

8.1 Introduction

Consider the *initial value problem (IVP)*:

$$y' \equiv \frac{dy}{dt} = f(t, y), \quad y(t_0) = y_0, \quad t_0 \leq t \leq T.$$

Geometrically: solutions are a one parameter family of curves $y = y(t)$ in (t, y) -plane.

Assume solution $y = Y(t)$ exists & is unique \Rightarrow exactly one solution curve passes through each point (t_0, y_0) .

Example 8.1

$$\frac{dy}{dt} = t y^{1/3}, \quad y(1) = 1.$$

General solution:

$$y = \pm \left(\frac{t^2 + C}{3} \right)^{3/2}.$$

Different C give different solution curves. $C = 2$ for solution curve through $(1, 1)$:

$$Y = \left(\frac{t^2 + 2}{3} \right)^{3/2}.$$

E.g. $Y(1.1) = (1.07)^{3/2} = 1.106816606 \dots$

If family of solutions diverge as t increases from t_0 IVP is *ill-conditioned (unstable)*, otherwise *well-conditioned stable*.

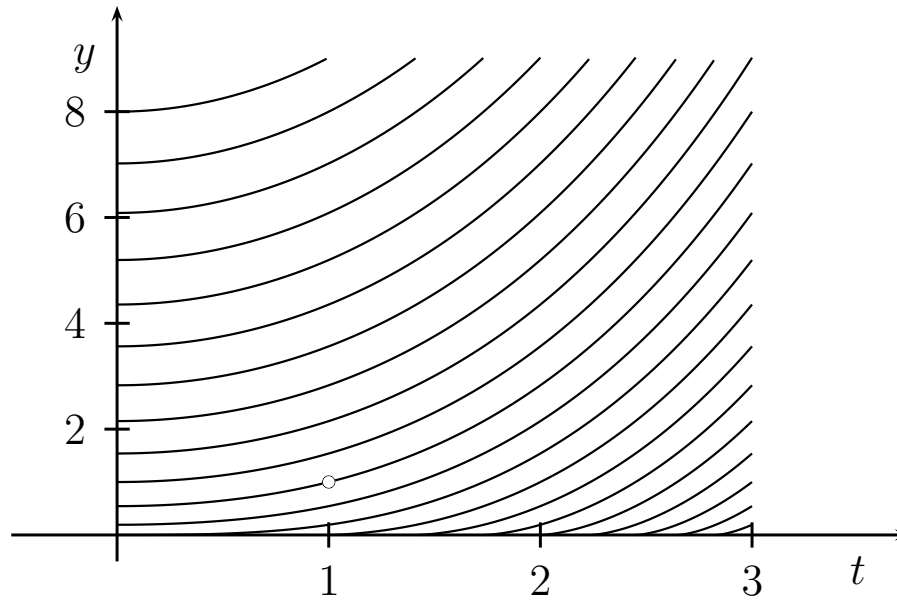


Figure 8.1: Solution curves of Example 8.1 for $C = -8, -7, \dots, 12$. The circle is the initial point $(1, 1)$.

IVP (linearly) well-conditioned if $J = \partial f / \partial y < 0$ in $t_0 \leq t \leq T$ (ode unperturbed).

Let $y = Y(t)$ be perturbed at $t = \tau$ to $Y(\tau) + \delta$. If no further perturbation, $y = Y(t) + \eta(t)$, where

$$\eta' = f(t, Y + \eta) - Y' = f(t, Y + \eta) - f(t, Y) = f_y(t, \xi)\eta = J\eta, \quad \eta(\tau) = \delta, \quad \tau \leq t \leq T,$$

where ξ lies between Y & $Y + \eta$, by MVT. Hence

$$\eta(t) = \delta e^{\int_{\tau}^t J dt}.$$

$$J < 0 \quad \Rightarrow \quad |\eta(t)| < |\delta|, \quad t > \tau,$$

i.e. disturbance does not grow.

$J \leq J_0 < 0$, J_0 constant, $\Rightarrow \eta(t) \rightarrow 0$ as $t \rightarrow \infty$.

$J \geq 0 \Rightarrow$ perturbations grow \Rightarrow IVP ill-conditioned.

IVP is *stiff* on $[t_0, T]$ if solution slowly varies over most of $[t_0, T]$, $J < 0$ & $J \gg 1$. E.g.

$$y' = -1000(y - \sin t) + \cos t, \quad y(0) = 1, \quad 0 \leq t \leq \pi.$$

$$\text{Solution: } y = e^{-1000t} + \sin t.$$

Varies slowly like $\sin t$ over most of $[0, \pi]$ but drops rapidly to zero near $t = 0$. Not stiff on $[0, 0.005]$.

Expansion methods: determine constants c_i where

$$Y(t) \approx \sum_{i=1}^N c_i \phi_i(t)$$

in terms of known basis functions $\{\phi_i(t)\}$, e.g. polynomial, trigonometric, rational, orthogonal functions.

Discretisation methods: approximate $Y(t)$ at t_1, t_2, \dots, t_n by y_1, y_2, \dots, y_n , where $y_i \approx Y(t_i)$ & usually $y_i \neq y(t_i)$, $i = 1, 2, \dots, n$. Ignore roundoff error for the present. $f_i \equiv f(t_i, y_i)$, $Y_i \equiv Y(t_i)$, $Y'_i \equiv Y'(t_i)$. In general, y_{k+1} is calculated from y_1, y_2, \dots, y_k and f_1, f_2, \dots, f_k , which have already been computed, and y_0 . Method is *m-step* if y_{k+1} is calculated from $y_k, y_{k-1}, \dots, y_{k-m+1}$ & $f_k, f_{k-1}, \dots, f_{k-m+1}$. **Not self-starting if $m > 1$.**

In general, (t_1, y_1) does not lie on solution curve. Even if no further errors are made y_2, y_3, \dots, y_n will lie on the wrong solution curve.

The error in the first step has propagated to the succeeding steps. In practice, errors occur at every step; e.g. (t_2, y_2) does not lie on the solution curve through (t_1, y_1) . However, the numerical

solution will be acceptable provided all the points (t_i, y_i) , $i = 1, 2, \dots, n$, lie closer than some given tolerance to the solution curve through (t_0, y_0) .

The computed solution \tilde{y}_k includes roundoff error, usually a secondary consideration since $|y_k - \tilde{y}_k| \ll |Y_k - y_k|$.

Can derive discretisation methods using various techniques. E.g. integration of the differential equation,

$$y(t) = y_0 + \int_{t_0}^t f(t, y) dt.$$

Unknown y occurs in integrand; compare numerical quadrature.

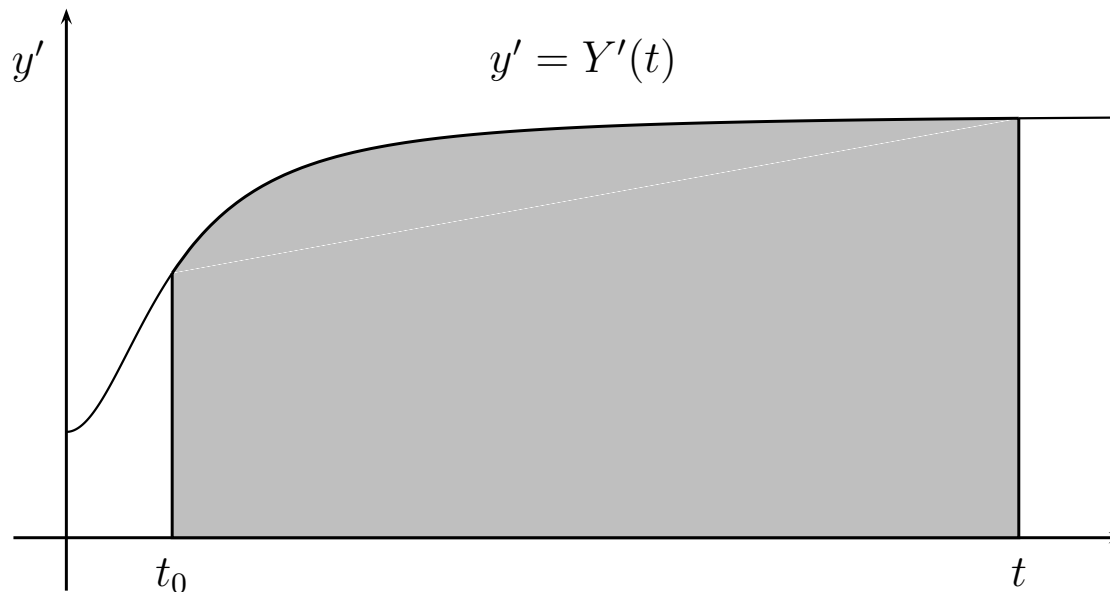


Figure 8.2: Geometric interpretation of quadrature methods.

