Butcher Algebras for Butcher Systems *

Sergey Khashin
Department of Mathematics, Ivanovo State University, Ivanovo, Russia
khash2@gmail.com
Preliminary version

2012.08.01

Abstract

We investigate rigorously the properties of the Butcher upper and lower algebras introduced earlier. This investigation provides a new representation of the order conditions which leads to a new approach to simplifying conditions and a way to obtain new methods of high orders explicitly.

1 Introduction

The Runge-Kutta (RK) methods are one of the most popular numerical methods for the solution of ordinary differential equations.

Initially introduced in the beginning of 20th century, the methods have been defined by a system of equations, which have been hard to obtain explicitly. In the beginning of 1960s, Butcher [Butcher(1963), Butcher(1964a), Butcher(1964b)] suggested a convenient form (Butcher's systems) for these systems of equations, the idea which became a foundation of the modern RK theory. A good exposition can be found in [Butcher(2008), Butcher(2011), Hairer(2000)].

Researchers working on RK methods have shown dynamically changing approaches to solve the order conditions in systematic ways over a number of decades, for example, [Albrecht(1985), Butcher(2008), Hairer(2000), Verner(1996), Feagin (2007)]. Algorithms have been coded to compute coefficients of various methods by allowing arbitrary parameters to be selected for the optimization of accuracy and stability properties. Several classifications of designs for solutions of RK methods and pairs of methods are known, and these classifications also identify differences between families of methods. For example, a 17 stage method of order 10 was developed by Hairer in 1978 [Hairer(1978)], and more recent research by Terry Feagin [Feagin (2007)] on pairs of methods of orders up to 14 indicate that the construction of high order explicit RK methods are of interest.

In [Khashin(2009)] using the filtrations and the graded algebras constructions, we introduced the idea of Butcher upper and lower algebras. These structures have not been artificially imposed on RK, but rather have been "noticed", in other words these are natural structures for RK methods.

In the present paper we prove some important properties for the constructed Butcher algebras. The most important result contains in Theorems 5.6 and 5.7.

In particular, Theorems 5.6 and 5.7 imply a way to significantly reduce the number of equations. Namely, condition (5) means that RK method exists if certain linear equality holds for every vector in certain vector sub-space, which implies that this linear equality must hold for each of the basis vectors

*(Preliminary version)

from this vector sub-space. Therefore, the total number of equations to solve will be near to the dimension of the subspace.

The previous paper [Khashin(2009)] indicates that dimensions of some sub-spaces are less than the number of stages for displayed methods from orders 6 to 9.

The paper is organized as follows. Sec. 2 contains basic notions on RK methods and tree theory. Sec. 3 describes the first preliminary modifications of the Butcher equations. In Sec. 4 we outline the construction of Butcher lower and upper algebras and other algebraic constructions already presented in [Khashin(2009)]. Sec. 5 contains the main contribution of the present paper.

2 Preliminaries

Let an s-stage RK methods be defined by its *Butcher tableau*

$$\begin{array}{c|ccccc} c_2 & a_{21} & & & & \\ c_3 & a_{31} & a_{32} & & & \\ & \dots & & & & \\ c_s & a_{s1} & a_{s2} & \dots & a_{s,s-1} & \\ \hline & b_1 & b_2 & \dots & b_{s-1} & b_s \end{array} \tag{1}$$

Then the corresponding order conditions (*Butcher equations*) for a RK method [Butcher(2008), Hairer(2000)] of order p are as follows.

$$\sum_{j=1}^{s} b_j \Phi_{tj}(A) = b^T \Phi_t(A) = (b, \Phi_t(A)) = 1/\gamma(t) \ , \tag{2}$$

where t is an arbitrary tree of order $\rho(t) \leq p$, and $\Phi_t(A)$ is a homogeneous polynomial of degree $(w(t)-1)$ of coefficients from the matrix A.

Below we introduce several standard for Rooted Trees theory definitions and constructions.

Definition 2.1. We denote the set of all non-isomorphic rooted trees as $\mathcal{T}$.

Definition 2.2. The product of two trees t_1 and t_2, $t_1 \cdot t_2$ is the tree obtained by superimposing the roots of these trees. In other words, the root of tree t_1 and the root of tree t_2 become one point, one root. The set $\mathcal{T}$ is a commutative *semigroup* with respect to this operation.

