The Euler’s Spreadsheet Calculator Using VBA programming For Solving Ordinary Differential Equations

Kim Gaik Tay¹,a, Sie Long Kek²,b, Tau Han Cheong³,c and Rosmila Abdul-Kahar⁴,d

¹ Department of Communication Engineering, Faculty of Electrical and Electronic Engineering, Universiti Tun Hussein Onn Malaysia
²,⁴ Department of Mathematics & Statistics, Faculty of Science, Technology and Human Development, Universiti Tun Hussein Onn Malaysia
³Faculty of Education, Universiti Teknologi Mara

a tay@uthm.edu.my, b slkek@uthm.edu.my, c cheongtauhan@uthm.edu.my, d rosmila@uthm.edu.my

Keywords: Excel spreadsheet, ordinary differential equations spreadsheet calculator, Euler’s method

Abstract. Motivated by the works of a Richardson’s Extrapolation spreadsheet calculator for differentiation, we have developed the Euler’s spreadsheet calculator using VBA programming to solve ordinary differential equations (ODEs). Users simply need to enter the independent and dependent variables used, a starting value and ending value for the independent variable, an initial value for the dependent variable, the step size, the ODE and exact function for the ODE. Lastly click the APPLY button which is associated with the VBA programming written to solve the ODEs by the Euler’s method, and finally its full solution is automatically calculated and displayed. Hopefully, this Euler’s ODEs spreadsheet calculator can help educators to prepare their marking scheme easily and assist students in checking their answers.

Introduction

The computing approaches of the ordinary differential equations (ODEs) can be roughly divided into the exact solution method and the numerical method. Since the use of the exact solution method is limited to the linear ODEs, the application of the numerical method is seen to be practical in solving engineering problems. This is because the exact solution of the nonlinear ODEs is difficult to be analysed and its existence might be questioned. In view of these, the applications of the numerical method could approximate the solution of the ODEs, particularly for the nonlinear ODEs. As such, it is necessary to develop a tool in order to solve the ODEs easily.

A series of papers working on solving numerical methods in classrooms and examination situations using spreadsheet which focuses on systems of nonlinear and linear equations, approximation of interpolation, computing of eigenvalues, numerical differentiation, ordinary differential equations (ODEs) by the Fourth-order Runge-Kutta (RK4) and the Laplace equation can be seen in [1-13]. Only the work of numerical differentiations [8-10] deals with spreadsheet calculator while the rest of work involves spreadsheet tips on solving respective numerical methods without using VBA programming. In our definition, a spreadsheet calculator is easy to use without having to type any commands in the spreadsheet. Users only need to input the required information, and then its full solution will automatically be calculated. Thus, the aim of this paper is to develop the Euler’s spreadsheet calculator using VBA programming to solve ODEs.
**Euler’s method for Ordinary Differential Equations**

Consider a general form of the first-order ordinary differential equation given below:

\[
\frac{dy}{dx} = f(x, y)
\]  

(1)

with the initial value \( y(x_0) = y_0 \) for the interval \( x_0 \leq x \leq x_n \). Here, \( x \) is the independent variable, \( y \) is the dependent variable, \( n \) is the number of point values, and \( f \) is the function of the derivation. The aim is to determine the unknown function \( y(x) \) whose derivative satisfies (1) and the corresponding initial values. In doing so, let us discretize the interval \( x_0 \leq x \leq x_n \) to be

\[
x_0, x_1 = x_0 + h, x_2 = x_0 + 2h, ..., x_n = x_0 + nh
\]  

(2)

where \( h \) is the fixed step size. On this basis, the unknown function \( y(x) \) can be written as the first-order Taylor series approximation at the corresponding points as given in (2), given below:

\[
\begin{align*}
\quad y(x_0) &= y_0 \\
y(x_1) &= y(x_0) + hf(x_0, y(x_0)), \quad h = x_1 - x_0 \\
y(x_2) &= y(x_1) + hf(x_1, y(x_1)), \quad h = x_2 - x_1 \\
\vdots \\
y(x_{n-1}) &= y(x_{n-2}) + hf(x_{n-2}, y(x_{n-2})), \quad h = x_{n-1} - x_{n-2} \\
y(x_n) &= y(x_{n-1}) + hf(x_{n-1}, y(x_{n-1})), \quad h = x_n - x_{n-1}
\end{align*}
\]  

(3)

Notice that (3) can simply be formulated as

\[
y_{i+1} = y_i + hf(x_i, y_i), \quad i = 0, 1, 2, \ldots n
\]  

(4)

Hence, the Euler’s method, which is defined by (4), gives the numerical solution of (1) in order to determine the unknown function \( y(x) \).

**Numerical Example**

For illustration, consider the RC-circuit as shown in Figure 1.

![Fig. 1: RC-circuit](image)

The governing first order ordinary differential equation is given by

\[
R \frac{di}{dt} + \frac{i}{C} = E'(t)
\]  

(5)

where \( R \) is the resistance (ohms), \( C \) is the capacitance (farads), \( i \) is the current (ampere), and \( E(t) \) is the voltage (volts).
Given $E(t) = \sin(100t)$ volts, $R = 5$ ohms, $C = 0.1$ farads and at the initial time $t = 0$ the initial current is $i = 0$. We want to solve the differential equation (5), which is the RC-circuit ODE, for time interval $0 \leq t \leq 5$ seconds with the time step size $\Delta t = 0.01$ seconds by using the Euler’s method. If the exact solution is given by

$$i = \frac{10}{2501} \cos 100t + \frac{500}{2501} \sin 100t - \frac{10}{2501} e^{-2t}, \quad (6)$$

then the absolute errors are calculated at each iteration.

