ISSN 2278-3091
Volume 9, No.1.1, 2020
International Journal of Advanced Trends in Computer Science and Engineering
Available Online at http://www.warse.org/IJATCSE/static/pdf/file/ijatcse7091.12020.pdf
https://doi.org/10.30534/ijatcse/2020/7091.12020

# Initial-Value Problem of the First-Order Ordinary Differential Equations Graphical User Interface Excel Spreadsheet Calculator using Visual Basic Application Programming

**K. G. Tay[1], T. H. Cheong[2], S. L. Kek[3] , R. Abdul-Kahar [4]**
[1]Universiti Tun Hussein Onn Malaysia, 86400 Parit Raja, Batu Pahat, Malaysia, tay@uthm.edu.my
[2]Universiti Teknologi Mara, Shah Alam, Selangor, Malaysia.

## ABSTRACT

In this paper, we developed an ordinary differential Equation (ODE) graphical user interface (GUI) Excel spreadsheet calculator for solving an initial-value problem (IVP) of the first-order ODE by using the Euler, midpoint, Heun and RK4 methods via Visual basic application (VBA) programming. Users just need to input the required information to solve the ODE in an attractive GUI and once the Compute button is clicked, the numerical scheme which was written in VBA programming will be computed and its full solutions will be displayed in Excel spreadsheet. It is user-friendly and interactive as users will be prompted with error message if each of the required information is not pro-vided. A summative evaluation of this ODE GUI Excel spreadsheet calculator was conducted by involving 55 postgraduate students by using the Google forms. The findings showed that the majority of the students agreed that ODE GUI Excel spreadsheet calculator provided an interesting and interactive learning environment.

**Key words :** graphical user interface, Magnetization

## 1. INTRODUCTION

An ordinary differential equation (ODE) of order n is a differential equation consists of the functions of only one independ-ent variable and its derivatives. ODEs arise in many physical prob-lems including engineering, physics, economics and biology.

When the analytical solution of an ODE is not available, we have to seek its approximate solution by using numerical methods. There are several numerical methods such as the Euler's, Taylor series, midpoint, Heun and fourth-order Runge-Kutta (RK4) methods to solve an initial-value problem (IVP) of the first-order ODE. Solv-ing an IVP of the first-order ODE using numerical methods espe-cially the RK4 method using calculators or Excel Spreadsheet can be tedious because of its complicated and repetitive calculations. Hence, there is a need to develop a tool for expert users who need its quick solution of the ODE.

A spreadsheet calculator for solving an IVP of the first-order ODE and system of ODEs by the RK4 method using VBA programming in Excel Spreadsheet was developed by Tay et.al [1] and Tay et. al [2] respectively.

A spreadsheet calculator was developed by Tay et. al [3] for solving an IVP of the first-order ODE by the Euler's method with VBA programming in Excel Spreadsheet. Eventually, Tay et. al [4] up-graded the spreadsheet calculator in Tay et. al [3] into a user friend-ly and interactive graphical user interface (GUI) version so that users would not confuse the input with the output which were dis-played in the same worksheet in the previous version.

Ghaddar [5] developed functions in Excel spreadsheet to solve a single and system of ODES of IVP and boundary-value problem (BVP) with the aid of Add-In software library in Ghaddar [6] However, the Add-In software library in Ghaddar [6] is not free hence it reduces the usage of this function in Excel spreadsheet for solving ODEs. On the other hand, Hingmire, Gosavi and Patil [7] developed Spreadsheet Calculator for solving ODE using Milne's method through VBA programming, however they didn't develop the GUI to capture the inputs.

Motivated with the work of Tay et.al [4], here we intend to develop a user friendly and interactive GUI spreadsheet calculator for solv-ing an IVP of the first-order ODE by the Euler, midpoint, Heun and RK4 method**.**.
copy the image to the Windows clipboard and then Edit | Paste Special | Picture (with "Float over text" unchecked).

