COSC/MATH 4340: Numerical Methods for Differential Equations

Ordinary Differential Equations (ODEs): Initial Value Problems (IVPs)

Class: COSC/MATH 4340
Semester: Spring 2024

Instructor: Dane Taylor
Office: Eng 4087
Email: dane.taylor@uwyo.edu

Introduction

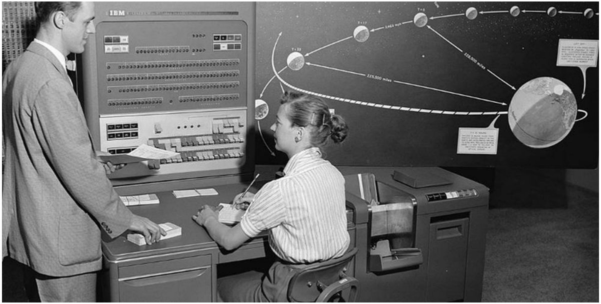
Researchers and engineers often seek to solve an ordinary differential equation (ODE)

$$\frac{dy}{dt} = f(t, y).$$

Applications arise in all fields and include simulating a fluid flow, modeling disease spreading, solving robotics movements, or designing AI algorithms.

Often, the equation is too large or difficult to solve by hand, and we instead rely on approximating the solution using a computer.

Introduction

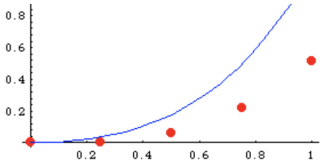


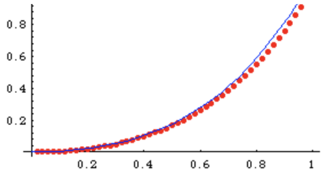
“Human computers” eventually replaced by electronic computers to predict, e.g., orbital motions.

Introduction

We can **numerically solve** $\frac{dy}{dt} = f(t, y)$ by obtaining a sequence of approximations $y_i \approx y(t_i)$ at times $t_i = t_0 + hi$, where $i = 0, 1, \ldots, N$ and $h = (T - t_0)/N$ is the time step.

Smaller time steps h yield more accurate approximations but are more time consuming.





Introduction

Definition

An **initial value problem (IVP)** aims to solve an ODE with a given initial condition

$$\begin{cases} \frac{dy}{dt} & = f(t, y) \\ y(t_0) & = y_0 \end{cases}$$

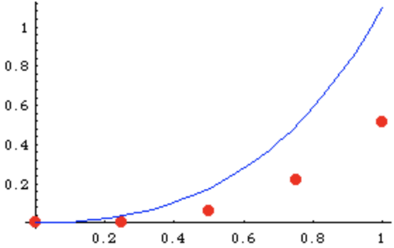
for a solution $y(t)$ over some range $t \in [t_0, T]$.

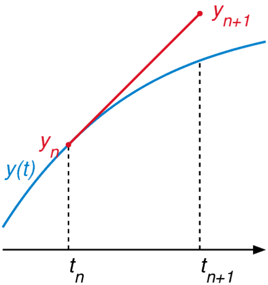
Definition

If $f(t, y) = f(t)$, i.e., it doesn't depend on y, then the ODE is called **autonomous**. Otherwise, it is **non-autonomous**

Forward Euler Method

The simplest numerical method to solve an IVP is **forward Euler**, which uses the slope $f(t_i, y_i)$ at y_i to predict y_{i+1} for each i





(See notes for derivation.)

Truncated Taylor Expansions

Consider the approximation of $y(t)$ near t_0

$$y(t) = y(t_0) + (t - t_0)y'(t_0) + \frac{(t-t_0)^2}{2}y''(t_0) + \frac{(t-t_0)^3}{3!}y'''(t_0) + \\ + \frac{(t-t_0)^4}{4!}y^{(4)}(t_0) + \dots$$

The intermediate value theorem implies that there exists a value $\xi \in [t_0, t]$ allowing truncation without error:

$$y(t) = y(t_0) + (t - t_0)y'(t_0) + \frac{(t-t_0)^2}{2}y''(\xi)$$

$$y(t) = y(t_0) + (t - t_0)y'(t_0) + \frac{(t-t_0)^2}{2}y''(t_0) + \frac{(t-t_0)^3}{2}y'''(\xi).$$

(See notes and code.)