Example 8.2

$$\int_{t_0}^{t_1} f(t, y) dt \approx (t_1 - t_0)f(t_0, y_0), \quad y(t_1) \approx y_0 + (t_1 - t_0)f(t_0, y_0).$$

Repeat to get *Euler's method*,

$$y_{k+1} = y_k + hf(t_k, y_k).$$

$h = t_{k+1} - t_k$ is the *step-size*. *Backward Euler method*:

$$\int_{t_0}^{t_1} f(t, y) dt \approx (t_1 - t_0)f(t_1, y_1), \quad y_{k+1} = y_k + hf(t_{k+1}, y_{k+1}).$$

The trapezoidal rule gives the *trapezoidal rule*,

$$y_{k+1} = y_k + \frac{h}{2} \{f(t_k, y_k) + f(t_{k+1}, y_{k+1})\}.$$

Simpson's $\frac{1}{3}$ -rule gives *Milne's method*,

$$y_{k+1} = y_{k-1} + \frac{h}{3} \{f(t_{k-1}, y_{k-1}) + 4f(t_k, y_k) + f(t_{k+1}, y_{k+1})\}.$$

Not all good methods. E.g. Milne's method: more accurate than others above but (weakly) unstable.

Discretisation methods must be convergent & consistent.

1. Method is *convergent* if

$$y_k(h) \rightarrow Y(t), \quad \text{as } h_{\max} \rightarrow 0, k \rightarrow \infty, t_k(h) \rightarrow t,$$

where $h_{\max} = \max_i h_i$. Smaller steps give approximate solutions closer to exact solution. Larger number of steps to reach fixed value of t . Round-off error ignored.

2. Method is *consistent* if

$$\frac{y_{k+1}(h) - y_k(h)}{h_{k+1}} \rightarrow Y'(t), \quad \text{as } h_{\max} \rightarrow 0, k \rightarrow \infty, t_k(h) \rightarrow t.$$

3. *Local truncation error* or *local error* of method at t_k is

$$e_k^T := \bar{y}_k - Y_k,$$

where \bar{y}_k is determined using method assuming preceding values of y_i are exact, i.e. $y_i = Y_i$ for $i = 0, 1, \dots, k-1$.

4. Method is *order p* if local truncation error = $O(h^{p+1})$.

5. *Propagated error* at t_k is

$$e_k^P := y_k - \bar{y}_k.$$

Propagated error vanishes if no error at preceding steps t_i , $i = 0, 1, 2, \dots, k-1$.

6. *Global truncation error* or *global error* at t_k is

$$e_k := y_k - Y_k = (y_k - \bar{y}_k) + (\bar{y}_k - Y_k) = e_k^P + e_k^T.$$

7. *Stability*: method may amplify local truncation error at each step to eventually swamp true solution.

If propagated error is not amplified at each step, i.e. $|e_k^P| < |e_{k-1}^P|$, method is *stable*; otherwise unstable.

Linear stability: approximate ode $y' = f(t, y)$ locally by

$$(y - Y)' = f(t, y) - f(t, Y) = f_y(t, \xi)(y - Y).$$

“Locally” means for $h \ll \mathcal{L}$, where $\mathcal{L} =$ length scale of $Y(t)$; $f_y(t, \xi)$ treated as a constant. (For systems replace f_y by Jacobian of system.) Thus stability is usually only considered for $f(t, y) = \lambda y$, where λ is a constant (complex for systems).

Methods in this course are convergent when stable.

8.2 Systems and Higher-Order Equations

Methods mostly generalise to *first-order systems*,

$$\begin{aligned}y_1' &= f_1(t, y_1, y_2, \dots, y_n) \\y_2' &= f_2(t, y_1, y_2, \dots, y_n) \\&\vdots \\y_n' &= f_n(t, y_1, y_2, \dots, y_n).\end{aligned}$$

with initial conditions

$$y_i(t_0) = y_i^0, \quad i = 1, 2, \dots, n.$$

In vector notation,

$$\mathbf{y}' = \mathbf{f}(t, \mathbf{y}), \quad \mathbf{y}(t_0) = \mathbf{y}_0.$$

A first-order system of n differential equations requires n scalar conditions to determine the n independent constants of integration. Conditions at one point give IVP. Conditions at two or more (up to n) different points give boundary-value problems.

Higher-order ode's & systems of higher-order ode's subject to initial conditions can be transformed to first-order systems. E.g. convert

$$u^{(n)} = f(t, u, u^{(1)}, u^{(2)}, \dots, u^{(n-1)}),$$

where $u^{(j)} = d^j u / dt^j$ ($j = 1, \dots, n$), to

$$\begin{aligned}y_1' &= y_2 \\y_2' &= y_3\end{aligned}$$

$$\begin{aligned} & \vdots \\ y'_{n-1} &= y_n \\ y_n &= f(t, y_1, y_2, \dots, y_{n-1}) \end{aligned}$$

by taking $y_1 = u, y_2 = u^{(1)}, y_3 = u^{(2)}, \dots, y_n = u^{(n-1)}$.

First-order system is (linearly) *well-conditioned* (or *stable*) if *Jacobian matrix*,

$$\mathbf{J}(y) = \begin{pmatrix} \frac{\partial f_1}{\partial y_1} & \frac{\partial f_1}{\partial y_2} & \dots & \frac{\partial f_1}{\partial y_n} \\ \frac{\partial f_2}{\partial y_1} & \frac{\partial f_2}{\partial y_2} & \dots & \frac{\partial f_2}{\partial y_n} \\ \vdots & \vdots & \dots & \vdots \\ \frac{\partial f_n}{\partial y_1} & \frac{\partial f_n}{\partial y_2} & \dots & \frac{\partial f_n}{\partial y_n} \end{pmatrix},$$

has only eigenvalues with negative real part. If \mathbf{J} has eigenvalues with positive real part then system is usually ill-conditioned. If $\boldsymbol{\eta}$ is a perturbation in \mathbf{y} , then

$$\boldsymbol{\eta}' = \mathbf{f}(t, \mathbf{y} + \boldsymbol{\eta}) - \mathbf{f}(t, \mathbf{y}) \approx \mathbf{J}\boldsymbol{\eta},$$

using a Taylor expansion. Perturbations $\boldsymbol{\eta}$ will be damped out if eigenvalues all have negative real parts.

If \mathbf{J} has eigenvalues with widely differing negative real parts system is *stiff*.

8.3 Truncated Taylor Series

Expand exact solution in 3-term Taylor series about t_0 ,

$$Y(t) = y(t_0) + Y'(t_0)(t - t_0) + \frac{1}{2}Y''(t_0)(t - t_0)^2 + \frac{1}{6}Y'''(\xi)(t - t_0)^3.$$

Determine $Y'(t_0)$ & $Y''(t_0)$ using ode: $Y'(t_0) = f(t_0, y_0)$.

To find $Y''(t_0)$ differentiate $y' = f(t, y)$:

$$y'' = \frac{\partial f}{\partial t} + \frac{\partial f}{\partial y} \frac{dy}{dt} = \frac{\partial f}{\partial t} + \frac{\partial f}{\partial y} f.$$

Thus $Y_0'' = f_t(t_0, y_0) + f_y(t_0, y_0)Y_0'$.

Example 8.3

$$\frac{dy}{dt} = t y^{1/3}, \quad y(1) = 1.$$

Four term Taylor expansion about $t = 1$.