## 2. IVP OF THE FIRST-ORDER DIFFERENTIAL EQUA-TIONS

Consider an IVP of the first-order differential equation

$$y' = f(x,y), \quad y(x_0) = y_0, \quad x_0 \le x \le x_n. \quad (1)$$

By using numerical methods, the interval of $x$ was divided into $n$ subintervals with step size $h$ in $x_i = x_0 + ih$. We needed to solve $y$ from $i=1$ to $n$. It was noticed that by using numerical methods, we only obtained discrete values of $y$. However if we solved the differential equation analytically, we obtained a continuous function of $y(x)$. Four numerical methods, i.e. the Euler, Heun, Midpoint, and the Fourth-order Runge-Kutta method as given in Table 1 will be utilized here.

**Table 1:** Numerical methods for solving IVP of the first-order DE

| Methods | Formula |
|---|---|
| Euler | $y_{i+1} = y_i + hy_i' = y_i + hf(x_i, y_i), \quad i = 0,1,2,\ldots n$ |
| Midpoint | $y_{i+1} = y_i + k_2, \quad i = 0,1,2,\ldots n$ <br> where <br> $k_1 = hf(x_i, y_i), \quad k_2 = hf\left(x_i + \frac{h}{2}, y_i + \frac{k_1}{2}\right),$ |
| Heun | $y_{i+1} = y_i + \frac{k_1 + k_2}{2}, \quad i = 0,1,2,\ldots n$ <br> where <br> $k_1 = hf(x_i, y_i), \quad k_2 = hf(x+h, y_i + k_1).$ |
| RK4 | $y_{i+1} = y_i + \frac{k_1 + 2k_2 + 2k_3 + k_4}{6}, \quad i = 0,1,2,\ldots n$ <br> Where <br> $k_1 = hf(x_i, y_i), \quad k_2 = hf\left(x_i + \frac{h}{2}, y_i + \frac{k_1}{2}\right),$ <br> $k_3 = hf\left(x_i + \frac{h}{2}, y_i + \frac{k_2}{2}\right), \quad k_4 = hf\left(x_i + h, y_i + k_3\right).$ |

Here, given the initial condition $y(x_0) = y_0$, then we can solve $y_1, y_2, \ldots y_n$ by using the above mentioned numerical methods.

## 3. NUMERICAL EXAMPLES

The *RC* series circuit can be described by the following differential equation

$$RC\frac{dV}{dt} + V = Ee^{-\frac{t}{RC}}$$

with initial condition $V(0) = 1$. If $R = 10\ \Omega$, $C = 0.1$ F, and $E = 15$ V. The voltage $V$ for $0 \le t \le 0.05$ in the *RC* series circuit was found using the Euler's, midpoint, Heun's and RK4 methods with $\Delta t = 0.01$. The absolute errors were found if the exact solution is given by $V(t) = e^{-t}(15t+1)$.

### 3.1. Solution

Step 1: By substituting $R = 10\ \Omega$, $C = 0.1$ F, and $E = 15$ V into the first-order differential equation, we obtained

$$\frac{dV}{dt} = 15e^{-t} - V = f(t, V).$$

Step 2: Given $h = 0.01$, initial condition $V_0 = 1$, the specific formula by the aforementioned methods are given by Table 2

