Solving Nonlinear Equation with any Integer Order of Convergence by A New Neotan-type

Pohanyar Hayatullah masomy

Teacher Department of Mathematics, faculty of Science, Allahabad University, India.

Abstract

In this paper, we present a new Newton-type iterative method to solve nonlinear equations with any integer order of convergence. It is constructed by the combination of two existing low order methods. This method can also be extended to solve systems of nonlinear equations. Finally, we give some numerical examples to demonstrate our method is more efficient than other classical iterative methods.

Keywords: Nonlinear Equations; Systems of Nonlinear Equations; Newton’s Method; Iterative Methods; Order of Convergence

1. Introduction

Solving nonlinear equations is an important issue in pure and applied mathematics. Researchers have developed various effective methods to find a single root \( x^* \) of the nonlinear equation \( f(x) = 0 \), where \( f : D \subset \mathbb{R} \rightarrow \mathbb{R} \) is a scalar function on an open interval \( D \). As we know Newton’s method is one of the best iterative methods to find \( x^* \) by using

\[
x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}
\]

The converges quadratically in some neighborhood of \( x^* \) [1]

Potra and Ptak [2] proposed a modification of Newton’s method with third-order convergence defined by

\[
x_{n+1} = x_n - \frac{f(x_n) + f'(x_n)(x_n - f(x_n)/f'(x_n))}{f'(x_n)}
\]

King [3] constructed a one-parameter family of fourth-order methods which is written as
\[
\begin{aligned}
  x_{n+1} &= x_n - \frac{f(x_n)}{f'(x_n)} \\
  x_{n+1} &= x_n - \frac{f(x_n) + f'(x_n - f(x_n)/f'(x_n))}{f'(x_n)},
\end{aligned}
\]

Where \( \beta \in R \) is a constant.

There are plenty of modified iterative methods to improve the order of convergence or to simplify the computation in open literatures. For more details, see [4-13] and the references therein. In this paper, we develop a new Newton-type iterative method to find a single root of nonlinear equations with any integer order of convergence.

2. Preliminaries

We firstly introduce two lemmas before proposing our Newton-type iterative method for solving nonlinear equations.

**Lemma 1.** Assume that \( f \in c^p(D) \) and there is a simple root \( \alpha \) of the nonlinear equation \( f(x) = 0 \), where \( f : D \subset R \rightarrow R \) is a scalar function on an open interval \( D \). If there is an iterative method \( \varphi(x) \) with the order of convergence \( p \) (\( p \) is an integer) which produce the sequence \( \{x_n\} \) then we have

\[
x_{n+1} - \alpha = A(x_n - \alpha)^p + o((x_n - \alpha)^{p+1}) \quad \text{where} \quad x_n \subset U(\alpha)
\]

(1)

Where constant \( A \neq 0 \) and \( U(\alpha) \) is a neighborhood of \( \alpha \).

**Proof.** Using the Taylor expansion and the definition of the convergence order of the iterative method, we know

\[
x_{n+1} = \varphi(x_n)
= \varphi(\alpha) + \varphi(\alpha)(x_n - \alpha) + \frac{\varphi(\alpha)}{2}(x_n - \alpha)^2 + \cdots + \frac{\varphi^{(p)}(\alpha)}{p!}(x_n - \alpha)^p + o((x_n - \alpha)^{p+1})
= \alpha + \frac{\varphi^{(p)}(\alpha)}{p!}(x_n - \alpha)^p + o((x_n - \alpha)^{p+1})
\]

Then, we get (1). The lemma 1 proof is finished.

