Chapter 4
Algebraic analysis and integration methods

4.1 Introduction

Integration methods [14] are a generalization of Runge–Kutta methods in which the index set for the stages is no longer restricted to a finite set $\{1,2,\dots,s\}$. If the index set is a closed interval, or more generally, the union of a finite number of closed intervals, they are referred to as “continuous Runge–Kutta methods”..

A Runge–Kutta method can be written

$$\begin{aligned} Y_i &= y_0 + h\sum_{j\in I} a_{ij} f(Y_j), \quad i \in I = \{1,2,\dots,s\},\\ y_1 &= y_0 + h\sum_{i\in I} b_i f(Y_i), \end{aligned}$$

and, in contrast, a continuous Runge–Kutta method takes the form

$$\begin{aligned} Y(\xi) &= y_0 + h\int_{\xi\in I} a(\xi,\eta) f\big(Y(\eta)\big)\,\mathrm{d}\eta, \quad \xi \in I = [0,1],\\ y_1 &= y_0 + h\int_{\xi\in I} b(\xi) f\big(Y(\xi)\big)\,\mathrm{d}\xi. \end{aligned}$$

The composition group of general linear methods is a subgroup of the group of B-series, as is clear from Theorem 4.6B (p. 165). This provides an alternative approach to Theorem 3.9C (p. 139). The subgroups of B are important because of their applications to simplifying assumptions of Runge–Kutta.

Chapter outline

In Section 4.2 the definition and principal properties of integration methods and continuous Runge–Kutta methods will be introduced. This is followed in Section 4.3 by a discussion of the reducibility of Runge–Kutta mehods and, in Section 4.4, a

J. C. Butcher, *B-Series*, Springer Series in Computational Mathematics 55,
https://doi.org/10.1007/978-3-030-70956-3_4

similar analysis is given for general linear methods . In Section 4.5, compositions of Runge–Kutta methods are studied, followed by a consideration of the general linear case in Chapter 4.6. This section includes work on the composition of elementary weights.

The order-defining subgroup $1+\mathsf{O}_{p+1}$ is introduced in Section 4.7, where O_{p+1} can be thought of as an algebraic counterpart to $\mathscr{O}(h^{p+1})$. Further subgroups are also introduced. These include subgroups corresponding to simplifying assumptions, such as the assumptions introduced in [9]. The linear spaces B^* and B_0, and linear operators on these spaces, are introduced in section 4.8.

4.2 Integration methods

"Integration methods" refers to a generalization of Runge–Kutta methods introduced in [14] (Butcher, 1972). In place of the usual finite index set $\{1,2,\dots,s\}$, the stages can be indexed by an interval $[0,1]$ or some other set. We will assume this set is Hausdorff and compact [81] (Rudin, 1976).

For X a compact Hausdorff set, let $B(X)$ denote the Banach algebra of continuous real-valued functions on X. That is, $B(X)$ is a vector space and at the same time has a product defined pointwise over X.

Definition 4.2A An integration method is a triple (X,A,b), where A is a linear operator on $B(X)$ and b is a linear functional on $B(X)$.

In these definitions, the norm $\|A\|$ is defined as $\sup_{\|\xi\|\neq 0}\|A\xi\|/\|\xi\|$. The norm $\|b\|$ is defined as $\sup_{\|\xi\|\neq 0}|b\xi|/\|\xi\|$.

Given an N-dimensional initial value problem, $y'(x)=f\big(y(x)\big)$, $y(x_0)=y_0$, the "solution" using (X,A,b) is found from

$$Y=\mathbf{1}y_0+hA(f\circ Y), \tag{4.2 a}$$
$$y_1=y_0+hb(f\circ Y). \tag{4.2 b}$$

Theorem 4.2B The solution to (4.2 a), (4.2 b) exists and is unique, for $h<(L\|A\|)^{-1}$, where L is the Lipschitz constant for f.

Proof. The mapping $Y\mapsto \mathbf{1}y_0+hA(f\circ Y)$ is a contraction. □

Continuous Runge–Kutta methods

Let X be a closed interval I, for example, $I=[0,1]$. For these methods, it is convenient to interpret A, b as integrals

$$\int_0^1 A(\xi,\eta)Y(\eta)\,\mathrm{d}\eta,$$
$$\int_0^1 b(\eta)Y(\eta)\,\mathrm{d}\eta.$$

Example methods

Picard–Lindelöf theorem

The initial value problem,

$$y'(x) = f\big(y(x)\big),$$
$$y(x_0) = y_0,$$

and its solution y_1 at $x_0 + h$, can be recast as an integral equation and gives the Picard–Lindelöf theorem [30] (Coddington, Levinson,1955), in the form

$$y(x_0 + h\xi) = y_0 + h\int_0^\xi f\big(y(x_0 + h\xi)\big)\,\mathrm{d}\xi, \quad \xi \in [0,1], \tag{4.2 c}$$

$$y_1 = y_0 + h\int_0^1 f\big(y(x_0 + h\xi)\big)\,\mathrm{d}\xi. \tag{4.2 d}$$

This example of an integration method on $[0,1]$ will be referred to as "the Picard method". It can be written in the form

$$Y(\xi) = y(x_0) + h\int_0^\xi (f\circ Y)(\eta)\,\mathrm{d}\eta,$$
$$y_1 = y(x_0) + h\int_0^1 (f\circ Y)(\eta)\,\mathrm{d}\eta.$$

Just as for Runge–Kutta methods, specific integration methods can be characterized by their elementary weights. In particular, the method given by (4.2 c,4.2 d) has elementary weights $\Phi(\mathrm{t}) = 1/\mathrm{t}!$, as in the right-hand sides of Runge–Kutta order conditions. This is to be expected because we can identify the value of y_1 with the flow of $y'(x) = f\big(y(x)\big)$ through a stepsize h. That is, (4.2 d) can be regarded as a limiting case of a Runge–Kutta method as the order tends to infinity.

Average Vector Field method

The Average Vector Field method (AVF method) [80] (Quispel, McLaren, 2008), and other related energy-preserving methods, fits naturally into the family of integration methods. The AVF method can be written in the form

$$y(x_0 + h\xi) = y(x_0) + h\xi\int_0^\xi f\big(y(x_0 + h\eta)\big)\,\mathrm{d}\eta,$$
$$y_1 = y(x_0) + h\int_0^1 f\big(y(x_0 + h\eta)\big)\,\mathrm{d}\eta$$

or in the form

$$Y(\xi) = y(x_0) + h\int_0^{\xi} \xi (f \circ Y)(\eta)\,\mathrm{d}\eta, \tag{4.2 e}$$

$$y_1 = y(x_0) + h\int_0^{1} (f \circ Y)(\eta)\,\mathrm{d}\eta. \tag{4.2 f}$$

Similarity with Runge–Kutta methods

In the two equations in the Picard–Lindelöf formulation, (4.2 c) corresponds to the computation of the stages in a Runge–Kutta method, and (4.2 d) corresponds to the computation of the output to the step. That is, the integrals in these two lines correspond to

$$\begin{array}{c|c} c & A \\ \hline \end{array} \quad \text{and} \quad \begin{array}{c|c} \hline & b^{\mathsf{T}} \end{array} \quad \text{in} \quad \begin{array}{c|c} c & A \\ \hline & b^{\mathsf{T}} \end{array}, \quad \text{respectively.}$$

Similar remarks apply to the AVF method, with (4.2 e) corresponding to the stage calculation and (4.2 f) corresponding to the evaluation of the output.

In discussing integration methods, we will often find it convenient to distinguish the stage computation only, from the full method. We will use the expression "integration system" for the stages-only aspect of an integration method.

Elementary weights of integration methods

Recall the formulae for elementary weights given by Definition 3.6A (p. 125):

$$\begin{aligned} \varphi_i(\tau) &= \sum_{j=1}^{s} a_{ij}, \qquad\qquad i = 1,2,\dots s, \\ \Phi(\tau) &= \sum_{j=1}^{s} b_{ij}, \\ \varphi_i([\mathrm{t}_1\mathrm{t}_2\cdots\mathrm{t}_n]) &= \sum_{j=1}^{s} a_{ij}\prod_{k=1}^{n}\varphi_j(\mathrm{t}_k), \\ \Phi([\mathrm{t}_1\mathrm{t}_2\cdots\mathrm{t}_n]) &= \sum_{i=1}^{s} b_i\prod_{k=1}^{n}\varphi_i(\mathrm{t}_k). \end{aligned}$$

The generalization to integration methods on $[0,1]$ is given by re-interpreting this definition:

$$\begin{aligned} \varphi(\tau) &= A\mathbf{1}, \\ \Phi(\tau) &= b^{\mathsf{T}}\mathbf{1}, \\ \varphi([\mathrm{t}_1\mathrm{t}_2\cdots\mathrm{t}_m]) &= A\,\mathrm{dot}_{i=1}^{m}\,\varphi(\mathrm{t}_i), \\ \Phi([\mathrm{t}_1\mathrm{t}_2\cdots\mathrm{t}_m]) &= b^{\mathsf{T}}\,\mathrm{dot}_{i=1}^{m}\,\varphi(\mathrm{t}_i). \end{aligned}$$

Exercise 39 Find $\Phi(\mathrm{t})$ for $|\mathrm{t}| \le 4$ for the Picard method.

