{N,i} \end{bmatrix}, \quad J(x) = \begin{bmatrix} \frac{\partial f_{1}}{\partial x_{1}} & \frac{\partial f_{1}}{\partial x_{2}} & \cdots & \frac{\partial f_{1}}{\partial x_{n}} \\ \vdots \\ \frac{\partial f_{n}}{\partial x_{1}} & \frac{\partial f_{n}}{\partial x_{2}} & \cdots & \frac{\partial f_{n}}{\partial x_{n}} \end{bmatrix}$$
(5)

which is an implicit method, since  $x_{n+1}$  occurs on both sides of the equation. To implement this implicit method, one has to compute the approximate solution implicitly, which is itself a difficult problem. To overcome this drawback, one usually uses the prediction and correction technique. With the formulation (5) and with selecting *Predictor-Corrector of Newton* method we will have followed a two-step iterative method for solving the system of nonlinear equation (2) as follows

**Algorithm 1** For a given  $x_0$ , compute the approximate solution  $x_{i+1}$  by iterative scheme.

$$y_i = x_i - J^{-1}(x_i)F(x_i),$$
(6)

$$x_{i+1} = x_i - 12 \left[ J(x_i) + 10J(w_i) + J(y_i) \right]^{-1} F(x_i)$$
<sup>(7)</sup>

## 4. Convergence Analysis

In this section, we consider the convergence of our algorithm for nonlinear equations, In similar way, one can prove the convergence of a system of nonlinear equations.

**Theorem 1** Let *r* be a sample zero of sufficient differentiable function  $f :\subseteq R \to R$  for an open interval *I*. If  $x_0$  is sufficiently close to r, then the two step method defined by our algorithm (5) has convergence is at least of order 3.

Proof. Consider to

$$y_n = x_n - \frac{f(x_n)}{f'(x_n)},\tag{8}$$

$$x_{n+1} = x_n - \frac{12f(x_n)}{f'(x_n) + 10f'(\frac{x_n + y_n}{2}) + f'(y_n)}.$$
(9)

Let r be a simple zero of f. Since f is sufficiently differentiable, by expanding  $f(x_n)$  and  $f'(x_n)$  about r, we get

$$f(x_n) = f(r) + (x_n - r)f'(r) + \frac{(x_n - r)^2}{2!}f^{(2)}(r) + \frac{(x_n - r)^3}{3!}f^{(3)}(r) + \frac{(x_n - r)^4}{4!}f^{(4)}(r) + \cdots,$$

then

$$f(x_n) = f'(r)[e_n + c_2 e_n^2 + c_3 e_n^3 + c_4 e_n^4 + \cdots],$$
(10)

and

$$f'(x_n) = f'(r)[1 + 2c_2e_n + 3c_3e_n^2 + 4c_4e_n^3 + 5c_5e_n^4 + \cdots],$$
(11)

where  $c_k = \frac{1}{k!} \frac{f^{(k)}(r)}{f'(r)}$ , k = 1, 2, 3, ... and  $e_n = x_n - r$ .

Now from (10) and (11), we have

$$\frac{f(x_n)}{f'(x_n)} = e_n - c_2 e_n^2 + 2(c_2^2 - c_3)e_n^3 + (7c_2c_3 - 4c_2^3 - 3c_4)e_n^4 + \cdots,$$
(12)

From (8) and (12), we get

$$y_n = r + c_2 e_n^2 + 2(c_3 - c_2^2) e_n^3 + (-7c_2 c_3 + 4c_2^3 + 3c_4) e_n^4 + \cdots,$$
(13)

From (13), we get,

$$f(y_n) = f'(r)[(y_n - r) + c_2(y_n - r)^2 + c_3(y_n - r)^3 + c_4(y_n - r)^4 + \cdots]$$

and

$$\begin{aligned} f'(y_n) &= f'(r)[1 + 2c_2(y_n - r) + 3c_3(y_n - r)^2 + 4c_4(y_n - r)^3 + 5c_5(y_n - r)^4 + \cdots] \\ &= f'(r)[1 + 2c_2^2e_n^2 + 4(c_2c_3 - c_2^3)e_n^3 + (-11c_2^2c_3 + 8c_2^4 + 6c_2c_4)e_n^4 + \cdots] \end{aligned}$$

