The Euler’s Graphical User Interface Spreadsheet Calculator For Solving Ordinary Differential Equations By Visual Basic For Application Programming

Kim Gaik Tay¹, Tau Han Cheong², Ming Foong Lee¹, Sie Long Kek¹ and Rosmila Abdul-Kahar¹

¹Universiti Tun Hussein Onn Malaysia, 86400 Parit Raja, Batu Pahat, Johor, Malaysia
²UnivertitiTeknologi Mara, Shah Alam, Selangor Malaysia²

Corresponding author: tay@uthm.edu.my

Abstract. In the previous work on Euler’s spreadsheet calculator for solving an ordinary differential equation, the Visual Basic for Application (VBA) programming was used, however, a graphical user interface was not developed to capture users input. This weakness may make users confuse on the input and output since those input and output are displayed in the same worksheet. Besides, the existing Euler’s spreadsheet calculator is not interactive as there is no prompt message if there is a mistake in inputting the parameters. On top of that, there are no users’ instructions to guide users to input the derivative function. Hence, in this paper, we improved previous limitations by developing a user-friendly and interactive graphical user interface. This improvement is aimed to capture users’ input with users’ instructions and interactive prompt error messages by using VBA programming. This Euler’s graphical user interface spreadsheet calculator is not acted as a black box as users can click on any cells in the worksheet to see the formula used to implement the numerical scheme. In this way, it could enhance self-learning and life-long learning in implementing the numerical scheme in a spreadsheet and later in any programming language.

1. Introduction

Applying numerical methods in classroom purpose by using calculator can be tedious and boring due to their repetitive calculations. Hence, there is a series of papers which is working on applying numerical methods in classroom purpose by using a spreadsheet with or without using Visual Basic for Application (VBA) programming has been discussed. Moreover, by employing VBA programming, a spreadsheet calculator or solver with a user-friendly graphical user interface (GUI) has been developed in recent papers. For examples, by using VBA programming, a nonlinear equation GUI [1] has been developed.

Originally, without using VBA programming and GUI, solving the non-linear system by using Newton’s method in a spreadsheet was discussed in the paper [2], whereas Kek and Tay [3] developed a solver for a system of linear equations in a spreadsheet. The interpolation solver using Newton’s divided difference and Lagrange interpolation was explored in the studies [4] and [5] respectively. On the other hand, a bivariate approximation spreadsheet calculator [6] has been developed. On top of that, approximating the dominant and smallest eigenvalue using spreadsheet has been discussed in the works [7] and [8] respectively. Next, a numerical differentiation spreadsheet calculator [9] has been
developed. Then, a series of work on developing Richardson’ extrapolation spreadsheet calculator or solver [10-14] can be discovered. Later, a Romberg Spreadsheet Calculator [15] has been created for numerical integration.

Furthermore, the tips on how to implementing the fourth-order Runge-Kutta (RK4) method on solving a single ODE and a system of ODEs in the spreadsheet were demonstrated in [16-17], respectively. While a spreadsheet calculator as discussed in [18-19] has been developed for solving a system of ODEs and a single ODE by using the RK4 method with VBA programming. A spreadsheet calculator [20] has been developed for solving a single ODE by Euler’s method with VBA programming. Lastly, in [21] it has shown the tips on how to solve Laplace equation using a spreadsheet.

It is commented that the study [20] works on solving ODE by Euler’s method with VBA programming, however, there is no GUI provided, so users may confuse the input and output. Besides, the existing Euler’s spreadsheet calculator is not interactive as there is no prompt message if there is a mistake in inputting the parameters. On top of that, there are no users’ instructions to guide users to input the derivative function. Hence, in this paper, we developed an interactive and user-friendly GUI for solving a single ODE by Euler’s method with users’ instruction and interactive prompt error message by using VBA programming.