Exercise 40 Find $\Phi(\mathrm{t})$ for $|\mathrm{t}| \le 4$ for the average vector field method.

4.3 Equivalence and reducibility of Runge–Kutta methods

Flavours of equivalence

It will be convenient to consider only the stages of a generic Runge–Kutta method

$$\begin{array}{c|c} c & A \\ \hline \end{array}\,,$$

rather than a full tableau

$$\begin{array}{c|c} c & A \\ \hline & b^{\mathsf{T}} \end{array}\,.$$

Equivalence

For a given initial value problem $y' = f(y)$, $y(x_0) = y_0$, satisfying a Lipschitz condition, let Y_i, $i = 1, 2, \ldots, s$, be defined by

$$Y_i = y_0 + h\sum_{j=1}^{s} a_{ij} f(Y_j), \quad h \le h_0.$$

Definition 4.3A Two stages are equivalent, written $i \equiv j$, if for any such problem, $Y_i = Y_j$.

φ-equivalence

Recall the elementary weight vector of A given by Definition 3.6A (p. 125). In the present context, this is given by

$$\varphi = \begin{bmatrix} \varphi_1(\mathrm{t}) & \varphi_2(\mathrm{t}) & \cdots & \varphi_s(\mathrm{t}) \end{bmatrix}^{\mathsf{T}},$$

where $\varphi_i(\mathrm{t})$ is the appropriate elementary weight for the tableau

$$\begin{array}{c|c} c & A \\ \hline & e_i^{\mathsf{T}} A \end{array}\,.$$

Definition 4.3B Two stages are φ-equivalent, written $i \overset{\varphi}{\equiv} j$, if for all t, $\varphi_i(\mathrm{t}) = \varphi_j(\mathrm{t})$.

***P*-equivalence**

Let P be a partition of the stages of A; that is

$$P = \begin{bmatrix} P_1 & P_2 & \cdots & P_S \end{bmatrix},$$

where the components, $P_1, P_2, \dots P_S$, are disjoint, with union $\{1,2,\dots,s\}$.

Definition 4.3C A is P-reducible if for all $I, J = 1,2,\dots,S$,

$$\sum_{k \in P_J} a_{ik} = \sum_{k \in P_J} a_{jk}, \quad i, j \in P_I.$$

Definition 4.3D Two stages are P-equivalent, written $i \overset{P}{\equiv} j$ if there exists P such that A is P-reducible and i, j are in the same component.

The concept of P-equivalence is related to Hundsdorfer–Spijker reducibility [58] (Hundsdorfer, Spijker, 1981).

Main result on stage equivalence

Theorem 4.3E Let i, j be two stages. Then

$$i \equiv j \quad \text{if and only if} \quad i \overset{\varphi}{\equiv} j \quad \text{if and only if} \quad i \overset{P}{\equiv} j.$$

Proof.

$\boxed{\textbf{if } i \equiv j \textbf{ then } i \overset{\varphi}{\equiv} j}$

For given t, use Theorem 3.8D (p. 132), with $n = |\mathrm{t}|$, and $\theta_k = 0$ except for $\mathrm{t}_k = \mathrm{t}$, in which case $\theta_k = 1$. From the B-series for $x^\intercal Y_i$ and $x^\intercal Y_j$, we have $\varphi_i(\mathrm{t})h^n/\sigma(\mathrm{t}) + \mathscr{O}(h^{n+1}) = \varphi_j(\mathrm{t})h^n/\sigma(\mathrm{t}) + \mathscr{O}(h^{n+1})$, for $h < h_0$; and it follows that $\varphi_i(\mathrm{t}) = \varphi_j(\mathrm{t})$.

$\boxed{\textbf{if } i \overset{\varphi}{\equiv} j \textbf{ then } i \overset{P}{\equiv} j}$

Define P such that i, j are in the same component if and only if $i \overset{\varphi}{\equiv} j$. We will show that A is P-reducible. Let $V \in \mathbb{R}^s$ denote the vector space spanned by $\varphi(\mathrm{t})$ for all t. It can be seen that (a) $\mathbf{1} \in V$, (b) if $u, v \in V$ then $\mathrm{dot}_2\, uv \in V$, (c) if $u \in V$ then $Au \in V$. Define $\widehat{V} \in \mathbb{R}^S$ such that, for $v \in V$, $\widehat{v} \in \widehat{V}$ is defined by $v_i = \widehat{v}_I$, for $i \in P_I$. Because $\widehat{V}$ satisfies the conditions of Theorem 3.8A (p. 129), it contains members arbitrarily close to e_I, for $I = 1, 2, \dots, S$ so that V contains members arbitrarily close to x defined by $x_k = 1$ for $k \in P_i$ and $x_k = 0$ otherwise. Hence $(Ax)_\ell$ has constant value for all ℓ in any of the components of P.

$\boxed{\textbf{if } i \overset{P}{\equiv} j \textbf{ then } i \equiv j}$

For $\ell = 1, 2, \dots, s$, define the sequence $Y_\ell^{[k]}$, $k = 0, 1, 2, \dots$, by $Y_\ell^{[0]} = y_0$ and s

$$Y_\ell^{[k]} = y_0 + h \sum_{m=1}^{s} a_{\ell m} f\big(Y_m^{[k-1]}\big), \quad k \geq 1.$$

By induction, $Y_i^{[k]} = Y_j^{[k]}$ for all k. Hence, in the limit as $k \to \infty$, $Y_i = Y_j$. □

Reduced tableau

Definition 4.3F If the stages of a method are partitioned as in Definition 4.3C, the pre-reduced method is the S-stage-tableau

$$\begin{array}{c|c} \widehat{c} & \widehat{A} \\ \hline & \widehat{b}^{\mathsf{T}} \end{array} = \begin{array}{c|cccc} \widehat{c}_1 & \widehat{a}_{11} & \widehat{a}_{12} & \dots & \widehat{a}_{1S} \\ \widehat{c}_2 & \widehat{a}_{21} & \widehat{a}_{22} & \dots & \widehat{a}_{2S} \\ \vdots & \vdots & \vdots & & \vdots \\ \widehat{c}_S & \widehat{a}_{S1} & \widehat{a}_{S2} & \dots & \widehat{a}_{SS} \\ \hline & \widehat{b}_1 & \widehat{b}_2 & \dots & \widehat{b}_S \end{array}, \tag{4.3 a}$$

where

$$\begin{aligned} \widehat{c}_I &= c_i, & i \in P_I, \\ \widehat{a}_{IJ} &= \sum_{j \in P_J} a_{ij}, & i \in P_I, \\ \widehat{b}_I &= \sum_{i \in P_I} b_i. \end{aligned}$$

Essential stages of a pre-reduced method are defined recursively:

Definition 4.3G A stage J of (4.3 a) is essential if $\widehat{b}_J \neq 0$, or there exists an essential stage I such that $\widehat{a}_{IJ} \neq 0$.

For the method (4.3 a), a non-essential stage can be deleted. For example, the deletion of stage K gives

$$\begin{array}{c|cccccc}
\widehat{c}_1 & \widehat{a}_{11} & \cdots & \widehat{a}_{1,K-1} & \widehat{a}_{1,K+1} & \cdots & \widehat{a}_{1S} \\
\vdots & \vdots & & \vdots & \vdots & & \vdots \\
\widehat{c}_{K-1} & \widehat{a}_{K-1,1} & \cdots & \widehat{a}_{K-1,K-1} & \widehat{a}_{K-1,K+1} & \cdots & \widehat{a}_{K-1,S} \\
\widehat{c}_{K+1} & \widehat{a}_{K+1,1} & \cdots & \widehat{a}_{K+1,K-1} & \widehat{a}_{K+1,K+1} & \cdots & \widehat{a}_{K+1,S} \\
\vdots & \vdots & & \vdots & \vdots & & \vdots \\
\widehat{c}_S & \widehat{a}_{S1} & \cdots & \widehat{a}_{S,K-1} & \widehat{a}_{S,K+1} & \cdots & \widehat{a}_{SS} \\
\hline
& \widehat{b}_1 & \cdots & \widehat{b}_{K-1} & \widehat{b}_{K+1} & \cdots & \widehat{b}_S
\end{array},$$

Definition 4.3H A reduced method is a pre-reduced method from which all non-essential stages have been deleted.

Definition 4.3H is related to Dahlquist–Jeltsch reduction [39] (Dahlquist, Jeltsch, 2006).