Let t_0 be the tree consisting of only one vertex. Then for every tree t we have:

$$t \cdot t_0 = t_0 \cdot t = t \ .$$

So, semigroup $\mathcal{T}$ has a identity, t_0. Semigroup $\mathcal{T}$ is generated by all one-legged trees, that is by those elements of $\mathcal{T}$ that have only one edge coming out from the root.

Definition 2.3. The following operation, α acts on semigroup $\mathcal{T}$: for every $t \in \mathcal{T}$

$$\alpha : \ t \mapsto \alpha t \ ,$$

where αt is the tree obtained from the tree t by adding an vertex "under" the root and this new tree has this newly added vertex as the root.

The tree αt is always one-legged.

Theorem 2.4. *Every tree $t \in \mathcal{T}$ can be obtained from t_0 by combination of operations α and multiplication of trees.*

Proof. If t has multiple legs then t is the product of the corresponding one-legged trees. If t has only one leg, then t is the result of α-operation applied to the tree of weight $(w(t)-1)$ obtained when the root of t is deleted. □

Definition 2.5. Let $t \in \mathcal{T}$. The weight $w(t)$ of t is the number of edges in tree t.

The following statements follow from Definitions 2 and 4.

Proposition 2.6. *The following properties hold:*

1. $w(t_0) = 0$,
2. $w(t_1 \cdot t_2) = w(t_1) + w(t_2)$ *for any* $t_1, t_2 \in \mathcal{T}$,
3. $w(\alpha t) = w(t) + 1$ *for any* $t \in \mathcal{T}$.

Definition 2.7. If t is a rooted tree with the root at the bottom and v is a vertex of t, then we denote by t_v the subtree of the tree t which lays over the vertex v and has v as the root.

3 Preliminary Modifications of Butcher Equations

Definition 3.1. Let $t \in \mathcal{T}$. Then $\delta(t)$ is the product of all orders $(w(t_v)+1)$, where v denotes a vertex of t and v is not the root:

$$\delta(t) = \prod_{v \neq \text{root}} (w(t_v) + 1) \ .$$

This definition is a modification of the Butcher's $\gamma(t)$ [Butcher(2008)]. $\delta(t)$ will be more convenient for our purposes. One can see that

$$\gamma(t) = \delta(t)(w(t) + 1) \ .$$

The main advantage of $\delta(t)$ over $\gamma(t)$ is that it is a multiplicative function:

Proposition 3.2. *The following properties hold.*

1. $\delta(t_0) = 1$,
2. $\delta(t_1 \cdot t_2) = \delta(t_1)\delta(t_2)$ *for any* $t_1, t_2 \in \mathcal{T}$,
3. $\delta(\alpha t) = \delta(t)(w(t) + 1)$ *for any* $t \in \mathcal{T}$.

The proof of this proposition follows from Definitions 4 and 6.

Consider a vector space $\mathbb{R}^s$, where vectors are considered as columns. Let $*$ be the coordinate-wise multiplication in $\mathbb{R}^s$. Let

$$e = \begin{pmatrix} 1 \\ \vdots \\ 1 \\ 1 \end{pmatrix} \ , \ d = \begin{pmatrix} 0 \\ \vdots \\ 0 \\ 1 \end{pmatrix} \ .$$

Definition 3.3 (Butcher)**.** Let $t \in \mathcal{T}$ and A be a square $s \times s$ matrix. Define vector $\Phi_t(A)$ recursively.

1. $\Phi_{t_0}(A) = e$ for identity tree t_0,
2. $\Phi_{\alpha t}(A) = A(\Phi_t(A))$ for every $t \in \mathcal{T}$,
3. $\Phi_{t_1 \cdot t_2}(A) = \Phi_{t_1}(A) * \Phi_{t_2}(A)$ for the product of two trees t_1 and t_2.

Let A and $b = (b_1, \dots b_s)$ be the matrix and the row-vector defining an s-stage RK method of order p,

$$A = \begin{pmatrix} 0 & 0 & \dots & 0 \\ a_{21} & 0 & \dots & 0 \\ & & \dots & \\ a_{s1} & a_{s2} & \dots & 0 \end{pmatrix}.$$

As in [Khashin(2009)] we consider instead of the pair (A, b), the extended matrix $\widetilde{A}$, which is matrix A with an extra row b at the bottom, and an extra column of zeros on the right to make it square.

Theorem 3.4. *[Butcher(1963), Butcher(1964b)] A method is of order p if and only if*

$$(b, \Phi_t(A)) = \frac{1}{\gamma(t)}$$

for all trees t with $0 \le w(t) < p$.

