AM225: General structure of Runge–Kutta order conditions

In AM225 we have introduced several Runge–Kutta methods and examined their order of accuracy p. For low-order methods, the standard procedure is to Taylor expand the true solution and the numerical solution for a stepsize h, and compare terms up to $O(h^p)$. However, as the order of the method increases, these calculations become complicated and tedious. For example, at fourth order, there are eight separate conditions to satisfy, most of which are nonlinear in the Butcher tableau parameters.

In this document we consider the Runge–Kutta order conditions for arbitrary p. While the problem may at first seem intractable, we show that it is underpinned by a beautiful mathematical theory involving *trees*. The theory is practical and useful: it allows the order conditions for an arbitrary p to be immediately written down, makes it easy to estimate error for any method, and provides insight into how to derive high-order methods. This document is based on the excellent book by Hairer *et al.* [1]. It provides an abridged version of the complete derivation, and omits some of the longer proofs.

Preminary definitions

The document follows the notation of Hairer *et al.*, whereby a step of size h is taken from (x_0, y_0) to (x_1, y_1). The Runge–Kutta method has s intermediate steps, which are computed as

$$k_i = f\left(x_0 + c_i h, y_0 + h \sum_{j=1}^{i-1} a_{ij} k_j\right) \tag{1}$$

for $i = 1, \ldots, s$, after which the solution is given by

$$y_1 = y_0 + h \sum_{i=1}^{s} b_i k_i. \tag{2}$$

Here the coefficients a_{ij}, b_i, and c_i define the method. Usually the simplification $c_i = \sum_j a_{ij}$ is assumed, since it expresses that each of the k_i is evaluated at a first-order approximation of the solution.

As described in the slides, we can restrict attention to autonomous differential equations

$$y' = f(y) \tag{3}$$

for a vector function $y(x) \in \mathbb{R}^n$ without loss of generality. Using capital superscript indices to denote vector components this becomes

$$(y^J)' = f^J(y^1, y^2, \ldots, y^n) \tag{4}$$

for $J = 1, \ldots, n$. Rather than work with the Runge–Kutta steps directly, we work with their arguments g_i such that $k_i = f(g_i)$. Then

$$g_i^J = y_0 + \sum_{j=1}^{i-1} a_{ij} h f^J(g_j^1, \ldots, g_j^n) \tag{5}$$

for $i = 1, \dots, s$ and

$$y_1^J = y_0 + \sum_{j=1}^{s} b_j h f^J(g_j^1, \dots, g_j^n). \tag{6}$$

Note that there is a close correspondence between Eqs. 5 and 6, with the right hand sides being the same apart from replacing a_{ij} with b_j.

Taylor series: the true solution

We now aim to compare the Taylor series expansion of the true solution at $h = 0$ with the numerical one. We consider up to the third derivative in order to deduce a general pattern. For the true solution, the first derivative is

$$(y^J)^{(1)} = f^J(y). \tag{7}$$

The second derivative is

$$(y^J)^{(2)} = \sum_K f_K^J(y)(y^K)^{(1)} = \sum_K f_K^J(y) f^K(y), \tag{8}$$

where subscripts are used to denote derivatives, and so $f_K^J = \partial f^J / \partial y^K$ is the Jacobian. The third derivative is

$$(y^J)^{(3)} = \sum_{K,L} f_{KL}^J(y) f^K(y) f^L(y) + \sum_{K,L} f_K^J(y) f_L^K(y) f^L(y). \tag{9}$$

Taylor series: the numerical solution

For the numerical solution, it is useful to recall Leibniz' formula

$$\left.(h\varphi(h))^{(q)}\right|_{h=0} = \left.q(\varphi(h))^{(q-1)}\right|_{h=0} \tag{10}$$

for an arbitrary q-differentiable function $\varphi(h)$. We now consider the derivatives of g_i at $h = 0$. At zeroth order

$$\left.(g_i^J)^{(0)}\right|_{h=0} = y_0^J \tag{11}$$

and at first order

$$\left.(g_i^J)^{(1)}\right|_{h=0} = \left.\sum_j a_{ij} f^J\right|_{y=y_0}. \tag{12}$$

To proceed to higher order it is useful to first calculate the derivatives of $f^J(g_i)$. The first derivative is

$$(f^J(g_j))^{(1)} = \sum_K f_K^J(g_j)(g_j^K)^{(1)} \tag{13}$$

and the second derivative is

$$(f^J(g_j))^{(2)} = \sum_{K,L} f^J_{KL}(g_j)(g^K_j)^{(1)}(g^L_j)^{(1)} + \sum_K f^J_K(g_j)(g^K_j)^{(2)}. \tag{14}$$

Using these identities, the second and third derivatives of g^J_i at $h = 0$ are

$$(g^J_i)^{(2)}\Big|_{h=0} = 2\sum_{j,k} a_{ij}a_{jk} \sum_K f^J_K f^K \Big|_{y=y_0} \tag{15}$$

and

$$(g^J_i)^{(3)}\Big|_{h=0} = 3\sum_{j,k} a_{ij}a_{jk}a_{jl} \sum_{K,L} f^J_{KL} f^K f^L \Big|_{y=y_0} + 3\times 2\sum_{j,k,l} a_{ij}a_{jk}a_{kl} \sum_{K,L} f^J_K f^K_L f^L \Big|_{y=y_0}, \tag{16}$$

respectively. Using the correspondence mentioned above, the formulae in Eqs. 11, 12, 15, and 16 also apply to the derivatives of y_1 under the replacement of a_{ij} with b_j.