2. Euler’s method for Solving Ordinary Differential Equations
Consider a general form of the first-order ODE (1) as given below:

\[
\frac{dy}{dx} = f(x, y), \quad y(x_0) = y_0, \quad x_n(h)x_n
\]  

(1)

Here, \( x \) is the independent variable, \( y \) is the dependent variable, \( n \) is the number of point values, \( f \) is the function of the derivation and \( h \) the fixed step size. The aim is to determine the discrete values of unknown function \( y(x) \) whose derivative satisfies (1) and the corresponding initial value. In doing so, the interval \( x \) is discretized to be \( x_0, x_1 = x_0 + h, x_2 = x_0 + 2h, ..., x_n = x_0 + nh \)

By taking the Taylor series approximation on the dependent variable \( y(x) \) at the point \( x_{k+1} \) up to first order, we get

\[
y(x_{k+1}) = y(x_k) + hy'(x_k)
\]

\[
y_{k+1} = y_k + hf(x_k, y_k), \quad k = 0,1,n-1
\]

(2)

Hence, Euler’s method which is given by (2) gives the numerical approximation to determine the discrete values of the unknown \( y(x) \).

3. Numerical Example
A simple RL-electrical circuit consists of electrical current \( i \) (in amperes), resistance \( R \) (in ohms), inductance \( L \) (in Henrys), and electromotive force \( E(t) \) (in volts) as shown in figure 1 below.

![Figure 1. RL-circuit](image)

Given \( E(t) = 50(1-e^{-t}) \), \( R = 5\Omega \) and \( L = 0.5 \), then by applying Kirchhoff’s Law, the RL-circuit can be modeled as \( \frac{di}{dt} + 10i = 100(1-e^{-t}) \). Find the current \( i(t) \) for \( 0 \leq t \leq 0.05 \) with \( \Delta t = 0.01 \). from
the above 1\textsuperscript{st}-order differential equation if there is no current flows through the circuit at initial time \( t = 0 \). If the exact solution is \( i(t) = 10 - \frac{100}{9} e^{-t} + \frac{10}{9} e^{-10t} \), find the absolute errors at each iteration.

3.1. Solution

**Step 1:** Given \( \frac{di}{dt} + 10i = 100(1 - e^{-t}) \), rewrite in the form of equation (1), hence

\[
\frac{di}{dt} = 100(1 - e^{-t}) - 10i = f(t, i)
\]

**Step 2:** The solution by Euler’s method is given by

\[
i_{k+1} = i_k + hf(t_k, i_k), \quad k = 0, 1, \ldots, 4
\]

\[
i_k = i_k + 0.01 \left[ 100(1 - e^{-t}) - 10i \right], \quad k = 0, 1, \ldots, 4
\]

**Step 3:** Apply the Euler GUI spreadsheet calculator which is given in the next section to obtain the numerical solution.

4. The Euler GUI Spreadsheet Calculator

Figure 2 shows the initial Euler GUI Spreadsheet calculator worksheet. It consists of the **Input** button, information of the solved ODE in the box and solution of the ODEs under the box.

| A | B | C | D | E | F | G | H | I | J | K | L |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **EULER SPREADSHEET CALCULATOR** | INPUT |

**DO NOT MODIFIED ANY VALUE INSIDE THIS BOX**

| Independent Var | x | Depend Var | y | Recurrence Eqn | \( y_{i+1} = y_i + hf(x_i, y_i) \) |
|---|---|---|---|---|---|
| f(x,y) | \( x+y \) |

| Range | \( 0 \leq x \leq 1.2 \) | \( y(x) = 0 \) | 0 | Step size, \( h = 0.2 \) |
|---|---|---|---|---|
| # | \( x[#] \) | \( y[#] \) | Exact | Error |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 |
| 1 | 0.2 | 0 | 0.0214 | 0.0214 |
| 2 | 0.4 | 0.04 | 0.09182 | 0.05182 |
| 3 | 0.6 | 0.128 | 0.22212 | 0.09412 |
| 4 | 0.8 | 0.2736 | 0.42554 | 0.15194 |
| 5 | 1 | 0.48832 | 0.71828 | 0.22996 |

**Figure 2.** Initial Euler GUI Spreadsheet calculator worksheet

Once the Input button is clicked, the default GUI will be popped up as shown in figure 3. Users may click the Compute button to recalculate the default typed ODE and its full solution will be given in figure 2. To input a new ODE, users may click the Clear button to clear the default parameters. To close the GUI, users may click the Close button. The top portion of the GUI displays the general form of the IVP of the first-order ODE with its intervals as well as the Euler’s method formula. On the other hand, the left portion enables users to define the independent and dependent variables of the ODE to be solved, the right-hand side derivative function of the ODE, the range to be solved, initial condition and a step size \( h \). Users are also allowed to choose the desired accuracy of calculations (number of decimal places used in the calculations) by selecting the drop down box of **Accuracy.** If
the exact solution is available, users may enter it in the left bottom part. The instructions to input the parameters are written in the right part of the GUI.