Let us rewrite Butcher equations in a different form using the notion of the extended matrix, $\widetilde{A}$.

Theorem 3.5. *An extended matrix $\widetilde{A}$ defines a RK method of order p if and only if*

$$(d, \Phi_t(\widetilde{A})) = \frac{1}{\delta(t)} \tag{3}$$

holds for every tree t of the weight less or equal to p. Moreover, it is enough to consider one-legged trees only.

Proof. Since sum of the coordinates of vector $b * \Phi_t(A)$ equals to the scalar product $(d, \widetilde{A} \cdot \Phi_t(\widetilde{A}))$, where "$*$" denotes coordinate-wise multiplication of vectors, then Butcher equations can be written in the form

$$(d, \widetilde{A} \cdot \Phi_t(\widetilde{A})) = 1/\gamma(t) \tag{4}$$

for all trees of weight $w(t)$, $0 \le w(t) < p$.

Let t be an arbitrary tree of the weight less than p, then $t_1 = \alpha t$ is an one-legged tree of the weight less or equal to p. Since $(d, \widetilde{A} \cdot \Phi_t(\widetilde{A})) = (d, \Phi_{\alpha t}(\widetilde{A}))$ and $\delta(\alpha t) = \gamma(t)$, then equation (4) can be rewritten in the form

$$(d, \Phi_{t_1}(\widetilde{A})) = 1/\delta(t_1)$$

for each one-legged tree t_1 of the weight less than or equal to p.

The equations in this form are multiplicative, because $\Phi_{t_2 \cdot t_3}(\widetilde{A}) = \Phi_{t_2}(\widetilde{A}) * \Phi_{t_3}(\widetilde{A})$ and $\delta_{t_2 \cdot t_3} = \delta_{t_2}\delta_{t_3}$, then the equations for multi-legged trees are the consequences of those for one-legged equations. □

4 Butcher algebras

In this section we recall some algebraic constructions introduced in [Khashin(2009)].

Let A be an *arbitrary* $n \times n$ lower triangular matrix with zero diagonal. Consider subspaces $L_k = < \Phi_t(A) >$ of $\mathbb{R}^n$ where t is a tree of weight k. And filtration [1] of the space $\mathbb{R}^n$ for every given matrix A: $M_k = \sum_{i=0}^{k} L_i$.

[1] The definitions of filtrations and of adjoint graded algebras are standard and can be found for example in [Lang, Serge (2002)], p.172, [Levin A.(2008)], p.37.

Definition 4.1. We say that the adjoint algebra corresponding to this filtration,

$$B(A) = \bigoplus_{k=0}^{n} B_k(A) = \bigoplus_{k=0}^{n} M_k/M_{k-1}$$

is an *upper Butcher algebra* of matrix A.

Definition 4.2. For an arbitrary tree t denote by $\Phi'(t)(A)$ vector

$$\Phi'_t(A) = \delta(t)\Phi_t(A) - \underbrace{Ae * \cdots * Ae}_{d} ,$$

where $d = w(t)$ is the weight of the tree. For $t = t_0,\ d = 0$ and $\Phi'(t_0) = 0$.

Definition 4.3. For a given matrix A consider subspaces L'_k, $k = 0, 1, \ldots$ generated by vectors $\Phi'_t(A)$ for all trees t of weight k.

Consider another filtration of the space $\underline{\mathbb{R}}^n$, by $M'_k = \sum_{i=0}^{k} L'_k$.

Definition 4.4. We say that the adjoint algebra corresponding to this filtration,

$$B'(A) = \bigoplus_{k=0}^{n} B'_k(A) = \bigoplus_{k=0}^{n} M'_k/M'_{k-1}$$

is a *lower Butcher algebra.*

Both algebras B and B' are commutative, associative, graduated, and finite dimensional.

By construction the action of multiplication by a matrix A is defined on the both algebras. Multiplication by a vector Ae is also defined in both algebras: in algebra B it is simply multiplication in B, while in B' it is an additional operator.

By construction, $L'_0(A) = L'_1(A) = 0$, and so the 0-th and 1-st components of the lower Butcher algebra have dimension zero, and $L'_2(A) =< 2A(A(e)) - Ae * Ae >$, so its dimension is 1.

