ISSN 1749-3889 (print), 1749-3897 (online)
International Journal of Nonlinear Science
Vol.6(2008) No.1,pp.29-33

A Family of Predictor-Corrector Methods Based on Weight Combination of Quadratures for Solving Nonlinear Equations

Ali R. Soheili *, S.A. Ahmadian, J. Naghipoor
Department of Mathematics, University of Sistan & Baluchestan, Zahedan, Iran
(Received 21 July 2007, accepted 18 February 2008)

Abstract: In this paper, we propose and analyze some new predictor-corrector methods for solving nonlinear equations using the weight combination of mid-point, Trapezoidal and Simpson quadrature formulas. We prove that these new methods are better than the *newton* method. Several examples are given to illustrate the efficiency of the proposed methods.

Keywords: predictor-corrector methods; convergence; quadrature formulas; nonlinear equations; numerical examples

1 Introduction

Finding the roots of non-linear equations are common yet important problem in science and engineering. Analytical methods for solving such equations are difficult or almost non-existent. Therefore it is only possible to obtain approximate solutions by numerical techniques based on iteration procedures [1,5,6]. It is well known that the quadrature formulas[1], play an important and significant rule in the evaluation of the integrals. It has been shown [2] that these quadrature formulas can be used to develop some iterative methods for solving nonlinear equations.we suggest and analyze some new-iterative methods by using the weight combination of the midpoint, Trapezoidal and Simpson quadrature formulas. This method is an implicit-type method. To implement this, we can use the *Newton* and the *Halley* methods and some newly developed method by Noor[2,3,4], as predictor method and then use this new method as a corrector method .A comparison between these new methods with that of *Newton* method is given. Several examples are given to illustrate the efficiency and advantage of these two-step methods.

2 Iterative methods

Suppose that r be the simple zero of a sufficiently differentiable function and consider the numerical solution of equation $f(x) = 0$, then

$$f(x) = f(x_n) + \int_{x_n}^{x} f'(t)dt. \tag{1}$$

If we approximate $\int_{x_n}^{x} f'(t)dt$ with average of midpoint and Simpson quadrature formulas then we have

$$\int_{x_n}^{x} f'(t)dt = \frac{x - x_n}{2} f'(\frac{x_n + x}{2}) + \frac{x - x_n}{12}[f'(x_n) + 4f'(\frac{x_n + x}{2}) + f'(x)]. \tag{2}$$

From (2.1) and (2.2), we have

$$f(x) = f(x_n) + \frac{x - x_n}{12}[f'(x_n) + 10f'(\frac{x_n + x}{2}) + f'(x)].$$

*Corresponding author. *E-mail address*: soheili@math.usb.ac.ir

Copyright©World Academic Press, World Academic Union
IJNS.2008.08.15/159

Since $f(x) = 0$ then

$$x = x_n - \frac{12f(x_n)}{f'(x_n) + 10f'(\frac{x_n+x}{2}) + f'(x)}.$$

With this fixed point formulation and any iterative method as predictor we will have following implicit iterative method.

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

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

Using the predictor type technique, we suggest the following two-step method which is obtained by combining the *Halley* method.

Algorithm 2 *For a given x_0, compute the approximate solution x_{n+1} by the iterative scheme.*

$$y_n = x_n - \frac{2f(x_n)f'(x_n)}{2(f'(x_n))^2 - f(x_n)f''(x_n)},$$

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

For approximating $\int_{x_n}^{x} f'(t)dt$, if we combine Trapezoidal and Simpson quadrature formulas with weight factor $\frac{1}{2}$, then we have

$$\int_{x_n}^{x} f'(t)dt = \frac{x-x_n}{4}[f'(x_n) + f'(x)] + \frac{x-x_n}{12}[f'(x_n) + 4f'(\frac{x_n+x}{2}) + f'(x)]. \tag{3}$$

So from (2.1) and (2.3) and $f(x) = 0$, we can obtain

$$x = x_n - \frac{3f(x_n)}{f'(x_n) + f'(\frac{x_n+x}{2}) + f'(x)}.$$

In similar way we will have following algorithm which our predictor is the well-known *Newton* method.

