Qatar University

College of Arts and Sciences

Department of Math, Statistics, and Physics

Course Project - MATH 217 Mathematics for Engineering

Numerical methods for ordinary differential equations

Submitted to: Prof. Mohamed Mabrok

Project Group Member:

Marim Elhanafy 201803468
Hagar Elsayed 201805123
Hend AlHajri 201706079
Habiba Zaky 201906435

Date: 17-12-2022

Table of Contents

List Of Figures

List Of Tables

Abstract

The solution of an ordinary differential equation means finding an explicit expression for y in terms of a finite number of elementary functions of x. This kind of solution is known as closed or finite form of solution. In the absence of such a solution, we have recourse to numerical methods of solution. In most of these methods, we replace the differential equation with a simple different equation and then solve it. These methods yield solutions either as a power series in x from which the values of y can be found by direct substitution, a set of values of x and y. The methods of Euler, Runge-Kutta, Milne, Adams-Bashforth, etc. belong to a specific class of solutions. In these methods, the next point on the curve is evaluated in short steps ahead, by performing iterations until sufficient accuracy is achieved. As such, these methods are called step-by-step methods. The methods of Euler and Runge-Kutta are discussed in this report and applied using MATLAB code with a second-order differential equation.

Introduction

Several problems in science and technology can be formulated into differential equations. The analytical methods of solving differential equations are applicable only to a limited class of equations. Quite often differential equations appearing in physical problems do not belong to any of these familiar types and one is obliged to resort to numerical methods. These methods are of even greater importance when we realize that computing machines are now readily available, which reduces numerical work considerably. Numerical methods make solving problems easier and more efficient compared to the use of analytical solutions. According to Table 1, there are a lot of methods for solving an Initial Value Problem (IVP).

Table 1: Finite different methods for solving an IVP* [1]

Methods for solving the differential equation $\frac{d}{dt}\mathbf{y}(t) = \mathbf{f}(t, \mathbf{y})$			
Method	Difference Formula	τ_j	Properties
Euler	$\mathbf{y}_{j+1} = \mathbf{y}_j + k\mathbf{f}_j$	$O(k)$	Explicit; Conditionally A-stable
Backward Euler	$\mathbf{y}_{j+1} = \mathbf{y}_j + k\mathbf{f}_{j+1}$	$O(k)$	Implicit; A-stable
Trapezoidal	$\mathbf{y}_{j+1} = \mathbf{y}_j + \frac{k}{2}(\mathbf{f}_j + \mathbf{f}_{j+1})$	$O(k^2)$	Implicit; A-stable
Heun (RK2)	$\mathbf{y}_{j+1} = \mathbf{y}_j + \frac{1}{2}(\mathbf{k}_1 + \mathbf{k}_2)$ where $\mathbf{k}_1 = k\mathbf{f}_j$ $\mathbf{k}_2 = k\mathbf{f}(t_{j+1}, \mathbf{y}_j + \mathbf{k}_1)$	$O(k^2)$	Explicit; Conditionally A-stable
Classical Runge–Kutta (RK4)	$\mathbf{y}_{j+1} = \mathbf{y}_j + \frac{1}{6}(\mathbf{k}_1 + 2\mathbf{k}_2 + 2\mathbf{k}_3 + \mathbf{k}_4)$ where $\mathbf{k}_1 = k\mathbf{f}_j$ $\mathbf{k}_2 = k\mathbf{f}(t_j + \frac{k}{2}, \mathbf{y}_j + \frac{1}{2}\mathbf{k}_1)$ $\mathbf{k}_3 = k\mathbf{f}(t_j + \frac{k}{2}, \mathbf{y}_j + \frac{1}{2}\mathbf{k}_2)$ $\mathbf{k}_4 = k\mathbf{f}(t_{j+1}, \mathbf{y}_j + \mathbf{k}_3)$	$O(k^4)$	Explicit; Conditionally A-stable

*** The points t1, t2, t3, . . . are equally spaced with step size $k = t_j+1 - t_j$. Also, $f_j = f(t_j, y_j)$ and τ_j is the truncation error for the method.**

Methods

Systems described by differential equations are often more complex to the point that simple analytical solutions are not accessible. Hence, numerical methods come handy especially with many software available for iterative methods to provide an accurate solution [2]. The methods to be discussed are Euler's method and Runge-Kutta implemented using MATLAB software.