Expanding  $f'(\frac{x_n+y_n}{2})$  about *r*, we get

$$\begin{aligned} f'(\frac{x_n + y_n}{2}) &= f'(r)[1 + 2c_2(\frac{x_n + y_n}{2} - r) + 3c_3(\frac{x_n + y_n}{2} - r)^2 + 4c_4(\frac{x_n + y_n}{2} - r)^3 + \cdots], \\ &= f'(r)[1 + c_2e_n + (c_2^2 + \frac{3}{4}c_3)e_n^2 + (\frac{7}{2}c_2c_3 - 2c_2^3 + \frac{1}{2}c_4)e_n^3 \\ &+ (\frac{9}{2}c_2c_4 - \frac{29}{4}c_2^2c_3 + \frac{5}{16}c_5 + 4c_2^4 + 3c_3^2)e_n^4 + \cdots], \end{aligned}$$

then

$$f'(x_n) + 10f'(\frac{x_n + y_n}{2}) + f'(y_n)$$

$$= f'(r)[12 + 12c_2e_n + (12c_2^2 + \frac{21}{2}c_3)e_n^2 + (39c_2c_3 - 24c_2^3 + 9c_4)e_n^3 + (48c_2^4 + 30c_3^2 + 51c_2c_4 - \frac{129}{2}c_2^2c_3 + \frac{65}{8}c_5)e_n^4 + \cdots]$$
  
=  $12f'(r)[1 + c_2e_n + (\frac{7}{8}c_3 + c_2^2)e_n^2 + (\frac{13}{4}c_2c_3 + \frac{3}{4}c_4 - 2c_2^3)e_n^3 + (\frac{5}{2}c_3^2 - \frac{53}{8}c_2^2c_3 + \frac{17}{4}c_2c_4 + \frac{65}{96}c_5 + 4c_2^4)e_n^4 + \cdots]$   
From (9),  $e_{n+1} = x_{n+1} - r$  and  $e_n = x_n - r$ 

$$e_{n+1} = e_n - \frac{12f(x_n)}{f'(x_n) + 10f'(\frac{x_n + y_n}{2}) + f'(y_n)}.$$

Then we will have

$$e_{n+1} = e_n - \left[e_n - (c_2^2 - \frac{1}{8}c_3)e_n^3 - (\frac{27}{8}c_2c_3 - \frac{1}{4}c_4 - 3c_2^3)e_n^4 + \cdots\right]$$

Finally

$$e_{n+1} = (c_2^2 - \frac{1}{8}c_3)e_n^3 + (\frac{27}{8}c_2c_3 - \frac{1}{4}c_4 - 3c_2^3)e_n^4 + \cdots$$
$$\lim_{n \to \infty} \frac{e_{n+1}}{e_n^3} = c_2^2 - \frac{1}{8}c_3 = \frac{1}{4}\left(\frac{f''(r)}{f'(r)}\right)^2 - \frac{1}{48}\frac{f'''(r)}{f'(r)}.$$

which shows that Algorithm 1 is at least a third order convergent method, the required result.

Since asymptotic convergence of Newton method is  $c_2$  and from Theorem 1, we deduce that the convergence rate of our algorithm is better than the Newton method. And the cubic convergent method is vastly superior to the linear and the quadratically convergent methods.

## 5. Numerical Examples

In all of our examples, the maximum number of iteration is N = 200 and our examples are tested with precision  $\varepsilon = 10^{-15}$ . The following stopping criteria is used for computer programs:  $||x_{n+1} - x_n|| < \varepsilon$  or  $||F(x_n)|| < \varepsilon$ . The comparisons with Homotopy perturbation method (HPM) and Newton-Raphson method (NM) are based on the number of iterations *n* which are presented in Tables. The following table contains the test functions used to test the performance of both methods. Here, the numerical computations listed in tables were performed on a computer algebra system called Maple 13 with 20 digits.

