Mathematical and Computational Applications, Vol. 13, No. 2, pp. 123-128, 2008.

A ROOT-FINDING ALGORITHM WITH FIFTH ORDER DERIVATIVES

M. Pakdemirli*, H. Boyacı* and H. A. Yurtsever**

*Department of Mechanical Engineering, Celal Bayar University, 45140 Muradiye, Manisa, Turkey. mpak@bayar.edu.tr, hakan.boyaci@bayar.edu.tr
**Department of Mathematics, Georgetown University, Washington DC, USA

Abstract – Perturbation theory is used to generate a root finding algorithm with fifth order derivatives. The algorithm is called Quintuple-Correction-Term algorithm. The new algorithm is contrasted with the previous Quadruple-Correction-Term and Triple-Correction-Term algorithms in the literature. It is found that adding a fifth correction term in the algorithm does not improve the performance.
Keywords – Root finding algorithms, perturbation theory

1. INTRODUCTION

Perturbation theory is well established and used in search of approximate solutions of algebraic equations, differential equations, integro-differential equations, difference equations etc. In finding the roots of a function, perturbation methods can be used. Many examples of algebraic equations with small parameters were treated in the book by Nayfeh [1]. By expanding the root in a perturbation series, each correction term was calculated in order to find the approximate root. As is usual in perturbation methods, the correction terms were calculated once and no iterations over the corrections were made. In the book by Hinch [2], the perturbation method and iteration method were treated as separate methods. A combination of perturbations with iterations or the so-called "perturbation-iteration method" would be a better choice. In fact, the well known formulas such as Newton Raphson and its second and higher order corrections, namely the Householder's iteration and Schroder family [3] can be derived from perturbations. Depending on the number of terms taken in the perturbation expansion, on the number of terms in the Taylor expansion and the way the resulting equations are separated, different iteration formulas which may or may not belong to the mentioned class of iteration formulas can be generated.

The link between perturbations and root finding algorithms was exploited in a recent work [4]. Root finding formulas consisting of up to third order derivatives were derived in that work. With referral to the number of terms taken in the perturbation expansions, formulas were classified as Single-Correction-Term algorithms, Double-Correction- Term algorithms, Triple-Correction-Term algorithms. Taylor expansions were taken up to third order derivatives in that work. The root finding algorithms of [4] were contrasted with those derived by Abbasbandy [5] using modified Adomian decomposition method. In a recent work, Pakdemirli *et al.* [6] derived three root finding algorithms all containing fourth order derivatives. In deriving the algorithms, they took 4 correction terms in the Taylor series expansion. Taking two, three and four correction terms in the perturbation expansion however yields three completely different algorithms. The main conclusion of that work [6] was that the best algorithms were obtained by taking equal number of correction terms both in the Taylor Series expansions and perturbation expansions.

Motivated by the previous results [4, 6], in this work, a new algorithm is developed with 5 correction terms both in the Taylor series expansions and perturbation expansions. The algorithm is called Quintuple-Correction-Term algorithm and contrasted with the Triple and Quadruple-Correction-Term algorithms presented previously. Two test functions with several initial values are used to determine the performance of the algorithms. It is found that adding a fifth correction term to the algorithm does not improve the performance.

2. QUINTUPLE-CORRECTION-TERM ALGORITHM

In this section, a single point iteration formula consisting fifth order derivative will be derived by using perturbation theory.

To find the roots of the nonlinear equation

$$f(x) = 0 \tag{1}$$

one may assume a perturbation expansion of the below form with five correction terms

$$x = x_0 + \varepsilon x_1 + \varepsilon^2 x_2 + \varepsilon^3 x_3 + \varepsilon^4 x_4 + \varepsilon^5 x_5 + \dots \tag{2}$$

