BIT Numerical Mathematics **43:** 723–742, 2003.

COST EFFICIENT LIE GROUP INTEGRATORS IN THE RKMK CLASS*

F. CASAS[1] and B. OWREN[2]

[1] *Departament de Matemàtiques, Universitat Jaume I, 12071-Castellón, Spain. email: Fernando.Casas@uji.es*

[2] *Department of Mathematical Sciences, NTNU, N-7491 Trondheim, Norway, and Center for Advanced Study, Drammensveien 78, 0271 Oslo, Norway. email: Brynjulf.Owren@math.ntnu.no*

Abstract.

In this work a systematic procedure is implemented in order to minimise the computational cost of the Runge–Kutta–Munthe-Kaas (RKMK) class of Lie-group solvers. The process consists of the application of a linear transformation to the stages of the method and the analysis of a graded free Lie algebra to reduce the number of commutators involved. We consider here RKMK integration methods up to order seven based on some of the most popular Runge–Kutta schemes.

AMS subject classification (2000): 65L05.

Key words: time integration, geometric integration, numerical integration of ordinary differential equations on manifolds, numerical analysis, Lie algebras, Lie groups.

1 Introduction.

The integration schemes of Munthe-Kaas [9] can be applied to differential equations on homogeneous manifolds

$$\dot{y} = f(y) \cdot y, \qquad y(0) = y_0 \in \mathcal{M}. \tag{1.1}$$

Generally, (1.1) is induced by a transitive action by a Lie group G on $\mathcal{M}$, so that $f : \mathcal{M} \to \mathfrak{g}$ is a map from the manifold to the Lie algebra $\mathfrak{g}$ of G. The product $Z \cdot m$, $Z \in \mathfrak{g}$, $m \in \mathcal{M}$ is then understood as

$$Z \cdot m = \left.\frac{\mathrm{d}}{\mathrm{d}t}\right|_{t=0} \exp(tZ) \cdot m.$$

A special case is when G is a subgroup of $GL(n)$, the Lie group of invertible $n \times n$ matrices and $\mathcal{M} = G$. Of particular interest in applications is the group $G = SO(n)$, the set of $n \times n$ orthogonal matrices. The corresponding Lie algebra $\mathfrak{g} = \mathfrak{so}(n)$ is the set of $n \times n$ skew-symmetric matrices, see also Iserles *et al.* [7] for more details.

* Submitted December 2002. Accepted September 2003. Communicated by Christian Lubich.

One may solve (1.1) by transforming the differential equation from $\mathcal{M}$ to $\mathfrak{g}$. This was achieved by setting $y(t) = \exp(\sigma(t))\cdot y_0$ with $\sigma(0) = 0$ in a neighborhood of $y_0 \in \mathcal{M}$ and then derive a differential equation for $\sigma(t)$. It is now well-known that one gets

$$\dot{\sigma} = d\exp_{\sigma}^{-1}(f(\exp(\sigma)\cdot y_0)). \tag{1.2}$$

The map $d\exp_u^{-1} : \mathfrak{g} \to \mathfrak{g}$ is expressed in terms of commutators through the infinite series

$$d\exp_u^{-1}(v) = \sum_{k=0}^{\infty} \frac{B_k}{k!}\mathrm{ad}_u^k v = v - \frac{1}{2}[u,v] + \frac{1}{12}[u,[u,v]] + \cdots, \tag{1.3}$$

where $\mathrm{ad}_u^0 v = v$, $\mathrm{ad}_u^k v = [u, \mathrm{ad}_u^{k-1} v]$ and B_k are the Bernoulli numbers. The idea of Munthe-Kaas was to approximate the solution of (1.2) by means of a classical Runge–Kutta method, and then transform the result $\sigma_1 \approx \sigma(h)$ back to $\mathcal{M}$ by setting $y_1 = \exp(\sigma_1)\cdot y_0$. The series (1.3) can be truncated because it is used only in cases where the argument $u = \mathcal{O}(h)$. Using an explicit pth order Runge–Kutta method with s stages, one can write the corresponding Munthe-Kaas (RKMK) Lie group scheme in the following form, where h is the stepsize and a_{ij}, b_i are the Runge–Kutta parameters.

ALGORITHM 1.1.

for $i = 1 : s$ ***do***
  $u_i = \sum_{j=1}^{i-1} a_{ij}\ \texttt{dexpinv}(u_j, k_j, p-1)$
  $k_i = hf(\exp(u_i)\cdot y_0)$
end do
$v = \sum_i b_i\ \texttt{dexpinv}(u_i, k_i, p)$
$y_1 = \exp(v)\cdot y_0$

Here $\texttt{dexpinv}(u, v, p)$ denotes a pth order approximation to $d\exp_u^{-1}(v)$, i.e. $d\exp_{tu}^{-1}(v) - \texttt{dexpinv}(tu, v, p) = \mathcal{O}(t^{p+1})$ for all $u, v \in \mathfrak{g}$ at $t = 0$. In [9] it was suggested that this approximation is obtained simply by truncating the series (1.3). Keeping in mind that the Bernoulli numbers with odd indices (except the first) vanish, one finds that the number of commutators to be computed in each step is $(p-2)(s-1)$ for p even and $(p-1)(s-1)$ for p odd. However, it was noted in [10] that substantial savings can be made for low order methods, by applying a linear transformation to the stages k_i in the algorithm. One introduces transformed variables

$$Q_i = \sum_{j=1}^{i} V_{i,j} k_j = \mathcal{O}(h^{q_i}), \quad i = 1, \ldots, s, \tag{1.4}$$

where the constants V_{ij} are chosen such that the resultant integers q_i are as large as possible. Then, it is evident that commutators like

$$[Q_{i_1}, [Q_{i_2}, \ldots, [Q_{i_{m-1}}, Q_{i_m}]\ldots]] = \mathcal{O}(h^{q_{i_1}+\cdots+q_{i_m}})$$

which makes it easy to discard terms of order higher than the method itself. The discussion of complexity in [10] is just based on counting the number of commutators N_2 whose order does not exceed that of the method. Their treatment depends on the particular Runge–Kutta method which is used, but for some of the most popular schemes we can deduce the following table, where *Orig* indicates the number of commutators from the schemes of [9], whereas the column *FLA* is what one obtains from approach of [10].

Method	p	s	Orig	FLA
RK4	4	4	6	4
DOPRI5	5	6	20	12
Butcher6	6	7	24	26
Butcher7	7	9	48	60
RKF78	8	13	72	133

It is important to note that the numbers in the *FLA* column do not count the actual number of commutators that must be computed. For instance the expression $[Q_1, Q_2] + [Q_1, Q_3]$ would be counted as two commutators whereas in this case one only needs to compute the single commutator $[Q_1, Q_2 + Q_3]$. Still, the rapid increase in this number for higher orders led the authors of [10] to believe that their approach would lead to schemes with a reduced number of commutators only for the low and moderate order cases.

Recently, there has been some progress in reducing the complexity in integration schemes based on the Magnus series expansion. Iserles and Nørsett [8] have studied these methods extensively and also in [10] the question of reducing the number of commutators is addressed. In Celledoni *et al.* [5], the authors have tried to quantify the computational cost associated to various Lie group integrators. However, even more recently Blanes *et al.* [2] have found remarkable savings in computational complexity for Magnus series schemes compared to what was previously known, the optimal number of commutators for orders 4, 6 and 8 being 1, 3 and 6 respectively.

