Journal of Computational and Applied Mathematics 291 (2016) 380–390

Contents lists available at ScienceDirect

Journal of Computational and Applied Mathematics

journal homepage: www.elsevier.com/locate/cam

Efficient numerical integration of Nth-order non-autonomous linear differential equations

Philipp Bader[a,*], Sergio Blanes[b], Fernando Casas[c], Enrique Ponsoda[b]

[a] Department of Mathematics and Statistics, La Trobe University, 3086 Bundoora VIC, Australia

[b] Instituto de Matemática Multidisciplinar, Building 8G, second floor, Universitat Politècnica de València. 46022 Valencia, Spain

[c] Departament de Matemàtiques and IMAC, Universitat Jaume I, 12071 Castellón, Spain

ARTICLE INFO

Article history:
Received 15 October 2014

Keywords:
Higher order linear differential equation
Nonautonomous coefficients
Magnus expansion

ABSTRACT

We consider the numerical integration of high-order linear non-homogeneous differential equations, written as first order homogeneous linear equations, and using exponential methods. Integrators like Magnus expansions or commutator-free methods belong to the class of exponential methods showing high accuracy on stiff or oscillatory problems, but the computation of the exponentials or their action on vectors can be computationally costly. The first order differential equations to be solved present a special algebraic structure (associated with the companion matrix) which allows to build new methods (hybrid methods between Magnus and commutator-free methods). The new methods are of similar accuracy as standard exponential methods with a reduced complexity. Additional parameters can be included into the scheme for optimization purposes. We illustrate how these methods can be obtained and present several sixth-order methods which are tested in several numerical experiments.

1. Introduction

In this work we consider the numerical integration of the Nth-order non-autonomous and non-homogeneous linear differential equations

$$\mathcal{L}(t)x = g(t), \tag{1}$$

where $\mathcal{L}(t)$ is a non-autonomous linear operator

$$\mathcal{L}(t)x = x^{(N)} + f_{N-1}(t)x^{(N-1)} + \cdots + f_1(t)x' + f_0(t)x, \tag{2}$$

and $x, g \in \mathbb{C}^{m\times d}$, $f_i \in \mathbb{C}^{m\times m}$, $x^{(i)} \equiv \frac{d^i x}{dt^i}$.

It is usual to write Eq. (1) as a first order non-homogeneous linear system of equations. However, to simplify the analysis, we write the non-homogeneous problem as a[1] $(N+1)$-dimensional homogeneous problem by introducing $z = (y, 1)^T \in$

* Corresponding author.
E-mail addresses: p.bader@latrobe.edu.au (P. Bader), serblaza@imm.upv.es (S. Blanes), fernando.casas@mat.uji.es (F. Casas), eponsoda@imm.upv.es (E. Ponsoda).

[1] For simplicity in the presentation and without loss of generality, we will take $m = d = 1$ and $t_0 = 0$.

http://dx.doi.org/10.1016/j.cam.2015.02.052

$\mathbb{C}^{N+1}, y \equiv (y_1, \dots, y_N)^T = (x, \dots, x^{(N-1)})^T$, $G(t) = (0, \dots, 0, g(t))^T \in \mathbb{C}^N$ which satisfies the homogeneous linear equation

$$z' = M(t)z, \qquad z(0) = (y(0), 1)^T, \tag{3}$$

with

$$M(t) = \begin{pmatrix} A(t) & G(t) \\ 0_N^T & 0_1 \end{pmatrix} = \begin{pmatrix} 0 & 1 & 0 & \cdots & & 0 & 0 \\ & 0 & 1 & \ddots & & 0 & 0 \\ \vdots & & \ddots & \ddots & & \vdots & \vdots \\ & & & & 1 & 0 & 0 \\ & & & & 0 & 1 & 0 \\ -f_0 & & \cdots & & -f_{N-2} & -f_{N-1} & g \\ 0 & 0 & & \cdots & 0 & 0 & 0 \end{pmatrix}, \tag{4}$$

where 0_N is the zero vector of dimension N, and $A \in \mathbb{C}^{N\times N}$ is the companion matrix.

The second order autonomous matrix differential equations of Apostol–Kolodner type [1,2] and its generalization to higher order [3,4]

$$x^{(N)} = Mx,$$

belong to this class. These equations have been extensively studied and their formal solution can be written in closed form. However, if the matrix M is time-dependent, a numerical method is required.