Solution

$$\begin{aligned} y' &= t y^{1/3} \\ y'' &= y^{1/3} + \frac{1}{3}t y^{-2/3} y' \\ y''' &= \frac{2}{3} y^{-2/3} y' - \frac{2}{9}t y^{-5/3} (y')^2 + \frac{1}{3}t y^{-2/3} y'' \end{aligned}$$

$Y'(1) = 1$, $Y''(1) = 4/3$, $Y'''(1) = 8/9$. Hence

$$Y(t) = 1 + (t - 1) + \frac{2}{3}(t - 1)^2 + \frac{4}{27}(t - 1)^3 + \dots$$

$Y(1.1) \approx 1.106815$, 6 significant figures.

Accuracy of Taylor series worsens the further t is from t_0 . So expand in new Taylor series about t_1 near t_0 using y_1 from old series to approximate $Y(t_1)$. Repeat. Thus,

$$Y_{k+1} = Y_k + hY'_k + \frac{1}{2}h^2Y''_k + \frac{1}{6}h^3Y'''(\xi).$$

Express Y'_k and Y''_k in terms of f and its derivatives,

$$Y_{k+1} = Y_k + hf(t_k, Y_k) + \frac{1}{2}h^2\{f_t(t_k, Y_k) + f_y(t_k, Y_k)f(t_k, Y_k)\} + O(h^3).$$

The one-step three-term Taylor series discretisation method

$$\begin{aligned}y_{k+1} &= y_k + hy'_k + \frac{1}{2}h^2y''_k, \\ &= y_k + hf(t_k, y_k) + \frac{1}{2}h^2\{f_t(t_k, y_k) + f_y(t_k, y_k)f(t_k, y_k)\}.\end{aligned}$$

More terms in Taylor series can usually achieve greater accuracy but **disadvantages:**

$f(t, y)$ may be too complicated to differentiate;

differentiations increase chance of blunders;

Taylor series may not exist; e.g. $y' = t^{1/2}y$, $y(0) = 0$, $y''(0)$ does not exist.

Taylor series are not a general purpose method.

8.4 Euler's Method

No differentiation in Taylor series method if only two terms:

$$\text{Euler's method: } y_{k+1} = y_k + h y'_k = y_k + h f(t_k, y_k).$$

Poor accuracy requires very small step-size h & large number of steps.
Use more accurate methods unless storage limitations demand Euler.

Example 8.4

$$\frac{dy}{dt} = t y^{1/3}, \quad y(1) = 1.$$

Find $Y(1.1)$ ($= 1.1068\dots$ exactly) using Euler's method with (a) 1 step & (b) 10 steps.

Solution $t_0 = 1, y_0 = 1, f(t, y) = t y^{1/3}$.

$$(a) h = 0.1, y_1 = 1 + 0.1f(1, 1) = 1 + 0.1 \times 1 = 1.100 \Rightarrow Y(1.1) \approx 1.100.$$

$$(b) h = 0.01, Y(1.1) \approx 1.106118.$$

Taylor series of $Y(t)$ about t_k :

$$Y_{k+1} = Y_k + h f(t_k, Y_k) + \frac{1}{2} h^2 Y''(\xi_k), \quad t_k < \xi_k < t_{k+1}.$$

Subtract Euler's method to get the *global error* at t_{k+1} ,

$$e_{k+1} = Y_{k+1} - y_{k+1} = Y_k - y_k + h\{f(t_k, Y_k) - f(t_k, y_k)\} + \frac{1}{2} h^2 Y''(\xi_k).$$

Global error consists of two parts (ignore round-off error):

1. **Local error at t_{k+1} :** $e_{k+1}^T = \frac{1}{2}h^2 Y''(\xi_k)$.

$Y_k = y_k \Rightarrow$ global error = local truncation error.

2. **Propagated error at t_{k+1} :**

$$e_{k+1}^P = Y_k - y_k + h\{f(t_k, Y_k) - f(t_k, y_k)\} = (1 + hJ(\xi))(Y_k - y_k) = (1 + hJ)e_k,$$

where ξ lies between Y_k & y_k .

Thus

$$e_{k+1} \approx (1 + hJ)e_k + e_{k+1}^T.$$

Error decays if amplification factor $1 + hJ$ satisfies $|1 + hJ| < 1$, i.e. if $-2 < hJ < 0$.

Well-conditioned IVP $\Rightarrow J < 0 \Rightarrow$ Euler's method stable only if $0 < h < -2/J$.

Stiff IVP $\Rightarrow J < 0 \& |J| \gg 1 \Rightarrow$ very small step-size for stability.

Essential difficulty of stiff problems: very small steps to maintain stability.

Local error control:

$$y'' \approx y''_{k-1,k} := \frac{y'_k - y'_{k-1}}{t_k - t_{k-1}}.$$

Local error from t_k to $t_{k+1} < \epsilon$ if h chosen such that

$$\frac{1}{2}|Y''|h^2 \approx \frac{1}{2}|y''_{k-1,k}|h^2 \leq \epsilon \quad \text{or} \quad h \leq \sqrt{\frac{2\epsilon}{|y''_{k-1,k}|}}.$$

Local relative error control:

$$h \leq \min \left\{ 0.9 \sqrt{\frac{2\epsilon|y_k|}{|y''_{k-1,k}|}}, h_{\max} \right\} .$$

h_{\max} = maximum step-size; safety factor = 0.9; change step only every few steps. Check derivative approximation using computed value of y_{k+1}

$$y'' \approx y''_{k,k+1} := \frac{y'_{k+1} - y'_k}{t_{k+1} - t_k} .$$

Accept step change only if $\frac{1}{2}|y''_{k,k+1}|h^2$ small enough, otherwise reject & reduce h further.

Euler's method for systems:

stable if $|1 + h\lambda| < 1$ for any eigenvalue λ of system Jacobian matrix \mathbf{J} .

Geometrically: interior of circle, centre -1 , radius 1, in complex plane.

8.5 Runge-Kutta Methods

8.5.1 Explicit Two-Stage Runge-Kutta Methods

Achieve accuracy of any truncated Taylor series method without differentiation by extra evaluations of $f(t, y)$ at each step. E.g.

$$\begin{aligned}k_1 &= hf(t_k, y_k) \\k_2 &= hf(t_k + \alpha h, y_k + \beta k_1) \\y_{k+1} &= y_k + ak_1 + bk_2.\end{aligned}$$

Taylor expansion for function of two variables:

$$f(x_0 + x, y_0 + y) = f(x_0, y_0) + f_x(x_0, y_0)x + f_y(x_0, y_0)y + \dots$$

Replace x_0 , x , y_0 & y by t_k , αh , y_k and βh :

$$k_2 = hf(t_k, y_k) + h\{f_t(t_k, y_k)\alpha h + f_y(t_k, y_k)\beta k_1\} + O(h^3).$$

Substitute into method:

$$y_{k+1} = y_k + (a + b)hf(t_k, y_k) + h^2\{b\alpha f_t(t_k, y_k) + b\beta f_y(t_k, y_k)f(t_k, y_k)\} + O(h^3).$$

Subtract from Taylor expansion of $Y(t)$ about t_k to obtain global error :

$$e_{k+1} := Y_{k+1} - y_{k+1} = e_{k+1}^P + e_{k+1}^T,$$

where *propagated error*:

$$e_{k+1}^P = Y_k - y_k + (a + b)h\{f(t_k, Y_k) - f(t_k, y_k)\} + b\alpha h^2\{f_t(t_k, Y_k) - f_t(t_k, y_k)\} + b\beta h^2\{f_y(t_k, Y_k)f(t_k, Y_k) - f_y(t_k, y_k)f(t_k, y_k)\} + O(h^3)$$

& *local error*:

$$e_{k+1}^T = (1 - a - b)hf(t_k, Y_k) + \left(\frac{1}{2} - b\alpha\right)h^2f_t(t_k, Y_k) + \left(\frac{1}{2} - b\beta\right)h^2f_y(t_k, Y_k)f(t_k, Y_k) + O(h^3).$$

Propagated error vanishes if $y_k = Y_k$; usually non-zero due to errors at previous steps.

