Analysis Of Perturbance Coefficient-Based Seeded Secant Iteration Method

Ozuomba Simeon
Department of Electrical/Electronic and Computer Engineering
University of Uyo
Akwa Ibom State, Nigeria
simeonoz@yahoo.com

Abstract— **In this paper, analysis of perturbance coefficient-based seeded secant iteration method was presented. Unlike the classical secant method which requires two initial guess values for finding the root of the function, the seeded secant method requires only one initial guess value to perform the iteration. Available studies have shown the use of one initial arbitrary guess value, x_0 and a perturbance coefficient, $\delta x = 0.01$. The study in this paper seeks to determine the range of values for x_0 and δx that will yield best results in terms of convergence cycle. Five functions from different published works were selected, the perturbance coefficient-based seeded secant iteration method was simulation in Matlab software for finding the root of each of the five functions for different values of x_0 and δx. The results shows that a perturbance coefficient value of $\delta x \leq 0.000001$ gives best convergence cycle results. Also, the initial guess root value should be within $\pm$ 20% above or below the expected actual root of the function.**

Keywords— Iteration Method, Secant Method, Seeded Secant Iteration, Classical Secant Method, Perturbance coefficient-Based Seeded Secant

I. INTRODUCTION

Normally, the classical secant numerical iteration method requires two initial (guess root) values to be selected for the iteration purpose and the two selected values are expected to be close to one another [1,2,3,4,5]. The question of how close the two initial values should be is not yet answered. On the other hand, some modified versions of the secant method (referred in this paper as seeded secant) have opted for one initial value (the seed). There are two versions of such seeded secant method [6,7,8,9,10]. In the first version, x_0 is the initial guess value and the second value for the iteration is obtained by adding $f(x_0)$ to x_0. The first version of seeded secant is referred in this paper as the residue-based seeded secant. This is because $f(x_0)$ is the residue (error) obtained when the root is assumed to be x_0. If x_0 is the root of $f(x)$ then $f(x_0)$ will be zero (0).

In the second version of the seeded secant, x_0 is the initial guess value and the second value for the iteration is obtained by adding to x_0 a value $\boldsymbol{\delta}$ computed using a perturbance coefficient denoted as $\boldsymbol{\delta}_x$, where $\boldsymbol{\delta} = (\boldsymbol{\delta}_x)x_0$. In the published works, the perturbance coefficient value of $\boldsymbol{\delta}_x$, = 0.01 was used to run the secant iteration. The second version of the seeded secant is referred in this paper as perturbance coefficient-based seeded secant. This is because $\boldsymbol{\delta}_x$, is the perturbance coefficient (small fraction) that is used to compute the perturbance value , $\boldsymbol{\delta}$ which is added to the assumed root, x_0 to obtain the second initial root as $x_0 + \boldsymbol{\delta}$ **or** $x_0 + (\boldsymbol{\delta}_x)x_0$.

In any case, the convergence of the seeded secants is faster when the initial guess value is close to the actual root of the equation. Again, how close the initial guess should be is not yet answered. In this paper, the focus in on the analysis of the perturbance coefficient-based seeded secant method to determine the appropriate range of value for the small perturbance coefficient relative to the guess seed value and also the appropriate range of values for the seed (the single initial guess value) relative to the actual root of the equation.

II. THEORETICAL BACKGROUND ON SECANT METHOD

A. The Classical Secant Method

In the classical secant method, it is generally assumed that the function f(x) has a root x in the range x_L and x_U and it is differentiable in that range x_L and x_U [11,12,13,14]. The algorithm for the classical secant method is as follows:

Step 1:
- **Step 1.1: Input:** Initial values for x_L and x_U
- **Step 1.2: Input:** Desired Accuracy, ϵ
- **Step 1.3: Input:** Maximum Number of Iterations , *n*

Step 2: For K = 1 To n Step 1 do:

Step 3:
- **Step 3.1: Compute** $f(x_L)$
- **Step 3.2: Compute** $f(x_U)$

Step 4: $\boldsymbol{x} = x_U - f(x_U)\left(\frac{x_U - x_L}{f(x_U) - f(x_L)}\right)$

