Chapter 1

Numerical solution of ordinary differential equations - initial value problem

Numerical integration of ordinary differential equations is a frequent task of numerical analysis in chemical engineering problems. Numerical integration of differential equations is used if the equations are nonlinear or if we have a large system of linear equations with constant coefficients, where the analytical solution can be found, but it is in the form of long and complicated expressions containing exponential functions. Numerical integration of such systems is more efficient both in human time and in computer time. Numerical integration of linear equations with non-constant coefficients is also more efficient than the analytical solution; in the case of inner diffusion in porous catalyst with a chemical reaction of the 1. order the analytical solution contains Bessel functions, which can be evaluated more conveniently when we use numerical integration of the original equations than to evaluate Bessel functions.

Many problems in chemical engineering involve solution of ordinary differential equations. These are dynamical problems in isotropic media and stationary problems with a single space variable. The former include batch reactor, differential distillation, non-stationary regime of a distillation column etc. The latter include tube reactors and heat exchangers.

In some chemical engineering problems dynamic balance must be solved with accumulation that terms differ by several orders of magnitude. This corresponds to physical processes where some dependent variables relax very fast while others approach the stationary state slowly. This type of problems is called “stiff” and it is difficult to solve. Stiff problems often arise in reactor engineering (radical reactions, complex reactions with one of them very fast) and in system engineering (dynamic regime of a distillation column with a mixture containing one volatile component or one component with trace concentration).

Problems in dynamics of counter-current separation devices or systems of interacting devices lead to systems of hundreds of ordinary differential equations.

Solution of such problems often requires special algorithms.

We do not discuss differential-algebraic equations (DAE) that can be express in the form

$$\boldsymbol{F}(\boldsymbol{y}', \boldsymbol{y}) = \boldsymbol{0},$$

that cannot be solved in $\boldsymbol{y}'$. These equations appear in several chemical engineering problems and they are difficult to solve. The reader is invited to check the specialized literature [?], [?], [?].

1.1 Euler's method and the method of Taylor's expansion

Consider a single differential equation

$$y' = f(x, y) \tag{1.1}$$

with the initial condition

$$y(a) = c. \tag{1.2}$$

We want to find the solution $y(x)$ in discrete points (nodes) $a = x_0 < x_1 < x_2 < \ldots < x_N = b$ i.e. we want to find numbers $y_0 = c$, $y_1, \ldots, y_N$, approximating the values $y(x_0), y(x_1), \ldots, y(x_N)$ of the exact solution in the nodes $x_0, \ldots, x_N$. We often consider the equidistant grid, i.e. $x_{n+1} - x_n = h$; $n = 0, 1, \ldots, N-1$. The number h is called the step size. The approximation y_n of the exact solution $y(x_n)$ in x_n is computed from the values of the approximate solution evaluated in previous nodes. If y_{n+1} is expressed by k values $y_n, y_{n-1}, \ldots, y_{n+1-k}$, the method is called a k-step method. If we replace the derivative y' in $x = x_n$ by the difference formula using two points x_n and x_{n+1} (see formula 1 in table **??**) we get the Euler's method

$$y_{n+1} = y_n + hf(x_n, y_n), \qquad n = 0, 1, 2, \ldots, N-1, \tag{1.3}$$

with

$$y_0 = c\,.$$

The computation using the Euler's method is very easy. We can illustrate it by the following example. Solve the equation $y' = y$; $y(0) = 1$ using the Euler's method. The recurrent relation (1.3) is

$$y_{n+1} = (1+h)y_n\,, \qquad y_0 = 1\,,$$

i.e.

$$y_n = (1+h)^n\,.$$

For a given x we have $n = \frac{x}{h}$, and thus

$$y_n = (1+h)^{\frac{x}{h}} = [(1+h)^{\frac{1}{h}}]^x.$$

For $h \to 0_+$ the approximate solution y_n converges to the exact solution e^x.

Denoting $y(x)$ the exact solution, the difference

$$e_n = y_n - y(x_n) \tag{1.4}$$

is called the global approximation error or the global discretization error and y_n is called the theoretical approximation of the solution. Another type of error comes from the fact that we cannot compute the value y_n exactly. Denoting $\tilde{y}_n$ the values that are computed instead of y_n, the difference

$$r_n = \tilde{y}_n - y_n \tag{1.5}$$

is called the round-off error. Then the total error is given by the triangle inequality

$$|\tilde{y}_n - y(x_n)| \leq |e_n| + |r_n| \,. \tag{1.6}$$

The values $\tilde{y}_n$ are called the numerical approximation. In what follows we deal with the theoretical approximation only, though the round-off error is also important, because they may be larger than the approximation error in some cases. We also skip the derivation of the error estimates because it is out of the scope of this text.

If the function $f(x, y)$ satisfies the Lipschitz condition in y, i.e. if there is a constant $L > 0$ such that

$$|f(x,y) - f(x,y^*)| \leq L|y - y^*| \tag{1.7}$$

is true for $x \in [a, b]$ and any y and y^* and if the exact solution $y(x)$ of the equation (1.1) is twice differentiable in the interval $[a, b]$, and denoting

$$N(x) = \frac{1}{2} \max_{t\in[a,x]} |y''(t)| \,, \tag{1.8}$$

then the global approximation error of the Euler's method can be estimated by

$$|e_n| \leq hN(x_n)E_L(x_n - a) \,. \tag{1.9}$$

Here

$$E_L(x) = \begin{cases} \dfrac{e^{Lx} - 1}{L} & \text{if } L > 0 \\ x & \text{if } L = 0 \end{cases} \tag{1.10}$$

is the so called Lipschitz function.