On the other hand, high order nonlinear differential equations of the form

$$F(t, x, x', \dots, x^{(N)}) = 0$$

arise in many fields in physics and engineering (see [5–8] and references therein) either with initial or boundary conditions. The shooting method for the problem with boundary conditions usually requires the numerical integration of a non-autonomous linear equation. The method of quasilinearization also requires the numerical integration of non-autonomous linear equations of the form (1), iteratively [5].

We also remark that the numerical integration of a close to a linear problem

$$\mathcal{L}(t)x = g(t) + \epsilon \mathcal{N}(t, x), \tag{5}$$

where $|\epsilon| \ll 1$ and $\mathcal{N}$ is a nonlinear operator depending on $t, x, \dots, x^{(N-1)}$, can be efficiently carried out if the linear part is numerically integrated to a relatively high accuracy and separately from the non-linear part. Then, splitting methods for perturbed problems can be used and have shown a high performance [9].

In the autonomous situation, the solution of (3) can be written in closed form

$$z(t) = \exp(tM)z(0), \tag{6}$$

or, equivalently,

$$y(t) = e^{tA}y(0) + t\varphi(tA)g = y(0) + t\varphi(tA)(Ay(0) + g), \tag{7}$$

where $\varphi(z) = (e^z - 1)/z$. In some cases it can be more convenient, from the numerical point of view, to use approximations to the exponential matrix acting on a vector and in some other cases the use of the φ matrix acting on a vector is preferable [10–14].

If the problem is explicitly time-dependent a closed-form solution is not available and numerical methods have to be used on a time mesh (for simplicity, we consider a constant time step: $t_0 = 0, t_1 = h, \dots, t_N = Nh = t_f$). Standard methods like Runge–Kutta, multistep or extrapolation methods are, in general, not suitable for problems where the matrix A has an algebraic structure (e.g. if $f_{N-1} = 0$ the system is volume preserving) or if the solution is oscillatory.

Alternatively, one can use exponential methods like Magnus and Fer expansions or commutator-free methods. They preserve the algebraic structure of the exact solution and show a high performance for stiff and oscillatory problems. The main drawback is the computational cost to compute the action of the exponentials on vectors. While the computation of the exponential of a companion matrix acting on a vector can be carried out at a moderate cost for relatively small time steps, the exponents appearing in Magnus and Fer methods are much more involved (and computationally costly) due to their reduced sparsity.

Commutator-free methods correspond, for this problem, to a composition of exponentials of companion matrices and then can be computed efficiently. The main difficulty is that at least two exponentials are necessary to obtain fourth-order methods in the time step, h, and at least five for sixth-order methods.[2] In addition, for methods of order greater than four,

[2] A four-exponential sixth-order method exists, but it shows a very poor performance and it is not recommended in practice.

at least one of these companion matrices has to be integrated backwards in time, and this could cause step-size restrictions for stiff problems.

In this work, we analyze the structure of the elements associated with the Lie algebra generated by the matrix $M(t)$ evaluated at a given set of points, say $M_1 = M(\tau_1), \ldots, M_k = M(\tau_k)$ for some values of $\tau_1, \ldots, \tau_k$. By definition, linear combinations or commutators of elements of a given Lie algebra remain in the Lie algebra. In addition, we observe that

$$C_\sigma = \sum_{j=1}^{k} a_j M_j = \begin{pmatrix} 0 & \sigma & 0 & \cdots & & 0 & 0 \\ & 0 & \sigma & \ddots & & 0 & 0 \\ \vdots & & \ddots & \ddots & & \vdots & \vdots \\ & & & & \sigma & 0 & 0 \\ & & & & 0 & \sigma & 0 \\ -\tilde{f}_0 & & \cdots & & -\tilde{f}_{N-2} & -\tilde{f}_{N-1} & \tilde{g} \\ 0 & 0 & & \cdots & 0 & 0 & 0 \end{pmatrix}, \tag{8}$$

where

$$\sigma = \sum_{j=1}^{k} a_j, \qquad \tilde{f}_i = \sum_{j=1}^{k} a_j f_i(\tau_j), \qquad \tilde{g} = \sum_{j=1}^{k} a_j g(\tau_j)$$

which we call *companion matrix* when $\sigma \neq 0$. Notice that, when $\sigma = 0$ the computation of $\exp(C_\sigma)$ is trivial.

