Eur. Phys. J. C 12, 521–534 (2000)
Digital Object Identifier (DOI) 10.1007/s100529900235

THE EUROPEAN
PHYSICAL JOURNAL C

Runge–Kutta methods and renormalization

Ch. Brouder

Laboratoire de Minéralogie-Cristallographie, CNRS UMR7590, Universités Paris 6, Paris 7, IPGP, Case 115, 4 place Jussieu, 75252 Paris Cedex 05, France (e-mail: brouder@lmcp.jussieu.fr)

Received: 6 July 1999 / Published online: 10 December 1999

Abstract. Rooted trees have been used to calculate the solution of nonlinear flow equations and Runge–Kutta methods. More recently, rooted trees have helped systematizing the algebra underlying renormalization in quantum field theories. The Butcher group and B-series establish a link between these two approaches to rooted trees. On the one hand, this link allows for an alternative representation of the algebra of renormalization, leading to nonperturbative results. On the other hand, it helps to renormalize singular flow equations. The usual approach is extended here to nonlinear partial differential equations. A nonlinear Born expansion is given, and renormalization is used to partly remove the secular terms of the perturbative expansion.

1 Introduction

The purpose of this paper is to point out a link between two apparently remote concepts: Runge–Kutta methods and renormalization.

Runge published in 1895 [1] an efficient algorithm to compute the solution of ordinary differential equations. For an equation of the type $\mathrm{d}y/\mathrm{d}s = f(y(s))$, he defined $k_1 = f(y_n)$, $k_2 = f(y_n + hk_1/2)$, $y_{n+1} = y_n + hk_2$. His algorithm was improved in 1901 by Kutta [2] and became known as the Runge–Kutta method. It is now one of the most widely used numerical methods.

In 1972, Butcher published an extraordinary article in which he analyzed general Runge–Kutta methods on the basis of the algebra of rooted trees (ART). He showed that the Runge–Kutta methods form a group[1] and found explicit expressions for the inverse of a method or the product of two methods. He also gave an explicit perturbative solution of nonlinear differential equations, written as a series indexed by rooted trees (now called B-series). Important developments were made in 1974 by Hairer and Wanner [3]. Since then, B-series are used in the analysis of Runge–Kutta methods.

On the other hand, renormalization enables us to remove infinities in quantum field theory. Recently, Kreimer discovered a Hopf algebra of rooted trees that brings order and beauty to the intricate combinatorics of renormalization [4]. He established formulas that automate the subtraction of infinities to all orders of the perturbation expansion and proved the effectiveness of his method for the practical computation of renormalized quantities in joint works with Broadhurst [5] and Delbourgo [6]. Moreover, his approach sheds new light on the mechanics of the renormalization group [7], and on the problem of overlapping divergences [8,9] by showing that overlapping divergent diagrams correspond to a sum of decorated rooted trees, while nonoverlapping ones correspond to a single tree. Furthermore, Connes and Kreimer revealed a deep connection between the ART and the Hopf algebra of diffeomorphisms [10].

In 1986, Dür ([11], p. 88–90), studied Butcher's group and discovered a Hopf algebra and a Lie algebra of rooted trees which are identical (up to order) to those described by Connes and Kreimer in [10]. This establishes a clear connection between Butcher's group and renormalization. The purpose of this paper is to further explore this connection. First, some concepts developed by Kreimer using Hopf algebra will be translated into Butcher's language. This will enable us to give an alternative derivation of some of Kreimer's results, and to prove a "remarkable property" that was conjectured in [5]. Second, Butcher's approach will be applied to continuous Runge–Kutta methods, and it will be expanded to give a B-series for the solution of nonlinear partial differential equations. This B-series is a nonlinear analog of the Born series of linear differential equations.

Since Butcher's theory is not a common tool of physics, this paper will be reasonably self-contained and, it is hoped, pedagogical. After an introduction to rooted trees, the relation between differentials and rooted trees is presented. Then Butcher's approach to Runge–Kutta methods is sketched. Several B-series are calculated, and a connection with the Hopf structure of the ART is exhibited. The application of Runge–Kutta methods to renormalization is expounded by use of a toy model of field theory (not of renormalization) which is solved nonperturbatively. Finally, the solution of nonlinear partial differential equations is written as a formal B-series.

[1] Hairer and Wanner called it the Butcher group [3].

2 The rooted trees

In this section, the rooted trees are introduced and useful functions on rooted trees are defined.

A rooted tree is a graph with a designated vertex, called a root, such that there is a unique path from the root to any other vertex in the tree [12]. Several examples of rooted trees are given in the next section in which the root is the black point and the other vertices are white points. We follow the convention of nature and put the root at the bottom of the tree. The number of edges of the unique path from a vertex v to the root is called the level number of vertex v. The root has level number 0. For any vertex v, the children of v are the vertices v' with an edge common with v and a level number greater than that of v. A vertex without children is called a leaf. Rooted trees are sometimes called pointed trees or arborescences.

The tree with one vertex is designated as $\bullet$, and the tree with zero vertices as 1.

2.1 Operations and functions on trees

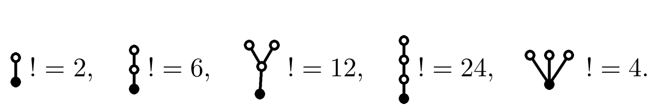
An important operation is the grafting of trees. If t_1, ... ,t_k are trees, $t = B_+(t_1\, t_2\, \ldots t_k)$ is defined as the tree obtained by the creation of a new vertex r and the joining of the roots of t_1, ... ,t_k to r, which becomes the root of t. In rooted trees, the branches may be permuted; for example, $B_+(t_1 t_2)$ and $B_+(t_2 t_1)$ represent the same tree.

A number of functions on rooted trees will be used in the following. We define first $|t|$, which designates the number of vertices of a tree t. Clearly, $|B_+(t_1\, t_2\, \ldots t_k)| = |t_1| + |t_2| + \cdots + |t_k| + 1$.

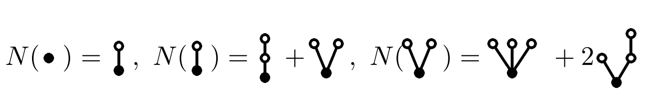
The tree factorial $t!$ is a natural number defined recursively as

$$\bullet\,! = 1,$$
$$B_+(t_1\, t_2\, \ldots t_k)! = |B_+(t_1\, t_2\, \ldots t_k)|\, t_1! t_2! \cdots t_k!.$$

The notation $t!$ is taken from Kreimer [7] because $t!$ generalizes the factorial of a number. Besides, $t!$ has also similarities with the product of hooklengthes of a Young diagram in the representation theory of the symmetric group [13]. A few examples may be useful:



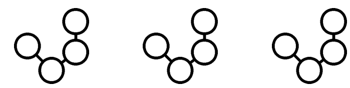
In [10], Connes and Kreimer define a natural growth operator N on trees: $N(t)$ is the set of $|t|$ trees t_i, where each t_i is a tree with $|t|+1$ vertices obtained by attaching an additional leaf to a vertex of t. For example, $N(1) = \bullet$:



Some trees may appear with multiplicity.

2.2 The tree multiplicity $\alpha(t)$

A central function over trees is the tree multiplicity $\alpha(t)$, which is defined in [7] as the number of times tree t appears in $N^n(1)$ where $n = |t|$ is the number of vertices of t. In the literature ([14,15], p. 92, [16], p. 147), $\alpha(t)$ is considered as the number of "heap-ordered trees" with shape t, where a heap-ordered tree with shape t is a labeling of each vertex of t (i.e., a bijection between the vertices and the set of numbers $0, 1, \ldots, |t| - 1$) such that the labels decrease along the path going from any vertex to the root. This is called a monotonic labeling in [16], p. 147. For instance,



There are $(n-1)!$ heap-ordered trees with n vertices, summing over trees with all possible shapes. This can be seen by the defining of a bijection between the permutations of $n-1$ numbers and the heap-ordered trees. Let $(p_1, \ldots, p_{n-1})$ be a permutation of $(1, \ldots, n-1)$; then

- p_1 is a subroot, labeled p_1
- for i=2 to $n-1$
  - if all p_j for $1 \le j \le i$ are such that $p_j > p_i$, then p_i is a subroot, labeled p_i
  - otherwise, let p_j be first number such that $p_j < p_i$, in the series $p_{i-1}, p_{i-2}, \ldots, p_1$, then the ith vertex, labeled p_i, is linked to p_j by an edge
- when all $(p_1, \ldots, p_{n-1})$ have been processed, all subroots are linked to a common root, labeled 0.

Clearly, this procedure generates heap-ordered trees.

For a given tree t, it is possible to calculate $\alpha(t)$ by a formula given in [14] (see also. [15], p. 92 and [7]):

$$\alpha(t) = \frac{|t|!}{t!\sigma(t)}, \tag{1}$$

where $\sigma(t)$ is the symmetry factor of t, defined as ([15] p. 89)

$$\sigma(\bullet) = 1,$$
$$\sigma(B_+(t_1^{n_1} \ldots t_k^{n_k})) = n_1!\sigma(t_1)^{n_1} \ldots n_k!\sigma(t_k)^{n_k}. \tag{2}$$

The notation $t = B_+(t_1^{n_1} \ldots t_k^{n_k})$ means that t is obtained by grafting n_1 times tree t_1, ... , n_k times tree t_k, where the k trees t_1, ... , t_k are all different.

These functions over trees have been defined independently by several authors and various notations exist in the literature. $t = B_+(t_1\, t_2\, \ldots t_k)$ is also denoted by $t = [t_1, t_2, \ldots, t_k]$, but we avoid this notation because of the possible confusion with commutators. The alternative notations for $|t|$ are $r(t)$, $\rho(t)$, and $\#t$. The tree factorial $t!$ is denoted $\gamma(t)$ in numerical analysis. $\alpha(t)$ is also written $CM(t)$. The symmetry factor $\sigma(t)$ is also called S_t.

Finally, we use the term algebra of rooted trees and not Hopf algebra of rooted trees, because thanks to the work of Butcher, the Hopf structure is only one aspect of the ART.