Inserting (2) into (1) and expanding in a Taylor series up to fifth order derivative terms yields

$$\begin{aligned} f(x_0 + \varepsilon x_1 + \varepsilon^2 x_2 + \varepsilon^3 x_3 + \varepsilon^4 x_4 + \varepsilon^5 x_5) \cong\ & f(x_0) + f'(x_0)\,(\varepsilon x_1 + \varepsilon^2 x_2 + \varepsilon^3 x_3 + \varepsilon^4 x_4 + \varepsilon^5 x_5) \\ & + \frac{f''(x_0)}{2!}\,(\varepsilon x_1 + \varepsilon^2 x_2 + \varepsilon^3 x_3 + \varepsilon^4 x_4 + \varepsilon^5 x_5)^2 \\ & + \frac{f'''(x_0)}{3!}\,(\varepsilon x_1 + \varepsilon^2 x_2 + \varepsilon^3 x_3 + \varepsilon^4 x_4 + \varepsilon^5 x_5)^3 \\ & + \frac{f^{iv}(x_0)}{4!}\,(\varepsilon x_1 + \varepsilon^2 x_2 + \varepsilon^3 x_3 + \varepsilon^4 x_4 + \varepsilon^5 x_5)^4 \\ & + \frac{f^{v}(x_0)}{5!}\,(\varepsilon x_1 + \varepsilon^2 x_2 + \varepsilon^3 x_3 + \varepsilon^4 x_4 + \varepsilon^5 x_5)^5 \end{aligned} \tag{3}$$

Re-arranging the terms with respect to similar powers of ε, one has

$$\begin{aligned} f(x_0 + \varepsilon x_1 + \varepsilon^2 x_2 + \varepsilon^3 x_3 + \varepsilon^4 x_4 + \varepsilon^5 x_5) \cong\ & f(x_0) + \varepsilon x_1 f'(x_0) + \varepsilon^2 [\, x_2 f'(x_0) + \frac{1}{2} x_1^2 f''(x_0) \,] \\ & + \varepsilon^3 [\, x_3 f'(x_0) + x_1 x_2 f''(x_0) + \frac{1}{6} x_1^3 f'''(x_0) \,] \\ & + \varepsilon^4 [\, x_4 f'(x_0) + \frac{1}{2} x_2^2 f''(x_0) + x_1 x_3 f''(x_0) \\ & \quad + \frac{1}{2} x_1^2 x_2 f'''(x_0) + \frac{1}{24} x_1^4 f^{iv}(x_0) \,] \\ & + \varepsilon^5 [\, x_5 f'(x_0) + x_1 x_4 f''(x_0) + x_2 x_3 f''(x_0) \\ & \quad + \frac{1}{2} x_1 x_2^2 f'''(x_0) + \frac{1}{2} x_1^2 x_3 f'''(x_0) \\ & \quad + \frac{1}{6} x_1^3 x_2 f^{iv}(x_0) + \frac{1}{120} x_1^5 f^{v}(x_0) \,] = 0 \end{aligned} \tag{4}$$

Equation (4) contains five unknowns x_1, x_2, x_3, x_4 and x_5 which requires five equations to be solved. Hence equation (4) is separated into five blocks as follows

$$f(x_0) + \varepsilon x_1 f'(x_0) = 0 \tag{5}$$

$$\varepsilon^2 [\, x_2 f'(x_0) + \frac{1}{2} x_1^2 f''(x_0) \,] = 0 \tag{6}$$

$$\varepsilon^3 [\, x_3 f'(x_0) + x_1 x_2 f''(x_0) + \frac{1}{6} x_1^3 f'''(x_0) \,] = 0 \tag{7}$$

$$\varepsilon^4 [\, x_4 f'(x_0) + \frac{1}{2} x_2^2 f''(x_0) + x_1 x_3 f''(x_0) + \frac{1}{2} x_1^2 x_2 f'''(x_0) + \frac{1}{24} x_1^4 f^{iv}(x_0) \,] = 0 \tag{8}$$

$$\varepsilon^5 [\, x_5 f'(x_0) + x_1 x_4 f''(x_0) + x_2 x_3 f''(x_0) + \frac{1}{2} x_1 x_2^2 f'''(x_0) + \frac{1}{2} x_1^2 x_3 f'''(x_0) + \frac{1}{6} x_1^3 x_2 f^{iv}(x_0) + \frac{1}{120} x_1^5 f^{v}(x_0) \,] = 0 \tag{9}$$