Step 5:
- **Step 5.1:** If $|\boldsymbol{x} - x_U| < \epsilon$ Then
  - **Step 5.1.1:** Output $\boldsymbol{x}$
  - **Step 5.1.2:** Goto Step 8;
- **Step 5.2: Else**
  - **Step 5.2.1:** $x_L = x_U$
  - **Step 5.2.2:** $x_U = \boldsymbol{x}$
- **Step 5.3: EndIf**

Step 6: Next K

Step 7: Output "Maximum number of iterations exceeded; Method failed to find the root"

Step 8 Stop

B. The Perturbance coefficient-Based Seeded Secant Method

In the perturbance coefficient-based seeded secant method, it is assumed that the function f(x) has a root x near a seed point x_p. Importantly, the function is differentiable in the range x and x_p. Also, the seeded secant uses a small perturbance coefficient value ($\boldsymbol{\delta_x}$) of say 0.01 along with the seed value, x to do the iteration. Then, $\boldsymbol{\delta} = (\boldsymbol{\delta_x})\, x_o = (\mathbf{0.01})\, x_o$. The algorithm for the seeded secant method is as follows:

Step 1:

Step 1.1: Input: Initial value for x_o

Step 1.2: Set value for $\boldsymbol{\delta_x} = 0.01$

Step 1.3: Input: Desired Accuracy, ϵ

Step 1.4: $\boldsymbol{\delta} = (\boldsymbol{\delta_x})\, x_o = (\mathbf{0.01})\, x_o$

Step 1.5: Input: Maximum Number of Iterations, *n*

Step 2: For K = 1 To n Step 1 do:

Step 3:

Step 3.1: Compute $f(x_{k-1})$

Step 3.2: Compute $f(x_{k-1} + \boldsymbol{\delta})$

Step 4: $x_k = x_{k-1} - f(x_{k-1})\left(\frac{\delta}{f(x_{k-1}+\delta) - f(x_{k-1})}\right)$

Step 5:

Step 5.1: If $|x_k - x_{k-1}| < \epsilon$ Then

Step 5.1.1: Output x_k

Step 5.1.2: Goto Step 8;

Step 5.3: EndIf

Step 6: Next K

Step 7: Output "Maximum number of iterations exceeded; Method failed to find the root"

Step 8 Stop

III. METHODOLOGY

In the available published works, a constant perturbance coefficient value of $\delta x = 0.01$ is used for the Perturbance coefficient-Based Seeded Secant iteration. As such, in the published works, δx is defined as follows;

$$\boldsymbol{\delta x} = \frac{1}{10^2} = \frac{1}{100} = \mathbf{0.01} \qquad (1)$$

In this paper, the perturbance coefficient value is defined as follows;

$$\boldsymbol{\delta x} = \frac{1}{10^{W}} = \mathbf{10^{-W}} \qquad (2)$$

Hence, in this study, for a given function a seed value, x_o is selected and the value of w in Eq 2 is varied from 1 to 9 and for each value of w the resultant perturbance coefficient value, **δx** is used in the Perturbance coefficient-Based Seeded Secant iterations. The convergence cycle number, N_w for each **δx** is noted. Hence, the effect of the perturbance coefficient value, **δx** on the convergence cycle for the given function and seed value can be assessed. In this study, the seed value is expressed as a percentage of the actual root as follows:

$$\mathbf{p} = \left(\frac{x_o}{x_{Cn}}\right) \mathbf{100}\ \% \qquad (3)$$

Where x_{Cn} is the value of x at the convergence cycle. Again, similar iterations are conducted for the same function but with different seed values, x_o (that give different P values). Specifically, values of x_o that give p = 70%, 90 %, 110% 200% , 300%, 400% and 500% are used. Furthermore, the same iterations are repeated for different functions. The results obtained enable the generalization of the effect of the perturbance coefficient value, δ_w and the seed value, x_o on the convergence cycle for any function. In all the cases , $\boldsymbol{\delta} = (\boldsymbol{\delta_x})x$.