3 Differentials and rooted trees

Here, rooted trees are associated to differentials of a function, and the solution of a general nonlinear flow equation is given as a sum indexed by rooted trees. Assume that we want to solve the flow equation $(\mathrm{d}/\mathrm{d}s)x(s) = F[x(s)]$, $x(s_0) = x_0$, where s is a real, x is in $\mathbb{R}^N$, and F is a smooth function from $\mathbb{R}^N$ to $\mathbb{R}^N$, with components $f^i(x)$. This is the equation of the flow of a vector field.

3.1 Calculation of the nth derivative

Let us write the derivatives of the ith component of $x(s)$ with respect to s:

$$\begin{aligned}
\frac{\mathrm{d}^2x^i(s)}{\mathrm{d}s^2} &= \frac{\mathrm{d}}{\mathrm{d}s}f^i[x(s)] = \sum_j \frac{\partial f^i}{\partial x_j}[x(s)]\frac{\mathrm{d}x^j}{\mathrm{d}s} \\
&= \sum_j \frac{\partial f^i}{\partial x_j}[x(s)]f^j[x(s)], \\
\frac{\mathrm{d}^3x^i(s)}{\mathrm{d}s^3} &= \frac{\mathrm{d}}{\mathrm{d}s}\left(\sum_j \frac{\partial f^i}{\partial x_j}[x(s)]f^j[x(s)]\right) \\
&= \sum_{jk} \frac{\partial^2 f^i}{\partial x_j \partial x_k}[x(s)]f^j[x(s)]f^k[x(s)] + \\
&\quad \sum_{jk} \frac{\partial f^i}{\partial x_j}[x(s)]\frac{\partial f^j}{\partial x_k}[x(s)]f^k[x(s)].
\end{aligned}$$

A simplified notation is now required. Let

$$\begin{aligned}
f^i &= f^i[x(s)], \\
f^i_{j_1j_2\cdots j_k} &= \frac{\partial^k f^i}{\partial x_{j_1}\cdots\partial x_{j_k}}[x(s)],
\end{aligned}$$

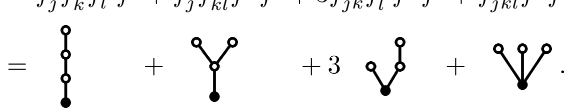
so that

$$\begin{aligned}
\frac{\mathrm{d}x^i(s)}{\mathrm{d}s} &= f^i, \quad \frac{\mathrm{d}^2x^i(s)}{\mathrm{d}s^2} = f^i_j f^j, \\
\frac{\mathrm{d}^3x^i(s)}{\mathrm{d}s^3} &= f^i_{jk}f^jf^k + f^i_jf^j_kf^k,
\end{aligned}$$

where summation over identical indices appearing in lower and upper positions is implicitly assumed.

With this notation, we can write the next term as

$$\frac{\mathrm{d}^4x^i}{\mathrm{d}s^4} = f^i_jf^j_kf^k_lf^l + f^i_jf^j_{kl}f^kf^l + 3f^i_{jk}f^j_lf^kf^l + f^i_{jkl}f^jf^kf^l$$



In the last line of this equation, a rooted tree was associated to each term of the sum. This relation between differentials and rooted tree was established by Arthur Cayley in 1857 [17]. With this notation, there is a one-to-one relation between a rooted tree with n vertices and a term of $\mathrm{d}^nx(s)/\mathrm{d}s^n$.

3.2 Elementary differentials

To make more precise the relation between differentials and rooted trees, we follow Butcher ([15] p. 154) and call "elementary differentials" the δ_t (with components δ^i_t) defined recursively for each rooted tree t by:

$$\begin{aligned}
\delta^i_\bullet &= f^i, \\
\delta^i_t &= f^i_{j_1j_2\cdots j_k}\delta^{j_1}_{t_1}\delta^{j_2}_{t_2}\cdots\delta^{j_k}_{t_k},
\end{aligned} \tag{3}$$

when $t = B_+(t_1\,t_2\,\cdots t_k)$.

Using this correspondence between rooted trees and differentiation of expressions, we see that differentiating $\delta_t(s)$ with respect to s adds a leaf to each vertex of t. Therefore,

$$\frac{\mathrm{d}\delta_t}{\mathrm{d}s} = N\delta_t,$$

where N is the natural growth operator introduced in Sect. 2.1. More precisely, if $N(t) = \sum_i m_it_i$, then $N\delta_t$ is defined as $\sum_i m_i\delta_{t_i}$.

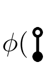
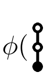
Hence, the solution of the flow equation is

$$\begin{aligned}
x(s) &= x_0 + \sum_{n=1}^{\infty}\frac{(s-s_0)^n}{n!}\frac{\mathrm{d}^nx}{\mathrm{d}s^n}(s_0) \\
&= x_0 + \sum_{n=1}^{\infty}\frac{(s-s_0)^n}{n!}N^{n-1}\delta_\bullet(s_0) \\
&= x_0 + \sum_t \frac{(s-s_0)^{|t|}}{|t|!}\alpha(t)\delta_t(s_0),
\end{aligned} \tag{4}$$

where $|t|$ and $\alpha(t)$ have been defined in Sect. 2. The argument s_0 of $\delta_t(s_0)$ in (4) means that the derivatives are calculated at the point $x_0 = x(s_0)$. Thus, it would also be possible to write $\delta_t(x_0)$.

4 Runge–Kutta methods

In this section, the usual discrete Runge–Kutta methods are introduced, and the operations of Butcher's group are defined. We have seen that sums over trees appear quite naturally with differential equations. So, if one is given a map ϕ that assigns a value (e.g., a real number, a complex number, a vector) to each tree t, is there a function F such that $\phi(t) = \delta_t$? Generally, the answer is no. Consider a map ϕ such that all components are equal (and also denoted ϕ):

$\phi(\bullet) = 1$, $\phi($  $) = a$, $\phi($  $) = b$,

so that for any i, $f^i = 1$, $f^i_jf^j = a$ and $f^i_jf^j_kf^k = b$. The first two equations give $\sum_j f^i_j = a$, so the third one gives $f^i_jf^j_kf^k = \sum_j f^i_ja = a^2$, and $\phi(t)$ cannot be represented with elementary differentials (i.e., it cannot be the δ_t) of a function F if $b \neq a^2$. In fact, the number of maps reachable as elementary differentials is rather small.

To define the algebra of rooted trees, we need a product of trees. The product of t and t' is written tt' (the product of two trees is not a tree, but a simple juxtaposition of trees; one might think of a forest). This product is assumed commutative. In the algebra, a tree t can be multiplied by a scalar λ (written λt), and two trees t and t' can be added (giving $t+t'$). We assume that the product is distributive with respect to the sum. For instance, $t(\lambda_1 t_1 + \lambda_2 t_2) = \lambda_1 t t_1 + \lambda_2 t t_2$.

Given a map ϕ defined over the rooted trees, we can extend it to a homomorphism of the algebra of rooted trees by defining $\phi(\lambda_1 t_1 + \lambda_2 t_2) = \lambda_1\phi(t_1) + \lambda_2\phi(t_2)$ and $\phi(tt') = \phi(t)\phi(t')$ where the componentwise product was used on the right-hand side.

If vector flows (i.e., δ_t) are not enough to span all possible ϕ, what more general equation can do that? As we shall see now, the answer is the Runge–Kutta methods [18].

4.1 Butcher's approach to the Runge–Kutta methods

Some efficient numerical algorithms for solving a flow equation $\mathrm{d}x(s)/\mathrm{d}s = F[x(s)]$ are known as Runge–Kutta methods. They are determined by an $m \times m$ matrix a and an m-dimensional vector b, and at each step, a vector x_n is defined as a function of the previous value x_{n-1} by:

$$X_i = x_{n-1} + h\sum_{j=1}^{m} a_{ij}F(X_j),$$
$$x_n = x_{n-1} + h\sum_{j=1}^{m} b_j F(X_j),$$

where i ranges from 1 to m. If the matrix a is such that $a_{ij} = 0$ if $j \geq i$, then the method is called explicit (because each X_i can be calculated explicitly); otherwise the method is implicit.

In 1963, Butcher showed that the solution of the corresponding equations,

$$X_i(s) = x_0 + (s-s_0)\sum_{j=1}^{m} a_{ij}F(X_j(s)),$$
$$x(s) = x_0 + (s-s_0)\sum_{j=1}^{m} b_j F(X_j(s)),$$

is given by

$$X_i(s) = x_0 + \sum_t \frac{(s-s_0)^{|t|}}{|t|!}\alpha(t)t!\sum_{j=1}^{m} a_{ij}\phi_j(t)\delta_t(s_0),$$
$$x(s) = x_0 + \sum_t \frac{(s-s_0)^{|t|}}{|t|!}\alpha(t)t!\phi(t)\delta_t(s_0). \quad (5)$$

These series over trees are called B-series in the numerical analysis literature, in honor of John Butcher ([16], p. 264). The homomorphism ϕ is defined recursively as a function of a and b, for $i = 1,\ldots,m$:

$$\phi_i(\bullet) = 1,$$
$$\phi_i(B_+(t_1\cdots t_k)) = \sum_{j_1\ldots j_k} a_{ij_1}\ldots a_{ij_k}\phi_{j_1}(t_1)\ldots\phi_{j_k}(t_k),$$
$$\phi(t) = \sum_{i=1}^{m} b_i\phi_i(t).$$

When one compares (4) and (5), it is clear that the Runge–Kutta solution approximates the solution of the original flow equation up to order n if $\phi(t) = 1/t!$ for all trees with up to n vertices.

In 1972 [18], Butcher made further progress. First, he showed that Runge–Kutta methods are "dense" in the space of rooted tree homomorphisms. More precisely, he showed that given any finite set of trees T_0 and any function θ from T_0 to $\mathbb{R}$, there is a Runge–Kutta method (i.e., a matrix a and a vector b) such that the corresponding ϕ agrees with θ on T_0 (see also [15] p. 167).

