Opuscula Math. 34, no. 2 (2014), 243–255
http://dx.doi.org/10.7494/OpMath.2014.34.2.243 OPUSCULA MATHEMATICA

Dedicated to the Memory of Professor Zdzisław Kamont

LOCAL ERROR STRUCTURES AND ORDER CONDITIONS IN TERMS OF LIE ELEMENTS FOR EXPONENTIAL SPLITTING SCHEMES

Winfried Auzinger and Wolfgang Herfort

Communicated by Alexander Gomilko

Abstract. We discuss the structure of the local error of exponential operator splitting methods. In particular, it is shown that the leading error term is a Lie element, i.e., a linear combination of higher-degree commutators of the given operators. This structural assertion can be used to formulate a simple algorithm for the automatic generation of a minimal set of polynomial equations representing the order conditions, for the general case as well as in symmetric settings.

Keywords: exponential splitting schemes, local error, defect, order conditions, free Lie algebra.

Mathematics Subject Classification: 17B08, 17B80, 65J08, 65M15, 68W30.

1. INTRODUCTION AND BACKGROUND

The main result of this paper is a statement about the structure of the leading local error term of a consistent exponential splitting method (1.2a); see Theorem 2.6. This has two major implications:

(i) The a priori and a posteriori local error theory from [1], relying on such a local error structure, extends to schemes of arbitrary order.
(ii) Systems of polynomial equations in the method's coefficients defining a minimal, non-redundant set of order conditions can be set up in an elementary way.

Statement (ii) is in contrast to several related techniques to generate order conditions; see [5–7, 9, 11–14] and references therein. The difference is the following: The proof of Theorem 2.6 below is based on a qualitative, structural argument exploiting the Baker-Campbell-Hausdorff (BCH) formula (1.10). For the algorithmic implementation of order conditions we do not make any use of explicit expansions based on BCH.

Rather, we leave the job of setting up the desired system of equations to the computer. In particular, exploiting the assertion of Theorem 2.6 leads to a simple algorithmic formulation (see Algorithm 2 and its variants). Algorithm 2 may be considered as a form of *recursive implicit elimination*, justified by the fact that, due to Theorem 2.6, the condition for order p generated by the algorithm is correct assuming that the conditions for lower orders already hold.

In principle, any set of conditions for arbitrary order may be generated in this way; it is 'merely' limited by computational resources. Namely, the computational effort rapidly grows with the number of stages s and the desired order p since the number of generated terms always multiplies by s when the order is increased by 1. (This type of computational complexity is inherent also to alternative approaches.)

Thus, an open questions remains. 'Leaving the job ... to the computer' is convenient and easy to implement but computationally rather expensive. A deeper understanding of the structure of order conditions might help optimizing the procedure.

Remark 1.1. In this paper we give detailed arguments for the practically most relevant case of a splitting into two operators, see (1.1). However, all this can be extended to the general case of an additive multi-component splitting. In the general case, relation (1.8) below, for instance, is to be replaced by a corresponding multinomial expansion.

1.1. EXPONENTIAL SPLITTING SCHEMES, LOCAL ERROR, AND DEFECT

Consider a linear evolution equation

$$\tfrac{\mathrm{d}}{\mathrm{d}t}\, u(t) = H\, u(t) = (A+B)\, u(t), \quad u(0) \text{ given.} \tag{1.1}$$

In computational practice, $A, B \in \mathbb{C}^{d\times d}$ are matrices arising from spatial discretization of a partial differential equation. However, our considerations are relevant in a more general setting, see Remark 1.3.

Exponential splitting approximations are of the form

$$\mathcal{S}(t) = S_{1\ldots s}(t) := \mathcal{S}_1(t) \cdots \mathcal{S}_s(t) \approx \mathrm{e}^{tH}, \tag{1.2a}$$

where

$$\mathcal{S}_j(t) = \mathrm{e}^{tA_j}\, \mathrm{e}^{tB_j}, \quad A_j = a_j\, A, \quad B_j = b_j\, B, \quad j = 1 \ldots s. \tag{1.2b}$$

Here, the stages $\mathcal{S}_j(t)$ satisfy the Sylvester-type evolution equations

$$\tfrac{\mathrm{d}}{\mathrm{d}t}\mathcal{S}_j(t) = A_j\, \mathcal{S}_j(t) + \mathcal{S}_j(t) B_j = a_j A\, \mathcal{S}_j(t) + b_j\, \mathcal{S}_j(t) B, \tag{1.3a}$$

$$\mathcal{S}_j(0) = I. \tag{1.3b}$$

In [1] the local error of higher-order splitting schemes is represented and analyzed in the following way. With the defect

$$\mathcal{D}(t) = \tfrac{\mathrm{d}}{\mathrm{d}t}\, \mathcal{S}(t) - H\, \mathcal{S}(t) \tag{1.4}$$

of the splitting operator with respect to the given evolution equation (1.1), the local error operator

$$\mathcal{L}(t) = \mathcal{S}(t) - \mathrm{e}^{tH} \tag{1.5a}$$

has the integral representation

$$\mathcal{L}(t) = \int_0^t \mathrm{e}^{(t-\tau)H}\mathcal{D}(\tau)\,\mathrm{d}\tau, \quad \mathcal{L}(0) = 0. \tag{1.5b}$$