Algorithm 3 *For a given x_0, compute the approximate solution x_{n+1} by the iterative scheme.*

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

$$x_{n+1} = x_n - \frac{3f(x_n)}{f'(x_n) + f'(\frac{x_n+y_n}{2}) + f'(y_n)}.$$

For approximating $\int_{x_n}^{x} f'(t)dt$, if we combine mid-point, Trapezoidal and Simpson quadrature formulas with $\frac{1}{4}$, $\frac{1}{4}$ and $\frac{1}{2}$ weight factors respectively, then we will have

$$\int_{x_n}^{x} f'(t)dt = \frac{x-x_n}{4}f'(\frac{x_n+x}{2}) + \frac{x-x_n}{8}[f'(x_n) + f'(x)] + \frac{x-x_n}{12}[f'(x_n) + 4f'(\frac{x_n+x}{2}) + f'(x)]. \tag{4}$$

Since $f(x) = 0$, from (2.1) and (2.4) we obtain following fixed point formulation.

$$x = x_n - \frac{24f(x_n)}{5f'(x_n) + 14f'(\frac{x_n+x}{2}) + 5f'(x)}.$$

Same as algorithm 2. with selecting the *Halley* method as a predictor we will have following algorithm

Algorithm 4 *For a given x_0, compute the approximate solution x_{n+1} by the iterative scheme.*

$$y_n = x_n - \frac{2f(x_n)f'(x_n)}{2(f'(x_n))^2 - f(x_n)f''(x_n)},$$

$$x_{n+1} = x_n - \frac{24f(x_n)}{5f'(x_n) + 14f'(\frac{x_n+y_n}{2}) + 5f'(y_n)}.$$

3 Convergence analysis

In this section, we consider the convergence of Algorithm 3. In similar way, one can prove the convergence of other two step algorithms.

Theorem 5 *Let $r \in I$ be a sample zero of sufficiently differentiable function $f : I \subseteq R \to R$ for an open interval I. If x_0 is sufficiently close to r, then the two step method defined by Algorithm 3 has quadratic convergence and it's asymptotic convergence is $\frac{f^{(2)}(r)}{6f'(r)}$.*

Proof. Consider to

$$y_n = x_n - \frac{f(x_n)}{f'(x_n)}, \tag{5}$$

$$x_{n+1} = x_n - \frac{3f(x_n)}{f'(x_n) + f'(\frac{x_n+y_n}{2}) + f'(y_n)}. \tag{6}$$

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

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

then

$$f(x_n) = f'(r)[e_n + c_2e_n^2 + c_3e_n^3 + c_4e_n^4 + ...], \tag{7}$$

and

$$f'(x_n) = f'(r)[1 + 2c_2e_n + 3c_3e_n^2 + 4c_4e_n^3 + 5c_5e_n^4 + ...], \tag{8}$$

where $c_k = \frac{1}{k!}\frac{f^{(k)}(r)}{f'(r)}$, $k = 1, 2, 3, ...$ and $e_n = x_n - r$.
Now, from (3.7) and (3.8), we have

$$\frac{f(x_n)}{f'(x_n)} = e_n - c_2e_n^2 + 2(c_2^2 - c_3)e_n^3 + (-7c_2c_3 + 4c_2^3 + 3c_4)e_n^4 + ..., \tag{9}$$

From (3.5) and (3.9), we get

$$y_n = r + c_2e_n^2 + 2(c_3 - c_2^2)e_n^3 + (-7c_2c_3 + 4c_2^3 + 3c_4)e_n^4 + ... \tag{10}$$