Exercise 41 Find the reduced form of the tableau

$$\begin{array}{c|cccc}
\frac{1}{2} & \frac{1}{6} & 1 & -1 & \frac{1}{3} \\
\frac{2}{3} & \frac{1}{3} & 1 & -\frac{1}{3} & 0 \\
\frac{2}{3} & 1 & \frac{1}{3} & \frac{1}{3} & -\frac{2}{3} \\
\frac{1}{2} & 1 & -\frac{1}{2} & \frac{1}{2} & -\frac{1}{2} \\
\hline
& \frac{1}{2} & \frac{1}{4} & -\frac{1}{4} & \frac{1}{2}
\end{array}.$$

4.4 Equivalence and reducibility of integration methods

The aim of this section is to show how concepts associated with Runge–Kutta methods extend in a natural way to integration methods.

Reducible integration systems

Definition 4.4A The Banach algebra $B(X,A)$ is the closed sub-algebra of $B(X)$ containing the unit function $\mathbf{1}$, Av and uv whenever u and v are in the algebra.

Definition 4.4B $\xi_1,\xi_2 \in X$ are A-equivalent, written $\xi_1 \overset{A}{\equiv} \xi_2$, if for all $u \in B(X,A)$, $u(\xi_1) = u(\xi_2)$.

Definition 4.4C $\xi_1,\xi_2 \in X$ are φ-equivalent, written $\xi_1 \overset{\varphi}{\equiv} \xi_2$, if for all $\mathrm{t} \in \mathrm{T}$, $\varphi_{\xi_1}(\mathrm{t}) = \varphi_{\xi_2}(\mathrm{t})$.

Theorem 4.4D $\xi_1,\xi_2 \in X$ are φ-equivalent if and only if they are A-equivalent.

Proof. We will use the "height" of a tree introduced in Section 2.3 (p. 50). Define B_0 as the linear space spanned by $A\mathbf{1}$ and for $n = 1,2,\ldots$, define B_n as the Banach algebra containing uv whenever $u,v \in B_{n-1}$ and u and Au whenever $u \in B_{n-1}$. By induction, B_n contains $\varphi(\mathrm{t})$ for height$(\mathrm{t}) \le n$. Since every member of $B(X,A)$ lies in some B_n, $\xi_1 \overset{A}{\equiv} \xi_2$ implies that $\varphi_{\xi_1}(\mathrm{t}) = \varphi_{\xi_2}(\mathrm{t})$ for each t. □

On the basis of this result, we will use "equivalence" as a short-hand for either "φ-equivalence" or "A-equivalence", and use the notation $\xi_1 \equiv \xi_2$.

Pre-reduced and reduced methods

If (X,A,b) is an integration method then $\widehat{X} := X/\equiv$ will denote the set of equivalence classes into which X is partitioned. For $\widehat{xi} \in \widehat{X}$, $\widehat{\mathbf{1}}_{\widehat{\xi}}(\xi) = 1$, if $\xi \in \widehat{\xi}$, and 0 otherwise.

We also define $\widehat{b}_{\widehat{\xi}} = b\widehat{\mathbf{1}}_{\widehat{\xi}}$ and

$$\widehat{A}_{\widehat{\xi},\widehat{\xi}'} = A_{(\xi,\cdot)}\widehat{\mathbf{1}}_{\widehat{\xi}'}, \qquad \xi \in \widehat{\xi}.$$

Definition 4.4E Given an integration method (X,A,b), $(\widehat{X},\widehat{A},\widehat{b})$, is the corresponding pre-reduced method.

Equivalence of methods

Definition 4.4F Two integration methods are "equivalent" if their elementary weight functions are related by

$$\Phi(\mathrm{t}) = \overline{\Phi}(\mathrm{t}), \quad \mathrm{t} \in \mathrm{T}.$$

4.5 Compositions of Runge–Kutta methods

Given two Runge–Kutta methods

$$M = \begin{array}{c|cccc} c_1 & a_{11} & a_{12} & \cdots & a_{1s} \\ c_2 & a_{21} & a_{22} & \cdots & a_{2s} \\ \vdots & \vdots & \vdots & & \vdots \\ c_s & a_{s1} & a_{s2} & \cdots & a_{ss} \\ \hline & b_1 & b_2 & \cdots & b_s \end{array}\,, \qquad \overline{M} = \begin{array}{c|cccc} \overline{c}_1 & \overline{a}_{11} & \overline{a}_{12} & \cdots & \overline{a}_{1\overline{s}} \\ \overline{c}_2 & \overline{a}_{21} & \overline{a}_{22} & \cdots & \overline{a}_{2\overline{s}} \\ \vdots & \vdots & \vdots & & \vdots \\ \overline{c}_{\overline{s}} & \overline{a}_{\overline{s}1} & \overline{a}_{\overline{s}2} & \cdots & \overline{a}_{\overline{s}\overline{s}} \\ \hline & \overline{b}_1 & \overline{b}_2 & \cdots & \overline{b}_{\overline{s}} \end{array}\,,$$

a new tableau can be constructed with $s+\overline{s}$ stages:

$$M\overline{M} := \begin{array}{c|cccccccc} c_1 & a_{11} & a_{12} & \cdots & a_{1s} & 0 & 0 & \cdots & 0 \\ c_2 & a_{21} & a_{22} & \cdots & a_{2s} & 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & & \vdots & \vdots & \vdots & & \vdots \\ c_s & a_{s1} & a_{s2} & \cdots & a_{ss} & 0 & 0 & \cdots & 0 \\ c_{s+1} & b_1 & b_2 & \cdots & b_s & \overline{a}_{11} & \overline{a}_{12} & \cdots & \overline{a}_{1\overline{s}} \\ c_{s+2} & b_1 & b_2 & \cdots & b_s & \overline{a}_{21} & \overline{a}_{22} & \cdots & \overline{a}_{2\overline{s}} \\ \vdots & \vdots & \vdots & & \vdots & \vdots & \vdots & & \vdots \\ c_{s+\overline{s}} & b_1 & b_2 & \cdots & b_s & \overline{a}_{\overline{s}1} & \overline{a}_{\overline{s}2} & \cdots & \overline{a}_{\overline{s}\overline{s}} \\ \hline & b_1 & b_2 & \cdots & b_s & \overline{b}_1 & \overline{b}_2 & \cdots & \overline{b}_{\overline{s}} \end{array}\,,$$

where

$$c_{s+i} = \overline{c}_i + \sum_{j=1}^{s} b_j, \qquad i = 1,2,\dots,\overline{s}.$$

To see the significance of this construction, consider the application of the two Runge–Kutta methods in sequence so that y_1 is the result of applying M to y_0 and y_2 is the result of applying $\overline{M}$ to y_1. Let Y_1, Y_2, ..., Y_s denote the stages computed in the first of these steps and $\overline{Y}_1$, $\overline{Y}_2$, ..., $\overline{Y}_s$ the stages computed in the second step. Throughout this discussion, F_i and $\overline{F}_i$ will denote $f(Y_i)$ and $f(\overline{Y}_i)$, respectively.

We have

$$\begin{aligned} Y_i &= & y_0 + h\sum_{j=1}^{s} a_{ij}F_j, & \qquad i = 1,2,\dots,s, \\ y_1 &= y_0 + h\sum_{i=1}^{s} b_iF_i, & & \\ \overline{Y}_i &= y_1 + h\sum_{j=1}^{s} \overline{a}_{ij}\overline{F}_j = & y_0 + h\sum_{i=1}^{s} b_iF_i + h\sum_{j=1}^{s} \overline{a}_{ij}\overline{F}_j. & \qquad i = 1,2,\dots,\overline{s}, \end{aligned}$$

$$y_2 = y_1 + h\sum_{i=1}^{\bar{s}} \bar{b}_i \overline{F}_i \quad = y_0 + h\sum_{i=1}^{s} b_i F_i + h\sum_{i=1}^{\bar{s}} \bar{b}_i \overline{F}_i,$$

so that the tableau $M\overline{M}$ defines the composed Runge–Kutta method for computing y_2 directly from y_0.

The inverting method

Given the tableau

$$M = \begin{array}{c|cccc} c_1 & a_{11} & a_{12} & \cdots & a_{1s} \\ c_2 & a_{21} & a_{22} & \cdots & a_{2s} \\ \vdots & \vdots & \vdots & & \vdots \\ c_s & a_{s1} & a_{s2} & \cdots & a_{ss} \\ \hline & b_1 & b_2 & \cdots & b_s \end{array}$$

and the stage and output formulae

$$\begin{aligned} Y_i &= y_0 + h\sum_{j=1}^{s} a_{ij}F_j, \quad i = 1,2,\ldots,s, \\ y_1 &= y_0 + h\sum_{i=1}^{s} b_i F_i, \end{aligned}$$

we can solve for y_0 in terms of y_1 to obtain

$$\begin{aligned} Y_i &= y_1 + h\sum_{j=1}^{s} (a_{ij} - b_j)F_j, \quad i = 1,2,\ldots,s, \\ y_0 &= y_1 + h\sum_{i=1}^{s} (-b_i)F_i. \end{aligned}$$

This defines the inverting method

$$M^- = \begin{array}{c|cccc} c_1 - \sum_{i=1}^{s} b_i & a_{11} - b_1 & a_{12} - b_2 & \cdots & a_{1s} - b_s \\ c_2 - \sum_{i=1}^{s} b_i & a_{21} - b_1 & a_{22} - b_2 & \cdots & a_{2s} - b_s \\ \vdots & \vdots & \vdots & & \vdots \\ c_s - \sum_{i=1}^{s} b_i & a_{s1} - b_1 & a_{s2} - b_2 & \cdots & a_{ss} - b_s \\ \hline & -b_1 & -b_2 & \cdots & -b_s \end{array},$$

which exactly undoes the work of M.