Solving (5), one obtains

$$\varepsilon x_1 = -\frac{f(x_0)}{f'(x_0)} \tag{10}$$

Equation (6) is solved for $\varepsilon^2 x_2$ with equation (10) substituted when necessary yielding

$$\varepsilon^2 x_2 = -\frac{1}{2}\frac{f^2 f''}{f'^3} \tag{11}$$

Here, for the sake of brevity, $f = f(x_0)$ is taken. Using (10) and (11), equation (7) is solved as

$$\varepsilon^3 x_3 = \frac{f^3}{6f'^5}[\, f''' f' - 3 f''^2 \,] \tag{12}$$

Then one solves equation (8) by using (10), (11), and (12) and obtains

$$\varepsilon^4 x_4 = -\frac{f^4}{24 f'^7}[\, 15 f''^3 - 10 f' f'' f''' + f'^2 f^{iv} \,] \tag{13}$$

Finally equation (9) is solved using (10)-(13) as follows

$$\varepsilon^5 x_5 = \frac{f^5}{120 f'^9}[\, -105 f''^4 + 90 f' f''^2 f''' - 15 f'^2 f'' f^{iv} + 15 f' f''^2 f^{iv} - 10 f'^2 f'''^2 + f'^3 f^{v} \,] \tag{14}$$

The iterative scheme is then constructed by inserting the correction terms to the perturbation expansion

$$x_{n+1} = x_n - \frac{f}{f'} - \frac{f^2 f''}{2f'^3} + \frac{f^3}{6f'^5}[\, f''' f' - 3 f''^2 \,] - \frac{f^4}{24 f'^7}[\, 15 f''^3 - 10 f' f'' f''' + f'^2 f^{iv} \,]$$
$$+ \frac{f^5}{120 f'^9}[\, -105 f''^4 + 90 f' f''^2 f''' - 15 f'^2 f'' f^{iv} + 15 f' f''^2 f^{iv} - 10 f'^2 f'''^2 + f'^3 f^{v} \,] \tag{15}$$

The above derived algorithm is named as Quintuple-Correction-Term algorithm. When the last term is deleted, one has the Quadruple-Correction-Term algorithm presented in [6] and when last two terms are deleted, one has the Triple-Correction-Term algorithm presented in [4]. Numerical comparisons of the three different schemes will be presented next.

Table 1- Roots of $f(x) = e^{-x} - x = 0$ by different methods

	Newton-Raphson	Triple Correction Term [4]	Quadruple Correction Term [6]	Quintuple Correction Term
initial value	0.1	0.1	0.1	0.1
first iteration	0.52252	0.57822	0.56723	0.56751
second iteration	0.56678	0.56714	0.56714	0.56714
third iteration	0.56714			
initial value	0.2	0.2	0.2	0.2
first iteration	0.54020	0.57055	0.56716	0.56725
second iteration	0.56701	0.56714	0.56714	0.56714
third iteration	0.56714			
initial value	0.3	0.3	0.3	0.3
first iteration	0.55322	0.56793	0.56715	0.56716
second iteration	0.56711	0.56714	0.56714	0.56714
third iteration	0.56714			
initial value	0.4	0.4	0.4	0.4
first iteration	0.56184	0.56725	0.56714	0.56714
second iteration	0.56714	0.56714		
initial value	0.5	0.5	0.5	0.5
first iteration	0.56631	0.56715	0.56714	0.56714
second iteration	0.56714	0.56714		
initial value	0.6	0.6	0.6	0.6
first iteration	0.56695	0.56714	0.56714	0.56714
second iteration	0.56714			
initial value	0.7	0.7	0.7	0.7
first iteration	0.56408	0.56715	0.56714	0.56714
second iteration	0.56714	0.56714		
initial value	0.8	0.8	0.8	0.8
first iteration	0.55805	0.56719	0.56714	0.56714
second iteration	0.56713	0.56714		
third iteration	0.56714			
initial value	0.9	0.9	0.9	0.9
first iteration	0.54920	0.56723	0.56714	0.56711
second iteration	0.56708	0.56714		0.56714
third iteration	0.56714			
initial value	1.0	1.0	1.0	1.0
first iteration	0.53788	0.56745	0.56714	0.56703
second iteration	0.56699	0.56714		0.56714
third iteration	0.56714			