**Table 2**: Numerical solutions for solving IVP of the first-order DE

| Methods | Formula |
|---|---|
| Euler | $V_{k+1} = V_k + 0.01(15e^{-t} - V) \quad k = 0,1,2,\ldots 5$ |
| Midpoint | $V_{k+1} = V_k + k_2, \quad k = 0,1,2,\ldots 5$ <br> where $k_1 = 0.01(15e^{-t} - V), \quad k_2 = 0.01\left(15e^{-\left(t+\frac{h}{2}\right)} - \left(V + \frac{k_1}{2}\right)\right)$ |
| Heun | $V_{k+1} = V_k + \frac{k_1 + k_2}{2}, \quad k = 0,1,2,\ldots 5$ <br> where $k_1 = 0.01(15e^{-t} - V), \quad k_2 = 0.01\left(15e^{-(t+h)} - (V + k_1)\right).$ |
| RK4 | $V_{k+1} = V_k + \frac{k_1 + 2k_2 + 2k_3 + k_4}{6}, \quad k = 0,1,2,\ldots 5$ <br> $k_1 = 0.01(15e^{-t} - V),$ <br> $k_2 = 0.01\left(15e^{-\left(t+\frac{h}{2}\right)} - \left(V + \frac{k_1}{2}\right)\right),$ <br> Where <br> $k_3 = 0.01\left(15e^{-\left(t+\frac{h}{2}\right)} - \left(V + \frac{k_2}{2}\right)\right),$ <br> $k_4 = 0.01\left(15e^{-(t+h)} - (V + k_3)\right).$ |

Step 3: Applied the first-order ODE GUI Excel spreadsheet calculator which is given in the next section to obtain the solutions for respective methods.

## 4. THE FIRST-ORDER ODE GUI EXCEL SPREADSHEET CALCULATOR

Figure 1 shows the initial ODE GUI Excel Spreadsheet calculator worksheet. It consists of the title in the top box, Input button below the title box, information of the solved ODE in the middle box and solution of the ODEs under the middle box.

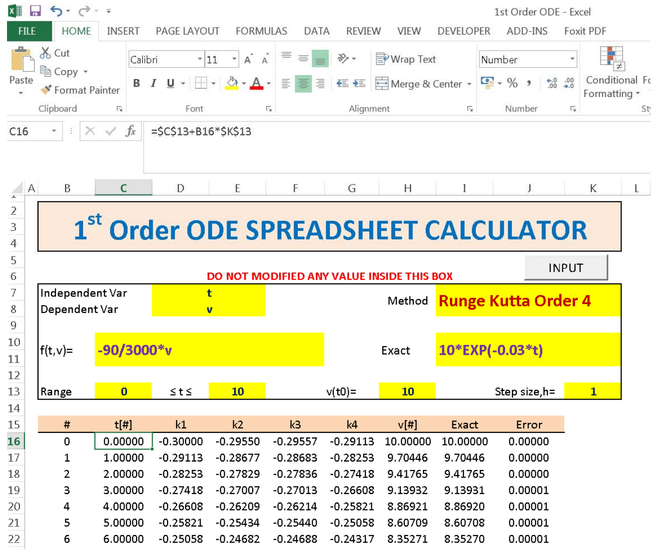



**Figure 1:** The main menu

Once the input button is clicked, Figure 2 pops up. It consists of parameters of the ODE, instructions, Compute, Clear and Close buttons. There are default parameters provided in the GUI, and users may click Compute button to solve the ODE. To input a new ODE, users may click Clear button to clear all provided parameters values. The Close button will close Figure 2.

To solve the numerical example in Section 3, we typed in the following parameters values as shown in Figure 3. We can select the desired accuracy of calculation (number of decimal places used in calculation) by dropping down the Accuracy drop menu. In the Method drop down menu, users can choose desired method of solution such as Euler's, Midpoint, Heun and the RK4 methods. Users may type the exact solution in the exact solution text box if the exact solution is available. To get the full solution, users have to click the Compute button.

I.