The two composed methods MM^- and M^-M should, in some sense, be equivalent to the identity method with zero stages and a zero-dimensional b^T vector. To investigate this question we first reduce it to a single case by:

Theorem 4.5A Let M be a tableau and M^- denote its inverting method. Then

$$(M^-)^- = M.$$

The following theorem also applies to M^-M, because of Theorem 4.5A:

Theorem 4.5B Given a tableau M, the Runge–Kutta method MM^- applied to an initial value problem (f, y_0) with Lipschitz condition L, with hL sufficiently small, gives a result equal to y_0.

Proof. We will assume h is sufficiently small for a certain iteration scheme to be a contraction. To describe this iteration scheme, write $Y_i^{[k]}$ for iteration number k, and stage number i, in the step using M and $\overline{Y}_i^{[k]}$ for the corresponding iteration in the M^- step. For iteration zero, define

$$Y_i^{[0]} = \overline{Y}_i^{[0]} = y_0, \qquad i = 1, 2, \dots, s$$

and define subsequent iterations by

$$\begin{aligned}
Y_i^{[k]} &= y_0 + h \sum_{j=1}^{s} a_{ij} f\big(Y_j^{[k-1]}\big), \qquad i = 1, 2, \dots, s, \;\; k = 1, 2, \dots, \\
\overline{Y}_i^{[k]} &= y_0 + h \sum_{j=1}^{s} b_j f\big(Y_j^{[k-1]}\big) + h \sum_{j=1}^{s} (a_{ij} - b_j) f\big(\overline{Y}_j^{[k-1]}\big).
\end{aligned}$$

It can be verified, using induction on k, that

$$Y_i^{[k]} = \overline{Y}_i^{[k]}, \qquad i = 1, 2, \dots, s, \quad k = 1, 2, \dots$$

Hence, for the limiting values, $Y_i = \overline{Y}_i$, $i = 1, 2, \dots, s$. □

To look at this question in terms of reduced tableaux, we have the result:

Theorem 4.5C Given a tableau M, the reduced tableau of MM^- is the identity method.

Proof. The composite tableau is

$$\begin{array}{c|cc}
c & A & 0 \\
c & \theta\mathbf{1} & A - \theta\mathbf{1} \\
\hline
 & b^{\mathsf{T}} & -b^{\mathsf{T}}
\end{array},$$

where $\theta = b^{\mathsf{T}}\mathbf{1}$. According to Definition 4.3C (p. 156), the tableau is P-reducible, with $P = [\;\; \{1, s+1\} \quad \{2, s+2\} \quad \cdots \quad \{s, 2s\} \;\;]$, and the pre-reduced tableau is

$$\begin{array}{c|c} c & A \\ \hline & \mathbf{0}^{\mathsf{T}} \end{array},$$

with no essential stages and the reduced tableau of the identity Runge–Kutta method. □

4.6 Compositions of integration methods

If we have two integration methods $M_1 = (X_1, A_1, b_1^{\mathsf{T}})$ and $M_2 = (X_2, A_2, b_2^{\mathsf{T}})$, which are to be used sequentially,

$$\begin{aligned} Y_1 &= \mathbf{1}y_0 + hA_1(f \circ Y_1), \\ y_1 &= y_0 + hb_1^{\mathsf{T}}(f \circ 1), \\ Y_2 &= \mathbf{1}y_1 + hA_2(f \circ Y_2), \\ y_2 &= y_1 + hb_2^{\mathsf{T}}(f \circ Y_2). \end{aligned}$$

Conventionally, assume $X_1 \cap X_2 = \emptyset$ and write $X := X_1 \cup X_2$, $(X, A, b^{\mathsf{T}}) =: M = M_1 M_2$. For $\varphi : X \to \mathbb{R}^N$, write $\varphi_1 := \varphi\big|_{X_1}$ $\varphi_2 := \varphi\big|_{X_2}$. The operator A and the functional b^{T} are defined by

$$\begin{aligned} (A\varphi)(\xi) &= \begin{cases} (A_1\varphi_1)(\xi), & \xi \in X_1, \\ \mathbf{1}(b_1^{\mathsf{T}}\varphi_1)(\xi) + A_2(\varphi_2)(\xi), & \xi \in X_2, \end{cases} \\ b^{\mathsf{T}}(\varphi) &= (b_1^{\mathsf{T}}\varphi_1) + (b_2^{\mathsf{T}}\varphi_2). \end{aligned}$$

Elementary weights of composed methods

Theorem 4.6A Let $\Psi(\mathrm{t})$, $\Omega(\mathrm{t})$, and $\Phi(\mathrm{t})$ denote the elementary weights for M_1, M_2, and $M = M_1 M_2$, respectively. Then

$$\Phi(\mathrm{t}) = \Psi(\mathrm{t}) + \sum_{\mathrm{t}' \leq \mathrm{t}} \Psi(\mathrm{t} \smallsetminus \mathrm{t}')\Omega(\mathrm{t}').$$

Proof. Let $\mathrm{t} = (V, E, r)$ and let V' be a connected subset of V such that $V' = \emptyset$ or $r \in V'$. The possible choices of V' will be denoted by V'_k, $k = 0, 1, 2, \ldots, n$, with $V_0 = \emptyset$. For each such V'_k, t'_k will denote the corresponding subtree of t (with special case $\mathrm{t}'_0 = \varnothing$). Note that $V \smallsetminus V'_k$ defines $\mathrm{t} \smallsetminus \mathrm{t}'_k$. For conciseness, child$(i)$ will be written $C(i)$. For each $i \in V$, define $\psi_i \in B(X_1)$, $\omega_i \in B(X_2)$, by the recursion,

$$\begin{aligned}\psi_i &= A_1 \operatorname{dot}_{j\in C(i)} \psi_j,\\ \omega_i &= \mathbf{1}A_1 \operatorname{dot}_{j\in C(i)} \psi_j + A_2 \operatorname{dot}_{j\in C(i)} \omega_j,\\ \Phi(\mathrm{t}) &= b_1 \operatorname{dot}_{r\in C(i)} \psi_j + b_2 \operatorname{dot}_{r\in C(i)} \omega_j.\end{aligned} \tag{4.6 a}$$

For $V' = V'_k$, let Φ_k indicate that the recursions for (4.6 a) are replaced by

$$\psi_i = A_1 \operatorname{dot}_{j\in C(i)} \psi_j, \qquad i \neq r,$$

$$\omega_i = \begin{cases} \mathbf{1}b_1 \operatorname{dot}_{j\in C(i)} \psi_j, & i \notin V'_k, \quad i \neq r, & (4.6\,\mathrm{b})\\ A_2 \operatorname{dot}_{j\in C(i)} \omega_j, & i \in V'_k, \quad i \neq r, & (4.6\,\mathrm{c})\end{cases}$$

$$\Phi_k(\mathrm{t}) = \begin{cases} b_1 \operatorname{dot}_{j\in C(r)} \psi_j, & r \notin V'_k, & (4.6\,\mathrm{d})\\ b_2 \operatorname{dot}_{j\in C(r)} \omega_j, & r \in V'_k, & (4.6\,\mathrm{e})\end{cases}$$

so that $\Phi = \sum_{k=0}^{n} \Phi_k$. The evaluation of Φ_0 involves the iterations (4.6 b), (4.6 d), so that $\Phi_0 = \Psi(\mathrm{t})$. In (4.6 c), (4.6 e), write $C(i) = D(i) \cup E(i)$, where $D(i) \subset V \setminus V'_k$, $E(i) \subset V'_k$. For $k > 0$, rewrite (4.6 c), (4.6 e) in the form

$$\begin{aligned}\omega_i &= \Big(\prod_{j\in D(i)} b_1\psi_j\Big)\Big(A_2 \operatorname{dot}_{j\in E(i)} \omega_j\Big), \quad i \in V'_k, \quad i \neq r,\\ \Phi_k(\mathrm{t}) &= \Big(\prod_{j\in D(r)} b_1\psi_j\Big)\Big(b_2 \operatorname{dot}_{j\in E(r)} \omega_j\Big), \quad r \in V'_k,\end{aligned}$$

so that

$$\begin{aligned}\Phi_k(\mathrm{t}) &= \Big(\prod_{j=1}^{s} \prod_{j\in D(i)} b_1\psi_j\Big)\widetilde{\Phi}_k(\mathrm{t})\\ &= \Psi(\mathrm{t}\setminus\mathrm{t}')\widetilde{\Phi}_k(\mathrm{t}),\end{aligned}$$

