

# Nonlinear Equation Graphical User Interface Solver Using Excel VBA Programming

Cheong Tau Han<sup>1</sup>, Tay Kim Gaik<sup>2</sup> and Rosmila Abdul Kahar<sup>3</sup>

<sup>1</sup>Faculty of Education, Universiti Teknologi Mara, Malaysia.

<sup>2</sup>Faculty of Electrical & Electronic Engineering, Universiti Tun Hussien Onn, Malaysia.

<sup>3</sup>Department of Mathematics & Statistics, Universiti Tun Hussien Onn, Malaysia.

chgtauhan@gmail.com

**Abstract**— Learning numerical analysis always said to be bored and learners easily get frustrated in the process of getting the solution. Many students had given up in this subject. To overcome the problem, there are a lot of ready-made mathematical software to help students in understanding this subject. However, cost to purchase the license always becomes the concern of students even though with the discounted academic price. Therefore, we developed a Bisection and Newton-Raphson spreadsheet calculator, written using Visual Basic Application (VBA), which is more user friendly graphical user input form in order to help students in mastering the numerical concepts but worried less on the calculation part. Users just need to enter minimum input such as nonlinear function, initial interval or initial guess and derivative function to get a full list of calculation and result. The spreadsheet calculator had been tested among students and feedback were analyzed. We believe this is an ideal version and can easily adapted by users especially who are not familiar with Excel.

**Index Terms**— Excel; Numerical Root Finding; VBA.

## I. INTRODUCTION

The root-finding problem is one of the most important computational problems. It arises in a wide variety of real world applications in physics, chemistry, biosciences, engineering, economic and etc. For example, determination of the charge ( $q$ ) on the capacitor in an electric circuit based on Kirchoff's second law. This will involve implicit nonlinear equation where this will increase the complexity to find the exact solution. In such situations, an approximation of root finding is important to provide a practical solution. Many root finding algorithms had been well developed such as bracketing method, open method and etc. Among these methods, bisection method (bracketing method) and Newton method (open methods) are popular and easy to be used [11].

Many mathematical software had been built in these methods in their package for user to apply. However, due to constraints such as cost and hardware problems, not all users can adapt it. According to the works of [1-9], Excel spreadsheet shows the calculation ability for a wide range of numerical methods. Tay [10] had developed several of spreadsheets in Excel to solve numerical problems, however their works are not development of spreadsheet calculator. Their works are some Excel tips on how to solve nonlinear equations using Excel spreadsheet. Follow her tips, users have to enter the Excel command in Excel spreadsheet layout, not

in a graphical user interface in order to solve the nonlinear equations. Later, they have to manually drag the formula down to copy down the iterations up to the desired accuracy. Their works are interactive teaching and learning tips in Excel on how to solve nonlinear equations and suitable for beginner of learner of nonlinear equations. However, based on students' feedback [2,3,8], the spreadsheet was not user friendly enough. Majority of the students has no experience in using Microsoft Excel in solving numerical problem. They only possessed basic knowledge in Microsoft Excel. The spreadsheet had the limitation where the functions were not displayed and the process to get solution are manually manipulated by dragging from cell to another cell. This becomes very inconvenient for the users who do not know Excel well. Therefore, we improved the limitation of the previous version by rewrite the algorithm using VBA plus add on some graphic into the spreadsheet. Hence, in this work, we improve the limitation of their works in terms of automation and suitable for learners who know how to solve nonlinear equations numerically but need its full solution repetitively and quickly.

Since almost all laptops are equipped with Microsoft Office Excel so users do not need extra cost to purchase a costly licensed software. Besides the nature of Excel which is in tabular form is suitable for numerical calculations. Hence, we choose Excel as a tool to develop the nonlinear equations calculator. To make it fully automated as a software, we need a programming behind it. The programming behind Excel is VBA programming.

## II. BISECTION AND NEWTON-RAPHSON METHOD

In this study, two root finding algorithms, bisection and Newton-Raphson methods, had been integrated into this version of spreadsheet calculator.

### A. Bisection Method

Bisection method is a bracketing method where two initial guesses of the root are required. These guesses must "bracket" or be either side of the root. Bisection is a type of incremental search method in which the interval is always divided into half. If a function changes sign over an interval, the function value at the midpoint is evaluated. According to the intermediate value theorem, the location of the root is fall in

between the midpoint of the subinterval within which the sign changes occur.