# 5.1 Small Systems of Nonlinear Equations

*Example 1* Consider the following system of nonlinear equations (Golbabai & Javidi, 2007), which have an exact solution at (1, 1) using initial approximation  $x_0 = [0.8, 0.8]^T$ .

$$F_1(X) = \begin{cases} f_1(x, y) = x^2 - 10x + y^2 + 8 = 0\\ f_2(x, y) = xy^2 + x - 10y + 8 = 0. \end{cases}$$

Table 1. Numerical results for Example 1

$F_1$	Newton-Raphson		HI	PM	Present method		
NI	х у		Х	У	Х	У	
1	.987766198459447	.985772541912097	0.998270352547736	0.997314611742872	.99909530066340	.99880312274025	
2	.999935603124850	.999916479630138	0.999999995819455	0.999999993343152	.999999999980016	.999999999972169	
3	.9999999997926065	.9999999997265203	0.999999999999999999	1.0000000000000000	1.00000000000000	1.00000000000000	

*Example 2* Consider the following system of nonlinear equations (Ozel, 2010) which have an exact solution at (1, 1) using initial approximation  $x_0 = [0.5, 0.5]^T$ 

$$F_2(X) = \begin{cases} f_1(x, y) = x^2 y - 1 + x - y = 0\\ f_2(x, y) = x^2 + y^2 - 2 = 0, \end{cases}$$

Table 2. Numerical results for Example 2

$F_2$	Newton-	Raphson	HI	PM	Present method		
NI	х у		х у		Х	У	
1	1.3888888888888888888888888888888888888	1.11111111111111111	0.584019204389574	0.752400548696844	0.855913518714005	1.013756564266330	
2	1.069355784167179	.9869163809021366	0.905274466918989	0.954281702145681	0.999461132799700	1.000043296108220	
3	1.000848429856370	1.001604421204614	0.999532744494716	0.999814121537017	0.9999999999974393	1.00000000005090	
4	1.000001146049355	1.000000499177722	0.9999999999957910	0.999999999981766	1.0000000000000000	1.0000000000000000	

*Example 3* Consider the following system of nonlinear equations (Golbabai & Javidi, 2007), which have an exact solution at (1, 1) using initial approximation  $x_0 = [0.8, 0.8]^T$ 

$$F_3(X) = \begin{cases} f_1(x, y) = x^4 y - xy + 2x - y - 1 = 0\\ f_2(x, y) = ye^{-x} + x - y - e^{-1} = 0, \end{cases}$$

Table 3. Numerical results for Example 3

$F_3$	Newton-	Raphson	HI	PM	Present method		
NI	х у		Х	x Y		У	
1	1.202154745465234	1.252500021237429	0.530135422051184	0.375112183418822	.915589914489711	.900999078672304	
2	1.066458500832922	1.071778294601873	1.167578846210930	0.826948359050393	.995063851974896	.994094932023846	
3	1.008970459700100	1.009951147648418	1.002859536677230	0.999714844009216	.999999280230771	.999999153894058	
4	1.000190149835321	1.000217129214667	1.00000085472060	1.00000076036040	1.000000000000000	1.00000000000001	

*Example 4* Consider the following system of nonlinear equations (Hosseini & Kafash, 2010) which has a solution, at X = (0.532370372327903, 0.351257447590883), using initial approximation  $x_0 = [0.53, 0.35]^T$ 

$$F_4(X) = \begin{cases} f_1(x, y) = x^3 + y^3 - 6x + 3 = 0\\ f_2(x, y) = x^3 - y^3 - 6y - 2 = 0, \end{cases}$$

Table 4. Numerical results for Example 4

$F_4$	Newton-	Raphson	HI	PM	Present method			
NI	x y		Х	У	Х	У		
1	.5323682133175740	.3512560178033324	0.532370365139782	0.351257442898170	.532370369897060	.351257446061485		
2	.5323703723259607	.3512574475897934	0.532370372327903	0.351257447590883	.532370372327902	.351257447590882		
3	.5323703723279030	.3512574475908831	0.532370372327903	0.351257447590883				
4	.5323703723279030	.3512574475908831						