The aim of this paper is to combine the approach of [10] with that of [2] to obtain Lie group integrators in the RKMK class with significant reduction in the number of commutators compared to what is presently known. We start by a case study in Section 2, where we consider methods of order 4 as a special illustration. The treatment of this case exhibits the main ideas, but involves little of the machinery of Butcher series and graded free Lie algebras needed to treat the full general case. In Section 3, we extend and generalize the approach of [10] by using Butcher's order theory to obtain a suitable basis for the graded free Lie algebra associated with the stages of an explicit RKMK method. Then, in section 4 we present an approach for further minimizing the number of commutators.

2 Fourth-order RKMK methods. A case study.

In this section we illustrate the main features of the procedure leading from Algorithm 1.1 to the construction of efficient Lie-group solvers for the nonlinear differential equation (1.1). As stated in the introduction, it consists of three parts: (i) first, one has to find the transformation (1.4); (ii) second, the internal stages u_i and v must be expressed in terms of the new variables Q_j provided by (1.4) and (iii) finally some optimization strategy has to be applied to reduce the total number of commutators involved and thus also the computational complexity of the algorithm.

In this process it is important to recall that, once u_i and v are expressed in terms of Q_m, they can be considered as elements of the graded free Lie algebra generated by $\{Q_i\}_{i\geq 1}$ with grades $w(Q_i) = q_i$, as suggested by (1.4) [10]. Therefore we can apply the optimization technique devised in [2] to write an element of a graded free Lie algebra with the minimum number of commutators. This will be particularly relevant for high order methods.

For simplicity, here we only consider 4th-order schemes and Algorithm 1.1 as applied to Equation (1.1) defined in a matrix Lie group. Then $\exp(v) = \mathrm{e}^v$ is the usual matrix exponential.

LEMMA 2.1. *In the explicit RKMK Algorithm* 1.1 *one has for* $i \geq 1$

$$k_i = hf(y_0) + c_i h^2 f'(y_0) f(y_0) y_0 + \mathcal{O}(h^3), \tag{2.1}$$

$$u_i = c_i h f(y_0) + \mathcal{O}(h^2) \tag{2.2}$$

with $c_i = \sum_{j=1}^{i-1} a_{ij}$.

PROOF. We use induction on the stage index. Let us denote $\tilde{k}_i \equiv d\exp_{u_i}^{-1}(k_i)$. It is clear that $u_1 = 0$, $\tilde{k}_1 = k_1 = hf(y_0)$ and therefore $u_2 = a_{21}k_1 = c_2 h f(y_0)$. In general, suppose (2.2) is true for $i = l \geq 2$. Then, by expanding in Taylor series,

$$\begin{aligned} k_l &= hf(\mathrm{e}^{u_l} y_0) = hf((I + u_l + \cdots)y_0) = hf(y_0) + hf'(y_0)u_l y_0 + \mathcal{O}(h^3) \\ &= hf(y_0) + c_l h^2 f'(y_0) f(y_0) y_0 + \mathcal{O}(h^3). \end{aligned}$$

On the other hand, it is evident that $[u_l, k_l] = \mathcal{O}(h^3)$, so that $\tilde{k}_l = k_l + \mathcal{O}(h^3)$ and finally

$$\begin{aligned} u_{l+1} &= \sum_{j=1}^{l} a_{l+1,j} k_l + \mathcal{O}(h^3) = h \sum_{j=1}^{l} a_{l+1,j} f(y_0) + \mathcal{O}(h^2) \\ &= h c_{l+1} f(y_0) + \mathcal{O}(h^2). \end{aligned}$$

PROPOSITION 2.2. *The linear combination* $Q_i \equiv \sum_{j=1}^{i} V_{i,j} k_j = \mathcal{O}(h^3)$, $i \geq 3$, *if and only if*

$$\sum_{j=1}^{i} V_{i,j} = 0 \quad \textit{and} \quad \sum_{j=2}^{i} V_{i,j} c_j = 0. \tag{2.3}$$

PROOF. From Lemma 2.1 we have

$$\begin{aligned} Q_i &= V_{i,1}hf(y_0) + \sum_{j=2}^{i} V_{i,j}(hf(y_0) + c_j h^2 f'(y_0)f(y_0)y_0 + \mathcal{O}(h^3)) \\ &= \left(\sum_{j=1}^{i} V_{i,j}\right) hf(y_0) + \left(\sum_{j=2}^{i} V_{i,j}c_j\right) h^2 f'(y_0)f(y_0)y_0 + \mathcal{O}(h^3) \end{aligned}$$

from which the result follows. We observe, in particular, that $Q_1 = k_1 = \mathcal{O}(h)$ and $Q_2 = V_{1,2}(-k_1 + k_2) = \mathcal{O}(h^2)$ if $c_2 \neq 0$.

By solving (2.3), the first Q_i elements for any RKMK method are ($V_{i,i} = 1$)

$$\begin{aligned} Q_1 &= k_1 = \mathcal{O}(h), \\ Q_2 &= k_2 - k_1 = \mathcal{O}(h^2), \\ Q_3 &= k_3 - \frac{c_3}{c_2}k_2 + \frac{c_3 - c_2}{c_2}k_1 = \mathcal{O}(h^3), \\ Q_4 &= k_4 + V_{4,3}k_3 - \frac{c_4 + c_3V_{4,3}}{c_2}k_2 - \frac{c_2 - c_4 + (c_2 - c_3)V_{4,3}}{c_2}k_1 = \mathcal{O}(h^3). \end{aligned} \tag{2.4}$$

The next step to build methods of order four with 4 stages is to rephrase Algorithm 1.1 in terms of $Q_1, \ldots, Q_4$ and retain terms up to order $\mathcal{O}(h^3)$ in u_i, $1 \leq i \leq 4$, and up to order $\mathcal{O}(h^4)$ in v. This gives us

$$\begin{aligned} u_1 &= 0, \\ u_2 &= c_2Q_1 \\ u_3 &= c_3Q_1 + a_{32}Q_2 - \frac{1}{2}a_{21}a_{32}[Q_1, Q_2] \\ u_4 &= c_4Q_1 + \left(a_{42} + a_{43}\frac{c_3}{c_2}\right)Q_2 + a_{43}Q_3 - \\ &\quad - \frac{1}{2}\left(a_{21}a_{42} - a_{32}a_{43} + a_{43}\frac{c_3^2}{c_2}\right)[Q_1, Q_2], \\ v &= Q_1 + \frac{1}{2c_2}Q_2 + (b_3 - b_4V_{4,3})Q_3 + b_4Q_4 - \frac{1}{12c_2}[Q_1, Q_2] + \\ &\quad + \frac{1}{2}(-2c_3b_3 + b_3 + b_4V_{4,3})[Q_1, Q_3] - \frac{1}{2}b_4[Q_1, Q_4]. \end{aligned} \tag{2.5}$$

Remarkably, the coefficient of $[Q_1, [Q_1, Q_2]]$ in v vanishes identically so that, in principle, with this formulation 3 commutators have to be computed (instead of 6 in terms of the k_i). In fact, if we denote

$$\begin{aligned} d_1 &= [Q_1, Q_2], \\ d_2 &= [Q_1, (-2c_3b_3 + b_3 + b_4V_{4,3})Q_3 - b_4Q_4] \end{aligned} \tag{2.6}$$

then

$$v = Q_1 + \frac{1}{2c_2}Q_2 + (b_3 - b_4V_{4,3})Q_3 + b_4Q_4 - \frac{1}{12c_2}d_1 + \frac{1}{2}d_2 \tag{2.7}$$

involves only 2 commutators. The final algorithm requiring the minimum number of commutators is as follows:

ALGORITHM 2.1.

for $i = 1 : 4$ ***do***
  u_i *given by* (2.5)
  $k_i = hf(\exp(u_i) \cdot y_0)$
  Q_i *given by* (2.4)
end do
v *given by* (2.7)
$y_1 = \exp(v) \cdot y_0$

REMARKS. (1) We can fix the parameter $V_{4,3} \equiv \alpha$ so that $d_2 = -b_4[Q_1, Q_4]$.
(2) One could consider 4th-order Runge–Kutta methods with more than four stages. Even then the resulting RKMK algorithm in terms of the Q_i requires the computation of only 2 commutators as long as $Q_i = \mathcal{O}(h^3)$ for $i \geq 3$. For in this case u_i, $i = 3, \ldots, s$, involves only $[Q_1, Q_2]$ and the commutators $[Q_1, [Q_1, Q_2]]$, $[Q_1, Q_3]$, $[Q_1, Q_4]$, $[Q_1, Q_5]$, etc., appearing in v can be grouped together.