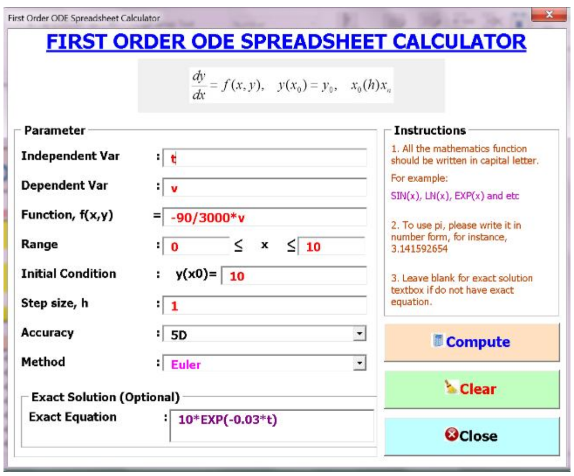



**Figure 2:** The initial parameters values

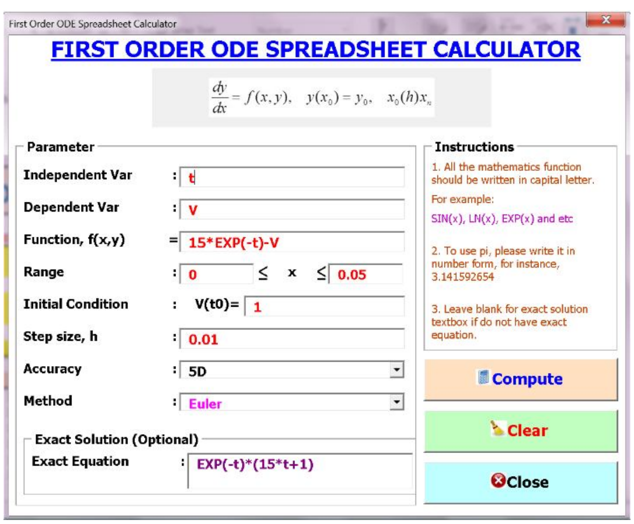
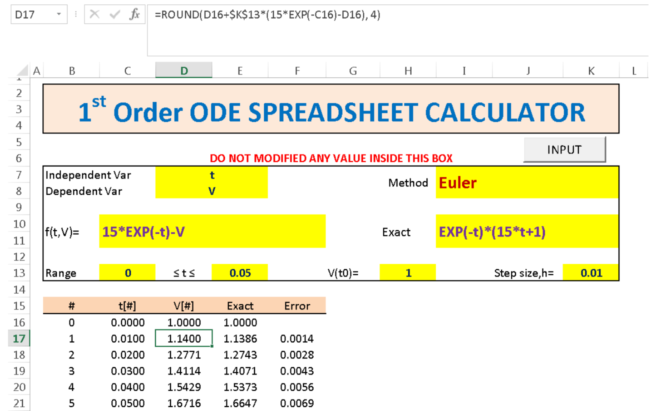


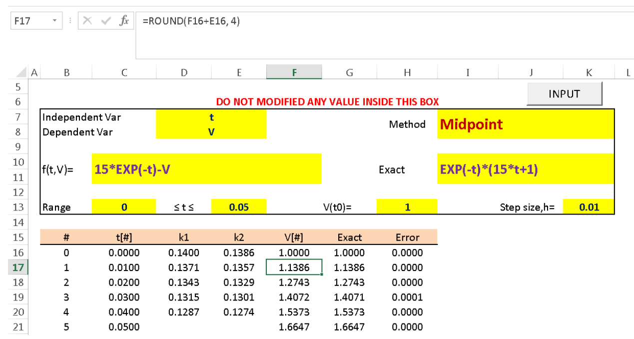

**Figure 3**: The parameters value for numerical example in Section 3

Figures 4-5 show the full solutions of the numerical example in Section 3 by the Euler's and midpoint methods. Heun's and RK4 methods give the same solutions as given in midpoint method in Figure 5. While Figures 6-7 display the graphical solutions of the numerical example in Section 3 by the Euler's and midpoint method. Heun's and RK4 methods give the same graphical solutions as given Figure 7.




| # | t[#] | V[#] | Exact | Error |
|---|---|---|---|---|
| 0 | 0.0000 | 1.0000 | 1.0000 | |
| 1 | 0.0100 | 1.1400 | 1.1386 | 0.0014 |
| 2 | 0.0200 | 1.2771 | 1.2743 | 0.0028 |
| 3 | 0.0300 | 1.4114 | 1.4071 | 0.0043 |
| 4 | 0.0400 | 1.5429 | 1.5373 | 0.0056 |
| 5 | 0.0500 | 1.6716 | 1.6647 | 0.0069 |