Successive differentiation of (1.5) and evaluation at $t = 0$ shows that the asymptotic order $p \geq 1$,

$$\mathcal{L}(t) = \mathcal{O}(t^{p+1}) \quad \text{for} \quad t \to 0 \tag{1.6}$$

is characterized by the equivalent conditions

$$0 = \tfrac{\mathrm{d}}{\mathrm{d}t}\,\mathcal{L}(0) = \tfrac{\mathrm{d}^2}{\mathrm{d}t^2}\,\mathcal{L}(0) = \ldots = \tfrac{\mathrm{d}^p}{\mathrm{d}t^p}\,\mathcal{L}(0), \tag{1.7a}$$

$$\Leftrightarrow \quad 0 = \mathcal{D}(0) = \tfrac{\mathrm{d}}{\mathrm{d}t}\,\mathcal{D}(0) = \ldots = \tfrac{\mathrm{d}^{p-1}}{\mathrm{d}t^{p-1}}\,\mathcal{D}(0). \tag{1.7b}$$

For a given number s of stages, expressions for the derivatives $\frac{\mathrm{d}^n}{\mathrm{d}t^n}\mathcal{D}(0)$ can be generated as follows:

Algorithm 1.

- Start with the symbolic expression $\mathcal{S}(t) = \mathcal{S}_1(t) \cdots \mathcal{S}_s(t)$. The symbolic variables A, B as well as the $\mathcal{S}_j(t)$ are declared to be non-commuting.
- Differentiate $\mathcal{S}(t)$ (chain rule) and initialize [1)]

$$\mathcal{X}(t) := \tfrac{\mathrm{d}}{\mathrm{d}t}\,\mathcal{S}(t) - H\,\mathcal{S}(t), \quad H = A + B.$$

- In $\mathcal{X}(t)$, replace by the terms $\frac{\mathrm{d}}{\mathrm{d}t}\,\mathcal{S}_j(t)$ by $a_j A\,\mathcal{S}_j(t) + b_j\,\mathcal{S}_j(t)B$ (see (1.3a)).
- Evaluate $\mathcal{X}(t)$ at $t = 0$, i.e., replace the terms $\mathcal{S}_j(t)$ by 1 (see (1.3b)), resulting in an expression for $\mathcal{D}(0)$.
- For $q = 1, 2, \ldots$:
  - Differentiate $\mathcal{X}(t)$ and set [2)] $\mathcal{X}(t) := \frac{\mathrm{d}}{\mathrm{d}t}\mathcal{X}(t)$.
  - In $\mathcal{X}(t)$, replace the terms $\frac{\mathrm{d}}{\mathrm{d}t}\,\mathcal{S}_j(t)$ by $a_j A\,\mathcal{S}_j(t) + b_j\,\mathcal{S}_j(t)B$.
  - Evaluate $\mathcal{X}(t)$ at $t = 0$, i.e., replace the terms $\mathcal{S}_j(t)$ by 1, resulting in an expression for $\frac{\mathrm{d}^q}{\mathrm{d}t^q}\,\mathcal{D}(0)$.

[1)] Here, $\mathcal{X}(t) = \mathcal{D}(t)$.
[2)] Here, $\mathcal{X}(t) = \frac{\mathrm{d}^q}{\mathrm{d}t^q}\,\mathcal{D}(t)$.

Example 1.2. As an illustration, we describe the initial steps of Algorithm 1 for $s = 2$:

$$\begin{aligned}\mathcal{D}(0) &= (a_1 + a_2 - 1)\,A + (b_1 + b_2 - 1)\,B,\\ \tfrac{\mathrm{d}}{\mathrm{d}t}\,\mathcal{D}(0) &= (a_1 + a_2)(a_1 + a_2 - 1)\,A^2 +\\ &\quad + (2\,a_1\,b_1 + 2\,a_1\,b_2 + 2\,a_2\,b_2 - b_1 - b_2)\,AB +\\ &\quad + (2\,a_2\,b_1 - a_1 - a_2)\,BA +\\ &\quad + (b_1 + b_2)(b_1 + b_2 - 1)\,B^2.\end{aligned}$$

At a first glance, the derivative $\frac{\mathrm{d}}{\mathrm{d}t}\,\mathcal{D}(0)$ has an intricate structure. Now we repeat this computation, assuming that the first-order condition $\mathcal{D}(0) = 0$ is satisfied. I.e., we assume $a_1 + a_2 = b_1 + b_2 = 1$ and modify the initialization of $\mathcal{X}(t)$ in Algorithm 1 accordingly: $\mathcal{X}(t) := \frac{\mathrm{d}}{\mathrm{d}t}\,\mathcal{S}(t) - ((a_1 + a_2)A + (b_1 + b_2)B)\,\mathcal{S}(t)$. This results in

$$\begin{aligned}\mathcal{D}(0) &= 0,\\ \tfrac{\mathrm{d}}{\mathrm{d}t}\,\mathcal{D}(0) &= (a_1\,b_1 + a_1\,b_2 - a_2\,b_1 + a_2\,b_2)\,(AB - BA).\end{aligned}$$