The algorithm of the bisection is listed as below.

Inputs:

- $f(x)$  = The given function
- $a_0, b_0$  = The two initial guess such that  $f(a_0)f(b_0) < 0$
- $\epsilon$  = The error tolerance

Output: An approximation of the root of  $f(x) = 0$  in  $[a_0, b_0]$ .

For  $i = 0, 1, 2, \dots$ , do until satisfied the stopping criteria.

Step 1: Compute  $c_i = \frac{a_i + b_i}{2}$ .

Step 2: If  $f(a_i)f(c_i) < 0$ , then choose  $a_{i+1} = a_i, b_{i+1} = c_i$ . Repeat Step 1.

If  $f(b_i)f(c_i) < 0$ , then choose  $a_{i+1} = c_i, b_{i+1} = b_i$ . Repeat Step 1.

If  $|f(c_i)| < \epsilon$ , then stop searching

### B. Newton-Raphson Method

Newton-Raphson is an open method which only requires a starting guess unlike bracket method. Therefore, it may sometimes diverge from the true root as the computation progresses. However, when the open method converges, it usually converges faster than bracket method.

Newton-Raphson method was derived based on the concept of finding the root of tangent line of the curve. This method requires three inputs from users, such as nonlinear function, derivative of the nonlinear function and initial guess. To use this method, users must make sure the second derivative of the function is continuous and first derivative at the root is not equal to zero. This will lead to the equation as stated below.

$$x_{i+1} = x_i - \frac{f(x_i)}{f'(x_i)}$$

The algorithm of the Newton-Raphson method is stated as below.

Inputs:

- $f(x)$  = The given nonlinear function
- $x_0$  = The initial guess
- $\epsilon$  = The error tolerance

Output: An approximation of the root of  $f(x) = 0$ .

For  $i = 0, 1, 2, \dots$ , do until satisfied the stopping criteria.

Step 1: Compute  $f(x_i)$  and  $f'(x_i)$ .

Step 2: Compute  $x_{i+1} = x_i - \frac{f(x_i)}{f'(x_i)}$

Step 3: Test the convergence or stopping criteria.

If  $f(x_i) < \epsilon$ , then stop searching. Else repeat step 2.

### III. NUMERICAL EXAMPLE

In this section, an Aerospace problem [11] will be solved using the two algorithms mentioned above.

#### Question:

The upward velocity of a rocket can be computed by the following formula:

$$v = u \ln\left(\frac{m_0}{m_0 - qt}\right) - gt$$

where  $v$  = upward velocity,  $u$  = the velocity at which fuel is expelled relative to the rocket,  $m_0$  = the initial mass of the rocket at time  $t = 0$ ,  $q$  = the fuel consumption rate and  $g$  = the downward acceleration of gravity (assumed constant  $9.80665 \text{ m/s}^2$ ). If  $u = 2000 \text{ m/s}$ ,  $m_0 = 150,000 \text{ kg}$  and  $q = 2700 \text{ kg/s}$ , compute the time  $t$  at which  $v = 750 \text{ m/s}$ . (Hints:  $t$  is somewhere between 10 and 50s).

#### Solution:

To find the time  $t$  when  $v = 750 \text{ m/s}$ , substitute all the given values into the given equation.

$$v = u \ln\left(\frac{m_0}{m_0 - qt}\right) - gt$$

$$v = 2000 \ln\left(\frac{150000}{150000 - 2700t}\right) - 9.80665t$$

$$v = 2000[\ln(150000) - \ln(150000 - 2700t)] - 9.80665t$$

When  $v = 750$ , this will lead to

$$2000[\ln(150000) - \ln(150000 - 2700t)] - 9.80665t - 750 = 0 \quad (1)$$

Solve equation(1) in order to get the value of  $t$

The solution by Bisection and Newton's methods are shown in Figure (3a) and Figure (3b).

### IV. ROOT FINDING SPREADSHEET CALCULATOR

Figure 1 illustrates the initial layout of the root finding spreadsheet calculator. This spreadsheet calculator able to solve by using two different methods such as bisection and Newton methods.



