Contents

List of Abbreviations

IVP Initial Value Problem

DE Differential Equation

ODE Ordinary Differential Equation

LTE Local Truncation Error

GTE Global Truncation Error

Chapter 1

INTRODUCTION

An Initial Value Problem (IVP) is a Differential Equation (DE) together with one or more initial values.[8][11] It takes what would otherwise be an entire rainbow of possible solutions and whittles them down to one specific solution. The basic idea behind this problem is that, once you differentiate a function, you lose some information about that function, more specifically, you lose the constant. By integrating $y'(x)$, you get a family of solutions that only differ by a constant.[7]

1.1 Definition

An IVP is a differential equation: [11]

$$\begin{aligned} y'(t) &= f(t, y(t)) \text{ with } f : \Omega \subset \mathbb{R} \times \mathbb{R}^n \to \mathbb{R}^n \\ (t_0, y_0) &\in \Omega, \text{ called the initial condition.} \end{aligned} \tag{1.1}$$

Observations:

1. The given f in (1.1) is the defining function of IVP.
2. A unique solution, $y(t)$, of the (1.1) exists and it satisfies $y(t_0) = y_0$.

Example

Given $y'(t) = 5$ and $y(0) = -3$, find $y(t)$.

Solution:

We first integrate our $y'(t)$, then we substitute our initial condition to determine the constant (from our integration).

$$\begin{aligned}
\int y'(t)dt &= \int 5dt \\
y(t) &= 5x + c \quad \text{where } c \text{ is the constant of integration} \\
\text{using } y(t=0) &= -3 \\
-3 &= 5(0) + c \\
-3 &= c \\
y(t) &= 2x - 3
\end{aligned}$$

Remark: Note that with a different $y(0)$, the solution would be different.

1.2 Objective

In real-life situations, the differential equation that models a problem is too complicated to solve exactly, therefore one of the ways which is used to solved such problems is using methods which approximates the solution of the original problem.[2] In this report, I will discuss methods that approximates solutions at certain specified timestamps.
They are: [§ 2.1] The Euler's Method, [§ 2.2] The Modified Euler's Method, [§ 2.3] The 2nd-Order Runge-Kutta Method, [§ 2.4] The 4th-Order Runge-Kutta Method, [§ 2.5] The Adams-Bashforth 4th-Order Explicit, [§ 2.6] The Adams 4th-Order Predictor Corrector, [§ 2.7] The Runge-Kutta-Fehlberg, and [§ 2.8] The Predictor-Corrector methods

Chapter 2

METHODS

2.1 The Euler's Method

The Euler method, named after Leonhard Euler, was published in his three-volume work *Institutiones Calculi Integralis* in the years 1768 to 1770, and republished in his collected works.(Euler, 1913) [3] The Euler method is a first-order numerical procedure for solving Ordinary Differential Equation (ODE) with a given initial value. It is the most basic explicit method for numerical integration of ODE and is the simplest Runge–Kutta method.[9]

The fundamental idea of the method is based on the principle that, we can compute (or approximate) the shape of an unknown curve - in the form of a differential equation $f(t, y)$, which starts at a given point y_0 and time t_0. With this information known, we can proceed to calculate the slope (tangent line) of the curve at y_0.

The tangent line is [5]

$$y = y_0 + f(t_0, y_0) \cdot (t - t_0)$$

Now we assume that $f(t0, y0)$ is sufficiently accurate, and thus, taking a small step along that tangent line, we can approximate the actual value of the solution, y_1, at timestamp t_1, using the formula:

$$y_1 = y_0 + f(t_0, y_0) \cdot (t_1 - t_0) \tag{2.1}$$

In general, we continue to find the next approximated solution y_{n+1} at t_{n+1}, if we have the nth timestamp t_n and the approximation to the solution at this point, y_n.