3 Search for the optimal transformation.

3.1 General considerations.

As it has been clearly established in the preceding section, in the search for optimal schemes it is crucial one can find transformations Q_i of the stages with as high order as possible as in (1.4). In the following we generalize Proposition 2.2 and develop a systematic procedure to obtain this transformation for any classical Runge–Kutta scheme. The following result from [10] is useful for this purpose.

PROPOSITION 3.1. *Suppose that there exists real numbers* $r_1, \ldots, r_s$ *such that*

$$\sum_{j=1}^{s} r_j d\exp_{u_i}^{-1} k_j = \mathcal{O}(h^q)$$

for some integer q*, where* u_i, k_j, h *are as in Algorithm* 1.1*. Then*

$$\sum_{j=1}^{s} r_j k_j = \mathcal{O}(h^q).$$

The result also applies when $\tilde{k}_i := d\exp_{u_i}^{-1}(k_i)$ is replaced by its pth order approximation as in Algorithm 1.1 as long as $q \leq p$. Since $u_i, \tilde{k}_i$ and v are obtained from applying a classical Runge–Kutta method, we can use the Butcher theory [3]. We will here apply this theory by using the notation from [6], known as the

B-series approach. We work in a somewhat simpler setting than the general case, because it is always true that the initial value in (1.2) is zero. The quantities u_i, v as well as the stage derivatives $\tilde{k}_i$ can be formally expressed in a B-series expansion, $u_i = B(\mathbf{u}_i)$, $\tilde{k}_i = B(\tilde{\mathbf{k}}_i)$, $v = B(\mathbf{v})$ where in general B is defined as

$$B(\mathbf{a}) = \sum_{\tau \in T} \frac{h^{|\tau|}}{|\tau|!} \mathbf{a}(\tau) F(\tau)(0). \tag{3.1}$$

Here, T is the set of *rooted trees*, where $\tau \in T$ has $|\tau|$ nodes. The map $\mathbf{a} : T \to \mathbb{R}$ represents the coefficients of the B-series. The elementary differentials $F(\tau)$ depend on the derivatives of the right-hand side of (1.2) evaluated at 0. An element of T is either the one-node tree which we simply denote by 1, or it is formed as $\tau = [\tau_1, \ldots, \tau_\mu]$ where each $\tau_i \in T$. This is the tree obtained by joining the roots of the trees $\tau_1, \ldots, \tau_\mu$ to a new common root. Consequently, $|\tau| = 1 + |\tau_1| + \cdots + |\tau_\mu|$. The coefficients in the B-series for u_i, $\tilde{k}_i$, and v are related as follows:

$$\begin{aligned} &\tilde{\mathbf{k}}_i(1) = 1, \quad \text{and for } \tau = [\tau_1, \ldots, \tau_\mu] \\ &\mathbf{u}_i(\tau) = \sum_{j=1}^{s} a_{ij} \tilde{\mathbf{k}}_j(\tau), \quad \tilde{\mathbf{k}}_i(\tau) = |\tau| \mathbf{u}_i(\tau_1) \cdots \mathbf{u}_i(\tau_\mu), \\ &\mathbf{v}(\tau) = \sum_{j=1}^{s} b_j \tilde{\mathbf{k}}_j(\tau). \end{aligned} \tag{3.2}$$

In view of Proposition 3.1 we will, for each $i = 1, \ldots, s$, search for constants $r_1, \ldots, r_i$ such that

$$\sum_{j=1}^{i} r_j \tilde{k}_j = \mathcal{O}(h^{q_i}),$$

where q_i is as large as possible. We shall always assume in this expression that $r_i = 1$. Thus, for $i = 1$ we get q_1=1. However, for $i > 1$ we obtain from (3.1) the conditions

$$\sum_{j=1}^{i} r_j \tilde{\mathbf{k}}_j(\tau) = 0, \quad \forall \tau \text{ such that } |\tau| \le q_i - 1. \tag{3.3}$$

To proceed, it is instructive to enumerate the trees, increasingly in terms of $|\tau|$ and consider the matrix corresponding to (3.3)

$$\left[\begin{array}{ccccc} 1 & 1 & 1 & \cdots & 1 \\ \hline 0 & c_2 & c_3 & \cdots & c_i \\ \hline 0 & c_2^2 & c_3^2 & \cdots & c_i^2 \\ 0 & 0 & a_{32}c_2 & \cdots & \sum_j a_{ij}c_j \\ \hline \vdots & \vdots & \vdots & & \vdots \\ \hline \vdots & \vdots & \vdots & & \vdots \end{array}\right] \tag{3.4}$$

The horizontal lines are distinguishing the orders. Denoting by N_q the number of conditions for order $\leq q$, one has

q	1	2	3	4	5	6	7	8
N_q	1	2	4	8	17	37	85	200

We let $\mathbf{K}_{i,q}$ be the $N_q \times i$ matrix whose rows correspond to all trees such that $|\tau| \leq q$. For a given set of Runge–Kutta coefficients (a_{ij}), the construction of the transformation (1.4) is not difficult, the following algorithm gives the lower triangular matrix V as well as the grades $(q_1, \ldots, q_s)$.

ALGORITHM 3.1.

$V_{1,1} := 1$
for $i = 2 : s$ ***do***
  $q = 1$
  while $rank(\mathbf{K}_{i,q}) = rank(\mathbf{K}_{i-1,q})$ ***do***
    $q = q + 1$
  end while
  $q_i = q$
  Solve $\mathbf{K}_{i-1,q-1}\mathbf{r} = -\mathbf{s}_i$
  Set $(V_{i,1}, \ldots, V_{i,i-1}) = \mathbf{r}^T$, $V_{i,i} = 1$
end for

where $\mathbf{s}_i$ *is the ith column of* $\mathbf{K}_{i,q}$.

Let us now turn to the general case of an s-stage pth order explicit irreducible Runge–Kutta method. We always scale the transformation by setting $V_{i,i} = 1$ in (1.4).

3.2 Analysis stage by stage.

First stage. The first transformed stage is $Q_1 = k_1 = \mathcal{O}(h)$.

Second stage. Considering the first row of (3.4), we see that $Q_2 = k_2 - k_1 = \mathcal{O}(h^2)$, and no more can be achieved since $c_2 \neq 0$ in irreducible schemes.

Third stage. Considering the first two rows of (3.4) one finds that

$$Q_3 = \frac{c_3 - c_2}{c_2} k_1 - \frac{c_3}{c_2} k_2 + k_3 = \mathcal{O}(h^3).$$

This is the best one can do, because by imposing the conditions for the next order one finds that the third stage will coincide with either the first or the second stage, i.e. a reducible method.

