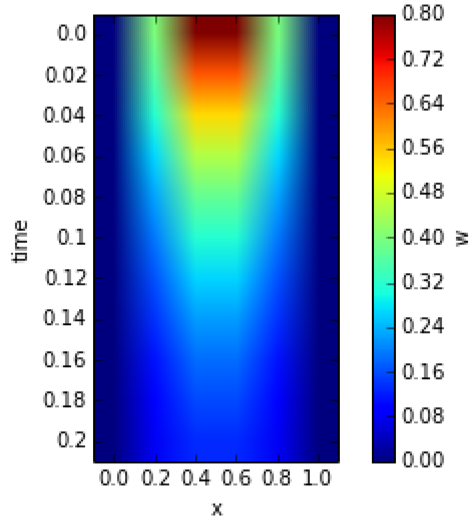
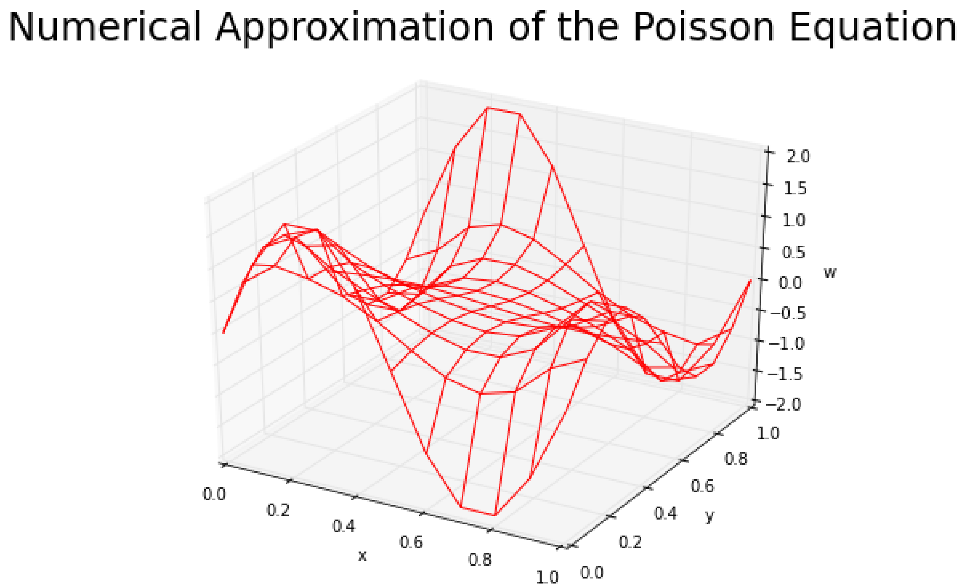
John S Butler



Numerical Methods and Machine Learning Methods for Differential Equations



CONTENTS

ACRONYMS

IVP Initial Value Problems

BVP Boundary Value Problems

ODE Ordinary Differential Equations

PDE Partial Differential Equations

RK Runge Kutta

OVERVIEW

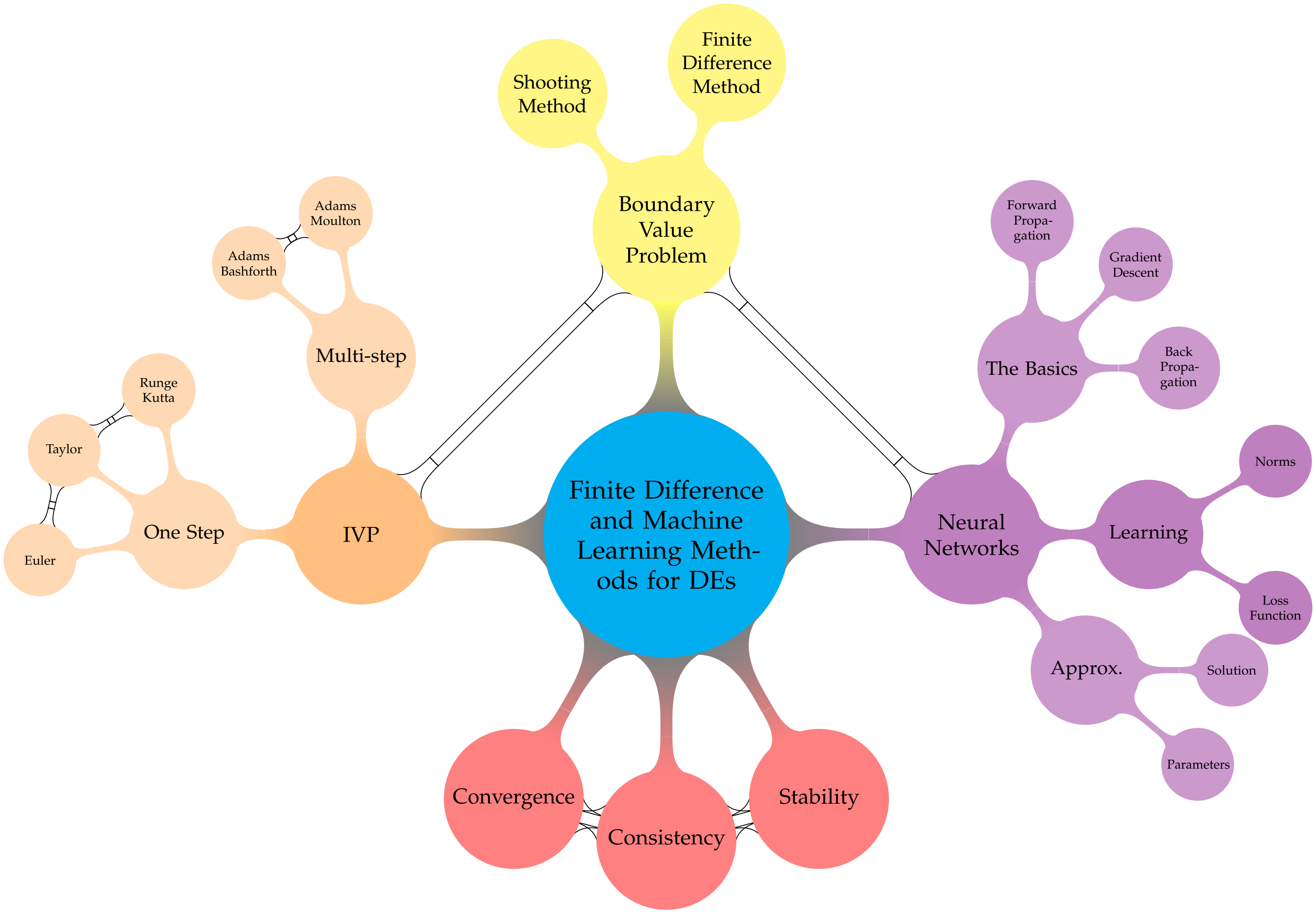
This book is divided into three parts;

- Numerical Solutions to Initial Value Differential Equations, Chapters 1-5.
- Numerical Solutions to Ordinary Differential Equations, Chapter 6.
- Machine Learning methods for Ordinary Differential Equations, Chapters ?-?.

The first part introduces numerical methods for initial value problems starting with one step methods (Chapters 1-3), moving into explicit and implicit numerical multi-step methods (Chapter 4). Chapter 5 brings together the concepts of consistency, convergence and stability.

Chapter 1 covers some simple theorems of initial value problems as well as the Euler Method. Chapter 2 introduces higher order methods using the Taylor method. These methods are then extend in Chapter 3 to Runge Kutta methods. Chapter 4 introduces the Adams-Bashforth and Adams-Moulton explicit and implicit multi-step methods. Chapter 5 discusses the consistency, convergence of one-step and multi-step methods.

All the notes are supplemented by iPython notebooks.


Finite Difference and Machine Learning Methods for DEs
IVP
One Step
Euler
Taylor
Runge Kutta
Multi-step
Adams Bashforth
Adams Moulton
Boundary Value Problem
Shooting Method
Finite Difference Method
Neural Networks
The Basics
Forward Propagation
Gradient Descent
Back Propagation
Learning
Norms
Loss Function
Approx.
Solution
Parameters
Convergence
Consistency
Stability

Part I

NUMERICAL SOLUTIONS TO ORDINARY DIFFERENTIAL EQUATIONS

Part II

INITIAL VALUE PROBLEMS

1

NUMERICAL SOLUTIONS TO INITIAL VALUE PROBLEMS

Differential equations have numerous applications to describe dynamics from physics to biology to economics.

Initial value problems are subset of Ordinary Differential Equation (ODE's) with the form

$$y' = f(x) \tag{1}$$

f is a function. The general solution to (1) is

$$y = \int f(x)dx + c,$$

containing an arbitrary constant c. In order to determine the solution uniquely it is necessary to impose an initial condition,

$$y(x_0) = y_0. \tag{2}$$

Example 1

Simple Example
The differential equation describes the rate of change of an oscillating input. The general solution of the equation

$$y' = \sin(x) \tag{3}$$

is,

$$y = -\cos(x) + c,$$

with the initial condition,

$$y(0) = 2,$$

then it is easy to find $c = 2$.
Thus the desired solution is,

$$y = 2 - \cos(x).$$

The more general Ordinary Differential Equation is of the form

$$y' = f(x, y), \tag{4}$$

is approached in a similar fashion.
Let us consider

$$y' = a(x)y(x) + b(x),$$

The given functions $a(x)$ and $b(x)$ are assumed continuous for this equation

$$f(x,z) = a(x)z(x) + b(x),$$

and the general solution can be found using the method of integrating factors.

Example 2

General Example
Differential equations of the form

$$y'(x) = \lambda y(x) + b(x), \quad x \geq x_0, \tag{5}$$

where λ is a given constant and $b(x)$ is a continuous integrable function has a unique analytic. Multiplying the equation (5) by the integrating factor $e^{-\lambda x}$, we can reformulate

$$\frac{d(e^{-\lambda x}y(x))}{dx} = e^{-\lambda x}b(x).$$

Integrating both sides from x_0 to x we obtain

$$(e^{-\lambda x}y(x)) = c + \int_{x_0}^{x} e^{-\lambda t}b(t)dt,$$

so the general solution is

$$y(x) = ce^{\lambda x} + \int_{x_0}^{x} e^{\lambda(x-t)}b(t)dt,$$

with c an arbitrary constant

$$c = e^{-\lambda x_0}y(x_0).$$

For a great number of Initial Value Problems there is no known exact (analytic) solution as the equations are non-linear, for example $y' = e^{xy^4}$, or discontinuous or stochastic. There for a numerical method is used to approximate the solution.

1.1 NUMERICAL APPROXIMATION OF DIFFERENTIATION

1.1.1 *Derivation of Forward Euler for one step*

The left hand side of a initial value problem $\frac{df}{dx}$ can be approximated by **Taylors theorem** expand about a point x_0 giving:

$$f(x_1) = f(x_0) + (x_1 - x_0)f^{'}(x_0) + \tau, \tag{6}$$

where τ is the truncation error,

$$h\tau = \frac{(x_1 - x_0)^2}{2!} f^{''}(\xi), \qquad \xi \in [x_0, x_1]. \tag{7}$$

Rearranging and letting $h = x_1 - x_0$ the equation becomes

$$f^{'}(x_0) = \frac{f(x_1) - f(x_0)}{h} - \frac{h}{2} f^{''}(\xi).$$

Term $-\frac{h}{2}f^{''}(\xi)$ is known as the local truncation error. The forward Euler method can also be derived using a variation on the Lagrange interpolation formula called the divided difference.
Any function $f(x)$ can be approximated by a polynomial of degree $P_n(x)$ and an error term,

$$\begin{aligned} f(x) &= P_n(x) + error, \\ &= f(x_0) + f[x_0, x_1](x - x_0) + f[x_0, x_1, x_2](x - x_0)(x - x_1), \\ &\quad +... + f[x_0, ..., x_n]\Pi_{i=0}^{n-1}(x - x_i) + error, \end{aligned}$$

where

$$f[x_0, x_1] = \frac{f(x_1) - f(x_0)}{x_1 - x_0},$$

$$f[x_0, x_1, x_2] = \frac{f[x_1, x_2] - f[x_0, x_1]}{x_2 - x_0},$$

$$f[x_0, x_1, .., x_n] = \frac{f[x_1, x_2, ..., x_n] - f[x_0, x_1, ..., x_{n-1}]}{x_n - x_0},$$

Differentiating $P_n(x)$

$$\begin{aligned} P_n^{'}(x) &= f[x_0, x_1] + f[x_0, x_1, x_2]\{(x - x_0) + (x - x_1)\}, \\ &\quad +... + f[x_0, ..., x_n] \sum_{i=0}^{n-1} \frac{(x - x_0)...(x - x_{n-1})}{(x - x_i)}, \end{aligned}$$

and the error becomes

$$error = (x - x_0)...(x - x_n)\frac{f^{n+1}(\xi)}{(n+1)!}.$$

Applying this to define our first derivative, we have

$$f^{'}(x) = f[x_0, x_1] = \frac{f(x_1) - f(x_0)}{x_1 - x_0},$$

this leads us other formulas for computing the derivatives

$$f^{'}(x) = \frac{f(x_1) - f(x_0)}{x_1 - x_0} + O(h), \quad \text{Euler},$$

$$f^{'}(x) = \frac{f(x_1) - f(x_{-1})}{x_1 - x_{-1}} + O(h^2), \quad \text{Central}.$$

Using the same method we can get out computational estimates for the 2nd derivative

$$f^{''}(x_0) = \frac{f_2 - 2f_1 + f_0}{h^2} + O(h^2),$$

$$f^{''}(x_0) = \frac{f_1 - 2f_0 + f_{-1}}{h^2} + O(h^2), \quad \text{central}.$$

Example 3

To numerically solve the first order Ordinary Differential Equation (4)

$$y^{'} = f(x, y),$$

$$a \leq x \leq b,$$

the derivative y' is approximated by

$$\frac{w_{i+1} - w_i}{x_{i+1} - x_i} = \frac{w_{i+1} - w_i}{h},$$

where w_i is the numerical approximation of y at x_i. The Differential Equation is converted to a discrete difference equation with steps of size h,

$$\frac{w_{i+1} - w_i}{h} = f(x_i, w).$$

Rearranging the difference equation gives the equation

$$w_{i+1} = w_i + hf(x_i, w),$$

which can be used to approximate the solution at w_{i+1} given information about y at point x_i.

1.1.1.1 *Simple example ODE* $y' = \sin(x)$

Example 4

Applying the Euler formula to the first order equation with an oscillating input (3)

$$y' = \sin(x),$$

$$0 \leq x \leq 10.$$

The equation can be approximated using the forward Euler as

$$\frac{w_{i+1} - w_i}{h} = \sin(x_i).$$

Rearranging the equation gives the discrete difference equation with the unknowns on the left and the know values of the right

$$w_{i+1} = w_i + h\sin(x_i).$$

The Python code bellow implements this difference equation. The output of the code is shown in Figure 1.1.1.

```python
# Numerical solution of a Cosine differential equation
import numpy as np
import math
import matplotlib.pyplot as plt

h=0.01
a=0
b=10

N=int(b-a/h)
w=np.zeros(N)
x=np.zeros(N)
Analytic_Solution=np.zeros(N)

# Initial Conditions
w[0]=1.0
x[0]=0
Analytic_Solution[0]=1.0
for i in range (1,N):
    w[i]=w[i-1]+h*math.sin(x[i-1])
    x[i]=x[i-1]+h
    Analytic_Solution[i]=2.0-math.cos(x[i])

fig = plt.figure(figsize=(8,4))

# --- left hand plot
ax = fig.add_subplot(1,3,1)
plt.plot(x,w,color='red')
#ax.legend(loc='best')
plt.title('Numerical Solution')
```

```

# --- right hand plot
ax = fig.add_subplot(1,3,2)
plt.plot(x,Analytic_Solution,color='blue')
plt.title('Analytic Solution')

#ax.legend(loc='best')
ax = fig.add_subplot(1,3,3)
plt.plot(x,Analytic_Solution-w,color='blue')
plt.title('Error')

# --- title, explanatory text and save
fig.suptitle('Sine Solution', fontsize=20)
plt.tight_layout()
plt.subplots_adjust(top=0.85)
```

Listing 1.1: Python Numerical and Analytical Solution of Eqn 3

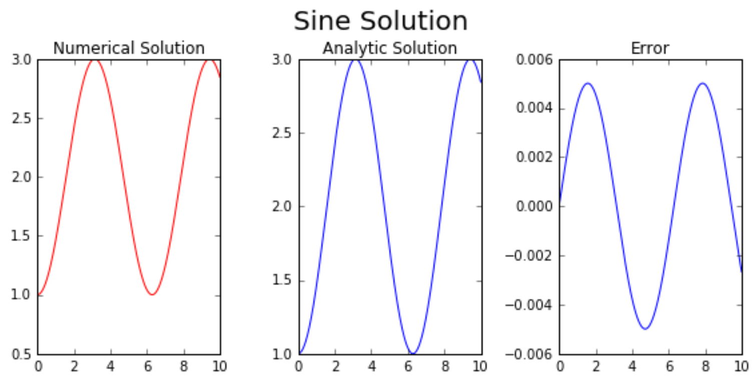


Figure 1.1.1: Python output: Numerical (left), Analytic (middle) and error(right) for $y' = \sin(x)$ Equation 3 with h=0.01

1.1.1.2 *Simple example problem population growth $y^{'} = \varepsilon y$.*

Example 5

Simple population growth can be describe as a first order differential equation of the form:

$$y^{'} = \varepsilon y. \tag{8}$$

This has an exact solution of

$$y = Ce^{\varepsilon x}.$$

Given the initial condition of condition

$$y(0) = 1$$

and a rate of change of

$$\varepsilon = 0.5$$

the analytic solution is

$$y = e^{0.5x}.$$

Example 6

Applying the Euler formula to the first order equation (8)

$$y^{'} = 0.5y$$

is approximated by

$$\frac{w_{i+1} - w_i}{h} = 0.5w_i.$$

Rearranging the equation gives the difference equation

$$w_{i+1} = w_i + h(0.5w_i).$$

The Python code below and the output is plotted in Figure 1.1.2.

```
# Numerical solution of a differential equation
import numpy as np
import math
import matplotlib.pyplot as plt

h=0.01
tau=0.5
a=0
b=10

N=int((b-a)/h)
w=np.zeros(N)
x=np.zeros(N)
Analytic_Solution=np.zeros(N)

Numerical_Solution[0]=1
x[0]=0
w[0]=1

for i in range (1,N):
    w[i]=w[i-1]+dx*(tau)*w[i-1]
    x[i]=x[i-1]+dx
    Analytic_Solution[i]=math.exp(tau*x[i])


fig = plt.figure(figsize=(8,4))
# --- left hand plot
ax = fig.add_subplot(1,3,1)
```

```python
plt.plot(x,w,color='red')
#ax.legend(loc='best')
plt.title('Numerical Solution')

# --- right hand plot
ax = fig.add_subplot(1,3,2)
plt.plot(x,Analytic_Solution,color='blue')
plt.title('Analytic Solution')

#ax.legend(loc='best')
ax = fig.add_subplot(1,3,3)
plt.plot(x,Analytic_Solution-Numerical_Solution,
    color='blue')
plt.title('Error')

# --- title, explanatory text and save
fig.suptitle('Exponential Growth Solution',
    fontsize=20)
plt.tight_layout()
plt.subplots_adjust(top=0.85)
```

Listing 1.2: Python Numerical and Analytical Solution of Eqn 8

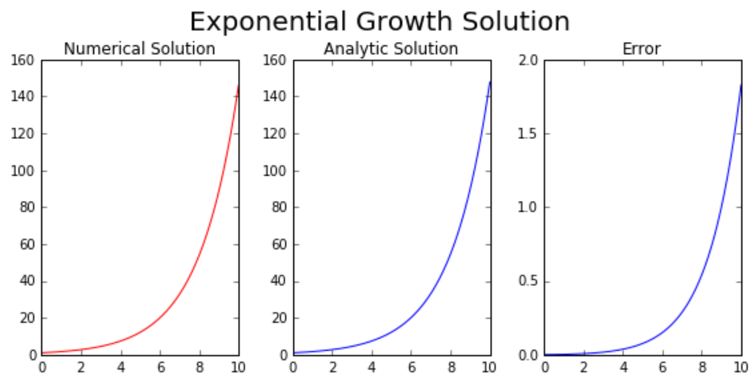


Figure 1.1.2: Python output: Numerical (left), Analytic (middle) and error(right) for $y' = \varepsilon y$ Eqn 8 with h=0.01 and $\varepsilon = 0.5$

1.1.1.3 *Example of exponential growth with a wiggle*

Example 7

An extension of the exponential growth differential equation includes a sinusoidal component

$$y' = \varepsilon(y + y\sin(x)). \tag{9}$$

This complicates the exact solution but the numerical approach is more or less the same. The difference equation is

$$w_{i+1} = w_i + h(0.5w_i + w_i \sin(x_i)).$$

Figure 1.1.3 illustrates the numerical solution of the differential equation.

