Problems involving ordinary differential equations (ODEs) can always be reduced to the study of sets of first-order differential equations. For example the second-order equation

$$\frac{d^2y}{dx^2} + q(x)\frac{dy}{dx} = r(x) \tag{17.0.1}$$

can be rewritten as two first-order equations,

$$\begin{aligned} \frac{dy}{dx} &= z(x) \\ \frac{dz}{dx} &= r(x) - q(x)z(x) \end{aligned} \tag{17.0.2}$$

The generic problem in ordinary differential equations is thus reduced to the study of a set of N coupled *first-order* differential equations for the functions y_i, $i = 0, 1, \ldots, N-1$, having the general form

$$\frac{dy_i(x)}{dx} = f_i(x, y_0, \ldots, y_{N-1}), \qquad i = 0, \ldots, N-1 \tag{17.0.3}$$

where the functions f_i on the right-hand side are known.

Initial value problem using vector notation:

$$y'(x) = f(x, y) \quad y(x_0) = a; \quad x \geq x_0$$

where $x \in \mathbb{R}$ and $y(x)$ is an N-dimensional function. The N-dimensional vector a is called the *Initial Condition.*

Example:

$$\begin{aligned} u'(x) &= u(x)\cos(v(x)) \quad u(0) = 1 \\ v'(x) &= -u(x)^3 \qquad v(0) = \frac{\pi}{2} \end{aligned}$$

$$\begin{aligned} N &= 2 \\ x_0 &= 0 \\ a &= \begin{pmatrix} 1 \\ \frac{\pi}{2} \end{pmatrix} \\ y(x) &= \begin{pmatrix} u(x) \\ v(x) \end{pmatrix} \\ f(x, y) &= \begin{pmatrix} u(x)\cos(v(x)) \\ -u(x)^3 \end{pmatrix} \end{aligned}$$

Numerical solution:

In numerical solutions to ordinary differential equations, we first define a stepsize h and

$$\begin{aligned} x_n &= x_0 + nh \quad n = 0, 1, 2, \ldots \\ y_n &\simeq y(x_n) \qquad n = 0, 1, 2, \ldots \end{aligned}$$

where y_n is the numerical approximation to the true value $y(x_n)$.

Explicit one-step methods (Runge-Kutta methods):

$$y_{n+1} = F(x, h, y_n, f)$$

(the function $F(x, h, y_n, f)$ is often computed with some set of sequential substeps).

We will often write this as

$$y_{n+1} = F(x, h, y_n, f) + \mathcal{O}(h^{k+1})$$

where the last term indicates that if we would have $y_n \equiv y(x_n)$, then we would get $\|y_{n+1} - y(x_{n+1}\| = \mathcal{O}(h^{k+1})$, where k is called the *order* of the numerical method.

1st order Runge-Kutta (Euler): $$y_{n+1} = y_n + hf(x_n, y_n) + O(h^2)$$

2nd order Runge-Kutta (Midpoint):

$$k_1 = hf(x_n, y_n)$$

$$k_2 = hf\left(x_n + \tfrac{1}{2}h, y_n + \tfrac{1}{2}k_1\right)$$

$$y_{n+1} = y_n + k_2 + O(h^3)$$

4th order Runge-Kutta:

$$
\begin{aligned}
k_1 &= hf(x_n, y_n) \\
k_2 &= hf(x_n + \tfrac{1}{2}h, y_n + \tfrac{1}{2}k_1) \\
k_3 &= hf(x_n + \tfrac{1}{2}h, y_n + \tfrac{1}{2}k_2) \\
k_4 &= hf(x_n + h, y_n + k_3) \\
y_{n+1} &= y_n + \tfrac{1}{6}k_1 + \tfrac{1}{3}k_2 + \tfrac{1}{3}k_3 + \tfrac{1}{6}k_4 + O(h^5)
\end{aligned}
$$

Function evaluations:

As usual, we assume that the computationally expensive part is the computation of the function f(x,y). For 1st order Runge-Kutta, we have one function evaluation per step. For 2nd order Runge-Kutta, it is two function evaluations and for 4th order Runge-Kutta, it is four function evaluations. Of course, we also seem to get higher accuracy. Our aim is to find the most suitable method that gives the accuracy we want with as few function evaluations as possible.