4.2 Further developments

Furthermore, Butcher proved that the combinatorics he used to study Runge–Kutta methods in 1963 [14] was hiding an algebra. If (a,b) and (a',b') are two Runge–Kutta methods (having dimensions m and m', respectively) with the corresponding homomorphisms ϕ and ϕ', then the product of these Runge–Kutta methods is the (a'',b'') of dimension $m+m'$, defined by ([15], p. 312 et seq.)

$$\begin{aligned}
a''_{ij} &= a_{ij} \quad \text{if} \quad 1\leq i\leq m \quad \text{and} \quad 1\leq j\leq m,\\
a''_{ij} &= a'_{ij} \quad \text{if} \quad m+1\leq i\leq m+m'\\
&\qquad\qquad \text{and} \quad m+1\leq j\leq m+m',\\
a''_{ij} &= b_j \quad \text{if} \quad m+1\leq i\leq m+m' \quad \text{and} \quad 1\leq j\leq m,\\
a''_{ij} &= 0 \quad \text{if} \quad 1\leq i\leq m \quad \text{and} \quad m+1\leq j\leq m+m',\\
b''_i &= b_i \quad \text{if} \quad 1\leq i\leq m,\\
b''_i &= b'_i \quad \text{if} \quad m+1\leq i\leq m+m'.
\end{aligned}$$

The homomorphism corresponding to (a'',b'') is denoted $\phi'' = \phi\star\phi'$ [2].

Butcher proved that the ϕ derived from Runge–Kutta methods form a group for the product $\star$. Thus, each element ϕ has an inverse[3]. This inverse is quite important in practice, since it is involved in the concept of self-adjoint Runge–Kutta methods, which have long-term stability in time-reversal symmetric problems ([16], p. 219). In fact, Butcher found an explicit expression for all the operations of the Hopf structure of the ART.

Hairer and Wanner ([16], p. 267) built upon the work of Butcher and proved the following important result: If

[2] $(\phi\star\phi')(t) = m[(\phi\otimes\phi')\Delta(t)]$, the product is the convolution product of the Hopf algebra of rooted trees.

[3] $\phi^{-1}(t) = \phi[S(t)]$, where S is the antipode of the Hopf algebra structure.

we denote

$$B(\phi, F) = 1 + \sum_t \frac{(s-s_0)^{|t|}}{|t|!}\alpha(t)t!\phi(t)\delta_t(s_0),$$

then

$$B(\phi', B(\phi, F)) = B(\phi \star \phi', F).$$

The Hairer and Wanner theorem corresponds to the generalized Chen iterated integral theorem proven by Kreimer in [7][4].

5 The continuous limit

In his seminal article [18], Butcher did not restrict his treatment to finite sets of indices. It is possible to consider the continuous limit of Runge–Kutta methods. A possible form of it is an integral equation, which we write arbitrarily between 0 and 1:

$$X_u(s) = x_0 + (s-s_0)\int_0^1 \mathrm{d}v\, a(u,v)F(X_v(s)),$$
$$x(s) = x_0 + (s-s_0)\int_0^1 \mathrm{d}u\, b(u)F(X_u(s)),$$

the solution of which is

$$X_u(s) = x_0 + \sum_t \frac{(s-s_0)^{|t|}}{|t|!}\alpha(t)t!\int_0^1 \mathrm{d}v a(u,v)\phi_v(t)\delta_t(s_0),$$
$$x(s) = x_0 + \sum_t \frac{(s-s_0)^{|t|}}{|t|!}\alpha(t)t!\phi(t)\delta_t(s_0). \quad (6)$$

The homomorphism ϕ is defined recursively as a function of a and b:

$$\phi_u(\bullet) = 1,$$
$$\phi_u(B_+(t_1\cdots t_k)) = \int_0^1 \mathrm{d}u_1 a(u,u_1)\phi_{u_1}(t_1)\ldots \int_0^1 \mathrm{d}u_k a(u,u_k)\phi_{u_k}(t_k),$$
$$\phi(t) = \int_0^1 \mathrm{d}u b(u)\phi_u(t).$$

This definition is very close to the discrete version given in Sect. 4.1. Continuous Runge–Kutta (RK) methods do not seem to have been used much, except for an example in Butcher's book ([15] p. 325). Therefore, this section will take a slow pace and give many examples.

[4] I thank Dirk Kreimer for drawing my attention to this point.

5.1 Butcher's example

It will be useful in the following to have the results of a modified version of Butcher's example, so we consider:

$$X_u(s) = x_0 + (s-s_0)\int_0^u F[X_v(s)]\mathrm{d}v, \quad (7)$$
$$x(s) = x_0 + (s-s_0)\int_0^1 F[X_u(s)]\mathrm{d}u,$$

which corresponds to $a(u,v) = 1_{[0,u]}(v)$, $b(u) = 1$. Here $1_{[0,u]}(v)$ is the characteristic function of the interval $[0,u]$: it is 1 if $0 \le v \le u$ and 0 otherwise. This Runge–Kutta method will be referred to as the "simple integral method".

If we take the derivative of (7) with respect to u, we obtain

$$\frac{\mathrm{d}}{\mathrm{d}u}X_u(s) = (s-s_0)F[X_u(s)],$$

so $X_u(s) = y(s_0 + (s-s_0)u)$, where $y(s)$ is the solution

$$y(s) = x_0 + \int_{s_0}^s F[y(s')]\mathrm{d}s'.$$

Moreover,

$$\begin{aligned} x(s) &= x_0 + (s-s_0)\int_0^1 F[X_u(s)]\mathrm{d}u \\ &= x_0 + (s-s_0)\int_0^1 F[y(s_0+(s-s_0)u)]\mathrm{d}u \\ &= x_0 + \int_{s_0}^s F[y(s')]\mathrm{d}s' = y(s). \end{aligned}$$

The corresponding homomorphism $\phi(t)$ is defined by

$$\phi_u(\bullet) = 1,$$
$$\phi_u(B_+(t_1\cdots t_k)) = \int_0^u \mathrm{d}u_1\phi_{u_1}(t_1)\cdots\int_0^u \mathrm{d}u_k\phi_{u_k}(t_k),$$
$$\phi(t) = \int_0^1 \mathrm{d}u\phi_u(t).$$

Using the facts that $|B_+(t_1\cdots t_k)| = (|t_1| + \cdots + |t_k| + 1)$ and $B_+(t_1\cdots t_k)! = (|t_1| + \cdots + |t_k| + 1)t_1!\ldots t_k!$ it is proven that the solutions of these equations are

$$\phi_u(t) = \frac{|t|u^{|t|-1}}{t!},$$
$$\phi(t) = \frac{1}{t!}.$$

If we introduce $\phi(t) = 1/t!$ into (5), we obtain (4). Thus we confirm that the solution of the equation

$$x(s) = x_0 + \int_{s_0}^s F[x(s')]\mathrm{d}s'$$

is

$$x(s) = x_0 + \sum_t \frac{(s-s_0)^{|t|}}{|t|!}\alpha(t)\delta_t(s_0).$$

5.2 First applications

The above example already brings some interesting applications. But we must start by giving a way to calculate $\delta_t(s_0)$ in a simple case.

5.2.1 Calculation of $\delta_t(s_0)$

To obtain specific results, we must choose a particular function F. The simplest choice is to take a vector function F, where all components are identical; for any i from 1 to N, $f^i(x) = f(\bar{x})$, and $\bar{x}$ is the average over the components of x: $\bar{x} = \sum_{j=1}^{N} x^j/N$. We assume that f has the series expansion

$$f(s) = \sum_{n=0}^{\infty} \frac{f^{(n)}(0)s^n}{n!}.$$

From the definition of δ_t in (3), one can show recursively that for $i = 1, \ldots, N$, $\delta_t^i(0)$ is independent of i (and will be denoted δ_t) and

$$\delta_\bullet = f(0),$$
$$\delta_t = f^{(k)}(0)\delta_{t_1}\delta_{t_2}\cdots\delta_{t_k} \text{ when } t = B_+(t_1\,t_2\,\cdots t_k). \quad (8)$$

In [7], Kreimer defined a similar quantity that he called B_t. Here δ_t and B_t will be used as synonymous.

The simplest case is $f(s) = \exp s$ and $s_0 = 0$, where $f^{(n)}(0) = 1$ and $\delta_t = 1$ for all trees t.

5.2.2 Weighted sums of rooted trees

If we take $f = \exp$, $s_0 = 0$ and $x_0 = 0$ in the simple integral method (see Sect. 5.1), we have to solve the equation

$$x(s) = \int_0^s \exp[x(s')]\mathrm{d}s',$$

which can be differentiated to give $x'(s) = \exp(x(s))$ with $x(0) = 0$. This has the solution

$$x(s) = -\log(1-s) = \sum_{n=1}^{\infty} \frac{s^n}{n}.$$

On the other hand, the homomorphism corresponding to the simple integral method is $\phi(t) = 1/t!$, and the B-series for this problem is

$$x(s) = \sum_t \frac{s^{|t|}}{|t|!}\alpha(t).$$

Comparing the last two results, we find

$$\sum_{|t|=n} \alpha(t) = (n-1)!$$

In other words, the number of heap-ordered trees with n vertices is $(n-1)!$, as was said in Sect. 2.2.

5.2.3 Derivatives of inverse functions

We can try to extend the last example to an arbitrary function $f(x)$. The equation to solve becomes

$$x(s) = \int_0^s f[x(s')]\mathrm{d}s', \quad (9)$$

or $x'(s) = f(x(s))$ with $x(0) = 0$. Let

$$S(x) = \int_0^x \frac{\mathrm{d}y}{f(y)},$$

which gives us $s = S(x)$, and the solution of (9) is $x(s) = S^{-1}(s)$, where S^{-1} is the inverse function of S. If $f = \exp$, $S(x) = 1 - \exp(-x)$ and we confirm that $x(s) = -\log(1-s)$.

We can use this result to calculate the derivatives of a function $x(s)$, given as the inverse of a function $S(x)$. To do this, we define $f(x) = 1/S'(x)$ and, using (4), we obtain

$$x^{(n)}(0) = \sum_{|t|=n} \alpha(t)\delta_t, \quad (10)$$

where δ_t is calculated from $f(x)$ using (8) in Sect. 5.2.1.