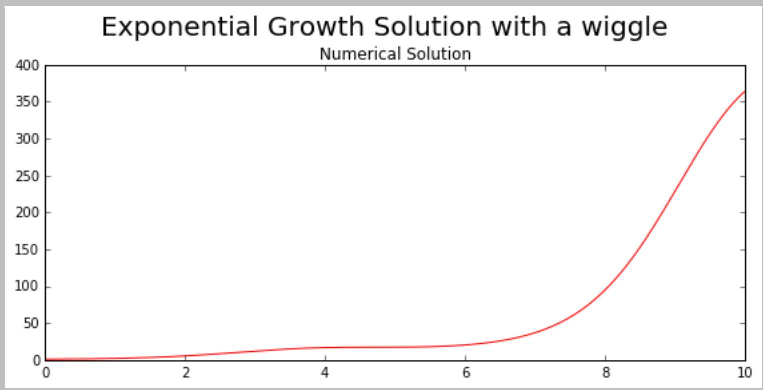


Figure 1.1.3: Python output: Numerical solution for $y' = \varepsilon(y + ysin(x))$ Equation 9 with h=0.01 and $\varepsilon = 0.5$

1.1.2 *Theorems about Ordinary Differential Equations*

Definition A function $f(t,y)$ is said to satisfy a **Lipschitz Condition** in the variable y on the set $D \subset R^2$ if a constant $L > 0$ exist with the property that

$$|f(t,y_1) - f(t,y_2)| < L|y_1 - y_2|,$$

whenever $(t,y_1),(t,y_2) \in D$. The constant L is call the Lipschitz Condition of f.

Definition A set $D \subset R^2$ is said to be convex if whenever $(t_1,y_1),(t_2,y_2)$ belong to D the point $((1-\lambda)t_1 + \lambda t_2, (1-\lambda)y_1 + \lambda y_2)$ also belongs in D for each $\lambda \in [0,1]$.

Theorem 1.1.1. *Suppose $f(t,y)$ is defined on a convex set $D \subset R^2$. If a constant $L > 0$ exists with*

$$L \leq \left| \frac{\partial f(t,y)}{\partial y} \right|,$$

then f satisfies a Lipschitz Condition an D in the variable y with Lipschitz constant L.

Theorem 1.1.2. *Suppose that $D = \{(t,y) | a \leq t \leq b, -\infty < y < \infty\}$, and $f(t,y)$ is continuous on D in the variable y then the initial value problem has a unique solution $y(t)$ for $a \leq t \leq b$.*

Definition The initial-value problem

$$\frac{dy}{dt} = f(t,y), \quad a \leq t \leq b,$$

with initial condition

$$y(a) = \alpha,$$

is said to be well posed if:

- A unique solution $y(t)$ to the problem exists;
- For any $\varepsilon > 0$ there exists a positive constant $k(\varepsilon)$ with the property that whenever $|\varepsilon_0| < \varepsilon$ and with $|\delta(t)|\varepsilon$ on $[a,b]$ a unique solution $z(t)$ to the problem

$$\frac{dz}{dt} = f(t,z) + \delta(t), \quad a \leq t \leq b, \tag{10}$$

$$z(a) = \alpha + \varepsilon_0,$$

exists with

$$|z(t) - y(t)| < k(\varepsilon)\varepsilon.$$

The problem specified by (10) is called a perturbed problem associated with the original problem.

It assumes the possibility of an error $\delta(\varepsilon)$ being introduced to the statement of the differential equation as well as an error ε_0 being present in the initial condition.

Theorem 1.1.3. *Suppose $D = \{(t,y)|a \leq t \leq b, -\infty < y < \infty\}$. If $f(t,y)$ is continuous and satisfies a Lipschitz Condition in the variable y on the set D, then the initial value problem*

$$\frac{dy}{dt} = f(t,y), \quad a \leq t \leq b,$$

with initial condition

$$y(a) = \alpha,$$

is well-posed.

Example 8

$$y'(x) = -y(x) + 1, \quad 0 \leq x \leq b, \quad y(0) = 1$$

has the solution $y(x) = 1$. The perturbed problem

$$z'(x) = -z(x) + 1, \quad 0 \leq x \leq b, \quad z(0) = 1 + \varepsilon,$$

has the solution $z(x) = 1 + \varepsilon e^{-x}$ $x \leq 0$.
Thus

$$y(x) - z(x) = -\varepsilon e^{-x}$$

$$|y(x) - z(x)| \leq |\varepsilon| \quad x \geq 0$$

Therefore the problem is said to be stable. This is illustrated in Figure 1.1.4.

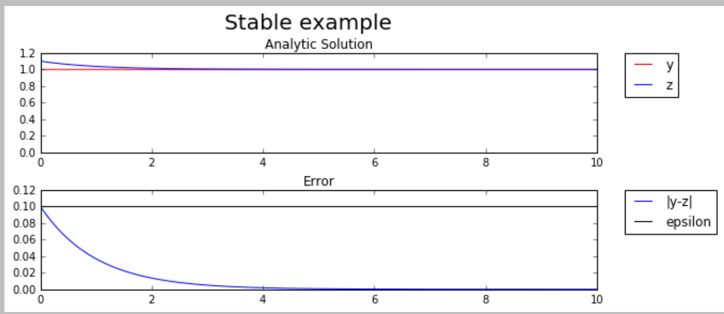


Figure 1.1.4: Python output: Illustrating Stability $y'(x) = -y(x) + 1$ with the initial condition $y(0) = 1$ and $z'(x) = -z(x) + 1$ with the initial condition $z(0) = 1 + \varepsilon$, $\varepsilon = 0.1$

1.2 ONE-STEP METHODS

Dividing $[a, b]$ in to N subsections such that we now have N+1 points of equal spacing $h = \frac{b-a}{N}$. This gives the formula $t_i = a + ih$ for $i = 0, 1, ..., N$. One-Step Methods for Ordinary Differential Equation's only use one previous point to get the approximation for the next point. The initial condition gives $y(a = t_0) = \alpha$, this gives the starting point of our one step method. The general formula for One-step methods is

$$w_{i+1} = w_i + h\Phi(t_i, w_i, h),$$

where w_i is the approximated solution of the Ordinary Differential Equation at the point t_i

$$w_i \approx y_i.$$

1.2.1 *Euler's Method*

The simplest example of a one step method is Euler. The derivative is replaced by the Euler approximation. The Ordinary Differential Equation

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

is discretised

$$\frac{y_i - y_{i-1}}{h} = f(t_{i-1}, y_{i-1}) + T$$

T is the truncation error.

Example 9

Consider the Initial Value Problem

$$y^{'} = -\frac{y^2}{1+t}, \quad a = 0 \le t \le b = 0.5,$$

with the initial condition $y(0) = 1$ the Euler approximation is

$$w_{i+1} = w_i - \frac{hw_i^2}{1+t_i},$$

where w_i is the approximation of y at t_i.
Solving, let $t_i = ih$ where $h = 0.05$, from the initial condition we have $w_0 = 1$ at $i = 0$ our method is

$$w_1 = w_0 - \frac{0.05w_0^2}{1+t_0} = 1 - \frac{0.05}{1+0} = 0.95$$

and so forth, each approximation w_i requiring the previous, thus creating a sequence starting at w_0 to w_n. The Table below show the numerical approximation for 10 steps.

i	t_i	w_i
0	0	1
1	0.05	0.95
2	0.1	0.90702381
3	0.15	0.86962871
4	0.2	0.8367481
5	0.25	0.80757529
6	0.3	0.78148818
7	0.35	0.7579988
8	0.4	0.73671872
9	0.45	0.71733463

Lemma 1.2.1. *For all $x \geq 0.1$ and any positive m we have*

$$0 \le (1+x)^m \le e^{mx}.$$

Lemma 1.2.2. *If s and t are positive real numbers $\{a_i\}_{i=0}^{N}$ is a sequence satisfying $a_0 \geq \frac{-q}{s}$ and $a_{i+1} \leq (1+s)a_i + q$ then,*

$$a_{i+1} \le e^{(i+1)s}\left(a_0 + \frac{q}{s}\right) - \frac{q}{s}.$$

Theorem 1.2.3. *Suppose f is continuous and satisfies a Lipschitz Condition with constant L on $D = \{(t, y)|a \leq t \leq b, -\infty < y < \infty\}$ and that a constant M exists with the property that*

$$|y''(t)| \leq M.$$

Let $y(t)$ denote the unique solution of the Initial Value Problem

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

and $w_0, w_1, ..., w_N$ be the approx generated by the Euler method for some positive integer N. Then for $i = 0, 1, ..., N$

$$|y(t_i) - w_i| \leq \frac{Mh}{2L}|e^{L(t_i - a)} - 1|.$$

Proof. When $i = 0$ the result is clearly true since $y(t_0) = w_0 = \alpha$. From Taylor we have,

$$y(t_{i+1}) = y(t_i) + hf(t_i, y(t_i)) + \frac{h^2}{2}y''(\xi_i),$$

where $x_i \geq \xi_i \geq x_{i+1}$, and from this we get the Euler approximation

$$w_{i+1} = w_i + hf(t_i, w_i).$$

Consequently we have

$$y(t_{i+1}) - w_{i+1} = y(t_i) - w_i + h[f(t_i, y(t_i)) - f(t_i, w_i)] + \frac{h^2}{2}y''(\xi_i),$$

and

$$|y(t_{i+1}) - w_{i+1}| \leq |y(t_i) - w_i| + h|f(t_i, y(t_i)) - f(t_i, w_i)| + \frac{h^2}{2}|y''(\xi_i)|.$$

Since f satisfies a Lipschitz Condition in the second variable with constant L and $|y''| \leq M$ we have

$$|y(t_{i+1}) - w_{i+1}| \leq (1 + hL)|y(t_i) - w_i| + \frac{h^2}{2}M.$$

Using Lemma 1.2.1 and 1.2.2 and letting $a_j = (y_j - w_j)$ for each $j = 0, .., N$ while $s = hL$ and $q = \frac{h^2M}{2}$ we see that

$$|y(t_{i+1} - w_{i+1}| \leq e^{(i+1)hL}(|y(t_0) - w_0| + \frac{h^2M}{2hL}) - \frac{h^2M}{2hL}.$$

Since $w_0 - y_0 = 0$ and $(i + 1)h = t_{i+1} - t_0 = t_{i+1} - a$ we have

$$|y(t_i) - w_i| \leq \frac{Mh}{2L}|e^{L(t_i - a)} - 1|,$$

for each $i = 0, 1, ..N-1$. □

Example 10

$$y^{'} = y - t^2 + 1, \quad 0 \leq t \leq 2, \quad y(0) = 0.5,$$

the Euler approximation is

$$w_{i+1} = w_i + h(w_i - t_i^2 + 1)$$

choosing $h = 0.2$, $t_i = 0.2i$ and $w_0 = 0.5$.
$f(t, y) = y - t^2 + 1$

$$\frac{\partial f}{\partial y} = 1$$

so $L = 1$. The exact solution is $y(t) = (t+1)^2 - \frac{1}{2}e^t$ from this we have

$$y^{''}(t) = 2 - 0.5e^t,$$

$$|y^{''}(t)| \leq 0.5e^2 - 2, \quad t \in [0, 2].$$

Using the above inequality we have we have

$$|y_i - w_i| \leq \frac{h}{2}(0.5e^2 - 2)(e^{t_i} - 1).$$

Figure 1.2.1 illustrates the upper bound of the error and the actual error.

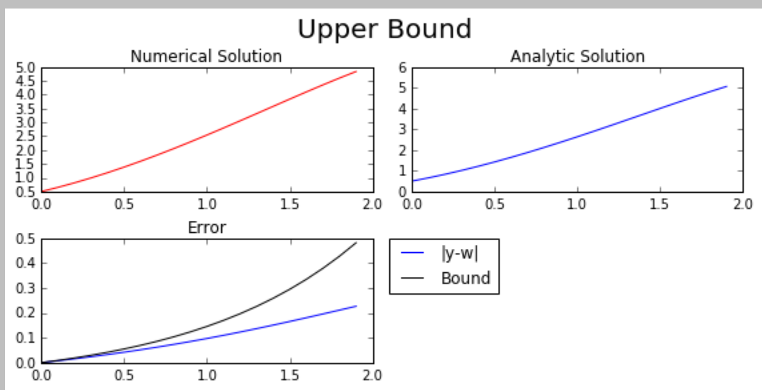


Figure 1.2.1: Python output: Illustrating upper bound $y^{'} = y - t^2 + 1$ with the initial condition $y(0) = 0.5$

Euler method is a typical one step method, in general such methods are given by function $\Phi(t, y; h; f)$. Our initial condition is $w_0 = y_0$, for $i = 0, 1, ..$

$$w_{i+1} = w_i + h\Phi(t_i, w_i : h : f)$$

with $t_{i+1} = t_i + h$.
In the Euler case $\Phi(t, y; h; f) = f(t, y)$ and is of order 1.
Theorem 1.2.3 can be extend to higher order one step methods with the variation

$$|y(t_i) - w_i| \leq \frac{Mh^p}{2L}|e^{L(t_i - a)} - 1|$$

where p is the order of the method.

Definition The difference method $w_0 = \alpha$

$$w_{i+1} = w_i + h\Phi(t_i, w_i),$$

for $i = 0, 1, ..., N - 1$ has a local truncation error given by

$$\begin{aligned} \tau_{i+1}(h) &= \frac{y_{i+1} - (y_i + h\Phi(t_i, y_i))}{h}, \\ &= \frac{y_{i+1} - y_i}{h} - \Phi(t_i, y_i), \end{aligned}$$

for each $i = 0, .., N - 1$ where as usual $y_i = y(t_i)$ denotes the exact solution at t_i.

For Euler method the local truncation error at the ith step for the problem

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

is

$$\tau_{i+1}(h) = \frac{y_{i+1} - y_i}{h} - f(t_i, y_i),$$

for $i = 0, .., N - 1$.
But we know Euler has

$$\tau_{i+1} = \frac{h}{2} y''(\xi_i), \quad \xi_i \in (t_i, t_{i+1}),$$

When $y''(t)$ is known to be bounded by a constant M on $[a, b]$ this implies

$$|t_{i+1}(h)| \leq \frac{h}{2} M \sim O(h).$$

$O(h)$ indicates a linear order of error. The higher the order the more accurate the method.

1.3 PROBLEM SHEET

1. Show that the following functions satisfy the Lipschitz condition on y on the indicated set D:
   a) $f(t,y) = ty^3$, $D = \{(t,y); -1 \leq t \leq 1, 0 \leq y \leq 10\}$;
   b) $f(t,y) = \frac{t^2y^2}{1+t^2}$, $D = \{(t,y); 0 \leq t, -10 \leq y \leq 10\}$.

2. Apply Euler's Method to approximate the solution of the given initial value problems using the indicated number of time steps. Compare the approximate solution with the given exact solution, and compare the actual error with the theoretical error
   a) $y' = t - y, \quad (0 \leq t \leq 4)$
   with the initial condition $y(0) = 1$,
   $N = 4$, $y(t) = 2e^{-t} + t - 1$,

   The Lipschitz constant is determined on $D = \{(t,y); 0 \leq t \leq 4, y \in \mathbb{R}\}$.
   b) $y' = y - t, \quad (0 \leq t \leq 2)$
   with the initial condition $y(0) = 2$,
   $N = 4$, $y(t) = e^t + t + 1$.

   The Lipschitz constant is determined on $D = \{(t,y); 0 \leq t \leq 2, y \in \mathbb{R}\}$.

2

HIGHER ORDER METHODS

2.1 HIGHER ORDER TAYLOR METHODS

The Taylor expansion

$$y(t_{i+1}) = y(t_i) + hy^{'}(t_i) + \frac{h^2}{2}y^{'}(t_i) + .. + \frac{h^{n+1}}{(n+1)!}y^{n+1}(\xi_i)$$

can be used to design more accurate higher order methods. By differentiating the original Ordinary Differential Equation $y^{'} = f(t, y)$ higher ordered can be derived method it requires the function to be continuous and differentiable.
In the general case of Taylor of order n:

$$w_0 = \alpha$$

$$w_{i+1} = w_i + hT^n(t_i, w_i), \text{ for } \ i = 0, ..., N-1,$$

where

$$T^n(t_i, w_i) = f(t_i, w_i) + \frac{h}{2}f'(t_i, w_i) + ... \frac{h^{n-1}}{n!}f^{n-1}(t_i, w_i). \qquad (11)$$

Example 11

Applying the general Taylor method to create methods of order two and four to the initial value problem

$$y^{'} = y - t^2 + 1, \quad 0 \leq t \leq 2, \quad y(0) = 0.5,$$

from this we have

$$f^{'}(t, y(t)) = \frac{d}{dt}(y - t^2 + 1) = y' - 2t = y - t^2 + 1 - 2t,$$

$$f^{''}(t, y(t)) = y - t^2 - 2t - 1,$$
$$f^{'''}(t, y(t)) = y - t^2 - 2t - 1.$$

From these derivatives we have

$$\begin{aligned} T^2(t_i,w_i) &= f(t_i,w_i)+\frac{h}{2}f'(t_i,w_i)\\ &= w_i-t_i^2+1+\frac{h}{2}(w_i-t_i^2-2t_i+1)\\ &= \left(1+\frac{h}{2}\right)(w_i-t_i^2-2t_i+1)-ht_i \end{aligned}$$

and

$$\begin{aligned} T^4(t_i,w_i) &= f(t_i,w_i)+\frac{h}{2}f'(t_i,w_i)\\ &\quad +\frac{h^2}{6}f''(t_i,w_i)+\frac{h^3}{24}f'''(t_i,w_i)\\ &= \left(1+\frac{h}{2}+\frac{h^2}{6}+\frac{h^3}{24}\right)(w_i-t_i^2)\\ &\quad -\left(1+\frac{h}{3}+\frac{h^2}{12}\right)ht_i\\ &\quad +1+\frac{h}{2}-\frac{h^2}{6}-\frac{h^3}{24} \end{aligned}$$

From these equations we have, Taylor of order two

$$w_0=0.5$$

$$w_{i+1}=w_i+h\left[\left(1+\frac{h}{2}\right)(w_i-t_i^2-2t_i+1)-ht_i\right]$$

and Taylor of order 4

$$\begin{aligned} w_{i+1} &= w_i+h\left[\left(1+\frac{h}{2}+\frac{h^2}{6}+\frac{h^3}{24}\right)(w_i-t_i^2)\right.\\ &\quad \left.-\left(1+\frac{h}{3}+\frac{h^2}{12}\right)ht_i+1+\frac{h}{2}-\frac{h^2}{6}-\frac{h^3}{24}\right] \end{aligned}$$

The local truncation error for the 2nd order method is

$$\tau_{i+1}(h)=\frac{y_{i+1}-y_i}{h}-T^2(t_i,y_i)=\frac{h^2}{6}f^2(\xi_i,y(x_i))$$

where $\xi\in(t_i,t_{i+1})$.

In general if $y\in C^{n+1}[a,b]$

$$\tau_{i+1}(h)=\frac{h^n}{(n+1)!}f^n(\xi_i,y(\xi_i))\;O(h^n).$$

The issue is that for every differential equation a new method has be to derived.