We only need to modify (2.1) in this manner:

$$y_{n+1} = y_n + f(t_n, y_n) \cdot (t_{n+1} - t_n) \tag{2.2a}$$

If we assume uniform step sizes between times, t, we can define, $h = t_{n+1} - t_n$. Therefore, the formula is simplified as [5]

$$y_{n+1} = y_n + h \cdot f(t_n, y_n) \tag{2.2b}$$

The Truncation Errors

1. The **Local Truncation Error (LTE)** of the Euler method is the error made in a single step. It is the difference between the numerical solution after one step, y_1, and the exact solution (obtained using Taylor's expansion) at time $t_1 = t_0 + h$.[9]

$$\begin{aligned}\text{The numerical solution: } y_1 &= y_0 + hf(t_0, y_0)\\ \text{The exact solution: } y(t_0+h) &= y(t_0) + hy'(t_0) + \frac{1}{2}h^2 y''(t_0) + O(h^3)\\ \text{LTE} = y(t_0+h) - y_1 &= \frac{1}{2}h^2 y''(t_0) + O(h^3)\end{aligned}$$

2. The **Global Truncation Error (GTE)** is the error at a fixed time t_i, after however many steps the method needs to take to reach that time from the initial time. The global truncation error is the cumulative effect of the local truncation errors committed in each step.[1]

$$|y(t_i) - y_i| \leq \frac{hM}{2L}(e^{L(t_i - t_0)} - 1)$$

   where M is an upper bound on the second derivative of y on the given interval and L is the Lipschitz constant of f.[1]

2.1.1 The Pseudocode

To approximate the solution of the IVP

$$y' = f(t, y), \;\; a \leq t \leq b, \;\; y(a) = \alpha$$

at N equally spaced numbers in the interval $[a, b]$: [2]

Algorithm 1 :: Euler's Method

Require: endpoints a, b; integer N; initial condition α
Ensure: approximation w to y at the $(N+1)$ values of t

$h = (b-a)/N$; $t_0 = a$; $w_0 = \alpha$
for $i = 0, 1, 2, \cdots, N-1$ **do**
  $w_{i+1} = w_i + hf(t_i, w_i)$; $\{$Compute next $w_i\}$
  $t = a + ih$ $\{$Compute next $t_i\}$
end for
return (t, w)

2.2 The Modified Euler's Method

Euler's method is used as the foundation for Modified Euler's method. Euler's method uses the line tangent to the function at the beginning of the interval as an estimate of the slope of the function over the interval, assuming that if the step size is small, the error will be small. However, even when extremely small step sizes are used, over a large number of steps the error starts to accumulate and the estimate diverges from the actual functional value.[10]
The Modified Euler (which may sparingly be referred to as the Heun's method [10]) was developed to improve the approximated solution at t_{i+1} by taking the arithmetic average of the approximated solution at the slopes t_i and t_{i+1}.
The procedure for calculating the numerical solution to the (1.1) by first computing the Euler method to roughly estimate the coordinates of the next point in the solution, and then, the original estimate is recalculated using the rough estimate [4]:

$$\begin{aligned} \text{rough estimate: } & \tilde{y}_{i+1} = y_i + hf(t_i, y_i) \\ \text{original estimate: } & y_{i+1} = y_i + \frac{h}{2}\left[f(t_i, y_i) + f(t_{i+1}, \tilde{y}_{i+1})\right] \end{aligned} \tag{2.2}$$

where h is the step size an $t_{i+1} = t_i + h$.

The Truncation Errors

the local truncation error is $O(h^3)$. The modified Euler Method is second order accurate.

2.2.1 The Pseudocode

To approximate the solution of the IVP

$$y' = f(t, y), \;\; a \leq t \leq b, \;\; y(a) = \alpha$$

at N equally spaced numbers in the interval $[a, b]$:

Algorithm 2 :: Modified Euler's Method

Require: endpoints a, b; integer N; initial condition α
Ensure: approximation w to y at the N values of t

$h = (b - a)/N;\;\; t_0 = a;\;\; w_0 = \alpha$
for $i = 0, 1, 2, \cdots, N-1$ **do**
  $\tilde{w}_{i+1} = w_i + hf(t_i, w_i);$ {Compute rough (next) w_i}
  $w_{i+1} = w_i + \frac{h}{2}\left[f(t_i, w_i) + f(t_i + h, \tilde{w}_{i+1})\right];$ {Compute corrected (next) w_i}
  $t = a + ih$ {Compute next t_i}
end for
return (t, w)

2.3 The 2nd-Order Runge-Kutta Method

The Truncation Errors

2.3.1 The Pseudocode

To approximate the solution of the IVP

$$y' = f(t, y), \;\; a \le t \le b, \;\; y(a) = \alpha$$

at N equally spaced numbers in the interval $[a, b]$:

Algorithm 3 :: 2nd-Order Runge-Kutta Method

Require: endpoints a, b; integer N; initial condition α
Ensure: approximation w to y at the N values of t

$h = (b - a)/N;\;\; t_0 = a;\;\; w_0 = \alpha$
for $i = 0, 1, 2, \cdots, N - 1$ **do**
  $k_1 = f(t_i, w_i);$ $\{$Compute $k_1\}$
  $k_2 = f\left(t_i + \frac{h}{2}, w_i + \frac{h}{2}k_1\right);$ $\{$Compute $k_2\}$
  $w_{i+1} = w_i + hk_2;$ $\{$Compute next $w_i\}$
  $t = a + ih$ $\{$Compute next $t_i\}$
end for
return (t, w)

2.4 The 4th-Order Runge-Kutta Method

The Truncation Errors

2.4.1 The Pseudocode

To approximate the solution of the IVP

$$y' = f(t, y), \;\; a \leq t \leq b, \;\; y(a) = \alpha$$

at N equally spaced numbers in the interval $[a, b]$: [2]

Algorithm 4 :: 4th-Order Runge-Kutta Method

Require: endpoints a, b; integer N; initial condition α
Ensure: approximation w to y at the N values of t

$h = (b-a)/N$; $t_0 = a$; $w_0 = \alpha$
for $i = 0, 1, 2, \cdots, N-1$ **do**
  $k_1 = f(t_i, w_i)$; {Compute k_1 *to* k_4}
  $k_2 = f\left(t_i + \frac{h}{2}, w_i + \frac{h}{2}k_1\right)$;
  $k_3 = f\left(t_i + \frac{h}{2}, w_i + \frac{h}{2}k_2\right)$;
  $k_4 = f\left(t_i + h, w_i + hk_3\right)$;
  $K = k_1 + 2k_2 + 2k_3 + k_4$
  $w_{i+1} = w_i + \frac{h}{6}K$; {Compute next w_i}
  $t = a + ih$ {Compute next t_i}
end for
return (t, w)

2.5 The Adams-Bashforth 4th-Order Explicit Method

The Truncation Errors

2.5.1 The Pseudocode

To approximate the solution of the IVP

$$y' = f(t, y), \;\; a \leq t \leq b, \;\; y(a) = \alpha$$

at N equally spaced numbers in the interval $[a, b]$: [2]

2.6 The Adams 4th-Order Predictor-Corrector Method

The Truncation Errors

2.6.1 The Pseudocode

2.7 The Runge-Kutta-Fehlberg Method

The Truncation Errors

2.7.1 The Pseudocode

2.8 The Predictor-Corrector Method with variable step sizes

The Truncation Errors

2.8.1 The Pseudocode

Chapter 3

NUMERICAL EXPERIMENTS

REFERENCES

K. Atkinson. *An introduction to numerical analysis.* John wiley & sons, 1991.

R. L. Burden, J. D. Faires, and A. M. Burden. *Numerical analysis.* Cengage learning, 2015.

J. C. Butcher. *Numerical methods for ordinary differential equations.* John Wiley & Sons, 2016.

W.-f. Ch'en, D. De Kee, and P. N. Kaloni. Advanced mathematics for engineering and science. *(No Title).*

P. Dawkins. Differential equations. URL `https://tutorial.math.lamar.edu/classes/de/eulersmethod.aspx`. Accessed: 04-15-2023.

L. Euler. *Institutiones Calculi Integralis 2nd Part*, volume 1. Springer Science & Business Media, 1913.

K. King. How to solve initial value problems. URL `https://www.kristakingmath.com/blog/solving-initial-value-problems`. Accessed: 04-15-2023.

R. Melton. Initial-value problems. URL `https://courses.lumenlearning.com/calculus2/chapter/initial-value-problems`. Accessed: 04-15-2023.

wikipedia.org. Initial value problem, a. URL `https://en.wikipedia.org/wiki/Euler_method`. Accessed: 04-15-2023.

wikipedia.org. Heun's method, b. URL `https://en.wikipedia.org/wiki/Heun%27s_method`. Accessed: 04-15-2023.

wikipedia.org. Initial value problem, c. URL `https://en.wikipedia.org/wiki/Initial_value_problem`. Accessed: 04-15-2023.