Figure 3. The default GUI

The parameters to be entered in order to solve the numerical example in section 3 are shown in figure 4. The full solutions of the numerical example in section 3 is displayed in Excel worksheet as given in figure 5. Notice that the GUI is not acted like a black box where students cannot learn the process of the Euler’s method for solving an ODE. This is because users can click in any cell in figure 5 to see the formula used in order to solve an ODE by Euler’s method. For example, users can click in cell D17 and its formula appears on top of the formula bar as shown in figure 5. This will enhance the self-learning and life-long learning of the students since in class students are taught how to solve the numerical methods using calculator only. For beginners who just learn numerical methods without the knowledge of programming, Excel may serve as a beginning step for them to implement and visualize the numerical scheme as the nature of Excel is in tabular form and users can see its immediate formula and results in the worksheet.
Figure 4. Main GUI
Figure 5. Full Solution

The graph of the exact solution is shown in red solid line, Euler’s solution is shown in blue point while the absolute error between exact solution and Euler’s solution is depicted in green point as given in figure 6. The left grid is for exact and Euler’s solution, whereas the right grid is for absolute error.

Figure 6. The graphs of exact solution, Euler’s solution and their absolute errors
On top of that, the GUI is interactive as the defined variables and the inputted function are not tally, an error message will pop out as seen in figure 7. Besides, if the starting or ending points of the interval to be solved is not provided, but Compute button is clicked, then an error message as seen in figures 8-9 will appear. Furthermore, if the initial condition is not entered, whereas Compute button is clicked, again another error message will pop out as displayed in figure 10. Lastly, if Compute button is clicked without a step size $h$ then users will be prompted by a corresponding error message as pictured in figure 11.

![Euler Spreadsheet Calculator](image)

**Figure 7.** Error Message for variable and derivative function

![Euler Spreadsheet Calculator](image)

**Figure 8.** Error message of starting point
Figure 9. Error message of ending point

Figure 10. Error message of initial condition
5. Conclusion
An interactive Euler’s GUI spreadsheet calculator for solving a single ODE has been developed by using VBA programming. We hope that this Euler’s GUI spreadsheet calculator can serve as a numerical tool to help educators to prepare their marking scheme and students to obtain full solution quickly and easily. Once users click on any cell in the solution which is given in Excel worksheet, they can see the corresponding formula used to solve an ODE by using Euler’s method. Thus, this Euler’s GUI spreadsheet calculator is not only a black box which let students obtaining the full solution without understanding its procedure, but will enhance self-learning and life-long learning process of the students toward to implement the numerical scheme in Excel spreadsheet and in any programming language later on.

Acknowledgement
This work was supported by Office for Research, Innovation, Commercialization and Consultancy Management (ORICC), Universiti Tun Hussein Onn Malaysia (UTHM) Fund.

References
[1] Cheong T H, Tay K G and Abdul-Kahar, R 2016 Nonlinear Equation Graphical User Interface Solver using Excel VBA Programming Journal of Telecommunication, Electronic and Computer Engineering 8(2) 59-63
[2] Tay K G, Kek S L and Abdul-Kahar R 2009 Solving Non-Linear Systems by Newton’s Method Using Spreadsheet Excel Proceeding of the 3rd International Conference on Science and Mathematics Education (CoSMED 2009) 452-456
[3] Kek S L and Tay K G 2008 Solver for System of Linear Equations Proceeding of the National Symposium on Application of Science Mathematics 2008 (SKASM 2008) 605-615.
[4] Kek S L and Tay K G 2009 Design of Spreadsheet Solver for Polynomial Interpolation, National Seminar on Science and Technology 2009 (PKPST 2009) 69-73
[5] Tay K G, Kek S L and Abdul-Kahar R 2010 Lagrange Interpolating Polynomial Solver Using
Spreadsheet Excel Proceeding of the National Symposium on Application of Science Mathematics 2010 (SKASM 2010) and 18th Mathematical Science National Symposium (SKSM 2010) 331-337