The following properties for the exponential of matrices or their action on vectors will be used in this work:

Given $B_i \in \mathbb{C}^{k_2 \times k_i},\ i = 1, 2, 3$ with $(k_1 + k_2 + k_3 = k)$, we have that

$$\exp(B) \equiv \exp \begin{pmatrix} & 0_{k_1,k} & \\ B_1 & B_2 & B_3 \\ & 0_{k_3,k} & \end{pmatrix} = I_{k,k} + \begin{pmatrix} & 0_{k_1,k} & \\ \varphi(B_2)B_1 & \varphi(B_2)B_2 & \varphi(B_2)B_3 \\ & 0_{k_3,k} & \end{pmatrix}. \tag{9}$$

If $k_2 \ll k$ then, since $B_2 \in \mathbb{C}^{k_2 \times k_2}$, it is very simple and cheap to compute $\varphi(B_2)$ and, consequently, $\exp(B)$ or its action on a vector. Given $v = (v_{k_1}, v_{k_2}, v_{k_3})^T \in \mathbb{C}^k,\ B = (B_1, B_2, B_3) \in \mathbb{C}^{k_2 \times k}$ and denoting $B \cdot v = B_1 v_{k_1} + B_2 v_{k_2} + B_3 v_{k_3} \in \mathbb{C}^{k_2}$, we have that

$$\exp(B)v = (v_{k_1}, v_{k_2} + \varphi(B_2)(B \cdot v), v_{k_3}).$$

For the schemes derived below, the matrices B_2 will have small norms, typically $B_2 = \mathcal{O}(h^s)$ with $s \geq 2$, and the matrix $\varphi(B_2)$ can then be approximated, for example, using only the first few terms of its Taylor expansion.

We also stress the following properties of some elements of this Lie algebra:

1. If $\sigma = 0$, then $C_{\sigma=0}$ is a matrix with only one row with non zero elements ($k_2 = 1$).
2. The commutator
$$[C_{\sigma_1}, C_{\sigma_2}] = C_{\sigma_1} C_{\sigma_2} - C_{\sigma_2} C_{\sigma_1}$$
is a matrix with only two rows with non zero elements and it is also trivial to compute the exponential of this matrix ($k_2 = 2$).
3. Each additional commutator introduces a new non-empty row in the matrix.

We analyze how to obtain numerical methods at different orders by considering exponentials of elements of the Lie algebra such that the evaluation of the exponential for these elements can be cheaply and efficiently computed. Apart from the order conditions, there are additional constraints to be considered, for example, sixth-order methods without commutators necessarily involve the exponential of a companion matrix, e^{C_σ}, with a negative σ.

Allowing exponentials of elements of the Lie algebra with a low computational cost which includes certain commutators, we derive sixth-order methods with positive σ.

1.1. Numerical integration by standard methods: Runge–Kutta methods

We consider Runge–Kutta (RK) methods as a representative of standard numerical integrators. The general class of s-stage (explicit or implicit) Runge–Kutta methods are characterized by the real numbers a_{ij}, b_i ($i, j = 1, \ldots, s$) and $c_i = \sum_{j=1}^{s} a_{ij}$. For this linear problem they take the form

$$Z_i = z_n + h \sum_{j=1}^{s} a_{ij} M_j Z_j, \quad i = 1, \ldots, s$$

$$z_{n+1} = z_n + h \sum_{i=1}^{s} b_i M_i Z_i, \tag{10}$$

where $M_i = M(t_n + c_i h)$. If $a_{ij} = 0,\ j \geq i$ then the method is explicit and one can compute (and store) the vectors $Z_1, \ldots, Z_s$ sequentially. Otherwise, the method is implicit and one has to solve the linear system of equations

$$\begin{pmatrix} I - ha_{11}M_1 & -ha_{12}M_2 & \cdots & -ha_{1s}M_s \\ -ha_{21}M_1 & I - ha_{22}M_2 & \cdots & -ha_{2s}M_s \\ \vdots & \vdots & & \vdots \\ -ha_{s1}M_1 & -ha_{s2}M_2 & \cdots & I - ha_{ss}M_s \end{pmatrix} \begin{pmatrix} Z_1 \\ Z_2 \\ \vdots \\ Z_s \end{pmatrix} = \begin{pmatrix} z_n \\ z_n \\ \vdots \\ z_n \end{pmatrix}.$$

Explicit RK methods require only s products MZ and they need to store s vectors ($M_i Z_i,\ i = 1, \ldots, s$). In this sense, RK methods can be considered as very cheap methods. However, in general, they require s evaluations of the functions $f_i(t)$ (some methods require less number of evaluations and this depends on the nodes, c_i, of the method) and, since they can be considered as polynomial approximations to the solution, a poor performance is expected for stiff and oscillatory problems.