Fourth stage. As we have shown in Section 2, for this stage one obtains a one-parameter family of grade 3 vectors

$$Q_4 = \frac{c_4 - c_2 + \alpha(c_3 - c_2)}{c_2} k_1 - \frac{c_4 + \alpha c_3}{c_2} k_2 + \alpha k_3 + k_4 + \mathcal{O}(h^3).$$

However, to obtain grade 4 it is necessary to force the 4×4 matrix $\mathbf{K}_{4,3}$ to be singular. We define $\phi_i = \sum_j a_{ij} c_j$ and note that $\phi_1 = \phi_2 = 0$. Using $c_2 \neq 0$ we impose

$$\begin{vmatrix} 1 & c_3 & c_4 \\ c_2 & c_3^2 & c_4^2 \\ 0 & \phi_3 & \phi_4 \end{vmatrix} = c_4(c_2 - c_4)\phi_3 + c_3(c_3 - c_2)\phi_4 = 0. \tag{3.5}$$

1. *Methods of order 4 with 4 stages.* Unfortunately, this condition is incompatible with the corresponding RK scheme of 4 stages having order 4. To see this, we note for instance from [6] that for such schemes $c_4 = 1$ and we invoke the two order conditions

$$b_3\phi_3 + b_4\phi_4 = \frac{1}{6}, \qquad b_3c_3\phi_3 + b_4\phi_4 = \frac{1}{8}. \tag{3.6}$$

   Since $\phi_3 = a_{32}c_2$ is nonzero, we must impose from (3.5) the (necessary) condition

$$\begin{vmatrix} b_3 & b_4 & \frac{1}{6} \\ b_3c_3 & b_4 & \frac{1}{8} \\ c_2 - 1 & c_3(c_3 - c_2) & 0 \end{vmatrix} = \frac{1}{24}(c_3(4c_3 - 3)(c_3 - c_2)b_3 + (1 - c_2)b_4) = 0. \tag{3.7}$$

   Finally, we combine the quadrature conditions

$$\sum_i b_i c_i^{q-1} = \frac{1}{q}, \quad q = 2, 3, 4$$

   with (3.7) and we get at last the (necessary) condition

$$\begin{vmatrix} 1 & c_3 & 1 & \frac{1}{2} \\ c_2 & c_3^2 & 1 & \frac{1}{3} \\ c_2^2 & c_3^3 & 1 & \frac{1}{4} \\ 0 & c_3(4c_3 - 3)(c_3 - c_2) & 1 - c_2 & 0 \end{vmatrix} = \frac{1}{6}c_2c_3(1 - c_2)(1 - c_3)(c_2 - c_3) = 0.$$

   There are only three possible cases of coinciding node values [6, p. 138]: $(c_2, c_3) = (1/2, 0), (1/2, 1/2)$ and $(1, 1/2)$. The first two are incompatible with (3.5) and the last one would require $\phi_4 = 0$ and thereby contradict the two conditions (3.6).
2. *Methods of order 5 with 6 stages.* For 5th order methods there are 17 order conditions. Imposing the simplifying assumption $\phi_i = \frac{1}{2}c_i^2$, $i \neq 2$, one finds that either $c_4 = 0$ or $c_4 = c_3$ for the fourth stage to have grade 4. A long and fairly technical analysis shows that these conditions are incompatible with the 17 order conditions for order 5. We have not pursued the question of whether it is possible to obtain 5th order methods with 6 stages such that the fourth stage has grade 4. However, for every known method of order 5 we have looked at, $Q_4 = \mathcal{O}(h^3)$ is the best that can be achieved.

In general, by adding enough stages, one can always achieve $Q_4 = \mathcal{O}(h^4)$, but the extra cost seems not to be compensated by the possible reduction in commutator calculations.

ith stage, $i > 4$. As long as $\mathbf{K}_{4,3}$ is non-singular one can easily achieve grade 3 for each Q_i where $i \geq 3$, in fact, one will always have $i - 3$ free parameters for these transformed stages. To completely characterize the highest attainable grade for subsequent stages seems complicated and perhaps not very useful because one may in this way easily exclude the most popular of the classical Runge–Kutta schemes. However, in [6, pp. 155–156] we can find an explicit procedure for constructing 5th order schemes with 6 stages. The free parameters are c_2, c_3, c_4, c_5 and $a_{42} = \lambda$. It is easy to deduce that by leaving the free parameters unrestricted, the grade sequence will be $(1, 2, 3, 3, 4, 4)$. The transformed stages $Q_1, \dots, Q_4$ must be chosen as described above, whereas

$$Q_5 = k_5 - \frac{c_5(c_5 - c_3)}{c_4(c_4 - c_3)}k_4 + \frac{c_5(c_5 - c_4)}{c_3(c_4 - c_3)}k_3 - \frac{(c_5 - c_3)(c_5 - c_4)}{c_3 c_4} = \mathcal{O}(h^4)$$

and it is impossible to achieve higher grade. For the sixth stage, one can set $Q_6 = \sum_{i=1}^{6} r_i k_i$, where $r_2 = 0$, $r_6 = 1$, and r_1, r_3, r_4, r_5 result from solving the system

$$\sum_{i=1}^{5} r_i c_i^{q-1} = -1, \quad q = 1, 2, 3, 4.$$

Then Q_6 has grade 4. But demanding that

$$a_{4,2} = \lambda = \frac{1}{8}\frac{c_3^2(15c_3^2 - 12c_3 + 2)}{c_2(5c_3^2 - 4c_3 + 1)^3} \quad \text{and} \quad c_4 = \frac{1}{2}\frac{c_3}{5c_3^2 - 4c_3 + 1}$$

one obtains $Q_6 = \mathcal{O}(h^5)$. The Runge–Kutta–Fehlberg method of order 5 presented below does not fall within this class because $c_6 = \frac{1}{2} \neq 1$, but the Dormand–Prince method does.

3.3 Some examples of high order methods.

We present here some of the most used explicit Runge–Kutta methods/pairs and give for each of them the grade sequence $(q_1, \dots, q_s)$ so that $Q_i = \sum_{j=1}^{i} V_{i,j} k_j = \mathcal{O}(h^{q_i})$. The corresponding transformation matrix V can be found in the reference [4].

5th-order

RKF45, Runge–Kutta–Fehlberg pair. The coefficients can be found in [6, p. 177]. The best possible sequence of grades for this method is $(1, 2, 3, 3, 4, 5)$.

DOPRI5(4), a method of Dormand and Prince. The coefficients of this widely used 7-stage pair can be found also in [6, p. 178]. In this case the grade sequence is $(1, 2, 3, 3, 4, 4, 5)$.

6th-order

Butcher6, a 6th-order method of Butcher. The coefficients of this method are found for instance by choosing the abscissae $c_2 = \frac{1}{2}$, $c_3 = \frac{2}{3}$, $c_5 = \frac{5}{6}$ and $c_6 = \frac{1}{6}$, and then follow the recipe in Butcher [3, pp. 200–204]. The grade sequence is $(1, 2, 3, 3, 4, 5, 4)$.

The 6(5) pair of Verner (DVERK). The coefficients of this embedded formula are given in [6, p. 181]. Now the optimal grade sequence is $(1, 2, 3, 3, 4, 4, 5, 6)$ and the transformation matrix is

$$V = \begin{bmatrix} 1 \\ -1 & 1 \\ \frac{3}{5} & -\frac{8}{5} & 1 \\ 3 & -4 & 0 & 1 \\ -\frac{17}{32} & 0 & \frac{125}{96} & -\frac{85}{48} & 1 \\ -\frac{11}{8} & 0 & \frac{25}{8} & -\frac{11}{4} & 0 & 1 \\ -\frac{621}{1000} & 0 & -\frac{69}{136} & \frac{23}{100} & -\frac{216}{2125} & 0 & 1 \\ -\frac{77}{860} & 0 & -\frac{875}{8772} & \frac{77}{774} & -\frac{7392}{84065} & -\frac{42}{43} & \frac{1375}{8901} & 1 \end{bmatrix} \tag{3.8}$$

7th-order

Butcher7, a 7th order method of Butcher. In [3, p. 207] one can find the coefficients of a 7th order scheme with 9 stages.