We observe: *If it is assumed that the first-order condition is satisfied,* $\mathcal{D}(0) = 0$*, then*

$$\tfrac{\mathrm{d}}{\mathrm{d}t}\,\mathcal{D}(0) = c_2\,[A, B],$$

with a homogeneous polynomial c_2 *of degree* 2. Thus, the second-order condition is given by

$$c_2 = c_2(a_1, a_2, b_1, b_2) = a_1\,b_1 + a_1\,b_2 - a_2\,b_1 + a_2\,b_2 = 0.$$

In [1] the analogous structure of the terms $\frac{\mathrm{d}^p}{\mathrm{d}t^p}\,\mathcal{D}(0)$ in terms of commutators of A and B is explicitly investigated up to order $p = 3$. For $p \geq 4$, computing these expressions become laborious.

The higher derivatives $\frac{\mathrm{d}^q}{\mathrm{d}t^q}\,\mathcal{S}(0)$, which are generated in course of Algorithm 1, can also be expressed by means of the following straightforward expansion. Since, as a consequence of (1.3a), we have

$$\tfrac{\mathrm{d}^k}{\mathrm{d}t^k}\,\mathcal{S}_j(0) = \sum_{\ell=0}^{k} \binom{k}{\ell} A_j^{k-\ell} B_j^{\ell}, \quad k = 0, 1, 2, \ldots, \tag{1.8}$$

n-fold differentiation of $\mathcal{S}(t) = \mathcal{S}_1(t) \cdots \mathcal{S}_s(t)$ yields the multinomial Leibniz representation (with $\mathbf{k} = (k_1, \ldots, k_s)$)

$$\tfrac{\mathrm{d}^q}{\mathrm{d}t^q}\,\mathcal{S}(0) = \sum_{|\mathbf{k}|=q} \binom{q}{\mathbf{k}} \prod_{j=1}^{s} \sum_{\ell=0}^{k_j} \binom{k_j}{\ell} A_j^{k_j-\ell} B_j^{\ell}, \quad q = 0, 1, 2, \ldots. \tag{1.9a}$$

Thus,

$$\tfrac{\mathrm{d}^q}{\mathrm{d}t^q}\,\mathcal{L}(0) = \tfrac{\mathrm{d}^{q-1}}{\mathrm{d}t^{q-1}}\,\mathcal{D}(0) = \tfrac{\mathrm{d}^q}{\mathrm{d}t^q}\,\mathcal{S}(0) - H\tfrac{\mathrm{d}^{q-1}}{\mathrm{d}t^{q-1}}\,\mathcal{S}(0), \quad q = 1, 2, 3, \ldots, \tag{1.9b}$$

with $\frac{\mathrm{d}^{q-1}}{\mathrm{d}t^{q-1}}\,\mathcal{S}(0)$, $\frac{\mathrm{d}^{q}}{\mathrm{d}t^{q}}\,\mathcal{S}(0)$ according to (1.9a). Equivalently, with $\frac{\mathrm{d}^{q}}{\mathrm{d}t^{q}}\,\mathrm{e}^{tH}(0) = H^q$ we have

$$\tfrac{\mathrm{d}^{q}}{\mathrm{d}t^{q}}\,\mathcal{L}(0) = \tfrac{\mathrm{d}^{q-1}}{\mathrm{d}t^{q-1}}\,\mathcal{D}(0) = \tfrac{\mathrm{d}^{q}}{\mathrm{d}t^{q}}\,\mathcal{S}(0) - H^q, \quad q = 1,2,3,\ldots, \tag{1.9c}$$

with $\frac{\mathrm{d}^{q}}{\mathrm{d}t^{q}}\,\mathcal{S}(0)$ from (1.9a).

Step-wise multinomial expansion according to (1.9) generates a rapidly growing number of terms as q increases. This representation provides no immediate clue on the structure of $\frac{\mathrm{d}^{q}}{\mathrm{d}t^{q}}\,\mathcal{L}(0) = \frac{\mathrm{d}^{q-1}}{\mathrm{d}t^{q-1}}\,\mathcal{D}(0)$ in terms of higher-degree commutators of A and B.

We are aiming for a combination of this expansion with a qualitative structural argument. Using Lie theory we prove that a hierarchical dependence of order conditions, as indicated in Example 1.2, is valid in general. Algorithm 2 below based on (1.9) generates a minimal, non-redundant set of order conditions.

The extension to schemes with special symmetries is also discussed.

Remark 1.3. The order conditions discussed in this paper remain valid if nonlinear evolution equations are considered and splitting schemes are defined via successive application of (nonlinear) subflows. This can be argued using the calculus of Lie derivatives, as explained in [9]. Moreover, they remain valid in a more general Banach space setting and for unbounded (differential) operators, provided the numerical process remains in the domain of definition of these operators; see for instance [1,10].