Figure 1: Initial layout of Root Finding Spreadsheet Calculator

Once the Input button is clicked, a form will pop up as shown in Figure 2a and Figure 2b to capture all required information to solve for the root of a nonlinear equation using particular method. Users need to define the variable used in the function and method to be used in searching the root of the

nonlinear equation. If Bisection method is chosen, users need to define the interval range where the root is located (Figure 2a), else if Newton method is chosen, users need to input an initial guess for the root as well as the derivative function (Figure 2b). Later, user need to decide the number of decimal places, up to nine decimal places, used in the calculation.

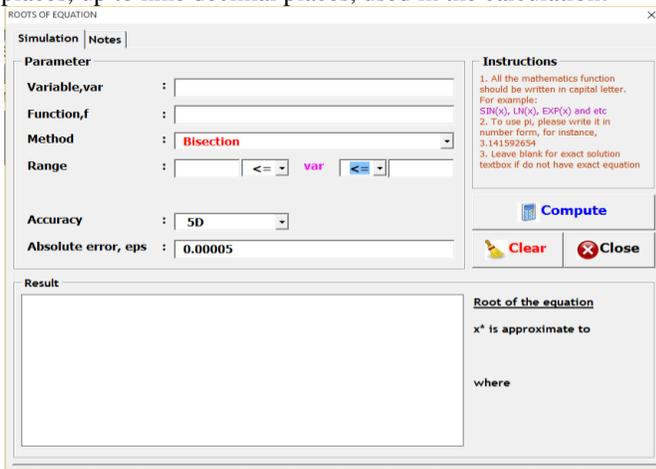


Figure 2a: User interface of bisection method

If bisection method is chosen, the result will be displayed in the Result panel. Users can view the calculation in the list box and the root of equation is printed on the right hand side of the list box with the stopping criteria value.

However, for the process of learning numerical analysis, even solution is important, the concept and the process are much more important. Therefore, authors have programmed to print the calculation of each iteration on the excel spreadsheet with highlighted color for some cell that satisfied the algorithm of bisection.

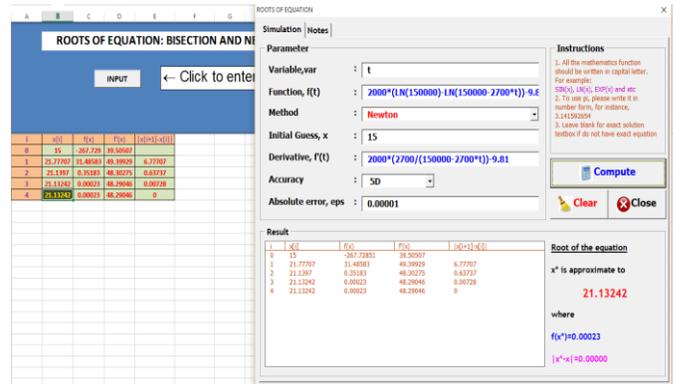


Figure 3b: Result of solving equation (1) using Newton method

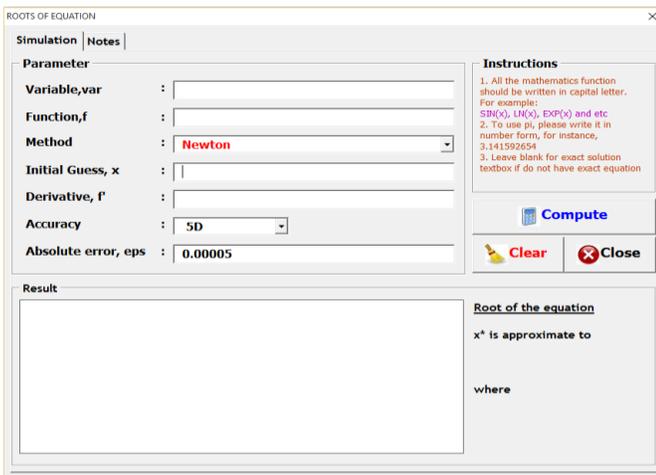


Figure 2b: User interface of Newton's method

After everything has been entered as shown in Figure 3a and Figure 3b, users can click the *Compute* button to get the answer as given in Figure 3a and Figure 3b.