Example: a third-order method

To obtain a method of order p it is necessary for the numerical and true solutions to agree to that order. At first order, comparing Eqs. 7 and 12 shows that

$$\sum_j b_j = 1. \tag{17}$$

At second order, comparing Eqs. 8 and 15 shows that

$$2\sum_{j,k} b_j a_{jk} = 1. \tag{18}$$

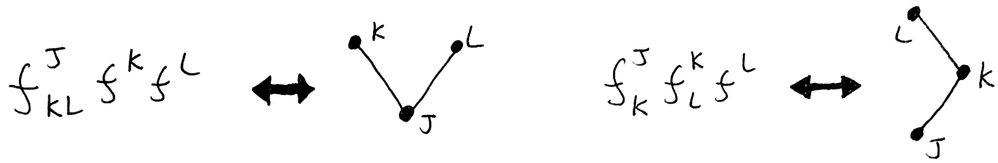
At third order, comparing Eqs. 9 and 16 results in two separate conditions:

$$3\sum_{j,k,l} b_j a_{jk} a_{jl} = 1, \qquad 6\sum_{j,k,l} b_j a_{jk} a_{kl} = 1. \tag{19}$$

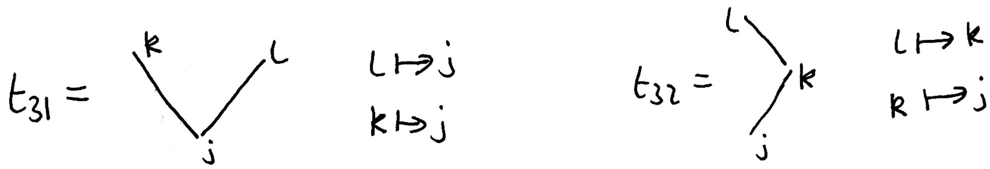
While the extension of this procedure to higher orders is conceptually clear, the calculations will rapidly become very complicated and thus we search for a different viewpoint.

The tree correspondence

To infer a general structure, note that equations such as Eq. 16 involve sums over pairs of indices. A given term such as $\sum_K f^J_K f^K$ can be alternatively represented a graph, where the superscript indices represent vertices and a summed index makes an edge between two vertices. At third order, this results in two graph, as shown below.



Note that since J is not summed over, it has a special status. Hence each graph can be interpreted as a tree with J at the root. In all terms computed in the Taylor series of the numerical solution, the corresponding sums over the lower-case indices exactly match in form. Thus the tree encapsulates both the sum over the a_{ij} terms and the sum over the f terms. We define a tree in terms of mappings to parent nodes, as shown below.



Definition 1 *Let $A_q = \{j < k < l < \ldots\}$ be an ordered chain of q indices. A* ***(rooted) labeled tree*** *is a mapping*

$$t : A_q \backslash \{j\} \to A_q \tag{20}$$

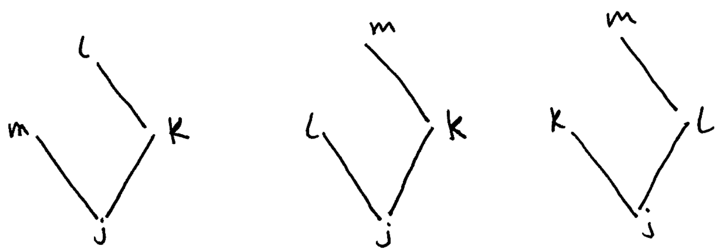
such that $t(z) < z$ for all $z \in A_q \backslash \{j\}$. The set of all labeled trees of order q is denoted by LT_q.

Definition 2 *For a labeled tree $t \in LT_q$, define*

$$F^J(t)(y) = \sum_{K,L,\ldots} f^J_{K,\ldots}(y) f^K_{\ldots}(y) f^L_{\ldots}(y) \ldots \tag{21}$$

to be a product of the f terms, where the superscript indices correspond to the vertices of the tree and each edge creates a summation over an lower–upper index pair.

Note that the three labeled trees



are topologically alike, and the corresponding differentials

$$\sum_{K,L,M} f^J_{KM} f^M f^K_L F^L, \qquad \sum_{K,L,M} f^J_{KM} f^L f^K_M F^M, \qquad \sum_{K,L,M} f^J_{KM} f^K f^L_M F^M \tag{22}$$

are identical. This motivates a further definition.

Definition 3 *Define a* ***tree*** *of order q to be the equivalence class of labeled trees under index permutations. Let $\alpha(t)$ be the number of elements in the equivalence class. Let T_q be the set of all trees of order q.*

The main results

With these definitions in place, we now show how the Taylor series expansions can be rewritten as sums over trees. Several of the theorems quoted here involve considerable work to prove (see Hairer *et al.* for full details), although where they originate from is conceptually clear.