*Example 5* In a case three dimension, consider the following systems of nonlinear functions (Awawdeh, 2009), and (Hosseini & Kafash, 2010) where  $x^* = (1.0421496, 1.0310913, 0.92384815)^T$  is the solution of this system, using initial approximation  $x_0 = [5, 4, 2.]^T$ 

$$F_5(X) = \begin{cases} f_1(x, y, z) = 15x + y^2 - 4z - 13 = 0\\ f_2(x, y, z) = x^2 + 10y - e^{-z} - 11 = 0\\ f_3(x, y, z) = y^3 - 25z + 22 = 0, \end{cases}$$

Table 5. Numerical results for Example 5

$F_5$		HPM		Present method				
NI	Х	У	Z	Х	У	Z		
1	0.858102935922979	0.995520909735443	-1.728542081154890	0.952626720902665	0.992732766696450	-0.851587051863595		
2	0.938083254462730	1.791392442270120	1.107814211025650	1.041578243270440	1.040378737490350	0.923799502004400		
3	1.046987147534600	1.030216151827310	0.940390410810817	1.042149561658460	1.031091271275050	0.923848158979542		
4	1.042149563794840	1.031091242405770	0.923848150437484	1.042149560576930	1.031091271839400	0.923848154879367		
5	1.042149560576930	1.031091271839400	0.923848154879367	1.042149560576930	1.031091271839400	0.923848154879367		
6	1.042149560576930	1.031091271839400	0.923848154879367					

*Example 6* In a three dimension case, consider the following systems of nonlinear functions (Awawdeh, 2009) where  $x^* = (.5, 0, -.5235987755982)^T$  is the solution of this system, using initial approximation  $x_0 = [5., 4., 2.]^T$ 

$$F_6(X) = \begin{cases} f_1(x, y, z) = 3x - \cos(yz) - 0.5 = 0\\ f_2(x, y, z) = x^2 - 81(y + 0.1)^2 + \sin z + 1.06 = 0\\ f_3(x, y, z) = e^{-xy} + 20z + \frac{10\pi - 3}{3} = 0, \end{cases}$$

$\overline{F_6}$		HPM		Present method				
NI	Х	У	Z	Х	У	Z		
1	41.174379524898	2.3305907045959	-0.4735987765029	0.3264837147307	1.59964685726156	0.575667883890184		
2	0.8354962027095	-4.6996404543213	-0.4735987755982	0.4391032545486	0.47233538404662	-0.51724957483991		
3	1.0703427923466	-1.8282755657884	-0.7967631804740	0.5008820391011	0.10621342346362	-0.52072671134879		
4	0.4537241849972	-0.7521205589332	-0.5330687052474	0.5000787511491	0.00872754527967	-0.52337058806636		
5	0.4968557700375	-0.3558140012447	-0.5330863345798	0.5000001340783	0.00001467555805	-0.52359839168772		
6	0.4979211091684	-0.2242487141958	-0.5294699721794	0.50000000000000	0.00000000000007	-0.52359877559829		

Table 6. Numerical results for Example 6

Table 7. Numerical results for Number of iterations

Number of iterations	$F_1$	$F_2$	$F_3$	$F_4$	$F_5$	$F_6$
NR	5	7	8	4	7	15
HPM	4	5	6	3	6	10
Present method	3	4	5	3	5	8

## 5.2 Large Systems of Nonlinear Equations

In this subsection, we test with some sparse systems with m unknowns. For the following examples, we can see that all NR and HPM methods with the iteration numbers in comparison with the present method.

Example 7 Consider the following system of nonlinear equations (Darvishi & Shin, 2011):

$$F_7$$
:  $f_i = e^{x_i} - 1$ ,  $i = 1, 2, ..., m$ .

The exact solution of this system is  $x^* = (0, 0, ..., 0)^T$ . To solve this system we set  $x_0 = (0.5, 0.5, ..., 0.5)$  as an initial guess. The results are presented in Table 8.

Example 8 Consider the following system of nonlinear equations:

$$F_8$$
:  $f_i = x_i^2 - \cos(x_i - 1), \quad i = 1, 2, ..., m.$ 

One of the exact solutions of this system is  $x^* = (1, 1, ..., 1)^T$ . To solve this system we set  $x_0 = (2, 2, ..., 2)$  as an initial guess. The results are presented in Table 8.