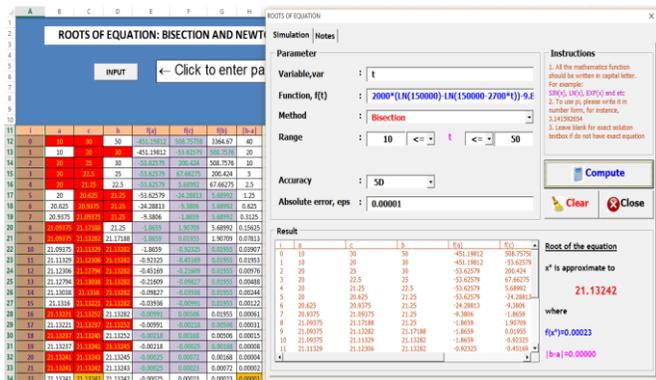


Figure 3a: Result of solving equation (1) using bisection method

If Newton-Raphson method is chosen, the result will be displayed in the Result panel. Users can view the calculation in the list box and the root of equation is printed on the right hand side of the list box with the stopping criteria value. For more colorful view, user always can read from the spreadsheet.

As one of the objectives to develop this spreadsheet calculator is to create a calculator that is much more user friendly, authors have programmed some rules to prevent crashes of the program caused by user.

Users must enter a valid variable (Figure 4a), a valid function (Figure 4b), a valid range (Figure 4c) or a valid initial guess or a valid derivative function (Figure 4d) before proceed to click the *Compute* button. Otherwise a warning message will pop up and the searching process will be terminated. Nevertheless, if the defined variable is different from the variable in the function, the process will be halted (Figure 4e).

The computation process will terminate when the stopping criteria is met. Later, a graph, as shown in Figure 5, to show the roots of equation will pop-up to show to user for better understanding. The blue line graph shows the shape of the function, while the brown point plot shows the approximation root in every iteration.

## V. THE EVALUATION OF ROOT FINDING SPREADSHEET CALCULATOR

We have tested the root finding spreadsheet calculator on 74 Undergraduate Engineering students who took Engineering Mathematics IV (Numerical Methods) course in Universiti Tun Hussein Onn Malaysia in Semester I 2014/2015.

There are four aspects of evaluation being conducted for this spreadsheet calculator, namely content, design &

presentation, teaching strategy design, and technical aspect by using a questionnaire that consisted of 24 items. A 5-point Likert scale was used to rate the response of the participants towards the root finding spreadsheet calculator. The participants were asked to answer each question by rating each item using the following respond scale: 1 = strongly disagree; 2 = disagree; 3 = neutral; 4 = agree and 5 = strongly agree.