IV. RESULTS AND DISCUSSIONS

A. Selected Functions

Five different functions were used in the study and they are as follows:

1) $\mathbf{f(x) = x^{3.5} - 80 = 0}$
2) $\mathbf{f(x) = x^6 - x - 1 = 0}$
3) $\mathbf{f(x) = cos(x) - xe^x = 0}$
4) $\mathbf{f(x) = x^{-0.5} - Ln(1 + x^{0.5}) + 0.5Ln(1 - x) = 0}$
5) $\mathbf{f(x) = x^2 - sin(x) - 0.5 = 0}$

B. Determination of the convergence cycle for x_o = 90% of Actual Root and δx of 0.01

Existing works on the Perturbance coefficient-Based Seeded Secant used initial value, x_o that are about 90% to 110% of the actual root (x_{cn}) of the function and a perturbance coefficient (δx) of 0.01. We now consider the first function in the list , that is; $f(x) = x^{3.5} - 80$. In order to select the initial root value that is close enough to the actual root of the function, the graph of the function is plotted. In Figure 1, the graph of the first function $f(x) = x^{3.5} - 80$ is plotted. From Figure 1, the root of the function is about 3.497. An initial root value of 90% of the actual root and a perturbance coefficient value of 0.01 is used. That means, for the first function, $x_o = 0.9(Xcn) = 0.9(3.497) = 3.148$ (as shown in Table 1 and Table 2). Similarly, the expected actual root of all the listed functions and the selected initial root values are shown in Table 1. In each case, x_o/x_{cn} is 90% of the expected actual root, as shown in Table 2 on the column with the heading as $(x_o/x_{cn})100\%$. Also, $\delta x = 0.01$ is used for each of the functions. The desired accuracy, $\epsilon = 1x10^{-13}$. That means, the iteration stops when $f(x) \leq 1x10^{-13}$. The Perturbance coefficient-Based Seeded Secant iteration result for the first function is shown in Table 2 for , $x_o = 0.9(x_{cn}) = 0.9(3.497) = 3.148$, $\delta x = 0.01$ and $\epsilon = 1x10^{-13}$. The results show that the iteration converged in the 8th cycle with the root as $x_8 = 3.497357$.

x	$f(x) = x^{3.5} - 80$
3.470	-2.1689
3.475	-1.77567
3.480	-1.38103
3.485	-0.98496
3.490	-0.58748
3.495	-0.18856
3.500	0.21178
3.505	0.613556
3.510	1.016767
3.515	1.421416
3.520	1.827507
3.525	2.235043
3.530	2.644027
3.535	3.054461

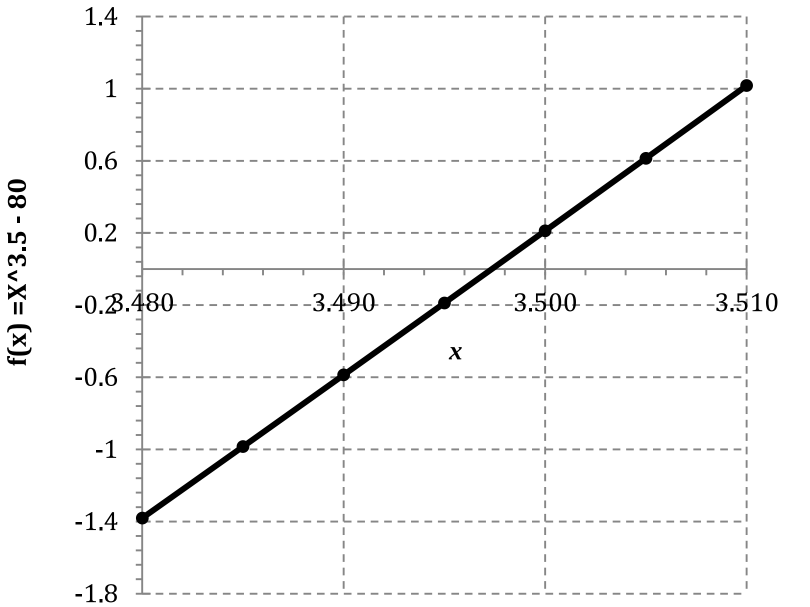

Figure 1 The graph of the function $f(x) = x^{3.5} - 80$

Table 1 The expected actual root of all the listed functions and the selected initial root values