2.2 Problem Sheet 2 - Higher Order Methods (Taylor)

1. Apply 2nd Order Taylor Method to approximate the solution of the given initial value problems using the indicated number of time steps. Compare the approximate solution with the given exact solution, and compare the actual error with the theoretical local and global error

   a) $y' = t - y, \quad (0 \leq t \leq 4)$
   with the initial condition $y(0) = 1$,
   $N = 4$, $y(t) = 2e^{-t} + t - 1$,

   The Lipschitz constant is determined on $D = \{(t, y); 0 \leq t \leq 4, y \in \mathbb{R}\}$.

   b) $y' = y - t, \quad (0 \leq t \leq 2)$
   with the initial condition $y(0) = 2$,
   $N = 4$, $y(t) = e^{t} + t + 1$.

   The Lipschitz constant is determined on $D = \{(t, y); 0 \leq t \leq 2, y \in \mathbb{R}\}$.

2. Apply 3rd Order Taylor Method to approximate the solution of the given initial value problems using the indicated number of time steps. Compare the approximate solution with the given exact solution, and compare the actual error with the theoretical local and global error

   a) $y' = t - y, \quad (0 \leq t \leq 4)$
   with the initial condition $y(0) = 1$,
   $N = 4$, $y(t) = 2e^{-t} + t - 1$,

   The Lipschitz constant is determined on $D = \{(t, y); 0 \leq t \leq 4, y \in \mathbb{R}\}$.

   b) $y' = y - t, \quad (0 \leq t \leq 2)$
   with the initial condition $y(0) = 2$,
   $N = 4$, $y(t) = e^{t} + t + 1$.

   The Lipschitz constant is determined on $D = \{(t, y); 0 \leq t \leq 2, y \in \mathbb{R}\}$.

3. Apply the Taylor method to approximate the solution of initial value problem

$$y' = ty + ty^2, \quad (0 \leq t \leq 2), \quad y(0) = \frac{1}{2}$$

   using $N = 4$ steps.

3

RUNGE-KUTTA METHOD

The Runge-Kutta method (RK) method is closely related to the Taylor series expansions but no differentiation of f is necessary.
All RK methods will be written in the form

$$w_{n+1} = w_n + hF(t, w, h; f), \quad n \geq 0. \tag{12}$$

The truncation error for (12) is defined by

$$T_n(y) = y(t_{n+1}) - y(t_n) - hF(t_n, y(t_n), h; f)$$

where the error is written as $\tau_n(y)$

$$T_n = h\tau_n(y).$$

Rearranging we get

$$y(t_{n+1}) = y(t_n) - hF(t_n, y(t_n), h; f) + h\tau_n(y).$$

Theorem 3.0.1. *Suppose f(t,y) and all its partial derivatives of order less than or equal to n+1 are continuous on* $D = \{(t, y) | a \leq t \leq b, c \leq y \leq d\}$ *and let* $(t_0, y_0) \in D$ *for every* $(t, y) \in D$, $\exists\ \xi \in (t, t_0)$ *and* $\mu \in (y, y_0)$ *with*

$$f(t, y) = P_n(t, y) + R_n(t, y)$$

where

$$\begin{aligned} P_n(t,y) \quad = \quad & f(t_0, y_0) + \left[(t - t_0)\frac{\partial f}{\partial t}(t_0, y_0) + (y - y_0)\frac{\partial f}{\partial y}(t_0, y_0)\right] \\ & + \left[\frac{(t - t_0)^2}{2}\frac{\partial^2 f}{\partial t^2}(t_0, y_0) + (y - y_0)(t - t_0)\frac{\partial^2 f}{\partial y \partial t}(t_0, y_0)\right. \\ & \left. + \frac{(y - y_0)^2}{2}\frac{\partial^2 f}{\partial y^2}(t_0, y_0)\right] \\ & + \ldots + \\ & + \left[\frac{1}{n!}\sum_{j=0}^{n}\left(\begin{array}{c} n \\ j \end{array}\right)(t - t_0)^{n-j}(y - y_0)^j\frac{\partial^n f}{\partial y^j \partial t^{n-j}}(t_0, y_0)\right] \end{aligned}$$

and

$$R_n(t,y) = \left[\frac{1}{(n+1)!}\sum_{j=0}^{n+1}\left(\begin{array}{c} n+1 \\ j \end{array}\right)(t-t_0)^{n+1-j}(y-y_0)^j \frac{\partial^{n+1} f}{\partial y^j \partial t^{n+1-j}}(\xi,\mu)\right]$$

3.1 DERIVATION OF SECOND ORDER RUNGE-KUTTA

Consider the explicit one-step method

$$\frac{w_{i+1}-w_i}{h} = F(f,t_i,w_i,h) \tag{13}$$

with

$$F(f,t,y,h) = a_0 k_1 + a_1 k_2, \tag{14}$$

$$F(f,t,y,h) = a_0 f(t,y) + a_1 f(t+\alpha_1, y+\beta_1), \tag{15}$$

where $a_0 + a_1 = 1$.
There is a free parameter in the derivation of the Runge-Kutta method for this reason a_0 must be choosen

Deriving the second order Runge-Kutta method by using Theorem 3.0.1 to determine values for values a_1, α_1 and β_1 with the property that $a_1 f(t+\alpha_1, y+\beta_1)$ approximates the second order Taylor

$$f(t,y) + \frac{h}{2}f'(t,y)$$

with error no greater than than $O(h^2)$, the local truncation error for the Taylor method of order two.
Using

$$f'(t,y) = \frac{\partial f}{\partial t}(y,t) + \frac{\partial f}{\partial y}(t,y).y'(t),$$

the second order Taylor can be re-written as

$$f(t,y) + \frac{h}{2}\frac{\partial f}{\partial t}(y,t) + \frac{h}{2}\frac{\partial f}{\partial y}(t,y).f(t,y). \tag{16}$$

Expanding $a_1 f(t+\alpha_1, y+\beta_1)$ in its Taylor polynomial of degree one about (t,y) gives

$$a_1 f(t+\alpha_1, y+\beta_1) = a_1 f(t,y) + a_1\alpha_1\frac{\partial f}{\partial t}(t,y) + a_1\beta_1\frac{\partial f}{\partial y} + a_1 R_1(t+\alpha_1, y+\beta_1) \tag{17}$$

where

$$R_1(t+\alpha_1, y+\beta_1) = \frac{\alpha_1^2}{2}\frac{\partial^2 f}{\partial t^2}(\xi,\mu) + \alpha_1\beta_1\frac{\partial^2 f}{\partial t \partial y}(\xi,\mu) + \frac{\beta_1^2}{2}\frac{\partial^2 f}{\partial y^2}(\xi,\mu),$$

for some $\xi \in [t, t+\alpha_1]$ and $\mu \in [y, y+\beta_1]$.
Matching the coefficients and its derivatives in eqns (16) and (17) gives the equations

$$f(t,y) : a_1 = 1$$

$$\frac{\partial f}{\partial t}(t,y) : a_1\alpha_1 = \frac{h}{2}$$

and

$$\frac{\partial f}{\partial y}(t,y) : a_1\beta_1 = \frac{h}{2}f(t,y).$$

3.1.1 *Runge-Kutta second order: Midpoint method*

Choosing $a_0 = 0$ gives the unique values $a_1 = 1$, $\alpha_1 = \frac{h}{2}$ and $\beta_1 = \frac{h}{2}f(t,y)$ so

$$T^2(t,y) = f(t+\frac{h}{2}, y+\frac{h}{2}f(t,y)) - R_1(t+\frac{h}{2}, y+\frac{h}{2}f(t,y))$$

and from

$$R_1(t+\frac{h}{2}, y+\frac{h}{2}f(t,y)) = \frac{h^2}{8}\frac{\partial^2 f}{\partial t^2}(\xi,\mu) + \frac{h^2}{4}\frac{\partial^2 f}{\partial t \partial y}(\xi,\mu) + \frac{h^2}{8}g(t,y)^2\frac{\partial^2 f}{\partial y^2}(\xi,\mu),$$

for some $\xi \in [t, t+\frac{h}{2}]$ and $\mu \in [y, y+\frac{h}{2}f(t,y)]$.
If all the second-order partial derivatives are bounded then

$$R_1(t+\frac{h}{2}, y+\frac{h}{2}f(t,y)) \sim O(h^2).$$

The Midpoint second order Runge-Kutta for the initial value problem

$$y' = f(t,y)$$

with the initial condition $y(t_0) = \alpha$ is given by

$$w_0 = \alpha,$$

$$w_{i+1} = w_i + hf(t_i + \frac{h}{2}, y_i + \frac{h}{2}f(t_i, w_i)),$$

with an error of order $O(h^2)$. The Figure 3.3.2 illustrates the solution to the $y' = -xy$

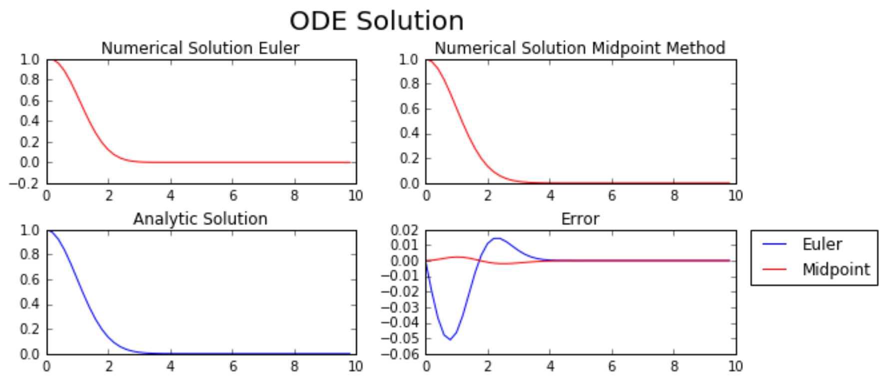


Figure 3.1.1: Python output: Illustrating upper bound $y' = -xy$ with the initial condition $y(0) = 1$

for each $i = 0, 1, ...N - 1$.

3.1.2 *2nd Order Runge-Kutta $a_0 = 0.5$: Heun's method*

Choosing $a_0 = 0.5$ gives the unique values $a_1 = 0.5$, $\alpha_1 = h$ and $\beta_1 = hf(t, y)$ such that

$$T^2(t,y) = F(t,y) = 0.5f(t,y) + 0.5f(t+h, y+hf(t,y)) - R_1(t+h, y+hf(t,y))$$

and the error value from

$$R_1(t+h, y+hf(t,y)) = \frac{h^2}{2}\frac{\partial^2 f}{\partial t^2}(\xi,\mu) + h^2\frac{\partial^2 f}{\partial t \partial y}(\xi,\mu) + \frac{h^2}{2}f(t,y)^2\frac{\partial^2 f}{\partial y^2}(\xi,\mu),$$

for some $\xi \in [t, t+h]$ and $\mu \in [y, y+hf(t,y)]$.

Thus Heun's second order Runge-Kutta for the initial value problem

$$y' = f(t,y)$$

with the initial condition $y(t_0) = \alpha$ is given by

$$w_0 = \alpha,$$

$$w_{i+1} = w_i + \frac{h}{2}[f(t_i, w_i) + f(t_i + h, y_i + hf(t_i, w_i))]$$

with an error of order $O(h^2)$.
For ease of calculation this can be rewritten as:

$$k_1 = f(t_i, w_i),$$

$$k_2 = f(t_i + h, w_i + hk_1),$$

$$w_{i+1} = w_i + \frac{h}{2}[k_1 + k_2].$$

3.2 Third Order Runge-Kutta Methods

Higher order methods are derived in a similar fashion. For the Third Order Runge-Kutta methods

$$\frac{w_{i+1} - w_i}{h} = F(f, t_i, w_i, h), \tag{18}$$

with

$$F(f, t, w, h) = a_0 k_1 + a_1 k_2 + a_2 k_3, \tag{19}$$

where

$$a_0 + a_1 + a_2 = 1$$

and

$$\begin{aligned} k_1 &= f(t_i, w_i), \\ k_2 &= f(t_i + \alpha_1 h, t_i + \beta_{11} k_1), \\ k_3 &= f(t_i + \alpha_2 h, t_i + \beta_{21} k_1 + \beta_{22} k_2)). \end{aligned}$$

The values of a_0, a_1, a_2, α_1, α_2, β_{11},β_{21}, β_{22} are derived by group the Taylor expansion,

$$\begin{aligned} y_{i+1} &= y_i + hf(t_i, y_i) + \frac{h^2}{2}(f_t + f_y f)_{(t_i, y_i)} \\ &\quad + \frac{h^3}{6}\left(f_{tt} + 2f_{ty} f + f_t f_y + f_{yy} f^2 + f_y f_y f\right)_{(t_i, y_i)} \\ &\quad + O(h^4), \end{aligned}$$

with the 3rd order expand form:

$$y_{i+1} = y_i + ha_1 f(t_i, y_i) + ha_2(f + \alpha_1 h f_t + \beta_{11} h f_y f$$

$$+\frac{h^2}{2}(f_{tt}\alpha_1^2 + f_{yy}\beta_{11}^2 f^2 + 2f_{ty}\alpha_1\beta_{11} f) + O_2(h^3))$$

$$+ha_3(f + \alpha_2 h f_t + f_y \left(\beta_{21} h f + \beta_{22} h (f + \alpha_1 h f_t + \beta_{11} h f_y f + O_3(h^2)\right))$$

$$+\frac{1}{2}(f_{tt}(\alpha_2 h)^2 + f_{yy} h^2 (\beta_{21} f + \beta_{22}(f + \underbrace{\alpha_1 h f_t + \beta_{11} h f_y f + O_4(h^2)}_{O_5(h)}))^2$$

$$+2f_{ty}\alpha_2 h^2(\beta_{21} f + \beta_{22}(f + \underbrace{\alpha_1 h f_t + \beta_{11} h f_y f + O_4(h^2)}_{O_5(h)})))).$$

This results in 8 equations with 8 unknowns, but only 6 of these equations are independent. For this reason the are two free parameters to choose.

For example, we can choose that

$$\alpha_2 = 1, \beta_{11} = \frac{1}{2},$$

then we obtain the following difference equation.

$$w_{i+1} = w_i + \frac{h}{6}(k_1 + 4k_2 + k_3),$$

where

$$k_1 = f(t_i, w_i),$$
$$k_2 = f(t_n + 1/2h, w_n + 1/2hk_1),$$
$$k_3 = f(t_n + h, w_n - hk_1 + 2hk_2).$$

3.3 Runge-Kutta fourth order

The most commonly used Runge-Kutta method is the 4th Order Runge-Kutta method, which is given by the formulae,

$$w_0 = \alpha,$$

$$k_1 = hf(t_i, w_i),$$
$$k_2 = hf(t_i + \frac{h}{2}, w_i + \frac{1}{2}k_1),$$
$$k_3 = hf(t_i + \frac{h}{2}, w_i + \frac{1}{2}k_2),$$
$$k_4 = hf(t_{i+1}, w_i + k_3),$$
$$w_{i+1} = w_i + \frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4).$$

Example 12

Example Midpoint method,

$$y^{'} = y - t^2 + 1, \quad 0 \leq t \leq 2, \quad y(0) = 0.5,$$

$$N = 10, \quad t_i = 0.2i, \quad h = 0.2,$$
$$w_0 = 0.5,$$

$$\begin{aligned} w_{i+1} &= w_i + 0.2f(t_i + \frac{0.2}{2}, w_i + \frac{0.2}{2}f(t_i, w_i)) \\ &= w_i + 0.2f(t_i + 0.1, w_i + 0.1(w_i - t_i^2 + 1)) \\ &= w_i + 0.2(w_i + 0.1(w_i - t_i^2 + 1) - (t_i + 0.1)^2 + 1) \end{aligned}$$

Example 13

Example Runge-Kutta fourth order method

$$y' = y - t^2 + 1, \quad 0 \leq t \leq 2, \quad y(0) = 0.5,$$

$$N = 10, \quad t_i = 0.2i, \quad h = 0.2,$$

$$w_0 = 0.5,$$

$$\begin{aligned}
k_1 &= h(w_i - t_i^2 + 1), \\
k_2 &= h(w_i + \frac{1}{2}k_1 - (t_i + \frac{h}{2})^2 + 1) \\
k_3 &= h(w_i + \frac{1}{2}k_2 - (t_i + \frac{h}{2})^2 + 1) \\
k_4 &= h(w_i + \frac{1}{2}k_3 - (t_i + h)^2 + 1) \\
w_{i+1} &= w_i + \frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4)
\end{aligned}$$

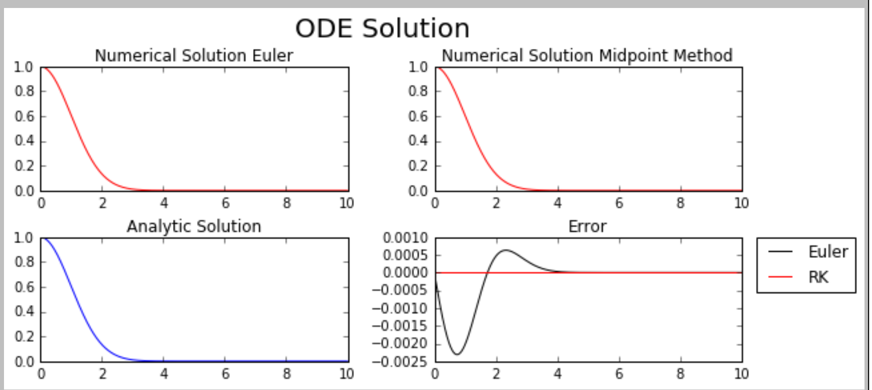


Figure 3.3.1: Python output: Illustrating upper bound $y' = -xy$ with the initial condition $y(0) = 1$

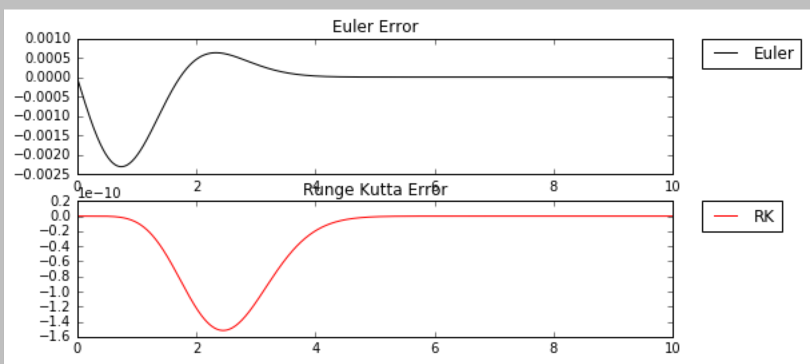


Figure 3.3.2: Python output: Illustrating upper bound $y' = -xy$ with the initial condition $y(0) = 1$

3.4 BUTCHER TABLEAU

Another way of representing a Runge-Kutta method is called the Butcher tableau named after John C Butcher (31 March 1933).

$$y_{i+1} = y_i + h \sum_{n=1}^{s} a_n k_n,$$

where

$$\begin{aligned}
k_1 &= f(t_i, y_i),\\
k_2 &= f(t_i + \alpha_2 h, y_i + h(\beta_{21} k_1)),\\
k_3 &= f(t_i + \alpha_3 h, y_i + h(\beta_{31} k_1 + \beta_{32} k_2)),\\
&\vdots\\
k_s &= f(t_i + \alpha_s h, y_i + h(\beta_{s1} k_1 + \beta_{s2} k_2 + \cdots + \beta_{s,s-1} k_{s-1})).
\end{aligned}$$

These data are usually arranged in a mnemonic device, known as a Butcher tableau

$$\begin{array}{c|ccccc}
0 & & & & & \\
\alpha_2 & \beta_{21} & & & & \\
\alpha_3 & \beta_{31} & \beta_{31} & & & \\
\vdots & \vdots & \vdots & & & \\
\alpha_s & \beta_{s1} & \beta_{s2} & \cdots & \beta_{ss-1} & \\
\hline
 & a_1 & a_2 & \cdots & a_{s-1} & a_s
\end{array}$$

The method is consistent if

$$\sum_{j}^{s-1} \beta_{sj} = \alpha_s.$$

3.4.1 *Heun's Method*

The Butcher's Tableau for Heun's Method is:

$$\begin{array}{c|cc}
0 & & \\
1 & 1 & \\
\hline
 & \frac{1}{2} & \frac{1}{2}
\end{array}$$

3.4.2 *4th Order Runge-Kutta*

The Butcher's Tableau for the 4th Order Runge-Kutta is:

$$\begin{array}{c|cccc}
0 & & & & \\
\frac{1}{2} & \frac{1}{2} & & & \\
\frac{1}{2} & 0 & \frac{1}{2} & & \\
1 & 0 & 0 & 1 & \\
\hline
 & \frac{1}{6} & \frac{2}{6} & \frac{2}{6} & \frac{1}{6}
\end{array}$$

3.5 CONVERGENCE ANALYSIS

In order to obtain convergence of the general Runge-Kutta we need to have the truncation $\tau_n(y) \to 0$ as $h \to 0$. Since,

$$\tau(y) = \frac{y(t_{n+1}) - y(t_n)}{h} - F(t_n, y(t_n), h; f),$$

we require,

$$F(x, y, h; f) \to y'(x) = f(x, y(x)).$$

More precisely define,

$$\delta(h) = \max_{a \le t \le b; -\infty < y < \infty} |f(t, y) - F(t, y, h; f)|,$$

and assume,

$$\delta(h) \to 0, \quad \text{as } h \to 0. \tag{20}$$

This is called the consistency condition for the RK method.
We will also need a Lipschitz Condition on F:

$$|F(t, y, h; f) - F(t, z, h; f)| \le |y - z|, \tag{21}$$

for all $a \le t \le b$ and $-\infty < y, z < \infty$ and small $h > 0$.