Assuming the function f has the first partial derivative in $\Omega = [a, b] \times (-\infty, \infty)$ continuous then we can estimate $N(x)$ by

$$2N(x) \leq N = \max_{(x,y)\in\Omega} |f_x(x,y) + f_y(x,y)f(x,y)| \,, \tag{1.11}$$

where the index x and y denotes the partial derivative with respect to x and y resp.

The estimates (1.9) are usually very pessimistic, which can be illustrated by the following example:

$$y' = y \,, \qquad y(0) = 1 \,.$$

The exact solution is $y(x) = e^x$. Equation (1.7) gives $L = 1$. The estimate $N(x)$ can be done from the exact solution, i.e.

$$2N(x) = e^x \,.$$

Table 1.1: Global approximation error e_n and its theoretical estimate (1.12), $h = 2^{-6}$

x_n	1	2	3	4	5
y_n	2.69735	7.27567	19.62499	52.93537	142.7850
e_n	-0.02093	-0.11339	-0.46055	-1.66278	-5.6282
estimate (1.12)	0.03649	0.36882	2.99487	22.86218	170.9223

According to (1.9) we have

$$|e_n| \le \frac{1}{2} h \mathrm{e}^{x_n} (\mathrm{e}^{x_n} - 1) \; . \qquad (1.12)$$

Table 1.1 compares this theoretical estimate with the real global approximation error for $h = 2^{-6}$.

The estimate (1.9) shows that the error of the Euler's method for a given x is proportional to the first power of the step size h, i.e. $\mathcal{O}(h)$ (see **??**). We say the Euler's method is of the first order. Thus the Richardson extrapolation can be used for an a posteriori error estimate (see (**??**)).

Fig. 1.1 illustrates the round-off error. The global approximation error is proportional to h while the round-off error is proportional to $1/h$ (the smaller the h the greater the number of arithmetic operations). As a result there is a certain "optimal" step size $h_{\rm opt}$ giving the least total error.

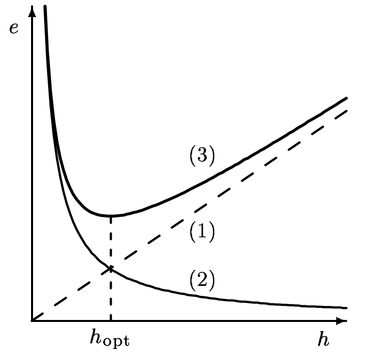

Figure 1.1: Global approximation error e_n (1), round-off error (2) and the total error (3) for the Euler's method.

We do not want to use $h_{\rm opt}$ as the step size, because then the round-off error is of the same size as the approximation error and the Richardson's extrapolation cannot be used for the estimate of the total approximation error. The only way how to estimate the round-off error is to repeat the computation with different precision (different size of the floating numbers used by the computer).

Modern algorithms adjust the step size h automatically according to the local approximation error to get the final approximation with the required accuracy with a small number of operations (see 1.2, **??**).

For special cases the method of Taylor's expansion can be used. If the function f in (1.1) has enough derivatives then we can write

$$\begin{aligned} y'' &= \frac{df}{dx}(x, y(x)) = f_x(x,y) + f_y(x,y)y' = \\ &= f_x(x,y) + f_y(x,y)f(x,y) \; , \end{aligned} \qquad (1.13)$$

where the index x or y denotes the partial derivative with respect to x or y resp. The third derivative is

$$y''' = f_{xx} + 2ff_{xy} + f_{yy}f^2 + f_xf_y + ff_y^2 \,, \tag{1.14}$$

etc. The change in $y(x)$ can be found by the Taylor's expansion

$$\begin{aligned} y(x_n+h) &\doteq y_{n+1} = \qquad\qquad (1.15)\\ &= y_n + hy'(x_n) + \frac{h^2}{2}y''(x_n) + \ldots + \frac{h^p}{p!}y^{(p)}(x_n) + \mathcal{O}(h^{p+1}) \,. \end{aligned}$$

The method (1.15) is called the method of Taylor's expansion of order p. Its global error is of order p, i.e. $\mathcal{O}(h^p)$.

Example 1.1.1 *Use the method of Taylor's expansion of the third order to solve the initial value problem*

$$y' = \frac{4}{x^2} - y^2 - \frac{y}{x} \,, \qquad y(1) = 0 \,. \tag{1.16}$$

Solution:
According to (1.15) for $n = 0, 1, 2, \ldots$ *we have*

$$y(x_n+h) \doteq y_{n+1} = y_n + hy'(x_n) + \frac{h^2}{2}y''(x_n) + \frac{h^3}{3!}y'''(x_n) + \mathcal{O}(h^4) \,.$$