Function	δx	x0	xo +δ	f(xo)	f(xo +δx)	x1
1	0.01	**3.1476215**	**3.1576215**	-24.6727902	-24.0551319	3.5470784
2	0.01	**1.0212517**	**1.0312517**	-0.8867718	-0.8284664	1.1733425
3	0.01	**0.4659816**	**0.4759816**	0.1508030	0.1227065	0.5196547
4	0.01	**0.6253353**	**0.6353353**	0.1910579	0.1640292	0.6960224
5	0.01	**1.0764738**	**1.0864738**	-0.2214944	-0.2045652	1.2073094

Table 2 The convergence cycle (cn) for the five functions when the Perturbance coefficient-Based Seeded Secant iterations is used to determine the root of each of the functions

Function S/N	Actual Root (Xcn)	Xo	(x_o/x_{cn})100%	δx	(δx/Xo)	cn	f(xcn)
1	**3.50**	**3.15**	**90.0**	**0.01**	**3.18E-03**	**8**	**0.00E+00**
2	1.13	1.02	90.0	**0.01**	**9.79E-03**	11	**-8.88E-16**
3	0.52	0.47	90.0	**0.01**	**2.15E-02**	8	**0.00E+00**
4	0.69	0.63	90.0	**0.01**	**1.60E-02**	7	**0.00E+00**
5	1.20	1.08	90.0	**0.01**	**9.29E-03**	8	**0.00E+00**

C. Determination of the convergence cycle for x_o = 90% of Actual Root and δx of 10^{-1}, 10^{-2},…, 10^{-9}

The Perturbance coefficient-Based Seeded Secant iteration simulated for the various functions for x_o = 90% of actual root, δx of 10^{-1}, 10^{-2},…, 10^{-9} (that is for w = 1,2,3,4,…,9.) and $\epsilon = 1x10^{-13}$. The Perturbance coefficient-Based Seeded Secant iteration result for the first function is shown in Table 3 for , $x_o = 0.9(Xcn) = 0.9(3.497) = 3.148$, w = 1 (that is δx =$\mathbf{10^{-1}}$ = 0.1) and $\epsilon = 1x10^{-13}$. The results show that the iteration converged in the 11th cycle which is above the convergence cycle of 8 in the case when w = 2 (that is δx =$\mathbf{10^{-2}}$ = 0.01). The convergence cycle for the other functions for , $x_o = 0.9(Xcn) = 0.9(3.497) = 3.148$, w = 1 (that is δx =$\mathbf{10^{-1}}$ = 0.1) and $\epsilon = 1x10^{-13}$ are shown in Table 4.

The convergence cycle for the five functions for , $x_o = 0.9(Xcn) = 0.9(3.497) = 3.148$, w = 1 , 2,3,…,9 (that is δx =$\mathbf{10^{-1}}$, $\mathbf{10^{-2}}$,…, $\mathbf{10^{-9}}$) and $\epsilon = 1x10^{-13}$ are shown in Table 5 and Figure 2. From the results it can be seen that for the functions studied, the convergence cycle decreases as the perturbance coefficient , δx decreases. However, the convergence cycle remains the same for all δx ≥ $\mathbf{10^{-6}}$. Essentially, the appropriate perturbance coefficient, (δx) is $\mathbf{10^{-6}}$.