Example 14

Looking at the midpoint method

$$\begin{aligned}
|F(t, w, h; f) - F(t, z, h; f)| &= \Big| f(t + \frac{h}{2}, w + \frac{h}{2} f(t, w)) \\
&\quad - f(t + \frac{h}{2}, z + \frac{h}{2} f(t, z)) \Big| \\
&\le K \left| w - z + \frac{h}{2}[f(t, w) - f(t, z)] \right| \\
&\le K \left(1 + \frac{h}{2} K \right) |w - z|
\end{aligned}$$

Theorem 3.5.1. *Assume that the Runge-Kutta method satisfies the Lipschitz Condition. Then for the initial value problems*

$$y' = f(x, y),$$

$$y(x_0) = y_0.$$

The numerical solution $\{w_n\}$ satisfies

$$\max_{a \le x \le b} |y(x_n) - w_n| \le e^{(b-a)L} |y_0 - w_0| + \left[\frac{e^{(b-a)L} - 1}{L} \right] \tau(h)$$

where

$$\tau(h) = \max_{a \le x \le b} |\tau_n(h)|,$$

If the consistency condition

$$\delta(h) \to 0 \text{ as } h \to 0,$$

where

$$\delta(h) = \max_{a \le x \le b} |f(x,y) - F(x,y;h;f)|.$$

Proof. Subtracting

$$w_{n+1} = w_n + hF(t_n, w_n, h; f),$$

and

$$y(t_{n+1}) = y(t_n) + hF(t_n, y(t_n), h; f) + h\tau_n(h),$$

we obtain

$$e_{n+1} = e_n + h[F(t_n, w_n, h; f) - F(t_n, w_n, h; f)] + h\tau_n(h),$$

in which $e_n = y(t_n) - w_n$. Apply the Lipschitz Condition L and the truncation error we obtain

$$|e_{n+1}| \le (1 + hL)|e_n| + h\tau_n(h).$$

This nicely leads to the result.
In most cases it is known by direct computation that $\tau(h) \to 0$ as $h \to 0$ an in that case convergence of $\{w_n\}$ and $y(t_n)$ is immediately proved.
But all we need to know is that (20) is satisfied . To see this we write

$$\begin{array}{rcl} h\tau_n & = & y(t_{n+1}) - y(t_n) - hF(t_n, y(t_n), h; f), \\ & = & hy^{'}(t_n) + \frac{h^2}{2}y^{''}(\xi_n) - hF(t_n, y(t_n), h; f), \\ h|\tau_n| & \le & h\delta(h) + \frac{h^2}{2}|y^{''}|. \\ |\tau_n| & \le & \delta(h) + \frac{h}{2}|y^{''}|. \end{array}$$

Thus $\tau(h) \to 0$ as $h \to 0$ □

From this we have

Corollary 3.5.2. *If the RK method has a truncation error $\tau(h) = O(h^{m+1})$ then the rate of convergence of $\{w_n\}$ to $Y(t)$ is $O(h^m)$.*

3.6 THE CHOICE OF METHOD AND STEP-SIZE

An interesting question is since Runge-Kutta method is 4th order but requires 4 steps and Euler only required 3 is it more beneficial to use a smaller h than a higher order method?
But this does lead us to the question of how do we define our h to

maximize the solution we have.
An ideal difference-equation method

$$w_{i+1} = w_i + h\phi(t_i, w_i, h) \quad i = 0, .., n-1$$

for approximating the solution $y(t)$ to the Initial Value Problem $y^{'} = f(t, y)$ would have the property that given a tolerance $\varepsilon > 0$ the minimal number of mesh points would be used to ensure that the global error $|y(t_i) - w_i|$ would not exceed ε for any $i = 0, ..., N$.
We do this by finding an appropriate choice of mesh points. Although we cannot generally determine the global error of a method there is a close relation between local truncation and global error. By using methods of differing order we can predict the local truncation error and using this prediction choose a step size that will keep global error in check.
Suppose we have two techniques

1. An nth order Taylor method of the form

$$y(t_{i+1}) = y(t_i) + h\phi(t_i, y(t_i), h_i) + O(h^{n+1})$$

producing approximations

$$w_0 = \alpha$$

$$w_{i+1} = w_i + h\phi(t_i, w_i, h_i)$$

with local truncation $\tau_{i+1} = O(h^n)$.

2. An (n+1)st order Taylor of the form

$$y(t_{i+1}) = y(t_i) + h\psi(t_i, y(t_i), h_i) + O(h^{n+2})$$

producing approximations

$$v_0 = \alpha$$

$$v_{i+1} = v_i + h\psi(t_i, v_i, h_i)$$

with local truncation $v_{i+1} = O(h^{n+1})$.

We first make the assumption that $w_i \approx y(t_i) \approx v_i$ and choose a fixed step size to generate w_{i+1} and v_{i+1} to approximate $y(t_{i+1})$. Then

$$\begin{aligned}
\tau_{i+1} &= \frac{y(t_{i+1}) - y(t_i)}{h} - \phi(t_i, y(t_i), h) \\
&= \frac{y(t_{i+1}) - w_i}{h} - \phi(t_i, w_i, h) \\
&= \frac{y(t_{i+1}) - (w_i + h\phi(t_i, w_i, h))}{h} \\
&= \frac{y(t_{i+1}) - w_{i+1}}{h}
\end{aligned}$$

Similarly

$$\Upsilon_{i+1} = \frac{y(t_{i+1}) - v_{i+1}}{h}$$

As a consequence

$$\begin{aligned}
\tau_{i+1} &= \frac{y(t_{i+1}) - w_{i+1}}{h} \\
&= \frac{(y(t_{i+1}) - v_{i+1}) + (v_{i+1} - w_{i+1})}{h} \\
&= \Upsilon_{i+1}(h) + \frac{(v_{i+1} - w_{i+1})}{h}.
\end{aligned}$$

but $\tau_{i+1}(h)$ is $O(h^n)$ and $\Upsilon_{i+1}(h)$ is $O(h^{n+1})$ so the significant factor of $\tau_{i+1}(h)$ must come from $\frac{(v_{i+1}-w_{i+1})}{h}$. This gives us an easily computed approximation of $O(h^n)$ method,

$$\tau_{i+1} \approx \frac{(v_{i+1} - w_{i+1})}{h}.$$

The object is not to estimate the local truncation error but to adjust step size to keep it within a specified bound. To do this we assume that since $\tau_{i+1}(h)$ is $O(h^n)$ a number K independent of h exists with,

$$\tau_{i+1}(h) \approx Kh^n.$$

Then the local truncation error produced by applying the nth order method with a new step size qh can be estimated using the original approximations w_{i+1} and v_{i+1}

$$\tau_{i+1}(qh) \approx K(qh)^n \approx q^n \tau_{i+1}(h) \approx \frac{q^n}{h}(v_{i+1} - w_{i+1}),$$

to bound $\tau_{i+1}(qh)$ by ε we choose q such that

$$\frac{q^n}{h}|v_{i+1} - w_{i+1}| \approx \tau_{i+1}(qh) \leq \varepsilon,$$

which leads to

$$q \leq \left(\frac{\varepsilon h}{|v_{i+1} - w_{i+1}|}\right)^{\frac{1}{n}},$$

which can be used to control the error.

3.7 PROBLEM SHEET 3 - RUNGE-KUTTA

1. Apply the Midpoint Method to approximate the solution of the given initial value problems using the indicated number of time steps. Compare the approximate solution with the given exact solution

   a) $y' = t - y, \quad (0 \leq t \leq 4)$,
   with the initial condition $y(0) = 1$,
   $N = 4$, with the exact solution $y(t) = 2e^{-t} + t - 1$.

   b) $y' = y - t, \quad (0 \leq t \leq 2)$,
   with the initial condition $y(0) = 2$,
   $N = 4$, with the exact solution $y(t) = e^{t} + t + 1$.

2. Apply the 4th Order Runge-Kutta Method to approximate the solution of the given initial value problems using the indicated number of time steps. Compare the approximate solution with the given exact solution

   a) $y' = t - y, \quad (0 \leq t \leq 4)$,
   with the initial condition $y(0) = 1$,
   $N = 4$, with the exact solution $y(t) = 2e^{-t} + t - 1$.

   b) $y' = y - t, \quad (0 \leq t \leq 2)$
   with the initial condition $y(0) = 2$,
   $N = 4$, with the exact solution $y(t) = e^{t} + t + 1$.

3. Derive the difference equation for the Midpoint Runge-Kutta method

$$w_{n+1} = w_n + k_2,$$
$$k_1 = hf(t_n, w_n),$$
$$k_2 = hf(t_n + \frac{1}{2}h, w_n + \frac{1}{2}k_1)$$

   for solving the ordinary differential equation

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

   by using a formula of the form

$$w_{n+1} = w_n + ak_1 + bk_2,$$

where k_1 is defined as above,

$$k_2 = hf(t_n + \alpha h, w_n + \beta k_1),$$

and a, b, α and β are constants are determined. Prove that $a + b = 1$ and $b\alpha = b\beta = \frac{1}{2}$ and choose appropriate values to give the Midpoint Runge-Kutta method.

4

MULTI-STEP METHODS

Methods using the approximation at more than one previous point to determine the approx at the next point are called multi-step methods.

Definition An m-step multi-step method for solving the Initial Value Problem

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

is on whose difference equation for finding the approximation w_{i+1} at the mesh points t_{i+1} can be represented by the following equation, when m is an integer greater than 1,

$$w_{i+1} = a_{m-1}w_i + a_{m-2}w_{i-1} + ... + a_0 w_{i+1-m}$$

$$+h[b_m f(t_{i+1}, w_{i+1}) + b_{m-1} f(t_i, w_i) + ... + b_0 f(t_{i+1-m}, w_{i+1-m})] \quad (22)$$

for $i = m-1, m, ..., N-1$ where $h = \frac{b-a}{N}$ the $a_0, a_1, ..., a_m - 1$ and $b_0, b_1, ..., b_m$ are constants, and the starting values

$$w_0 = \alpha, \quad w_1 = \alpha_1, \quad w_2 = \alpha_2, \ ... \ w_{m-1} = \alpha_{m-1}$$

are specified.

When $b_m = 0$ the method is called **explicit** or open since (22) then gives w_{i+1} explicitly in terms of previously determined approximations.

When $b_m \neq 0$ the method is called **implicit** or closed since w_{i+1} occurs on both sides of (22).

Example 15

Fourth order Adams-Bashforth

$$w_0 = \alpha, \quad w_1 = \alpha_1, \quad w_2 = \alpha_2, \quad w_3 = \alpha_3,$$

$$\begin{aligned} w_{i+1} &= w_i + \frac{h}{24}[55f(t_i, w_i) - 59f(t_{i-1}, w_{i-1}) \\ &\quad +37f(t_{i-2}, w_{i-2}) - 9f(t_{i-3}, w_{i-3})]. \end{aligned}$$

For each $i = 3, 4, ..., N-1$ define an explicit four step method known as the fourth order Adams-Bashforth technique.
The equation

$$w_0 = \alpha \quad w_1 = \alpha_1 \quad w_2 = \alpha_2$$

$$\begin{aligned} w_{i+1} &= w_i + \frac{h}{24}[9f(t_{i+1}, w_{i+1}) + 19f(t_i, w_i) \\ &\quad -5f(t_{i-1}, w_{i-1}) + f(t_{i-2}, w_{i-2})] \end{aligned}$$

For each $i = 2, 4, ..., N-1$ define an implicit three step method known as the fourth order Adams-Moulton technique.

For the previous methods we need to generate α_1, α_2 and α_3 by using a one step method.

4.1 Derivation of a explicit multistep method

4.1.1 *General Derivation of a explicit method Adams-Bashforth*

$$y(t_{i+1}) - y(t_i) = \int_{t_i}^{t_{i+1}} y^{'}(t)dt = \int_{t_i}^{t_{i+1}} f(t, y(t))dt$$

Consequently

$$y(t_{i+1}) = y(t_i) + \int_{t_i}^{t_{i+1}} f(t, y(t))dt$$

Since we cannot integrate $f(t, y(t))$ without knowing $y(t)$ the solution to the problem we instead integrate an interpolating poly. $P(t)$ to $f(t, y(t))$ that is determined by some of the previous obtained data points $(t_0, w_0), (t_1, w_1), ..., (t_i, w_i)$. When we assume in addition that $y(t_i) \approx w_i$

$$y(t_{i+1}) \approx w_i + \int_{t_i}^{t_{i+1}} P(t)dt$$

We use Newton back-substitution to derive an Adams-Bashforth explicit m-step technique, we form the backward difference poly $P_{m-1}(t)$ through $(t_i, f(t_i)), (t_{i-1}, f(t_{i-1})), ..., (t_{i+1-m}, f(t_{i+1-m}))$

$$f(t, y(t)) = P_{m-1}(t) + f^m(\xi, y(\xi))\frac{(t-t_i)...(t-t_{i+1-m})}{m!} \tag{23}$$

$$P_{m-1}(t) = \Sigma_{j=1}^{m} L_{m-1,j}(t)\nabla^j f(t_{i+1-j}, y(t_{i+1-j})) \tag{24}$$

where

$$\nabla f(t_i, y(t_i)) = f(t_i, y(t_i)) - f(t_{i-1}, y(t_{i-1})),$$

$$\nabla^2 f(t_i, y(t_i)) = \nabla f(t_i, y(t_i)) - \nabla f(t_{i-1}, y(t_{i-1}))$$
$$= f(t_i, y(t_i)) - 2f(t_{i-1}, y(t_{i-1})) + f(t_{i-2}, y(t_{i-2})).$$

Derivation of a explicit two-step method Adams Bashforth

To derive two step Adams-Bashforth technique

$$\begin{aligned}
\int_{t_i}^{t_{i+1}} f(t,y)dt &= \int_{t_i}^{t_{i+1}} [f(t_i, y(t_i)) + \frac{(t-t_i)}{h}\nabla f(t_i, y(t_i)) + error]dt \\
y_{i+1} - y_i &= [tf(t_i, y(t_i)) + \frac{t(\frac{t}{2}-t_i)}{h}\nabla f(t_i, y(t_i))]_{t_i}^{t_{i+1}} + Error \\
y_{i+1} &= y(t_i) + (t_{i+1} - t_i) f(t_i, y(t_i)) \\
&\quad + \frac{\frac{t_{i+1}}{2} - t_{i+1}t_i + \frac{t_i^2}{2} - t_i^2}{h}\nabla(f(t_i, y(t_i)) + Error \\
&= y(t_i) + hf(t_i, y(t_i)) \\
&\quad + \frac{(t_{i+1}-t_i)^2}{2h}(f(t_i, y(t_i)) - f(t_{i-1}, y(t_{i-1}))) + Error \\
&= y(t_i) + hf(t_i, y(t_i)) \\
&\quad + \frac{1}{2}(f(t_i, y(t_i)) - f(t_{i-1}, y(t_{i-1}))) + Error \\
&= y(t_i) + \frac{h}{2}[3f(t_i, y(t_i)) - f(t_{i-1}, y(t_{i-1})) + Error].
\end{aligned}$$

The two step Adams-Bashforth is $w_0 = \alpha_0$ and $w_1 = \alpha_1$ with

$$w_{i+1} = w_i + \frac{h}{2}[3w_i - w_{i-1}] \quad for \quad i = 1, .., N-1$$

The local truncation error is

$$\tau_{i+1}(h) = \frac{y(t_i+1) - y(t_i)}{h} - \frac{1}{2}[3f(t_i, y(t_i)) - f(t_{i-1}, y(t_{i-1}))]$$

$$\tau_{i+1}(h) = \frac{Error}{h}$$

$$\begin{aligned}
Error &= \int_{t_i}^{t_{i+1}} \frac{(t-t_i)(t-t_{i-1})}{2!} f^2(\mu_i, y(\mu_i))]dt \\
&= \frac{5}{12}h^3 f^2(\mu_i, y(\mu_i))
\end{aligned}$$

$$\tau_{i+1}(h) = \frac{\frac{5}{12}h^3 f^2(\mu_i, y(\mu_i))}{h}$$

The local truncation error for the two step Adams-Bashforth methods is of order 2

$$\tau_{i+1}(h) = O(h^2)$$

General Derivation of a explicit method Adams-Bashforth (cont.)

Definition The Lagrange polynomial $L_{m-1,j}(t)$ has a degree of $m-1$ and is associated with the interpolation point t_j in the sense

$$L_{m-1,j}(t) = \begin{cases} 1 & i = j \\ 0 & i \neq j \end{cases},$$

$$L_{m-1,j}(t) = \frac{(t-t_0)...(t-t_{m-1})}{(t_j-t_0)...(t_j-t_{m-1})} = \prod_{k=0,k\neq j}^{m-1} \frac{t-t_k}{t_j-t_k}. \tag{25}$$

Introducing the variable $t = t_k + sh$ with $dt = hds$ into $L_{m-1}(t)$

$$L_{m-1,j}(t) = \prod_{k=0,k\neq j}^{m-1} \frac{t_i + sh - t_k}{(i+1)h} = (-1)^{(m-1)} \binom{-s}{(m-1)} \tag{26}$$

$$\begin{aligned} \int_{t_i}^{t_{i+1}} f(t,y(t))dt = \quad & \int_{t_i}^{t_{i+1}} \textstyle\sum_{k=0}^{m-1} \binom{-s}{k} \nabla^k f(t_i, y(t_i))dt \\ & + \int_{t_i}^{t_{i+1}} f^m(\xi, y(\xi)) \frac{(t-t_i)...(t-t_{i+1-m})}{m!} dt \\ = \quad & \textstyle\sum_{k=0}^{m-1} \nabla^k f(t_i, y(t_i)) h (-1)^k \int_0^1 \binom{-s}{k} ds \\ & + \frac{h^{m+1}}{m!} \int_0^1 s(s+1)...(s+m-1) f^m(\xi, y(\xi)) ds. \end{aligned}$$

The integrals $(-1)^k \int_0^1 \binom{-s}{k} ds$ for various values of k are computed as such,

Example 16

Example $k = 2$,

$$\begin{aligned} (-1)^2 \int_0^1 \binom{-s}{2} ds &= \int_0^1 \frac{-s(-s-1)}{1.2} ds \\ &= \frac{1}{2} \int_0^1 s^2 + sds \\ &= \frac{1}{2} \left[\frac{s^3}{3} + \frac{s^2}{2} \right]_0^1 = \frac{5}{12} \end{aligned}$$

k	0	1	2	3	4	...
$(-1)^k \int_0^1 \binom{-s}{k} ds$	1	$\frac{1}{2}$	$\frac{5}{12}$	$\frac{3}{8}$	.	..