Runge–Kutta–Fehlberg 7(8). The coefficients of this 7th order formula with 8th order error estimate, which is of frequent use in high precision computations, are reproduced in [6, p. 180]. The grade sequence is $(1, 2, 3, 3, 4, 4, 5, 5, 5, 6, 6, 7, 7)$.

4 Reducing the number of commutators.

4.1 General considerations.

Different strategies have been explored in the literature to reduce the total number of commutators in numerical Lie group solvers. In particular, the theory of graded free Lie algebras allows to obtain an upper bound on the number of linearly independent terms required for a method of order p, and thus also on the commutators involved [10]. More specifically, let us consider a free Lie algebra $\mathcal{L}_Q$ generated by the set $Q = \{Q_1, Q_2, \ldots, Q_n\}$. We introduce a grading function w on $\mathcal{L}_Q$ as follows. We assign a grade to the generators, $w(Q_l) = q_l$, $l = 1, \ldots, n$, with $q_l \leq q_{l+1}$, and then the grade is propagated in the Hall basis $\mathcal{H}$ of $\mathcal{L}_Q$ by additivity: the grade of an element $H \in \mathcal{H}$ of the form $H = [H_1, H_2]$ is $w(H) = w(H_1) + w(H_2)$. Then $\mathcal{H}$ splits into a disjoint union of sets $\mathcal{H}_j$ of grade j: $\mathcal{H} = \bigcup_{j=1}^{\infty} \mathcal{H}_j$ and $\mathcal{L}_Q$ is a graded free Lie algebra [10]. Observe that, in the context of RKMK methods, $Q_l = \mathcal{O}(h^{q_l})$ and the internal stages u_i and v can be expressed as linear combinations of elements of the Hall basis $\mathcal{H}$. The number of linearly independent terms in a scheme of order p can be determined simply by computing the dimension of the subspaces span$\{\mathcal{H}_j\}$, $j = 1, \ldots, p$.

In [2] an optimization technique has been proposed that in certain cases allows to write an element of a graded free Lie algebra with the minimum number of commutators. This procedure has been applied to numerical integrators based on the Magnus expansion, obtaining schemes of order 4, 6 and 8 involving 1, 3 and 6 commutators respectively (the minimum number in each case).

The general problem can be formulated in the following terms: given an element $Z \in \mathcal{L}_Q$ of the form

$$Z = \sum_{i=1}^{s} \sum_{j=1}^{\nu_i} \alpha_{i,j} H_{i,j},$$

where $H_{i,j}$ denotes the jth element of the set $\mathcal{H}_i$, obtain an approximate expression for Z up to grade s involving the minimum number of commutators.

The procedure to solve this problem commences by taking the most general commutator one can build with elements of the set $\{Q_1, Q_2, \ldots, Q_n\}$ such that $w(Q_n) = s$,

$$d_1 = \left[\sum_{i=1}^{n} a_{1,i} Q_i, \sum_{j=1}^{n} b_{1,j} Q_j \right].$$

For explicit RKMK methods, in general $n \geq s$. Next we write the most general commutator one can form with $\{Q_1, Q_2, \ldots, Q_n, d_1\}$,

$$d_2 = \left[\sum_{i=1}^{n} a_{2,i} Q_i + a_{2,n+1} d_1, \sum_{j=1}^{n} b_{2,j} Q_j + b_{2,n+1} d_1 \right]$$

and the action is repeated recursively r times to reproduce the term $[Q_1, [Q_1, \ldots [Q_1, Q_2]]]$, i.e., the term $H_{s,j}$ of $\mathcal{H}_s$ which involves the greatest number of nested commutators (if the corresponding coefficient $\alpha_{s,j} \neq 0$). The problem is reduced then to determine the coefficients $a_{i,j}$, $b_{i,j}$, α_i, γ_i such that

$$Z = \sum_{i=1}^{n} \alpha_i Q_i + \sum_{i=1}^{r} \gamma_i d_i + \Theta(s+1),$$

where $\Theta(s+1)$ represents terms in $\mathcal{L}_Q$ of grade $s+1$ or higher. This results in a nonlinear system of algebraic equations in the coefficients with no guarantee to have real solutions. If there are, then the minimum number of commutators required is precisely r. Otherwise, additional commutators d_{r+1}, etc. must be included in Z. In any case the number of variables to be determined grows tremendously with the grade, so that some simplifying assumptions must be introduced. In this respect it is useful to take into account whatever symmetry properties the element Z has [2].

For the particular class of explicit RKMK schemes with s stages one has a sequence of elements Z_i, $i = 1, \ldots, s+1$ in $\mathcal{L}_Q$ (the internal stages u_i and v) to approximate with the minimum number of commutators and, in addition, the evaluation of $Z_1, \ldots, Z_{i-1}$ is required for computing Z_i. This imposes severe restrictions on the optimization procedure so that, in general, the optimal number of commutators is much higher than the minimum number required to approximate each individual term in the algorithm.

4.2 Optimization of DVERK.

As an illustration of the above procedure we next consider Verner's method of order 6(5) (DVERK) and try to minimize the number of commutators in the corresponding RKMK scheme. When the transformation (1.4) with matrix (3.8) is applied to the internal stages $u_i = \sum_{j=1}^{i-1} a_{ij}\, \texttt{dexpinv}(u_j, k_j, 5)$, $i = 1, \ldots, 8$ and $v = \sum_{i=1}^{8} b_i\, \texttt{dexpinv}(u_i, k_i, 6)$ in Algorithm 1.1, we obtain (up to order $\mathcal{O}(h^5)$ in u_i and $\mathcal{O}(h^6)$ in v)

$$
\begin{aligned}
u_1 &= 0, \\
u_2 &= \frac{1}{6} Q_1, \\
u_3 &= \frac{4}{15} Q_1 + \frac{16}{75} Q_2 - \frac{4}{225}[1,2] + \frac{1}{2025}[1,1,2], \\
u_4 &= \frac{2}{3} Q_1 + \frac{4}{3} Q_2 + \frac{5}{2} Q_3 - \frac{2}{45}[1,2] - \frac{1}{3}[1,3] - \frac{67}{4050}[1,1,2] + \\
&\quad + \frac{2}{135}[1,1,3] + \frac{13}{8100}[1,1,1,2] - \frac{4}{15}[2,3] - \frac{88}{3375}[2,1,2], \\
u_5 &= \frac{5}{6} Q_1 + \frac{25}{12} Q_2 - \frac{425}{64} Q_3 + \frac{85}{96} Q_4 - \frac{31}{48}[1,2] + \frac{255}{128}[1,3] - \\
&\quad - \frac{85}{288}[1,4] + \frac{409}{4320}[1,1,2] - \frac{119}{384}[1,1,3] + \frac{85}{2592}[1,1,4] - \\
&\quad - \frac{731}{77760}[1,1,1,2] + \frac{493}{96}[2,3] - \frac{85}{144}[2,4] + \frac{493}{4050}[2,1,2], \\
u_6 &= Q_1 + 3Q_2 + \frac{55}{9} Q_3 + \frac{11}{36} Q_4 + \frac{88}{255} Q_5 - \frac{17}{90}[1,2] - \frac{319}{144}[1,3] - \\
&\quad - \frac{997}{8100}[1,1,2] - \frac{22}{153}[1,5] + \frac{23683}{38880}[1,1,3] - \frac{187}{3888}[1,1,4] + \\
&\quad + \frac{1837}{48600}[1,1,1,2] - \frac{32351}{4320}[2,3] + \frac{143}{432}[2,4] - \frac{42031}{81000}[2,1,2], \\
u_7 &= \frac{1}{15} Q_1 + \frac{1}{75} Q_2 - \frac{5}{4} Q_3 + \frac{9}{100} Q_4 + \frac{2484}{10625} Q_5 - \frac{29}{450}[1,2] - \\
&\quad - \frac{557}{600}[1,3] + \frac{39}{1000}[1,4] - \frac{10271}{202500}[1,1,2] - \frac{207}{2125}[1,5] + \\
&\quad + \frac{89957}{216000}[1,1,3] - \frac{443}{12000}[1,1,4] + \frac{36977}{1620000}[1,1,1,2] - \\
&\quad - \frac{121999}{24000}[2,3] + \frac{1209}{4000}[2,4] - \frac{408037}{1350000}[2,1,2],
\end{aligned}
$$