Here

$$\begin{aligned} x_0 = 1 \,, &\qquad y_0 = 0 \,, \\ y'(x_n) &= \frac{4}{x_n^2} - y_n^2 - \frac{y_n}{x_n} \,, \\ y''(x_n) &= -\frac{8}{x_n^3} - 2y_ny'(x_n) - \frac{y'(x_n)x_n - y_n}{x_n^2} = -\frac{12}{x_n^3} - \frac{6y_n}{x_n^2} + \frac{3y_n^2}{x_n} + 2y_n^3 \,, \\ y'''(x_n) &= \frac{24}{x_n^4} - 2(y'(x_n))^2 - 2y_ny''(x_n) - \frac{y''(x_n)x_n^2 - 2(y'(x_n)x_n - y_n)}{x_n^3} = \\ &= \frac{12}{x_n^4} - \frac{42y_n}{x_n^3} + \frac{21y_n^2}{x_n^2} - \frac{12y_n^3}{x_n} - 6y_n^4 \,. \end{aligned}$$

Table 1.2 shows the computed values of the solution in the point $x_N = 2$ *for various* N *(and thus for various* $h = 1/N$*).*

It is obvious that this method is not suitable for general equation, because analytical differentiation may be very laborious for higher orders. This method can be used even for systems of differential equations, but the complexity of the derivation increases. Richardson's extrapolation can be used as well as illustrated in Table 1.2.

1.2 Runge-Kutta methods

The analytical differentiation needed for the Taylor's expansion as shown in the previous section is a principal obstacle for most practical problems. We

Table 1.2: Solution of 1.16 using Taylor's expansion of order 3.

	x	1	1.2	1.4	1.6	1.8	2
$h = 0.2$	$y(x)$	0	0.576000	0.835950	0.920226	0.920287	0.884745
$h = 0.1$		0	0.581645	0.838338	0.919251	0.918141	0.882631
$h = 0.05$		0	0.582110	0.838443	0.919062	0.917872	0.882386
Richardson's extrapolation (see ??) in $x = 2$: $p = 3\,,\quad h_1 = 0.1\,,\quad h_2 = 0.05\,,$ $y_1(2) = 0.882631\,,\quad y_2(2) = 0.882386 \quad \Rightarrow \quad y_{12}(2) = 0.882351$							
Exact solution: $\quad y(x) = \dfrac{2(x^4-1)}{x(x^4+1)}\,, \qquad y(2) = 0.882353$							

show a method with similar properties (order of approximation) as the Taylor's expansion method, but without the need of analytical differentiation. Let us write the increment in the form

$$y_{n+1} = y_n + h\Phi(x_n, y_n; h) \tag{1.17}$$

where $y_n \sim y(x_n)$. For the Euler's method we had $\Phi(x,y;h) = f(x,y)$. Assume the increment function Φ in the form

$$\Phi(x,y;h) = a_1 f(x,y) + a_2 f\Big(x + p_1 h, y + p_2 h f(x,y)\Big) \tag{1.18}$$

where the constants a_1, a_2, p_1 and p_2 are to be found so that the method approximates the solution as good as possible. Put Φ from (1.18) into (1.17) and expand in powers of h (with $x = x_n$, $y = y_n$) :

$$y_{n+1} = y_n + h\Big\{(a_1+a_2)f(x,y) + ha_2\Big(p_1 f_x(x,y) + p_2 f_y(x,y) f(x,y)\Big) + \mathcal{O}(h^2)\Big\}\,. \tag{1.19}$$

We want the expansion (1.19) to agree with the Taylor's expansion

$$y(x_n+h) = y(x_n) + hf(x,y) + \frac{1}{2}h^2\Big(f_x(x,y) + f_y(x,y)f(x,y)\Big) + \mathcal{O}(h^3) \tag{1.20}$$

where y' was replaced by f and y'' was replaced by (1.13). Comparing the terms linear in h in (1.19) and (1.20) we get

$$a_1 + a_2 = 1. \tag{1.21}$$

The agreement of the terms quadratic in h (for any $f(x,y)$) requires

$$a_1 p_1 = \frac{1}{2} \quad , \quad a_2 p_2 = \frac{1}{2}\,. \tag{1.22}$$

It can be shown that the agreement of cubic terms in h cannot be achieved for general $f(x,y)$. We have three equations (1.21), (1.22) for four unknown parameters a_1, a_2, p_1, p_2. We can choose one of them, say $a_2 = \alpha$, then

$$a_1 = 1 - \alpha, \quad a_2 = \alpha, \quad p_1 = p_2 = \frac{1}{2\alpha} \tag{1.23}$$

where $\alpha \neq 0$ is a free parameter. Then the equation (1.17) using (1.18) has the form

$$y_{n+1} = y_n + (1-\alpha)hf(x_n, y_n) + \alpha hf\Big(x_n + \frac{h}{2\alpha}, y_n + \frac{h}{2\alpha}f(x_n, y_n)\Big) + \mathcal{O}(h^3) \ . \tag{1.24}$$

The result (1.24) can be conveniently written in successive equations

$$\begin{aligned} k_1 &= hf(x_n, y_n) \\ k_2 &= hf(x_n + \tfrac{h}{2\alpha}, y_n + \tfrac{1}{2\alpha}k_1) \\ y_{n+1} &= y_n + (1-\alpha)k_1 + \alpha k_2 \ . \end{aligned}$$

