Material for lectures: week of March 23rd

A Modified Euler method

We would like to improve on the naive Euler method, essentially improving on the error scaling as a function of stepsize from $\mathcal{O}(h)$ to $\mathcal{O}(h^2)$.

Considering as we did before, a simple test ODE:

$$x' = \frac{dx}{dt} = f(x,t)$$

we try a modification to the Euler method: evaluate f more often per step, matching more terms in the Taylor expansion thereby building a **higher-order scheme**.

The idea is to make a "trial" step to the midpoint of each subinterval (step). Use this midpoint to compute the full step across the interval i.e. to get from (x_n, t_n) to (x_{n+1}, t_{n+1}) we use the midpoint at $(x_{n+\frac{1}{2}}, t_{n+\frac{1}{2}})$.

Now, remember that we want to know x_{n+1} and that we know x_n at t_n, so to solve the ODE we also need to evaluate $\int_{x_n}^{x_{n+1}} f(x,t)dt$ which we can do using a Taylor expansion. Expand $f(x,t)$ in a Taylor series about the midpoint of the subinterval $[t_n, t_{n+1}]$ ie about $\frac{t_{n+1}-t_n}{2} = t_{n+\frac{1}{2}} = t_n + \frac{h}{2}$ as

$$f(x,t) \approx f(x_{n+\frac{1}{2}}, t_{n+\frac{1}{2}}) + (t - t_{n+\frac{1}{2}})\frac{df}{dt} + \ldots . \tag{1}$$

If $t = t_{n+\frac{1}{2}}$ i.e. the integral is evaluated around the midpoint then $(t - t_{n+\frac{1}{2}}) \to 0$ and

$$f(x,t) \approx f(x_{n+\frac{1}{2}}, t_{n+\frac{1}{2}}). \tag{2}$$

Substitute this in the expression for x_{n+1}

$$\begin{aligned} x_{n+1} &= x_n + hf(x,t) = x_n + hf(x_{n+\frac{1}{2}}, t_{n+\frac{1}{2}}) \\ &= x_n + hf(x_{n+\frac{1}{2}}, t_n + \frac{h}{2}). \end{aligned}$$

The only unknown on the RHS is $x_{n+\frac{1}{2}}$ and this we get from one Euler step:

$$x_{n+\frac{1}{2}} = x_n + \frac{h}{2} f(x_n, t_n), \tag{3}$$

taking care to note that here the step-size is $h/2$ to go from (x_n, t_n) to halfway across the interval $(x_{n+\frac{1}{2}}, t_{n+\frac{1}{2}})$.
The algorithm is the following:

Given $x_0 = x(t_0)$ the initial condition and considering $j = 1, \ldots, N$ with $N = (t_N - t_0)/h$ and $t_j = t_0 + jh$. Then evaluate in sequence

$$\begin{aligned} k_1 &= hf(x_j, t_j) \\ k_2 &= hf\left(x_j + \frac{k_1}{2}, t_j + \frac{h}{2}\right) \\ x_{j+1} &= x_j + k_2 \end{aligned}$$

This is called the Modified Euler or the *Second order Runge-Kutta Method.*

Exercise: Verify that the modified Euler solution in each step incurs an error $x_{n+1} = x_n + k_2 + \mathcal{O}(h^3)$ and has a global error of $\mathcal{O}(h^2)$.

The method is second order (in h) but requires two evaluations of f per step.

Looking again at our example problem $x' = x,\ x(0) = 1$ and comparing the result of the modified Euler algorithm with our naive Euler implementation yields the results shown in Figure 1 Just as before we can also monitor the error in the Euler approximate solution

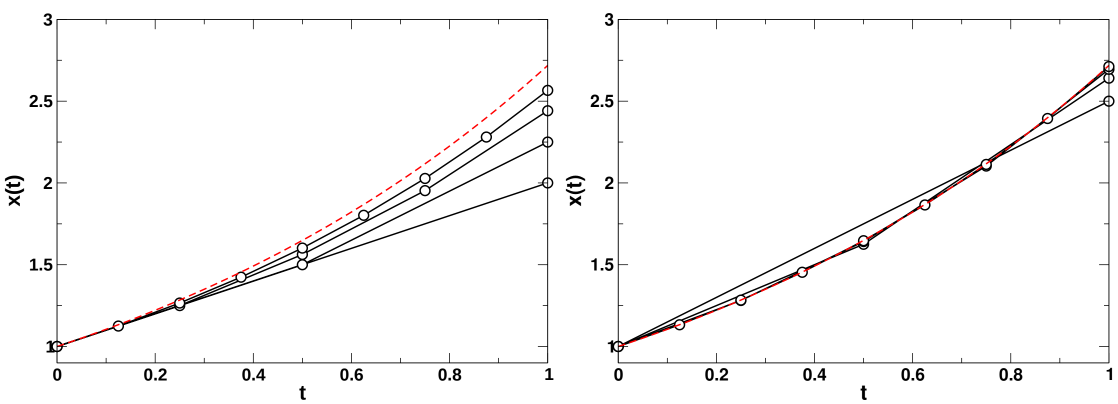

Figure 1: The left hand plot uses the simple Euler method to solve the ODE, the right-hand plot uses the modified Euler and you can see that it converges to the true solution (in red) much more quickly.

after N steps i.e. x_N compared to the true solution at the same time, $x(t)$ and the results from a numerical implementation are shown in Figure 2.