Local error contains Y_k but not y_k ; due to approximate nature of numerical method.

2nd-order 2-stage RK methods: generally $e_{k+1}^T = O(h)$, but $e_{k+1}^T = O(h^3)$ if:

$$a + b = 1, \quad b\alpha = \frac{1}{2}, \quad b\beta = \frac{1}{2}.$$

Modified Euler Method: $b = 1/2 \Rightarrow a = 1/2, \quad \alpha = 1 = \beta:$

$$\begin{aligned}k_1 &= hf(t_k, y_k) \\k_2 &= hf(t_{k+1}, y_k + k_1) \\y_{k+1} &= y_k + \frac{1}{2}(k_1 + k_2).\end{aligned}$$

Example 8.5

$$\frac{dy}{dt} = t y^{1/3}, \quad y(1) = 1.$$

Approximate $Y(1.1)$ using the modified Euler method.

Solution $t_0 = 1, y_0 = 1, h = 0.1, f(t, y) = t y^{1/3}.$

$$\begin{aligned}k_1 &= 0.1f(1, 1) = 0.1 \times 1 = 0.1 \\k_2 &= 0.1f(1.1, 1.1) = 0.113551 \\y_1 &= 1 + \frac{1}{2}(0.1 + 0.113551) = 1.10678.\end{aligned}$$

For 2nd-order 2-stage methods global error reduces to

$$\begin{aligned}e_{k+1} &= Y_k - y_k + h\{f(t_k, Y_k) - f(t_k, y_k)\} + \frac{1}{2}h^2\{f_t(t_k, Y_k) - f_t(t_k, y_k)\} \\ &\quad + \frac{1}{2}h^2\{f_y(t_k, Y_k)f(t_k, Y_k) - f_y(t_k, y_k)f(t_k, y_k)\} + O(h^3) \\ &= \left\{ \frac{\partial}{\partial y} \left(y + hf + \frac{1}{2}h^2(f_t + ff_y) \right)_{(t_k, \xi)} + O(h^3) \right\} e_k + e_{k+1}^T,\end{aligned}$$

where ξ lies between y_k & Y_k .

To $O(h^2)$ amplification factor is

$$G = \frac{\partial}{\partial y} \left\{ y + hf + \frac{1}{2}h^2(f_t + ff_y) \right\} \Big|_{(t_k, \xi)}.$$

Stability of two-stage methods requires $|G| < 1$.

Linear stability requires $|G| < 1$ for $f(t, y) = \lambda y$, i.e.

$$\left| 1 + h\lambda + \frac{1}{2}h^2\lambda^2 \right| < 1.$$

8.5.2 s -Stage Runge-Kutta Methods ($s > 2$)

Explicit s -stage Runge-Kutta method:

$$\begin{aligned}
 k_1 &= hf(t_k, y_k) \\
 k_2 &= hf(t_k + \alpha_2 h, y_k + \beta_{21} k_1) \\
 k_3 &= hf(t_k + \alpha_3 h, y_k + \beta_{31} k_1 + \beta_{32} k_2) \\
 &\vdots \\
 k_s &= hf(t_k + \alpha_s h, y_k + \beta_{s1} k_1 + \beta_{s2} k_2 + \dots + \beta_{s,s-1} k_{s-1}) \\
 y_{k+1} &= y_k + a_1 k_1 + a_2 k_2 + \dots + a_s k_s.
 \end{aligned}$$

α_2	β_{21}				
α_3	β_{31}	β_{32}			
\vdots					
α_s	β_{s1}	β_{s2}	\dots	$\beta_{s,s-1}$	
	a_1	a_2	\dots	a_{s-1}	a_s

Table 8.2: Tabular form of explicit Runge-Kutta methods.

Determine α_i , β_{ij} , a_j by agreement with Taylor expansion of solution to terms $O(h^p)$.
 Local error = $O(h^{p+1})$; order of method = p .

Highest order methods explicitly constructed of order 10 with 18 & 17 stages.

Classical Fourth-Order Four-Stage Method:

$$\begin{aligned}k_1 &= hf(t_k, y_k) \\k_2 &= hf\left(t_k + \frac{1}{2}h, y_k + \frac{1}{2}k_1\right) \\k_3 &= hf\left(t_k + \frac{1}{2}h, y_k + \frac{1}{2}k_2\right) \\k_4 &= hf(t_{k+1}, y_k + k_3) \\y_{k+1} &= y_k + \frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4).\end{aligned}$$

Example 8.6

$$\frac{dy}{dt} = ty^{1/3}, \quad y(1) = 1.$$

Approximate $Y(1.1)$ using the classical Runge-Kutta method.

Solution $t_0 = 1, y_0 = 1, h = 0.1, f(t, y) = ty^{1/3}.$

$$\begin{aligned}k_1 &= 0.1f(1, 1) = 0.1 \\k_2 &= 0.1f(1.05, 1.05) = 0.1067216175 \\k_3 &= 0.1f(1.05, 1.053360808) = 0.1068353600 \\k_4 &= 0.1f(1.1, 1.10683536) = 0.1137855274 \\y_1 &= 1 + \frac{1}{6}(0.1 + 2 \times 0.1067216175 + 2 \times 0.1068353600 + 0.1137855274) = 1.10681658.\end{aligned}$$

Runge-Kutta methods are easily applied to systems. E.g. the modified Euler method,

$$\begin{aligned}\mathbf{k}_1 &= h\mathbf{f}(t_k, \mathbf{y}_k) \\ \mathbf{k}_2 &= h\mathbf{f}(t_{k+1}, \mathbf{k}_k + \mathbf{k}_1) \\ \mathbf{k}_{k+1} &= \mathbf{y}_k + \frac{1}{2}(\mathbf{k}_1 + \mathbf{k}_2)\end{aligned}$$

where \mathbf{k}_1 & \mathbf{k}_2 are vectors. For system of two equations:

$$\begin{aligned}(k_1)_1 &= hf_1(t_k, (y_1)_k, (y_2)_k) \\ (k_1)_2 &= hf_2(t_k, (y_1)_k, (y_2)_k) \\ (k_2)_1 &= hf_1(t_{k+1}, (y_1)_k + (k_1)_1, (y_2)_k + (k_1)_2) \\ (k_2)_2 &= hf_2(t_{k+1}, (y_1)_k + (k_1)_1, (y_2)_k + (k_1)_2) \\ (y_1)_{k+1} &= (y_1)_k + \frac{1}{2}\{(k_1)_1 + (k_2)_1\} \\ (y_2)_{k+1} &= (y_2)_k + \frac{1}{2}\{(k_1)_2 + (k_2)_2\}.\end{aligned}$$

If method has order p for a single equation then it has order p for a system, if $p \leq 4$.

If $p > 4$ the order of method for systems may be $< p$.

8.5.3 Accuracy and Stability of Runge-Kutta Methods (A)

The global error of an explicit Runge-Kutta method of order p is $e_{k+1} = Ge_k + e_{k+1}^T$, where the amplification factor to $O(h^p)$ is

$$G = \frac{\partial}{\partial y} \left\{ y + hy^{(1)} + \frac{1}{2}h^2y^{(2)} + \dots + \frac{1}{p!}h^py^{(p)} \right\}_{(t_k, \xi)}.$$

The $y^{(j)}$ here must be expressed in terms of t and y using the differential equation. The conditions for stability and linear stability are, respectively, $|G| < 1$ and

$$\left| 1 + h\lambda + \frac{1}{2}h^2\lambda^2 + \dots + \frac{1}{p!}h^p\lambda^p \right| < 1.$$

For Runge-Kutta methods the region of linear stability increases with the order of the method.