Table 2- Roots of f(x) = tan(x) – tanh(x) = 0 by different methods

	Newton-Raphson	Triple Correction Term [4]	Quadruple Correction Term [6]	Quintuple Correction Term
initial value	3.5	3.5	3.5	3.5
first iteration	4.04861	3.93431	3.93285	3.91351
second iteration	3.94263	3.92660	3.92660	3.92660
third iteration	3.92686			
fourth iteration	3.92660			
initial value	3.6	3.6	3.6	3.6
first iteration	4.00712	3.92467	3.92928	3.91651
second iteration	3.93342	3.92660	3.92660	3.92660
third iteration	3.92665			
fourth iteration	3.92660			
initial value	3.7	3.7	3.7	3.7
first iteration	3.96951	3.92395	3.92715	3.92331
second iteration	3.92850	3.92660	3.92660	3.92660
third iteration	3.92660			
initial value	3.8	3.8	3.8	3.8
first iteration	3.94122	3.92578	3.92663	3.92630
second iteration	3.92682	3.92660	3.92660	3.92660
third iteration	3.92660			
initial value	3.9	3.9	3.9	3.9
first iteration	3.92730	3.92659	3.92660	3.92660
second iteration	3.92660	3.92660		
initial value	4.0	4.0	4.0	4.0
first iteration	3.93225	3.92683	3.92660	3.92664
second iteration	3.92663	3.92660		3.92660
third iteration	3.92660			
initial value	4.1	4.1	4.1	4.1
first iteration	3.95982	3.92701	3.92659	3.93085
second iteration	3.92773	3.92660	3.92660	3.92660
third iteration	3.92660			
initial value	4.2	4.2	4.2	4.2
first iteration	4.01291	3.85653	3.92740	3.97811
second iteration	3.93446	3.92642	3.92660	3.92660
third iteration	3.92666	3.92660		
fourth iteration	3.92660			
initial value	4.3	4.3	4.3	4.3
first iteration	4.09336	2.35043	3.93491	4.21394
second iteration	3.95723	4.37515	3.92660	3.99436
third iteration	3.92756	-3.92325		3.92663
fourth iteration	3.92660	-3.92532		3.92660

3. COMPARISONS AND CONCLUDING REMARKS

Numerical comparisons of Quintuple, Quadruple and Triple-Correction-Term algorithms as well as Newton-Raphson algorithm are given in this section. In Table1, the iteration numbers required to find the root of $f(x) = e^{-x} - x = 0$ is shown. Several initial values are taken and convergence to the root is displayed. Compared to the Newton-Raphson method (Single-Correction-Term Algorithm), all higher order algorithms require less iterations. Although Quadruple and Quintuple-Correction-Term algorithms are better than the Triple-Correction-Term algorithm regarding convergence to the root, it seems that Quadruple-Correction-Term algorithm performs slightly better than the Quintuple-Correction-Term algorithm. This may be due to the excess numerical calculations which may introduce error accumulation in the Quintuple-Correction-Term algorithm. In Table 2, results for a root of $f(x) = \tan(x) - \tanh(x) = 0$ are displayed. Similar conclusions can be withdrawn from the table such that the best algorithm is Quadruple, and the second best is the Quintuple-Correction-Term algorithm. The main conclusion of this study is that, there is no advantage in calculating the fifth correction term regarding convergence to the root. Quadruple-Correction-Term algorithm is better than the others.

REFERENCES

1. A. H. Nayfeh, *Introduction to Perturbation Techniques*, John Wiley and Sons, New York, 1981.
2. E. J. Hinch, *Perturbation Methods*, Cambridge University Press, New York, 1991.
3. A. S. Householder, *The Numerical Treatment of a Single Nonlinear Equation*, McGraw-Hill, New York, 1970.
4. M. Pakdemirli, and H. Boyacı, Generation of root finding algorithms via perturbation theory and some formulas, *Applied Mathematics and Computation* **184**, 783-788, 2007.
5. S. Abbasbandy, Improving Newton-Raphson method for nonlinear equations by modified Adomian decomposition method, *Applied Mathematics and Computation* **145**, 887-893, 2003.
6. M. Pakdemirli, H. Boyacı, and H. A. Yurtsever, Perturbative derivation and comparison of root-finding algorithms with fourth order derivatives, *Mathematical and Computational Applications* **12**, 117-124, 2007.