This method can also be used to find the function f satisfying given values for

$$a_n = \sum_{|t|=n} \alpha(t)\delta_t,$$

where δ_t is calculated from f. For instance, if we want

$$\sum_{|t|=n} \alpha(t)\delta_t = n!,$$

we must take $f(x) = (1+x)^2$.

5.3 Kreimer's sum

In [7], Kreimer calculates the sum

$$S_n = \sum_{|t|=n} \frac{\alpha(t)}{t!}$$

using combinatorial arguments. We calculate it now with Butcher's tools. The sum S_n comes in the B-series (5) with $\phi(t) = 1/(t!)^2$. Since this $\phi(t)$ is the square of the previous one, the corresponding Runge–Kutta method can be realized as the tensor product of two simple integral methods (see Sect. 5.1). In other words,

$$a(u,u',v,v') = a(u,v)a(u',v') = 1_{[0,u]}(v)1_{[0,u']}(v'),$$
$$b(u,u') = b(u)b(u') = 1$$

and the Runge–Kutta method is now

$$X_{uu'}(s) = x_0 + (s-s_0)\int_0^u \mathrm{d}v \int_0^{u'} \mathrm{d}v' f[X_{vv'}(s)],$$
$$x(s) = x_0 + (s-s_0)\int_0^1 \mathrm{d}u \int_0^1 \mathrm{d}u' f[X_{uu'}(s)].$$

The corresponding homomorphism $\phi(t)$ is given by

$$\begin{aligned}
\phi_{uu'}(\bullet) &= 1,\\
\phi_{uu'}(B_+(t_1\cdots t_k)) &= \int_0^u \mathrm{d}u_1 \int_0^{u'} \mathrm{d}u_1'\phi_{u_1u_1'}(t_1)\ldots\\
&\qquad \int_0^u \mathrm{d}u_k \int_0^{u'} \mathrm{d}u_k'\phi_{u_ku_k'}(t_k),\\
\phi(t) &= \int_0^1 \mathrm{d}u \int_0^1 \mathrm{d}u'\phi_{uu'}(t).
\end{aligned}$$

The solutions of these equations are

$$\begin{aligned}
\phi_{uu'}(t) &= \frac{|t|^2(uu')^{|t|-1}}{(t!)^2},\\
\phi(t) &= \frac{1}{(t!)^2},
\end{aligned}$$

so that from (5),

$$X_{uu'}(s) = x_0 + \sum_t \frac{(s-s_0)^{|t|}}{|t|!}\frac{\alpha(t)(uu')^{|t|}}{t!}\delta_t(s_0).$$

The conclusion is that $X_{uu'}(s)$ is in fact a function of uu' and not of u and u'. More precisely, we know from the general formula (5) that the B-series for the integral $x(s) = x_0 + (s-s_0)\int_0^1 \mathrm{d}u \int_0^1 \mathrm{d}u' f[X_{uu'}(s)]$ is

$$x(s) = x_0 + \sum_t \frac{(s-s_0)^{|t|}}{|t|!}\frac{\alpha(t)}{t!}\delta_t(s_0),$$

thus $X_{uu'}(s) = x(s_0 + (s-s_0)uu')$. If we use the successive changes of variables $w = uu'$, $v' = s_0 + (s-s_0)w$ and $v = s_0 + (s-s_0)u$, we find

$$\begin{aligned}
x(s) &= x_0 + (s-s_0)\int_0^1 \mathrm{d}u \int_0^1 \mathrm{d}u' f[x(s_0 + (s-s_0)uu')]\\
&= x_0 + (s-s_0)\int_0^1 \frac{\mathrm{d}u}{u}\int_0^u \mathrm{d}w f[x(s_0+(s-s_0)w)]\\
&= x_0 + \int_0^1 \frac{\mathrm{d}u}{u}\int_{s_0}^{s_0+(s-s_0)u} \mathrm{d}v' f[x(v')]\\
&= x_0 + \int_{s_0}^s \frac{\mathrm{d}v}{v-s_0}\int_{s_0}^v \mathrm{d}v' f[x(v')].
\end{aligned}$$

With the initial values $x_0 = s_0 = 0$, this gives us

$$x(s) = \int_0^s \frac{\mathrm{d}v}{v}\int_0^v \mathrm{d}v' f[x(v')],$$

or $sx'' + x' = f(x)$ with $x(0) = 0$ and $x'(0) = f(0)$. If we take again $f(x) = \exp(x)$ we find $sx'' + x' = \exp(x)$ with $x(0) = 0$ and $x'(0) = 1$, hence

$$x(s) = -2\log(1 - s/2) = \sum_{n=1}^\infty \frac{s^n}{n2^{n-1}}.$$

Comparing this with the B-series

$$x(s) = \sum_t \frac{s^{|t|}}{|t|!}\frac{\alpha(t)}{t!}, \qquad (11)$$

we obtain

$$S_n = \sum_{|t|=n}\frac{\alpha(t)}{t!} = \frac{(n-1)!}{2^{n-1}},$$

which is the result found by Kreimer in [7].

As a final example, we can consider the Runge–Kutta method $a(u,v) = 1$, $b(u) = 1$, which gives $\phi(t) = 1$ for all trees t. The equation for $x(s)$ is now a fixed-point problem $x(s) = s\exp(x(s))$, whose solution was already known by Euler [19],

$$x(s) = \sum_n \frac{s^n}{n!}n^{n-1},$$

so that

$$\sum_{|t|=n}\alpha(t)t! = n^{n-1}.$$

These examples show that B-series can be used as generating series for sums over trees.

5.4 Butcher's group

The operations of Butcher's group are given here for continuous RK methods. Without loss of generality, we can take $s_0 = 0$ and rewrite the definition of continuous RK methods as

$$\begin{aligned}
X_u(s) &= x_0 + sA_u[F(X_v(s))],\\
X(s) &= x_0 + sB[F(X_u(s))].
\end{aligned}$$

In the above formula, the notation $A_u[F(X_v(s))]$ means that $X_v(s)$ is a function of v over which the operator A_u integrates. The result, $A_u[F(X_v(s))]$, is now a function of u. Similarly, $B[F(X_u(s))]$ integrates over u. A_u and B are linear operators. For example, the simple integral method corresponds to $A_u = \int_0^u \mathrm{d}v$ and $B = \int_0^1 \mathrm{d}v$.

The solution of these equations is

$$\begin{aligned}
X_u(s) &= x_0 + \sum_t \frac{s^{|t|}}{|t|!}\alpha(t)t!A_u[\phi_v(t)]\delta_t,\\
X(s) &= x_0 + \sum_t \frac{s^{|t|}}{|t|!}\alpha(t)t!\phi(t)\delta_t,
\end{aligned}$$

where, as usual,

$$\begin{aligned}
\phi_u(\bullet) &= 1,\\
\phi_u(B_+(t_1\cdots t_k)) &= A_u[\phi_{u_1}(t_1)]\ldots A_u[\phi_{u_k}(t_k)],\\
\phi(t) &= B[\phi_u(t)].
\end{aligned}$$

Each pair of operators (A_u, B) defines a continuous RK method. We can now define a product of RK methods.

5.4.1 The product

Following [18], the product of ϕ and ϕ' is denoted $\phi'' = \phi \star \phi'$ and is defined as follows. Let A_u, B, and A'_u, B' be the continuous Runge–Kutta methods of, respectively, $\phi(t)$ and $\phi'(t)$. To be specific, we consider that u varies from 0 to 1. Then the Runge–Kutta method for $\phi'' = \phi\star\phi'$ is A''_u, B'', where u varies from 0 to 2 and

$$\begin{aligned}
&A''_u(X_v) = A_u(X_v) \quad \text{if } 0 \le u \le 1 \quad \text{and } 0 \le v \le 1,\\
&A''_u(X_v) = 0 \quad \text{if } 0 \le u \le 1 \quad \text{and } 1 \le v \le 2,\\
&A''_u(X_v) = B(X_v) \quad \text{if } 1 \le u \le 2 \quad \text{and } 0 \le v \le 1,\\
&A''_u(X_v) = A'_{u-1}(X_{v-1}) \quad \text{if } 1 \le u \le 2 \quad \text{and } 1 \le v \le 2,\\
&B''(X_v) = B(X_v) \quad \text{if } 0 \le v \le 1,\\
&B''(X_v) = B'(X_{v-1}) \quad \text{if } 1 \le v \le 2.
\end{aligned}$$

We show the formula in action:

$$\begin{aligned}
\phi''_u(\bullet) &= 1,\\
\phi''(\bullet) &= B(1) + B'(1) = \phi(\bullet) + \phi'(\bullet),\\
\phi''_u(\overset{\circ}{\underset{\bullet}{|}}) &= A''_u(\phi''_v(\bullet))\\
&= A_u(1)1_{[0,1]}(u) + (B(1) + A'_{u-1}(1))1_{[1,2]}(u),\\
\phi''(\overset{\circ}{\underset{\bullet}{|}}) &= B(A_u(1)) + B'(B(1) + A'_{u-1}(1))\\
&= B(\phi_u(\bullet)) + B(1)B'(1) + B'(\phi'_u(\bullet))\\
&= \phi(\overset{\circ}{\underset{\bullet}{|}}) + \phi(\bullet)\phi'(\bullet) + \phi'(\overset{\circ}{\underset{\bullet}{|}}).
\end{aligned}$$

This is exactly the convolution product defined by Kreimer in [7].