As the following considerations show, our approach can be used to generate a 'universal' set of equations (order conditions), e.g. for $s = 12$ and $p = 6$. Conditions for a smaller number of stages or lower order can be obtained by setting the superfluous variables a_j, b_j equal to zero and cancelling all terms containing these variables.

1.2. ALGEBRAIC SETTING

Let $\mathbb{C}\langle A,B\rangle$ denote the ring of formal power series in the non-commuting variables A and B. Introducing the Lie bracket

$$[A,B] = AB - BA \qquad \text{(the commutator of } A \text{ and } B\text{)}$$

we obtain a free Lie algebra $[\mathbb{C}\langle A,B\rangle]$ generated by A and B. We use the following terminology: A and B are commutators of degree 1, and commutators of degree q are of the form $[X,Y]$ where X and Y are commutators of degree α and $q-\alpha$.

A homogeneous *Lie element* of degree q is a complex linear combination of commutators of degree q. For the sake of a unique representation of an arbitrary homogeneous Lie element of degree q, we use the Lyndon basis[3] which is represented by so-called Lyndon words $L_{q,\ell}(A,B)$. Table 1 lists the Lyndon words up to degree $q = 6$. (Lyndon words of arbitrary degree can, e.g., be generated by an algorithm due to Duval [8].) Here, e.g., $A^2\,B^2$ represents the commutator $[A,[[A,B],B]]$, and the element $A^2\,B^2$ does not occur in any other commutator of degree 4.

[3] Also other basis sets may be used.

Table 1. Lyndon words (ℓ_q is the number of words of degree q)

q	ℓ_q	Lyndon words $L_{q,\ell}(A,B)$ of degree q
1	2	A, B
2	1	AB
3	2	A^2B, AB^2
4	3	A^3B, A^2B^2, AB^3
5	6	A^4B, A^3B^2, A^2BAB, A^2B^3, $ABAB^2$, AB^4
6	9	A^5B, A^4B^2, A^3BAB, A^3B^3, A^2BAB^2, A^2B^2AB, A^2B^4, $ABAB^3$, AB^5
		$\ldots$

Our main theoretical tool is the Baker-Campbell-Hausdorff (BCH) formula (cf. e.g. [3,9]),

$$\mathrm{e}^{tA}\,\mathrm{e}^{tB} = \mathrm{e}^{t(A+B)+\frac{t^2}{2}[A,B]+\frac{t^3}{12}([A,[A,B]]+[[A,B],B])+\ldots} \tag{1.10}$$

The exponent on the right-hand side is a locally convergent power series where the coefficients associated with t^q are homogeneous Lie elements of degree q.

We stress that our approach makes no explicit use of the detailed form of the terms in BCH and analogous expansions; we only refer to its structure for the purpose of qualitative argumentation.

2. THEORETICAL RESULTS

Remark 2.1. In the following, expansions of the form

$$\mathrm{e}^{tX^{[1]}+t^2X^{[2]}+\ldots} \tag{2.1}$$

are considered, where the exponent is a locally convergent power series and the coefficients $X^{[q]}$ associated with t^q are homogeneous Lie elements of degree q. We call (2.1) a *BCH-like expansion.*

The following argument is standard in the study of exponential splitting schemes, see [9].

Lemma 2.2. *Each splitting operator $\mathcal{S}(t)=\mathcal{S}_{1\ldots s}(t)$ of the form* (1.2a) *admits a BCH-like expansion*

$$\mathcal{S}(t)=\mathrm{e}^{tX_s}=\mathrm{e}^{tX_s^{[1]}+t^2X_s^{[2]}+\ldots}\,. \tag{2.2}$$

Proof. For $s=1$ we have $\mathcal{S}_1(t)=\mathrm{e}^{tA_1}\,\mathrm{e}^{tB_1}$, and the assertion follows immediately from the BCH formula (1.10). Assume that (2.2) is true for some $s\geq 1$. Then,

$$\mathcal{S}_{1\cdots s+1}(t)=\mathcal{S}_{1\cdots s}(t)\,\mathcal{S}_{s+1}(t)=\mathrm{e}^{tX_s}\,\mathrm{e}^{tY_{s+1}},$$

where

$$\mathcal{S}_{s+1}(t)=\mathrm{e}^{tY_{s+1}}=\mathrm{e}^{tY_{s+1}^{[1]}+t^2\,Y_{s+1}^{[2]}+\ldots}$$

is the BCH-like expansion of $\mathcal{S}_{s+1}(t)=\mathrm{e}^{tA_{s+1}}\,\mathrm{e}^{tB_{s+1}}$. The BCH expansion of the product $\mathcal{S}_{1\cdots s+1}(t)=\mathrm{e}^{tX_s}\,\mathrm{e}^{tY_{s+1}}$ involves commutators in terms of the $t^kX_s^{[k]}$ and

$t^\ell\, Y_{s+1}^{[\ell]}$, and reordering the resulting series with respect to powers of t yields the asserted form. □

We will make use of the analogous assertion for $\mathrm{e}^{-t(A+B)}\mathcal{S}(t)$:

Lemma 2.3. *For each splitting operator $\mathcal{S}(t) = \mathcal{S}_{1\ldots s}(t)$ of the form* (1.2a) *there exists a BCH-like expansion*

$$\mathrm{e}^{-t(A+B)}\mathcal{S}(t) = \mathrm{e}^{tZ_s} = \mathrm{e}^{tZ_s^{[1]} + t^2 Z_s^{[2]} + \cdots}\,. \tag{2.3}$$

Proof. Similar as for Lemma 2.2. Here, the induction is started by the BCH expansion of $\mathrm{e}^{-t(A+B)}\,\mathrm{e}^{tA_1}$, and the induction argument is analogous. □

Lemma 2.4. *The following assertions are equivalent, for $t \to 0$ and with $W_q \in \mathbb{C}\langle A, B\rangle$:*

$$\mathrm{e}^{t(A+B)} = \mathcal{S}(t) + t^q\, W_q + \mathcal{O}(t^{q+1}), \tag{2.4a}$$

$$\mathrm{e}^{-t(A+B)}\mathcal{S}(t) = 1 - t^q\, W_q + \mathcal{O}(t^{q+1}). \tag{2.4b}$$

Proof. Assume that (2.4a) holds. Multiplication by $\mathrm{e}^{-t(A+B)}$ gives

$$1 - \mathrm{e}^{-t(A+B)}\mathcal{S}(t) = \mathrm{e}^{-t(A+B)}\, t^q\, W_q + \mathcal{O}(t^{q+1}),$$

and the identity

$$\mathrm{e}^{-t(A+B)}\, t^q\, W_q = \big(1 - t(A+B) + \mathcal{O}(t^2)\big)\, t^q\, W_q = t^q\, W_q + \mathcal{O}(t^{q+1})$$

implies (2.4b). The reverse implication is proved in an analogous way. □

We also make use of the following equivalence:

Lemma 2.5. *For a BCH-like expansion*

$$\mathrm{e}^{tZ} = \mathrm{e}^{tZ^{[1]} + t^2 Z^{[2]} + \cdots}$$

of a time-dependent operator $Z = Z(t)$ and $q \in \mathbb{N}$, the following assertions are equivalent:

$$\mathrm{e}^{tZ} = 1 + t^q\, W_q + \mathcal{O}(t^{q+1}) \quad \textit{for} \;\; t \to 0, \tag{2.5a}$$

$$Z^{[1]} = \ldots = Z^{[q-1]} = 0. \tag{2.5b}$$

Here, $W_q = Z^{[q]}$ is a homogeneous Lie element of degree q.

Proof. Evidently, (2.5b) implies (2.5a). The reverse implication follows by an induction argument: Taylor expansion of e^{tZ} yields

$$\mathrm{e}^{tZ^{[1]} + t^2 Z^{[2]} + t^3 Z^{[3]} + \cdots} =$$

$$= 1 + t\, Z^{[1]} + t^2\big(\tfrac{1}{2}\,[(Z^{[1]})^2 + Z^{[2]}\big) + t^3\big(\tfrac{1}{6}\,[(Z^{[1]})^3 + Z^{[1]}Z^{[2]} + Z^{[3]}\big) + \ldots$$

Clearly, the coefficient of t^k involves a term $Z^{[k]}$ and certain products of terms $Z^{[\ell]}$ with $\ell < k$. Comparison with (2.5a) for arbitrary $q \geq 1$ yields

$$Z^{[1]} = 0 \;\Rightarrow\; Z^{[2]} = 0 \;\Rightarrow\; \ldots \;\Rightarrow\; Z^{[q-1]} = 0,$$

and hence $W_q = Z^{[q]}$, as asserted. □

These preparations permit us to state the following result.

Theorem 2.6. *Suppose that a splitting operator* $\mathcal{S}(t) = \mathcal{S}_{1\ldots s}(t)$ *of the form* (1.2a) *satisfies the order conditions* (1.7) *for some* $p \geq 1$*. Then, the local error operator* $\mathcal{L}(t) = \mathcal{S}(t) - \mathrm{e}^{t(A+B)}$ *satisfies*

$$\mathcal{L}(t) = t^{p+1}\, W_{p+1} + \mathcal{O}(t^{p+2}) \quad \textit{for} \;\; t \to 0. \tag{2.6}$$

Here, W_{p+1} *is a homogeneous Lie element of degree* $p+1$*.*

Proof. Using Lemma 2.4 with $q = p+1$ implies that (2.6) is equivalent to

$$\mathrm{e}^{-t(A+B)}\mathcal{S}(t) = 1 - t^{p+1}\, W_{p+1} + \mathcal{O}(t^{p+2}).$$

Lemma 2.3 shows that $\mathrm{e}^{-t(A+B)}\mathcal{S}(t)$ admits a BCH-like expansion (2.3), and therefore Lemma 2.5 applies. In particular, W_{p+1} is a homogeneous Lie element of degree $p+1$, as asserted. □

