Gen. Math. Notes, Vol. 2, No.2, February 2011, pp. 64-72
ISSN 2219-7184; Copyright © ICSRS Publication, 2011
www.i-csrs.org
Available free online at http://www.geman.in

An Improved Newton's Method Without Direct Function Evaluations

Behzad Ghanbari[1,*] and Mehdi Gholami Porshokouhi [2]

[1]Department of Mathematics, Takestan Branch, Islamic Azad University, Takestan, Iran
E-mail: b.ghanbary@yahoo.com
[2]Department of Mathematics, Takestan Branch, Islamic Azad University, Takestan, Iran
E-mail: m.gh.porshokouhi@tiau.ac.ir

(Received: 24-12-10 /Accepted: 30-1-11)

Abstract

Due to the fact that systems of nonlinear equations arise frequently in science and engineering, they have recently attracted researchers' interest. In this work, we present a new Newton-like approach which is independent of function evaluation and has been provided using an original idea that improves some definitions and notions of a recently proposed method [1] for solving systems of nonlinear. Also, the convergence of proposed method has been discussed. The computational advantages and convergence rate of the proposed method are also tested via some numerical experiments. From the obtained numerical results it seems that present approach affect considerably the overall performance in relation to Newton's method and its aforementioned variants.

Keywords: *Iterative methods, System of nonlinear equations, Newton's method*

1 Introduction

Let us consider the problem of finding a real zero, say $x^* = (x_1^*, x_2^*, \ldots, x_n^*)$, of a system of nonlinear equations

* Corresponding author

$$\begin{cases} f_1(x_1, x_2, \ldots, x_n) = 0, \\ f_2(x_1, x_2, \ldots, x_n) = 0, \\ \vdots \\ f_n(x_1, x_2, \ldots, x_n) = 0. \end{cases} \tag{1}$$

This system can be referred by $F(x) = 0$, where $F = (f_1, f_2, \ldots, f_n) : D \subseteq \mathbb{R}^n \to \mathbb{R}^n$ is continuously differentiable on an open neighborhood $D^* \subseteq D$ of x^*.

The most widely used iterative scheme for solving systems of nonlinear equations is Newton's method, giving by

$$x^{p+1} = x^p - F'(x^p)^{-1} F(x^p), \tag{2}$$

Where $F'(x^p)^{-1}$ denote the Jacobian matrix at the current approximation $x^p = (x_1^p, x_2^p, \ldots, x_n^p)$ and x^{p+1} is the next approximation. It is well known that the method has quadratic convergence.

Since there exists no general method that yields an exact solution of (1), in recent years, a large number of algorithms and methods of solutions of different orders have been derived and studied in the literature. These attempts and considerations are mainly because of their numerous influences and applications in real applications such as in science and engineering [2, 3].

Some approaches to solving such systems were proposed using decomposition method [4], quadrature formulas [5, 6] and other techniques [7, 8].

In some cases, we may encounter with different kinds of (1), while the nonlinear system is known with some precision only, for example when the function and derivative values depend on the results of numerical simulations [9], or in some cases the precision of the desired function is available at a prohibitive cost or when the function value results from the sum of an infinite series (e.g. Bessel or Airy functions [10, 11]).

In [12] a method without evaluation of nonlinear function values is proposed which can be applied for polynomial systems. Also, these methods are ideal for situations with unavailable accurate function values or high computational cost [13].

So, it is very important to obtain methods which are free of function evaluations, i.e., the required function values are not directly evaluated from the corresponding component functions f_i but are approximated by using appropriate quantities, as have recently used in so-called IWFEN method [1].

Here, we generalize some used definitions in IWFEN method and then by using a new geometrical interpretation a method which is a new improved Newton's method without direct function evaluations is presented.

The rest of this contribution is structured as follows. In Section 2, we develop the new method and its convergence theorem is proved. Some numerical examples and some comparisons between the results of different approaches and our proposed method are given in Section 3. Finally, conclusions are drawn in Section 4.

2 New Approach to the Solution of the Problem

To develop a new method, let us have the following definition of pivot points which extends the corresponding existing ones.