5.4.2 The inverse

For a homomorphism $\phi(t)$, we define now the inverse of ϕ. If the Runge–Kutta method for ϕ is A_u, B, then the Runge–Kutta method for $\phi^S = \phi^{-1}$ is $A^S_u = A_u - B$, $B^S = -B$. It is useful to see it working on simple cases:

$$\begin{aligned}
\phi^S_u(\bullet) &= 1 = \phi_u(\bullet),\\
\phi^S(\bullet) &= B^S(\phi^S_u(\bullet)) = -B(\phi_u(\bullet)) = -\phi(\bullet),\\
\phi^S_u(\overset{\circ}{\underset{\bullet}{|}}) &= A^S_u(\phi^S_v(\bullet)) = A_u(\phi_v(\bullet)) - B(\phi_v(\bullet))\\
&= \phi_u(\overset{\circ}{\underset{\bullet}{|}}) - \phi(\bullet),\\
\phi^S(\overset{\circ}{\underset{\bullet}{|}}) &= -B(\phi^S_u(\overset{\circ}{\underset{\bullet}{|}})) = -\phi(\overset{\circ}{\underset{\bullet}{|}}) + B(1)\phi(\bullet)\\
&= -\phi(\overset{\circ}{\underset{\bullet}{|}}) + \phi(\bullet)\phi(\bullet).
\end{aligned}$$

This is the antipode of Kreimer's approach [7].

6 Runge–Kutta methods for renormalization

In this section, we shall follow closely Kreimer's paper [7] and define for some of his operations on homomorphisms the corresponding transformation of the Runge–Kutta methods. Instead of attempting a general theory, we consider a specific example in detail.

6.1 Runge–Kutta method for bare quantities

We consider that a given bare physical quantity can be calculated as a sum over trees. Examples of such quantities are given in [5,6], where the Hopf algebra was applied to iterated one-loop integrals. Broadhurst and Kreimer [5] have studied the sum

$$X(s) = \sum_t \frac{s^{|t|}}{|t|!} B_t.$$

To define B_t, they start from a function $L(\delta,\epsilon)$, regular (and equal to 1) at the origin. For notational convenience, the function L will be restricted to only one argument $L(\delta)$, with $L(0) = 1$. The first step is the definition of B_n as

$$B_n = \frac{L(n\epsilon)}{n\epsilon}.$$

Notice that B_n is singular when $\epsilon \to 0$.

B_t is now obtained recursively from these B_n by

$$\begin{aligned}
&B_\bullet = B_1,\\
&B_t = B_{|t|}B_{t_1}\cdots B_{t_k} \text{ when } t = B_+(t_1\cdots t_k). \qquad (12)
\end{aligned}$$

To apply Butcher's method, we take $F(X_v) = \exp(X_v)$ and $x_0 = 0$; this gives $\delta_t = 1$. Then we must find a pair of operators (A_u,B) such that $\phi(t) = B_t$. We choose

$$A_u(X_v) = \frac{1}{\epsilon}\int_0^u \mathrm{d}v L(\epsilon\frac{\mathrm{d}}{\mathrm{d}v}v)X_v, \quad B(X_v) = A_1(X_v).$$

The only thing that we need in the following is the action of A_u on a monomial v^{n-1}

$$\begin{aligned}
A_u(v^{n-1}) &= \frac{1}{\epsilon}\int_0^u \mathrm{d}v L(\epsilon\frac{\mathrm{d}}{\mathrm{d}v}v)v^{n-1}\\
&= \frac{1}{\epsilon}\int_0^u v^{n-1}\mathrm{d}v L(n\epsilon) = B_n u^n. \qquad (13)
\end{aligned}$$

The quantity of interest $x(s)$ is then obtained by tensoring A_u with the simple integral method to obtain $\phi(t) = B_t/t!$.

6.2 The "twisted antipode"

In [7], Kreimer defines recursively S_R, a twisted antipode[5] depending on a given renormalization scheme R. We take as an example the toy model used by Kreimer, and we choose the minimal subtraction scheme: The Laurent expansion of ϕ is developed with respect to ϵ, and $R[\phi] = \langle\phi\rangle$ is the pole part of this expansion.

Following the results of Sect. 5.4.2, the Runge–Kutta method for $S_R(\phi)$ can be obtained from the Runge–Kutta method of ϕ by the definition $A^S_u(X) = A_u(X) - \langle A_1(X)\rangle$,

[5] $S_R(\phi)(t) = -R\big[\phi(t) + m[(S_R\otimes Id)(\phi\otimes\phi)P_2\Delta(t)]\big]$ in Hopf algebra terms.

$B^S(X) = -\langle A_1(X)\rangle$. Working out the first examples using (13), we find

$$
\begin{aligned}
\phi_u^S(\bullet) &= 1,\\
\phi^S(\bullet) &= -\langle A_1(1)\rangle = -\langle B_1\rangle,\\
\phi_u^S(\overset{\bullet}{\underset{\bullet}{|}}) &= A_u(\phi_v^S(\bullet)) - \langle A_1(\phi_v^S(\bullet))\rangle = A_u(1) - \langle A_1(1)\rangle\\
&= B_1 u - \langle B_1\rangle,\\
\phi^S(\overset{\bullet}{\underset{\bullet}{|}}) &= -\langle A_1(\phi_u^S(\overset{\bullet}{\underset{\bullet}{|}}))\rangle = -\langle B_2 B_1\rangle + \langle\langle B_1\rangle B_1\rangle.
\end{aligned}
$$

This is identical to the first results of Sect. 4 of [7].

6.3 Renormalized quantities

Finally, the renormalized quantities $x^R(s)$ are obtained from the convolution of $S_R(\phi)$ with ϕ. To obtain the corresponding Runge–Kutta method, we use the results of Sect. 5.4.1. One can show that the domain $1 \leq u \leq 2$ is not used, and the Runge–Kutta method for the renormalized quantity is simply $A_u^R(X) = A_u(X) - \langle A_1(X)\rangle$, $B^R(X) = A_1(X) - \langle A_1(X)\rangle$. It may seem surprising that such a simple equation encodes the full combinatorial complexity of renormalization. It is not even necessary to work examples out, because $A_u^R(X) = A_u^S(X)$, so that $\phi_u^R(t) = \phi_u^S(t)$, and the only difference comes from the action of B^R.

For an explicit calculation of $X^R(s)$, we do not need A_u^R and B^R which give us $\phi(t)$, but rather the tensor product of this method with the simple integral method to obtain $t!\phi(t)$. In detail, the equation for the renormalized quantity $X^R(s)$ is

$$
\begin{aligned}
X_{uu'}^R(s) &= \frac{s}{\epsilon}\int_0^u \mathrm{d}v \int_0^{u'} \mathrm{d}v' L(\epsilon\partial_v v)\mathrm{e}^{X_{vv'}(s)} -\\
&\qquad \langle \frac{s}{\epsilon}\int_0^1 \mathrm{d}v \int_0^{u'} \mathrm{d}v' L(\epsilon\partial_v v)\mathrm{e}^{X_{vv'}(s)}\rangle, \quad (14)\\
X^R(s) &= \frac{s}{\epsilon}\int_0^1 \mathrm{d}v \int_0^{1} \mathrm{d}v' L(\epsilon\partial_v v)\mathrm{e}^{X_{vv'}(s)} -\\
&\qquad \langle \frac{s}{\epsilon}\int_0^1 \mathrm{d}v \int_0^{1} \mathrm{d}v' L(\epsilon\partial_v v)\mathrm{e}^{X_{vv'}(s)}\rangle. \quad (15)
\end{aligned}
$$

For a general renormalization scheme R, one replaces $\langle A_u(X)\rangle$ by $R[A_u(X)]$.

7 Renormalization of Kreimer's toy model

In this section, we use Runge–Kutta methods to renormalize explicitly Kreimer's toy model for even functions $L(\epsilon)$. This toy model was studied for pedagogical reasons in [5,7], as it provides a convenient means to introduce renormalization theory. Generalizations to more realistic situations are discussed in [6]. In [5], remarkable properties of the renormalized sum of diagrams with "Connes–Moscovici weights" (i.e., $\alpha(t)$) were noticed. Some of the properties conjectured in [5] will be proven in this section.

7.1 Equation for the renormalized quantity

The role of the sum over u' in (15) is to add a factor $1/t!$, as in Sect. 5.3. Therefore, the same reasoning can be used to show that $X_{uu'}^R(s)$ is in fact a function of su' and we write $X_{uu'}^R(s) = X_u^R(su')$, which defines the function $X_u^R(s)$. The equation for $X_u^R(s)$ can be found from (15) and the relation $X_u^R(s) = X_{us}^R(1)$ as

$$
\begin{aligned}
X_u^R(s) &= \frac{1}{\epsilon}\int_0^u \mathrm{d}v \int_0^{s} \mathrm{d}s' L(\epsilon\partial_v v)\mathrm{e}^{X_v(s')} -\\
&\qquad \langle \frac{1}{\epsilon}\int_0^1 \mathrm{d}v \int_0^{s} \mathrm{d}s' L(\epsilon\partial_v v)\mathrm{e}^{X_v(s')}\rangle, \quad (16)\\
X^R(s) &= \frac{1}{\epsilon}\int_0^1 \mathrm{d}v \int_0^{s} \mathrm{d}s' L(\epsilon\partial_v v)\mathrm{e}^{X_v(s')} -\\
&\qquad \langle \frac{1}{\epsilon}\int_0^1 \mathrm{d}v \int_0^{s} \mathrm{d}s' L(\epsilon\partial_v v)\mathrm{e}^{X_v(s')}\rangle\\
&= X_1^R(s).
\end{aligned}
$$

To solve this equation, we expand $X_u^R(s)$ in a power series over u:

$$X_u^R(s) = \sum_{n=0}^{\infty} a_n(s) u^n.$$

A standard identity [20] gives us

$$
\begin{aligned}
\exp(X_u^R(s)) &= e^{a_0(s)}\sum_{n=0}^{\infty}\lambda_n(a)u^n, \quad \text{where}\\
\lambda_n(a) &= \sum_{|\alpha|=n}\frac{a_1^{\alpha_1}\cdots a_n^{\alpha_n}}{\alpha_1!\cdots\alpha_n!},
\end{aligned}
$$

with $|\alpha| = \alpha_1 + 2\alpha_2 + \cdots + n\alpha_n$. $\lambda_n(a)$ depends on s through its arguments $a_i(s)$. The sets of α_i for a given n can be obtained from the partitions of n: $(\mu_1,\ldots,\mu_n)$, where $\mu_1 \geq \cdots \geq \mu_n$ by $\alpha_n = \mu_n$, $\alpha_i = \mu_i - \mu_{i+1}$ for $i < n$.

