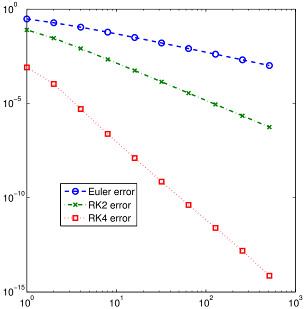
Initial Value Problems

See lecture notes for full details, and online notes for exercises

§2.6 Runge-Kutta 4

MA385 – Numerical Analysis 1

October 2017



It is possible to construct methods that have higher orders of accuracy than **RK2** methods. Of these, the most used are probably those that belong to the **Runge-Kutta 4 (RK4)** family, and have the property that

$$|y(t_n) - y_n| \leq Ch^4.$$

However, even writing down the general form of the RK4 method, and then deriving conditions on the parameters is rather complicated. Therefore, we'll focus on just one RK4 method, and use examples, rather than theory, to demonstrate that it is 4th-order.

. . . .

(*RK4 methods are optimal in a sense that they balance accuracy with efficiency: they are 4th-order, and have 4 stages, but proving that is beyond what we can cover in this module. Constructing even higher-order methods gets very complicated*).

$$
\begin{aligned}
k_1 &= f(t_i, y_i),\\
k_2 &= f(t_i + \frac{h}{2}, y_i + \frac{h}{2}k_1),\\
k_3 &= f(t_i + \frac{h}{2}, y_i + \frac{h}{2}k_2),\\
k_4 &= f(t_i + h, y_i + hk_3),\\
y_{i+1} &= y_i + \frac{h}{6}\big(k_1 + 2k_2 + 2k_3 + k_4\big).
\end{aligned}
$$

The RK4 method can be interpreted as follows:

As the following example shows, RK4 can be much more accurate than the Euler or RK2 methods for small h (i.e., large n). For the RK4, doubling n reduces the error by a factor of 8 (compared with 2 and 4 for the Euler and RK2 methods, respectively).

Example (2.4.1 (again))

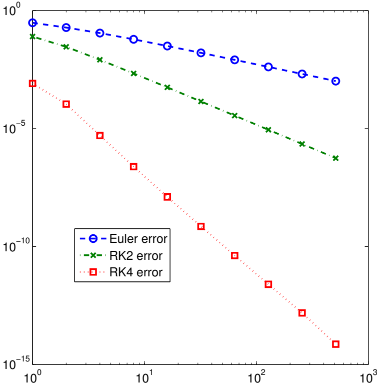
Compare Euler, Modified Euler, and RK4 for approximating $y(1)$ *where:* $y(0) = 1, \quad y'(t) = y\log(1+t^2)$.

Error: $|y(t_n) - y_n|$

n	Euler	Modified	RK4
1	3.02e-01	7.89e-02	8.14e-04
2	1.90e-01	2.90e-02	1.08e-04
4	1.11e-01	8.20e-03	5.07e-06
8	6.02e-02	2.16e-03	2.44e-07
16	3.14e-02	5.55e-04	1.27e-08
32	1.61e-02	1.40e-04	7.11e-10
64	8.13e-03	3.53e-05	4.18e-11
128	4.09e-03	8.84e-06	2.53e-12
256	2.05e-03	2.21e-06	1.54e-13
512	1.03e-03	5.54e-07	7.33e-15

Example (2.4.1 (again))

Compare Euler, Modified Euler, and RK4 for approximating $y(1)$ *where:* $y(0) = 1, \quad y'(t) = y \log(1 + t^2)$.



Although we won't do a detailed analysis of RK4, we can do a little. In particular, we would like to show it is

(i) consistent,

(ii) convergent and fourth-order, at least for some examples.

Example

It is easy to see that RK4 is consistent:

Example

In general, showing the rate of convergence is tricky. Instead, we'll demonstrate how the method relates to a Taylor Series expansion for the problem $y' = \lambda y$ where λ is a constant.

Many (seemingly different) RK have been proposed and studied. A unified approach of representing them was developed by John Butcher (Auckland, New Zealand): write an s-stage method as

$$\Phi(t_i, y_i; h) = \sum_{j=1}^{s} b_j k_j, \qquad \text{where}$$

$$\begin{aligned}
k_1 &= f(t_i + \alpha_1 h, y_i), \\
k_2 &= f(t_i + \alpha_2 h, y_i + \beta_{21} h k_1), \\
k_3 &= f(t_i + \alpha_3 h, y_i + \beta_{31} h k_1 + \beta_{32} h k_2), \\
&\ \vdots \\
k_s &= f(t_i + \alpha_s h, y_i + \beta_{s1} h k_1 + \ldots \beta_{s,s-1} h k_{s-1}),
\end{aligned}$$

The most convenient way to represent the coefficients is in a tableau:

$$
\begin{array}{c||ccccc}
\alpha_1 & & & & & \\
\alpha_2 & \beta_{21} & & & & \\
\alpha_3 & \beta_{31} & \beta_{32} & & & \\
\vdots & & & & & \\
\alpha_s & \beta_{s1} & \beta_{s2} & \cdots & \beta_{s,s-1} & \\
\hline\hline
 & b_1 & b_2 & \cdots & b_{s-1} & b_s
\end{array}
$$

The tableaux for basic Euler, Modified Euler, and RK4 are:

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

A Runge Kutta method has s stages if it involves s evaluations of the function f. (That it, its formula features $k_1, k_2, \dots, k_s$).

We've seen a 1-stage method that is 1^{st}-order.

We studied 2-stage methods that are 2^{nd}-order.

In an exercise, you'll construct a 3-stage method that is 3rd order.

And, of course, we have just considered a four-stage method that is 4^{th}-order.

It is tempting to think that for any s we can get a method of order s using s stages. However, it can be shown that, for example, to get a 5^{th}-order method, you need at least 6 stages; for a 7^{th}-order method, you need at least 9 stages. The theory involved is both intricate and intriguing, and involves aspects of group theory, graph theory, and differential equations. Students in third year might consider this as a topic for their final year project.