Forward Euler

Definition

The **forward Euler method** approximately solves and IVP by $y_0 = y(t_0)$ and

$$y_{i+1} = y_i + hf(t_i, y_i), \quad i = 0, 1, 2, \ldots, N$$

with $h = (T - t_0)/N$ and $t_i = t_0 + hi$.

Forward Euler

Example: Numerically solve

$$\begin{cases} \frac{dy}{dt} &= f(t, y) = 2y \\ y(t_0) &= y_0 = 1 \end{cases}$$

over the range $t \in [0, 1]$ using the FE method with $N = 5$ steps.

(See notes and code.)

Existence, Uniqueness, and Stability

Before attempting to solve of approximate $y(t)$, we should know if the solution $y(t)$ exists and is unique. This requires a strong notion for what "continuous" means

Definition

A function $f(t, y)$ is **Lipschitz continuous** over a region $\mathcal{S} = [t_0, T] \times [\alpha, \beta]$ if there exists a **Lipschitz constant** $L > 0$ such that

$$|f(t, y_1) - f(t, y_2)| \leq L|y_1 - y_2|$$

for any $t \in [t_0, T]$ and $y_1, y_2 \in [\alpha, \beta]$.

Note that f varies continuously with variable y in a very specific (and boundable) way

Existence, Uniqueness, and Stability

Lipschitz continuity example with $f(t, y) = ty + t^3$.

(See notes.)

Existence, Uniqueness, and Stability

Lipschitz continuity allows for an existence and uniqueness result

Theorem

Assume $f(t, y)$ is Lipschitz continuous over a region $\mathcal{S} = [t_0, T] \times [\alpha, \beta]$. Then the IVP with initial condition $y(t_0) = y_0$ for some $y_0 \in [\alpha, \beta]$ has exactly one solution $y(t)$ over the range $t \in [t_0, c]$ for some $c \in [t_0, T]$. Moreover, if f is Lipschitz continuous over the region $\mathcal{S} = [t_0, T] \times [-\infty, \infty]$, then the IVP has exactly one solution over $t \in [t_0, T]$.

Existence, Uniqueness, and Stability

Lipschitz continuity allows for a stability result

Theorem

Assume $f(t, y)$ is Lipschitz continuous over a region $\mathcal{S} = [t_0, T] \times [\alpha, \beta]$ with Lipschitz constant L, and consider two solutions $Y(t)$ and $Z(t)$ that solve the ODE with initial conditions $Y(t_0) = Y_0$ and $Z(t_0) = Z_0$, then

$$|Y(t) - Z(t)| \leq e^{L(t-t_0)}|Y_0 - Z_0|$$

If $Y(t)$ and $Z(t)$ are two **trajectories** with two different initial conditions, then the difference between the trajectories exponentially grows with L

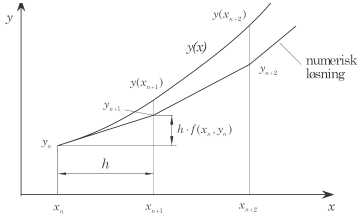
Local and Global Error

Definition

The **local truncation error** e_i for an IVP solver is the error the occurs at 1 step (assuming the previous step has zero error).

Definition

The **global error** E_i for an IVP solver is the error that accumulates across the iterated steps.



Local and Global Error

Recall the **forward Euler method** $y_{i+1} = y_i + hf(t_i, y_i)$

Theorem

The **local truncation error** e_i for the forward Euler method is second order, $e_i = \mathcal{O}(h^2)$, and if $|y''(t)| \leq M$ for some constant M, then $e_i \leq \frac{Mh^2}{2}$ for all i

Theorem

Considering an IVP with a Lipschitz function with Lipschitz constant L, the **global error** E_i for the forward Euler method is first order, $E_i = \mathcal{O}(h)$, and if $|y''(t)| \leq M$ for some constant M, then

$$E_i \leq \frac{Mh}{2L}(e^{Lhi} - 1)$$

(See notes for proof.)

Local and Global Error

Study error for earlier example: $f(t, y) = 2y(t)$.

(See notes and code.)