Example 9 Consider the following system of nonlinear equations (Darvishi & Shin, 2011):

$$F_9$$
:  $f_i = \cos x_i - 1$ ,  $i = 1, 2, ..., m$ .

One of the exact solutions of this system is  $x^* = (0, 0, ..., 0)$ . To solve this system we set  $x_0 = (2, 2, ..., 2)$  as an initial guess. The results are presented in Table 8.

Table 8. Numerical results for Example 7, 8 and 9

Number	of iterations	$F_7$	$F_8$	$F_9$	$F_7$	$F_8$	$F_9$	$F_7$	$F_8$	$F_9$
$\mathcal{E}$ :	$= 10^{-13}$		m=50	)		m=75		I	n=10	0
	NR	5	7	21	5	7	21	5	7	21
	HPM	4	5	15	4	5	15	4	5	15
Prese	nt method	3	4	14	3	4	14	3	4	14

#### 6. Conclusions

The present two step method is generalized and applied for solving system of nonlinear algebraic equations. The numerical results in Tables show that our method is very effective and provide highly accurate results in a less number of iterations as compared with Newton-Raphson method and Homotopy perturbation method.

#### References

Abbasbandy, S. (2005). Extended Newton's method for a system of nonlinear equations by modified Adomian decomposition method. *Journal of applied mathematical and computation*, *170*, 648-6567.

- Awawdeh, F. (2009). On new iterative method for solving systems of nonlinear equations. Springer. http://dx.doi.org/10.1007/s11075-009-9342-8
- Burden, R. L., & Farires, J. D. (1985). Numerical Analysis, 3rd ed. Boston.
- Chun, C. (2006). A new iterative method for solving nonlinear equations. *Journal of applied mathematical and computation*, 178(2), 415-422.
- Darvishi, M. T., & Barati A. (2007a). A third-order Newton-type method to solve systems of nonlinear equations. *Journal of applied mathematical and computation*, 187, 630-635.
- Darvishi, M. T., & Barati, A. (2007b). Super cubic iterative methods to solve systems nonlinear equations. *Journal* of applied mathematical and computation, 188, 1678-1685.
- Darvishi, M. T., & Shin, B. (2011). High-Order Newton-Krylov Methods to Solve Systems of Nonlinear Equations. *J. KSIAM*, 15, 19-30.
- Golbabai, A., & Javidi, M. (2007). A new family of iterative methods for solving system of nonlinear algebraic equations. *Journal of applied mathematical and computation, 190*, 1717-1722.
- Hafiz, M. A., & Bahgat, M. S. M. (2012). Modified of Householder iterative method for solving nonlinear system of equations. *Journal of mathematical and computational science* (Under Publication).
- He, J. H. (1999). Homotopy perturbation technique. *Computer methods in applied mechanics and engineering*, 178(3-4), 257-262. http://dx.doi.org/10.1016/S0045-7825(99)00018-3
- Hosseini, M. M., & Kafash, B. (2010). An Efficient Algorithm for Solving System of Nonlinear Equations. *Journal* of applied mathematical and computation, 4, 119 131.
- Jafari, H., & Gejji, V. D. (2006). Revised Adomian decomposition method for solving a system of nonlinear equations. *Journal of applied mathematical and computation*, 175, 1-7.
- Noor, M. A., & Noor, K. I. (2006). Improved iterative methods for solving nonlinear equations. *Journal of applied mathematical and computation*, *183*, 774-779.
- Ozel, M. (2010). A New Decomposition Method For Solving System Of Nonlinear Equations. *Journal of applied mathematical and computation*, *15*, 89-95.
- Soheili, A. R., Ahmadian, S. A., & Naghipoor, J. (2008). A Family of Predictor-Corrector Methods Based on Weight Combination of Quadratures for Solving Nonlinear Equations. *International journal of nonlinear science*, 6, 29-33.
- Vahidi, A. R., Javadi, Sh., & Khorasani, S. M. (2012). Solving System of Nonlinear Equations by restarted Adomain's method. *Journal of applied mathematical and computation*, *6*, 509-516.