On the other hand, implicit RK methods can reach order $2s$ and are suitable for stiff problems, but they require to compute the inverse of a matrix of dimension $(sN) \times (sN)$ whose computational cost, in general, is s^3 times more expensive than the inverse of a matrix of dimension $N \times N$ (e.g. for sixth-order methods with $s = 3$ we have that the inverse of this matrix is about $s^3 = 27$ times more expensive than the inverse of a matrix of dimension $N \times N$).

Exponential methods like Magnus integrators or commutator-free methods usually show a high accuracy and in this work, we propose new composition of exponentials with similar accuracy at lower computational cost. The order conditions for the new composition methods are obtained by equating with the formal solution given by the Magnus series expansion in a similar way as the Taylor method is used to obtain the order conditions for RK method after expanding all terms. For this reason, we briefly review some results for Magnus integrators.

2. Magnus based integrators

Given the homogeneous linear equation (3), with formal solution, $z(t) = \Phi(t)z(0)$, the Magnus expansion expresses the fundamental matrix solution in terms of a single exponential as [15]

$$\Phi(t) = \exp(\Omega(t)), \qquad \Omega(t) = \sum_{k=1}^{\infty} \Omega_k(t)$$

whose terms, $\Omega_k(t)$, are linear combinations of integrals and nested commutators involving the matrix M at different times. Thus, the first terms read

$$\Omega_1(t) = \int_0^t M(t_1)dt_1, \qquad \Omega_2(t) = \frac{1}{2}\int_0^t dt_1 \int_0^{t_1} dt_2 [M(t_1), M(t_2)], \ldots . \tag{11}$$

The algebraic problem to numerically approximate Ω considerably simplifies if we use the graded free Lie algebra generated by $\{\alpha_1, \ldots, \alpha_s\}$ [16] where

$$\alpha_{i+1} = \frac{h^{i+1}}{i!} \left. \frac{d^i M(t)}{dt^i} \right|_{t=h/2} \tag{12}$$

$i = 0, 1, \ldots, s-1$. Here $\alpha_i = \mathcal{O}(h^i)$ and then it can be considered as an element with grade $i = 1, 2, \ldots, s$ respectively. In particular, up to second-order, we have[3] $\Omega^{[2]} = \alpha_1$, up to fourth-order, we have

$$\Omega^{[4]} = \alpha_1 - \frac{1}{12}[\alpha_1, \alpha_2], \tag{13}$$

and up to sixth-order

$$\Omega^{[6]} = \alpha_1 + \frac{1}{12}\alpha_3 - \frac{1}{12}[12] + \frac{1}{240}[23] + \frac{1}{360}[113] - \frac{1}{240}[212] + \frac{1}{720}[1112] \tag{14}$$

where $[ij \ldots kl]$ represents the nested commutator $[\alpha_i, [\alpha_j, [\ldots, [\alpha_k, \alpha_l] \ldots]]]$. However, from the computational point of view, it is more convenient to replace the elements α_i (derivatives) by a linear combination of the matrix $M(t)$ evaluated at the nodes of a given quadrature rule (integrals). For example, it is possible to build methods of order $2s$ with only s symmetric collocation points [17]. In order to obtain methods which can be easily used with any quadrature rule, we introduce the averaged (or generalized momentum) matrices for the interval $[t_n, t_{n+1}]$

$$A^{(i)}(h) \equiv \frac{1}{h^i} \int_{t_n}^{t_n+h} \left(t - t_{1/2}\right)^i A(t)dt = \frac{1}{h^i} \int_{-h/2}^{h/2} t^i A(t + t_{1/2})dt, \tag{15}$$

for $i = 0, \ldots, s-1$ where $t_{1/2} = t_n + h/2$.

3 We denote by $\Omega^{[p]}$ an approximation (no unique) to the solution Ω up to order h^p, i.e. $\Omega^{[p]} = \Omega + O(h^{p+1})$.