Definition 1. *For any* $i \in \{1,2,\ldots,n\}$ *and* $p = 1,2,\ldots$ *and based on* $n-1$ *components of current point* x^p*, we generalize the definition of pivot points in [1]*

$$x_{pivot}^{p,i} = (x_1^p, x_2^p, \ldots, x_{n-1}^p, x_n^{p,i}) \tag{3}$$

as its new following form

$$x_{pivot}^{p,i} = (x_1^p, x_2^p, \ldots, x_{j(i)}^{p,i}, \ldots, x_{n-1}^p, x_n^p). \tag{4}$$

It is obvious that the only difference between our definition (4) and (3) is that in (3) $j(i)$ *has considered to be the fix value of* n*, but in our definition* $j(i) \in \{1,2,\ldots,n\}$ *is not known in prior and should be obtained in such way that will be introduced later.*
Moreover, we impose the pivot points (4) to be lied on a parallel line to $j(i)$ *axis, which passes through the current point* x^p *any iteration* p *of the algorithm. From the definition of the pivot points, it is obvious, that these points have the same* $n-1$ *components with the current point* x^p *and differ only at the* $j(i)$ *th component.*

Definition 2. *Let's define the functions* $g_{j(i)}^{p,i} : \mathbb{R} \to \mathbb{R}$, as

$$g_{j(i)}^{p,i}(t) = f_i(x_1^p, x_2^p, \ldots, x_{j(i)-1}^p, t, x_{j(i)+1}^p, \ldots, x_{n-1}^p, x_n^p). \tag{5}$$

From (5) and the imposed property to pivot points (4), it is evident that the unknown $j(i)$*-th component of (4),* $x_{j(i)}^{p,i}$*, can be found by solving each of the corresponding one dimensional equations*

$$g_{j(i)}^{p,i}(t) = 0 \tag{6}$$

According to Implicit Function Theorem [14] there exist unique mappings ϕ_i *such that* $x_{j(i)} = \phi_i(y), f_i(y; \phi_i(y)) = 0$ and therefore

$$x_{j(i)}^{p,i} = \phi_i(y^p).$$

Where $y = (x_1, x_2, \ldots, x_{j(i)-1}, x_{j(i)+1}, \ldots, x_{n-1}, x_n)$.
Similar in [1], in this paper the sign-function based method [15] is used for solving the corresponding one-dimensional equations.
It is clear that the solution of (6) is depending on the expression of the components f_i and the current approach x^p. That is, if any of the Eq. (6) has no zeros, we are not able to apply our proposed method on a system of equations. Here, similar to what was discussed in [1], we can adopted some techniques to guarantee the existence of pivot points. For example choosing a linear combination between the components like in [16] or applying either a reordering technique like in [17]). For the needs of this work we consider that we are always able to find the zeros of (6) is possible.
The key idea in this paper is to substitute the function value of $F(x^p) = (f_1(x^p), f_2(x^p), \ldots, f_{n-1}(x^p), f_n(x^p))$, in Newton's method (2) to the its suitable

approximation. Hence, let us use the first orders Taylor expansion of $g^{p}_{j(i)}$ around the point $t = x^{p}_{j(i)}$ as

$$g^{p,i}_{j(i)}(t) \approx g^{p,i}_{j(i)}(x^{p}_{j(i)}) + \frac{dg^{p,i}_{j(i)}}{dt}(x^{p}_{j(i)})(t - x^{p}_{j(i)}). \tag{7}$$

Setting $t = x^{p,i}_{j(i)}$ at (7) and in view of (5), we have

$$g^{p,i}_{j(i)}(x^{p,i}_{j(i)}) \approx f_i(x^p) + \partial_{j(i)} f_i(x^p)(x^{p,i}_{j(i)} - x^{p}_{j(i)}) \tag{8}$$

Due to the property of pivot points (i.e. $g^{p,i}_{j(i)}(x^{p,i}_{j(i)}) = 0$, the relation (8) becomes

$$f_i(x^p) \approx \partial_{j(i)} f_i(x^p)(x^{p}_{j(i)} - x^{p,i}_{j(i)}) \tag{9}$$

The relation (9) is so important for the development of our approach, because using (9) in (2) will transform it to a new Newton's method, which will not depend directly on the function values $f_i(x^p)$, but on $\partial_{j(i)} f_i(x^p)$ and the n-th components of the points x^p and $x^{p,i}_{pivot}$.

In Fig. 1, we can see the behavior of function $g^{p,i}_{j(i)}(t)$ for any $j(i) \in \{1,2,\ldots,n\}$. If we bring from the pivot point $B = (x^{p,i}_{pivot,j(i)}, 0)$, the parallel line to the tangent of the function at the point $P = (x^{p}_{j(i)}, f_i(x^p))$. From the similar triangles, the function value $f_i(x^p)$, denoted by the segment AP, can be approximated by the quantity $\partial_{j(i)} f_i(x^p)(x^{p,i}_{j(i)} - x^{p}_{j(i)})$, denoted by the segment AQ. This plot can give us a good idea to choose a suitable direction of $j(i)$.Using similarity in triangles, it can be verified that whenever segment BC has a fewer length, the approximation $\partial_{j(i)} f_i(x^p)(x^{p,i}_{j(i)} - x^{p}_{j(i)})$ instead of $f_i(x^p)$ is more valid. So, we should choose that direction $j(i)$ which minimizes the expression

$$|BC| = \left| x^{p}_{j(i)} - \frac{f_i(x^p)}{\partial_{j(i)} f_i(x^p)} - x^{p,i}_{j(i)} \right| \tag{10}$$