**Lemma 2.** Suppose that \( f \in c^{p+q}(D) \) and there is a simple root \( \alpha \) of the nonlinear equation \( f(x) = 0 \), where \( f : D \subset R \rightarrow R \) is a scalar function on an open interval \( D \). For two iterative functions \( u(x) \) and \( v(x) \), their convergence order are \( p \) and \( q \) respectively (\( p, q \) are integers and \( p > q \)). If their results of the \( n+1 \)th iteration are \( u_{n+1}, v_{n+1} \) then the convergence order of following iterative formula is \( p + q \)

\[
x_{n+1} = u_n - \frac{f(u_{n+1})}{f'(x_{n+1})},
\]

(2)

**Proof.** Since the order convergence of \( \{u_{n+1}\} \) and \( \{v_{n+1}\} \) is \( p \) and \( q \), for the neighborhood close to \( \alpha \),
we use lemma 1 to get
\[ u_{n+1} - \alpha = A(u_n - \alpha)^p + A_1(u_n - \alpha)^{p+1} + A_2(u_n - \alpha)^{p+2} + \ldots \]
\[ + A_s(u_n - \alpha)^{2p} + o((u_n - \alpha)^{2p+1}), \]
\[ v_{n+1} - \alpha = B(u_n - \alpha)^q + o((u_n - \alpha)^{q+1}), \]

Where $A \neq 0, B \neq 0$.

Let $e_n = x_n - \alpha = u_n - \alpha = v_n - \alpha$ when $n \to \infty$, and suppose, from the Taylor formula, we have
\[ f(u_{n+1}) = f(\alpha) + f'(\alpha)(u_{n+1} - \alpha) + \frac{f''(\alpha)}{2}(u_{n+1} - \alpha)^2 + o(u_n - \alpha)^3 \]
\[ = f'(\alpha)(u_{n+1} - \alpha) + \frac{f''(\alpha)}{f'(\alpha)}((u_n - \alpha)^2 + o(u_n - \alpha)^3) \]
\[ = f'(\alpha)[A \varepsilon_n^p + A_1 \varepsilon_{n+1}^q + A_2 \varepsilon_{n+2}^{p+2} + \ldots + \left(A_p + \frac{C}{2}\right) \varepsilon_n^{p+2} + O\left(\varepsilon_{n+1}^{2p+1}\right)] \]
\[ (3) \]

And
\[ f'(v_{n+1}) = f'(\alpha) + f''(\alpha)(v_{n+1} - \alpha) + o(u_n - \alpha)^2 \]
\[ = f'(\alpha)\left[1 + \frac{f''(\alpha)}{f'(\alpha)}(v_n - \alpha)^2 + o(v_n - \alpha)^2 \right] \]
\[ = f'(\alpha)(1 + CBE_n^q + O(e_{n+1}^q)) \]
\[ (4) \]

Note that
\[ \frac{1}{1 + x} = 1 - x + x^2 - x^3 + \ldots = 1 - x + o(x^2) \text{ when } |x| < 1 \]

From the convergence property of $\{v_{n+1}\}$ we know that when is sufficiently large,
\[ \varepsilon = |CBE_n^q + o(e_{n+1}^q)| < 1 \]

Substituting (3), (4) into (2), we have
\[ x_{n+1} = \alpha + A \varepsilon_n^p + A_1 \varepsilon_{n+1}^{p+1} + A_2 \varepsilon_{n+2}^{p+2} + \ldots + A_p \varepsilon_n^{2p} + O\left(\varepsilon_{n+1}^{2p+1}\right) \]
\[ - \left[A \varepsilon_n^p + A_1 \varepsilon_{n+1}^{p+1} + A_2 \varepsilon_{n+2}^{p+2} + \ldots + \left(A_p + \frac{C}{2}\right) \varepsilon_n^{p+2} + O\left(e_{n+1}^{p+1}\right) \right] \left(1 - \varepsilon + O\left(e^2\right) \right) \]
\[ = \alpha + ACBe_n^p \varepsilon_n^p + O\left(\varepsilon_{n+1}^{p+1}\right) \]
\[ = \alpha + ACBe_n^{p+q} + O\left(e_{n+1}^{p+q+1}\right), \text{ when } n \to \infty \]

Notice that $p > q$, hence $p + q < 2p$, therefore,
\[ e_{n+1} = ACBe_n^{p+q} + O\left(e_{n+1}^{p+q+1}\right), \quad ABC \neq 0 \]

This proves that iterative method defined by (2) has $p + q$ order convergence.
3. The Iterative Method for Nonlinear Equations

The construction of our Newton-type iterative method for solving nonlinear equations is based on following theorem.