5 Main Result: Alternative Forms of Butcher Equations

The following theorem shows that introduced subspaces L'_k can be defined efficiently via recursion. Indeed, if we consider L'_k using the definition then the number of generators will be equal to the number of trees of weight k, which is large, while the theorem below provides a more economical.

Theorem 5.1. *Let A be an arbitrary lower triangular matrix with zero diagonal. Subspaces L'_k can be defined recursively as follows:*

$$L'_k = \left(Ae * L'_{k-1}\right) + A(L'_{k-1}) + \sum_{i+j=k} L'_i * L'_j \ .$$

Lemma 5.2. *Let $t_1 = \alpha t_0$ be a tree with one edge. Then*

1. $w(t_1^k) = k$,
2. $\delta(t_1^k) = 1,$
3. $w(\alpha(t_1^k)) = k + 1,$
4. $\delta(\alpha(t_1^k)) = k + 1,$

5. $\Phi_{t_1^k}(A) = \underbrace{Ae * \cdots * Ae}_{k}$,

6. $\Phi'_{t_1^k}(A) = 0$,

7. $\Phi_{\alpha(t_1^k)}(A) = A(\underbrace{Ae * \cdots * Ae}_{k})$,

8. $\Phi'_{\alpha(t_1^k)}(A) = (k+1)A(\underbrace{Ae * \cdots * Ae}_{k}) - \underbrace{Ae * \cdots * Ae}_{k+1}$.

Proof. Direct computation. □

Lemma 5.3. *Let t be an arbitrary tree of weight $d = w(t)$. Then*

$$\Phi'_{\alpha(t)}(A) = (d+1)A(\Phi'_t(A)) + \Phi'_{\alpha(t_1^d)}(A) \ .$$

Proof.

$$w(\alpha(t)) = d+1, \quad \delta(\alpha(t)) = (d+1)\delta(t) \ ,$$

$$\Phi'_{\alpha(t)} = (d+1)\delta(t)A(\Phi_t) - \underbrace{Ae * \cdots * Ae}_{d+1} \ ,$$

From Definition 9, multiplication by $(d+1)A$ leads to

$$(d+1)A(\Phi'_t) = (d+1)\delta(t)A\Phi_t - (d+1)A(\underbrace{Ae * \cdots * Ae}_{d}) \ ,$$

and subtraction yelds

$$\Phi'_{\alpha(t)} - (d+1)A(\Phi'_t) = (d+1)A(\underbrace{Ae * \cdots * Ae}_{d}) - \underbrace{Ae * \cdots * Ae}_{d+1} \ .$$

q.e.d. □

Lemma 5.4. *Let t_2, t_3 be arbitrary trees of weights $d_2 = w(t_2)$, $d_3 = w(t_3)$. Then*

$$\Phi'_{t_2 \cdot t_3}(A) = \Phi'_{t_2}(A) * \Phi'_{t_3}(A) + \Phi'_{t_2}(A) * \underbrace{Ae * \cdots * Ae}_{d_3} + \Phi'_{t_3}(A) * \underbrace{Ae * \cdots * Ae}_{d_2} \ .$$

Proof.

$$w(t_2 \cdot t_3) = d_2 + d_3, \quad \delta(t_2 \cdot t_3) = \delta(t_2)\delta(t_3) \ ,$$

$$\Phi'_{t_2} = \delta(t_2)\Phi_{t_2} - \underbrace{Ae * \cdots * Ae}_{d_2} \ ,$$

$$\Phi'_{t_3} = \delta(t_3)\Phi_{t_3} - \underbrace{Ae * \cdots * Ae}_{d_3} \ ,$$

$$\Phi'_{t_2 \cdot t_3} = \delta(t_2)\delta(t_3)\Phi_{t_2} * \Phi_{t_3} - \underbrace{Ae * \cdots * Ae}_{d_2+d_3} \ ,$$

$$\Phi'_{t_2 \cdot t_3} - \Phi'_{t_2} * \Phi'_{t_3} =$$

$$= \delta(t_2)\Phi_{t_2} * \underbrace{Ae * \cdots * Ae}_{d_3} + \delta(t_3)\Phi_{t_3} * \underbrace{Ae * \cdots * Ae}_{d_2} - 2\underbrace{Ae * \cdots * Ae}_{d_2+d_3}$$

or

$$\Phi'_{t_2 \cdot t_3} - \Phi'_{t_2} * \Phi'_{t_3} - \Phi'_{t_2} * \underbrace{Ae * \cdots * Ae}_{d_3} - \Phi'_{t_3} * \underbrace{Ae * \cdots * Ae}_{d_2} = 0$$