From triangular inequality in (10), we have

$$|BC| \le \left| x^{p}_{j(i)} - x^{p,i}_{j(i)} \right| + \left| \frac{f_i(x^p)}{\partial_{j(i)} f_i(x^p)} \right| \tag{11}$$

It is clear that, the expression $\left| \frac{f_i(x^p)}{\partial_{j(i)} f_i(x^p)} \right|$ minimizing whenever the numerator expression achieves its maximum value.

Hence, in this paper we let $j(i) = J$ whereas for any $k \in \{1,2,\ldots,n\}$, J be the smallest index which we have $|\partial_J f_i(x^p)| \ge |\partial_k f_i(x^p)|$, i.e. that direction which has the steepest slope among the components of the gradient vector of function f_i at the point x^p.

Now, using (9) in Newton method (2), we have

$$V(x^p) L(x^p) + F'(x^p)(x - x^p) = 0 \tag{12}$$

Where

$$V(x^p)=\begin{bmatrix} \partial_{j(1)}f_1(x^p) & 0 & \dots & 0 \\ 0 & \partial_{j(2)}f_1(x^p) & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \partial_{j(n)}f_1(x^p) \end{bmatrix} \text{ and } L(x^p)=\begin{bmatrix} x^p_{j(1)}-\phi_1(y^p) \\ x^p_{j(2)}-\phi_2(y^p) \\ \vdots \\ x^p_{j(n)}-\phi_n(y^p) \end{bmatrix}=\begin{bmatrix} x^p_{j(1)}-x^{p,i}_{j(1)} \\ x^p_{j(2)}-x^{p,i}_{j(2)} \\ \vdots \\ x^p_{j(n)}-x^{p,i}_{j(n)} \end{bmatrix}.$$

Under the assumptions of Implicit Function Theorem the diagonal matrix $V(x^p)$ is invertible and (12) becomes

$$V(x^p)^{-1}F'(x^p)(x-x^p)=-L(x^p)$$

Now, we consider the function

$$L(x)=(x_{j(1)}-\phi_i(y),x_{j(2)}-\phi_2(y),\dots,x_{j(n)}-\phi_n(y))^T \tag{13}$$

Utilizing again the Implicit Function Theorem to derive $\partial_j\phi_i(x)$, we have

$$L'(x)=\begin{bmatrix} \frac{\partial_1 f_1(x)}{\partial_{j(1)}f_1(x)} & \frac{\partial_2 f_1(x)}{\partial_{j(1)}f_1(x)} & \dots & 1 & \dots & \frac{\partial_n f_1(x)}{\partial_{j(r)}f_1(x)} \\ \frac{\partial_1 f_2(x)}{\partial_{j(2)}f_2(x)} & \dots & 1 & \dots & \frac{\partial_s f_2(x)}{\partial_{j(2)}f_2(x)} & \frac{\partial_n f_2(x)}{\partial_{j(1)}f_2(x)} \\ \vdots & \vdots & \ddots & \ddots & \vdots & \vdots \\ \frac{\partial_1 f_n(x)}{\partial_{j(n)}f_n(x)} & \frac{\partial_2 f_n(x)}{\partial_{j(n)}f_n(x)} & \dots & \dots & 1 & \frac{\partial_n f_n(x)}{\partial_{j(n)}f_n(x)} \end{bmatrix}=V(x)^{-1}F'(x), \tag{14}$$

Where the entries of 1 occur at $j(i)$-th column for i-th row.
Finally, Eqs. (13) and (14) introduce iterative method given by

$$x^{p+1}=x^p-L'(x^p)^{-1}L(x^p), \tag{15}$$

A similar convergence theorem to what stated in [1], can be presented and proved, as follows.