The cases $\alpha = \frac{1}{2}$ and $\alpha = 1$ are well known and they are called improved Euler's method or Heun's method :

$$\begin{aligned} k_1 &= hf(x_n, y_n) \\ k_2 &= hf(x_n + h, y_n + k_1) \\ y_{n+1} &= y_n + \tfrac{1}{2}(k_1 + k_2) \end{aligned} \tag{1.25}$$

and modified Euler's method

$$\begin{aligned} k_1 &= hf(x_n, y_n) \\ k_2 &= hf(x_n + \tfrac{h}{2}, y_n + \tfrac{1}{2}k_1) \\ y_{n+1} &= y_n + k_2 \ . \end{aligned} \tag{1.26}$$

In some texts (1.25) is called modified Euler's method. Both of these methods have the local error $\mathcal{O}(h^3)$, and the global error $\mathcal{O}(h^2)$. They belong to the family of Runge-Kutta methods as the simplest examples of them. More complicated and more accurate methods can be derived by a similar approach. We mention some representatives of them of order 3, 4, and 5. A general Runge-Kutta method can be written in successive equations (with $x = x_n$, $y = y_n$):

$$\begin{aligned} k_1 &= hf(x, y) \\ k_2 &= hf(x + \alpha_1 h, y + \beta_{11}k_1) \\ k_3 &= hf(x + \alpha_2 h, y + \beta_{21}k_1 + \beta_{22}k_2) \\ &\vdots \\ k_{j+1} &= hf(x + \alpha_j h, y + \beta_{j1}k_1 + \beta_{j2}k_2 + \cdots + \beta_{jj}k_j) \\ y_{n+1} &= y_n + \gamma_1 k_1 + \gamma_2 k_2 + \cdots + \gamma_{j+1}k_{j+1} \ . \end{aligned} \tag{1.27}$$

The method (1.27) can be written in the form of Table 1.3. This table also lists some Runge-Kutta methods and their order (global error).

If we want to get the order m with the Runge-Kutta method then for $m = 2, 3, 4$ we need $2, 3, 4$ evaluations of the right hand side of the differential equation. For $m = 5$ we need at least 6 evaluations and for $m > 4$ we need more than m evaluations. Thus the methods of order greater than 4 are seldom used, because their advantages become important only when very high accuracy is needed.

Sometimes the solution has a different character for different values of the independent variable x, and a different step size h should be used to get the desired accuracy. If we choose the step size to be the minimum of all the required step sizes, the accuracy is achieved, but in some parts we integrate

Table 1.3: Overview of Runge-Kutta methods

Scheme of Runge-Kutta methods

α_1	β_{11}				
α_2	β_{21}	β_{22}			
α_3	β_{31}	β_{32}	β_{33}		
$\vdots$					
α_j	β_{j1}	β_{j2}	$\ldots$	β_{jj}	
	γ_1	γ_2	$\ldots$	γ_j	γ_{j+1}

Euler

improved (1.24)		$\mathcal{O}(h^2)$
1	1	
	$\frac{1}{2}$	$\frac{1}{2}$

modified (1.25)		$\mathcal{O}(h^2)$
$\frac{1}{2}$	$\frac{1}{2}$	
	0	1

Heun			$\mathcal{O}(h^3)$
$\frac{1}{3}$	$\frac{1}{3}$		
$\frac{2}{3}$	0	$\frac{2}{3}$	
	$\frac{1}{4}$	0	$\frac{3}{4}$

Kutta			$\mathcal{O}(h^3)$
$\frac{1}{2}$	$\frac{1}{2}$		
1	-1	2	
	$\frac{1}{6}$	$\frac{2}{3}$	$\frac{1}{6}$

Runge-Kutta order 4

standard				$\mathcal{O}(h^4)$
$\frac{1}{2}$	$\frac{1}{2}$			
$\frac{1}{2}$	0	$\frac{1}{2}$		
1	0	0	1	
	$\frac{1}{6}$	$\frac{1}{3}$	$\frac{1}{3}$	$\frac{1}{6}$

three eighth				$\mathcal{O}(h^4)$
$\frac{1}{3}$	$\frac{1}{3}$			
$\frac{2}{3}$	$-\frac{1}{3}$	1		
1	1	-1	1	
	$\frac{1}{8}$	$\frac{3}{8}$	$\frac{3}{8}$	$\frac{1}{8}$

Butcher order 5				$\mathcal{O}(h^5)$		
$\frac{1}{4}$	$\frac{1}{4}$					
$\frac{1}{4}$	$\frac{1}{8}$	$\frac{1}{8}$				
$\frac{1}{2}$	0	$-\frac{1}{2}$	1			
$\frac{3}{4}$	$\frac{3}{16}$	0	0	$\frac{9}{16}$		
1	$-\frac{3}{7}$	$\frac{2}{7}$	$\frac{12}{7}$	$-\frac{12}{7}$	$\frac{8}{7}$	
	$\frac{7}{90}$	0	$\frac{32}{90}$	$\frac{12}{90}$	$\frac{32}{90}$	$\frac{7}{90}$

unnecessarily accurate. This is not an effective approach. Single step methods (as Runge-Kutta e.g.) allow adaptive adjustment of the integration step size according to the character of the solution. A whole class of methods have been developed where the error in each step is estimated from the computed k_i, where the number of these k_i must be more than the minimal number of them. The first method of this kind was developed by Merson, others were found by Fehlberg. The Merson's method is of order 4 and it uses 5 evaluations of the right hand side $f(x, y)$. It can be written as follows:

$$\begin{aligned}
k_1 &= hf(x_0, y_0) & y_1 &= y_0 + \frac{k_1}{3} \\
k_2 &= hf(x_0 + \frac{h}{3}, y_1) & y_2 &= y_0 + \frac{k_1 + k_2}{6} \\
k_3 &= hf(x_0 + \frac{h}{3}, y_2) & y_3 &= y_0 + 0.125k_1 + 0.375k_3 \\
k_4 &= hf(x_0 + 0.5h, y_3) & y_4 &= y_0 + 0.5k_1 - 1.5k_3 + 2k_4 \\
k_5 &= hf(x_0 + h, y_4) & y_5 &= y_0 + \frac{k_1 + 4k_4 + k_5}{6} \ .
\end{aligned} \tag{1.28}$$

For small h assuming $f(x, y)$ approximated by

$$f(x, y) = Ax + By + C \tag{1.29}$$

Merson derived that the error or y_4 is $\frac{-h^5 y^{(5)}}{120}$ and the error of y_5 is $\frac{-h^5 y^{(5)}}{720}$. Then we can estimate the error of y_5 by

$$E = \frac{1}{5}(y_4 - y_5) \ . \tag{1.30}$$

If this estimate E is less than the desired error ε then the current step size is good. If not, we decrease the step size (by taking one half of it) and we recompute the last step. If $|E| < \frac{\varepsilon}{32}$ we can increase the step size (by taking its double). Instead of taking one half or the double of the step size, we can predict the optimal step size by

$$h_{new} = 0.8\, h_{old} \left(\frac{\varepsilon}{|E|}\right)^{0.2} . \tag{1.31}$$

The factor 0.8 is used to avoid the case when after prolongation we have to shorten the step size.

Each Runge-Kutta method can be used not just for a single differential equation but also for a system of differential equations of the first order. Then y, f, k_i become vectors. They can be used for equations of a higher order as well. Such a system can be converted into a system of the first order as illustrated by the following example. The equation

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

is equivalent to the system

$$y' = z \qquad z' = f(x, y, z) \ .$$

There are special Runge-Kutta methods for equations of the 2. order. Their advantages are weak so they are seldom used.

1.3 Multi step methods

When using single-step methods as described in the previous section, we do not make use of the course of the solution found so far. After each step we forget all the information and we start from scratch. This is not effective. Multi step methods have been designed to utilize a few last points of the solution.

The solution is computed in an equidistant grid of points with the step size h. We denote $x_i = x_0 + ih\,, y_i \approx y(x_i)\,, f_i = f(x_i, y_i)\,.$ A general linear multi-step method can be written as

$$\alpha_k y_{n+k} + \alpha_{k-1} y_{n+k-1} + \cdots + \alpha_0 y_n = h\Big(\beta_k f_{n+k} + \beta_{k-1} f_{n+k-1} + \cdots + \beta_0 f_n\Big) \tag{1.32}$$

assuming $\alpha_k \neq 0\,,\ \alpha_0^2 + \beta_0^2 > 0\,.$ This is called a k-step method. Let us denote the polynomial

$$\varrho(\xi) = \alpha_k \xi^k + \cdots + \alpha_1 \xi + \alpha_0\,. \tag{1.33}$$

A necessary condition for the convergence (i.e. for $h \to 0_+$ we approach the exact solution) of the linear multi-step method (1.32) is: all the roots of the polynomial $\varrho(\xi)$ must be in absolute value less than 1, or equal to 1 but then they must be of multiplicity 1. This is called the stability condition of the method. Methods that fail this condition are useless.

Let us define the adjoint differential operator

$$\begin{aligned} L[y(x);h] &= \alpha_k y(x+kh) + \alpha_{k-1} y(x+(k-1)h) + \cdots + \alpha_0 y(x) - \\ &\quad -h\Big(\beta_k y'(x+kh) + \beta_{k-1} y'(x+(k-1)h) + \cdots + \beta_0 y'(x)\Big)\,. \end{aligned} \tag{1.34}$$

Expanding $y(x+mh)$ and $y'(x+mh)$ by the Taylor's polynomial around x we get

$$\begin{aligned} y(x+mh) &= y(x) + mhy'(x) + \frac{1}{2}m^2h^2y''(x) + \cdots + \frac{1}{i!}m^ih^iy^{(i)}(x) + \cdots \\ hy'(x+mh) &= hy'(x) + mh^2y''(x) + \frac{1}{2}m^2h^3y'''(x) + \cdots + \frac{1}{i!}m^ih^{i+1}y^{(i+1)}(x) + \cdots \end{aligned}$$

Put these expansions into (1.34) and we have

$$L[y(x);h] = C_0 y(x) + C_1 hy'(x) + \cdots + C_q h^q y^{(q)}(x) + \cdots \tag{1.35}$$

where the coefficients C_q satisfy:

$$\begin{aligned} C_1 &= \alpha_0 + \alpha_1 + \cdots + \alpha_k \\ C_1 &= \alpha_1 + 2\alpha_2 + \cdots + k\ \alpha_k - (\beta_0 + \beta_1 + \cdots + \beta_k) \\ &\vdots \\ C_q &= \tfrac{1}{q!}(\alpha_1 + 2^q\alpha_2 + \cdots + k^q\alpha_k) - \tfrac{1}{(q-1)!}(\beta_1 + 2^{q-1}\beta_2 + \cdots k^{q-1}\beta_k). \end{aligned} \tag{1.36}$$