The computation procedure of the Euler’s method is summarized as follows:

Euler’s solution method
Step 1: Rewrite the ODE in (5) by substituting the given values. That is,

$$\frac{di}{dt} = \frac{100 \cos 100t}{5} - \frac{i}{5(0.1)} = 20 \cos 100t - 2i = f(t, i) \quad (7)$$

Step 2: By using the Euler’s method, formulate (7) into the form of (4), given by

$$i_{k+1} = i_k + hf_k = i_k + hf(t_k, i_k)$$

$$i_{k+1} = i_k + 0.01(20 \cos 100t - 2i)$$

with $h = \Delta t = 0.01$ and $i = 0$ when $t = 0$

Step 3: Apply the Euler’s spreadsheet calculator, which is discussed in the next section, to obtain the numerical solution.

The Euler’s Spreadsheet Calculator for Solving ODEs
In this section, the use of the Euler’s spreadsheet calculator is discussed. Figure 2 illustrates the template of the Euler’s spreadsheet calculator and the method of usage. Firstly, users enter the independent and dependent variables into the cells C4 and D4 respectively. Secondly, the initial values for both of the mentioned variables are entered into the cells B6 and C6 respectively, whereas the ending value of the independent variable is entered into the cell G6. Thirdly, the step size $h$ is entered into the cell H6. Fourthly, the ODE given by (7) is entered into the cell D7 naturally via programming syntax or mathematical form instead of using Excel command. Fifthly, the exact solution of the ODE, which is given by (6), is entered into the cell D6.

Users can select the desired accuracy upon the number of decimal places, which are ranged from one to nine decimal places from the drop down menu in cell I10. Finally, click the APPLY button in the cell D10, which is associated with the VBA programming for the computation of the Euler’s method. Then, the numerical solution of the ODE, which is shown in the red colour cells in Figure 2, is calculated automatically based on the desired accuracy.

The details of the programming are provided in the Appendix.
Conclusion

In this paper, a spreadsheet calculator, which applies the Euler’s method for solving the ODEs, was developed. In this spreadsheet calculator design, we employed the utility of the VBA programming to simplify the use of the spreadsheet calculator. This spreadsheet calculator is very user friendly since users only need to enter relevant information to compute the full solution of the ODEs which will then be displayed to the users. In future, we hope to develop a spreadsheet calculator for solving ODEs using other numerical methods. We hope this spreadsheet calculator can serve as a tool for educators and students who need full solution of the ODEs using the Euler’s method.

Acknowledgement

This work is fully supported by UTHM multidisciplinary research grant MDR vote 1109.

Appendix

Sub IIES()
Dim i As Integer
Dim f As String
Dim exactf As String
Range("B13:F65536").Clear
Range("B13:F65536").HorizontalAlignment = xlCenter

Figure 2: The Euler’s spreadsheet calculator for solving ODEs
'Label x₀, y₀, xₙ
Cells(5, 2) = Cells(4, 3) & "0"
Cells(5, 2).Characters(Start:=2, Length:=2).Font.Subscript = True
Cells(5, 3) = Cells(4, 4) & "0"
Cells(5, 3).Characters(Start:=2, Length:=2).Font.Subscript = True
Cells(5, 7) = Cells(4, 3) & "n"
Cells(5, 7).Characters(Start:=2, Length:=2).Font.Subscript = True

'Label f(x,y)
Cells(7, 3) = "f" & Cells(4, 3) & "," & Cells(4, 4) & ")"

'i₀, x₀, y₀
Cells(13, 2) = 0
Cells(13, 3) = Cells(6, 2)
Cells(13, 4) = Cells(6, 3)

'column i, x
For i = 1 To Cells(6, 9)
    Cells(13 + i, 2) = 0 + i
    Cells(13 + i, 3) = Cells(12 + i, 3) + Cells(6, 8)
Next i

'column y
For i = 1 To Cells(6, 9)
    f = Replace(Cells(7, 4), Cells(4, 4), Cells(12 + i, 4))
    Cells(13 + i, 4) = Round(Cells(12 + i, 4) + Cells(6, 8) * Evaluate(Replace(f, Cells(4, 3), Cells(12 + i, 3))), Cells(7, 12))
Next i

'column exact y
For i = 1 To (Cells(6, 9) + 1)
    exactf = Replace(Cells(6, 4), Cells(4, 4), Cells(12 + i, 4))
    Cells(12 + i, 5) = Round(Evaluate(Replace(exactf, Cells(4, 3), Cells(12 + i, 3))), Cells(7, 12))
Next i

'\|error\|
For i = 1 To (Cells(6, 9) + 1)
    Cells(12 + i, 6) = Abs(Cells(12 + i, 5) - Cells(12 + i, 4))
Next i

Range(Cells(13, 2), Cells(13 + Cells(6, 9), 6)).Borders.LineStyle = xlContinuous
Range(Cells(13, 2), Cells(13 + Cells(6, 9), 6)).ShrinkToFit = True
Range(Cells(13, 2), Cells(13 + Cells(6, 9), 6)).Interior.Color = RGB(255, 150, 150)
End Sub

References