**Figure 4:** The full solution for numerical example in Section 3 by Euler's method




| # | t[#] | k1 | k2 | V[#] | Exact | Error |
|---|---|---|---|---|---|---|
| 0 | 0.0000 | 0.1400 | 0.1386 | 1.0000 | 1.0000 | 0.0000 |
| 1 | 0.0100 | 0.1371 | 0.1357 | 1.1386 | 1.1386 | 0.0000 |
| 2 | 0.0200 | 0.1343 | 0.1329 | 1.2743 | 1.2743 | 0.0000 |
| 3 | 0.0300 | 0.1315 | 0.1301 | 1.4072 | 1.4071 | 0.0001 |
| 4 | 0.0400 | 0.1287 | 0.1274 | 1.5373 | 1.5373 | 0.0000 |
| 5 | 0.0500 | | | 1.6647 | 1.6647 | 0.0000 |

**Figure 5:** The full solution for numerical example in Section 3 by midpoint method

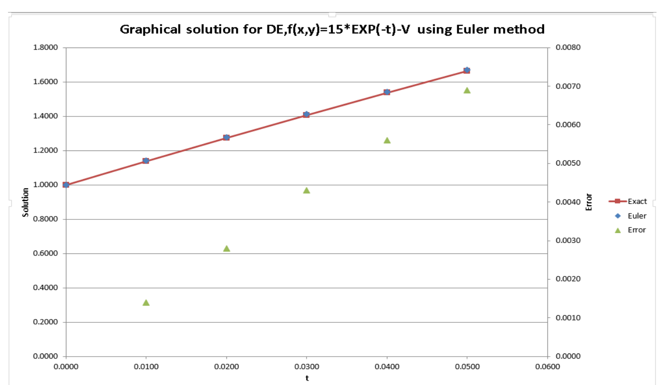



**Figure 6**: The graphical solution for numerical example in Section 3 by Euler's method

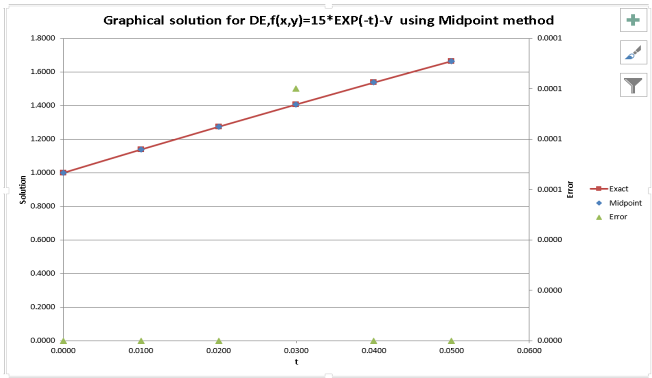



**Figure 7:** The graphical solution for numerical example in Section 3 by midpoint method

The ODE GUI Excel spreadsheet calculator is interactive because it prompts users accordingly if users do not input starting value, ending value, initial condition, a step size, or the defined variables are not tally with the given ODE function as given in Figures 8-12.