We say that the differential operator is of order p if

$$C_0 = C_1 = \cdots = C_p = 0, \quad C_{p+1} \neq 0. \tag{1.37}$$

Thus

$$L[y(x);h] = \mathcal{O}(h^{p+1}) \tag{1.38}$$

Table 1.4: Adams formulas

Adams-Bashforth						
i	0	1	2	3	4	5
β_{0i}	1					
$2\beta_{1i}$	3	-1				
$12\beta_{2i}$	23	-16	5			
$24\beta_{3i}$	55	-59	37	-9		
$720\beta_{4i}$	1901	-2774	2616	-1274	251	
$1440\beta_{5i}$	4227	-7673	9482	-6798	2627	-425
Adams-Moulton						
i	0	1	2	3	4	5
β_{0i}	1					
$2\beta_{1i}$	1	1				
$12\beta_{2i}$	5	8	-1			
$24\beta_{3i}$	9	19	-5	1		
$720\beta_{4i}$	251	646	-264	106	-19	
$1440\beta_{5i}$	475	1427	-798	482	-173	27

and the local error is $\mathcal{O}(h^{p+1})$, the global error is $\mathcal{O}(h^p)$. The process of finding the coefficients α and β so that (1.37) is satisfied is called the method of unknown coefficients. A method of order p approximates exactly a solution which is a polynomial of order not more than p. A necessary condition for getting the exact solution as $h \to 0_+$ is that the order of the adjoint differential operator is at least 1, i.e. $C_0 = 0$ and $C_1 = 0$. For k odd, the order of a stable operator cannot be greater than $k+1$. For k even, the order of a stable operator cannot be greater than $k+2$. To get $p = k+2$ all the roots of $\varrho(\xi)$ must be on the unit circle (in absolute value equal to 1) and the formula is designed so that as many as possible of the constants $C_0, C_1, C_2, \dots$ vanish.

1.4 Adams formulas

We present some special multi-step methods. Adams formulas have only two nonzero coefficients α_i in (1.32), namely the coefficients with the highest index. They split into two groups, explicit Adams-Bashforth formulas (with $\beta_k = 0$) and implicit Adams-Moulton formulas (with $\beta_k \neq 0$). Adams-Bashforth formulas are often written

$$y_{p+1} - y_p = h \sum_{i=0}^{q} \beta_{qi} f_{p-i} \,. \tag{1.39}$$

The coefficients β_{qi} are listed in Table 1.4. For $q = 0$ we have the Euler's method. For $q = 1$ we have

$$y_{p+1} = y_p + h\frac{(3f_p - f_{p-1})}{2} \,. \tag{1.40}$$

It is important that the wanted value y_{p+1} appears in (1.39) linearly and thus can be expressed explicitly. This is different in Adams-Moulton methods which are implicit

$$y_p - y_{p-1} = h \sum_{i=0}^{q} \beta_{qi} f_{p-i} \; . \qquad (1.41)$$

Here the wanted value y_p appears also in the nonlinear right hand side in f_p. To solve the nonlinear system of (algebraic) equations (1.41) with y and f being vectors, we must use some iteration method. Often a simple iteration

$$y_p^{\text{new}} - y_{p-1} = h\beta_{q0} f(x_p, y_p^{\text{old}}) + h \sum_{i=1}^{q} \beta_{qi} f_{p-i} \qquad (1.42)$$

is used which converges for sufficiently small h.

The coefficients for Adams-Moulton methods are given in Table 1.4. For $q = 0$ we have

$$y_p = y_{p-1} + hf_p \; , \qquad (1.43)$$

which can be called the "implicit Euler's method". Pro $q = 1$ we get

$$y_p = y_{p-1} + h(f_p + f_{p-1})/2 \; , \qquad (1.44)$$

which is called the trapezoidal rule (note the similarity with the formula for numerical evaluation of a definite integral with the same name).

The global error of the Adams-Bashforth formulas (1.39) is $\mathcal{O}(h^{q+1})$, for Adams-Moulton formulas (1.41) we get also $\mathcal{O}(h^{q+1})$. However, the order of the implicit methods is higher by one for the same number of the node points. The disadvantage being the implicit character of the method and the need to iterate. A combination of an explicit and an implicit method gives the "predictor - corrector" method. The explicit method is used as a predictor to get the initial value of y_p to use in the iteration in the implicit method. When we combine the Adams-Bashforth and the Adams-Moulton method of the 2.nd order we get the final "predictor - corrector" method of the 2.nd order

$$\begin{aligned} \bar{y} &= y_{p-1} + h(3f_{p-1} - f_{p-2})/2 \\ y_p &= y_{p-1} + h(f(x_p, \bar{y}) + f_{p-1})/2 \; . \end{aligned} \qquad (1.45)$$

There are many predictor - corrector methods. Also besides Adams methods, there are other methods, as Nyström's methods and Milne-Simpson methods to name a few. More details can be found in the original literature.