To second order, we can take $\alpha_1 = A^{(0)}$ (neglecting higher order terms), to order four, we can set (see [18] and references therein)

$$\alpha_1 = A^{(0)}, \qquad \alpha_2 = 12A^{(1)}, \tag{16}$$

and to order six

$$\alpha_1 = \frac{9}{4}A^{(0)} - 15A^{(2)}, \qquad \alpha_2 = 12A^{(1)}, \qquad \alpha_3 = -15A^{(0)} + 180A^{(2)}. \tag{17}$$

If $b_i, c_i,\ i = 1, \ldots, k$ denote the weights and nodes of a given quadrature rule of order $p \geq 2s$, then the momentum matrices can be computed as

$$A^{(i)} = h\sum_{j=1}^{k} b_j \left(c_j - \frac{1}{2}\right)^i A_j, \quad i = 0, \ldots, s-1, \tag{18}$$

with $A_j \equiv A(t_n + c_j h)$, and the corresponding numerical methods will remain of order $2s$.

Notice that, while $\Omega^{[2]}$ has the same sparsity as A, this is not the case for $\Omega^{[p]}$ with $p > 2$, and then the computational cost of the exponential (or its action on a vector) can grow considerably.

To circumvent this problem we can consider, for example, commutator-free methods which we briefly present.

2.1. Commutator-free Magnus integrators

Commutator-free (CF) methods can be a simple and efficient alternative to solve the non-autonomous problem (3). These methods can be written, for one time step h and m stages, as the composition

$$z_{n+1} = \exp(hC_{\sigma_m}) \cdots \exp(hC_{\sigma_2})\ \exp(hC_{\sigma_1}) z_n \tag{19}$$

here where each C_{σ_k} has the structure given in (8) and must satisfy the consistency condition $\sum_{k=1}^{m} \sigma_k = 1$.

Second order methods. Second order methods can be obtained with the very simple scheme

$$z_{n+1} = \exp(\alpha_1) z_n = \exp(M^{(0)}) z_n, \tag{20}$$

which could also be categorized as a second order Magnus or Fer integrator, and where we can approximate $M^{(0)}$, e.g., using the midpoint or the trapezoidal rule, i.e.,

$$M^{(0)} = hM(t_n + h/2) \quad \text{or} \quad M^{(0)} = \frac{h}{2}(M(t_n) + M(t_n + h)).$$

Fourth order methods. An additional exponential is needed for fourth-order methods. A simple two-stage method ($m = 2$) is given by [19] (using the relation (16))

$$\begin{aligned} z_{n+1} &= \exp\left(\frac{1}{2}\alpha_1 + \frac{1}{6}\alpha_2\right) \exp\left(\frac{1}{2}\alpha_1 - \frac{1}{6}\alpha_2\right) z_n \\ &= \exp\left(\frac{1}{2}M^{(0)} + 2M^{(1)}\right) \exp\left(\frac{1}{2}M^{(0)} - 2M^{(1)}\right) z_n \end{aligned} \tag{21}$$

where we can take, for example

$$M^{(0)} = \begin{cases} \dfrac{h}{6}\Big(M(t) + 4M(t + h/2) + M(t + h)\Big) & \text{Simpson rule,} \\ \dfrac{h}{2}\Big(M(t + c_1 h) + M(t + c_2 h)\Big) & \text{Gaussian quadrature rule,} \end{cases}$$

$$M^{(1)} = \begin{cases} \dfrac{h}{12}\Big(M(t + h) - M(t)\Big) & \text{Simpson rule,} \\ \dfrac{\sqrt{3}h}{12}\Big(M(t + c_2 h) - M(t + c_1 h)\Big) & \text{Gaussian quadrature rule,} \end{cases}$$

where $c_1 = \frac{1}{2} - \frac{\sqrt{3}}{6},\ c_2 = \frac{1}{2} + \frac{\sqrt{3}}{6}$.