Table 3 The Perturbance coefficient-Based Seeded Secant iteration result for the first function, $\mathbf{f(x)} = \mathbf{x^{3.5} - 80} = \mathbf{0}$

Iteration Cycle number (n)	δX	Xk	Xk + δX	f(xk)	f(xk+ δX)	x(k+1)
1	**1.0E-01**	**3.147622**	**3.247622**	**-2.47E+01**	**-18.2725**	**3.533114**
2	1.0E-01	3.533114	3.633114	2.90E+00	11.40642	3.49903
3	1.0E-01	3.49903	3.59903	1.34E-01	8.440108	3.497417
4	1.0E-01	3.497417	3.597417	4.76E-03	8.301406	3.497359
5	1.0E-01	3.497359	3.597359	1.67E-04	8.296476	3.497357
6	1.0E-01	3.497357	3.597357	5.83E-06	8.296303	3.497357
7	1.0E-01	3.497357	3.597357	2.04E-07	8.296297	3.497357
8	1.0E-01	3.497357	3.597357	7.14E-09	8.296297	3.497357
9	1.0E-01	3.497357	3.597357	2.50E-10	8.296297	3.497357
10	1.0E-01	3.497357	3.597357	8.71E-12	8.296297	3.497357
11	1.0E-01	3.497357	3.597357	3.27E-13	8.296297	3.497357

Table 4 The convergence cycle for the five functions for , $x_o = 0.9(Xcn) = 0.9(3.497)$ = 3.148 , w = 1 (that is δx =$\mathbf{10^{-1}}$ = 0.1) and $\epsilon = 1x10^{-13}$

Function	Actual Root	Xo	(Xo/Xn)100%	δx	(δx/Xo)	Cn	f(xcn)
1	**3.497**	**3.148**	**90.0**	**0.10**	**3.18E-02**	**12**	**0.00E+00**
2	1.135	1.021	90.0	**0.10**	**9.79E-02**	20	**8.44E-15**
3	0.518	0.466	90.0	**0.10**	**2.15E-01**	12	**5.00E-15**
4	0.695	0.625	90.0	**0.10**	**1.60E-01**	12	**2.55E-15**
5	1.196	1.076	90.0	**0.10**	**9.29E-02**	12	**0.00E+00**

Table 5 The convergence cycle for the five functions for , $x_o = 0.9(Xcn) = 0.9(3.497)$ = 3.148 , w = 1 , 2,3,…,9 (that is δx =$\mathbf{10^{-1}}$, $\mathbf{10^{-2}}$,…, $\mathbf{10^{-9}}$) and $\epsilon = 1x10^{-13}$

(Xo/Xn)100%	δx	Log(δx)	Conv. Cycle Fn 1	Conv. Cycle Fn 2	Conv. Cycle Fn 3	Conv. Cycle Fn 4	Conv. Cycle Fn 5
90	1.00E-01	-1.0	12	20	12	12	12
90	1.00E-02	-2.0	8	11	8	7	8
90	1.00E-03	-3.0	7	8	6	6	6
90	1.00E-04	-4.0	6	7	6	5	6
90	1.00E-05	-5.0	6	7	5	5	6
90	1.00E-06	-6.0	6	6	5	5	5
90	1.00E-07	-7.0	6	6	5	5	5
90	1.00E-08	-8.0	6	6	5	5	5
90	1.00E-09	-9.0	6	6	5	5	5