where $\widetilde{\Phi}_k(\mathrm{t})$ is defined by the recursion

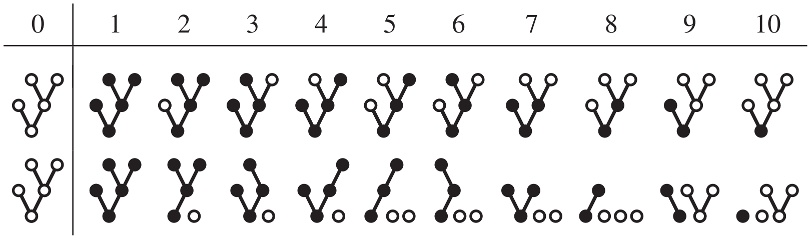
$$\begin{aligned}\omega_i &= A_2 \operatorname{dot}_{j\in E(i)} \omega_j, \quad i \in V'_k, \quad i \neq r,\\ \widetilde{\Phi}_k(\mathrm{t}) &= b_2 \operatorname{dot}_{j\in E(r)} \omega_j, \quad r \in V'_k,\end{aligned}$$

and it follows that $\widetilde{\Phi}_k(\mathrm{t}) = \Omega(\mathrm{t}')$ and

$$\Phi(\mathrm{t}) = \Phi_0(\mathrm{t}) + \sum_{k=1}^{n} \Psi(\mathrm{t}\setminus\mathrm{t}')\widetilde{\Phi}_k(\mathrm{t}) = \Psi(\mathrm{t}) + \sum_{k=1}^{n} \Psi(\mathrm{t}\setminus\mathrm{t}')\Omega_(\mathrm{t}'). \qquad \square$$

Comments on Theorem 4.6A

For a specific example of t, the choices of V' are illustrated in the upper line of (4.6 f). The 10 possible non-empty choices of V' are given, with vertices shown as $\bullet$. The remaining vertices of t are shown as o. In addition to $1,2,\ldots,10$, the special case headed 0 corresponds to $t' = \varnothing$. On the lower line of (4.6 f), t' is shown together with $t \smallsetminus t'$.



(4.6 f)

Recall Theorem 3.9C (p. 139), which we restate

> Theorem 4.6B (Reprise of Theorem 3.9C) Let $a \in \mathsf{B}$, $b \in \mathsf{B}^*$. Then
>
> $$(ab)(\varnothing) = b(\varnothing),$$
> $$(ab)(\mathrm{t}) = b(\varnothing)a(\mathrm{t}) + \sum_{\mathrm{t}' \leq \mathrm{t}} b(\mathrm{t}')a(\mathrm{t} \smallsetminus \mathrm{t}'), \quad \mathrm{t} \in \mathrm{T}. \tag{4.6 g}$$

Proof. Let p be a positive integer. By Theorem 3.8B (p. 130), there exist Runge–Kutta method tableaux M_1, M_2, with B-series coefficients, Ψ, Ω, respectively, such that $\Psi = a + \mathsf{O}_{p+1}$, $\Omega = b + \mathsf{O}_{p+1}$. For $M = M_1 M_2$, let Φ be the corresponding B-Series so that $\Phi = \Psi\Omega$. Use Theorem 4.6A so that (4.6 g) holds to order p. Since p is arbitrary, the result follows. □

4.7 The B-group and subgroups

Reinterpreting Theorem 4.6B in the case that $b \in \mathsf{B}$, we obtain a binary operation on this set given by

$$(ab)(\mathrm{t}) = a(\mathrm{t}) + \sum_{\mathrm{t}' \leq \mathrm{t}} b(\mathrm{t}')a(\mathrm{t} \smallsetminus \mathrm{t}'), \quad \mathrm{t} \in \mathrm{T}, \tag{4.7 a}$$

or, written another way, as

$$(ab)(\mathrm{t}) = \sum_{\varnothing \leq \mathrm{t}' \leq \mathrm{t}} b(\mathrm{t}')a(\mathrm{t} \smallsetminus \mathrm{t}'), \quad \mathrm{t} \in \mathrm{T}, \tag{4.7 b}$$

> Theorem 4.7A The set B equipped with the binary operation $(a,b) \mapsto ab$, given by (4.7 b), is a group.

Proof. We verify the three group axioms.
(i) B is associative because $(a(bc))(\mathrm{t})$ and $((ab)c)(\mathrm{t})$ are each equal to

$$\sum_{\varnothing \le \mathrm{t}'' \le \mathrm{t}' \le \mathrm{t}} a(\mathrm{t} \setminus \mathrm{t}')b(\mathrm{t}' \setminus \mathrm{t}'')c(\mathrm{t}'')$$

(ii) The identity element exists, given by

$$1(\varnothing) = 1, \qquad 1(\mathrm{t}) = 0, \quad \mathrm{t} \in \mathrm{T}$$

(iii) The inverse exists. For $a \in \mathsf{B}$, $a^{-1} = x$ is defined recursively by

$$x(\mathrm{t}) = -\sum_{\varnothing \le \mathrm{t}' < \mathrm{t}} x(\mathrm{t}')a(\mathrm{t} \setminus \mathrm{t}'), \quad \mathrm{t} \in \mathrm{T},$$

or

$$x(\mathrm{t}) = -\sum_{\varnothing < \mathrm{t}' \le \mathrm{t}} a(\mathrm{t}')x(\mathrm{t} \setminus \mathrm{t}'), \quad \mathrm{t} \in \mathrm{T}, \qquad \square$$

The "B-group" of Theorem 4.7A was introduced in [14] (Butcher, 1972) and named the "Butcher group" in [52] (Hairer, Wanner,1974).

Subgroups of B

If H is a subset of B then it is a "subgroup", written $H \le \mathsf{B}$, if $xy \in H$ whenever $x, y \in H$.

The order-defining subgroup

Definition 4.7B The subset O_{p+1} of B^0 is defined by $a \in \mathsf{O}_{p+1}$, if $a(\mathrm{t}) = 0$ for $\lvert\mathrm{t}\rvert \le p$.

Theorem 4.7C O_{p+1} is a linear subspace of B^*.

Proof. If $a, b \in \mathsf{O}_{p+1}$ then $a + b \in \mathsf{O}_{p+1}$, because $(a_b)(\mathrm{t}) = 0$ if $\lvert\mathrm{t}\rvert > p$. $\square$

The subgroup of B, $(1 + \mathsf{O}_{p+1})$, is of particular interest.

Theorem 4.7D $(1 + \mathsf{O}_{p+1})$ is a normal subgroup of B.

Proof. For convenience, write $H = (1 + \mathsf{O}_{p+1})$. This is a subgroup because if $h_1, h_2 \in H$, then $(h_1h_2)(\mathrm{t}) = h_1(\mathrm{t}) + h_2(\mathrm{t}) + Q$, where the quantity Q involves trees of order less than $\lvert\mathrm{t}\rvert$. If $x \in \mathsf{B}$, $h \in H$, so that $xh \in xH$, we construct $\overline{h}$ recursively through trees of increasing orders so that $\overline{h}x \in xh$. Up to order p it is seen that $\overline{h}$ and h agree and therefore $\overline{h} \in H$. □

> Definition 4.7E The quotient group $\mathsf{B}/(1+\mathsf{O}_{p+1})$ is defined to be B_p.

> Theorem 4.7F If $a \in b\mathsf{B}_p$ then
> $$B(a)_h y_0 = B(b)_h y_0 + \mathscr{O}(h^{p+1}).$$

Proof. If $|\mathrm{t}| \le p$, then $a(\mathrm{t}) = b(\mathrm{t})$. □

Hence, we can identify $\mathsf{B}/(1+\mathsf{O}_{p+1})$ with the group formed by restricting the mapping $T \to \mathbb{R}$ which defines elements of B to trees with order not exceeding p.

Informal explanation of O_{p+1} *and* $1+\mathsf{O}_{p+1}$

Use the example case $p = 4$. Members of O_5 and $1+\mathsf{O}_5$ are of the form below where the trees which index the various components are also shown

$$\begin{array}{rl} \mathsf{O}_5: & \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & a_9 & a_{10} & \cdots \end{bmatrix}^{\mathsf{T}} \\ 1+\mathsf{O}_5: & \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & a_9 & a_{10} & \cdots \end{bmatrix}^{\mathsf{T}} \end{array}$$

Linear combinations of members of O_5 are also members of this subspace of B^*; furthermore the product of two members of the group $1+\mathsf{O}_5$ is also in this group because $(ab)(\mathrm{t})$, for $|\mathrm{t}| \le 4$, is always zero.

If x and y are the coefficients in the B-series expansions corresponding to χ_h and y_h respectively, then $x = y + \mathsf{O}_5$ means that the terms up to order 4 are identical in x and y. This corresponds to $\chi_h = y_h + \mathscr{O}(h^5)$.