Theorem 1 *The exact solution of the ODE satisfies*

$$y^{(q)}(x_0) = \sum_{t \in LT_q} F(t)(y_0) = \sum_{t \in T_q} \alpha(t) F(t)(y_0). \tag{23}$$

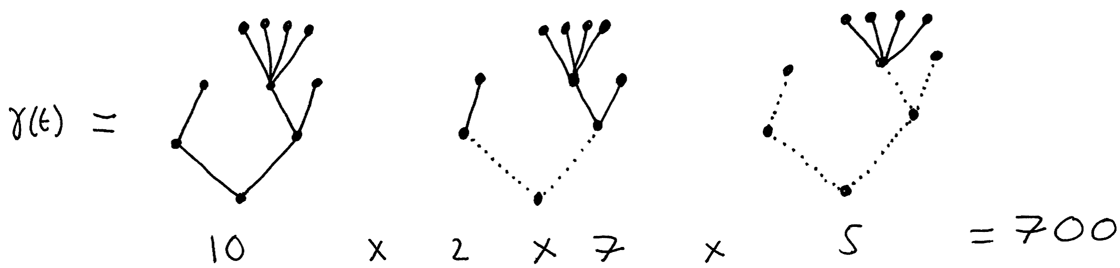
Definition 4 *For a labeled tree $t \in T_q$ denote*

$$\Phi_j(t) = \sum_{k,l,\ldots} a_{jk} a_{\ldots} \ldots \tag{24}$$

to be the sum over the $q-1$ remaining indices $k, l, \ldots$.

Definition 5 *Let $\rho(t)$ be the order of a tree. Define $\gamma(t)$ to be the product of $\rho(t)$ and all orders of trees that appear if the roots, one after another, are removed from t.*

For example,



Theorem 2 *The derivatives of g_i satisfy*

$$g_i^{(q)}\Big|_{h=0} = \sum_{t \in LT_q} \gamma(t) \sum_j a_{ij} \Phi_j(t) F(t)(y_0). \tag{25}$$

The numerical solution satisfies

$$\begin{aligned} y_1^{(q)}\Big|_{h=0} &= \sum_{t \in LT_q} \gamma(t) \sum_j b_j \Phi_j(t) F(t)(y_0) \\ &= \sum_{t \in T_q} \gamma(t) \alpha(t) \sum_j b_j \Phi_j(t) F(t)(y_0). \end{aligned} \tag{26}$$

By comparing Eqs. 23 and 26, we arrive at the main result.

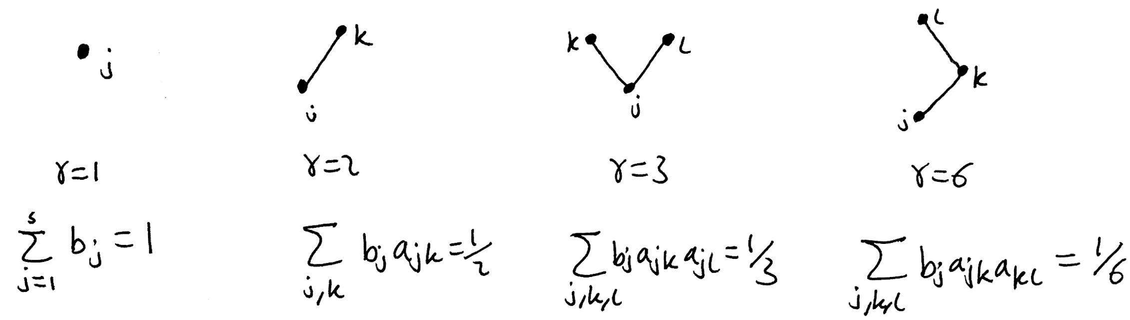
Theorem 3 *A Runge–Kutta method is of order p if and only if*

$$\sum_{j=1}^{s} b_j \Phi_j(t) = \frac{1}{\gamma(t)} \tag{27}$$

for all trees of order less than or equal to p.

The "if" part of Theorem 3 follows by equating terms in Eqs. 23 and 26. The "only if" part of the theorem is established by showing that for every term in Eqs. 23 and 26, there is an ODE system such that only that term is non-zero.

Theorem 3 provides us with a general procedure for finding the order conditions for arbitrary p: we must first find all trees, and then each one will give a corresponding condition. For example, for order three, we obtain the following.



This exactly matches the conditions that were found by direct calculation in Eqs. 17, 18, and 19. The number of conditions grows rapidly with the order p, as shown in the table below.

Order p	1	2	3	4	5	6	7	8	9	10
# trees	1	1	2	4	9	20	48	115	286	719
# conditions	1	2	4	8	17	37	85	200	486	1205

This makes it increasingly difficult to find high-order methods. For $p \geq 5$, it is no longer possible to find s-stage methods with $s = p$, and it becomes necessary to consider $s > p$.

References

[1] E. Hairer, S. P. Nørsett, and G. Wanner, *Solving Ordinary Differential Equations I: Nonstiff Problems*. Springer, 1993.