Theorem 1. If there is a simple root \( \alpha \) of the nonlinear equation \( f(x) = 0 \), where \( f : D \subset \mathbb{R} \rightarrow \mathbb{R} \) is a scalar function on an open interval \( D \), and all-order derivatives of \( f(x) \) exist on \( D \), then we can obtain the iterative formula from (2) with any integer order of convergence.

Proof. From the introduction section, we have introduced the iterative methods with order of convergence 2, 3, 4.

For the integer \( n \left( n \geq 5 \right) \) we suppose above theorem is correct, namely, there exist the iterative formula with \( n \)th order convergence which is constructed by two iterative methods with \( p \) and \( q \) order convergence. Here, \( p + q - n, \quad p > q \geq 2p \)

so \( p + 1 + q = n + 1, \quad p + 1 > q \geq 2 \). For the integer \( n + 1 \), the iterative formula with \( n + 1 \) order convergence can be realized by two iterative methods with \( p + 1 \) and \( q \) order convergence, where iterative methods with \( p + 1 \) and \( q \) order convergence exist.

Therefore, we can apply mathematical induction method to prove this theorem.

Remark 1. In our iterative method, we firstly use existed iterative method to calculate \( u(n + 1) \) and \( v(n + 1) \), then compute \( x(n + 1) \) by (2). Hence it is a three-step iterative method. If \( u(x) \) and \( v(x) \) are convergent and their orders are \( p \) and \( q \) respectively, then our method is convergent and its order is \( p + q \).

Remark 2. The iterative method with \( n \)th order convergence \( (n \) is an integer\) is not unique because two iterative methods with \( p \) and \( q \) order convergence can be chosen by various formulas as long as \( p + q = n \). So we think this iterative formula provides us a class of methods for solving non-linear equations.

4. The Iterative Method Extension for Systems of Nonlinear Equations

Considering the problem of finding a real root of \( f : D \subset \mathbb{R}^n \rightarrow \mathbb{R}^n \), that is, a real solution \( x^* \), nonlinear equations system \( F(x) = 0 \), of \( n \) equations with \( n \) variables. We have following Newton’s iterative method with second-order convergence

\[
x_{n+1} = x_n - \left[ F'(x_n) \right]^{-1} F(x_n)
\]

Although our iterative formula (2) is constructed to find a single root of a nonlinear equation, we can easily extend it to systems of nonlinear equations, We give corresponding theorem as follows.

Theorem 2. Let \( f : D \subset \mathbb{R}^n \rightarrow \mathbb{R}^n \) is \( k \)-time Fréchet differentiable in a convex set \( D \) containing the root of \( x^* \) of \( F(x) = 0 \), if there exist two iterative methods \( u(x) \) and \( v(x) \) whose convergence orders are \( p \) and \( q \) respectively \((p, q \text{ are integers and } p > q\) and we denote their results of the \( n + 1 \)th iteration are \( u_{n+1}, v_{n+1} \), then the following iterative formula has order of convergence \( p + q \).
\[ x_{n+1} = u_n - \frac{f(u_{n+1})}{f'(x_{n+1})}, \quad (5) \]

The proof is omitted here since it is similar to that of nonlinear equation.

**Remark 3.** For solving systems of nonlinear equations, we know Newton’s method has order of convergence two, and there have been other iterative methods with third-order and fourth-order convergence in [14,15]. Therefore, we can use (5) to construct the iterative formula with any integer order of convergence by mathematical induction.

### 5. Numerical Results

#### 5.1. For Nonlinear Equations

We apply our method to construct the iterative formula with the fifth order convergence and the eighth order convergence based on existing iterative formulas. We can use similar method to establish the iteration with higher order convergence, the construction is omitted here.