q.e.d. □ *Proof.* of the theorem. Observe Lemma 2 and Lemma 3 shows $A(L'_k) \subset L'_{k+1}$ and Lemma 3 shows that $L'_{k_1} * L'_{k_2} \subset L'_{k_1+k_2}$, when $k_1 > 1$, $k_2 > 1$. In addition for any tree t of weight k:

$$Ae * \Phi'_t(A) = \delta(t) Ae * \Phi_t(A) - \underbrace{Ae * \cdots * Ae}_{k+1} = \Phi'_{t_1 \cdot t}(A) \ ,$$

where $t_1 = \alpha t_0$ be a tree with one edge, shows that $Ae * L'_k \subset L'_{k+1}$. By applying this to the result of Lemma 3, we find that the expression for $\Phi_{t_2 \cdot t_3}$ shows that $L'_{k_1} * L'_{k_2} \subset L'_k$ when $k_1 + k_2 = k$. By rewriting the result of Lemma 2 for a tree with weight $w(t) = k$, we find $A(L'_k) \subset L'_{k+1}$.

□

Algebraic structures constructed in [Khashin(2009)] are possible if the considered filtrations agree with $*$-multiplication in $\mathbb{R}^n$ and with operator A. This fact for the filtration associated with the upper Butcher algebra is almost obvious, while the same fact for the one associated with the lower Butcher algebra is the fact not trivial, but it is now a corollary of the previous theorem.

Corollary 5.5. *Subspaces L'_k agree with the multiplication in $\mathbb{R}^n$ and with operator A, that is*

1. $Ae * L'_k \subset L'_{k+1}$,
2. $A(L'_k) \subset L'_{k+1}$,
3. $L'_{k_1} * L'_{k_2} \subset L'_{k_1+k_2}$,

Theorem 5.1 and Corollary 5.5 indicate that the algebraic structures constructed in [Khashin(2009)] are natural for Butcher equations.

The following theorems, Theorem 5.6 and 5.7 constitute the most important result of the paper. Theorem 5.6 gives a significantly simplified form for Butcher equations. Note that (d, v) equals to the last coordinate of vector v, and (5) is a linear system which has all vectors $v \in L_k(\widetilde{A})$ as solutions. Therefore, it is enough to verify equality (5) for the basis vectors of $L_k(\widetilde{A})$ only.

Theorem 5.6. *(Butcher equations in terms of spaces L_k) An extended matrix $\widetilde{A}$ of size $(s+1)\times(s+1)$ defines a RK method of order p if and only if*

$$(d, \widetilde{A}v) = \frac{(d, v)}{k+1} \tag{5}$$

holds for all k, $0 \le k < p$ and for all $v \in L_k(\widetilde{A})$.

Proof. Let matrix $\widetilde{A}$ define a RK method of order p, that is $(d, \Phi_t(\widetilde{A})) = 1/\delta(t)$ holds for every tree t of weight $w(t)$, $0 \le w(t) < p$. To prove that (5) holds, it is enough to verify that for every tree t of some weight k

$$(d, \widetilde{A} \cdot \Phi_t(\widetilde{A})) = \frac{(d, \Phi_t(\widetilde{A}))}{k+1} \ ,$$

which follows from $\delta(\alpha t) = \delta(t)(w(t) + 1)$.

Vise versa, let (5) hold. Then for $k = 0$ we have

$$(d, \widetilde{A}e) = (d, e) = 1$$

that is condition (5) holds for the trees of weight 0. If for some tree t we have

$$(d, \Phi_t(\widetilde{A})) = \frac{1}{\delta(t)} \ ,$$

then condition (5) implies

$$(d, \widetilde{A} \cdot \Phi_t(\widetilde{A})) = \frac{(d, \Phi_t(\widetilde{A}))}{k+1} ,$$

which gives an inductive proof for the theorem. □

The following Theorem 5.7 gives yet another form for Butcher equations. It is more convenient for further solution.