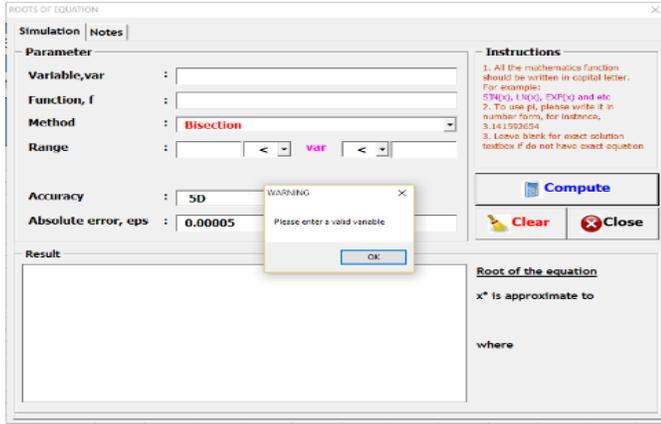


Figure 4a: Invalid variable is entered

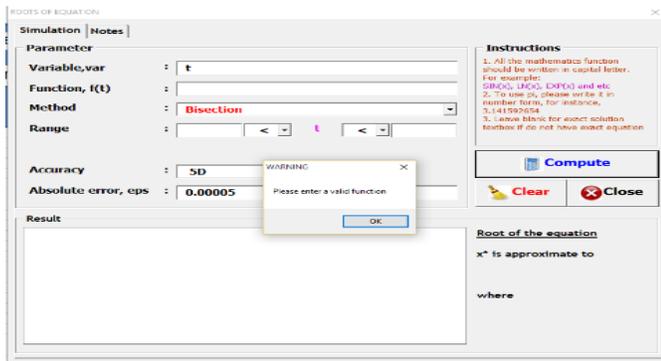


Figure 4b: Invalid function is entered

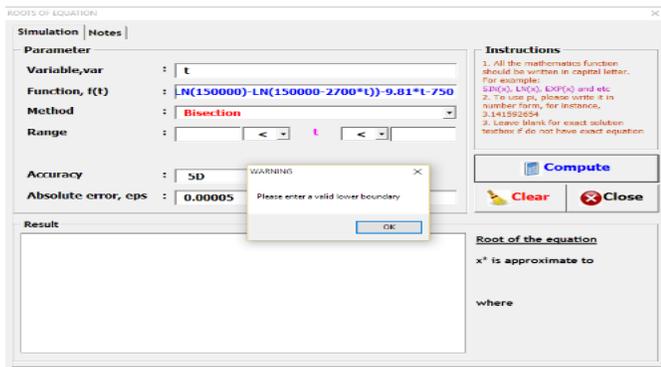


Figure 4c: Invalid range is entered

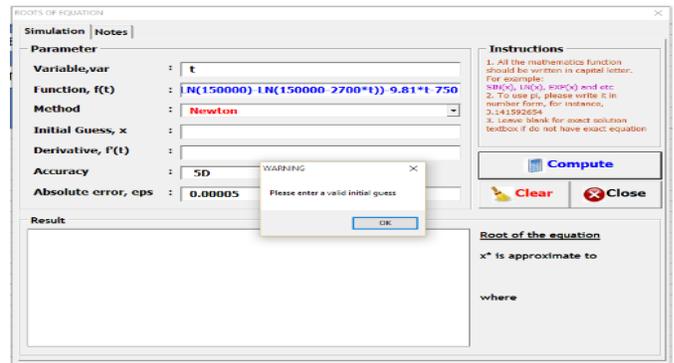


Figure 4d: Invalid initial guess is entered

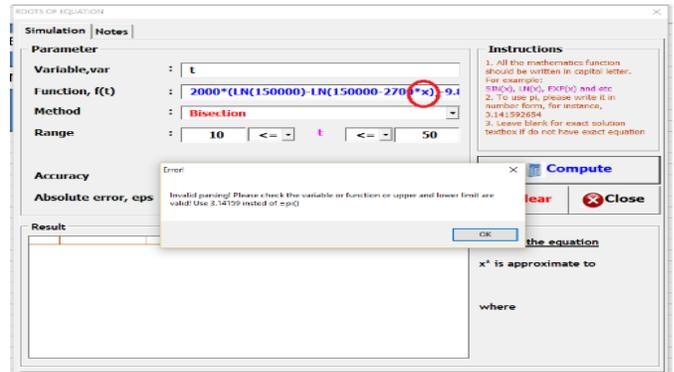


Figure 4e: Different variable used in function

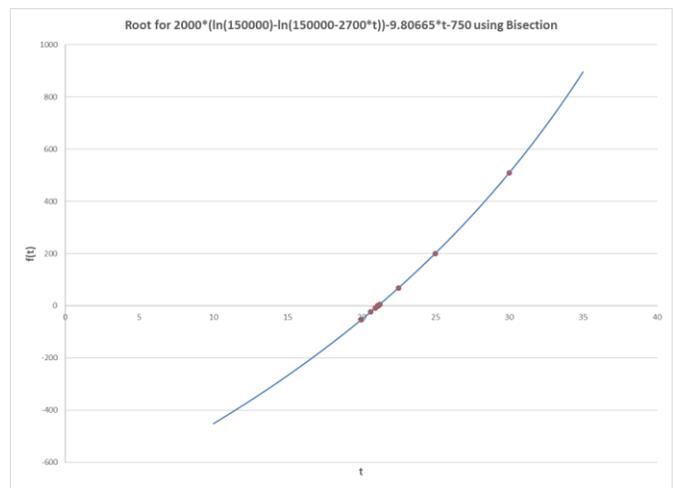


Figure 5: Graph for root of equation

The findings of the root finding spreadsheet calculator are shown in Table 1. Findings indicated that the majority of the students agreed that the root finding spreadsheet calculator is useful in terms of content ( $M=4.21$ ,  $SD=0.49$ ), design & presentation ( $M=4.19$ ,  $SD=0.49$ ), teaching strategy design ( $M=4.19$ ,  $SD=0.54$ ) and technical aspect ( $M=4.16$ ,  $SD=0.45$ ). According to the results shown in mode, the majority of the participants agreed that this root finding spreadsheet calculator is good for all four aspects. This can be proven by the statement given by the participants during the interview. Some of the participants mentioned that “It is satisfactory and can be applied in Numerical Methods course”, “It is interesting and effective” and “facilitates learning”. However, they

commented “the colour of the system is unattractive.”, “audio tutorial should be integrated and etc”.