If $x, y \in \mathbb{B}$, so that χ_h and y_h are central mappings, then $x = y + \mathsf{O}_5$ can be written in the form $x = y(1+\mathsf{O}_5)$.

Numerical significance of special subgroups

Although algebraic analysis is widely used in the study of many types of numerical methods, we will confine these remarks to Runge–Kutta methods. If a subgroup H has the property
$$\mathrm{E} \in H \le \mathsf{B},$$
then it might be simpler to search for approximations in H than in B directly. A particular case is the group D_1 which contains all solutions to the conditions for order 4 within explicit methods with 4 stages.

Introduction to B_p

In this discussion, we will use the standard numbering of trees introduced in Table 7 (p. 66). If a and b are two specific points with

$$a_i := a(\mathbf{t}_i), \quad b_i := b(\mathbf{t}_i), \qquad i = 1, 2, \dots, 17,$$

then, as far as order 3 trees, we define the product ab by giving the values $(ab)_i := (ab)(\mathbf{t}_i)$, as follows

$$\begin{aligned}
(ab)_1 &= a_1 + b_1, \\
(ab)_2 &= a_2 + a_1 b_1 + b_2, \\
(ab)_3 &= a_3 + a_1^2 b_1 + 2a_1 b_2 + b_3, \\
(ab)_4 &= a_4 + a_2 b_1 + a_1 b_2 + b_4.
\end{aligned} \tag{4.7 c}$$

For $|\mathrm{t}| \geq 4$, the formula for the product evaluated for this tree has the form

$$\begin{aligned}
(ab)(\mathrm{t}) = {}& a(\mathrm{t}) + b(\mathrm{t}) \\
& + \text{an expression involving } a \text{ and } b \text{ for lower order trees.}
\end{aligned}$$

Hence we can illustrate some of the group-theoretic properties of B, introduced in Section 3.4, using the restriction to just the first four trees. We will temporarily write B for the set $\mathbb{R}^4$, with a binary operation defined by (4.7 c). We will also write a^{-1} for the right inverse satisfying $a \cdot a^{-1} = 1$, where 1 denotes the left or right identity element $1(\mathbf{t}_i) = 0$, $i = 1, 2, 3, 4$. It is found that

$$\begin{aligned}
(a^{-1})_1 &= -a_1, \\
(a^{-1})_2 &= a_1^2 - a_2, \\
(a^{-1})_3 &= -a_1^3 + 2a_1 a_2 - a_3, \\
(a^{-1})_4 &= -a_1^3 + 2a_1 a_2 - a_4.
\end{aligned}$$

Theorem 4.7G B_p is a group.

Proof. It is only necessary to prove that B_p is associative. Introduce a third member $c \in \mathsf{B}$. We have

$$\begin{aligned}
\big((ab)c\big)_1 &= a_1 + b_1 + c_1 \\
&= \big(a(bc)\big)_1, \\
\big((ab)c\big)_2 &= (a_2 + a_1 b_1 + b_2) + (a_1 + b_1)c_1 + c_2 \\
&= a_2 + a_1(b_1 + c_1) + (b_2 + b_1 c_1 + c_2) \\
&= \big(a(bc)\big)_2, \\
\big((ab)c\big)_3 &= (a_3 + a_1^2 b_1 + 2a_1 b_2 + b_3) \\
&\qquad + (a_1 + b_1)^2 c_1 + 2(a_1 + b_1)c_2 + c_3
\end{aligned}$$

$$
\begin{aligned}
&= a_3 + a_1^2(b_1+c_1) + 2a_1(b_2+b_1c_1+c_2)\\
&\qquad + (b_3+b_1^2c_1+2b_1c_2+c_3)\\
&= \big(a(bc)\big)_3,\\
\big((ab)c\big)_4 &= (a_3+a_2b_1+a_1b_2+b_3)\\
&\qquad + (a_2+a_1b_1+b_2)c_1 + (a_1+b_1)c_2 + c_3\\
&= a_3 + a_2(b_1+c_1) + a_1(b_2+b_1c_1+c_2)\\
&\qquad + (b_3+b_2c_1+b_1c_2+c_3)\\
&= \big(a(bc)\big)_4. \qquad \square
\end{aligned}
$$

Subgroups of B_p for low p

The group B_p has some interesting sub-groups.

- The subgroup H_1 with elements defined by $a_2 = \frac{1}{2}a_1^2$, $a_3 = \frac{1}{3}a_1^3$, $a_4 = \frac{1}{6}a_1^3$.
- The subgroup H_2 with elements satisfying $a_1 = 0$.
- The subgroup H_3 with elements defined by $a_2 = \frac{1}{2}a_1^2$, $a_4 = \frac{1}{2}a_3$.
- The subgroup H_4 with elements defined by $a_2 = \frac{1}{2}a_1^2$.

These are all low order examples of subgroups of B but illustrate some important features. The subgroup H_1 represents, to order 3, the group generated by flows through a time a_1h. The subgroup H_2 represents approximations to the identity map to order 1. The subgroup H_3 illustrates the use of the simplifying assumption $C(2)$ and H_4 represents methods with time-reversal symmetry.

Examples and exercises

In these examples we will write an element of B as $a = \begin{bmatrix} a_1 & a_2 & a_3 & a_4 \end{bmatrix}$. The set H_1 is a subgroup of B because

$$
\begin{aligned}
&\begin{bmatrix} a_1 & \frac{1}{2}a_1^2 & \frac{1}{3}a_1^3 & \frac{1}{6}a_1^4 \end{bmatrix}\begin{bmatrix} b_1 & \frac{1}{2}b_1^2 & \frac{1}{3}b_1^3 & \frac{1}{6}b_1^4 \end{bmatrix}\\
&\qquad\qquad = \begin{bmatrix} (a_1+b) & \frac{1}{2}(a_1+b)^2 & \frac{1}{3}(a_1+b)^3 & \frac{1}{6}(a_1+b)^4 \end{bmatrix},
\end{aligned}
$$

but it is not a normal subgroup because there does not exist an a_1 such that

$$
\begin{bmatrix} 1 & 0 & 0 & 0 \end{bmatrix}\begin{bmatrix} 1 & \frac{1}{2} & \frac{1}{3} & \frac{1}{6} \end{bmatrix} = \begin{bmatrix} a_1 & \frac{1}{2}a_1^2 & \frac{1}{3}a_1^3 & \frac{1}{6}a_1^4 \end{bmatrix}\begin{bmatrix} 1 & 0 & 0 & 0 \end{bmatrix},
$$

where it is noted that the first component gives $a_1 + 1 = 2$ whereas the third component gives $\frac{1}{2}a_1^2 + \frac{1}{6}a_1^3 = 2$.

The set H_2 is a subgroup because $(ab)_1 = a_1 + b_1 = 0$; it is normal because, for $x \in \mathsf{B}$ and $a \in H_2$, $b \in H_2$ satisfying $xa = bx$ is given by $b = \begin{bmatrix} 0 & a_2 & a_3 + 2a_2x_1 & a_4 \end{bmatrix}$.

Exercise 42 Show that H_3 is a subgroup of B but is not a normal subgroup.

Exercise 43 Show that H_4 is a normal subgroup of B.

The Q subgroups

Definition 4.7H Q_p is the set of members of B such that, if $a \in \mathsf{Q}_p$, then

$$a([\tau^{k-1}]) = a(\tau)^k/k, \qquad k \leq p.$$

Q is the set of members of B such that, if $a \in \mathsf{Q}$, then

$$a([\tau^{k-1}]) = \tfrac{1}{k} a(\tau)^k, \qquad k = 1, 2, \ldots$$

Theorem 4.7I For any p, $\mathsf{Q}_p \leq \mathsf{B}$. Also $\mathsf{Q} \leq \mathsf{B}$.

Proof. We have

$$\begin{aligned}
(ab)\big([\tau^{k-1}]\big) &= a\big([\tau^{k-1}]\big) + \sum_{\mathrm{x} \leq [\tau^{k-1}]} a\big([\tau^{k-1}] \setminus \mathrm{x}\big) b(\mathrm{x}) \\
&= \frac{1}{k} a(\tau)^k + a(\tau)^{k-1} b(\tau) + \sum_{i=1}^{k-1} \binom{k-1}{i} a(\tau)^{k-1-i} b([\tau^i]) \\
&= \frac{1}{k} a(\tau)^k + a(\tau)^{k-1} b(\tau) + \sum_{i=1}^{k-1} \binom{k-1}{i} \Big(\frac{1}{k-i} a(\tau)^{k-i} \Big) \\
&= \tfrac{1}{k} \big(a(\tau) + b(\tau) \big)^k \\
&= \tfrac{1}{k} \big((ab)(\tau) \big)^k.
\end{aligned}$$ □

Realization in Runge–Kutta methods

The elementary weights of a Runge–Kutta method (A, b^{T}, c) lie in Q_p, if

$$b^{\mathsf{T}} c^{k-1} = \tfrac{1}{k}, \qquad k = 1, 2, \ldots, p. \tag{4.7 d}$$

The C_q subgroups

Given a positive integer q, we will construct a refinement of Q_q such that it can be realized by a Runge–Kutta method in which (4.7 d) applies, not only to the output vector b^{T}, but also to each row $e_i^{\mathsf{T}} A$, in the form

$$\sum_{j=1}^{s} a_{ij} c_j^{k-1} = \tfrac{1}{k} c_i^k, \qquad k = 1, 2, \ldots, q, \quad i = 1, 2, \ldots, s.$$

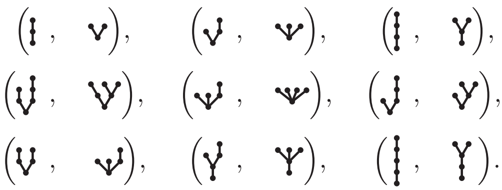
The case $k = 1$ will be recognised as the definition of c_i as the sum of the elements in row i of A.