The first few $\lambda_n(a)$ are

$$
\begin{aligned}
&\lambda_0(a) = 1, \quad \lambda_1(a) = a_1, \quad \lambda_2(a) = a_2 + \frac{a_1^2}{2},\\
&\lambda_3(a) = a_3 + a_1 a_2 + \frac{a_1^3}{6}.
\end{aligned}
$$

7.2 Solution of the equation

Introducing the series for $X_u^R(s)$ and $\exp(X_u^R(s))$ into (17), we obtain

$$
\begin{aligned}
\sum_{n=0}^{\infty} a_n(s)u^n &= \sum_{n=0}^{\infty} B_{n+1}\int_0^s \mathrm{e}^{a_0(s')}\lambda_n(a)\mathrm{d}s' u^{n+1} -\\
&\qquad \langle \sum_{n=0}^{\infty} B_{n+1}\int_0^s \mathrm{e}^{a_0(s')}\lambda_n(a)\mathrm{d}s'\rangle,
\end{aligned}
$$

or

$$a_0(s) = -\langle\sum_{n=0}^{\infty} B_{n+1}\int_0^s \mathrm{e}^{a_0(s')}\lambda_n(a)\mathrm{d}s'\rangle,$$
$$a_n(s) = B_n\int_0^s \mathrm{e}^{a_0(s')}\lambda_{n-1}(a)\mathrm{d}s' \quad \text{for} \quad n>0. \quad (17)$$

To solve this equation, we need to go back to the equation for the bare quantity,

$$X_u^0(s) = \frac{1}{\epsilon}\int_0^u \mathrm{d}v\int_0^s \mathrm{d}s' L(\epsilon\partial_v v)\mathrm{e}^{X_v^0(s')}. \quad (18)$$

Again $X_u^0(s)$ is a function of su; we define $X^0(s) = X_s^0(1)$, which satisfies

$$X^0(s) = \frac{1}{\epsilon}\int_0^s \frac{\mathrm{d}u}{u}\int_0^u \mathrm{d}v L(\epsilon\partial_v v)\mathrm{e}^{X^0(v)}. \quad (19)$$

The solution of this equation is given by the B-series

$$X^0(s) = \sum_{n=1}^{\infty}\bar{a}_n s^n \quad \text{and} \quad \bar{a}_n = \sum_{|t|=n}\frac{\alpha(t)B_t}{|t|!}. \quad (20)$$

On the other hand, we can also expand $\mathrm{e}^{X^0(v)}$ using the functions $\lambda_n(\bar{a})$ and introduce the resulting series in (19). This gives us

$$X^0(s) = \sum_{n=1}^{\infty}\frac{B_n}{n}\lambda_{n-1}(\bar{a})s^n.$$

By comparison with (20), we obtain the relation

$$\bar{a}_n = \frac{B_n}{n}\lambda_{n-1}(\bar{a}). \quad (21)$$

With this identity, we can now prove that for the renormalized quantities,

$$a_n(s) = (g(s))^n\bar{a}_n, \text{with}\, g(s) = \int_0^s \exp(a_0(s'))\mathrm{d}s'. (22)$$

Since $\lambda_0(a) = 1$ and $\bar{a}_1 = B_1$, this equation is true for $n = 1$, from (17). If (22) is true up to $n-1$, then $\lambda_{n-1}(a) = (g(s))^{n-1}\lambda_{n-1}(\bar{a})$, and the derivative of (17) gives us

$$a_n'(s) = B_n e^{a_0(s)}\lambda_{n-1}(a) = B_n g'(s)(g(s))^{n-1}\lambda_{n-1}(\bar{a})$$
$$= ng'(s)(g(s))^{n-1}\bar{a}_n,$$

by (21). Integrating this equation with the condition $a_n(0) = 0$ gives (22) at level n.

By this we have proved that $X_u^R(s) = a_0(s) + X^0(ug(s))$ and $X^R(s) = a_0(s) + X^0(g(s))$: The flow for the renormalized quantity is a reparametrization of the flow for the bare quantity, plus a singular term that cancels the singular part of the bare flow, since $a_0(s) = -\langle X^0(ug(s))\rangle$ from (17).

To determine $a_0(s)$ we proceed step by step. In (20) we expand $L(\epsilon\partial_v v)$ over ϵ. The first term is just 1, and we obtain (11) with the solution $x(s) = -2\log(1 - s/(2\epsilon))$. For the renormalized quantity, the most singular term becomes $X^0(g(s)) = -2\log(1 - g(s)/(2\epsilon))$. Since $X^R(s)$ is regular, this singular term, must be compensated by a corresponding term in $a_0(s)$. By equating the most singular terms we obtain $a_0(s) = 2\log(1 - g(s)/(2\epsilon))$. We know from (22) that $a_0(s) = \log(g'(s))$, and we obtain the most singular term as the solution of the equation $g'(s) = (1 - g(s)/(2\epsilon))^2$, which is:

$$g(s) = s/(1+\frac{s}{2\epsilon}),$$
$$a_0(s) = -2\log(1+\frac{s}{2\epsilon}).$$

By expanding $a_0(s)$ as a series in s, we obtain the most singular term observed in [5] and proven in [7]. One notices that the singularity of the nonpertubative term $a_0(s)$ is logarithmic ($2\log\epsilon$), and much smoother than the singularities coming from the expansion over s (i.e., the perturbative expression).

7.3 Differential equation for the finite part

In general, one should proceed at this point with the next singular term. To obtain it we denote $Y(s) = X^0(g(s))$; this change of variable gives the equation for $Y(s)$:

$$Y(s) = \frac{1}{\epsilon}\int_0^s \frac{g'(u)\mathrm{d}u}{g(u)}\int_0^u \mathrm{d}v g'(v)L(\epsilon + \epsilon\frac{g(v)}{g'(v)}\partial_v)\mathrm{e}^{Y(v)}.$$

Since $Y(s) = X^R(s) - a_0(s)$ and

$$a_0(s) = -2\log(1+\frac{s}{2\epsilon}) = -\int_0^s \frac{\mathrm{d}u}{u(\epsilon + u/2)}\int_0^u \mathrm{d}v,$$

we obtain the equation for $X^R(s)$:

$$X^R(s) = \frac{1}{\epsilon}\int_0^s \frac{\mathrm{d}u}{u(1+\frac{u}{2\epsilon})}\int_0^u \mathrm{d}v$$
$$\left[\frac{1}{(1+\frac{v}{2\epsilon})^2}L(\epsilon\partial_v v + \frac{v^2}{2}\partial_v)(1+\frac{v}{2\epsilon})^2\mathrm{e}^{X^R(v)} - 1\right].$$

The nice aspect of the above equation is that it seems to have a limit as ϵ tends to zero. In fact, it has a limit when L is even, as we shall show now.

Writing $\bar{X}(s) = \lim_{\epsilon\to 0} X^R(s)$, and taking the limit $\epsilon \to 0$ in the previous equation, we obtain

$$\bar{X}(s) = 2\int_0^s \frac{\mathrm{d}u}{u^2}\int_0^u \mathrm{d}v\left[\frac{1}{v^2}L(\frac{v^2}{2}\partial_v)v^2\mathrm{e}^{\bar{X}(v)} - 1\right],$$

or, in differential form:

$$\frac{1}{2}(s^2\bar{X}'(s))' = \frac{1}{s^2}L(\frac{s^2}{2}\frac{\mathrm{d}}{\mathrm{d}s})s^2\mathrm{e}^{\bar{X}(s)} - 1. \quad (23)$$

If $\bar{X}(s)$ and $L(\delta)$ are expanded as

$$\bar{X}(s) = \sum_{n=1}^{\infty} b_n s^n \quad \text{and} \quad L(\delta) = 1 + \sum_{n=1}^{\infty} L_n\delta^n,$$

and thus,

$$L(\frac{s^2}{2}\frac{\mathrm{d}}{\mathrm{d}s}) = 1 + \sum_{n=1}^{\infty} L_n(\frac{s^2}{2}\frac{\mathrm{d}}{\mathrm{d}s})^n,$$

we obtain the following relation for the term in s: $b_1 s = (b_1 + L_1/2)s$. If L_1 is not zero, we obtain a contradiction and must proceed further with the withdrawal of divergences. For simplicity, we shall assume that $L_1 = 0$. Then b_1 becomes a free parameter of $\bar{X}(s)$ that can be determined by a renormalization condition. All terms b_n with $n > 1$ can now be determined from b_1 and L_n ($n > 1$). All terms are regular. In other words, if $X^0(g(s))$ is expanded as a series $\sum_n a_n(\epsilon)s^n$, then

$$a_n(\epsilon) = \frac{(n-1)!}{2^{n-1}\epsilon^n} + R_n(\epsilon),$$

and $R_n(\epsilon)$ is regular if $L_1 = 0$. This fact was conjectured in [5] for even functions $L(\delta)$, which obviously satisfy $L_1 = 0$. Their solution corresponds to the case $b_1 = 0$. Broadhurst and Kreimer have also used a function $L(\epsilon, \delta)$. The present treatment can be applied to this more general situation, with the only change being that one has to use

$$L_n = n! \lim_{\epsilon \to 0} \lim_{\delta \to 0} \frac{\mathrm{d}^n}{\mathrm{d}\delta^n} L(\epsilon, \delta).$$

Clearly, (23) is much faster to solve than a computation of the sum over trees. For instance, the expansion could be calculated up to 20 loops (i.e., b_{20}) within a few seconds with a computer. Moreover, (23) is a nonperturbative equation for the renormalized quantity $\bar{X}(s)$.