Table 1  
Descriptive Analyses of root finding spreadsheet calculator Evaluation

Descriptive Statistics	Content	Design & Presentation	Teaching Strategy Design	Technical Aspect
Mean	4.21	4.19	4.19	4.16
Std. Deviation	0.49	0.49	0.54	0.45
Mode	4.00	4.00	4.00	4.00

## VI. CONCLUSION

A Root Finding spreadsheet calculator with graphical user interface input form was developed to approximate the root of single dimension equation using VBA programming. The spreadsheet calculator is very user friendly. It provides an alternative tool for approximating the root of equation via Bisection and Newton-Raphson methods. It can be used as a marking scheme for educators and students who need its full solutions. Last but not least, it effectively reduces the calculation time and increases the students’ learning interest.

## REFERENCES

- [1] K. G. Tay, S. L. Kek, and R. Abdul-Kahar, “A Bivariate Approximation Spreadsheet Calculator by Lagrange Interpolation,” *Spreadsheets in Education (eJSiE)*, vol. 7, no. 1, pp. 1-8, ISSN 1448-6156, 2014.
- [2] K. G. Tay, S. L. Kek, and R. Abdul-Kahar, “A Spreadsheet Solution of a System of Ordinary Differential Equations Using the Fourth-Order RungeKutta Method,” *Spreadsheets in Education (eJSiE)*, vol 5, no. 2, pp. 1-10, ISSN 1448-6156, 2012.
- [3] K. G. Tay, T. H. Cheong, M. F. Lee, S. L. Kek, and R. Abdul-Kahar, “A fourth-order Runge-Kutta (RK4) Spreadsheet Calculator For Solving A System of Two First-Order Ordinary Differential Equations Using Visual Basic (VBA) Programming,” *Spreadsheets in Education (eJSiE)*, vol. 8, no. 1, pp. 1-9, 2015.
- [4] K. G. Tay, S. L. Kek, R. Abdul-Kahar, M. A. Azlan, and M. F. Lee, “A Richardson’s Extrapolation Spreadsheet Calculator for Numerical Differentiation,” *Spreadsheets in Education (eJSiE)*, vol. 6, no. 2, pp. 1-5, ISSN 1448-6156, 2013.
- [5] K. G. Tay, T. H. Cheong, Nur Kamil Adli Mohd Nawar., S. L. Kek, and R. Abdul-Kahar, “A Romberg Integral Spreadsheet Calculator,” *Spreadsheets in Education (eJSiE)*, vol. 8, no. 2, pp. 2, 2015.
- [6] K. G. Tay, S. L. Kek, and R. Abdul Kahar, “Improved Richardson’s Extrapolation Spreadsheet Calculator for Numerical Differentiations,” in *Proc. 21<sup>st</sup> Mathematical Science National Symposium (SKSM 2013)*, Malaysia, 6-8 Nov 2013, AIP Conference Proceedings 1605, pp. 740-743, 2014.
- [7] K. G. Tay, S. L. Kek, and R. Abdul Kahar, “New Richardson’s extrapolation spreadsheet calculator using VBA programming for numerical differentiations,” in *Proc. International Conference on Mathematical, Engineering, Industrial Application (ICOMEIA 2014)*. 28-30 May 2014, AIP Conference Proceedings 1660, pp. 0500631-0500634, 2015.
- [8] K. G. Tay, T. H. Cheong, S. L. Kek and R. Abdul Kahar, “The Fourth version of Richardson’s Extrapolation Spreadsheet Calculator Using VBA Programming For Numerical Differentiations,” in *Proc. International Conference on Global Trends in Academic Research*, 2-3 June 2014, GTAR © 2014 Bali, Indonesia Global Illuminators.
- [9] K. G. Tay, S. L. Kek, T. H. Cheong and R. Abdul Kahar, “The euler’s spreadsheet calculator using VBA programming for solving ordinary differential equations,” in *Proc. International Integrated Engineering Summit (IIES 2014)*, Universiti Tun Hussein Onn Malaysia, Johor, 1-4 December 2014.
- [10] K. G. Tay, S. L. Kek, and R. Abdul Kahar, “Numerical Methods with Excel (2013),” Penerbit UTHM.
- [11] S. C. Chapra, and R. P. Canale, “Numerical Methods for Engineers,” 5<sup>th</sup> ed., McGraw Hill.