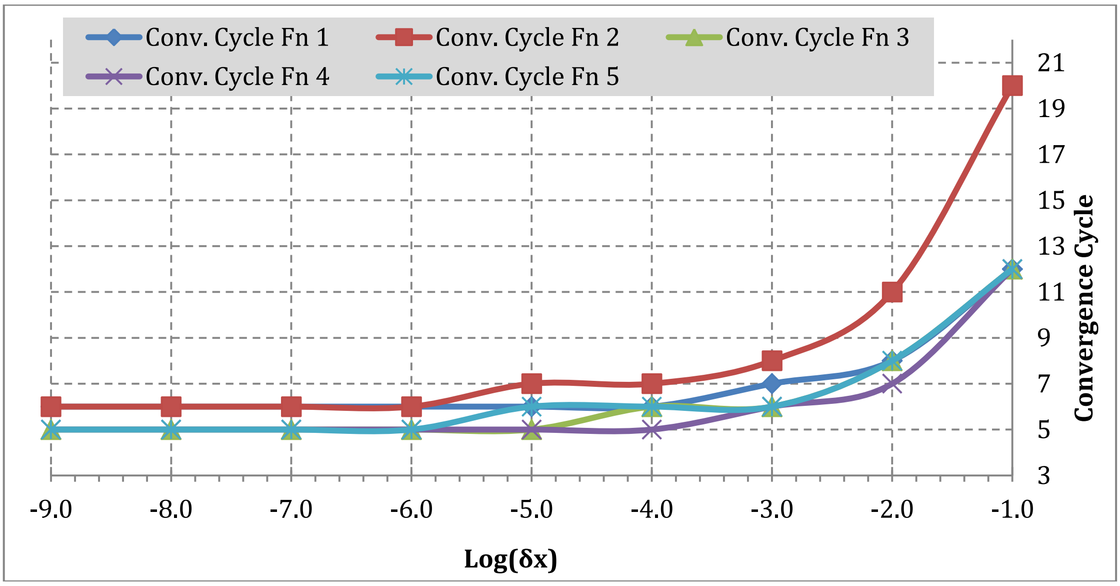

Figure 2 The convergence cycle for the five functions for , $x_o = 0.9(Xcn) = 0.9(3.497) = 3.148$, w = 1 , 2,3,…,9 (that is δx =$\mathbf{10^{-1}}, \mathbf{10^{-2}}, \ldots, \mathbf{10^{-9}}$) and $\epsilon = 1x10^{-13}$

D. Determination of the convergence cycle for $\delta x = 10^{-6}$ and x_o = 70%, 90 %, 110% 200% , 300%, 400% and 500% of the actual root

The Perturbance coefficient-Based Seeded Secant iteration was iterated for the various functions for for $\delta x = 10^{-6}$, x_o = 70%, 90 %, 110% 200% , 300%, 400% and 500% of the actual root and $\epsilon = 1x10^{-13}$. The convergence cycle for all the five functions are shown in Table 5. The results show that the convergence cycle decreases as the initial root , x_o get closer to the actual root . Hence, the convergence cycle when x_o = 99% of Xcn is smaller than the convergence cycle when x_o = 90% of Xcn.

In respect of the results in Table 6 and Figure 3, it can be recommended that, $x_o = x_{cn} \pm 20\%$ of x_{cn} is a good initial root for iteration. Also, in respect of the results in Table 4 and Figure 2, $\delta x = 10^{-6}$ is the good perturbance coefficient value for the iteration.

Table 6 The convergence cycle for all the five functions for for $\delta x = 10^{-6}$, x_o = 70%, 90 %, 110% 200% , 300%, 400% and 500% of the actual root and $\epsilon = 1x10^{-13}$.

δx	(Xo/Xn)100%	CONV. Cycle Fn 1	CONV. Cycle Fn 2	CONV. Cycle Fn 3	CONV. Cycle Fn 4	CONV. Cycle Fn 5
1.00E-06	70	7	12	6	5	6
1.00E-06	80	7	8	5	5	6
1.00E-06	90	6	6	5	5	5
1.00E-06	99	5	5	4	4	4
1.00E-06	101	5	5	4	4	4
1.00E-06	110	6	6	5	5	5
1.00E-06	120	6	7	5	6	6
1.00E-06	130	7	7	6	6	6