[6] Tay K G, Kek S L and Abdul-Kahar R 2014 A Bivariate Approximation Spreadsheet Calculator by Lagrange Interpolation *Spreadsheets in Education (eJSiE)* 7(1) 1-8

[7] Tay K G and Kek S L 2008 Approximating The Dominant Eigenvalue Using Power Method Through Spreadsheet Excel *Proceeding of the National Symposium on Application of Science Mathematics 2008 (SKASM 2008)* 599-604

[8] Tay K G and Kek S L 2009 Approximating the Smallest Eigenvalue Using Inverse Method Through Spreadsheet Excel *Proceeding of the 17th National Symposium on Mathematical Science (SKSM 2009)* 653-658

[9] Tay K G, Kek S L and Abdul-Kahar R 2013 Numerical Differentiation Spreadsheet Calculator *Proceedings of the National Symposium on Application of Science Mathematics 2008 (SKASM 2013)* 111-120

[10] Tay K G, Kek S L and Abdul-Kahar R, Azlan M A and Lee M F 2013 A Richardson’s Extrapolation Spreadsheet Calculator for Numerical Differentiation *Spreadsheets in Education (eJSiE)* 6(2) 1-5

[11] Tay K G, Kek S L and Abdul-Kaha, R 2014 Improved Richardson’s Extrapolation Spreadsheet Calculator for Numerical Differentiation *AIP Conference Proceedings*. 1605 740-743

[12] Tay K G, Kek S L Abdul-Kahar R 2015 New Richardson’s extrapolation spreadsheet calculator using VBA programming for numerical differentiations. *AIP Conference Proceedings* 1660 050063-1-050063-4

[13] Tay K G, Cheong T H, Kek S L and Abdul-Kahar R 2014 The Fourth Version of Richardson’s Extrapolation Spreadsheet Calculator Using VBA Programming for Numerical Differentiations *Proceeding of International of Global Trends in Academic Research (GTAR 2014)* 189-195

[14] Tay K G, Cheong T H, Kek S L and Abdul-Kahar R 2014. The Richardson’s Extrapolation Graphical User Interface Spreadsheet Solver *Journal of Telecommunication, Electronic and Computer Engineering* 8(2) 117-121

[15] Tay K G, Cheong T H, Mohd-Nawar N K A, Kek S L and Abdul-Kahar R 2015 A Romberg Spreadsheet Calculator *Spreadsheets in Education (eJSiE)* 8(2) 1-12

[16] Tay K G and Kek S L 2009 Fourth-Order Runge-Kutta Method Using Spreadsheet Excel *Proceedings of the 4th International Conference on Research and Education in Mathematics (ICREM4)* 666-672

[17] Tay K G, Kek S L and Abdul-Kahar R 2012 A Spreadsheet Solution of a System of Ordinary Differential Equations Using the Fourth-Order Runge-Kutta Method *Spreadsheets in Education (eJSiE)* 5(2) 1-10

[18] Tay K G, Cheong T H, Lee M F, Kek S L & Abdul-Kahar R 2015 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)* 8(1) 1-9

[19] Tay K G, Kek S L, Cheong T H, Abdul-Kahar R and Lee M F 2015 The Fourth-Order Runge-Kutta Spreadsheet Calculator Using Visual Basic Programming For Solving Ordinary Differential Equations *Procedia-Social and Behavioral Sciences*. 204 231-239

[20] Tay K G, Kek S L, Cheong T H and Abdul-Kahar, R. 2016 The Euler’s Spreadsheet Calculator Using Visual Basic Programming For Solving Ordinary Differential Equations. *ARPN Journal of Engineering and Science* 11(20) 11819-11822

[21] Tay K G, Kek S L and Abdul-Kahar R 2009 Solutions of Laplace’s Equations Using Spreadsheet Excel. *National Seminar on Science and Technology 2009 (PKPST 2009)*. 40-45