Theorem 1. *Let $F=(f_1,f_2,\dots,f_n):D\subseteq\mathbb{R}^n\to\mathbb{R}^n$ be sufficiently differentiable at each point of an open neighborhood D of $x^*\in\mathbb{R}^n$, that is a solution of the system $F(x)=0$. Let us suppose that $F'(x)$ is continuous and nonsingular in x^*. Then the sequence $\{x^k\}_{k\geq 1}$ obtained using the iterative scheme (14) for sufficiently close initial guess x^0, converges to x^* with convergence order of two.*

Proof. Using the mapping $L=(l_1,l_2,\dots,l_n)^T:D\subseteq\mathbb{R}^n\to\mathbb{R}^n$, where

$$l_i(x)=x_n-\phi_i(y),$$

The iteration of our method is given by $x^{p+1} = x^p - L'(x^p)^{-1} L(x^p)$ For the above mapping the well-known conditions of Newton's theorem (see [14]) are obviously fulfilled because of the form of $L'(x)$, stated in (14) and the property of pivot points $x_n^* - \varphi_i(y^*)$, for $i = 1,2,\ldots,n$. The convergence theorem for mapping L is proved, and the iterations x^p of (15) converge to x^* quadratic ally. □

Using this proposed idea, we may expect that our method will have better convergence than the method of IWFEN method. The results of experiments in Section 3 will confirm this conjecture.

Also, it should be pointed out that the proposed new procedure remains the cost of IWFEN method. This is because of it does not need more computational cost, since the used new partial derivatives have already been evaluated in the corresponding Jacobian matrix.

3 Numerical Examples

In this section, we perform some numerical experiments for testing the convergence of the iterative method proposed in the previous section.

In order to compare the results, we take the same examples which were used in [1]. In Tables 1-2, we present the results obtained, for various initial points by Newton's method, IWFEN and our new proposed method.

Example 1. The first system has two roots $x_1^* = (0.1, 0.1, 0.1)$ and $x_2^* = (-0.1, -0.1, -0.1)$. It is given by:

$$f_1(x_1, x_2, x_3) = x_1^3 - x_1 x_2 x_3 = 0,$$
$$f_1(x_1, x_2, x_3) = x_2^2 - x_1 x_3 = 0,$$
$$f_3(x_1, x_2, x_3) = 10 x_1 x_3 + x_2 - x_1 - 0.1 = 0.$$

Example 2. The second example is

$$f_1(x_1, x_2, x_3) = x_1 x_3 - x_3 e^{x_1^2} + 10^{-4} = 0,$$
$$f_2(x_1, x_2, x_3) = x_1(x_1^2 + x_2^2) + x_2^2(x_3 - x_2) = 0,$$
$$f_3(x_1, x_2, x_3) = x_1^3 + x_3^3 = 0.$$

With the solution $x^* = (-0.9999001 \times 10^{-4}, -0.9999001 \times 10^{-4}, 0.9999001 \times 10^{-4})$.

Results were obtained by using Maple software via 30 digit floating point arithmetic (Digits:=30). The iterative process will stop if

$$\left\| x^{k+1} - x^k \right\| < 10^{-14}.$$

From Tables 1-2, we see that the results of computation for the proposed iterative method admit the theoretical order of convergence in Theorem 1. Furthermore the proposed iterative method converges much faster than the other compared methods.

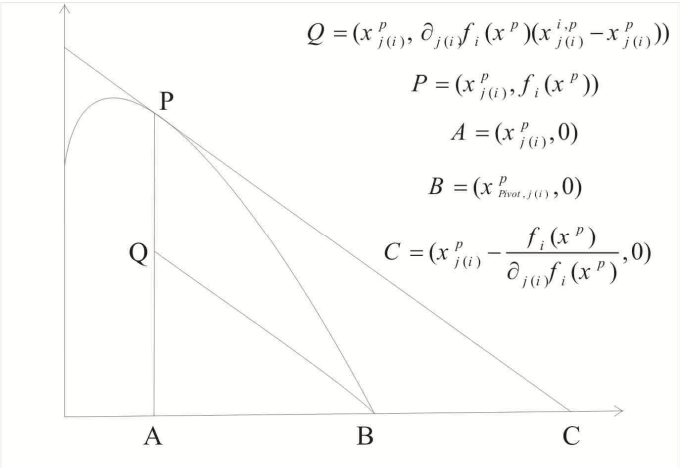

Fig 1. The behavior of function $g^{p,i}_{j(i)}(t)$

Table1
Comparison between different methods for example 1.

x^0_1	x^0_2	x^0_3	Newton			IWFEN			Present Method		
			IT	FE	x^*_i	IT	FE	x^*_i	IT	FE	x^*_i
0.4	*0.5*	*0.5*	*53*	*636*	x^*_2	*20*	*240*	x^*_2	16	144	x^*_2
-4	*-2*	*1*	*33*	*396*	x^*_2	*33*	*396*	x^*_2	23	207	x^*_2
-1	*-2*	*0.6*	*51*	*612*	x^*_1	*51*	*612*	x^*_1	10	90	x^*_1
-1	*-2*	*1*	*29*	*384*	x^*_2	*29*	*384*	x^*_2	19	171	x^*_2
0.5	*2*	*1*	*54*	*648*	x^*_1	*54*	*648*	x^*_1	18	162	x^*_1
5	*-2*	*-2*	*38*	*456*	x^*_1	*38*	*456*	x^*_1	18	162	x^*_1
10	*-2*	*-2*	*39*	*468*	x^*_1	*39*	*468*	x^*_1	33	297	x^*_1