(1) Fifth-order iterative formula

It is realized by Newton’s method and the third-order method in [2]

\[ x_{n+1} = x_n - \frac{f'(x_n) + f'(x_n - f(x_n)/f'(x_n))}{f''(x_n)}, \]

\[ x_{n+1} = u_{n+1} - \frac{f(u_{n+1})}{f'(x_n - f(x_n)/f'(x_n))}, \]

(2) Eighth-order iterative formula

It is constructed by the third-order method and above fifth-order method

\[ v_{n+1} = u_{n+1} - \frac{f(u_{n+1})}{f'(x_n - f(x_n)/f'(x_n))}, \]

\[ x_{n+1} = v_{n+1} - \frac{f(v_{n+1})}{f'(u_{n+1})}. \]

We do the computation with several numerical examples to test our iterative method. We firstly employ our fifth-order method and eighth-order method to solve some nonlinear equations and compared with Newton’s method. All computations were done by Matlab 7.4. The computation results are shown in table 1.

We find these results are consistent to the iterative property, namely, the iterative number of the eight-order method is less than that of the fifth-order method and the Newton’s method.

Table 1: comparison various method required such that \( |x_{n+1} - x_n| \leq 10^{-14} \)

10579 www.ijariie.com 147
<table>
<thead>
<tr>
<th>function</th>
<th>root</th>
<th>Initial valued</th>
<th>Newton method</th>
<th>Fifth-order method</th>
<th>Eighth-order method</th>
</tr>
</thead>
<tbody>
<tr>
<td>$x^2 + 4x^2 - 10$</td>
<td>$x_0 = 3$</td>
<td></td>
<td>$N=7$</td>
<td>$t=0.000118$</td>
<td>$N=4$</td>
</tr>
<tr>
<td></td>
<td>$x_0 = 10$</td>
<td></td>
<td>$N=10$</td>
<td>$t=0.000122$</td>
<td>$N=5$</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>$N=9$</td>
<td>$t=0.003869$</td>
<td>$N=5$</td>
</tr>
<tr>
<td>$\sin(x)^2 - x^2 - 1$</td>
<td>$x_0 = 10$</td>
<td></td>
<td>$N=9$</td>
<td>$t=0.000110$</td>
<td>$N=4$</td>
</tr>
<tr>
<td>$x^2 - e^x - 3x + 2$</td>
<td>$x_0 = 15$</td>
<td></td>
<td>$N=7$</td>
<td>$t=0.002818$</td>
<td>$N=4$</td>
</tr>
<tr>
<td>$xe^x - \sin(x)^2 + 3\cos(x) + 5$</td>
<td>$x_0 = 3$</td>
<td></td>
<td>$N=14$</td>
<td>$t=0.000233$</td>
<td>$N=4$</td>
</tr>
<tr>
<td>$e^{x^2 + 7x - 30} - 1$</td>
<td>$x_0 = 10$</td>
<td></td>
<td>$N=6$</td>
<td>$t=0.003563$</td>
<td>$N=3$</td>
</tr>
<tr>
<td>$\ln(x) + \sqrt{x} - 5$</td>
<td>$x_0 = -5$</td>
<td></td>
<td>$N=31$</td>
<td>$t=0.000163$</td>
<td>$N=15$</td>
</tr>
<tr>
<td></td>
<td>$x_0 = 4$</td>
<td></td>
<td>$N=20$</td>
<td>$t=0.002691$</td>
<td>$N=10$</td>
</tr>
<tr>
<td></td>
<td>$x_0 = 5$</td>
<td></td>
<td>$N=36$</td>
<td>$t=0.0002042$</td>
<td>$N=17$</td>
</tr>
<tr>
<td>$\sin(x)xe^x - 2x - 5$</td>
<td>$x_0 = 20$</td>
<td></td>
<td>$N=36$</td>
<td>$t=0.000159$</td>
<td>$N=4$</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>$N=5$</td>
<td>$t=0.000143$</td>
<td>$N=3$</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>$N=5$</td>
<td>$t=0.0000116$</td>
<td>$N=4$</td>
</tr>
<tr>
<td>$\sqrt{x} - \frac{1}{x} - 3$</td>
<td>$x_0 = 15$</td>
<td></td>
<td>$N=6$</td>
<td>$t=0.0000106$</td>
<td>$N=4$</td>
</tr>
<tr>
<td></td>
<td>$x_0 = -1$</td>
<td></td>
<td>$N=7$</td>
<td>$t=0.0003531$</td>
<td>$N=4$</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>$N=7$</td>
<td>$t=0.0000151$</td>
<td>$N=4$</td>
</tr>
<tr>
<td>$x^3 - 10$</td>
<td>$x_0 = 0$</td>
<td></td>
<td>$N=6$</td>
<td>$t=0.0002525$</td>
<td>$N=5$</td>
</tr>
<tr>
<td></td>
<td>$x_0 = 20$</td>
<td></td>
<td>$N=9$</td>
<td>$t=0.0000120$</td>
<td>$N=5$</td>
</tr>
<tr>
<td></td>
<td>$x_0 = 25$</td>
<td></td>
<td>$N=9$</td>
<td>$t=0.002114$</td>
<td>$N=3$</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>$N=3$</td>
<td>$t=0.000393$</td>
<td>$N=12$</td>
</tr>
</tbody>
</table>
5.2 For system of nonlinear equations