$$\begin{aligned} u_8 &= Q_1 + 3Q_2 + \frac{1375}{258}Q_3 + \frac{187}{516}Q_4 + \frac{220}{731}Q_5 + \frac{3850}{26703}Q_7 - \frac{59}{258}[1,2] - \\ &- \frac{935}{516}[1,3] - \frac{11}{344}[1,4] - \frac{2377}{23220}[1,1,2] - \frac{1881}{18275}[1,5] + \\ &+ \frac{30107}{74304}[1,1,3] - \frac{407}{12384}[1,1,4] + \frac{75823}{2786400}[1,1,1,2] - \\ &- \frac{46981}{8256}[2,3] + \frac{275}{1376}[2,4] - \frac{198583}{464400}[2,1,2], \\ v &= Q_1 + 3Q_2 + \frac{3}{4}Q_4 + \frac{12}{85}Q_5 + \frac{3}{44}Q_6 + \frac{43}{616}Q_8 - \frac{1}{2}[1,2] + \\ &+ \frac{25}{96}[1,3] - \frac{11}{48}[1,4] - \frac{4}{85}[1,5] - \frac{3}{88}[1,6] - \frac{5}{192}[1,1,3] + \\ &+ \frac{1}{96}[1,1,4] + \frac{1}{120}[1,1,1,2] + \frac{15}{16}[2,3] - \frac{3}{8}[2,4] - \frac{3}{20}[2,1,2] + \\ &+ \frac{25}{2484}[1,7] + \frac{1}{425}[1,1,5] + \frac{1}{176}[1,1,6] - \frac{5}{1152}[1,1,1,3] + \\ &+ \frac{11}{2880}[1,1,1,4] + \frac{1}{20}[1,2,1,2] - \frac{36}{425}[2,5] - \frac{9}{88}[2,6] - \\ &- \frac{5}{64}[2,1,3] - \frac{1}{160}[2,1,4] - \frac{15}{64}[3,4] + \frac{5}{32}[3,1,2] - \frac{1}{10}[4,1,2], \end{aligned}$$

where $[i_1, i_2, \ldots, i_{l-1}, i_l]$ stands for the nested commutator $[Q_{i_1}, [Q_{i_2}, \ldots [Q_{i_{l-1}}, Q_{i_l}]]]$. On the other hand, the expression of v for the embedded method reads

$$\begin{aligned} \hat{v} &= Q_1 + 3Q_2 + \frac{3}{4}Q_4 + \frac{12}{85}Q_5 + \frac{3}{44}Q_6 - \frac{1}{2}[1,2] + \frac{25}{96}[1,3] - \\ &- \frac{11}{48}[1,4] - \frac{4}{85}[1,5] - \frac{3}{88}[1,6] - \frac{5}{192}[1,1,3] + \frac{1}{96}[1,1,4] + \\ &+ \frac{1}{120}[1,1,1,2] + \frac{15}{16}[2,3] - \frac{3}{8}[2,4] - \frac{3}{20}[2,1,2] + \frac{1}{765}[1,1,5] + \\ &+ \frac{1}{176}[1,1,6] + \frac{1}{20736}[1,1,1,3] + \frac{37}{10368}[1,1,1,4] + \\ &+ \frac{229}{972000}[1,1,1,1,2] + \frac{127}{2700}[1,2,1,2] - \frac{13}{170}[2,5] - \frac{9}{88}[2,6] - \\ &- \frac{461}{3456}[2,1,3] - \frac{5}{1728}[2,1,4] - \frac{95}{384}[3,4] + \frac{299}{1728}[3,1,2] - \\ &- \frac{217}{2160}[4,1,2]. \end{aligned}$$

We observe that u_3 requires the evaluation of the commutators

$$d_1 = [Q_1, Q_2], \qquad d_2 = [Q_1, d_1] \tag{4.1}$$

so that

$$u_3 = \frac{4}{15}Q_1 + \frac{16}{75}Q_2 - \frac{4}{225}d_1 + \frac{1}{2025}d_2. \tag{4.2}$$

In u_4 we have the commutators $[Q_1, Q_3]$ and $[Q_1, [Q_1, Q_3]]$ which cannot be generated from u_3. Thus we introduce

$$d_3 = [Q_1, Q_3] \tag{4.3}$$

and the new commutator

$$d_4 = \left[\sum_{i=1}^{3} a_{4,i} Q_i + \sum_{i=4}^{6} a_{4,i} d_{i-3}, \sum_{j=1}^{3} b_{4,j} Q_j + \sum_{j=4}^{6} b_{4,j} d_{j-3}\right] \tag{4.4}$$

to reproduce the remaining terms. In fact, most of the coefficients in (4.4) are redundant: with

$$d_4 = \left[Q_1 + Q_2, -\frac{4}{15}Q_3 - \frac{88}{3375}d_1 + \frac{13}{8100}d_2 + \frac{2}{135}d_3\right] \tag{4.5}$$

we have

$$u_4 = \frac{2}{3}Q_1 + \frac{4}{3}Q_2 + \frac{5}{2}Q_3 - \frac{2}{45}d_1 + \frac{193}{20250}d_2 - \frac{1}{15}d_3 + d_4. \tag{4.6}$$

In the same way, at least two additional commutators are needed to write u_5 because $[Q_1, Q_4]$ and $[Q_1, [Q_1, Q_4]]$ cannot be generated as a linear combination of $d_1, \ldots, d_4$. Thus we introduce

$$d_5 = [Q_1, Q_4] \tag{4.7}$$

and a new commutator d_6 to reproduce the terms $[Q_1, [Q_1, Q_3]]$, $[Q_1, [Q_1, Q_4]]$, $[Q_2, Q_3]$, $[Q_2, Q_4]$, $[Q_1, [Q_1, [Q_1, Q_2]]]$ and $[Q_2, [Q_1, Q_2]]$ in u_5. This can be achieved if

$$d_6 = [Q_1 + Q_2, b_{6,3}Q_3 + b_{6,4}Q_4 + b_{6,5}d_1 + b_{6,6}d_2 + b_{6,7}d_3 + b_{6,9}d_5] \tag{4.8}$$

with coefficients

$$\begin{aligned} b_{6,3} &= \frac{493}{96}, \ b_{6,4} = -\frac{85}{144}, \ b_{6,5} = \frac{493}{4050}, \ b_{6,6} = -\frac{731}{77760}, \ b_{6,7} = -\frac{119}{384}, \\ b_{6,9} &= \frac{85}{2592}. \end{aligned}$$