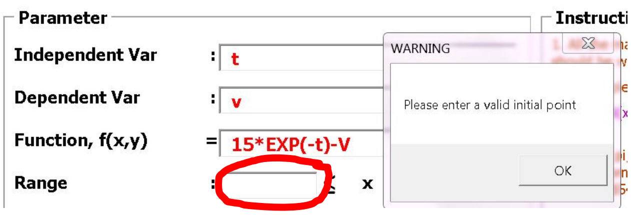



**Figure 8:** Error message if starting value is not provided

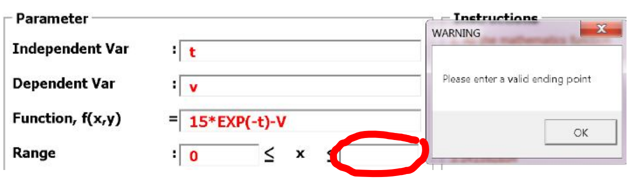



**Figure 9:** Error message if ending value is not provided

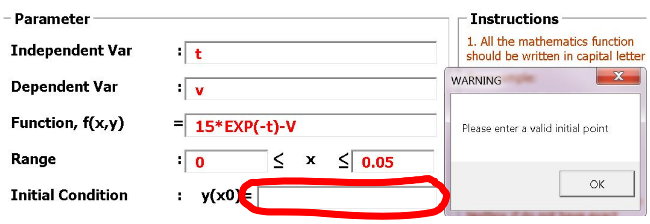



**Figure 10:** Error message if initial condition is not provided

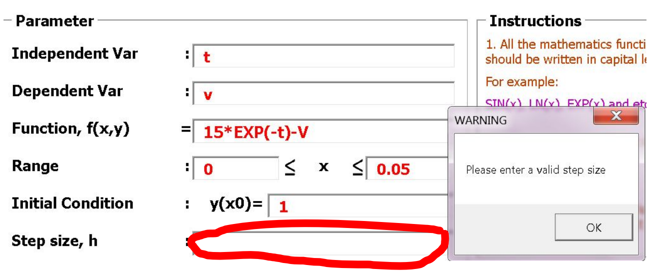



**Figure 11**: Error message if a step size is not provided

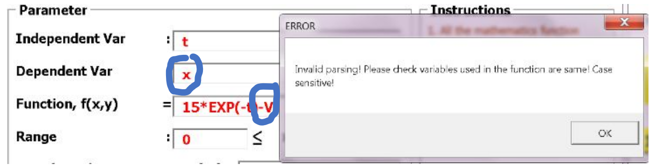



**Figure 12:** Error message if defined variables are not tally with given ODE function

The ODE GUI Excel spreadsheet calculator can promote self-learning for students as students can scroll down to see the formula used in order to implement the numerical methods as seen in the formula bar in Figures 4-5.

## 5. CONCLUSION

An ODE GUI Excel spreadsheet calculator was developed using VBA programming to solve an IVP of the first-order ODE by the Euler, midpoint, Heun and RK4 methods. The ODE GUI Excel spreadsheet calculator was found to be able to enhance learners' self-learning as learners could scroll down to see the formula used to solve the IVP of the first-order ODE by Euler, midpoint, Heun and RK4 methods. It is user friendly and interactive because error message prompts up if users do not provide necessary inputs

.

## ACKNOWLEDGEMENT

We are grateful to Universiti Tun Hussein Onn Malaysia for financially supporting this work under ORICC Fund.

## REFERENCES

1. Tay KG., Kek SL, Cheong TH, Abdul-Kahar R and Lee MF (2015), ***The Fourth-Order Runge-Kutta Spreadsheet Calculator Using Visual Basic Programming For Solving Ordinary Differential Equations. Procedia-Social and Behavioral Sciences*** 204, p. 231-239, 2015a. https://doi.org/10.1016/j.sbspro.2015.08.145
2. Tay KG, Cheong TH, Lee MF, Kek SL.& 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, 1-9.
3. Tay KG, Kek SL, Cheong TH 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, 11819-11822.
4. Tay, KG, Cheong TH, Lee MF, Kek SL and Abdul-Kahar R (2017), ***The Euler's Graphical User Interface Spreadsheet Calculator For Solving Ordinary Differential Equations By Visual Basic For Application Programming***. IOP Conf. Series: Materials Science and Engineering, 226, 1-12. https://doi.org/10.1088/1757-899X/226/1/012186

5. Ghaddar CK (2016), ***Unconventional Calculus Spreadsheet Functions,*** International Journal of Mathematics and Computational Sciences. 10194-200.
6. Ghaddar CK (2016), ExcelLab Reference Manual. www.excel-works.com.
7. Hingmire SP, Gosavi MR and Patil NA (2018), ***Milne's Spreadsheet Calculator Using VBA with Excel Programming for solving Ordinary Differential Equations***.11th International Conference on Science, Technology and Management.