All the multi-step methods have one big disadvantage: it is not possible to start the computation just with knowledge of the initial condition. These methods require the knowledge of the solution (and its derivatives) in a few nodes, one of them being the point where the initial condition is given. To get this information various means are used, we mention here the two simplest ones: using the Taylor's expansion when the function f is easy to differentiate and the Runge-Kutta method otherwise. It is important to use a method with the order not less than the order of the multi-step method used later. Using a high order of the multi-step method has no sense if the first few points are computed with a large error. Asymptotically (for $h \to 0$) the resulting method would have the order of the starting method, if it is lower than the order of the multi-step method used later. Using multi-step methods for systems of differential equations is formally the same, now y and f being vectors. The advantage of multi-step methods as compared to single-step methods is that the number of evaluations of the right hand side f is much lower for the same order of the method. The disadvantage is the need of starting values. Also it is difficult to adjust the step size h automatically so the effectiveness of these methods is reduced especially for cases when the solution changes its character considerably.

1.5 Numerical methods for stiff systems

Many physical problems lead to differential equations where the eigenvalues of the linearized system differ by several orders of magnitude, or they also change during integration. Such systems are called stiff. In what follows we try to define stiff systems and we show their properties important for numerical integration. To start with, consider a system of linear differential equations with constant coefficients

$$\boldsymbol{y}' = \mathbf{A}\boldsymbol{y} \, , \tag{1.46}$$

where $y = (y_1, y_2, y_3)^T$ and the matrix $\mathbf{A}$ is

$$\mathbf{A} = \begin{pmatrix} -0.1 & -49.9 & 0 \\ 0 & -50 & 0 \\ 0 & 70 & -120 \end{pmatrix} . \tag{1.47}$$

The reader is invited to write the general solution of (1.46). For initial condition

$$y_1(0) = 2 \quad y_2(0) = 1 \quad y_3(0) = 2 \, . \tag{1.48}$$

we get

$$y_1(x) = \exp^{-0.1x} + \exp^{-50x} \, , \quad y_2(x) = \exp^{-50x} \, , \quad y_3(x) = \exp^{-50x} + \exp^{-120x} \, . \tag{1.49}$$

The eigenvalues of the matrix $\mathbf{A}$ are

$$\lambda_1 = -120 \, , \quad \lambda_2 = -50 \, , \quad \lambda_3 = -0.1 \, . \tag{1.50}$$

The solutions y_1, y_2 and y_3 have quickly decreasing terms corresponding to the eigenvalues λ_1 and λ_2, which are negligible after a short period of x. After this short transient period, where the terms corresponding to λ_1 and λ_2 are not negligible, we could continue with numerical integration with a step size h determined by approximation of the term corresponding to λ_3. For a stable numerical integration most methods require that $|h\lambda_i|$, $i = 1, 2, \ldots$ be bounded by some small value roughly between 1 and 10 (here h is the integration step size and λ_i are the eigenvalues of the right hand side). As λ_1 is the largest in absolute value of the eigenvalues of the matrix $\mathbf{A}$, the stability of the method is given by the value $|120h|$. E.g. for the Euler's method we need $|120h| < 2$, giving the largest possible step size being $h = 1/60$.

Let us derive this result for the system (1.46) with the matrix (1.47). The Euler's method is

$$\boldsymbol{y}^{n+1} = \boldsymbol{y}^n + h\mathbf{A}\boldsymbol{y}^n = (\mathbf{E} + h\mathbf{A})\boldsymbol{y}^n \, . \tag{1.51}$$

As the eigenvalues of the matrix $\mathbf{A}$ are in the left complex half-plane then for $n \to \infty$ it should be that $\boldsymbol{y}^n \to \mathbf{0}$. This is given by the eigenvalues of the matrix

$$(\mathbf{E} + h\mathbf{A}) = \begin{pmatrix} 1 - 0.1h & -49.9h & 0 \\ 0 & 1 - 50h & 0 \\ 0 & 70h & 1 - 120h \end{pmatrix} . \tag{1.52}$$

The eigenvalues of the matrix $(\mathbf{E} + h\mathbf{A})$ are $\lambda_1 = 1 - 0.1h$, $\lambda_2 = 1 - 50h$, $\lambda_3 = 1 - 120h$. To get $\boldsymbol{y}^n \to \mathbf{0}$ it is necessary that all the eigenvalues of the matrix $(\mathbf{E} + h\mathbf{A})$ lie inside the unit circle. This gives the condition $h < \frac{1}{60}$.

Although the term corresponding to λ_1 is negligible, the stability condition requires a very small integration step size h. As a result the integration is slow, often unnecessarily precise, without the possibility to integrate less precise. We say a system of differential equations is stiff if it is stable i.e. its eigenvalues have negative real parts and these differ by several orders of magnitude. If the system $\boldsymbol{y}' = \boldsymbol{f}(\boldsymbol{y})$ of ordinary differential equations is nonlinear, it is characterized by the eigenvalues the Jacobi matrix $\{\frac{\partial \boldsymbol{f}}{\partial \boldsymbol{y}}\}$ of the right hand side. If in a linear system the matrix $\mathbf{A}$ depends on the independent variable x, i.e. $\mathbf{A} = \mathbf{A}(x)$, then the eigenvalues may differ with x similarly as in the nonlinear system.