The method of estimating the local truncation error from a higher derivative (y'' for the Euler method) relies on the form of the truncation error being known and reasonably simple. This is not true for Runge-Kutta methods. Instead, two other methods are commonly used for local error control, extrapolation and embedding. Extrapolation for a p -th order method uses the fact that the local error for a single step from t_k to t_{k+1} of size h is of the form

$$e_{k+1}^T = Ch^{p+1} + O(h^{p+2}),$$

where C is a constant. The local error for two steps of size $h/2$ from t_k to t_{k+1} is thus approximately $2C(h/2)^{p+1}$. If the corresponding estimates of $Y(t_{k+1})$ are $y_{k+1}(h)$ and $y_{k+1}(h/2)$, then, neglecting other errors,

$$Y(t_{k+1}) \approx y_{k+1}(h) + Ch^{p+1} \approx y_{k+1}(h/2) + 2C(h/2)^{p+1}.$$

This gives the following estimate for the local truncation error,

$$Ch^{p+1} \approx -\frac{y_{k+1}(h) - y_{k+1}(h/2)}{1 - 2^{-p}}.$$

If this is larger or much smaller than some specified error tolerance the stepsize h can be adjusted. To get a local error ϵ requires a new stepsize H such that $CH^{p+1} \approx \epsilon$. Eliminating C ,

$$\left(\frac{H}{h}\right)^{p+1} \approx \frac{(1 - 2^{-p})\epsilon}{y_{k+1}(h) - y_{k+1}(h/2)}$$

or

$$H \approx h \left\{ \frac{(1 - 2^{-p})\epsilon}{y_{k+1}(h) - y_{k+1}(h/2)} \right\}^{\frac{1}{p+1}}.$$

Various safeguards are needed, such as those in §8.4, to make a robust local error control. The major drawback of the method is the extra work required at each step.

Embedding methods use the stages of the method to calculate two approximations to the solution of different orders

$$\begin{aligned}y_{k+1} &= y_k + a_1k_1 + a_2k_2 + \dots + a_s k_s \\ \hat{y}_{k+1} &= y_k + \hat{a}_1k_1 + \hat{a}_2k_2 + \dots + \hat{a}_s k_s.\end{aligned}$$

$y_{k+1} - \hat{y}_{k+1}$ then provides an estimate of the local error. For example, England's fourth-order six-stage method

$$\begin{aligned}k_1 &= hf(t_k, y_k) \\ k_2 &= hf\left(t_k + \frac{1}{2}h, y_k + \frac{1}{2}k_1\right) \\ k_3 &= hf\left(t_k + \frac{1}{2}h, y_k + \frac{1}{4}k_1 + \frac{1}{4}k_2\right) \\ k_4 &= hf(t_{k+1}, y_k - k_2 + 2k_3) \\ k_5 &= hf\left(t_k + \frac{2}{3}h, y_k + \frac{1}{27}(7k_1 + 10k_2 + k_4)\right) \\ k_6 &= hf\left(t_k + \frac{1}{5}h, y_k + \frac{1}{625}(28k_1 - 125k_2 + 546k_3 + 54k_4 - 378k_5)\right) \\ y_{k+1} &= y_k + \frac{1}{6}(k_1 + 4k_3 + k_4)\end{aligned}$$

with the local error estimate

$$e_{k+1}^T = y_{k+1} - \hat{y}_{k+1} = \frac{1}{336}(-42k_1 - 224k_3 - 21k_4 + 162k_5 + 125k_6).$$

Another example is the Dormand-Prince 5(4) method — see Table 8.5.

8.6 Multistep Methods

General linear m -step method of fixed step-size ($|\alpha_m| + |\beta_m| \neq 0$):

$$y_{k+1} = \alpha_1 y_k + \alpha_2 y_{k-1} + \dots + \alpha_m y_{k-m+1} + h \{ \beta_0 f(t_{k+1}, y_{k+1}) + \beta_1 f(t_k, y_k) + \dots + \beta_m f(t_{k-m+1}, y_{k-m+1}) \}.$$

Explicit if $\beta_0 = 0$.

Implicit if $\beta_0 \neq 0$: unknown y_{k+1} occurs on both sides \Rightarrow solve non-linear equation for y_{k+1} at each step (if f non-linear in y).

Multistep methods for $m > 1$ more efficient.

Adams' Methods $\alpha_1 = 1, \quad \alpha_2 = \alpha_3 = \dots = \alpha_m = 0.$

Adams-Bashforth Methods Explicit $\beta_0 = 0$. $y'_i = f(t_i, y_i)$.

Polynomial $p_{m-1}^{AB}(t)$ of degree $m - 1$ interpolates m points (t_i, y'_i) ($i = k - m + 1, \dots, k$):

$$y(t_{k+1}) = y(t_k) + \int_{t_k}^{t_{k+1}} y' dt \approx y_{k+1} = y_k + \int_{t_k}^{t_{k+1}} p_{m-1}^{AB}(t) dt.$$

Adams-Moulton Methods Implicit $\beta_0 \neq 0$.

Polynomial $p_m^{AM}(t)$ of degree m interpolates $m + 1$ points (t_i, y'_i) ($i = k - m + 1, \dots, k + 1$):

$$y(t_{k+1}) = y(t_k) + \int_{t_k}^{t_{k+1}} y' dt \approx y_{k+1} = y_k + \int_{t_k}^{t_{k+1}} p_m^{AM}(t) dt.$$

p	β_1	β_2	β_3	β_4	β_5	β_6	C_p
1	1						$\frac{1}{2}$
2	$\frac{3}{2}$	$-\frac{1}{2}$					$\frac{5}{12}$
3	$\frac{23}{12}$	$-\frac{16}{12}$	$\frac{5}{12}$				$\frac{9}{24}$
4	$\frac{55}{24}$	$-\frac{59}{24}$	$\frac{37}{24}$	$-\frac{9}{24}$			$\frac{251}{720}$
5	$\frac{1901}{720}$	$-\frac{2744}{720}$	$\frac{2616}{720}$	$-\frac{1274}{720}$	$\frac{251}{720}$		$\frac{475}{1440}$
6	$\frac{4277}{1440}$	$-\frac{7923}{1440}$	$\frac{9982}{1440}$	$-\frac{7298}{1440}$	$\frac{2877}{1440}$	$-\frac{475}{1440}$	$\frac{19087}{60480}$

Table 8.3: Parameters of p th-order Adams-Bashforth method ($p = 1, \dots, 6$).

p	β_0	β_1	β_2	β_3	β_4	β_5	C_p
1	1	1					$-\frac{1}{2}$
2	$\frac{1}{2}$	$\frac{1}{2}$					$-\frac{1}{12}$
3	$\frac{5}{12}$	$\frac{8}{12}$	$-\frac{1}{12}$				$-\frac{1}{24}$
4	$\frac{9}{24}$	$\frac{19}{24}$	$-\frac{5}{24}$	$\frac{1}{24}$			$-\frac{19}{720}$
5	$\frac{251}{720}$	$\frac{646}{720}$	$-\frac{264}{720}$	$\frac{106}{720}$	$-\frac{19}{720}$		$-\frac{27}{1440}$
6	$\frac{475}{1440}$	$\frac{1427}{1440}$	$-\frac{798}{1440}$	$\frac{482}{1440}$	$-\frac{173}{1440}$	$\frac{27}{1440}$	$-\frac{863}{60480}$

Table 8.4: Parameters of p th order Adams-Moulton method ($p = 1, \dots, 6$).

p	β_0	α_1	α_2	α_3	α_4	α_5	α_6	C_p
1	1	1						$-\frac{1}{2}$
2	$\frac{2}{3}$	$\frac{4}{3}$	$-\frac{1}{3}$					$-\frac{1}{3}$
3	$\frac{6}{11}$	$\frac{18}{11}$	$-\frac{9}{11}$	$\frac{2}{11}$				$-\frac{1}{4}$
4	$\frac{12}{25}$	$\frac{48}{25}$	$-\frac{36}{25}$	$\frac{16}{25}$	$-\frac{3}{25}$			$-\frac{1}{5}$
5	$\frac{60}{137}$	$\frac{300}{137}$	$-\frac{300}{137}$	$\frac{200}{137}$	$-\frac{75}{137}$	$\frac{12}{137}$		$-\frac{1}{6}$
6	$\frac{60}{147}$	$\frac{360}{147}$	$-\frac{450}{147}$	$\frac{400}{147}$	$-\frac{225}{147}$	$\frac{72}{147}$	$-\frac{10}{147}$	$-\frac{1}{7}$

Table 8.5: Parameters of p th order Gear (closed) method ($p = 1, \dots, 6$).

k	y	y'	$y_{k+1} = \alpha_1 y_k + \beta_0 y'_{k+1} + \beta_1 y'_k + \beta_2 y'_{k-1} + \beta_3 y'_{k-2}$									
0	1	0	1	=	α_1							
1	t	1	1	=		β_0	+	β_1	+	β_2	+	β_3
2	t^2	$2t$	1	=		$2(\beta_0$			-	β_2	-	$2\beta_3)$
3	t^3	$3t^2$	1	=		$3(\beta_0$			+	β_2	+	$4\beta_3)$
4	t^4	$4t^3$	1	=		$4(\beta_0$			-	β_2	-	$8\beta_3)$

Table 8.6: Equations determining disposable parameters of 3-step Adams-Moulton method.