Definition of S_k

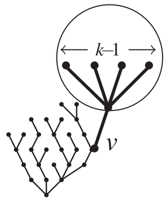
This aim is achieved by constructing sets of tree-tree pairs, S_k, $k = 2,3,\ldots,q$, and defining C_q in Definition 4.7J below. In constructing S_k, we recall that $\mathrm{t} * (k_{-1}\tau^k)$ has the value $k^{-1}\mathrm{t}(*\tau)^k$ and that $\mathrm{t} * (k^{-1}\mathrm{t}') := k^{-1}\mathrm{t} * \mathrm{t}'$. The following definition is used only in the current section

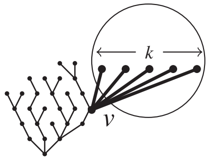
> Definition 4.7J The sets S_k, $k = 1,2,\ldots$ are defined by
>
> 1. $([[\tau^{k-1}]],[\tau^k]) \in S_k$,
> 2. $(\mathrm{t}'' * \mathrm{t}, \mathrm{t}'' * \mathrm{t}') \in S_k$ if $(\mathrm{t},\mathrm{t}') \in S_k, \mathrm{t}'' \in \mathrm{T}$.

Some examples in the case $k = 2$ are



The diagram (4.7 e) shows a generic example of trees t and t′ such that $(\mathrm{t},\mathrm{t}') \in S_k$.

$\mathrm{t} =$ 

$\mathrm{t}' =$ 

(4.7 e)

The discs surrounding the subtrees $[\tau^{k-1}]$ in t and $[\tau^k]$ in t′ are to identify specific vertices which will be referred to in the proof of Theorem 4.7L.

> Definition 4.7K C_q is the set of members of B such that, for $k = 1,2,\ldots,q$,
>
> $$a([\tau^{k-1}]) = k^{-1}a(\tau)^k, \tag{4.7 f}$$
> $$a(\mathrm{t}) = k^{-1}a(\mathrm{t}') \quad \text{if } (\mathrm{t},\mathrm{t}') \in S_k. \tag{4.7 g}$$

> Theorem 4.7L
>
> $$\mathsf{C}_q \le \mathsf{B}, \qquad q = 1,2,\ldots$$

Proof. Assume $a,b \in \mathsf{C}_k$. To show that if a,b satisfy (4.7 f), then the same is true for ab, use the argument in the proof of Theorem 4.7I. To show that (4.7 g) holds with a replaced by ab, let t and t$'$ be as in diagram (4.7 e), and evaluate $(ab)(\mathrm{t})$, $(ab)(\mathrm{t}')$ using

$$(ab)(\mathrm{t}) = a(\mathrm{t}) + \sum_{\mathrm{t}'' \le \mathrm{t}} a(\mathrm{t} \setminus \mathrm{t}'')b(\mathrm{t}''), \tag{4.7 h}$$

$$(ab)(\mathrm{t}') = a(\mathrm{t}') + \sum_{\mathrm{t}''' \le \mathrm{t}'} a(\mathrm{t}' \setminus \mathrm{t}''')b(\mathrm{t}'''). \tag{4.7 i}$$

Consider three cases:
(1) t$''$ and t$'''$ each contains the vertex v as well as the k descendants of v. Apart from the circled vertices, t$''$ and t$'''$ are identical.
(2) t$''$ and t$'''$ do not contain any of the k descendants of v and hence $\mathrm{t}'' = \mathrm{t}'''$.
(3) t$''$ and t$'''$ each contain ℓ of the k descendants of v, where $1 \le \ell \le k-1$. Apart from the circled vertices, t$''$ and t$'''$ are identical.
In case (1), t$''$ and t$'''$ have the same form as t and t$'$ in diagram (4.7 e), respectively, and hence, $b(\mathrm{t}) = k^{-1}b(\mathrm{t}')$. Furthermore, $\mathrm{t} \setminus \mathrm{t}' = \mathrm{t}' \setminus \mathrm{t}'''$. Hence, in this case, $a(\mathrm{t} \setminus \mathrm{t}'')b(\mathrm{t}'') = k^{-1}a(\mathrm{t}' \setminus \mathrm{t}''')b(\mathrm{t}''')$.
In case (2), $\mathrm{t}'' = \mathrm{t}''$. Furthermore, $\mathrm{t} \setminus \mathrm{t}''$ and $\mathrm{t}' \setminus \mathrm{t}'''$ have the same forms as in the two parts of (4.7 e). Hence, also in this case, $a(\mathrm{t} \setminus \mathrm{t}'')b(\mathrm{t}'') = k^{-1}a(\mathrm{t}' \setminus \mathrm{t}''')b(\mathrm{t}''')$.
In case (3), the terms in (4.7 h) and (4.7 i) need to have additional factors n and n' respectively inserted, to allow for replications of choices of $\ell - 1$ from $k-1$ (ℓ from k, respectively) vertices. That is,

$$n = \binom{k-1}{\ell-1}, \quad n' = \binom{k}{\ell} = \tfrac{k}{\ell}n.$$

We now have

$$na(\mathrm{t} \setminus \mathrm{t}'')b(\mathrm{t}'') = n'\tfrac{\ell}{k}a(\mathrm{t}' \setminus \mathrm{t}''')\ell^{-1}b(\mathrm{t}''') = k^{-1}n'a(\mathrm{t}' \setminus \mathrm{t}''')b(\mathrm{t}'''). \qquad \square$$

Realization in Runge–Kutta methods

A Runge–Kutta method has "stage-order q", written as C(q), if $Ac^{k-1} = \frac{1}{k}c^k$, $k = 1,2,\dots,q$ and $bc^{k-1} = \frac{1}{k}$. That is,

$$\left.\begin{aligned} \textstyle\sum_{j=1}^{s} a_{ij}c_j^{k-1} &= \tfrac{1}{k}c_i^k, \quad i = 1,2,\dots,s, \\ \textstyle\sum_{i=1}^{s} b_i c_i^{k-1} &= \tfrac{1}{k}, \end{aligned}\right\} \quad k = 1,2,\dots,q.$$

For C(q) methods, the corresponding B-series lie in C_q. All collocation methods, such as Gauss methods, satisfy C(s). For $q > 1$, C(q) is impossible for explicit Runge–Kutta methods but it is possible if some limited form of implicitness is allowed, such as for the fourth order method

$$\begin{array}{c|ccc} 0 & & & \\ \frac{1}{2} & \frac{1}{4} & \frac{1}{4} & \\ 1 & 0 & 1 & \\ \hline & \frac{1}{6} & \frac{2}{3} & \frac{1}{6} \end{array}.$$

Explicit methods with order greater than 4 have stage order 2 or higher, except that this does not apply to the second stage. It is necessary to compensate for this anomaly by imposing additional conditions, such as $b_2 = \sum_i b_i a_{i2} = \sum_i b_i c_i a_{i2} = \sum b_i a_{ij} a_{j2} = 0$.

The D subgroups

These subgroups have several important roles in numerical analysis. First, the subgroup D_1 has a historic connection with the work of Kutta [66]. Every fourth order Runge–Kutta method with four stages is a member of D_1. Although this is not generally true for higher order explicit methods, it is standard practice, in searching for such methods, to consider only methods belonging this subgroup. Thus it plays the role of a "simplfying assumption". Secondly, D_s contains the s-stage Gauss method. Finally, D defines canonical Runge–Kutta, which play a central role in the solution of Hamiltonian problems, as we will see in Chapter 7.