3. ALGORITHMIC GENERATION OF ORDER CONDITIONS

In terms of derivatives of the defect (1.4), Theorem 2.6 states: If the conditions (1.7) for some order $q-1$ hold, that is, if

$$\mathcal{D}(0) = \tfrac{\mathrm{d}}{\mathrm{d}t}\, \mathcal{D}(0) = \ldots = \tfrac{\mathrm{d}^{q-2}}{\mathrm{d}t^{q-2}}\, \mathcal{D}(0) = 0, \tag{3.1a}$$

then $\tfrac{\mathrm{d}^{q-1}}{\mathrm{d}t^{q-1}}\, \mathcal{D}(0)$ is a Lie element of degree q, i.e., it is a linear combination

$$\tfrac{\mathrm{d}^{q-1}}{\mathrm{d}t^{q-1}}\, \mathcal{D}(0) = \sum_{\ell=1}^{\ell_q} c_{q,\ell}\, C_{q,\ell}(A,B). \tag{3.1b}$$

Here, $C_{q,\ell}(A,B)$ is the unique commutator of degree q represented by the Lyndon word $L_{q,\ell}(A,B)$, see Sec. 1.2. We stress that $\tfrac{\mathrm{d}^{q-1}}{\mathrm{d}t^{q-1}}\, \mathcal{D}(0)$ is indeed *not* of the form (3.1b) if (3.1a) does not hold; cf. Example 1.2.

Making use of assertion (3.1), a non-redundant set of conditions for order p can be generated in an elementary way on the basis of [4] expansion (1.9).

[4] One may also proceed from Algorithm 1, but this turns out to be much less efficient in computational practice.

Algorithm 2. Generate conditions for order p.

- Define the conditions for order 1,

$$\begin{aligned} c_{1,1} &:= a_1 + \ldots + a_s - 1 = 0, \\ c_{1,2} &:= b_1 + \ldots + b_s - 1 = 0. \end{aligned}$$

- Set $A_j := a_j\, A$, $B_j := b_j\, B$, $j = 1 \ldots s$, and $H := \sum_{j=1}^{s}(A_j + B_j)$.
- Set $\mathcal{S}^{(1)} := \frac{\mathrm{d}}{\mathrm{d}t}\,\mathcal{S}(0) = H$.
- For $q = 2 \ldots p$:
  - Compute the expression [5] $\mathcal{S}^{(q)} := \frac{\mathrm{d}^q}{\mathrm{d}t^q}\,\mathcal{S}(0)$ by means of (1.9a).
  - Compute $\mathcal{D}^{(q-1)} := \frac{\mathrm{d}^{q-1}}{\mathrm{d}t^{q-1}}\,\mathcal{D}(0) = \mathcal{S}^{(q)} - H\,\mathcal{S}^{(q-1)}$ (see (1.9b)).
  - Extract the coefficients $c_{q,\ell}$, $\ell = 1 \ldots \ell_q$, in $\mathcal{D}^{(q-1)}$ of the ℓ_q Lyndon words $L_{q,\ell}(A, B)$ of degree q.
  - Define the conditions for order q,

$$c_{q,\ell} = 0, \quad \ell = 1 \ldots \ell_q.$$

  The $c_{q,\ell}$ are homogeneous polynomials of degree q in the variables a_j and b_j, with integer coefficients.

Algorithm 2 can be modified in several ways. We may also compute $\mathcal{D}^{(q-1)}$ according to (1.9c),

$$\frac{\mathrm{d}^{q-1}}{\mathrm{d}t^{q-1}}\,\mathcal{D}(0) = \mathcal{S}^{(q)} - H^q. \tag{3.2}$$

Also, in both versions we may set $H = A + B$. This leads to a sparser system of (nonhomogeneous) equations. For version (3.2) with $H = A + B$ we obtain equations of the form *homogeneous polynomial*$(\cdots) = 1$, where the coefficient 1 stems from H^q. This version appears to perform best, and it produces a sparser output than the other variants. Furthermore, coefficients can be extracted term by term from the multinomial representation (1.9a) of $S^{(q)}$ and summed up, without explicitly computing $S^{(q)}$.

We have implemented these variants of Algorithm 2 in Maple 17 [6] using the packages `combinat`, `combstruct` (for some elementary combinatorics), and `Physics` (for representing and manipulating expressions involving non-commuting symbolic variables).

Example 3.1. We list the resulting 5 equations for $s = p = 3$ (algorithmic version based on (3.2) with $H = A + B$):

$$\begin{aligned} a_1 + a_2 + a_3 &= 1, \\ b_1 + b_2 + b_3 &= 1, \\ 2\,a_2\,b_1 + 2\,a_3\,b_1 + 2\,a_3\,b_2 &= 1, \\ 3\,a_2\,b_1^2 + 3\,a_3\,b_1^2 + 6\,a_3\,b_1\,b_2 + 3\,a_3\,b_2^2 &= 1, \\ 3\,a_2^2\,b_1 + 6\,a_2\,a_3\,b_1 + 3\,a_3^2\,b_1 + 3\,a_3^2\,b_2 &= 1. \end{aligned}$$

[5] $\mathcal{S}^{(q)}$ is stored in memory for use in the subsequent step.

[6] Maple is a Trademark of MapleSoft, Inc.