Example 8.7 2-step Adams-Moulton method.

Solution Quadratic polynomial interpolating (t_{k-1}, y'_{k-1}) , (t_k, y'_k) & (t_{k+1}, y'_{k+1}) :

$$\begin{aligned} p_2^{AM}(t) &= \frac{(t - t_k)(t - t_{k+1})}{(t_{k-1} - t_k)(t_{k-1} - t_{k+1})} y'_{k-1} + \frac{(t - t_{k-1})(t - t_{k+1})}{(t_k - t_{k-1})(t_k - t_{k+1})} y'_k + \frac{(t - t_{k-1})(t - t_k)}{(t_{k+1} - t_k)(t_{k+1} - t_{k-1})} y'_{k+1} \\ &= \frac{1}{2h^2} (t - t_k)(t - t_{k+1}) y'_{k-1} - \frac{1}{h^2} (t - t_{k-1})(t - t_{k+1}) y'_k + \frac{1}{2h^2} (t - t_{k-1})(t - t_k) y'_{k+1}. \end{aligned}$$

Integrating

$$\int_{t_k}^{t_{k+1}} p_2^{AM}(t) dt = I_1 y'_{k-1} - I_2 y'_k + I_3 y'_{k+1},$$

where

$$I_1 = \frac{1}{2h^2} \int_{t_k}^{t_{k+1}} (t - t_k)(t - t_{k+1}) dt = \frac{1}{2h^2} \int_0^h \tau(\tau - h) d\tau = -\frac{1}{6} h^3,$$

letting $t = t_k + \tau$. Similarly,

$$I_2 = -\frac{2}{3} h^3, \quad I_3 = \frac{5}{6} h^3.$$

$$\int_{t_k}^{t_{k+1}} p_2^{AM}(t) dt = -\frac{h}{12} y'_{k-1} + \frac{2h}{3} y'_k + \frac{5h}{12} y'_{k+1}$$

$$y_{k+1} = y_k + \int_{t_k}^{t_{k+1}} p_2^{AM}(t) dt = y_k + \frac{h}{12} \{5f(t_{k+1}, y_{k+1}) + 8f(t_k, y_k) - f(t_{k-1}, y_{k-1})\},$$

replacing y' by appropriate f .

Gear's Methods $\beta_0 \neq 0, \quad \beta_1 = \beta_2 = \dots = \beta_m = 0.$

Polynomial $p_m^G(t)$ of degree m interpolates $m + 1$ points (t_i, y_i) ($i = k - m + 1, \dots, k + 1$):

$$\frac{dp_m^G(t_{k+1})}{dt} = f(t_{k+1}, y_{k+1}).$$

Example 8.8 2-step Gear method.

Solution Quadratic polynomial interpolating (t_{k-1}, y_{k-1}) (t_k, y_k) & (t_{k+1}, y_{k+1}) :

$$p_2^G(t) = \frac{1}{2h^2}(t - t_k)(t - t_{k+1})y_{k-1} - \frac{1}{h^2}(t - t_{k-1})(t - t_{k+1})y_k + \frac{1}{2h^2}(t - t_{k-1})(t - t_k)y_{k+1}.$$

Differentiating

$$\begin{aligned} f(t_{k+1}, y_{k+1}) &= \left. \frac{dp_2^G}{dt} \right|_{t_{k+1}} \\ &= \frac{1}{2h^2}(t_{k+1} - t_k)y_{k-1} - \frac{1}{h^2}(t_{k+1} - t_{k-1})y_k + \frac{1}{2h^2}(2t_{k+1} - t_{k-1} - t_k)y_{k+1} \\ &= \frac{1}{2h}y_{k-1} - \frac{2}{h}y_k + \frac{3}{2h}y_{k+1}. \end{aligned}$$

Rearranging

$$y_{k+1} = \frac{4}{3}y_k - \frac{1}{3}y_{k-1} + \frac{2h}{3}f(t_{k+1}, y_{k+1}).$$

m -step Adams-Moulton method exact for problems with polynomial solutions of degree $m + 1$. Since the method is linear in f , it must be exact for $y = t^k$, $k = 0, 1, \dots, m + 1$, and this determines the disposable parameters of the method, $\alpha_1, \beta_0, \beta_1, \dots, \beta_m$. Similarly, the m -step Adams-Bashforth method, with disposable parameters $\alpha_1, \beta_1, \dots, \beta_m$, and Gear's method, with disposable parameters $\alpha_1, \dots, \alpha_m, \beta_0$, are exact for problems with polynomial solutions of degree m , in particular for $y = t^k$, $k = 0, 1, \dots, m$.

Example 8.9 3-step Adams-Moulton method. Determine $\alpha_1, \beta_0, \beta_1, \beta_2, \beta_3$ by requiring method to be exact for $y = 1, t, t^2, t^3, t^4$. The factor h ensures parameters do not depend on h or k . Hence take

$$h = 1, \quad t_k = 0 \quad t_{k+1} = 1, \quad t_{k-1} = -1, \quad t_{k-2} = -2.$$

Solution

$$\alpha_1 = 1, \quad \beta_0 = \frac{9}{24}, \quad \beta_1 = \frac{19}{24}, \quad \beta_2 = -\frac{5}{24}, \quad \beta_3 = \frac{1}{24}.$$

8.6.1 Accuracy and Stability of Multistep Methods

The m -step Adams-Moulton, Adams-Bashforth and Gear's methods are of orders $m+1$, m and m , and the *local errors* are

$$e^T = C_p h^{p+1} y^{(p+1)}(\xi_k),$$

where p is the order. These methods are convergent within their regions of stability.

The propagated error of multistep methods at t_{k+1} is

$$e_{k+1}^P = y_{k+1} - \bar{y}_{k+1} = \sum_{i=1}^m \alpha_i y_{k-i+1} + h \sum_{i=0}^m \beta_i y'_{k-i+1} - \sum_{i=1}^m \alpha_i Y_{k-i+1} - h \sum_{i=0}^m \beta_i Y'_{k-i+1}.$$

The global error is

$$e_{k+1} = e_{k+1}^P + e_{k+1}^T = \sum_{i=1}^m \alpha_i e_{k-i+1} + h \sum_{i=0}^m \beta_i f_y(t_{k-i+1}, \xi_{k-i+1}) e_{k-i+1} + e_{k+1}^T.$$

Setting $\alpha_0 = -1$ this can be written as

$$\sum_{i=0}^m \{\alpha_i + h\beta_i f_y(t_{k-i+1}, \xi_{k-i+1})\} e_{k-i+1} = -e_{k+1}^T.$$

For linear stability, $f(t, y) = \lambda y$, and the errors are related by

$$\sum_{i=0}^m \{\alpha_i + h\lambda\beta_i\} e_{k-i+1} = -e_{k+1}^T.$$

This is an inhomogeneous finite-difference equation with constant coefficients. Solutions of the homogeneous equation are of the form

$$e_k = C_1 \xi_1^k + C_2 \xi_2^k + \dots + C_m \xi_m^k$$

if the associated characteristic equation

$$\rho(\xi) + h\lambda\sigma(\xi) = 0$$

has m distinct roots $\xi_1, \xi_2, \dots, \xi_m$, where

$$\rho(\xi) = \alpha_0\xi^m + \alpha_1\xi^{m-1} + \dots + \alpha_m, \quad \sigma(\xi) = \beta_0\xi^m + \beta_1\xi^{m-1} + \dots + \beta_m.$$

Secular terms must be added if there are repeated roots. The error will not grow and the method is *absolutely stable* if the roots of the characteristic polynomial satisfy $|\xi_i| \leq 1$, $i = 1, 2, \dots$ and there is no repeated root on the circle $|\xi| = 1$. There is a region of the complex $h\lambda$ plane for which the roots of the characteristic equation satisfy the absolute stability condition $|\xi| \leq 1$.