Example

$$\begin{aligned} u'(x) &= u(x)\cos(v(x)) \quad u(0) = 1 \\ v'(x) &= -u(x)^3 \qquad v(0) = \frac{\pi}{2} \end{aligned}$$

$$\begin{aligned} N &= 2 \\ x_0 &= 0 \\ a &= \begin{pmatrix} 1 \\ \frac{\pi}{2} \end{pmatrix} \equiv \quad \begin{aligned} u'(x) &= u(x)\cos(v(x)) \quad u(0) = 1 \\ v'(x) &= -u(x)^3 \qquad v(0) = \frac{\pi}{2} \end{aligned} \\ y(x) &= \begin{pmatrix} u(x) \\ v(x) \end{pmatrix} \\ f(x,y) &= \begin{pmatrix} u(x)\cos(v(x)) \\ -u(x)^3 \end{pmatrix} \end{aligned}$$

We can now perform one step with Euler with stepsize h to obtain

$$\begin{aligned} f(x_0, y_0) &= \begin{pmatrix} 0 \\ -1 \end{pmatrix} \\ y_1 &= y_0 + hf(x_0, y_0) = \begin{pmatrix} 1 \\ \frac{\pi}{2} \end{pmatrix} + h \begin{pmatrix} 0 \\ -1 \end{pmatrix} = \begin{pmatrix} 1 \\ \frac{\pi}{2} - h \end{pmatrix} \end{aligned}$$

Example

$$
\begin{aligned}
u'(x) &= u(x)\cos(v(x)) \quad u(0) = 1 \\
v'(x) &= -u(x)^3 \qquad\quad v(0) = \frac{\pi}{2}
\end{aligned}
$$

If we do one step with the midpoint method (2nd order Runge-Kutta), we get

$$
\begin{aligned}
f(x_0, y_0) &= \begin{pmatrix} 0 \\ -1 \end{pmatrix} \\
k_1 &= hf(x_0, y_0) = h \begin{pmatrix} 0 \\ -1 \end{pmatrix} \\
y_0 + \frac{1}{2}k_1 &= \begin{pmatrix} 1 \\ \frac{\pi}{2} - \frac{h}{2} \end{pmatrix} \\
k_2 &= hf(x_0 + \frac{h}{2}, y_0 + \frac{1}{2}k_1) = h \begin{pmatrix} \cos(\frac{\pi}{2} - \frac{h}{2}) \\ -1 \end{pmatrix} \\
y_1 &= y_0 + k_2 = \begin{pmatrix} 1 \\ \frac{\pi}{2} \end{pmatrix} + h \begin{pmatrix} \cos(\frac{\pi}{2} - \frac{h}{2}) \\ -1 \end{pmatrix} = \begin{pmatrix} 1 + h\cos(\frac{\pi}{2} - \frac{h}{2}) \\ \frac{\pi}{2} - h \end{pmatrix}
\end{aligned}
$$

Order of a numerical method (global order):

In applications, we will not be interested in the error after one step, but the error at some fixed x. If we use a a number of subdivisions

$$x_n = x_0 + nh \quad n = 0, \ldots, M$$

so that $x = Mh$, we get a first approximation of the error at x as

$$\|y_M - y(x)\| \simeq M * \mathcal{O}(h^{k+1})$$

Since $M = \frac{x}{h}$, we expect to get

$$\|y_M - y(x)\| \simeq \mathcal{O}(h^k)$$

which is why k (and not $k+1$) is called the order of the method.

The term for one step $\mathcal{O}(h^{k+1})$ is called the ”local order” of the method

Implicit one-step methods:

$$y_{n+1} = F(x, h, y_n, y_{n+1}, f)$$

Here only the Trapezoidal method:

$$y_{n+1} = y_n + \frac{h}{2}\left(f(x_n, y_n) + f(x_{n+1}, y_{n+1})\right) + \mathcal{O}(h^3)$$

Implementation:

Perform an Euler step

$$y^*_{n+1} = y_n + hf(x_n, y_n)$$

Then use Newtons method to solve the system of potentially non-linear equations

$$y_{n+1} - y_n - \frac{h}{2}\left(f(x_n, y_n) + f(x_{n+1}, y_{n+1})\right) = 0$$

using y^*_{n+1} as the initial guess. One or two iterations will usually do.

Two-step Leap-frog method:

$$y_{n+1} = y_{n-1} + 2hf(x_n, y_n) + \mathcal{O}(h^3)$$

Do an Euler step first to initiate the method with y_0 and y_1.

u'(x)=u(x)v(x)   u(0)=1
v'(x)=-u(x)^2   v(0)=1

Solve with Euler, Midpoint, Trapezoidal, Leap-frog and 4th order Runge-Kutta
Consider x=20 and estimate the order using Richardson
Subdivide h with 2 until you reach an accuracy on u(x) of 10^-6