Higher-order methods

You can expand on the idea presented here and derive even higher-order formulae. The most popular is the *fourth order Runge Kutta* which for each step requires the following

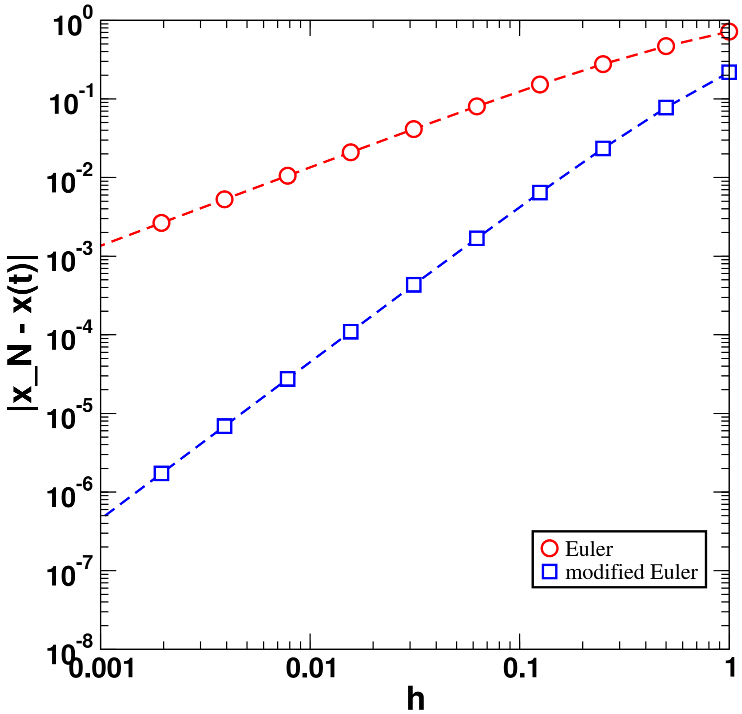

Figure 2: The error in the Euler and modified Euler methods as a function of the step size h.

evaluations:

$$\begin{aligned} k_1 &= hf(x_n, t_n) \\ k_2 &= hf(x_n + \frac{k_1}{2}, t_n + \frac{h}{2}) \\ k_3 &= hf(x_n + \frac{k_2}{2}, t_n + \frac{h}{2}) \\ k_4 &= hf(x_n + k_3, t_n + h) \end{aligned}$$

and finally

$$y_{n+1} = y_n + \frac{1}{6}\left(k_1 + 2k_2 + 2k_3 + k_4\right). \tag{4}$$

The global error (after n) steps is order 4, ie $x_n = x(t) + \mathcal{O}(h^4)$ and requires 4 evaluations of f per step. Figure 3 shows you how the error in the methods we have seen so far - Euler, modified Euler (aka 2nd order Runge-Kutta) and 4th order Runge Kutta - compare. The data are for the numerical solution of the same ODE seen in other plots ie $dx/dt = x(t), x(t = 0) = 1$.

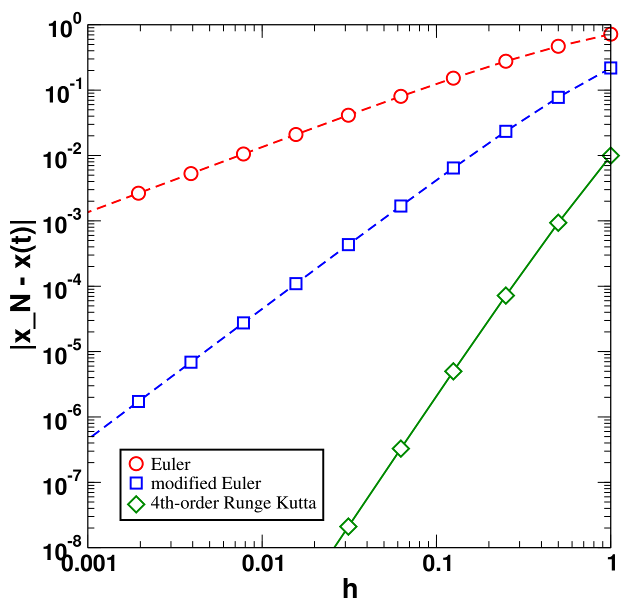

Figure 3: The (global) error in the numerical solution of $x' = x(t), x(0) = 1$, compared for Euler, modified Euler and 4th order Runge-Kutta, as described in the text.