For $h\lambda = 0$ the characteristic equation reduces to $\rho(\xi) = 0$. For small $h\lambda$ the roots of $\rho(\xi)$ provide a first approximation to the roots of $\rho(\xi) + h\lambda\sigma(\xi)$. It can be shown (Gear 1971) that a necessary and sufficient condition for a method of order $p \geq 1$ to be stable is that the roots of $\rho(\xi)$ satisfy $|\xi| \leq 1$ and that the roots on $|\xi| = 1$ be simple. A stable method is only *weakly stable* if there is more than one distinct root of $\rho(\xi)$ on $|\xi| = 1$ and it is *strongly stable* if there is only one root of $\rho(\xi)$ on $|\xi| = 1$. The Adams and Gear methods are strongly stable. With weakly stable methods such as Milne's method there is the possibility of parasitic instability: for $h\lambda$ small one of the roots of $\rho(\xi)$ on $|\xi| = 1$ behaves like the true solution, which is $y = Ce^{\lambda t}$, while the other root(s) on $|\xi| = 1$ contribute terms which may swamp the true solution.

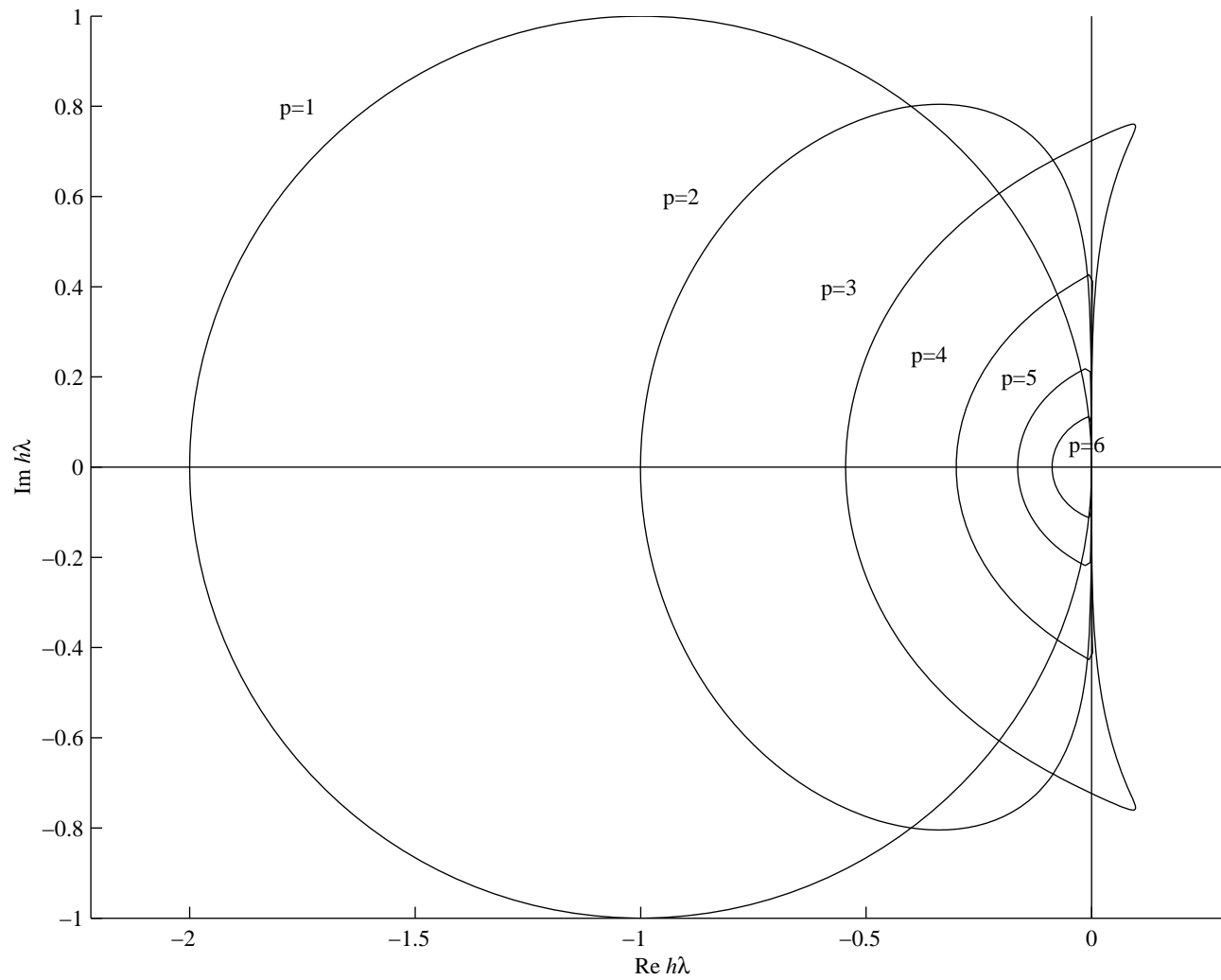


Figure 8.3: The p th order Adams-Bashforth method is stable within the contour labelled $p = 1, 2, 3, 4, 5, 6$.