Definition 4.7M $\mathsf{D}_p \subset \mathsf{B}$, $\mathsf{D}_{pq} \subset \mathsf{B}$ and $\mathsf{D} \subset \mathsf{B}$ are defined by

$$\begin{array}{lll} a \in \mathsf{D}_p \text{ if} & a(\mathrm{t}*\mathrm{t}') + a(\mathrm{t}'*\mathrm{t}) = a(\mathrm{t})a(\mathrm{t}'), \quad \mathrm{t},\mathrm{t}' \in T, & |\mathrm{t}| \le p, \\ a \in \mathsf{D}_{pq} \text{ if} & a(\mathrm{t}*\mathrm{t}') + a(\mathrm{t}'*\mathrm{t}) = a(\mathrm{t})a(\mathrm{t}'), \quad \mathrm{t},\mathrm{t}' \in T, & |\mathrm{t}| \le p, \ |\mathrm{t}'| \le q, \\ a \in \mathsf{D} \text{ if} & a(\mathrm{t}*\mathrm{t}') + a(\mathrm{t}'*\mathrm{t}) = a(\mathrm{t})a(\mathrm{t}'), \quad \mathrm{t},\mathrm{t}' \in T. & \end{array}$$

Theorem 4.7N Each of D_{pq}, D_p and D is a subgroup of B.

Proof. It will be sufficient to prove the result in the case of D_{pq}. Assume that $a,b \in \mathsf{D}_{pq}$, and evaluate $(ab)(\mathrm{t}*\mathrm{t}')$:

$$\begin{aligned}(ab)&(\mathrm{t}*\mathrm{t}') \\ &= a(\mathrm{t}*\mathrm{t}') + \sum_{\mathrm{x}\le\mathrm{t},\mathrm{x}'\le\mathrm{t}'} a(\mathrm{t}\setminus\mathrm{x})a(\mathrm{t}'\setminus\mathrm{x}')b(\mathrm{x}*\mathrm{x}') + \sum_{\mathrm{x}\le\mathrm{t}} a(\mathrm{t}\setminus\mathrm{x})a(\mathrm{t}')b(\mathrm{x}).\end{aligned}$$

Add the corresponding expression, with t and t′ interchanged, and we find

$$\begin{aligned}(ab)(\mathrm{t}*\mathrm{t}') + (ab)(\mathrm{t}'*\mathrm{t}) &= a(\mathrm{t})a(\mathrm{t}') + \sum_{\mathrm{x}\le\mathrm{t},\mathrm{x}'\le\mathrm{t}'} a(\mathrm{t}\setminus\mathrm{x})a(\mathrm{t}'\setminus\mathrm{x}')b(\mathrm{x})b(\mathrm{x}') \\ &\quad + \sum_{\mathrm{x}\le\mathrm{t}} a(\mathrm{t}\setminus\mathrm{x})a(\mathrm{t}')b(\mathrm{x}) + \sum_{\mathrm{x}'\le\mathrm{t}'} a(\mathrm{t}'\setminus\mathrm{x}')a(\mathrm{t})b(\mathrm{x}') \\ &= \Big(a(\mathrm{t}) + \sum_{\mathrm{x}\le\mathrm{t}} a(\mathrm{t}\setminus\mathrm{x})b(\mathrm{x})\Big)\Big(a(\mathrm{t}') + \sum_{\mathrm{x}'\le\mathrm{t}'} a(\mathrm{t}'\setminus\mathrm{x}')b(\mathrm{x}')\Big) \\ &= (ab)(\mathrm{t})(ab)(\mathrm{t}'). \qquad \square\end{aligned}$$

One of the aims of this chapter is to generalize these constructions to the full set of trees on which B is defined. A second aim is to interrelate B with the generalizations of Runge–Kutta methods introduced in [14] as "integration methods" in which the usual A in (A,b,c) is replaced by a linear operator and b is replaced by a linear functional in a space of functions on a possibly infinite index set. For example, the index set $\{1,2,\dots,s\}$ could be replaced by the interval $[0,1]$. The integration methods would thus include not only Runge–Kutta methods, but also the Picard construction

$$\begin{aligned} Y(x_0+h\xi) &= y(x_0)+h\int_0^{\xi} f\big(Y(x_0+h\xi)\big)\,\mathrm{d}\xi,\\ y_1 &= y(x_0)+h\int_0^{1} f\big(Y(x_0+h\xi)\big)\,\mathrm{d}\xi, \end{aligned}$$

where we see that A and b^{T} correspond respectively to the operations

$$\begin{aligned} \varphi &\mapsto \int_0 \varphi,\\ \varphi &\mapsto \int_0^1 \varphi. \end{aligned}$$

Discrete gradient methods also fall into the definition of integration methods.

4.8 Linear operators on B^* and B^0

B and its subspaces recalled

In B-series analysis, mappings defined in terms of the triple (y_0,f,h) are represented by $(\mathbf{B}_h y_0)a$ for $a\in\mathsf{B}^*$. The affine subspace B has a special role as the counterpart to central mappings which are within $\mathscr{O}(h)$ of id_h. The members of B act as multipliers operating on the linear space B^* and are typified by Runge–Kutta mappings. The linear subspace B_0^* corresponds to the space spanned by $\mathit{slope}_h\circ\mathcal{C}_h$, where $\mathcal{C}_h$ is a central mapping. This means that B_0^* is the span of BD.

An extended set of linear operators on B_0^*

In Chapter 2, Section 2.7 (p. 79), the set S of uni-valent stumps was introduced. In the case of $\tau_1\in S$, power-series $\phi(\tau_1)=a_0 1+a_1\tau_1+a_2\tau_1^2+\cdots$ were introduced. We will consider B-series ramifications of these expressions.

Motivation

By introducing linear operators, such as $hf'(y_0)$, into the computation, a Runge–Kutta method can be converted to a Rosenbrock method or some other generalization. For example, the method

$$Y_1 = y_0, \qquad F_1 = f(Y_1), \tag{4.8 a}$$
$$Y_2 = y_0+a_{21}hF_1, \qquad F_2 = f(Y_1), \tag{4.8 b}$$
$$L = hf'(y_0+g_1hF_1+g_2hF_2), \tag{4.8 c}$$

$$y_1 = y_0 + b_1 hF_1 + d_1 hLF_1 + b_2 hF_2 + d_2 hLF_2, \tag{4.8 d}$$

contains additional flexibility compared with a Runge–Kutta method. If (4.8 d) is replaced by

$$y_1 = y_0 + b_1 hF_1 + d_1 h\phi(L)F_1 + b_2 hF_2 + d_2 h\phi(L)F_2,$$

numerical properties can be enhanced. We aim to include the use of L within the B-series formulation.

The operator J

Corresponding to $hf'(y_0)$, we introduce the linear function $J : \mathsf{B}^0 \to \mathsf{B}^0$, satisfying $\mathbf{B}_h Jb = hf'(y_0)\mathbf{B}_h b$, where $b(\varnothing) = 0$. By evaluating term by term, we see that $(Jb)([\mathrm{t}]) = b(\mathrm{t})$ for $\mathrm{t} \in \mathrm{T}$, with $(Jb)(\mathrm{t}') = 0$ if t' cannot be written as $\mathrm{t}' = [\mathrm{t}]$. We also need to find the B-series for operations of the form $hf'(y_1) = hf'\big((\mathbf{B}_h a)y_0\big)$ so that we can handle expressions such as L in (4.8 c).

Theorem 4.8A

$$hf'\big((\mathbf{B}_h a)y_0\big)(\mathbf{B}_h b)y_0 = BB_h\big((aJ)(a^{-1}b)\big)y_0. \tag{4.8 e}$$

Proof. Let $y_1 = (\mathbf{B}_h a)y_0$, $b = a\widetilde{b}$ so that (4.8 e) becomes

$$hf'\big(y_1\big)(\mathbf{B}_h\widetilde{b})y_1 = BB_h\big((J)(\widetilde{b})\big)y_1.$$

which is (4.8 e) after the substitutions $y_0 \mapsto y_1$, $b \mapsto \widetilde{b}$. □

Summary of Chapter 4 and the way forward

Integration methods were introduced as a generalization of Runge–Kutta methods in which the index set $I = \{1, 2, \ldots, s\}$ is replaced by a more complicated alternative. Equivalence and reducibility of methods, with an emphasis on the Runge–Kutta case, were considered. Compositions of methods were introduced leading to the composition theorem for integration methods. A number of subgroups of B were introduced, many of which have a relationship with simplifying assumptions for Runge–Kutta methods.

The way forward

Subgroups of B are used in the construction of the working numerical methods considered in Chapters 5 and 6. Continuous Runge–Kutta methods have, as a natural application, the energy-preserving methods of Chapter 7.

Teaching and study notes

Possible supplementary reading includes

Butcher, J.C. *An algebraic theory of integration methods* (1972) [14]

Butcher, J.C. *Numerical Methods for Ordinary Differential Equations* (2016) [20]

Hairer, E., Nørsett, S.P. and Wanner, G. *Solving Ordinary Differential Equations I: Nonstiff Problems* (1993) [50]

Hairer, E. and Wanner, G. *Multistep-multistage-multiderivative methods for ordinary differential equations* (1973) [51]

Hairer, E. and Wanner, G. *On the Butcher group and general multi-value method* (1974) [52]

Projects

Project 12 Develop the topic of reducibility in Section 4.4 further so that pre-reduced methods become fully reduced by eliminating unnecessary stages.

Project 13 Investigate the conditions for order 4 for the method given by (4.8 a) – (4.8 d),