7.4 Alternative point of view

There is an alternative way to solve (18) for the bare quantity. We define a function $f(s)$ from $L(\delta)$ by

$$f(s,\epsilon) = \sum_{n=0}^{\infty} \frac{L(n\epsilon + \epsilon)}{n!} s^n = L(\epsilon \frac{\mathrm{d}}{\mathrm{d}s} s)\mathrm{e}^s.$$

A relation between $f(s,\epsilon)$ and $L(\delta)$ can also be established through the Mellin transforms of f and L as $M(f)(z,\epsilon) = M(L)(\epsilon - \epsilon z)\Gamma(z)$.

With $f(s,\epsilon)$ we can write the equation for the bare quantity as

$$X^0(s) = \frac{1}{\epsilon}\int_0^s \frac{\mathrm{d}u}{u}\int_0^u \mathrm{d}v f(X^0(v),\epsilon). \tag{24}$$

Alternatively, one can go from f to L and consider the results of the toy model as a method to renormalize equations of the type in (24).

8 n-dimensional problems

For applications to classical field theory, we need to develop Runge–Kutta methods for the n-dimensional analog of the flow equation: nonlinear partial differential equations. The purpose of the present section is to indicate how B-series can be used for this case. The method applies to equations of the form $L\psi(\mathbf{r}) = F[\psi(\mathbf{r})]$, where L is a linear differential operator (e.g., the nonlinear Schrödinger equation $\Delta\psi = \psi^3$).

8.1 Formulation

We need two starting elements: a function $\psi_0(\mathbf{r})$ which is the solution of $L\psi_0(\mathbf{r}) = 0$, and a Green function $G(\mathbf{r},\mathbf{r}')$, that is a solution of the equation $L_r G(\mathbf{r},\mathbf{r}') = \delta(\mathbf{r}-\mathbf{r}')$, with given boundary conditions. The function $\psi_0(\mathbf{r})$ will play the role of an initial value, and the Green function will decide in which "direction" one moves from the initial value. It will also state, in some sense, the boundary conditions of the solution $\psi(\mathbf{r})$.

Using these two functions, the differential equation $L\psi(\mathbf{r}) = F[\psi(\mathbf{r})]$ is transformed into

$$\psi(\mathbf{r}) = \psi_0(\mathbf{r}) + \int \mathrm{d}\mathbf{r}' G(\mathbf{r},\mathbf{r}') F[\psi(\mathbf{r}')]. \tag{25}$$

The action of L enables us to go from the second to the first equation.

As shown in the appendix, the solution of (25) is

$$\psi(\mathbf{r}) = \psi_0(\mathbf{r}) + \sum_t \frac{\alpha(t)t!}{|t|!}\phi_r(t), \tag{26}$$

where $\phi_r(t)$ is defined recursively by

$$\phi_r(\bullet) = \int \mathrm{d}r' G(\mathbf{r},\mathbf{r}') F[\psi_0(\mathbf{r}')],$$

$$\phi_r(B_+(t_1\cdots t_k)) = \int \mathrm{d}r' G(\mathbf{r},\mathbf{r}') F^{(k)}[\psi_0(\mathbf{r}')] \phi_{r'}(t_1)\ldots\phi_{r'}(t_k). \tag{27}$$

If ψ is a vector field and F has components f^i, the solution is still given by (26), and the recursive relation for the components ϕ^i becomes:

$$\phi^i_r(\bullet) = \int \mathrm{d}r' G^i_j(\mathbf{r},\mathbf{r}') f^j[\psi_0(\mathbf{r}')],$$

$$\phi^i_r(B_+(t_1\cdots t_k)) = \int \mathrm{d}r' G^i_j(\mathbf{r},\mathbf{r}') f^j_{j_1\ldots j_k}[\psi_0(\mathbf{r}')] \phi^{j_1}_{r'}(t_1)\ldots\phi^{j_k}_{r'}(t_k),$$

where $G^i_j(\mathbf{r},\mathbf{r}')$ is a component of the matrix Green function.

In the previous sections, the series (5) was written in terms of $\phi(t)$ (describing the effect of the Runge–Kutta method (a,b)) and δ_t (describing the effect of the function $F[x]$). In the present case, this separation is no longer possible, and $\phi(t)$ combines both pieces of information.

Several examples of the use of (26) will be given in the next section. It will be shown that (26) is equivalent to (5) for one-dimensional problems and that (26) gives the usual Born expansion when $F(\psi)$ is a linear operator. Finally, the example of a cubic Klein–Gordon equation will be briefly discussed.

8.2 Simple examples

In this section, (26) is applied to the one-dimensional problem and to the Schrödinger equation.

8.2.1 The one-dimensional case

It is instructive to observe how the one-dimensional case is obtained from (26). The linear differential operator is $L = \mathrm{d}/\mathrm{d}s$, so the initial function $\psi_0(s)$ must satisfy $\mathrm{d}\psi_0(s)/\mathrm{d}s = 0$: $\psi_0(s)$ is a constant that we write x_0. For the Green function $G(s,s')$, we have the equation $LG(s,s') = \delta(s-s')$, so $G(s,s') = \theta(s-s') + C(s')$, where $\theta(s)$ is the step function and $C(s')$ a function of s'. To determine $C(s')$, we note that in the simple integral method, there is an integral from s_0 to s. This gives us $C(s') = -\theta(s_0 - s')$, and we obtain

$$\int_{-\infty}^{\infty} G(s,s')f(s')\mathrm{d}s' = \int_{s_0}^{s} f(s')\mathrm{d}s',$$

which is the required expression.

Now, the role of ψ_0 and of the Green function is clear for the one-dimensional case: ψ_0 gives the initial value x_0 and G specifies (among other things) the starting point s_0. To complete the derivation of the one-dimensional case, we note that $\psi_0(s) = x_0$ does not depend on s, so the terms $F^{(k)}[\psi_0(s)] = F^{(k)}[x_0]$ are independent of s and can be grouped together to build δ_t as in (3). On the other hand, the integration over s' builds up $(s-s_0)^{|t|}/t!$, and we obtain (4).

8.2.2 The Schrödinger equation

If we write the Schrödinger equation as $(E + \Delta)\psi(\mathbf{r}) = V(\mathbf{r})\psi(\mathbf{r})$, we can apply (26) with $F[\psi] = V(\mathbf{r})\psi$ because, as is explained in the appendix, the expansion (26) is valid also for F depending on $\mathbf{r}$ (i.e., $F[\psi,\mathbf{r}]$).

We take for $\phi_0(\mathbf{r})$ a solution of $(E+\Delta)\phi_0(\mathbf{r}) = 0$ and for $G(\mathbf{r},\mathbf{r}')$ the scattering Green function in free space (e.g., $G(\mathbf{r}-\mathbf{r}') = -\mathrm{e}^{\mathrm{i}\sqrt{E}|\mathbf{r}-\mathbf{r}'|}/(4\pi|\mathbf{r}-\mathbf{r}'|)$ in three dimensions).

The calculation of $\phi(t)$ is straightforward because in such a linear problem, $F^{(k)} = 0$ for $k > 1$. Hence, the only rooted trees that survive are those with one branch. For these trees $\alpha(t) = 1$ and $t! = |t|!$ and we obtain

$$\psi(\mathbf{r}) = \psi_0(\mathbf{r}) + \int \mathrm{d}\mathbf{r}_1 G(\mathbf{r},\mathbf{r}_1)V(\mathbf{r}_1)\psi_0(\mathbf{r}_1) + \int \mathrm{d}\mathbf{r}_1\mathrm{d}\mathbf{r}_2 G(\mathbf{r},\mathbf{r}_1)V(\mathbf{r}_1)G(\mathbf{r}_1,\mathbf{r}_2)V(\mathbf{r}_2)\psi_0(\mathbf{r}_2) + \cdots$$

where we recognize the Born expansion of the Lippmann–Schwinger equation.

8.3 Nonlinear Klein–Gordon equation

As a more elaborate example, we consider the nonlinear Klein–Gordon equation for $v(x,t)$ studied by Jiang and Wong [21]

$$v_{tt} - \gamma^2 v_{xx} + c^2 v - \epsilon v^3 = 0, \tag{28}$$

with $t \geq 0$ and the boundary conditions $v(x,0) = \cos kx$ and $v_t(x,0) = 0$.

The Green function corresponding to this problem can be obtained by standard methods [22] as

$$G(x,t;x',t') = (\theta(t-t') - \theta(-t'))g(x-x',t-t'),$$

with

$$\begin{aligned} g(x,t) &= \frac{1}{2\pi}\int_{-\infty}^{\infty}\mathrm{d}q\frac{\sin(\omega(q)t)\cos qx}{\omega(q)}, \\ g(x,t) &= \frac{1}{2\gamma}J_0\big(c\sqrt{t^2 - q^2/\gamma^2}\big) \quad \text{for} \quad |q| < \gamma t \\ &= 0 \quad \text{for} \quad |q| > \gamma t. \end{aligned}$$

with $\omega(q) = \sqrt{\gamma^2 q^2 + c^2}$. The initial function is $\psi_0(x,t) = \cos wt \cos kx$ (where $w = \omega(k)$), the nonlinear function is $F(\psi) = \epsilon\psi^3$. The solution of the nonlinear equation (28) is given by formula (26), the second term of which is calculated by (27) as

$$\begin{aligned} \phi(\bullet\,;x,t) = {} & \frac{9\epsilon t}{32w}\sin wt\cos kx + \\ & \frac{3\epsilon}{128w^2}(\cos wt - \cos 3wt)\cos kx + \\ & \frac{3\epsilon}{128\gamma^2k^2}(\cos wt - \cos w_3 t)\cos 3kx + \\ & \frac{\epsilon}{128c^2}(\cos w_3 t - \cos 3wt)\cos 3kx, \end{aligned}$$

where $w_3 = \omega(3k)$, and we have written $\phi(\bullet\,;x,t)$ for the $\phi_r(\bullet)$ of (27).

This result agrees with that of [21]. In comparison with the standard method, where a differential equation must be solved at each step, taking care of the boundary conditions, the present method is completely automatic and well suited to symbolic computational packages.