Finally

$$u_5 = \frac{5}{6}Q_1 + \frac{25}{12}Q_2 - \frac{425}{64}Q_3 + \frac{85}{96}Q_4 - \frac{31}{48}d_1 - \frac{1753}{64800}d_2 - \frac{1207}{384}d_3 + \frac{85}{288}d_5 + d_6. \tag{4.9}$$

In the expression of u_6 one has the commutator $[Q_1, Q_5]$, which cannot be expressed as a linear combination of $d_1, \ldots, d_6$. Thus we need to include at least one additional commutator d_7. With

$$\begin{aligned} d_7 = &\left[Q_1 + Q_2, -\frac{32351}{4320}Q_3 + \frac{143}{432}Q_4 - \frac{22}{153}Q_5 - \frac{42031}{81000}d_1 + \right. \\ &\left. + \frac{1837}{48600}d_2 + \frac{23683}{38880}d_3 - \frac{187}{3888}d_5\right] \end{aligned} \tag{4.10}$$

then

$$u_6 = Q_1 + 3Q_2 + \frac{55}{9}Q_3 + \frac{11}{36}Q_4 + \frac{88}{255}Q_5 - \frac{17}{90}d_1 + \frac{10687}{27000}d_2 + + \frac{22781}{4320}d_3 - \frac{143}{432}d_5 + d_7. \tag{4.11}$$

The expressions of u_7 and u_8 do not contain additional commutators so that, in principle, one could express them as linear combinations of $d_1, \ldots, d_7$. This turns out to be the case for u_7,

$$u_7 = \frac{1}{15}Q_1 + \frac{1}{75}Q_2 - \frac{5}{4}Q_3 + \frac{9}{100}Q_4 + \frac{2484}{10625}Q_5 - \frac{29}{450}d_1 + + \frac{1018691}{4050000}d_2 + \frac{99719}{24000}d_3 - \frac{2507}{1000}d_4 - \frac{1053}{4000}d_5 - \frac{1404}{10625}d_6 + + \frac{1863}{2750}d_7 \tag{4.12}$$

but not for u_8, so that one additional commutator must be introduced:

$$d_8 = \left[Q_1 + Q_2, -\frac{46981}{8256}Q_3 + \frac{275}{1376}Q_4 - \frac{1881}{18275}Q_5 - \frac{198583}{464400}d_1 + + \frac{75823}{2786400}d_2 + \frac{30107}{74304}d_3 - \frac{407}{12384}d_5\right] \tag{4.13}$$

so that

$$u_8 = Q_1 + 3Q_2 + \frac{1375}{258}Q_3 + \frac{187}{516}Q_4 + \frac{220}{731}Q_5 + \frac{3850}{26703}Q_7 - \frac{59}{258}d_1 + + \frac{151043}{464400}d_2 + \frac{32021}{8256}d_3 - \frac{319}{1376}d_5 + d_8. \tag{4.14}$$

Finally, in v one has the new commutators $[Q_1, Q_6]$ and $[Q_1, [Q_1, Q_6]]$ so that at least two additional commutators are required to reproduce it up to order $\mathcal{O}(h^6)$. Let us introduce

$$d_9 = [Q_1, Q_9] \tag{4.15}$$

and, in order to avoid redundancies,

$$d_{10} = \left[\sum_{i=1}^{4} a_{10,i}Q_i, \sum_{j=3}^{7} b_{10,j}Q_j + \sum_{j=8}^{16} b_{10,j}d_{j-7}\right]. \tag{4.16}$$

Then

$$v = Q_1 + 3Q_2 + \frac{3}{4}Q_4 + \frac{12}{85}Q_5 + \frac{3}{44}Q_6 + \frac{43}{616}Q_8 + \sum_{i=1}^{10} \gamma_i d_i. \tag{4.17}$$

After expanding d_{10}, equating terms with v and solving the corresponding equations we get

$$
\begin{aligned}
&a_{10,1} = \tfrac{1}{1000}, && a_{10,2} = \tfrac{105964609689}{266416480000},\\
&a_{10,3} = -\tfrac{8643461104256535782827776169}{6153752550752682472960000}, && a_{10,4} = \tfrac{3419664172386303842612439933}{15384381376881706182400000},\\
&b_{10,3} = \tfrac{23392956798379165625O}{1247610945171345852969}, && b_{10,4} = -\tfrac{368121993802738025000}{1247610945171345852969},\\
&b_{10,5} = -\tfrac{42626636800}{200155373857}, && b_{10,6} = -\tfrac{3027460000}{11773845521},\\
&b_{10,7} = \tfrac{6250}{621}, && b_{10,8} = -\tfrac{850304064427933000}{1247610945171345852969},\\
&b_{10,9} = -\tfrac{324431382151464725}{7189486838179272}, && b_{10,10} = -\tfrac{2566642496229085375}{3195327483635232},\\
&b_{10,11} = \tfrac{406587275}{316624}, && b_{10,12} = \tfrac{82453548663197675}{1597663741817616},\\
&b_{10,13} = \tfrac{4212930}{30583}, && b_{10,14} = \tfrac{577797525}{870716},\\
&b_{10,15} = -\tfrac{413547125}{435358}, && b_{10,16} = \tfrac{125}{22},\\
&\gamma_1 = -\tfrac{1}{2}, && \gamma_2 = \tfrac{844953172496012601108646339}{5643193827199031562149380800},\\
&\gamma_3 = -\tfrac{27208938922924187993305463 5}{45145550617592252497195046 4}, && \gamma_4 = \tfrac{29817608309216341}{14911528256964416},\\
&\gamma_5 = \tfrac{30276197407357512902600687}{225727753087961262485975232}, && \gamma_6 = \tfrac{2782759594196403}{3600795175687430},\\
&\gamma_7 = \tfrac{1544992960649931}{41006702706652144}, && \gamma_8 = \tfrac{8252596669788325}{20503351353326072},\\
&\gamma_9 = -\tfrac{35055120083}{1036098405848}, && \gamma_{10} = 1.
\end{aligned}
\tag{4.18}
$$

On the other hand, to reproduce $\hat{v}$ we must introduce one additional commutator d_{11} similar to d_{10}:

$$
d_{11} = \left[\sum_{i=1}^{4} a_{11,i} Q_i, \sum_{j=3}^{7} b_{11,j} Q_j + \sum_{j=8}^{16} b_{11,j} d_{j-7}\right]. \tag{4.19}
$$

Then

$$
\hat{v} = Q_1 + 3Q_2 + \frac{3}{4} Q_4 + \frac{12}{85} Q_5 + \frac{3}{44} Q_6 + \sum_{i=1}^{9} \hat{\gamma}_i d_i + \hat{\gamma}_{11} d_{11} \tag{4.20}
$$

with

$$
\begin{aligned}
&a_{11,1} = \tfrac{1}{10000}, && a_{11,2} = \tfrac{182530798193}{4575730800000},\\
&a_{11,3} = -\tfrac{686597984321203921435659208559 53}{42389987541147861854135040000 00}, && a_{11,4} = \tfrac{2161597919193097703552114863 73}{847799750822957237082700800 0},\\
&b_{11,3} = \tfrac{759054368666031868903125 0}{403141656696581673989512 9}, && b_{11,4} = -\tfrac{118869760932016231931250 0}{403141656696581673989512 9},\\
&b_{11,5} = -\tfrac{5948450040000}{3103023569281}, && b_{11,6} = -\tfrac{5147697150000}{2007838780123},\\
&b_{11,7} = 0, && b_{11,8} = -\tfrac{2386307664398455618500 0}{403141656696581673989512 9},\\
&b_{11,9} = -\tfrac{4937850192461620488 5}{111459146362187976}, && b_{11,10} = -\tfrac{1171104867204907392125}{148612195149583968},\\
&b_{11,11} = \tfrac{36891785375}{2849616}, && b_{11,12} = \tfrac{37582182892976202625}{74306097574791984},\\
&b_{11,13} = \tfrac{131169950}{91749}, && b_{11,14} = \tfrac{6127951625}{870716},\\
&b_{11,15} = -\tfrac{13007096875}{1306074}, && b_{11,16} = \tfrac{625}{11}
\end{aligned}
\tag{4.21}
$$