Table 1: Table of Adams-Bashforth coefficients

As a consequence

$$\int_{t_i}^{t_{i+1}} f(t,y(t))dt = h\left[f(t_i,y(t_i)) + \frac{1}{2}\nabla f(t_i,y(t_i)) + \frac{5}{12}\nabla^2 f(t_i,y(t_i)) + ...\right]$$

$$+\frac{h^{m+1}}{m!}\int_0^1 s(s+1)...(s+m-1)f^m(\xi,y(\xi))ds.$$

Since $s(s+1)...(s+m-1)$ does not change sign on $[0,1]$ it can be stated that for some μ_i where $t_{i+1-m} < \mu_i < t_{i+1}$ the error term becomes

$$\frac{h^{m+1}}{m!}\int_0^1 s(s+1)...(s+m-1)f^m(\xi,y(\xi))ds$$

$$\frac{h^{m+1}}{m!}f^m(\mu,y(\mu))\int_0^1 s(s+1)...(s+m-1)ds$$

Since $y(t_{i+1}) - y(t_i) = \int_{t_i}^{t_{i+1}} f(s,y(s))ds$ this can be written as

$$y(t_{i+1}) = y(t_i) + h\left[f(t_i,y(t_i)) + \frac{1}{2}\nabla f(t_i,y(t_i)) + \frac{5}{12}\nabla^2 f(t_i,y(t_i)) + ...\right].$$

Example 17

To derive the two step Adams-Bashforth method

$$\begin{aligned}
y(t_{i+1}) &\approx y(t_i) + h[f(t_i,y(t_i)) + \frac{1}{2}(\nabla f(t_i,y(t_i)))] \\
&= y(t_i) + h[f(t_i,y(t_i)) \\
&\quad +\frac{1}{2}(f(t_i,y(t_i)) - f(t_{i-1},y(t_{i-1})))] \\
&= y(t_i) + \frac{h}{2}[3f(t_i,y(t_i)) - f(t_{i-1},y(t_{i-1}))].
\end{aligned}$$

The two step Adams-Bashforth is $w_0 = \alpha_0$ and $w_1 = \alpha_1$ with

$$w_{i+1} = w_i + \frac{h}{2}[3w_i - w_{i-1}], \quad for \;\; i = 1,..,N-1.$$

Definition If $y(t)$ is a solution of the Initial Value Problem

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

and

$$w_{i+1} = a_{m-1}w_i + a_{m-2}w_{i-1} + ... + a_0 w_{i+1-m}$$

$$+h[b_m f(t_{i+1},w_{i+1}) + b_{m-1}f(t_i,w_i) + ... + b_0 f(t_{i+1-m},w_{i+1-m})]$$

is the (i+1)th step in a multi-step method, the local truncation error at this step is

$$\tau_{i+1}(h) = \frac{y(t_{i+1}) - a_{m-1}y(t_i) - ... - a_0 y(t_{i+1-m})}{h}$$

$$-[b_m f(t_{i+1}, y(t_{i+1})) + b_{m-1} f(t_i, y(t_i)) + ... + b_0 f(t_{i+1-m}, y(t_{i+1-m}))]$$

for each $i = m-1, ..., N-1$.

Example 18

Truncation error for the two step Adams-Bashforth method is

$$h^3 f^2(\mu_i, y(\mu_i))(-1)^2 \int_0^1 \left(\begin{array}{c} -s \\ 2 \end{array} \right) ds = \frac{5h^3}{12} f^2(\mu_i, y(\mu_i))$$

using the fact that $f^2(\mu_i, y(\mu_i)) = y^3(\mu_i)$

$$\tau_{i+1}(h) = \frac{y(t_{i+1}) - y(t_i)}{h} - \frac{1}{2}[3f(t_i, y(t_i)) - f(t_{i-1}, y(t_{i-1}))]$$

$$= \frac{1}{h}\left[\frac{5h^3}{12} f^2(\mu_i, y(\mu_i))\right] = \frac{5}{12} h^2 y^3(\mu_i)$$

4.1.2 *Adams-Bashforth three step method*

$$w_0 = \alpha, \quad w_1 = \alpha_1, \quad w_2 = \alpha_2,$$

$$w_{i+1} = w_i + \frac{h}{12}[23f(t_i, w_i) - 16f(t_{i-1}, w_{i-1}) + 5f(t_{i-2}, w_{i-2})],$$

where $i = 2, 3, ..., N-1$.
The local truncation error is of order 3

$$\tau_{i+1}(h) = \frac{3}{8} h^3 y^4(\mu_i)$$

$\mu_i \in (t_{i-2}, t_{i+1})$.

4.1.3 *Adams-Bashforth four step method*

$$w_0 = \alpha, \quad w_1 = \alpha_1, \quad w_2 = \alpha_2, \quad w_3 = \alpha_3,$$

$$w_{i+1} = w_i + \frac{h}{24}[55f(t_i, w_i) - 59f(t_{i-1}, w_{i-1}) + 37f(t_{i-2}, w_{i-2}) - 9f(t_{i-3}, w_{i-3})],$$

where $i = 3, ..., N-1$.
The local truncation error is of order 4

$$\tau_{i+1}(h) = \frac{251}{720} h^4 y^5(\mu_i),$$

$\mu_i \in (t_{i-3}, t_{i+1})$.

Example 19

$$y^{'} = y - t^2 + 1, \quad 0 \le t \le 2, \quad y(0) = 0.5,$$

Adams-Bashforth two step $w_0 = \alpha_0$ and $w_1 = \alpha_1$ with

$$w_{i+1} = w_i + \frac{h}{2}[3f(t_i, w_i) - f(t_{i-1}, w_{i-1})], \quad for, \;\; i = 1, .., N-1,$$

truncation error

$$\tau_{i+1}(h) = \frac{5}{12}h^2 y^3(\mu_i), \qquad \mu_i \in (t_{i-1}, t_{i+1}).$$

1. Calculate α_0 and α_1.
   From the initial condition we have $w_0 = 0.5$.
   To calculate w_1 we use the modified Euler method.

$$w_0 \;\; = \;\; \alpha$$

   We only need this to calculate w_1

$$w_1 \;\; = \;\; w_0 + \frac{h}{2}[f(t_0, w_0) + f(t_1, w_0 + hf(t_0, w_0))].$$

$$\begin{aligned}
w_0 &= 0.5 \\
w_1 &= w_0 + \frac{h}{2}[f(t_0, w_0) + f(t_1, w_0 + hf(t_0, w_0))] \\
w_1 &= w_0 + \frac{0.2}{2}[w_0 - t_0^2 + 1 + w_0 + h(w_0 - t_0^2 + 1) - t_1^2 + 1] \\
&= 0.5 + \frac{0.2}{2}[0.5 - 0 + 1 + 0.5 + 0.2(1.5) - (0.2)^2 + 1] \\
&= 0.826
\end{aligned}$$

   we now have $\alpha_1 = w_1 = 0.826$.
2. Calculate w_i for $i = 2, ...N$.

$$\begin{aligned}
w_2 &= w_1 + \frac{h}{2}[3f(t_1, w_1) - f(t_0, w_0)] \\
&= w_1 + \frac{h}{2}[3(w_1 - t_1^2 + 1) - (w_0 - t_0^2 + 1)] \\
&= 0.826 + \frac{0.2}{2}[3(0.826 - 0.2^2 + 1) - (0.5 - 0^2 + 1)] \\
&= 0.8858 \\
&\;\;. \\
&\;\;. \\
&\;\;. \\
w_{i+1} &= w_i + \frac{0.2}{2}[3(w_i - t_i^2 + 1) - (w_{i-1} - t_{i-1}^2 + 1)]
\end{aligned}$$

this method can be generalised for all Adams-Bashforth.

4.2 Derivation of the implicit multi-step method

4.2.0.1 *Derivation of an implicit one-step method Adams Moulton*

To derive one step Adams-Moulton technique

$$\begin{aligned}
\int_{t_i}^{t_{i+1}} f(t,y)dt &= \int_{t_i}^{t_{i+1}} [f(t_{i+1}, y(t_{i+1})) + \frac{(t-t_{i+1})}{h}\nabla f(t_{i+1}, y(t_{i+1})) + error]dt \\
y_{i+1} - y_i &= [tf(t_{i+1}, y(t_{i+1})) \\
&\quad + \frac{t(\frac{t}{2} - t_{i+1})}{h}\nabla f(t_{i+1}, y(t_{i+1}))]_{t_i}^{t_{i+1}} + Error \\
y_{i+1} &= y(t_i) + (t_{i+1} - t_i)f(t_{i+1}, y(t_{i+1})) \\
&\quad + \frac{\frac{t_{i+1}^2}{2} - t_{i+1}^2 + t_i t_{i+1} - \frac{t_i^2}{2}}{h}\nabla(f(t_{i+1}, y(t_{i+1})) \\
&\quad + Error \\
&= y(t_i) + hf(t_{i+1}, y(t_{i+1})) \\
&\quad + \frac{-(t_{i+1} - t_i)^2}{2h}(f(t_{i+1}, y(t_{i+1})) - f(t_i, y(t_i))) \\
&\quad + Error \\
&= y(t_i) + hf(t_{i+1}, y(t_{i+1})) \\
&\quad - \frac{h}{2}(f(t_{i+1}, y(t_{i+1})) - f(t_i, y(t_i))) + Error \\
&= y(t_i) + \frac{h}{2}[f(t_{i+1}, y(t_{i+1})) + f(t_{i-1}, y(t_{i-1}))] + Error
\end{aligned}$$

The two step Adams-Moulton is $w_0 = \alpha_0$ and $w_1 = \alpha_1$ with

$$w_{i+1} = w_i + \frac{h}{2}[w_{i+1} + w_i] \quad for \;\; i = 0, .., N-1$$

The local truncation error is

$$\tau_{i+1}(h) = \frac{y(t_i + 1) - y(t_i)}{h} - \frac{1}{2}[f(t_{i+1}, y(t_{i+1})) + f(t_i, y(t_i))]$$

$$\tau_{i+1}(h) = \frac{Error}{h}$$

$$\begin{aligned}
Error &= \int_{t_i}^{t_{i+1}} \frac{(t-t_{i+1})(t-t_i)}{2!} f^2(\mu_i, y(\mu_i))]dt \\
&= \frac{1}{12}h^3 f^2(\mu_i, y(\mu_i))
\end{aligned}$$

$$\tau_{i+1}(h) = \frac{\frac{1}{12}h^3 f^2(\mu_i, y(\mu_i))}{h}$$

The local truncation error for the one step Adams-Moulton methods is of order 2

$$\tau_{i+1}(h) = O(h^2)$$

derivation of the implicit multi-step method (cont)

As before

$$\begin{aligned} y(t_{i+1}) - y(t_i) &= \int_{-1}^{0} y^{'}(t_{i+1} + sh)ds \\ &= \sum_{k=0}^{m-1} \nabla^k f(t_{i+1}, y(t_{i+1}))h(-1)^k \int_{-1}^{0} \begin{pmatrix} -s \\ k \end{pmatrix} ds \\ &\quad + \frac{h^{m+1}}{m!} \int_{-1}^{0} s(s+1)...(s+m-1)f^m(\xi, y(\xi))ds. \end{aligned}$$

Example 20

For k=3 we have

$$\begin{aligned} (-1)^3 \int_{-1}^{0} \begin{pmatrix} -s \\ k \end{pmatrix} ds &= \int_{-1}^{0} \frac{-s(-s-1)(-s-2)}{1.2.3} ds \\ &= \frac{1}{6}\left[\frac{s^4}{4} + s^3 + s^2\right]_{-1}^{0} = -\frac{1}{24}. \end{aligned}$$

The general form of the Adams-Moulton method is

$$\begin{aligned} y(t_{i+1}) &= y(t_i) + h[f(t_{i+1}, y(t_{i+1})) - \frac{1}{2}\nabla f(t_{i+1}, y(t_{i+1})) - \frac{1}{12}\nabla^2 f(t_{i+1}, y(t_{i+1})) - ...] \\ &\quad + \frac{h^{m+1}}{m!} \int_{-1}^{0} s(s+1)...(s+m-1)f^m(\xi, y(\xi))ds. \end{aligned}$$

adams-moulton two step method

$$w_0 = \alpha, \quad w_1 = \alpha_1,$$

$$w_{i+1} = w_i + \frac{h}{12}[5f(t_{i+1}, w_{i+1}) + 8f(t_i, w_i) - f(t_{i-1}, w_{i-1})]$$

where i=2,3,...,N-1.
The local truncation error is

$$\tau_{i+1}(h) = -\frac{1}{24}h^3 y^4(\mu_i),$$

$\mu_i \in (t_{i-1}, t_{i+1})$.

adams-moulton three step method

$$w_0 = \alpha, \quad w_1 = \alpha_1, \quad w_2 = \alpha_2,$$

$$w_{i+1} = w_i + \frac{h}{24}[9f(t_{i+1}, w_{i+1}) + 19f(t_i, w_i) - 5f(t_{i-1}, w_{i-1}) + f(t_{i-2}, w_{i-2})],$$

where i=3,...,N-1.
The local truncation error is

$$\tau_{i+1}(h) = -\frac{19}{720}h^4 y^5(\mu_i),$$

$\mu_i \in (t_{i-2}, t_{i+1})$.

Example 21

$$y' = y - t^2 + 1 \quad 0 \le t \le 2 \quad y(0) = 0.5$$

Adams-Moulton two step $w_0 = \alpha_0$ and $w_1 = \alpha_1$ with

$$w_{i+1} = w_i + \frac{h}{12}[5f(t_{i+1}, w_{i+1}) + 8f(t_i, w_i) - f(t_{i-1}, w_{i-1})]$$

$$for \;\; i = 2, .., N-1$$

truncation error $\tau_{i+1}(h) = -\frac{1}{24}h^3 y^4(\mu_i)$ $\mu_i \in (t_{i-1}, t_{i+1})$.

1. Calculate α_0 and α_1
   From the initial condition we have $w_0 = 0.5$
   To calculate w_1 we use the modified Euler method.
   we now have $\alpha_1 = w_1 = 0.826$
2. Calculate w_i for $i = 2, ...N$

$$\begin{aligned}
w_2 &= w_1 + \frac{h}{12}[5f(t_2, w_2) \\
&\quad +8f(t_1, w_1) - f(t_0, w_0)] \\
&= w_1 + \frac{h}{12}[5(w_2 + t_2^2 + 1) \\
&\quad +8(w_1 + t_1^2 + 1) - (w_0 + t_0^2 + 1)] \\
&= w_1 + \frac{h}{12}[5(w_2 + t_2^2 + 1) \\
&\quad +8(w_1 + t_1^2 + 1) - (w_0 + t_0^2 + 1)] \\
&. \\
&. \\
&. \\
(1 - \frac{5h}{12})w_{i+1} &= \frac{h}{12}[[8 + \frac{12}{h}]w_i - 5t_{i+1}^2 \\
&\quad -8t_i^2 + t_{i-1}^2 + 12]
\end{aligned}$$

This, of course can be generalised.
The only unfortunate aspect of the implicit method is that you must convert it into an explicit method, this is not always possible. eg $y' = e^y$.

4.3 Table of Adam's methods

Order	Formula	LTE
1	$y_{n+1} = y_n + hf_n$	$\frac{h^1}{2}y''(\eta)$
2	$y_{n+1} = y_n + \frac{h}{2}[3f_n - f_{n-1}]$	$\frac{5h^2}{12}y'''(\eta)$
3	$y_{n+1} = y_n + \frac{h}{12}[23f_n - 16f_{n-1} + 5f_{n-2}]$	$\frac{3h^3}{8}y^{(4)}(\eta)$
4	$y_{n+1} = y_n + \frac{h}{24}[55f_n - 59f_{n-1} + 37f_{n-2} - 9f_{n-3}]$	$\frac{251h^4}{720}y^{(5)}(\eta)$

Table 2: Adams-Bashforth formulas of different order. LTE stands for local truncation error.

Order	Formula	LTE
0	$y_{n+1} = y_n + hf_{n+1}$	$-\frac{h^1}{2}y''(\eta)$
1	$y_{n+1} = y_n + \frac{h}{2}[f_{n+1} + f_n]$	$-\frac{h^2}{12}y'''(\eta)$
2	$y_{n+1} = y_n + \frac{h}{12}[5f_{n+1} + 8f_n - f_{n-1}]$	$-\frac{h^3}{24}y^{(4)}(\eta)$
3	$y_{n+1} = y_n + \frac{h}{24}[9f_{n+1} + 19f_n - 5f_{n-1} + f_{n-2}]$	$-\frac{19h^4}{720}y^{(5)}(\eta)$

Table 3: Adams-Moulton formulas of different order. LTE stands for local truncation error.

4.4 Predictor-corrector method

In practice implicit methods are not used as above. They are used to improve approximations obtained by explicit methods. The combination of the two is called predictor-corrector method.

Example 22

Consider the following forth order method for solving an initial-value problem.

1. Calculate w_0, w_1, w_2, w_3 for the four step Adams-Bashforth method, to do this we use a 4th order one step method, eg Runge Kutta.
2. Calculate an approximation w_4 to $y(t_4)$ using the Adams-Bashforth method as the predictor.

$$w_4^0 = w_3 + \frac{h}{24}[55f(t_3, w_3) - 59f(t_2, w_2) + 37f(t_1, w_1) - 9f(t_0, w_0)]$$

3. This approximation is improved by inserting w_4^0 in the RHS of the three step Adams-Moulton and using it as a corrector

$$w_4^1 = w_3 + \frac{h}{24}[9f(t_4, w_4^0) + 19f(t_3, w_3) - 5f(t_2, w_2) + f(t_1, w_1)]$$

   The only new function evaluation is $f(t_4, w_4^0)$.
4. w_4^1 is the approximation of $y(t_4)$.
5. Repeat steps 2-4 for calculating the approximation of $y(t_5)$.
6. Repeat til $y(t_n)$.

Improved approximations to $y(t_{i+1})$ can be obtained by integrating the Adams Moulton formula

$$w_{i+1}^{k+1} = w_i + \frac{h}{24}[9f(t_{i+1}, w_{i+1}^k) + 19f(t_i, w_i) - 5f(t_{i-1}, w_{i-1}) + f(t_{i-2}, w_{i-2})]$$

w_{i+1}^{k+1} converges to the approximation of the implicit method rather than the solution $y(t_{i+1})$.
A more effective method is to reduce step-size if improved accuracy is needed.

4.5 Improved step-size multi-step method

As the predictor corrector technique produces two approximations of each step it is a natural candidate for error-control (see previous section).