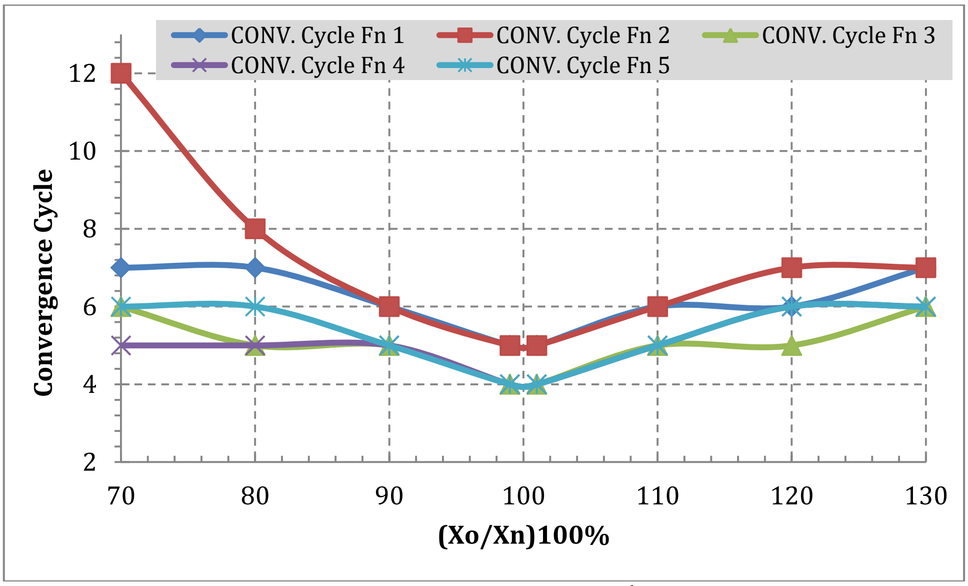

Figure 3 The convergence cycle for all the five functions for for $\delta x = 10^{-6}$, $x_o = 70\%$, 90 %, 110% 200% , 300%, 400% and 500% of the actual root and $\epsilon = 1x10^{-13}$.

V. CONCLUSION

A version of secant numerical iteration method referred in paper as perturbance coefficient-based seeded secant was studied. The study focused on the determination of appropriate initial root guess value and the appropriate value of the perturbance coefficient value for the initial root. Five different functions were used in the study and the results obtained showed that a perturbance coefficient value of 0.000001 gives best convergence cycle results. Also, the initial guess root value should be within $\pm 20\%$ above or below the expected actual root of the function.

REFERENCES

1. Chun, S., & Kwasinski, A. (2011). Analysis of classical root-finding methods applied to digital maximum power point tracking for sustainable photovoltaic energy generation. *IEEE transactions on power electronics*, *26*(12), 3730-3743.
2. Pourjafari, E., & Mojallali, H. (2012). Solving nonlinear equations systems with a new approach based on invasive weed optimization algorithm and clustering. *Swarm and Evolutionary Computation*, *4*, 33-43.
3. Sra, S. (2011, September). Fast projections onto ℓ 1, q-norm balls for grouped feature selection. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases* (pp. 305-317). Springer, Berlin, Heidelberg..
4. Kim, J., & Kwasinski, A. (2014, September). Maximum power point tracking for multiple photovoltaic modules using root-finding methods. In *2014 IEEE Energy Conversion Congress and Exposition (ECCE)* (pp. 9-16). IEEE..
5. Atkinson, K. E. (2008). *An introduction to numerical analysis*. John wiley & sons.
6. Amat, Sergio, and Sonia Busquier. "A modified secant method for semismooth equations." *Applied mathematics letters* 16.6 (2003): 877-881.
7. Babaie-Kafaki, S., Ghanbari, R., & Mahdavi-Amiri, N. (2010). Two new conjugate gradient methods based on modified secant equations. *Journal of computational and applied Mathematics*, *234*(5), 1374-1386..
8. Papakonstantinou, J. M., & Tapia, R. A. (2013). Origin and evolution of the secant method in one dimension. *The American Mathematical Monthly*, *120*(6), 500-517.
9. King, Richard F. "A secant method for multiple roots." *BIT Numerical Mathematics* 17.3 (1977): 321-328.
10. Fernández-Torres, Gustavo. "A Novel Geometric Modification to the Newton-Secant Method to Achieve Convergence of Order and Its Dynamics." *Modelling and Simulation in Engineering* 2015 (2015).
11. Kincaid, D., Kincaid, D. R., & Cheney, E. W. (2009). *Numerical analysis: mathematics of scientific computing* (Vol. 2). American Mathematical Soc..
12. Gentle, J. E. (2009). Solution of Nonlinear Equations and Optimization. In *Computational Statistics* (pp. 241-304). Springer, New York, NY..
13. Tiruneh, Ababu Teklemariam, William N. Ndlela, and Stanley J. Nkambule. "A three point formula for finding roots of equations by the method of least squares." *arXiv preprint arXiv:1302.1966* (2013).
14. Baskar, S., and S. Sivaji Ganesh. "Introduction to Numerical Analysis." *Lecture Notes for MA* 214 (2005).