Table2
Comparison between different methods for example 2.

x^0_1	x^0_2	x^0_3	Newton		IWFEN		Present Method	
			IT	FE	IT	FE	IT	FE

2	*2*	*2*	*42*	*504*	*38*	*342*	12	108
-2	*-2*	*-2*	*27*	*324*	*27*	*243*	7	63
3	*3*	*5*	*92*	*1104*	*18*	*162*	24	216
4	*4*	*4*	*73*	*876*	*26*	*234*	27	243
0.5	*0.5*	*0.5*	*46*	*552*	*32*	*288*	7	63
1	*1*	*5*	*37*	*444*	*37*	*333*	11	99
-4	*-1*	*-2*	*28*	*336*	*26*	*234*	17	153

4 Conclusion

In this paper we present a generalization of a recently proposed variant of Newton's method for solving nonlinear systems. This new method has the order of convergence two and is independent of function evaluation. Also, the proposed method can be used in some systems where the function calculations are quite costly or cannot be done precisely. As seen in Tables [1-2], the numerical results of the proposed method are quite satisfactory and admit the geometrical explanations. In some cases the results of ourselves are very acceptable and there is a sufficient reduction on the number of iterations and hence the proposed method seems to be a reliable refinement for Newton’s method and some its recent modifications.

References

[1] E.N. Malihoutsaki, I.A. Nikas and T.N. Grapsa, Improved Newton’s method without direct function evaluations, *Journal of Computational and Applied Mathematics*, 227 (2009), 206-212.

[2] R.L. Burden and J.D. Faires, *Numerical Analysis*, (*7th ed*.), PWS Publishing Company, Boston, (2001).

[3] J.M. Ortega, W.C. Rheinboldt, *Iterative Solution of Nonlinear Equations in Several Variables*, Academic Press, (1970).

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

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

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

[7] D.K.R. Babajee, M.Z. Dauhooa, M.T. Darvishi, A. Karami and A. Barati, Analysis of two Chebyshev-like third order methods free from second derivatives for solving systems of nonlinear equations, *Journal of Computational and Applied Mathematics*, 233 (2010), 2002-2012.

[8] J. Biazar and B. Ghanbary, A new approach for solving systems of nonlinear equations, *International Mathematical Forum*, 3(38) 2008, 1889–1889.

[9] M. Kupferschmid and J.G. Ecker, A note on solution of nonlinear programming problems with imprecise function and gradient values, *Math. Program. Study*, 31 (1987), 129-138.

[10] M.N. Vrahatis, T.N. Grapsa, O. Ragos and F.A. Zafiropoulos, On the localization and computation of zeros of Bessel functions, *Z. Angew. Math. Mech.*, 77(6) (1997), 467-475.

[11] M.N. Vrahatis, O. Ragos, F.A. Zafiropoulos and T.N. Grapsa, Locating and computing zeros of Airy functions, *Z. Angew. Math. Mech.*, 76 (7) (1996), 419-422.

[12] W. Chen, A Newton method without evaluation of nonlinear function values, *CoRR*, cs.CE/9906011, (1999).

[13] T.N. Grapsa and E.N. Malihoutsaki, Newton's method without direct function evaluations, in: E. Lipitakis (Ed.), *Proceedings of 8th Hellenic European Research on Computer Mathematics & its Applications-Conference, HERCMA 2007*, (2007).

[14] J.M. Ortega and W.C. Rheinboldt, *Iterative Solution of Nonlinear Equations in Several Variables*, Academic Press, New York, (1970).

[15] T.N. Grapsa and M.N. Vrahatis, A dimension-reducing method for solving systems of nonlinear equations in, *Int. J. Comput. Math.*,32 (1990), 205-216.

[16] T.N. Grapsa and M.N. Vrahatis, A dimension-reducing method for unconstrained optimization, *J. Comput. Appl. Math.*, 66(1-2) (1996), 239-253.

[17] D.G. Sotiropoulos, J.A. Nikas and T.N. Grapsa, Improving the efficiency of a polynomial system solver via a reordering technique, in: D.T. Tsa-halis (Ed.), *Proceedings of 4th GRACM Congress on Computational Mechanics*, III (2002).