From (3.10), we get

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

and

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

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

$$\begin{aligned} f'(\frac{x_n + y_n}{2}) &= f'(r)[1 + 2c_2(\frac{x_n + y_n}{2} - r) + 3c_3(\frac{x_n + y_n}{2} - r)^3 + 4c_4(\frac{x_n + y_n}{2} - r)^4 + ...] \\ &= f'(r)[1 + 2c_2e_n + (2c_2^2 + \frac{3}{4}c_3 + \frac{1}{2}c_4)e_n^2 + (4c_2c_3 - 4c_2^3)e_n^3 \\ &\quad +(\frac{-61}{4}c_2^2c_3 + 8c_2^4 + 6c_2c_4)e_n^4 + ...]. \end{aligned}$$

Then

$$f'(x_n)+f'(y_n)+f'(\frac{x_n+y_n}{2})=3f'(r)[1+\frac{4}{3}c_2e_n+\frac{1}{3}(4c_2^2+\frac{15}{4}c_3+\frac{1}{2}c_4)e_n^2$$
$$+\frac{1}{3}(4c_4+8c_2c_3-8c_2^3)e_n^3+\frac{1}{3}(\frac{-97}{4}c_2c_3+5c_5+16c_2^4+12c_2c_4)e_n^4+...].$$

From (3.6), $e_{n+1}=x_{n+1}-r$ and $e_n=x_n-r$

$$e_{n+1}=e_n-\frac{3f(x_n)}{f'(x_n)+f'(\frac{x_n+y_n}{2})+f'(y_n)}.$$

Then we will have

$$e_{n+1}=e_n-[e_n+c_2e_n^2+c_3e_n^3+c_4e_n^4+...][1-(\frac{4}{3}c_2e_n+\frac{1}{3}(4c_2^2+\frac{15}{4}c_3+\frac{1}{2}c_4)e_n^2$$
$$+\frac{1}{3}(4c_4+8c_2c_3-8c_2^2)e_n^3+...)+(\frac{4}{3}c_2e_n+\frac{1}{3}(4c_2^2+\frac{15}{4}c_3+\frac{1}{2}c_4)e_n^2+...)^2+...].$$

Finally

$$e_{n+1}=e_n-(e_n+(c_2-\frac{4}{3}c_2)e_n^2+(-\frac{4}{3}c_2^2-\frac{5}{4}c_3-\frac{1}{6}c_4-\frac{4}{3}c_2^2+c_3)e_n^3+...,$$
$$e_{n+1}=\frac{c_2}{3}e_n^2+(\frac{8c_2^2}{3}+\frac{c_3}{4}+\frac{c_4}{6})e_n^3+...$$
$$\lim_{n\to\infty}\frac{e_{n+1}}{e_n^2}=\frac{c_2}{3}=\frac{f^{(2)}(r)}{6f'(r)}.$$

■

Since asymptotic convergence of *Newton* method is c_2 and from Theorem 5, we result that the convergence rate of Algorithm 3 is better than the *Newton* method.

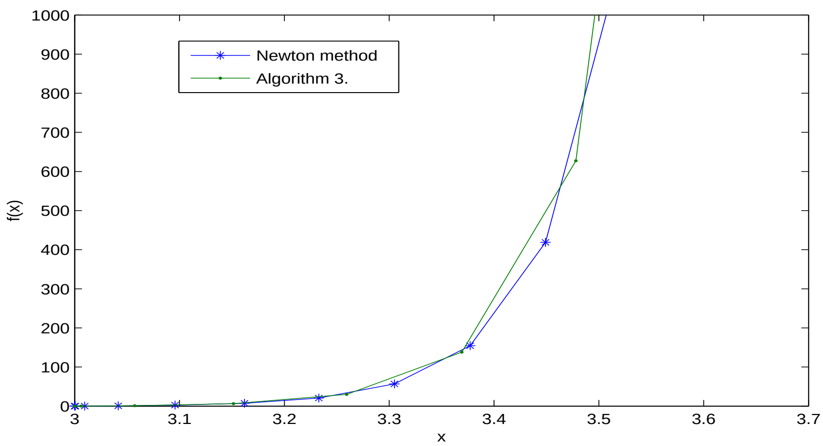

Figure 1: The number of iteration between the Newton method and Algorithm 3. with common starting value $x_0 = 4$