R[14] give the third-order iterative formula for the system of nonlinear equations

\[ x_{n+1} = x_n - F'(x_n)^{-1} F(x_n), \]
\[ x_{n+1} = x_n - F'(x_n)^{-1} \left( F(x_n) + F'(x_n)x_{n+1} \right). \]

Based on this result and Neoton’s method, we construct the fifed-order iterative formula

\[ x_{n+1}^* = x_n - F'(x_n)^{-1} F(x_n), \]
\[ x_{n+1}^* = x_n - F'(x_n)^{-1} \left( F(x_n) + F'(x_n)x_{n+1} \right), \]
\[ x_{n+1} = x_{n+1}^* - F'(x_{n+1})^{-1} F(x_{n+1}). \]

We also do the computation to solve two examples cited in [14], the experimental result is shown in table 2 and 3. We find that for the some accuracy requirement, the iterative number by our methods is less then that by [14].

Example 1. \[ \begin{cases} x_1^2 + x_2^2 + x_3^2 = 1, \\ 2x_1^2 + x_2^2 - 4x_3^2 = 0, \\ 3x_1^2 - 4x_2^2 + x_3^2 = 0, \end{cases} \]

**Initial value** \( x_0 = (0.5, 0.5, 0.5)^T \)

Table 2 the result by our iterative method

<table>
<thead>
<tr>
<th>n</th>
<th>( x_1 )</th>
<th>( x_2 )</th>
<th>( x_3 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.696448843864469</td>
<td>0.628319947168216</td>
<td>0.342567396978022</td>
</tr>
<tr>
<td>2</td>
<td>0.698288609971492</td>
<td>0.628524297960214</td>
<td>0.342564189689569</td>
</tr>
<tr>
<td>3</td>
<td>0.698288609971514</td>
<td>0.628524297960214</td>
<td>0.342564189689569</td>
</tr>
</tbody>
</table>

Example 2. \[ \begin{cases} x_1^2 + 3\log x_1 - x_2^2 = 0, \\ 2x_1^2 - x_1x_2 - 5x_1 = 0, \end{cases} \]

**Initial value** \( x_0 = (3, 4, 2.2)^T \)

Table 3 the result by our iterative method

<table>
<thead>
<tr>
<th>n</th>
<th>( x_1 )</th>
<th>( x_2 )</th>
<th>( x_3 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.696448843864469</td>
<td>0.628319947168216</td>
<td>0.342567396978022</td>
</tr>
<tr>
<td>2</td>
<td>0.698288609971492</td>
<td>0.628524297960214</td>
<td>0.342564189689569</td>
</tr>
<tr>
<td>3</td>
<td>0.698288609971514</td>
<td>0.628524297960214</td>
<td>0.342564189689569</td>
</tr>
<tr>
<td>n</td>
<td>( x_1 )</td>
<td>( x_2 )</td>
<td></td>
</tr>
<tr>
<td>---</td>
<td>---</td>
<td>---</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>1.488028273809524</td>
<td>0.755985863095238</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1.488033871712585</td>
<td>0.755983064143708</td>
<td></td>
</tr>
</tbody>
</table>

### 6. Conclusions

In this paper, we introduce a new iterative method with any integer order of convergence to solve nonlinear equations. The construction formula is simple and it can be applied for systems of nonlinear equations. Numerical examples are shown that our new iterative method is more accurate and effective.

### References