For this system, the general one-dimensional solution manifold is easily determined using Maple. A rational solution is given by

$$\begin{aligned} a_1 &= \tfrac{7}{24}, \quad a_2 = \tfrac{3}{4}, \quad a_3 = -\tfrac{1}{24}, \\ b_1 &= \tfrac{2}{3}, \quad b_2 = -\tfrac{2}{3}, \quad b_3 = 1. \end{aligned}$$

Example 3.2. For $s = 7$ and $p = 5$ we obtain 14 polynomial equations of degree ≤ 5 in 14 variables. This system involves already about 1.800 individual terms (for version (3.2) with $H = A + B$).

Some new results concerning the numerical determination of coefficients from our order conditions will be reported in [2].

Remark 3.3. Using a computer algebra system is convenient and straightforward, but not very efficient in so far as for higher values of s and p a rapidly growing number of terms is generated. These could be coded in a more efficient way in any programming language supporting appropriate data structures for representing the terms in (1.9a). Up to now we have not implemented such a code but expect it to perform significantly better. Its complexity is the same, but computing time and memory requirements are expected to be smaller.

4. SYMMETRIES

Smaller systems of equations are obtained if we restrict ourselves to schemes with special symmetries.

4.1. SCHEMES WITH REFLECTED COEFFICIENTS

Assume

$$a_j \equiv b_{s+1-j}, \quad \text{and} \quad b_j \equiv a_{s+1-j}. \tag{4.1}$$

In this case, the two first-order conditions reduce to a single equation. The number of higher-order conditions appears reduced: Consider

$$\begin{aligned} \mathcal{S}(t) &= \mathrm{e}^{t\,a_1 A}\,\mathrm{e}^{t\,b_1 B}\,\cdots\,\mathrm{e}^{t\,a_s A}\,\mathrm{e}^{t\,b_s B} = \\ &= \mathrm{e}^{t\,a_1 A}\,\mathrm{e}^{t\,b_1 B}\,\cdots\,\mathrm{e}^{t\,b_1 A}\,\mathrm{e}^{t\,a_1 B}; \end{aligned} \tag{4.2a}$$

interchanging the role of A and B we obtain

$$\tilde{\mathcal{S}}(t) = \mathrm{e}^{t\,a_1 B}\,\mathrm{e}^{t\,b_1 A}\,\cdots\,\mathrm{e}^{t\,b_1 B}\,\mathrm{e}^{t\,a_1 A}. \tag{4.2b}$$

Since A and B are arbitrary non-commuting operators, the order conditions for the the schemes (4.2a) and (4.2b) are equivalent. These conditions are certain expressions in A and B or B and A, respectively, and therefore due to symmetry they are redundant. A look at Table 1 shows that it is now sufficient to consider the coefficients of a reduced set of Lyndon words specified in Table 2.

Algorithm 2 can be modified by choosing the reduced set of variables and imposing the appropriately reduced set of equations.

Table 2. A reduced set of Lyndon words

q	#	Lyndon words $L_{q,\ell}(A,B)$ of degree q
1	1	A
2	1	AB
3	1	A^2B
4	2	A^3B, A^2B^2
5	3	A^4B, A^3B^2, A^2BAB
6	5	A^5B, A^4B^2, A^3BAB, A^3B^3, A^2BAB^2
		...

4.2. SYMMETRIC SCHEMES

Symmetric schemes satisfying

$$a_j \equiv a_{s+1-j}, \quad b_j \equiv b_{s-j}, \quad \text{and} \quad b_s = 0, \tag{4.3a}$$

are of particular relevance for accurate long-time integration, e.g., for time-reversible evolution equations; see for instance [9,11]. A splitting operator (1.2a) with symmetric coefficients (4.3a) satisfies

$$\mathcal{S}^{-1}(t) = \mathcal{S}(-t). \tag{4.3b}$$

Lemma 2.2 implies

$$\begin{aligned}\mathcal{S}(-t) &= \mathrm{e}^{-tX_s^{[1]} + t^2 X_s^{[2]} - t^3 X_s^{[3]} + t^4 X_s^{[4]} - \dots},\\ \mathcal{S}^{-1}(t) &= \mathrm{e}^{-tX_s^{[1]} - t^2 X_s^{[2]} - t^3 X_s^{[3]} - t^4 X_s^{[4]} - \dots},\end{aligned}$$

and therefore all coefficients $X_s^{[2\ell]}$, $\ell = 1,2,\dots$ vanish.