Dahlquist defined the so called A-stability (absolute stability) this way. Consider the scalar equation

$$y' = \lambda y \tag{1.53}$$

with $\operatorname{Re} \lambda < 0$. We say a numerical integration method generating the sequence $y_n \doteq y(x_n)$ with the integration step size h is A-stable (absolutely stable) if in the recurrent relation describing the method used to solve (1.53)

$$y_{n+1} = P(h\lambda) y_n \tag{1.54}$$

the quantity P (depending on $h\lambda$) satisfies

$$|P(h\lambda)| < 1 \tag{1.55}$$

for arbitrarily large step size h, assuming $\operatorname{Re} \lambda < 0$. This definition means

$$|y_n| \to 0 \,, \qquad n \to \infty \tag{1.56}$$

for any $h > 0$ assuming $\operatorname{Re} \lambda < 0$. There are modifications of this definition, e.g. a method is called L-stable if

$$|P(h\lambda)| \to 0 \,, \qquad h \to \infty \,. \tag{1.57}$$

The problem of stiff systems has two sides: stability and accuracy. If we use a method that is not absolutely stable, i.e. the region of $h\lambda$ satisfying (1.55) does not cover the entire left complex half plane, eigenvalues with large negative part require a very small integration step size, so that the integration is not effective. If an absolutely stable method is used there are no problems with stability, but the term corresponding to the largest eigenvalues in absolute value may be approximated not very precisely for some values of the step size h.

Table 1.5: Coefficients of semi-implicit Runge-Kutta methods

Method	Rosenbrock	Rosenbrock	Calahan
order	2.	3.	3.
a_1	$1-\sqrt{2}/2$	1.40824829	0.788675134
a_2	$1-\sqrt{2}/2$	0.59175171	0.788675134
b_1	$(\sqrt{2}-1)/2$	0.17378667	0.788675134
c_1	0	0.17378667	0
w_1	0	-0.41315432	0.75
w_2	1	1.41315432	0.25

1.6 Implicit single-step methods

It is easy to show that none of the explicit Runge-Kutta methods presented in Table 1.3 is A-stable. E.g. consider the improved Euler's method (1.25). For the differential equation (1.53) and the step size h we get

$$y_{n+1} = \Big[1 + h\lambda + \frac{1}{2}h^2\lambda^2\Big] y_n = P(h\lambda) y_n \,. \tag{1.58}$$

It is easy to show that for $h\lambda = -4$ we have $P(h\lambda) = 5$ and thus this method is not A-stable. Most of the A-stable methods are implicit, with the disadvantage to solve a system of nonlinear algebraic equations in each integration step using some iteration method. The Newton's method (or a similar iteration method) can be used. The initial approximation is usually good enough to use 1 to 3 iterations in each step. We show an example of a semi-implicit Runge-Kutta method without the need of iteration.

Consider an autonomous system of differential equations

$$\boldsymbol{y}' = \boldsymbol{f}(\boldsymbol{y}).$$

The method can be described by this algorithm:

$$\begin{aligned} \boldsymbol{k}_1 &= h\Big(\mathbf{E} - ha_1\mathbf{J}(\boldsymbol{y}_n)\Big)^{-1}\boldsymbol{f}(\boldsymbol{y}_n) \\ \boldsymbol{k}_2 &= h\Big(\mathbf{E} - ha_2\mathbf{J}(\boldsymbol{y}_n + c_1\boldsymbol{k}_1)\Big)^{-1}\boldsymbol{f}(\boldsymbol{y}_n + b_1\boldsymbol{k}_1) \end{aligned} \tag{1.59}$$

$$\boldsymbol{y}_{n+1} = \boldsymbol{y}_n + w_1\boldsymbol{k}_1 + w_2\boldsymbol{k}_2. \tag{1.60}$$

Here $\mathbf{J}(\boldsymbol{y}) = \{\partial \boldsymbol{f}/\partial \boldsymbol{y}\}$ is the Jacobi matrix of the right hand side. The coefficients a_1, a_2, b_1, c_1, w_1 and w_2 are shown in Table 1.5. All these methods are A-stable as can be verified by applying them to the equation (1.53). Note that to find $\boldsymbol{k}_1$ and $\boldsymbol{k}_2$ the evaluation of the Jacobi matrix is needed (for the Rosenbrock method of order 3 two evaluations are needed) and also solving a system of linear algebraic equations (instead of computing the inverse matrix) is necessary. No iteration method is needed unlike the implicit methods.

There are many semi-implicit Runge-Kutta methods, here we showed only three of them. One of the first A-stable methods is the trapezoidal rule (1.44). Substituting into (1.53) we get

$$P(h\lambda) = \frac{1 + h\lambda/2}{1 - h\lambda/2}. \tag{1.61}$$

For $h\lambda$ from the left complex half-plane we have $|P(h\lambda)| < 1$ and thus the method is A-stable. However for $|h\lambda| \to \infty$ we have $|P(h\lambda)| \to 1$, and thus this method is not L-stable. Note that we have to use some iteration method to find y_p from (1.44) if the function f is nonlinear.

Another example of an A-stable method is the implicit Euler's method as a special case of Adams-Moulton methods for $k = 0$ (see Table 1.3). This method is L-stable (verify it yourself) but its order in only 1 and thus it is not very effective. For solution of stiff problems free software is available, let us mention LSODE as an example.

* * *

For further study see [?], [?], [?], [?], [?], [?], [?], [?], [?], [?], [?], [?], [?].