Example 23

The Adams-Bashforth 4-step method is the predictor:

$$w_0 = \alpha, \quad w_1 = \alpha_1, \quad w_2 = \alpha_2, \quad w_3 = \alpha_3,$$

$$\begin{aligned} y(t_{i+1}) \quad = \quad & y(t_i) + \frac{h}{24}[55f(t_i, y(t_i)) - 59f(t_{i-1}, y(t_{i-1})) \\ & +37f(t_{i-2}, y(t_{i-2})) - 9f(t_{i-3}, y(t_{i-3}))] \\ & +\frac{251}{720}h^5 y^5(\mu_i), \end{aligned}$$

$\mu_i \in (t_{i-3}, t_{i+1})$ the truncation error is

$$\frac{y(t_{i+1}) - w_{i+1}^0}{h} = \frac{251}{720}h^4 y^5(\mu_i) \qquad (27)$$

Similarly for Adams-Moulton three step method is the corrector

$$\begin{aligned} y(t_{i+1}) &= y(t_i) + \frac{h}{24}[9f(t_{i+1}, y(t_{i+1})) + 19f(t_i, y(t_i)) \\ &\quad -5f(t_{i-1}, y(t_{i-1})) + f(t_{i-2}, y(t_{i-2}))] \\ &\quad -\frac{19}{720}h^5 y^5(\xi_i) \end{aligned}$$

$\xi_i \in (t_{i-2}, t_{i+1})$ the truncation error is

$$\frac{y(t_{i+1}) - w_{i+1}}{h} = -\frac{19}{720}h^4 y^5(\xi_i) \tag{28}$$

We make the assumption that for small h

$$y^5(\xi_i) \approx y^5(\mu_i).$$

Subtracting (27) from (28) we have

$$\frac{w_{i+1} - w^0_{i+1}}{h} = \frac{h^4}{720}[251y^5(\mu_i) + 19y^5(\xi_i)] \approx \frac{3}{8}h^4 y^5(\xi_i),$$

$$y^5(\xi_i) \approx \frac{8}{3h^5}(w_{i+1} - w^0_{i+1}). \tag{29}$$

Using this result to eliminate $h^4 y^5(\xi_i)$ from (28)

$$\begin{aligned} |\tau_{i+1}(h)| = \frac{|y(t_{i+1}) - w_{i+1}|}{h} &\approx \frac{19h^4}{720}\frac{8}{3h^5}(|w_{i+1} - w^0_{i+1}|) \\ &= \frac{19|w_{i+1} - w^0_{i+1}|}{270h}, \end{aligned}$$

Now consider a new step size qh generating new approximations $\hat{w}_0, \hat{w}_1, .., \hat{w}_i$ The objective is to choose a q so that the local truncation is bounded by a tol ε

$$\frac{|y(t_i + qh) - \hat{w}_{i+1}|}{qh} = \frac{19h^4}{720}|y^5(\nu)|q^4 \approx \frac{19h^4}{720}\frac{8}{3h^5}(|w_{i+1} - w^0_{i+1}|)q^4$$

and we need to choose a q so that

$$\frac{|y(t_i + qh) - \hat{w}_{i+1}|}{qh} \approx \frac{19q^4|w_{i+1} - w^0_{i+1}|}{270h} < \varepsilon,$$

that is, we choose so that

$$q < \left(\frac{270}{19}\frac{h\varepsilon}{|w_{i+1} - w^0_{i+1}|}\right)^{\frac{1}{4}} \approx 2\left(\frac{h\varepsilon}{|w_{i+1} - w^0_{i+1}|}\right)^{\frac{1}{4}}.$$

q is normally chosen as

$$q = 1.5 \left(\frac{h\varepsilon}{|w_{i+1} - w_{i+1}^{0}|} \right)^{\frac{1}{4}}$$

With this knowledge we can change step sizes and control out error.

4.6 PROBLEM SHEET 4 - MULTISTEP METHODS

1. Apply the 3-step Adams-Bashforth to approximate the solution of the given initial value problems using the indicated number of time steps. Compare the approximate solution with the given exact solution

   a) $y' = t - y, \quad (0 \leq t \leq 4)$
   with the initial condition $y(0) = 1$,
   $N = 4$, $y(t) = 2e^{-t} + t - 1$

   b) $y' = y - t, \quad (0 \leq t \leq 2)$
   with the initial condition $y(0) = 2$,
   $N = 4$, $y(t) = e^{t} + t + 1$

2. Apply the 2-step Adams-Moulton Method to approximate the solution of the given initial value problems using the indicated number of time steps. Compare the approximate solution with the given exact solution

   a) $y' = t - y, \quad (0 \leq t \leq 4)$
   with the initial condition $y(0) = 1$,
   $N = 4$, $y(t) = 2e^{-t} + t - 1$

   b) $y' = y - t, \quad (0 \leq t \leq 2)$
   with the initial condition $y(0) = 2$,
   $N = 4$, $y(t) = e^{t} + t + 1$

3. Derive the difference equation for the 1-step Adams-Bashforth method:

$$w_{n+1} = w_n + hf(t_n, w_n),$$

   with the local truncation error

$$\tau_{n+1}(h) = \frac{h}{2} y^2(\mu_n)$$

   where $\mu_n \in (t_n, t_{n+1})$.

4. Derive the difference equation for the 2-step Adams-Bashforth method:

$$w_{n+1} = w_n + \left(\frac{3}{2} hf(t_n, w_n) - \frac{1}{2} hf(t_{n-1}, w_{n-1})\right),$$

   with the local truncation error

$$\tau_{n+1}(h) = \frac{5h^2}{12} y^3(\mu_n)$$

where $\mu_n \in (t_{n-1}, t_{n+1})$.

5. Derive the difference equation for the 3-step Adams-Bashforth method:

$$w_{n+1} = w_n + (\frac{23}{12}hf(t_n, w_n) - \frac{4}{3}hf(t_{n-1}, w_{n-1}) + \frac{5}{12}hf(t_{n-2}, w_{n-2})),$$

with the local truncation error

$$\tau_{n+1}(h) = \frac{9h^3}{24}y^4(\mu_n)$$

where $\mu_n \in (t_{n-2}, t_{n+1})$.

6. Derive the difference equation for the 0-step Adams-Moulton method:

$$w_{n+1} = w_n + hf(t_{n+1}, w_{n+1}),$$

with the local truncation error

$$\tau_{n+1}(h) = -\frac{h}{2}y^2(\mu_n)$$

where $\mu_n \in (t_{n-2}, t_{n+1})$.

7. Derive the difference equation for the 1-step Adams-Moulton method:

$$w_{n+1} = w_n + \frac{1}{2}hf(t_{n+1}, w_{n+1}) + \frac{1}{2}hf(t_n, w_n),$$

with the local truncation error

$$\tau_{n+1}(h) = -\frac{h^2}{12}y^3(\mu_n)$$

where $\mu_n \in (t_n, t_{n+1})$.

8. Derive the difference equation for the 2-step Adams-Moulton method:

$$w_{n+1} = w_n + \frac{5}{12}hf(t_{n+1}, w_{n+1}) + \frac{8}{12}hf(t_n, w_n) - \frac{1}{12}hf(t_{n-1}, w_{n-1}),$$

with the local truncation error

$$\tau_{n+1}(h) = -\frac{h^3}{24}y^4(\mu_n)$$

where $\mu_n \in (t_{n-1}, t_{n+1})$.

9. Derive the difference equation for the 3-step Adams-Moulton method:

$$w_{n+1} = w_n + \frac{9}{24}hf(t_{n+1}, w_{n+1}) + \frac{19}{24}hf(t_n, w_n) - \frac{5}{24}hf(t_{n-1}, w_{n-1}) + \frac{1}{24}hf(t_{n-2}, w_{n-2}),$$

with the local truncation error

$$\tau_{n+1}(h) = -\frac{h^4}{720} y^5(\mu_n)$$

where $\mu_n \in (t_{n-2}, t_{n+1})$.

5

CONSISTENCY, CONVERGENCE AND STABILITY

5.1 ONE STEP METHODS

Stability is why some methods give satisfactory results and some do not.

Definition A one-step method with local truncation error $\tau_i(h)$ at the ith step is said to be **consistent** with the differential equation it approximates if

$$\lim_{h\to 0}(\max_{1\le i\le N}|\tau_i(h)|) = 0$$

where

$$\tau_i(h) = \frac{y_{i+1} - y_i}{h} - F(t_i, y_i, h, f)$$

As $h \to 0$ does $F(t_i, y_i, h, f) \to f(t, y)$.

Definition A one step method difference equation is said to be **convergent** with respect to the differential equation and w_i, the approximation obtained from the difference method at the ith step.

$$\max_{h\to 0}\max_{1\le i\le N}|y(t_i) - w_i| = 0$$

For Euler's method we have

$$\max_{1\le i\le N}|w_i - y(t_i)| \le \frac{Mh}{2L}|e^{L(b-a)} - 1|$$

so Euler's method is convergent wrt to a differential equation.

Theorem 5.1.1. *Suppose the initial value problem*

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

is approximated by a one step difference method in the form

$$\begin{aligned} w_0 &= \alpha \\ w_{i+1} &= w_i + hF(t_i, w_i : h) \end{aligned}$$

Suppose also that a number $h_0 > 0$ exists and that $F(t_i, w_i : h)$ is continuous and satisfies a Lipschitz Condition in the variable w with Lipschitz constant L on the set

$$D = \{(t, w, h) | a \leq t \leq b, -\infty < w < \infty, 0 \leq h \leq h_0\}$$

Then

1. *The method is stable;*
2. *The difference method is convergent if and only if it is consistent - that is iff*
$$F(t_i, w_i : 0) = f(t, y) \text{ for all } a \leq t \leq b$$
3. *If a function τ exists and for each $i = 1, 2, .., N$, the local truncation error $\tau_i(h)$ satisfies $|\tau_i(h)| \leq \tau(h)$ whenever $0 \leq h \leq h_0$, then*
$$|y(t_i) - w_i| \leq \frac{\tau(h)}{L} e^{L(t_i - a)}.$$

Example 24

Consider the modified Euler method given by

$$w_0 = \alpha$$

$$w_{i+1} = w_i + \frac{h}{2}[f(t_i, w_i) + f(t_{i+1}, w_i + hf(t_i, w_i))]$$

Verify that this method satisfies the theorem. For this method

$$F(t, w : h) = \frac{1}{2} f(t, w) + \frac{1}{2} f(t + h, w + hf(t, w))$$

If f satisfies the Lipschitz Condition on $\{(t, w) | a \leq t \leq b, -\infty < w < \infty\}$ in the variable w with constant L, then

$$\begin{aligned} F(t, w : h) - F(t, \hat{w} : h) &= \frac{1}{2} f(t, w) + \frac{1}{2} f(t + h, w + hf(t, w)) \\ &\quad - \frac{1}{2} f(t, \hat{w}) - \frac{1}{2} f(t + h, \hat{w} + hf(t, \hat{w})) \end{aligned}$$

the Lipschitz Condition on f leads to

$$\begin{aligned}
|F(t,w:h)-F(t,\hat{w}:h)| &\leq \frac{1}{2}L|w-\hat{w}| \\
&\quad +\frac{1}{2}L|w+hf(t,w)-\hat{w}-hf(t,\hat{w})| \\
&\leq L|w-\hat{w}|+\frac{1}{2}L|hf(t,w)-hf(t,\hat{w})| \\
&\leq L|w-\hat{w}|+\frac{1}{2}hL^2|w-\hat{w}| \\
&\leq \left(L+\frac{1}{2}hL^2\right)|w-\hat{w}|
\end{aligned}$$

Therefore, F satisfies a Lipschitz Condition in w on the set

$$D=\{(t,w,h)|a\leq t\leq b,-\infty<w<\infty,0\leq h\leq h_0\}$$

for any $h_0>0$ with constant $L^{'}=\left(L+\frac{1}{2}hL^2\right)$
Finally, if f is continuous on $\{(t,w)|a\leq t\leq b,-\infty< w<\infty\}$, then F is continuous on

$$D=\{(t,w,h)|a\leq t\leq b,-\infty<w<\infty,0\leq h\leq h_0\}$$

so this implies that the method is stable. Letting $h=0$ we have

$$F(t,w:0)=\frac{1}{2}f(t,w)+\frac{1}{2}f(t+0,w+0f(t,w))=f(t,w)$$

so the consistency condition holds.
We know that the method is of order $O(h^2)$

5.2 MULTI-STEP METHODS

The general multi-step method for approximating the solution to the Initial Value Problem

$$y^{'}=f(t,y)\quad a\leq t\leq b\ \ y(a)=\alpha$$

can be written in the form

$$w_0=\alpha\quad w_1=\alpha_1\quad ...\quad w_{m-1}=\alpha_{m-1}$$

$$w_{i+1}=a_{m-1}w_i+a_{m-2}w_{i-1}+...+a_0w_{i+1-m}+hF(t_i,h,w_{i+1},...,w_{i+1-m}),$$

for each $i = m-1, ..., N-1$ where $a_0, a_1, .., a_{m-1}$ are constants.
The local truncation error for a multi-step method expressed in this form is

$$\tau_{i+1}(h) = \frac{y(t_{i+1}) - a_{m-1}y(t_i) - a_{m-2}y(t_{i-1}) + ... - a_0 y(t_{i+1-m})}{h}$$
$$+F(t_i, h, y(t_{i+1}), ..., y(t_{i+1-m}))$$

for each $i = m-1, ..., N-1$.

Definition A multi-step method is **consistent** if both

$$\lim_{h \to 0} |\tau_i(h)| = 0 \quad \text{for all } i = m, ..., N$$

$$\lim_{h \to 0} |\alpha_i - y(t_i)| = 0 \quad \text{for all } i = 0, .., m-1$$

Definition A multi-step method is **convergent** if the solution to the difference equation approaches the solution of the differential equation as the step size approaches zero.

$$\lim_{h \to 0} |w_i - y(t_i)| = 0$$

Theorem 5.2.1. *Suppose the Initial Value Problem*

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

is approximated by an Adams predictor-corrector method with an m-step Adams-Bashforth predictor equation

$$w_{i+1} = w_i + h[b_{m-1} f(t_i, w_i) + ... + b_0 f(t_{i+1-m}, w_{i+1-m})]$$

with local truncation error $\tau_{i+1}(h)$ *and an (m-1)-step Adams-Moulton equation*

$$w_{i+1} = w_i + h[\hat{b}_{m-1} f(t_{i+1}, w_{i+1}) + ... + \hat{b}_0 f(t_{i+2-m}, w_{i+2-m})]$$

with local truncation error $\hat{\tau}_{i+1}(h)$. *In addition suppose that* $f(t, y)$ *and* $f_y(t, y)$ *are continuous on* $= \{(t, y) | a \leq t \leq b, -\infty < y < \infty\}$ *and that* fy *is bounded. Then the local truncation error* $\sigma_{i+1}(h)$ *of the predictor-corrector method is*

$$\sigma_{i+1}(h) = \hat{\tau}_{i+1}(h) + h\tau_{i+1}(h)\hat{b}_{m-1}\frac{\partial f}{\partial y}(t_{i+1}, \theta_{i+1})$$

where $\theta_{i+1} \in [0, h\tau_{i+1}(h)$.
Moreover, there exists constants k_1 *and* k_2 *such that*

$$|w_i - y(t_i)| \leq \left[\max_{0 \leq j \leq m-1} |w_j - y(t_j)| + k_1 \sigma(h)\right] e^{k_2(t_i - a)}.$$

where $\sigma(h) = \max_{m \leq i \leq N} |\sigma_i(h)|$.

Definition Associated with the difference equation

$$w_0 = \alpha \quad w_1 = \alpha_1 \quad ... \quad w_{m-1} = \alpha_{m-1}$$

$$w_{i+1} = a_{m-1}w_i + a_{m-2}w_{i-1} + ... + a_0 w_{i+1-m} + hF(t_i, h, w_{i+1}, ..., w_{i+1-m}),$$

is the characteristic equation given by

$$\lambda^m - a_{m-1}\lambda^{m-1} - a_{m-2}\lambda^{m-2} - ... - a_0 = 0$$

Definition Let $\lambda_1, ..., \lambda_m$ denote the roots of the that characteristic equation

$$\lambda^m - a_{m-1}\lambda^{m-1} - a_{m-2}\lambda^{m-2} - ... - a_0 = 0$$

associated with the multi-step difference method

$$w_0 = \alpha \quad w_1 = \alpha_1 \quad ... \quad w_{m-1} = \alpha_{m-1}$$

$$w_{i+1} = a_{m-1}w_i + a_{m-2}w_{i-1} + ... + a_0 w_{i+1-m} + hF(t_i, h, w_{i+1}, ..., w_{i+1-m}),$$

If $|\lambda_i| \leq 1$ for each $i = 1, ..., m$ and all roots with absolute value 1 are simple roots then the difference equation is said to satisfy the **root condition**.

Definition

1. Methods that satisfy the root condition and have $\lambda = 1$ as the only root of the characteristic equation of magnitude one are called **strongly stable**;
2. Methods that satisfy the root condition and have more than one distinct root with magnitude one are called **weakly stable**;
3. Methods that do not satisfy the root condition are called **unstable**.

Theorem 5.2.2. *A multi-step method of the form*

$$w_0 = \alpha \quad w_1 = \alpha_1 \quad ... \quad w_{m-1} = \alpha_{m-1}$$

$$w_{i+1} = a_{m-1}w_i + a_{m-2}w_{i-1} + ... + a_0 w_{i+1-m} + hF(t_i, h, w_{i+1}, ..., w_{i+1-m})$$

is stable iff it satisfies the root condition. Moreover if the difference method is consistent with the differential equation then the method is stable iff it is convergent.

Example 25

We have seen that the fourth order Adams-Bashforth method can be expressed as

$$w_{i+1} = a_{m-1}w_i + a_{m-2}w_{i-1} + ... + a_0 w_{i+1-m} + hF(t_i, h, w_{i+1}, w_i, ..., w_{i-3})$$

where

$$F(t_i, h, w_{i+1}, w_i, ..., w_{i-3}) =$$

$$\frac{1}{24}[55f(t_i, w_i) - 59f(t_{i-1}, w_{i-1}) + 37f(t_{i-2}, w_{i-2}) - 9f(t_{i-3}, w_{i-3})]$$

so $m = 4, a_0 = 0, a_1 = 0, a_2 = 0$ and $a_3 = 1$.
The characteristic equation is

$$\lambda^4 - \lambda^3 = \lambda^3(\lambda - 1) = 0$$

which has the roots $\lambda_1 = 1, \lambda_2 = 0, \lambda_3 = 0$ and $\lambda_4 = 0$.
It satisfies the root condition and is strongly stable.

Example 26

The explicit multi-step method given by

$$w_{i+1} = w_{i-3} + \frac{4h}{3}[2f(t_i, w_i) - f(t_{i-1}, w_{i-1}) + f(t_{i-2}, w_{i-2})]$$

has a characteristic equation

$$\lambda^4 - 1 = 0$$

which has the roots $\lambda_1 = 1, \lambda_2 = -1, \lambda_3 = i$ and $\lambda_4 = -i$, the method satisfies the root condition, but is only weakly stable.

Example 27

The explicit multi-step method given by

$$w_{i+1} = aw_{i-3} + \frac{4h}{3}[2f(t_i, w_i) - f(t_{i-1}, w_{i-1}) + f(t_{i-2}, w_{i-2})]$$

has a characteristic equation

$$\lambda^4 - a = 0$$

which has the roots $\lambda_1 = \sqrt[4]{a}, \lambda_2 = -\sqrt[4]{a}, \lambda_3 = i\sqrt[4]{a}$ and $\lambda_4 = -i\sqrt[4]{a}$, when $a > 1$ the method dose not satisfy the root condition, and hence is unstable.

Example 28

Solving the Initial Value Problem

$$y' = -0.5y^2 \quad y(0) = 1$$

Using a weakly stable method

$$w_{i+1} = w_{i-3} + \frac{4h}{3}[2w_i - w_{i-1} + w_{i-2}]$$

Using an two different unstable method

1.

$$w_{i+1} = 1.0001w_{i-3} + \frac{4h}{3}[2f(t_i, w_i) - f(t_{i-1}, w_{i-1}) + f(t_{i-2}, w_{i-2})]$$

2.

$$w_{i+1} = 1.5w_{i-3} + \frac{4h}{3}[2f(t_i, w_i) - f(t_{i-1}, w_{i-1}) + f(t_{i-2}, w_{i-2})]$$

$$\lambda^4 - a = 0$$

which has the roots $\lambda_1 = \sqrt[4]{a}, \lambda_2 = -\sqrt[4]{a}, \lambda_3 = i\sqrt[4]{a}$ and $\lambda_4 = -i\sqrt[4]{a}$, when $a > 1$ the method dose not satisfy the root condition, and hence is unstable.