4 Numerical experiments

In all of our examples, the maximum number of iteration is $n = 200$ and our examples are tested with precision $\varepsilon = 1\times 10^{-15}$. The following stopping criteria is used for computer programs:
(i) $|f(x_{n+1})| < \varepsilon$. (ii) $|x_{n+1}-x_n| < \varepsilon$.
Table 1 presents iteration number comparison of algorithms 2, 3 and 4 with the *Newton* method in given precision. In Table 2, the CPU time (per second) of our algorithms and *Newton* method are compared . All numerical results show here, are obtained on a pentium IV processor at 3.00 GHz.
Fig. 1 presents convergence comparison between the *Newton* method and Algorithm 3 for $f(x) = e^{x^2+7x-30}-1$ from starting value $x_0 = 4$.

Table 1: Examples and comparison between algorithms.

Equation	x_0	Newton	Algorithm 2	Algorithm 3	Algorithm 4	x_n
$e^{x^2+7x-30}-1=0$	4	20	9	13	10	3.000000000000000
$x^3-10=0$	1.5	7	4	5	4	2.154434690031884
$x^2-e^x-3x+2=0$	2	6	4	4	4	-1.207647827130919
$\sin^2(x)-x^2+1=0$	-1	7	4	5	4	-1.404491648215341
$x^{10}-1=0$	1.5	10	5	7	6	1.000000000000000
$11x^{11}-1=0$	0.7	8	4	6	4	0.804133097503664
$\sin(\frac{1}{x})-x=0$	2	6	4	4	4	0.897539461280487

5 Conclusion

In Theorem 1. we proved that asymptotic convergence of algorithm 3. is less than the *Newton* method. Then this two step method is better than the *Newton* method . One can prove that other two-step algorithms proposed here are better than the *Newton* method too. In Table 1. we can see accuracy and efficiency of our two-step methods when compared with the *Newton* method.

Table 2: The CPU time (per second) of algorithms.

Equation	Newton	Algorithm 2	Algorithm 3	Algorithm 4
$e^{x^2+7x-30}-1=0$	0.171875	0.078125	0.109375	0.093750
$x^3-10=0$	0.078125	0.031250	0.031250	0.046875
$x^2-e^x-3x+2=0$	0.046875	0.031250	0.031250	0.031250
$\sin^2(x)-x^2+1=0$	0.046875	0.031250	0.046875	0.031250
$x^{10}-1=0$	0.078125	0.031250	0.046875	0.031250
$11x^{11}-1=0$	0.062500	0.031250	0.031250	0.031250
$\sin(\frac{1}{x})-x=0$	0.046875	0.031250	0.031250	0.031250

References

[1] K.E. Atkinson: An introduction to numerical analysis. 2nd ed., John Wiley & Sons, *New York* (1987)

[2] M. Aslam Noor:New iterative methods for nonlinear equations.*J.Appl.Math.Comput.Inpress*

[3] M. Aslam Noor, Khalida Inayat Noor: Predictor-corrector Halley method for nonlinear equations . *J. Appl. Math. Comput*.188: 1587-1591 (2007)

[4] M. Aslam Noor, F. Ahmad: On a predictor-corrector method for solving nonlinear equations. J. Appl. Math. Comput.183 :128-133(2006)

[5] C.T. Kelly: Iterative Methods for Linear and Nonlinear Equations. *SIAM, Philadelphia, PA* (1995)

[6] J.F. Traub: Iterative Methods for Solution of Equations. *Prentice-Hall, Englewood Cliffs, NJ*(1964)