1. Euler

To start with, Euler's method is the most basic method that was later improved into different versions [3]. As shown in Figure 1, the method is used to find an approximation to the solution curve of the initial value problem $y' = F(x, y)$ with $y(0) = y_0$ using an adaptation of 1st order Taylor series method as following: $y_{i+1} = y_i + f(t_i, y_i) * h$. Where h is the step size, and f (x, y) represents the slope of the estimation. In general, the smaller the step size the more accurate the solution is.

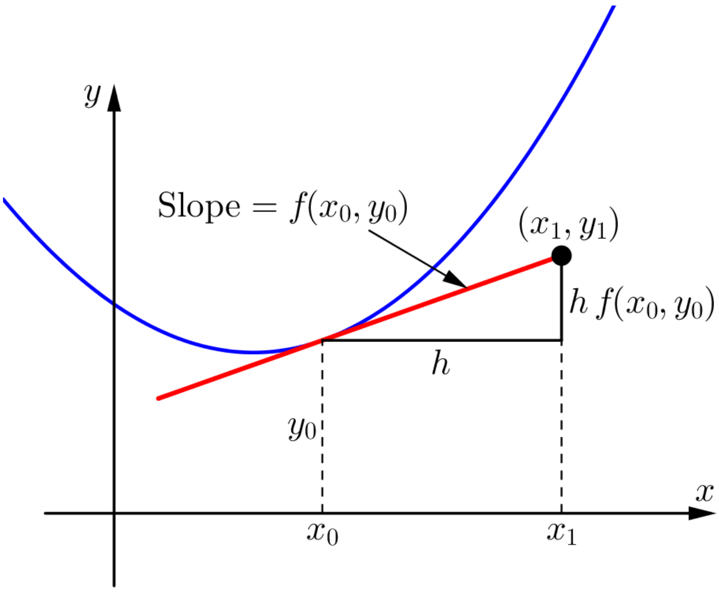

Figure 1: Euler's method diagram

The main advantages of Euler's method are it is simple and direct, and it can be used for nonlinear IVPs. While its disadvantages are it is less accurate and numerically unstable, and its approximation error is proportional to the step size h. Hence good approximation is obtained with a very small h. This requires more time discretization leading to a large computation time.

2. Runge-Kutta

Furthermore, Runge-Kutta method is a higher order approximation to the midpoint method. Instead of dividing by the midpoint to estimate the derivative, the function will be divided over 4 steps across the entire interval to obtain 4 derivatives evaluations to overcome the problem of the previous Euler's method step size problem [4].

As following:

$$K_1 = f(t_i, y_i)$$

$$K_2 = f\left(t_i + \frac{1}{2}h,\ y_i + \frac{1}{2}K_1 h\right)$$

$$K_3 = f\left(t_i + \frac{1}{2}h, y_i + \frac{1}{2}K_2 h\right)$$

$$K_4 = f(t_i + h, y_i + K_3 h)$$

$$y_{i+1} = y(t_0 + ih) = y_i + \frac{1}{6}(K_1 + 2\,K_2 + 2\,K_3 + K_4)h$$

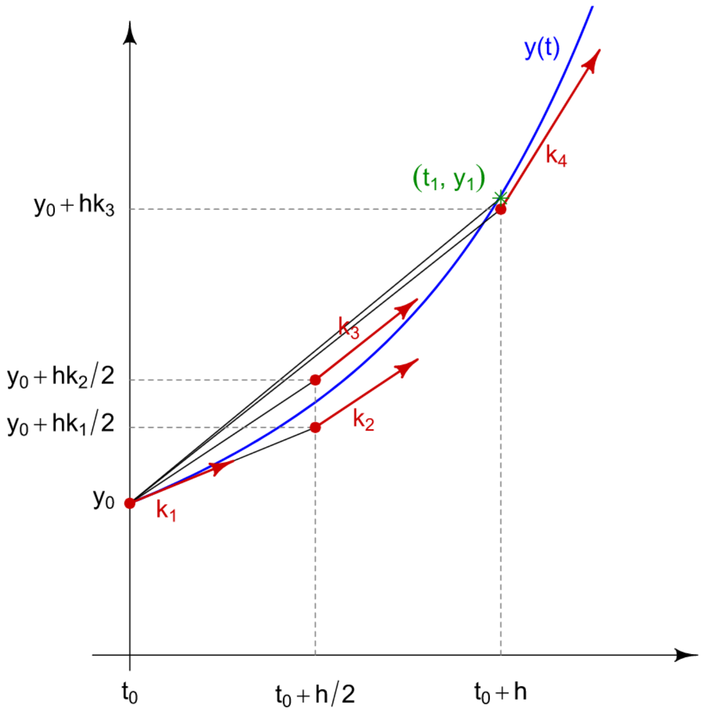

Figure 2: Runge-Kutta method diagram

For Runge-Kutta methods, the primary disadvantages are that they require significantly more computing time than multi-step methods of comparable accuracy, and they do not easily yield good global estimates of the truncation error. However, for the straightforward dynamical systems under investigation in this course, the advantage of the relative simplicity and ease of use of Runge-Kutta methods far outweigh the disadvantage of their relatively high computational cost [5].

MATLAB Code

For a better view of the difference in accuracy between the two methods, the same simple equation is implemented in MATLAB software under the same conditions.

The equation used is $\frac{dy}{dt} = 2t$, with range from 0 to 10, initial condition $y(0) = 0$, and step size $h = 1$. The actual solution is $y(t) = t^2$. For Euler and Runge-Kutta methods, we used the implemented function (eulode) and the built-in function (ode45), respectively. This is the formula for "Percentage Error" used for calculating the error for both Euler and Runge-Kutta methods:

$$\frac{|\text{Approximate Value} - \text{Exact Value}|}{|\text{Exact Value}|} \times 100\%$$

The "|" symbols mean absolute value, so negatives become positive.

***The MATLAB code is attached at the end of the report.**

***The eulode function's code is attached at the end of the report.**

Results and Discussion

Convergence and order are two concepts used for numerical analysis. Where the convergence is whether the method approximates the solution, and the order is how well it approximates it [6]. First, both Euler and Runge-Kutta are convergent because their numerical solution approaches the exact solution as the step size h goes to 0. From the code results, we can see that the accuracy of Runge-Kutta is higher than Euler's method for the same step-size h. Moreover, the numerical solutions obtained by the two proposed methods are in good agreement with the true solutions as shown in the code's graphs. Second, it is observable that the Euler method has a lower computational complexity of $O(h)$ compared to the fourth order Runge-Kutta method, which has a complexity of $O(h^4)$. As a result, the Euler method is faster to execute under the same conditions. However, the Runge-Kutta method has a lower percentage error at $t = 10$, with a value of 0% compared to the Euler method's error of 10%.

Conclusion

In this report, Euler's method, and Runge-Kutta method are used for solving ordinary differential equation (ODE) in initial value problems (IVP). Finding more accurate results needs a smaller step size for both methods, however Runge-Kutta presents more accurate results as it follows a 4-order method. From the study, the Runge-Kutta method was found to be generally more accurate and also the approximate solution converged faster to the exact solution when compared to the Euler method. It may be concluded that the Runge-Kutta method is powerful and more efficient in finding numerical solutions of initial value problems (IVP).

Task’s Distribution

Table 2: Task’s Distribution

Task	Done By
Euler Method	Marim Elhanafy – Habiba Zaky
Runge-Kutta Method	Hagar Elsayed – Hend Al-Hajri
Report	All

References

[1] Holmes, M. H. (Ed.). (2007). *Introduction to numerical methods in differential equations*. New York, NY: Springer New York.

[2] Ahmad, N. (2017). A COMPARATIVE STUDY ON NUMERICAL SOLUTION OF ORDINARY DIFFRENTIAL EQUATION BY DIFFERENT METHOD WITH INITIAL VALUE PROBLEM ARTICLE INFO NUMERICAL SOLUTION OF INTEGRAL EQUATION.

[3] "Numerical Solution of ordinary differential equations," *University of Lucknow*. [Online]. Available: https://www.lkouniv.ac.in/.

[4] K. E. Atkinson, W. Han, and D. Stewart, *Numerical solution of ordinary differential equations*. Hoboken, NJ: Wiley, 2009.

[5] Parviz Moin, Engineering Numerical Analysis, Cambridge, University press, 2001

[6] Higham, N. J. (2002). Accuracy and stability of numerical algorithms (Vol. 80). SIAM.