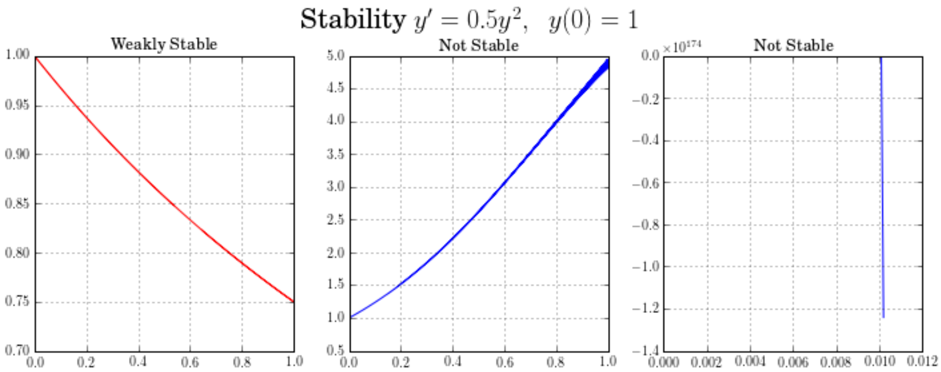


Figure 5.2.1: Python output: Left: Weakly stable solution, middle: unstable, right: very unstable

5.3 Problem Sheet 5 - Consistency, Convergence and Stability

1. Determine whether the 2-step Adams-Bashforth Method is consistent, stable and convergent

$$w_{n+1} = w_n + (\frac{3}{2}hf(t_n, w_n) - \frac{1}{2}hf(t_{n-1}, w_{n-1})),$$

2. Determine whether the 2-step Adams-Moulton Method is consistent, stable and convergent

$$w_{n+1} = w_n + \frac{5}{12}hf(t_{n+1}, w_{n+1}) + \frac{8}{12}hf(t_n, w_n) - \frac{1}{12}hf(t_{n-1}, w_{n-1}),$$

3. Determine whether the linear multistep following methods are consistent, stable and convergent

   a)

$$w_{n+1} = w_{n-1} + \frac{1}{3}h[f(t_{n+1}, w_{n+1}) + 4f(t_n, w_n) + f(t_{n-1}, w_{n-1})].$$

   b)

$$w_{n+1} = \frac{4}{3}w_n - \frac{1}{3}w_{n-1} + \frac{2}{3}h[f(t_{n+1}, w_{n+1})].$$

5.4 INITIAL VALUE PROBLEM REVIEW QUESTIONS

1. a) Derive the Euler approximation show it has a local truncation error of $O(h)$ of the Ordinary Differential Equation

$$y'(x) = f(x, y)$$

with initial condition

$$y(a) = \alpha.$$

[8 marks]

b) Suppose f is a continuous and satisfies a Lipschitz condition with constant L on $D = \{(t, y)|a \leq t \leq b, -\infty < y < \infty\}$ and that a constant M exists with the property that

$$|y''(t)| \leq M.$$

Let $y(t)$ denote the unique solution of the IVP

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

and $w_0, w_1, ..., w_N$ be the approx generated by the Euler method for some positive integer N. Then show for $i = 0, 1, ..., N$

$$|y(t_i) - w_i| \leq \frac{Mh}{2L}|e^{L(t_i - a)} - 1|$$

You may assume the two lemmas:
If s and t are positive real numbers $\{a_i\}_{i=0}^{N}$ is a sequence satisfying $a_0 \geq \frac{-t}{s}$ and $a_{i+1} \leq (1 + s)a_i + t$ then

$$a_{i+1} \leq e^{(i+1)s}\left(a_0 + \frac{t}{s}\right) - \frac{t}{s}$$

For all $x \geq 0.1$ and any positive m we have

$$0 \leq (1 + x)^m \leq e^{mx}$$

[17 marks]

c) Use Euler's method to estimate the solution of

$$y' = (1 - x)y^2 - y; \quad y(0) = 1$$

at x=1, using $h = 0.25$.

[8 marks]

2. a) Derive the difference equation for the Midpoint Runge-Kutta method

$$w_{n+1} = w_n + k_2$$
$$k_1 = hf(t_n, w_n)$$
$$k_2 = hf(t_n + \frac{1}{2}h, w_n + \frac{1}{2}k_1)$$

for solving the ordinary differential equation

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

by using a formula of the form

$$w_{n+1} = w_n + ak_1 + bk_2$$

where k_1 is defined as above,

$$k_2 = hf(t_n + \alpha h, w_n + \beta k_1)$$

and a, b, α and β are constants are determined. Prove that $a + b = 1$ and $b\alpha = b\beta = \frac{1}{2}$ and choose appropriate values to give the Midpoint Runge-Kutta method.

[18 marks]

b) Show that the midpoint Runge-Kutta method is stable.

[5 marks]

c) Use the Runge-Kutta method to approximate the solutions to the following initial value problem

$$y^{'} = 1 + (t - y)^2, \ \ 2 \leq t \leq 3, \ \ y(2) = 1$$

with $h = 0.2$ with the exact solution $y(t) = t + \frac{1}{1-t}$.

[10 marks]

3. a) Derive the two step Adams-Bashforth method:

$$w_{n+1} = w_n + (\frac{3}{2}hf(t_n, w_n) - \frac{1}{2}hf(t_{n-1}, w_{n-1})),$$

and the local truncation error

$$\tau_{n+1}(h) = -\frac{5h^2}{12}y^3(\mu_n)$$

[18 marks]

b) Apply the two step Adams-Bashforth method to approximate the solution of the initial value problem:

$$y' = ty - y, \quad (0 \leq t \leq 2) \quad y(0) = 1$$

. Using $N = 4$ steps, given that $y_1 = 0.6872$.

[15 marks]

4. a) Derive the Adams-Moulton two step method and its truncation error which is of the form

$$w_0 = \alpha_0 \quad w_1 = \alpha_1$$

$$w_{n+1} = w_n + \frac{h}{12}[5f(t_{n+1}, w_{n+1}) + 8f(t_n, w_{n1}) - f(t_{n-2}, w_{n-2})]$$

and the local truncation error

$$\tau_{n+1}(h) = -\frac{h^3}{24}y^4(\mu_n)$$

[23 marks]

b) Define the terms strongly stable, weakly stable and unstable with respect to the characteristic equation.

[5 marks]

c) Show that the Adams-Bashforth two step method is strongly stable.

[5 marks]

5. a) Given the initial value problem:

$$y' = f(t, y), \quad y(t_0) = y_0$$

and a numerical method which generates a numerical solution $(w_n)_{n=0}^{N}$, explain what it means for the method to be convergent.

[5 marks]

b) Using the 2-step Adams-Bashforth method:

$$w_{n+1} = w_n + \frac{3}{2}hf(t_n, w_n) - \frac{1}{2}hf(t_{n-1}, w_{n-1})$$

as a predictor, and the 2-step Adams-Moulton method:

$$w_{n+1} = w_n + \frac{h}{12}[5f(t_{n+1}, w_{n+1}) + 8f(t_n, w_{n1}) - f(t_{n-2}, w_{n-2})]$$

as a corrector, apply the 2-step Adams predictor-corrector method to approximate the solution of the initial value problem

$$y' = ty^3 - y, \quad (0 \leq t \leq 2), \ \ y(0) = 1$$

using N=4 steps, given $y_1 = 0.5$.

[18 marks]

c) Using the predictor corrector define a bound for the error by controlling the step size.

[10 marks]

6. a) Given the Midpoint point (Runge-Kutta) method

$$w_0 = y_0$$

$$w_{i+1} = w_i + hf(x_i + \frac{h}{2}, w_i + \frac{h}{2} f(x_i, w_i))$$

Assume that the Runge-Kutta method satisfies the Lipschitz condition. Then for the initial value problems

$$y' = f(x, y)$$

$$y(x_0) = Y_0$$

Show that the numerical solution $\{w_n\}$ satisfies

$$\max_{a \leq x \leq b} |y(x_n) - w_n| \leq e^{(b-a)L} |y_0 - w_0| + \left[\frac{e^{(b-a)L} - 1}{L}\right] \tau(h)$$

where

$$\tau(h) = \max_{a \leq x \leq b} |\tau_n(y)|$$

If the consistency condition

$$\delta(h) \to 0 \text{ as } h \to 0$$

where

$$\delta(h) = \max_{a \leq x \leq b} |f(x, y) - F(x, y; h; f)|$$

is satisfied then the numerical solution w_n converges to $Y(x_n)$.

[18 marks]

b) Consider the differential equation

$$y' - y + x - 2 = 0, \ \ 0 \leq x \leq 1, \ \ y(0) = 0.$$

Apply the midpoint method to approximate the solution at $y(0.4)$ using $h = 0.2$

[11 marks]

c) How would you improve on this result.

[4 marks]

Part III

NUMERICAL SOLUTIONS TO BOUNDARY VALUE PROBLEMS

6

BOUNDARY VALUE PROBLEMS

6.1 SYSTEMS OF EQUATIONS

An m-th order system of equation of first order Initial Value Problem can be expressed in the form

$$\begin{array}{c} \frac{du_1}{dt} = f_1(t, u_1, ..., u_m) \\ \frac{du_2}{dt} = f_2(t, u_1, ..., u_m) \\ . \\ . \\ . \\ \frac{du_m}{dt} = f_m(t, u_1, ..., u_m) \end{array} \tag{30}$$

for $a \leq t \leq b$ with the the initial conditions

$$\begin{array}{c} u_1(a) = \alpha_1 \\ u_2(a) = \alpha_2 \\ . \\ . \\ . \\ u_m(a) = \alpha_m. \end{array} \tag{31}$$

This can also be written in vector from

$$\mathbf{u}' = \mathbf{f}(t, \mathbf{u})$$

with initial conditions

$$\mathbf{u}(\mathbf{a}) = \mathbf{ff}.$$

Definition The function $f(t, u_1, ..., u_m)$ defined on the set

$$D = \{(t, u_1, .., u_m) | a \leq t \leq b, -\infty < u_i < \infty, i = 1, .., m\}$$

is said to be a **Lipschitz Condition** on D in the variables $u_1, ..., u_m$ if a constant L, the Lipschitz Constant, exists with the property that

$$|f(t, u_1, ..., u_m) - f(t, z_1, z_2, ..., z_m)| \leq L \sum_{j=1}^{m} |u_j - z_j|$$

for all $(t, u_1, ..., u_m)$ and $(t, z_1, z_2, ..., z_m)$ in D.

Theorem 6.1.1. *Suppose*

$$D = \{(t, u_1, .., u_m) | a \leq t \leq b, -\infty < u_i < \infty, i = 1, .., m\}$$

is continuous on D and satisfy a Lipschitz Condition. The system of 1st order equations subject th the initial conditions, has a unique solution $u_1(t), u_2(t), ..., u_m(t)$ *for* $a \leq t \leq b$.

Example 29

Using Euler method on the system

$$\begin{aligned} u' &= u^2 - 2uv \quad u(0) = 1 \\ v' &= tu + u^2 sinv \quad v(0) = -1 \end{aligned}$$

for $0 \leq t \leq 0.5$ and $h = 0.05$ the general Euler difference system of equations is of the form

$$\begin{aligned} \hat{u}_{i+1} &= \hat{u}_i + hf(t_i, \hat{u}_i, \hat{v}_i) \\ \hat{v}_{i+1} &= \hat{v}_i + hg(t_i, \hat{u}_i, \hat{v}_i) \end{aligned}$$

Applied the the Initial Value Problem

$$\begin{aligned} \hat{u}_{i+1} &= \hat{u}_i + 0.05(\hat{u}_i^2 - 2\hat{u}_i\hat{v}_i) \\ \hat{v}_{i+1} &= \hat{v}_i + 0.05(t_i\hat{u}_i + \hat{u}_i^2 sin(\hat{v}_i)) \end{aligned}$$

We know for $i = 0$, $\hat{u}_0 = 1$ and $\hat{v}_0 = -1$ from the initial conditions.
For i=0 we have

$$\begin{aligned} \hat{u}_1 &= \hat{u}_0 + 0.05(\hat{u}_0^2 - 2\hat{u}_0\hat{v}_0) = 1.15 \\ \hat{v}_1 &= \hat{v}_0 + 0.05(t_0\hat{u}_0 + \hat{u}_0^2 sin(\hat{v}_0)) = -1.042 \end{aligned}$$

and so forth.

6.2 Higher Order Equations

Definition A general mth order initial value problem

$$y^{(m)}(t) = f(t, y, ..., y^{(m-1)}) \quad a \leq t \leq b$$

with initial conditions

$$y(a) = \alpha_1, y^{'}(a) = \alpha_2, ..., y^{(m-1)}(a) = \alpha_m$$

can be converted into a system of equations as in (30) and (31)
Let $u_1(t) = y(t), u_1(t) = y^1(t), ..., u_m(t) = y^{(m-1)}(t)$. This produces the first order system of equations

$$\begin{aligned}
\frac{du_1}{dt} &= \frac{dy}{dt} = u_2 \\
\frac{du_2}{dt} &= \frac{dy'}{dt} = u_3 \\
&\vdots \\
\frac{du_{m-1}}{dt} &= \frac{dy^{(m-2)}}{dt} = u_m \\
\frac{du_m}{dt} &= \frac{dy^{(m-1)}}{dt} f_m(t, y, ..., y^{(m-1)}) = f(t, u_1, ..., u_m)
\end{aligned}$$

with initial conditions

$$\begin{aligned}
u_1(a) &= y(a) = \alpha_1 \\
u_2(a) &= y'(a) = \alpha_2 \\
&\vdots \\
u_m(a) &= y^{(m-1)}(a) = \alpha_m
\end{aligned}$$

Example 30

$$y'' + 3y' + 2y = e^t$$

with initial conditions $y(0) = 1$ and $y'(0) = 2$ can be converted to the system

$$\begin{aligned}
u' &= v & u(0) &= 1 \\
v' &= e^t - 2u - 3v & v(0) &= 2
\end{aligned}$$

the difference Euler equation is of the form

$$\begin{aligned}
\hat{u}_{i+1} &= \hat{u}_i + hv(t_i, \hat{u}_i, \hat{v}_i) \\
\hat{v}_{i+1} &= \hat{v}_i + h(e^{t_i} - 2\hat{u}_i - 3\hat{v}_i)
\end{aligned}$$

6.3 Boundary Value Problems

Consider the second order differential equation

$$y'' = f(x, y, y') \tag{32}$$

defined on an interval $a \leq x \leq b$. Here f is a function of three variables and y is an unknown. The general solution to 32 contains two arbitrary constants so in order to determine it uniquely it is necessary to impose two additional conditions on y. When one of these is given at $x = a$ and the other at $x = b$ the problem is called a boundary value problem

and associated conditions are called boundary conditions.
The simplest type of boundary conditions are

$$y(a) = \alpha$$

$$y(b) = \beta$$

for a given numbers α and β. However more general conditions such as

$$\lambda_1 y(a) + \lambda_2 y^{'}(a) = \alpha_1$$
$$\mu_1 y(b) + \mu_2 y^{'}(b) = \alpha_2$$

for given numbers α_i, λ_i and μ_i (i=1,2) are sometimes imposed.
Unlike Initial Value Problem whose problems are uniquely solvable boundary value problem can have no solution or many.

Example 31

The differential equation

$$y^{''} + y = 0$$

$$y_1(x) = y(x) \quad y_2(x) = y^{'}(x)$$
$$y_1^{'} = y_2$$
$$y_2^{'} = -y_1$$

It has the general solution

$$w(x) = C_1 sin(x) + C_2 cos(x)$$

where C_1, C_2 are constants.
The special solution $w(x) = sin(x)$ is the only solution that satisfies

$$w(0) = 0 \quad w(\frac{\pi}{2}) = 1$$

All functions of the form $w(x) = C_1 sin(x)$ where C_1 is an arbitrary constant, satisfies

$$w(0) = 0 \quad w(\pi) = 0$$

while there is no solution for the boundary conditions

$$w(0) = 0 \quad w(\pi) = 1.$$

◇

While we cannot state that all boundary value problem are unique we can say a few things.

6.4 Some theorems about boundary value problem

Writing the general linear subset Boundary Value Problem

$$\begin{aligned} & y'' = p(x)y' + q(x)y + g(x) \quad a < x < b \\ & A\begin{pmatrix} y(a) \\ y'(a) \end{pmatrix} + B\begin{pmatrix} y(b) \\ y'(b) \end{pmatrix} = \begin{pmatrix} \gamma_1 \\ \gamma_2 \end{pmatrix} \end{aligned} \tag{33}$$

The homogeneous problem is the case in which $g(x)$ and $\gamma_1 = \gamma_2 = 0$.

Theorem 6.4.1. *The non-homogeneous problem (33) has a unique solution $y(x)$ on $[a, b]$ for each set of given $\{g(x), \gamma_1, \gamma_2\}$ if and only if the homogeneous problem has only the trivial solutions $y(x) = 0$.*

For conditions under which the homogeneous problem (33) has only the zero solution we consider the following subset of problem

$$\begin{aligned} & y'' = p(x)y' + q(x)y + g(x) \quad a < x < b \\ & a_0 y(a) - a_1 y'(a) = \gamma_1 \\ & b_0 y(b) + b_1 y'(b) = \gamma_2 \end{aligned} \tag{34}$$

Assume the following conditions

$$\begin{aligned} & q(x) > 0 \quad a \le x \le b \\ & a_0, a_1 \ge 0 \quad b_0, b_1 \ge 0 \end{aligned} \tag{35}$$

$|a_1| + |a_0| \neq 0, |b_1| + |b_0| \neq 0, |a_0| + |b_0| \neq 0$ Then the homogeneous problem for (34) has only the zero solution therefore the theorem is applicable and the non-homogeneous problem has a unique solution for each set of data $\{g(x), \gamma_1, \gamma_2\}$.
The theory for a non-linear problem is far more complicated than that of a linear problem. Looking at the class of problems

$$\begin{aligned} & y'' = f(x, y, y') \quad a < x < b \\ & a_0 y(a) - a_1 y'(a) = \gamma_1 \\ & b_0 y(b) + b_1 y'(b) = \gamma_2 \end{aligned} \tag{36}$$

The function f is assumed to satisfy the following Lipschitz Condition

$$\begin{aligned} & |f(x, u_1, v_1) - f(x, u_2, v_2)| \le K_1 |u_1 - u_2| \\ & |f(x, u_1, v_1) - f(x, u_2, v_2)| \le K_2 |v_1 - v_2| \end{aligned} \tag{37}$$

for all points in the region

$$R = \{(x, u, v) | a \le x \le b, -\infty < u, v < \infty\}$$

Theorem 6.4.2. *The problem (36) assumes $f(x, u, v)$ is continuous on the region R and it satisfies the Lipschitz condition (37). In addition assume that f, on R, satisfies*

$$\frac{\partial f(x,u,v)}{\partial u} > 0 \quad \left|\frac{\partial f(x,u,v)}{\partial v}\right| \leq M$$

for some constant $M > 0$ for the boundary conditions of 36 assume that $|a_1| + |a_0| \neq 0, |b_1| + |b_0| \neq 0, |a_0| + |b_0| \neq 0$. The boundary value problem has a unique solution.

Example 32

The boundary value problem boundary value problem

$$y^{''} + e^{-xy} + sin(y^{'}) = 0 \quad 1 < x < 2$$

with $y(1) = y(2) = 0$, has

$$f(x, y, y^{'}) = -e^{-xy} - sin(y^{'})$$

Since

$$\frac{\partial f(x,y,y^{'})}{\partial y} = xe^{xy} > 0$$

and

$$\left|\frac{\partial f(x,y,y^{'})}{\partial y^{'}}\right| = |-cos(y^{'}) \leq 1$$

this problem has a unique solution. ◇

6.5 SHOOTING METHODS

The principal of the shooting method is to change our original boundary value problem boundary value problem into 2 Initial Value Problem.

6.5.1 *Linear Shooting method*

Looking at problem class (34). We break this down into two Initial Value Problem.

$$\begin{aligned} y_1^{''} &= p(x)y_1^{'} + q(x)y_1 + r(x), \quad y_1(a) = \alpha, \quad y_1^{'}(a) = 0 \\ y_2^{''} &= p(x)y_2^{'} + q(x)y_2, \quad y_2(a) = 0, \quad y_2^{'}(a) = 1 \end{aligned} \tag{38}$$

combining these results together to get the unique solution

$$y(x) = y_1(x) + \frac{\beta - y_1(b)}{y_2(b)}y_2(x) \tag{39}$$

provided that $y_2(b) \neq 0$.