Theorem 5.7. *An extended matrix $\widetilde{A}$ defines a RK method of order p if and only if*

$$\begin{array}{lll} 1) & (d, \widetilde{A}^k e) = 1/k!, & \text{for} \quad k \ = 0, \ldots, p \ , \\ 2) & \forall v \in L'_k : (d, v) = 0, & \text{for} \quad k \ < p \ . \end{array} \tag{6}$$

Proof. Prove the equivalence of the conditions to the conditions of Theorem 5.6. Let matrix $\widetilde{A}$ define a RK method of order p, that is for all $v \in L_k$, where $k < p$

$$(d, \widetilde{A}v) = (d, v)/(k+1) .$$

Since $e \in L_0$ and $(d, e) = 1$, then one can obtain consequently

$$\begin{array}{lllll} (d, \widetilde{A}e) & = & (d, e)/1 & = & 1 \ , \\ (d, \widetilde{A}^2 e) & = & (d, \widetilde{A}e)/2 & = & 1/2 \ , \\ \ldots & & & & \\ (d, \widetilde{A}^p e) & = & (d, \widetilde{A}^{p-1} e)/p & = & 1/p! \ . \end{array}$$

If $v \in L'_k(\widetilde{A})$, that is $v \in L_k(\widetilde{A})$ and $(d, v) = 0$, then from (5) one obtains $(d, \widetilde{A}v) = (d, v)/(k+1) = 0$, which implies $\widetilde{A}v \in L'_{k+1}$.

Vice versa, let (6) hold and $v \in L_k$. According to Theorem 5.6, we have to prove that $(d, \widetilde{A}v) = (d, v)/(k+1)$ for all $k < p$ and for all $v \in L_k(\widetilde{A})$. It is trivial for $k = 0$. Since $L_1(\widetilde{A}) =< Ae >$, this holds for $k = 1$. Now, let $1 < k < p$. Since $(d, \widetilde{A}^k e) = 1/k!$, we have

$$v = k! \cdot (d, v) \cdot \widetilde{A}^k e + w$$

for some vector $w \in L'_k$. Therefore,

$$(d, \widetilde{A}v) = (d, \widetilde{A}(k!(d, v) \cdot \widetilde{A}^k e)) + (d, \widetilde{A}w) =$$

$$= k!(d, v)(d, \widetilde{A}^{k+1} e) = (d, v)/(k+1) .$$

□ The definition below gives an idea of the classification of the RK methods following from the constructions above. Note that this classification is indeed very natural.

Definition 5.8. Let $\widetilde{A}$ is an extended Butcher matrix of some RK method. Let

$$r_i = r_i(\widetilde{A}) = \dim B'_i(\widetilde{A}) .$$

We say that the sequence $(r_1, r_2, \ldots, r_{s+1})$ is the type of the method.

References

[Albrecht(1985)] Albrecht P., The theory of A-methods, Numerische Mathematik 47 (1985), pp. 59-87.

[Butcher(1963)] Butcher J. Coefficients for the study of Runge-Kutta integration processes. J Austral Math Soc 3: 185–201, (1963).

[Butcher(1964a)] Butcher J. Implicit Runge–Kutta processes. Math. Comp. 18:50–64, (1964).

[Butcher(1964b)] Butcher J. On Runge–Kutta processes of high order. J. Austral Math Soc 4: 179–194, (1964).

[Butcher(2008)] Butcher J.C. Numerical methods for ordinary differential equations (2nd ed.). John Wiley & Sons, (2008).

[Butcher(2011)] Butcher J.C., On fifth and sixth order explicit Runge-Kutta methods: order conditions and order barriers. Can.Appl.Math.Q.17, 433445 (2009).

[Feagin (2007)] Feagin T., A tenth-order Runge-Kutta method with error estimate, Proceedings of the IAENG Conf. on Scientific Computing, (2007).

[Hairer(1978)] Hairer J., A Runge-Kutta method of order 10, J.Inst.Math.Applics, vol.21, 47–59, (1978).

[Hairer(2000)] Hairer J. Nørsett S.P. Solving ordinary differential equations I. Nonstiff Problems. 2Ed. Springer-Verlag, (2000).

[Khashin(2009)] Khashin S. (2009) A symbolic-numeric approach to the solution of the Butcher Equations. Canadian Appl.Math.Q.17(3), 555-569 (2009).

[Lang, Serge (2002)] Lang S., Algebra, Graduate Texts in Mathematics, 211 (Revised third ed.), New York: Springer-Verlag, ISBN 978-0-387-95385-4, MR1878556, (2002).

[Levin A.(2008)] Levin A., Difference algebra, Springer, ISBN 978-1-4020-6946-8, (2008).

[Verner(1996)] Verner J. High-order explicit Runge–Kutta pairs with low stage order. Appl Numer Math 22 (1-3), 345–357, (1996).