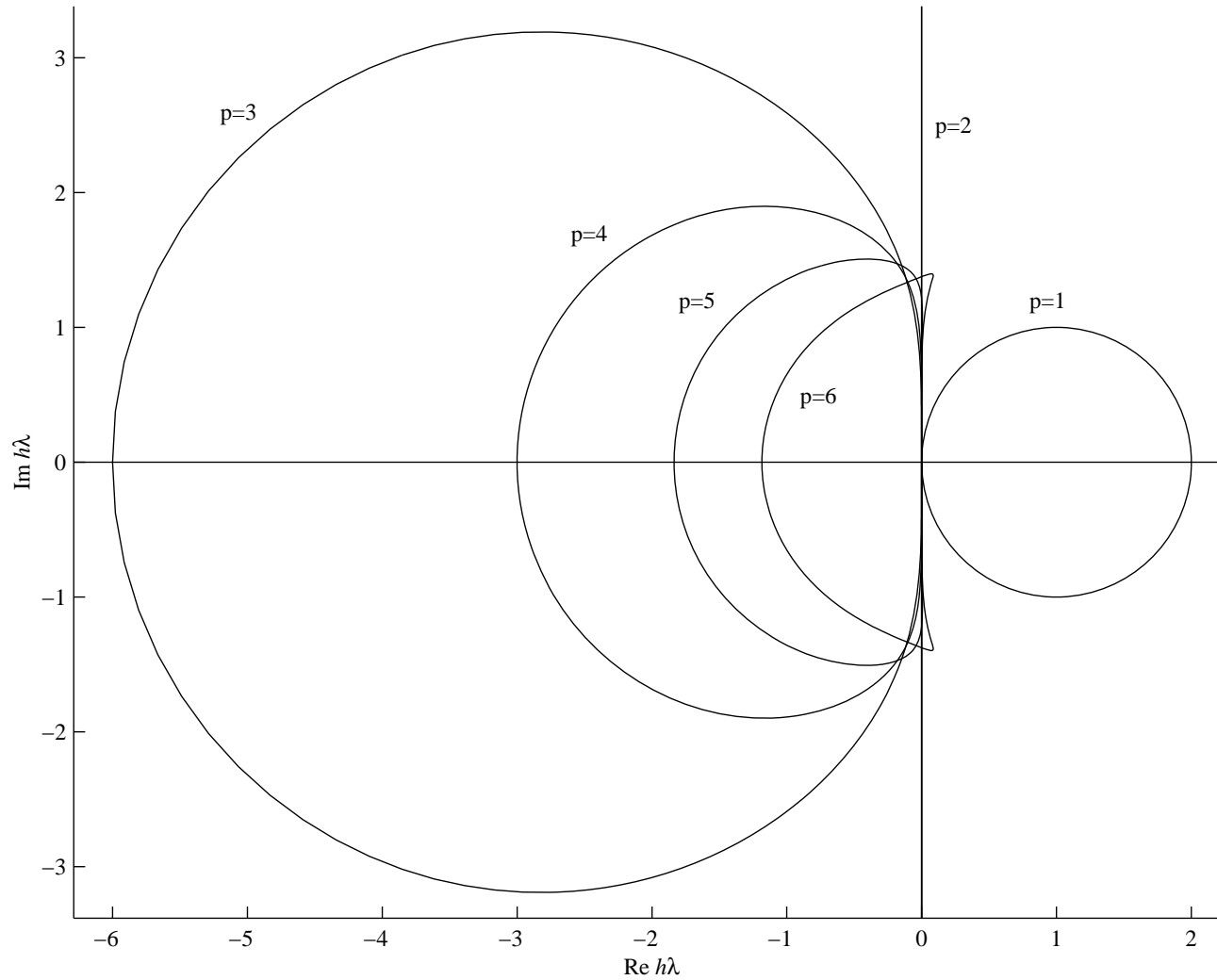


Figure 8.4: The p th order Adams-Moulton method is stable outside the contour labelled $p = 1$, to the left of the contour labelled $p = 2$ and within the contours labelled $p = 3, 4, 5, 6$.

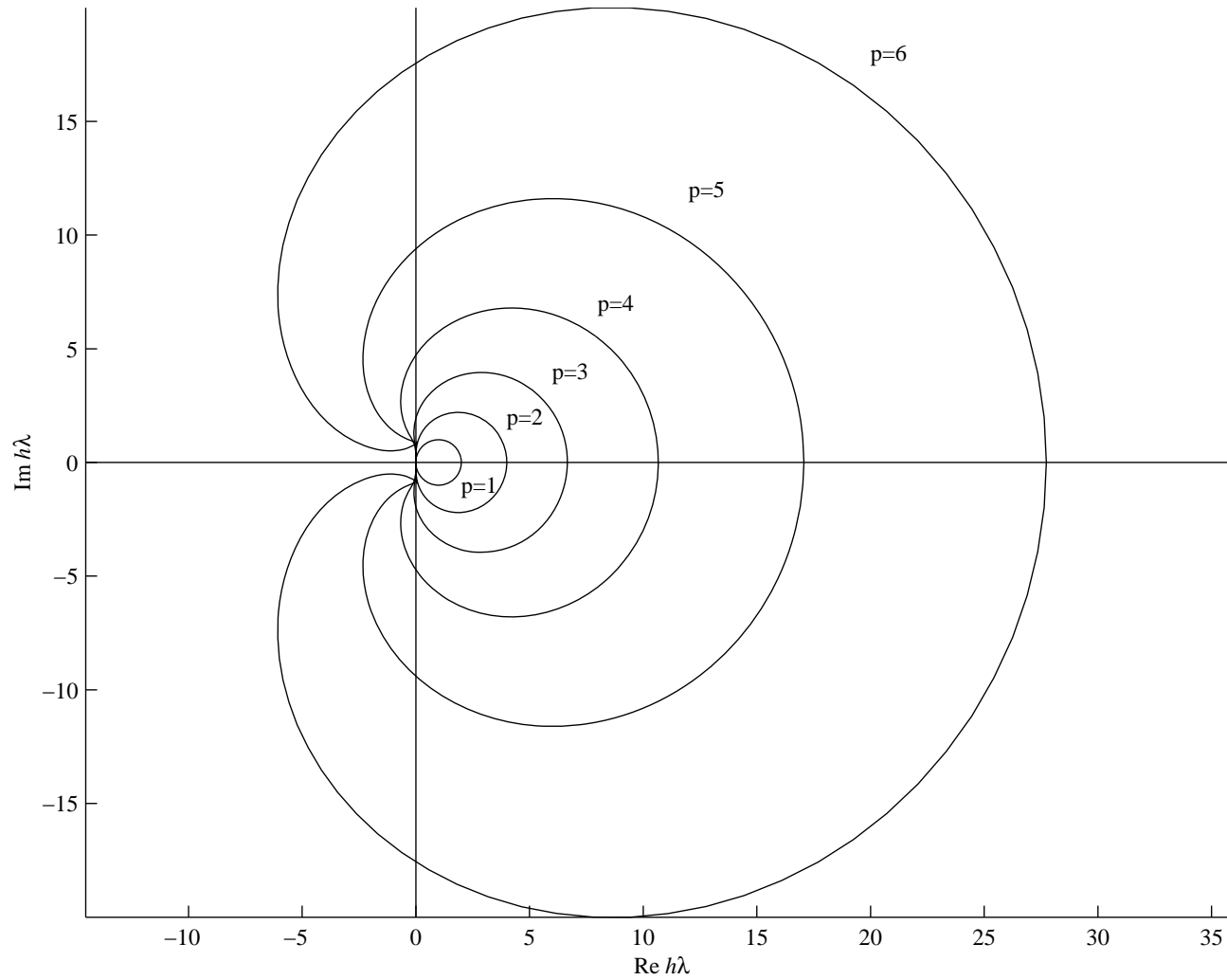


Figure 8.5: The p th order Gear method is stable outside the contours labelled $p = 1, 2, 3, 4, 5, 6$. Note the stability regions are infinite in the left half-plane.

8.6.2 Solution of Implicit Methods

Implicit methods are more stable than explicit methods.

Implicit methods require solution of non-linear equations at every step ($\beta_0 \equiv \beta$):

$$y_{k+1} = y_k + \beta h f(t_{k+1}, y_{k+1}) + g(y_k, y_{k-1}, \dots, y_{k-m+1})$$

$$0 = F(y) := y - y_k - \beta h f(t_{k+1}, y) - g.$$

Functional Iteration

$$y_{k+1}^{(j+1)} = y_k + \beta h f(t_{k+1}, y_{k+1}^{(j)}) + g.$$

Initial estimate $y_{k+1}^{(0)}$ of y_{k+1} predicted (P) by explicit method; e.g. Adams-Bashforth method of order m or $m + 1$ predictor for Adams-Moulton method of order $m + 1$. Function f is evaluated (E) & correction (C) $y_{k+1}^{(1)}$ is found.

For better stability another E is done giving a PECE method.

1 or 2 iterations give y_{k+1} to an accuracy of same order as local error of implicit method.

Adams methods generally used for these predictor-corrector methods.

Functional iteration methods only converge if $|\beta h J| < 1$:

$$\begin{aligned}y_{k+1}^{(j+1)} - y_{k+1}^{(j)} &= \{y_k + \beta h f(t_{k+1}, y_{k+1}^{(j)}) + g\} - \{y_k + \beta h f(t_{k+1}, y_{k+1}^{(j-1)}) + g\} \\ &= \beta h \{f(t_{k+1}, y_{k+1}^{(j)}) - f(t_{k+1}, y_{k+1}^{(j-1)})\} \\ &= \beta h J(\xi_{k+1}^{(j)}) (y_{k+1}^{(j)} - y_{k+1}^{(j-1)})\end{aligned}$$

For stiff problems, where $|J| \gg 1$, this forces such a small step-size, $h < (\beta|J|)^{-1} \ll 1$, that predictor-corrector methods are practically useless for stiff problems.

Newton's Method

$$y_{k+1}^{(j+1)} = y_{k+1}^{(j)} - \{F'(y_{k+1}^{(j)})\}^{-1} F(y_{k+1}^{(j)}), \quad F' = 1 - \beta h J.$$

Newton's method suitable for stiff problems, since it converges regardless of h if $y_{k+1}^{(0)}$ is sufficiently accurate.

Non-stiff: use Adams predictor-corrector methods: more accurate than Gear's methods for same number of function evaluations.

Stiff: use Gear's methods; Adams methods unstable without very small step-size.