Example 33

$$y'' = 2y' + 3y - 6$$

with boundary conditions

$$y(0) = 3$$

$$y(1) = e^3 + 2$$

The exact solution is

$$y = e^{3x} + 2$$

breaking this boundary value problem into two Initial Value Problem's

$$y_1'' = 2y_1' + 3y_1 - 6 \quad y_1(a) = 3, \quad y_1'(a) = 0 \tag{40}$$

$$y_2'' = 2y_2' + 3y_2 \quad y_2(a) = 0, \quad y_2'(a) = 1 \tag{41}$$

Discretising (40)

$$y_1 = u_1 \quad y_1' = u_2$$

$$u_1' = u_2 \quad u_1(a) = 3$$

$$u_2' = 2u_2 + 3u_1 - 6 \quad u_2(a) = 0$$

using the Euler method we have the two difference equations

$$u_{1i+1} = u_{1i} + hu_{2i}$$

$$u_{2i+1} = u_{2i} + h(2u_{2i} + 3u_{1i} - 6)$$

Discretising (41)

$$y_2 = w_1 \quad y_2' = w_2$$

$$w_1' = w_2 \quad w_1(a) = 0$$

$$w_2' = 2w_2 + 3w_1 \quad w_2(a) = 1$$

using the Euler method we have the two difference equations

$$w_{1i+1} = w_{1i} + hw_{2i}$$

$$w_{2i+1} = w_{2i} + h(2w_{2i} + 3w_{1i})$$

combining all these to get our solution

$$y_i = u_{1i} + \frac{\beta - u_1(b)}{w_1(b)} w_{1i}$$

It can be said

$$|y_i - y(x_i)| \leq Kh^n \left|1 + \frac{w_{1i}}{u_{1i}}\right|$$

$O(h^n)$ is the order of the method.

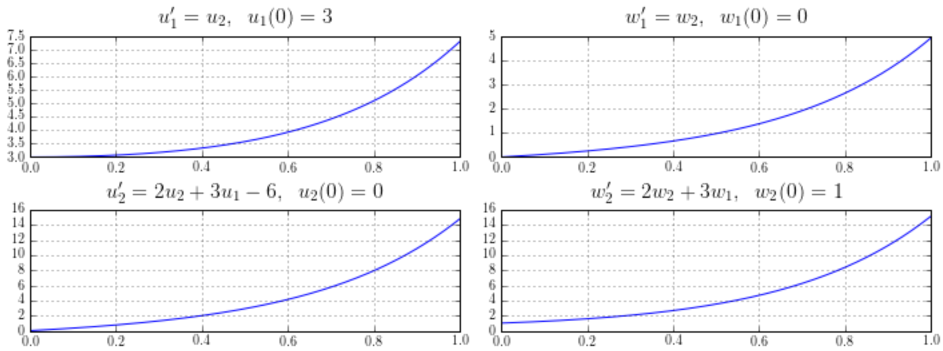


Figure 6.5.1: Python output: Shooting Method

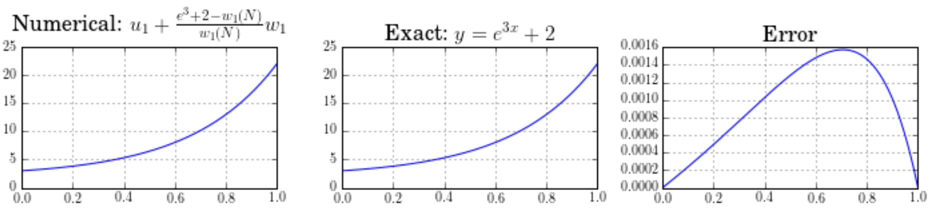


Figure 6.5.2: Python output: Shooting Method error

6.5.2 *The Shooting method for non-linear equations*

Example 34

$$y^{''} = -2yy^{'} \quad y(0) = 0 \;\; y(1) = 1$$

The corresponding initial value problem is

$$y^{''} = -2yy^{'} \quad y(0) = 0 \;\; y^{'}(0) = \lambda \tag{42}$$

Which reduces to the first order system, letting $y_1 = y$ and $y_2 = y^{'}$.

$$y_1^{'} = y_2 \quad y_1(0) = 0$$

$$y_2' = -2y_1y_2 \quad y_2'(0) = \lambda$$

Taking $\lambda = 1$ and $\lambda = 2$ as the first and second guess of $y'(0)$. (42) depends on two variable x and λ. $\diamond$

How to choose λ?
Our goal is to choose λ such that,

$$F(\lambda) = y(b, \lambda) - \beta = 0.$$

We use Newton's method to generate the sequence λ_k with only the initial approx λ_0. The iteration has the form

$$\lambda_k = \lambda_{k-1} - \frac{y(b, \lambda_{k-1}) - \beta}{\frac{dy}{d\lambda}(b, \lambda_{k-1})}$$

and requires knowledge of $\frac{dy}{d\lambda}(b, \lambda_{k-1})$. This present a difficulty since an explicit representation for $y(b, \lambda)$ is unknown we only know $y(b, \lambda_0), y(b, \lambda_1), ..., y(b, \lambda_{k-1})$.
Rewriting our Initial Value Problem we have it so that it depends on both x and λ.

$$y''(x, \lambda) = f(x, y(x, \lambda), y'(x, \lambda)), \quad a \leq x \leq b,$$

with the initial conditions,

$$y(a, \lambda) = \alpha \quad y'(a, \lambda) = \lambda,$$

differentiating with respect to λ and let $z(x, \lambda)$ denote $\frac{\partial y}{\partial \lambda}(x, \lambda)$ we have,

$$\frac{\partial}{\partial \lambda}(y'') = \frac{\partial f}{\partial \lambda} = \frac{\partial f}{\partial y}\frac{\partial y}{\partial \lambda} + \frac{\partial f}{\partial y'}\frac{\partial y'}{\partial \lambda}.$$

Now

$$\frac{\partial y'}{\partial \lambda} = \frac{\partial}{\partial \lambda}\frac{\partial y}{\partial x} = \frac{\partial}{\partial x}\left(\frac{\partial y}{\partial \lambda}\right) = \frac{\partial z}{\partial x} = z',$$

we have

$$z''(x, \lambda) = \frac{\partial f}{\partial y} z(x, \lambda) + \frac{\partial f}{\partial y'} z'(x, \lambda)$$

for $a \leq x \leq b$ and the initial conditions,

$$z(a, \lambda) = 0, \quad z'(a, \lambda) = 1$$

Now we have

$$\lambda_k = \lambda_{k-1} - \frac{y(b, \lambda_{k-1}) - \beta}{z(b, \lambda_{k-1})}$$

We can solve the original non-linear subset Boundary Value Problem by solving the two Initial Value Problem's.

Example 35

(cont.)

$$\frac{\partial f}{\partial y} = -2y^{'} \quad \frac{\partial f}{\partial y^{'}} = -2y$$

We now have the two Initial Value Problem's

$$y^{''} = -2yy^{'} \quad y(0) = 0 \;\; y^{'}(0) = \lambda$$

$$\begin{aligned} z^{''} &= \frac{\partial f}{\partial y} z(x,\lambda) + \frac{\partial f}{\partial y^{'}} z^{'}(x,\lambda) \\ &= -2y^{'}z - 2yz^{'} \quad z(0) = 0 \;\; z^{'}(0) = 1 \end{aligned}$$

Discretising we let $y_1 = y$, $y_2 = y^{'}$, $y_3 = z$ and $y_4 = z^{'}$.

$$\begin{aligned} y_1^{'} = y_2 &\quad : \quad y_1(0) = 0 \\ y_2^{'} = -2y_1y_2 &\quad : \quad y_2(0) = \lambda_k \\ z_1^{'} = z_2 &\quad : \quad z_1(0) = 0 \\ z_2^{'} = -2z_1y_2 - 2y_1z_2 &\quad : \quad y_2(0) = 1 \end{aligned}$$

with

$$\lambda_k = \lambda_{k-1} - \frac{y_1(b) - \beta}{y_3(b)}$$

Then solve using a one step method. ⋄

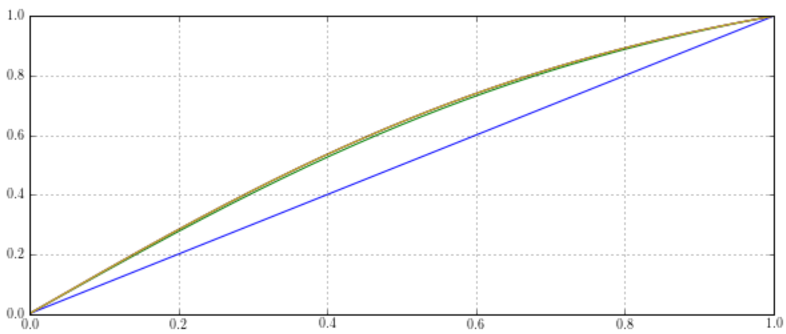


Figure 6.5.3: Python output: Nonlinear Shooting Method

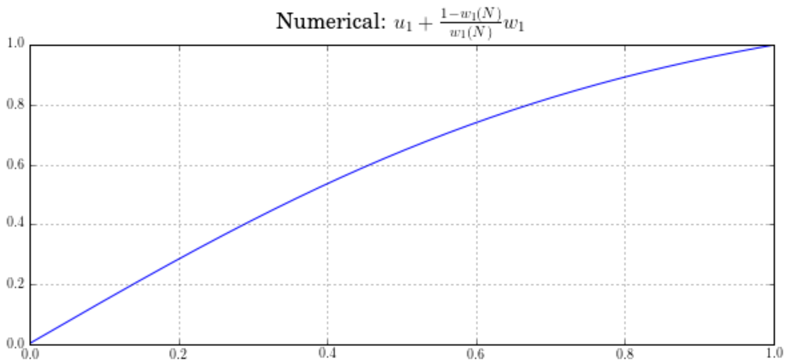


Figure 6.5.4: Python output: Nonlinear Shooting Method result

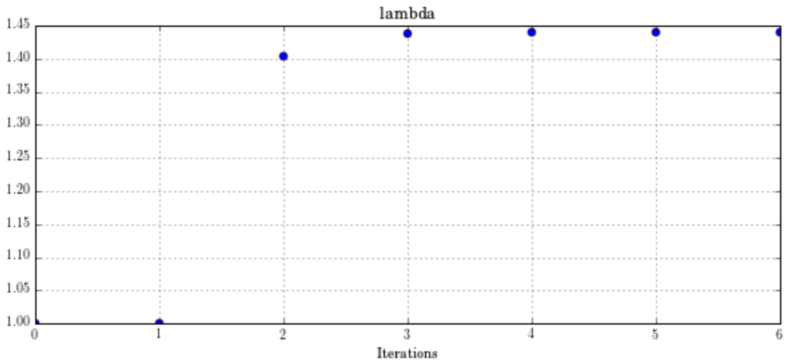


Figure 6.5.5: Python output: Nonlinear Shooting Method λ

6.6 FINITE DIFFERENCE METHOD

Each finite difference operator can be derived from Taylor expansion. Once again looking at a linear second order differential equation

$$y^{''} = p(x)y^{'} + q(x)y + r(x)$$

on $[a, b]$ subject to boundary conditions

$$y(a) = \alpha \quad y(b) = \beta$$

As with all cases we divide the area into even spaced mesh points

$$x_0 = a, \quad x_N = b \quad x_i = x_0 + ih \quad h = \frac{b-a}{N}$$

We now replace the derivatives $y^{'}(x)$ and $y^{''}(x)$ with the centered difference approximations

$$y^{'}(x) = \frac{1}{2h}(y(x_{i+1}) - y(x_{i-1})) - \frac{h^2}{12}y^3(\xi_i)$$

$$y''(x) = \frac{1}{h^2}(y(x_{i+1}) - 2y(x_i) + y(x_{i-1})) - \frac{h^2}{6}y^4(\mu_i)$$

for some $x_{i-1} \leq \xi_i \mu_i \leq x_{i+1}$ for i=1,...,N-1.
We now have the equation

$$\frac{1}{h^2}(y(x_{i+1}) - 2y(x_i) + y(x_{i-1})) = p(x_i)\frac{1}{2h}(y(x_{i+1}) - y(x_{i-1})) + q(x_i)y(x_i) + r(x_i)$$

This is rearranged such that we have all the unknown together,

$$\left(1 + \frac{hp(x_i)}{2}\right) y(x_{i-1}) - (2 + h^2 q(x_i))y(x_i) + \left(1 - \frac{hp(x_i)}{2}\right) y(x_{i+1}) = h^2 r(x_i)$$

for $i = 1, .., N-1$.
Since the values of $p(x_i), q(x_i)$ and $r(x_i)$ are known it represents a linear algebraic equation involving $y(x_{i-1}), y(x_i), y(x_{i+1})$.
This produces a system of $N-1$ linear equations with $N-1$ unknowns $y(x_1), ..., y(x_{N-1})$.
The first equation corresponding to $i = 1$ simplifies to

$$-(2 + h^2 q(x_1))y(x_1) + \left(1 - \frac{hp(x_1)}{2}\right) y(x_2) = h^2 r(x_1) - \left(1 + \frac{hp(x_1)}{2}\right) \alpha$$

because of the boundary condition $y(a) = \alpha$, and for $i = N-1$

$$\left(1 + \frac{hp(x_{N-1})}{2}\right) y(x_{N-2}) - (2 + h^2 q(x_{N-1}))y(x_{N-1}) = h^2 r(x_{N-1}) - \left(1 - \frac{hp(x_{N-1})}{2}\right) \beta$$

because $y(b) = \beta$.
The values of y_i, $(i = 1, ..., N-1)$ can therefore be found by solving the tridiagonal system

$$A\mathbf{y} = \mathbf{b}$$

where

$$A =$$

$$\begin{bmatrix}
-(2+h^2q(x_1)) & \left(1-\frac{hp(x_1)}{2}\right) & 0 & . & 0 \\
\left(1+\frac{hp(x_2)}{2}\right) & -(2+h^2q(x_2)) & \left(1-\frac{hp(x_2)}{2}\right) & 0 & . \\
0 & . & . & 0 & 0 \\
. & . & . & . & . \\
. & . & . & . & . \\
. & 0 & \left(1+\frac{hp(x_{N-2})}{2}\right) & -(2+h^2q(x_{N-2})) & \left(1-\frac{hp(x_{N-2})}{2}\right) \\
. & 0 & 0 & \left(1+\frac{hp(x_{N-1})}{2}\right) & -(2+h^2q(x_{N-1}))
\end{bmatrix}$$

$$y = \begin{pmatrix} y_1 \\ y_2 \\ . \\ . \\ y_{N-2} \\ y_{N-1} \end{pmatrix} \quad b = \begin{pmatrix} h^2 r_1 - \left(1 + \frac{hp_1}{2}\right)\alpha \\ h^2 r_2 \\ . \\ . \\ h^2 r_{N-2} \\ h^2 r_{N-1} - \left(1 - \frac{hp_1}{2}\right)\beta \end{pmatrix}$$

Example 36

Looking at the simple case

$$\frac{d^2y}{dx^2} = 4y, \quad y(0) = 1.1752, \quad y(1) = 10.0179.$$

Our difference equation is

$$\frac{1}{h^2}(y(x_{i+1}) - 2y(x_i) + y(x_{i-1})) = 4y(x_i) \quad i = 1, .., N-1$$

dividing $[0, 1]$ into 4 subintervals we have $h = \frac{1-0}{4}$

$$x_i = x_0 + ih = 0 + i(0.25)$$

In this simple example $q(x) = 4$, $p(x) = 0$ and $r(x) = 0$. Rearranging the equation we have

$$\frac{1}{h^2}(y(x_{i+1})) - \left(\frac{2}{h^2} + 4\right) y(x_i) + \frac{1}{h^2}(y(x_{i-1})) = 0$$

multiplying across by h^2

$$y(x_{i+1}) - (2 + 4h^2)y(x_i) + (y(x_{i-1})) = 0$$

with the boundary conditions $y(x_0) = 1.1752$ and $y(x_4) = 10.0179$. Our equations are of the form

$$\begin{aligned} y(x_2) - 2.25y(x_1) &= -1.1752 \\ y(x_3) - 2.25y(x_2) + y(x_1) &= 0 \\ -2.25y(x_3) + y(x_2) &= -10.0179 \end{aligned}$$

Putting this into matrix form

$$\begin{pmatrix} -2.25 & 1 & 0 \\ 1 & -2.25 & 1 \\ 0 & 1 & -2.25 \end{pmatrix} \begin{pmatrix} y_1 \\ y_2 \\ y_3 \end{pmatrix} = \begin{pmatrix} -1.1752 \\ 0 \\ -10.0179 \end{pmatrix}$$

x	y	Exact $sinh(2x+1)$
0	1.1752	1.1752
0.25	2.1467	2.1293
0.5	3.6549	3.6269
0.75	6.0768	6.0502
1.0	10.0179	10.0179

◇

Example 37

Looking at a more involved boundary value problem

$$y'' = xy' - 3y + e^x \quad y(0) = 1 \;\; y(1) = 2$$

Let N=5 then $h = \frac{1-0}{5} = 0.2$. The difference equation is of the form

$$\frac{1}{h^2}(y(x_{i+1}) - 2y(x_i) + y(x_{i-1})) = x_i \frac{1}{2h}(y(x_{i+1}) - y(x_{i-1})) - 3y(x_i) + e^{x_i}$$

Re arranging and putting $h = 0.2$

$$(1 + \frac{0.2(x_i)}{2})y(x_{i-1}) - (1.88)y(x_i) + (1 - \frac{0.2(x_i)}{2})y(x_{i+1}) = 0.04e^{x_i}$$

In matrix form this is

$$\begin{pmatrix} -1.88 & 0.98 & 0 & 0 \\ 1.04 & -1.88 & 0.96 & 0 \\ 0 & 1.06 & -1.88 & 0.94 \\ 0 & 0 & 1.08 & -1.88 \end{pmatrix} \begin{pmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \end{pmatrix} = \begin{pmatrix} 0.04e^{0.2} - 1.02 \\ 0.04e^{0.4} \\ 0.04e^{0.6} \\ 0.04e^{0.8} - 1.84 \end{pmatrix}$$

$y_1 = 1.4651$, $y_2 = 1.8196$, $y_3 = 2.0283$ and $y_4 = 2.1023$.

Solving a tri-diagonal system

To solve a tri-diagonal system we can use the method discussed in the approximation theory.

6.7 Problem Sheet 6 - Boundary Value Problems

1. Consider the boundary value problem

$$y'' = 3xy' - 4y + x^2, \;\; 1 \leq x \leq 2, \;\; y(1) = 2, \quad y(2) = -1.$$

Apply the linear shooting method to transform this equation into two second order initial value problems and approximate the solution using the Euler method with stepsize $h = \frac{1}{3}$.

Part IV

MACHINE LEARNING METHODS FOR DIFFERENTIAL EQUATIONS

BIBLIOGRAPHY

[1] Ian Jaques and Colin Judd, *Numerical Anaylsis*, Chapman and Hall Ltd, 1987

[2] J. Stoer and R. Bulirsch, *Introduction to Numerical Analysis*, Springer-Verlag 1980

[3] Walter Gautschi, *Numerical Analysis an Introduction*, Birkhauser 1997

[4] Curtis F. Gerald, Patrick O. Wheatley, *Applied Numerical Analysis*, Addison-Wesley 1994

[5] Richard L. Burden, J. Douglas Faires, *Numerical Analysis*, Brooks/Cole 1997

[6] Myron B Allen III, Eli L. Isaacson *Numerical Analysis for Applied Science* John Wiley and Sons, Inc. 1997

[7] G D Smith *Numerical Solution of Partial Differential Equations: Finite Difference Method* Oxford 1992

[8] Atkinson, Han *Elementary Numerical Analysis*

[9] Alfio Quarteroni, Riccardo Sacco, Fausto Saleri*Numerical Mathematics*

[10] John H Mathews, Kurtis Fink *Numerical Method using MATLAB*

[11] Brian Bradie, *A Friendly Introduction to Numerical Analysis*