A symmetric scheme has even order, as already stated in [14]. In the context of our approach, this can be seen by modifying the argument given in the proof of Lemma 2.4: Assume that the scheme has order p. Then, making use of (4.3b), we obtain

$$\begin{aligned}\mathcal{S}(t) - \mathrm{e}^{t(A+B)} &= t^{p+1}\, W_{p+1}(A,B) + \mathcal{O}(t^{p+2}),\\ \mathrm{e}^{-t(A+B)}\mathcal{S}(t) - I &= t^{p+1}\, W_{p+1}(A,B) + \mathcal{O}(t^{p+2}),\\ \mathrm{e}^{-t(A+B)}\mathcal{S}(t) - \mathcal{S}^{-1}(t)\,\mathcal{S}(t) &= t^{p+1}\, W_{p+1}(A,B) + \mathcal{O}(t^{p+2}),\\ \big(\mathrm{e}^{-t(A+B)} - \mathcal{S}(-t)\big)\,\mathcal{S}(t) &= t^{p+1}\, W_{p+1}(A,B) + \mathcal{O}(t^{p+2}),\\ \mathrm{e}^{-t(A+B)} - \mathcal{S}(-t) &= t^{p+1}\, W_{p+1}(A,B) + \mathcal{O}(t^{p+2}).\end{aligned} \tag{4.4a}$$

Here, $W_{p+1}(A,B)$ is a Lie element of degree $p{+}1$, a well-defined expression in A and B. On the other hand,

$$\mathrm{e}^{-t(A+B)} - \mathcal{S}(-t) = -(-t)^{p+1}\, W_{p+1}(-A,-B) + \mathcal{O}(t^{p+2}). \tag{4.4b}$$

If p is odd, then

$$-(-t)^{p+1}\, W_{p+1}(-A,-B) = -t^{p+1}\, W_{p+1}(A,B),$$

so that comparing (4.4b) with (4.4a) shows that $W_{p+1}(A,B) = 0$.

This means that the conditions for even order, i.e., the terms involving Lie elements of even degree, can be ignored, because for p even, the conditions for order p are automatically satisfied provided the conditions for order $p-1$ are valid.

Example 4.1. Setting up the conditions for a symmetric scheme of order $s=4$, we obtain 2 first-order conditions (ensuring order 2) and 2 third-order conditions (ensuring order 4) for the 4 variables a_1, b_1, a_2, b_2. Solving this system we obtain a pair of complex solutions, and a real solution corresponding to the scheme derived in [14].

For a symmetric scheme with $s=11$, with 11 independent coefficients, we obtain 10 equations for order $p=6$. This system involves about 3.700 individual terms (for version (3.2) with $H=A+B$).

Remark 4.2. With some modifications, our approach can be adapted for the case of composition schemes, e.g., compositions of stages of Strang type as in [5, 6].

REFERENCES

[1] W. Auzinger, O. Koch, M. Thalhammer, *Defect-based local error estimators for splitting methods, with application to Schrödinger equations, Part II. Higher order methods for linear problems*, J. Comput. Appl. Math. **255** (2013), 384–403.

[2] W. Auzinger et. al., *Adaptive time splitting methods*, in preparation.

[3] S. Blanes, F. Casas, *On the convergence and optimization of the Baker-Campbell--Hausdorff formula*, Lin. Alg. Appl. **378** (2004), 135–158.

[4] S. Blanes, P.C. Moan, *Practical symplectic partitioned Runge-Kutta and Runge-Kutta--Nyström methods*, J. Comput. Appl. Math. **142** (2002), 313–330.

[5] S. Blanes, F. Casas, P. Chartier, A. Murua, *Optimized high-order splitting methods for some classes of parabolic equations*, Math. Comp. **82** (2013), 1559–1576.

[6] S. Blanes, F. Casas, A. Murua, *Splitting and composition methods in the numerical integration of differential equations*, Bol. Soc. Esp. Mat. Apl. **45** (2008), 89–145.

[7] P. Chartier, A. Murua, *An algebraic theory of order*, M2AN Math. Model. Numer. Anal. **43** (2009), 607–630.

[8] J.-P. Duval, *Géneration d'une section des classes de conjugaison et arbre des mots de Lyndon de longueur bornée*, Theoret. Comput. Sci. **60** (1988), 255–283.

[9] E. Hairer, C. Lubich, G. Wanner, *Geometrical Numerical Integration – Stucture--Preserving Algorithms for Ordinary Differential Equations*, 2nd ed., Springer-Verlag, Berlin, Heidelberg, 2006.

[10] E. Hansen, A. Ostermann, *Exponential splitting for unbounded operators*, Math. Comp. **78** (2009), 1485–1496.

[11] M. Suzuki, *General theory of higher-order decomposition of exponential operators and symplectic integrators*, Phys. Lett. A **165** (1992), 387–395.

[12] M. Thalhammer, *High-order exponential operator splitting methods for time-dependent Schrödinger equations*, SIAM J. Numer. Anal. **46** (2008), 2022–2038.

[13] Z. Tsuboi, M. Suzuki, *Determining equations for higher-order decompositions of exponential operators*, Int. J. Mod. Phys. B **09** (1995), 3241–3268.

[14] H. Yoshida, *Construction of higher order symplectic integrators*, Phys. Lett. A **150** (1990), 262–268.

Winfried Auzinger
w.auzinger@tuwien.ac.at

Vienna University of Technology
Institute for Analysis and Scientific Computing
Wiedner Hauptstrasse 8–10/E101, A-1040 Vienna, Austria

Wolfgang Herfort
w.herfort@tuwien.ac.at

Vienna University of Technology
Institute for Analysis and Scientific Computing
Wiedner Hauptstrasse 8–10/E101, A-1040 Vienna, Austria

Received: September 4, 2013.
Revised: March 3, 2014.
Accepted: March 3, 2014.