and

$$\hat{\gamma}_1 = -\frac{1}{2}, \qquad \hat{\gamma}_2 = \frac{24826732408965635981474176 53}{16577184923363438434448770448},$$

$$\hat{\gamma}_3 = -\frac{119785970219681956126484569385}{198926219080361261213385245376}, \qquad \hat{\gamma}_4 = \frac{2543028894134407003}{1271459891835329504},$$

$$\hat{\gamma}_5 = \frac{133380623122199657462503239 37}{994631095401806306066926226 88}, \qquad \hat{\gamma}_6 = \frac{237303810989198421}{307028667062508545},$$

$$\hat{\gamma}_7 = \frac{134172361749673629}{3496514702547156136}, \qquad \hat{\gamma}_8 = \frac{702333383164435955}{1748257351273578068},$$

$$\hat{\gamma}_9 = -\frac{543474236859}{16062710240984}, \qquad \hat{\gamma}_{11} = 1.$$

Thus the RKMK method based on DVERK is formulated in terms of only 10 commutators. The final algorithm in its optimized form reads

ALGORITHM 4.1.

for $i = 1:8$ ***do***
  u_i *given by the optimized expressions*
  $k_i = hf(\exp(u_i) \cdot y_0)$
  $Q_i = \sum_{j=1}^{i} V_{i,j} k_j$, *with* V *given by* (3.8)
end do
v *given by* (4.17)
$y_1 = \exp(v) \cdot y_0$
$\hat{v}$ *given by* (4.20)
$\hat{y}_1 = \exp(\hat{v}) \cdot y_0$

4.3 Optimized RKMK integration schemes.

The same procedure can be carried out for the explicit RKMK methods based on the Runge–Kutta schemes mentioned in Section 3.3. The complete optimized expressions of the corresponding internal stages u_i and v are collected in [4]. Next we briefly describe the results obtained.

5th-order

The schemes RKF45 and DOPRI5(4) (6 stages) are expressed in terms of 5 commutators (6 with the embedded pair), this being the minimum number. A complete treatment of the optimization procedure of DOPRI5(4) can be found in [1].

6th-order

The 6th-order Butcher method also needs 10 commutators. Here, as is the case with DVERK, one additional commutator is necessary to write u_7 (it cannot be expressed as a linear combination of $d_1, \ldots, d_7$).

7th-order

The optimization process of the 7th-order methods Butcher7 and RKF7(8) considered in Section 3.3 is technically much more difficult for several reasons: they

Table 5.1: Number of commutators involved in some RKMK methods. Columns indicate order (p), number of stages (s), the number of commutators as presented originally in [9] (Orig), the number indicated by the free Lie algebra approach of [10] (FLA), and the number obtained by the approach of this paper (New). The numbers in parenthesis refer to the embedded pair (whenever it differs)

Method	p	s		Orig		FLA	New	
RKF45	5	6		20		11	5	(6)
DOPRI5(4)	5	6	(7)	20	(24)	12	5	(6)
DVERK	6	8		28		26	10	(11)
Butcher6	6	7		24		26	10	
Butcher7	7	9		48		60	21	
RKF7	7	11		60		64	23	

involve more stages (9 and 11, respectively), the computations for u_i have to be carried out up to order $\mathcal{O}(h^6)$ and v up to order $\mathcal{O}(h^7)$, and the dimensions of the homogeneous subspaces of the graded free Lie algebra grow very rapidly.

We can formulate Butcher7 with a total of 21 commutators, fifteen of them for the internal stages (the absolute minimum is 13, but no real solutions were obtained). This, although not optimal in a strict sense, is nevertheless a considerable improvement with respect to the original implementation in terms of the k_i.

For RKF7(8), 17 commutators are required to write u_i up to order $\mathcal{O}(h^6)$ instead of 15, the absolute minimum in this case. The 64 nonlinear equations involved in v can be solved quite easily by introducing 6 additional commutators. Thus the 7th-order method (which is using only the first 11 stages of RKF78) requires a total of 23 commutators.

5 Concluding remarks.

We have presented new versions of Lie group integrators of Runge–Kutta–Munthe-Kaas type which use a lower number of commutators per stage than what can be found in the literature. We have summarized the results in Table 5.1.

Notice that the numbers given for the new methods are not in all cases proved to be optimal, but they represent a substantial reduction compared to what has been previously known.

We believe that there is reason to be cautious in interpreting these numbers too rigidly. For instance, it is not known to which extent the reduction of complexity will affect the quality of the numerical approximation. Also, one should keep in mind that in some cases, the savings obtained by reducing the number of commutators could be insignificant compared for instance to the cost of calculating exponentials. Nevertheless, we still think that there are important problems for which the obtained reduced commutator counts may substantially improve the efficiency of the Lie group integrators presented here.

Here we have included, as an illustration of the procedure, the optimized expressions for the 6th-order scheme based on DVERK. The coefficients for the other methods analyzed in this work can be found in the technical report [4] and can be obtained in electronic format as well from the authors upon request.

Acknowledgments.

The work of FC has been partially supported by Fundació Bancaixa under project P1 1B2002-22. FC would like to thank the Center for Advanced Study, Oslo, Norway for its hospitality. The work of BO was sponsored by the Center for Advanced Study, Oslo, Norway.

REFERENCES

1. S. Blanes and F. Casas, *Optimization of Lie group methods for differential equations*, Future Generation Computer Systems, 19 (2003), pp. 331–339.
2. S. Blanes, F. Casas, and J. Ros, *High order optimized geometric integrators for linear differential equations*, BIT, 42(2) (2002), pp. 262–284.
3. J. C. Butcher, *The Numerical Analysis of Ordinary Differential Equations*, Wiley, Chichester, 1987.
4. F. Casas and B. Owren, *Cost efficient Lie group integrators in the RKMK class*, Technical Report Numerics no.5/2002, Norwegian University of Science and Technology, Trondheim, Norway, 2002.
5. E. Celledoni, A. Iserles, S. P. Nørsett, and B. Orel, *Complexity theory for Lie-group solvers*, J. Complexity, 18(1) (2002), pp. 242–286.
6. E. Hairer, S. P. Nørsett, and G. Wanner, *Solving Ordinary Differential Equations I, Nonstiff Problems*, Springer-Verlag, 2nd revised ed., 1993.
7. A. Iserles, H. Z. Munthe-Kaas, S. P. Nørsett, and A. Zanna, *Lie-group methods*, Acta Numerica, 9 (2000), pp. 215–365.
8. A. Iserles and S. P. Nørsett, *On the solution of linear differential equations in Lie groups*, Phil. Trans. Royal Soc. A, 357 (1999), pp. 983–1020.
9. H. Munthe-Kaas, *High order Runge–Kutta methods on manifolds*, Appl. Numer. Math., 29 (1999), pp. 115–127.
10. H. Munthe-Kaas and B. Owren, *Computations in a free Lie algebra*, Phil. Trans. Royal Soc. A, 357 (1999), pp. 957–981.