The unbounded term $[(9\epsilon t)/(32w)]\sin wt\cos kx$ in $\phi(\bullet\,;x,t)$ is called a secular term. It reduces the range of convergence of (26). It can be canceled by a renormalization of c defined by $c^2 \to c^2 - 9\epsilon/16$. With this new definition of c, the Klein–Gordon equation becomes

$$v_{tt} - \gamma^2 v_{xx} + c^2 v = \epsilon v^3 - \epsilon\frac{9}{16}v.$$

The Green function is the same as for (28), with a modified value of c, and the nonlinear term is now $F(\psi) = \epsilon\psi^3 - 9\epsilon\psi/16$. Let us call $\phi_R(\bullet\,;x,t)$ the first term of the B-series for this new equation:

$$\begin{aligned}\phi_R(\bullet\,;x,t) = {} & \frac{3\epsilon}{128w^2}(\cos wt - \cos 3wt)\cos kx + \\ & \frac{3\epsilon}{128\gamma^2k^2}(\cos wt - \cos w_3t)\cos 3kx + \\ & \frac{\epsilon}{128c^2}(\cos w_3t - \cos 3wt)\cos 3kx;\end{aligned}$$

the secular term has disappeared. However, the next term $\phi_R(\begin{smallmatrix}\circ\\ |\\ \bullet\end{smallmatrix}\,;x,t)$ still has a secular contribution,

$$\begin{aligned}\frac{3\epsilon^2 t}{1024w}\Big(& (\frac{1}{c^2} - \frac{3}{\gamma^2k^2})\sin w_3t\cos 3kx + \\ & (\frac{9}{2w^2} - \frac{1}{4c^2} + \frac{9}{4\gamma^2k^2})\sin wt\cos kx\Big).\end{aligned}$$

It would be interesting to see if this term can also be renormalized.

Within Butcher's approach, it is still necessary to find the generalization of RK equations to these nonlinear partial differential equations such that for any map $\phi_r(t)$, there is a member of the family of equations of which ϕ_r is a solution.

9 Conclusion

Butcher's approach to Runge–Kutta methods was applied to some simple renormalization problems. Since Cayley, it has been clear that the rooted trees are ideally suited to treat differentials. This was confirmed here by the presentation of a B-series solution of a class of nonlinear partial differential equations.

The recursive nature of the B-series makes them computationally efficient: $\phi_u(t)$ can be obtained by a simple operation from the $\phi_u(t')$ of smaller order t'.

Butcher's approach has still much to offer. In the numerical analysis literature, B-series have been generalized to treat flow equations on Lie groups. The main change [23] is to replace the algebra of rooted trees by the algebra of planar trees (also called ordered trees [24]). The elementary differentials then get a "quantized calculus" flavor, especially in the definition given by Munthe–Kaas [25] in terms of commutators with the vector field $F = f^i\partial_i$ (see also [26]). Using this generalized ART, extended work has been carried out recently for the numerical solution of differential equations on Lie groups (see [23,24] and the web site `http://www.math.ntnu.no/num/synode`).

B-series have been generalized in other directions, e.g., stochastic differential equations [27] and differential equations of the type $\mathrm{d}y/\mathrm{d}s = f(y,z)$, $g(y,z) = 0$, which are called differential algebraic equations [28].

In a subsequent paper [29], it will be shown that many features of Butcher's approach can be adapted to the equations with functional derivatives that are met in quantum field theory. The main difference comes from the fact that planar binary trees will be used instead of rooted trees.

Acknowledgements. It is a great pleasure to thank Dirk Kreimer, Alain Connes, and John Butcher for interest, encouragement, and discussions. I am grateful to Ale Frabetti, Philippe Sainctavit, Dirk Kreimer, and David Broadhurst for their thorough readings of the manuscript. This is IPGP contribution #1627.

10 Appendix

To prove that (26) is the solution of (25) if $\phi_r(t)$ is given by (27), we define $\psi(\mathbf{r})$ by

$$\psi(\mathbf{r}) = \psi_0(\mathbf{r}) + \sum_t \frac{\alpha(t)t!}{|t|!}\phi_r(t) = \psi_0(\mathbf{r}) + \sum_t \frac{1}{\sigma(t)}\phi_r(t),$$

where (1) is used. From a Taylor expansion of F, we can write

$$\begin{aligned}F[\psi(\mathbf{r})] &= \sum_{k=0}^{\infty} \frac{\big(\psi(\mathbf{r}) - \psi_0(\mathbf{r})\big)^k}{k!} F^{(k)}(\psi_0(\mathbf{r})) \\ &= \sum_{k=0}^{\infty} \frac{\big(\sum_t \phi_r(t)/\sigma(t)\big)^k}{k!} F^{(k)}(\psi_0(\mathbf{r})) \\ &= \sum_{k=0}^{\infty} \frac{1}{k!}\sum_{t_1\ldots t_k} \frac{1}{\sigma(t_1)}\cdots\frac{1}{\sigma(t_k)} \\ &\qquad F^{(k)}(\psi_0(\mathbf{r}))\phi_r(t_1)\ldots\phi_r(t_k).\end{aligned}$$

Now we multiply $F[\psi(\mathbf{r}')]$ by $G(\mathbf{r},\mathbf{r}')$ and integrate over $\mathbf{r}'$. For $k = 0$, we obtain simply $\phi_r(\bullet\,)$; for $k > 1$, the definition (27) of $\phi_r(t)$ gives us

$$\begin{aligned}\int \mathrm{d}\mathbf{r}' G(\mathbf{r},\mathbf{r}')F[\psi(\mathbf{r}')] &= \phi_r(\bullet\,) + \\ & \sum_{k=1}^{\infty}\frac{1}{k!}\sum_{t_1\ldots t_k}\frac{1}{\sigma(t_1)}\cdots\frac{1}{\sigma(t_k)}\phi_r(B_+(t_1\ldots t_k)).\end{aligned}$$

If we sum over all the k-tuples of trees $(t_1,\ldots,t_k)$ which give the same tree $B_+(t_1\ldots t_k)$, we obtain for this sum

$$\frac{1}{k!}\sum_{t_1\ldots t_k}\frac{1}{\sigma(t_1)}\cdots\frac{1}{\sigma(t_k)} = \frac{1}{\sigma\big(B_+(t_1\ldots t_k)\big)},$$

because of the definition of $\sigma(t)$ given by (2). Now, in the sum over trees that is left, each tree different from $\bullet$ is generated once and only once by $B_+(t_1\ldots t_k)$. Therefore,

$$\begin{aligned}\int \mathrm{d}\mathbf{r}' G(\mathbf{r},\mathbf{r}')F[\psi(\mathbf{r}')] &= \phi_r(\bullet\,) + \sum_{t\neq\bullet}\frac{1}{\sigma(t)}\phi_r(t) \\ &= \psi(\mathbf{r}) - \psi_0(\mathbf{r}).\end{aligned}$$

We have checked that $\psi(\mathbf{r})$ is a solution of (25).

It is clear that the above proof remains valid if F is vector-valued and if F depends on $\mathbf{r}$ ($F = F[\psi,\mathbf{r}]$), but the derivatives $F^{(k)}$ are still taken with respect to ψ only. Moreover, equations of the type $L\psi(\mathbf{r}) = f(\mathbf{r}) + F[\psi(\mathbf{r})]$, where $f(\mathbf{r})$ is a given function, can be solved with exactly the same formula, except that now $\psi_0(\mathbf{r}) = \int \mathrm{d}\mathbf{r}' G(\mathbf{r},\mathbf{r}')f(\mathbf{r}')$.

References

1. C. Runge, Math. Ann. **46**, 167 (1895)
2. W. Kutta, Zeit. Math. Phys. **46**, 435 (1901)
3. E. Hairer and G. Wanner, Computing **13**, 1 (1974)
4. D. Kreimer, Adv. Th. Math. Phys. **2**, 303 (1998)
5. D.J. Broadhurst and D. Kreimer, J. Symbolic Computation **27**, 581 (1999)
6. D. Kreimer and R. Delbourgo, Phys. Rev. D **60**, 105025 (1999)
7. D. Kreimer, Adv. Th. Math. Phys., hep-th/9901099, **3**, 1999 (in press)
8. D. Kreimer, Commun. Math. Phys. **204**, 669 (1999)
9. T. Krajewski and R Wulkenhaar, Eur. Phys. J. C **7**, 697 (1999)
10. A. Connes and D. Kreimer, Commun. Math. Phys. **199**, 203 (1998)
11. A. Dür, in *Möbius functions, incidence algebras and power series representations* (Springer, Berlin 1986)
12. A. Tucker, in *Applied Combinatorics* (Wiley, New York 1980).
13. S.V. Fomin and N. Lulov, J. Math. Sci. **87**, 4118 (1997)
14. J.C. Butcher, J. Austral. Math. Soc. **3**, 185 (1963)
15. J. C. Butcher, in *The Numerical Analysis of Ordinary Differential Equations* (Wiley, Chichester 1987)
16. E. Hairer, S.P. Nørsett, and G. Wanner, in *Solving Ordinary Differential Equations I* (Springer, Berlin 1993)
17. A. Cayley, Phil. Mag. **13**, 172 (1857)
18. J.C. Butcher, Math. Comput. **26**, 79 (1972)
19. J. Zeng and J. Ramanujan, **3**, 45 (1999)
20. M. Abramowitz and I.A. Stegun, in *Handbook of Mathematical Functions* (Dover, New York 1964)
21. X.-H. Jiang and R. Wong, Stud. Appl. Math. **102**, 375 (1999)
22. G. Barton, in *Elements of Green's Function and Propagation: Potentials, Diffusion and Waves* (Oxford University Press, Oxford 1989)
23. H. Munthe-Kaas, BIT **38**, 92 (1998)
24. B. Owren and A. Marthinsen, BIT **39**, 116 (1999)
25. H. Munthe-Kaas, BIT **35**, 572 (1995)
26. M. Ginocchio, Lett. Math. Phys. **34**, 343 (1995)
27. Y. Komori, T. Mitsui, and H. Sugiura, BIT **37**, 43 (1997)
28. E. Hairer and G. Wanner, in *Solving Ordinary Differential Equations II* (Springer, Berlin 1991)
29. Ch. Brouder, Eur. Phy. J. C **12**, 535 (2000)