With three exponentials, a standard method is

$$\begin{aligned} z_{n+1} &= \exp\left(\frac{1}{12}\alpha_2\right) \exp(\alpha_1) \exp\left(-\frac{1}{12}\alpha_2\right) z_n \\ &= \exp\left(M^{(1)}\right) \exp\left(M^{(0)}\right) \exp\left(-M^{(1)}\right) z_n. \end{aligned} \tag{22}$$

In general, the 2-exponential method provides slightly more accurate results. However, for the problem of interest (3)–(4), it is obvious from (12) that $\alpha_1 = hM(h/2)$ has the structure of a companion matrix (8), while the matrices associated with $\alpha_i,\ i > 1$ have only one non-zero row, i.e., they have the form

$$\alpha_2 = h^2 \begin{pmatrix} 0_{(N-1)\times(N+1)} \\ F_{1\times(N+1)} \\ 0_{1\times(N+1)} \end{pmatrix},$$

where $0_{m\times n}$ denotes a zero matrix of dimension $m \times n$ and $F_{(1\times N+1)} = (F_1, \ldots, F_{N+1})$ denotes a row vector. Notice that for this particular problem, $\exp\left(M^{(1)}\right)$ and $\exp\left(-M^{(1)}\right)$ can be written in a very simple closed form. Thus, the three-exponential method (22) is, in general, faster to compute than its counterpart with two exponentials (21) and can be more efficient.

Example 1. Let us consider the following fourth-order non-homogeneous linear equation

$$x^{(4)} + f_2(t)x'' + f_0(t)x = g(t),$$

with

$$f_0(t) = 100\left(1 + \frac{1}{4}\cos(t)\right), \qquad f_2(t) = 50\left(1 + \frac{1}{4}\sin(t)\right), \qquad g = \mathrm{erf}(t).$$

Notice that

$$(M^{(1)})^2 = 0 \Rightarrow \exp\left(M^{(1)}\right) = I + M^{(1)},$$

and then the 3-exponential method, for this problem, is given by

$$\left(I + M^{(1)}\right) \exp \begin{pmatrix} 0 & h & 0 & 0 & 0 \\ 0 & 0 & h & 0 & 0 \\ 0 & 0 & 0 & h & 0 \\ -f_0^{(0)} & 0 & -f_2^{(0)} & 0 & g^{(0)} \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix} \left(I - M^{(1)}\right) \tag{23}$$

being an algorithm with less complexity than the 2-exponential method because it only requires exponentiation of one companion matrix.

In Fig. 1, we show the 2-norm error of the fundamental matrix solution at $T = 10$ versus the number of time dependent function evaluations of the extended matrix $M(t)$. We compare with the explicit standard 4-stage fourth-order RK method (RK4) where $M(t)$ is evaluated at the same nodes as the Simpson rule, and the implicit 2-stage fourth-order RK method where $M(t)$ is evaluated at Gaussian nodes (GaussL4). The momentum integrals for the exponential methods have also been computed using Gaussian quadrature.

This problem has oscillatory solutions and the exponential methods are much more accurate at the same number of time-dependent function evaluations. Naturally, one also has to take into account the number of operations: In addition to the evaluations of the time-dependent functions, the explicit RK method requires a small number of products (and to store 4 vectors), the implicit RK method has to invert a matrix of twice the dimension of $M(t)$, and it is considerably more costly than the explicit method. The fourth-order exponential integrators Magnus integrator (Mag4, see (16)), the 2-exponential CF method using two α_1 elements from (21) (CF42) and the 3-exponential CF method using only one α_1-term (22) (CF41) require to approximate the exponentials up to a given order of accuracy [10]. The method CF42 is the most accurate but CF41, which is only slightly less accurate, is cheaper to compute. □

In general, sixth-order CF methods use compositions with $m \geq 5$ exponentials [20,19]. Each of the exponents involves α_1 which is the element responsible for the computational effort of the exponentiation. In addition, at least in one of the exponentials, α_1 is multiplied by a negative coefficient. These results motivated us to extend the analysis to order six.

We analyze new composition methods which allows us to obtain sixth-order methods with positive coefficients while being cheaper to compute than the existing CF methods.

3. New hybrid composition methods

In the present Lie algebra from the system (3)–(4), not only the matrix associated with α_2 has a particularly simple structure. We observe that the following elements are very similar:

$$\alpha_3 = h^3 \begin{pmatrix} 0_{(N-1)\times(N+1)} \\ G_{1\times(N+1)} \\ 0_{1\times(N+1)} \end{pmatrix}, \qquad [\alpha_j, \alpha_k] = h^{j+k} \begin{pmatrix} 0_{(N-2)\times(N+1)} \\ K_{2\times(N+1)} \\ 0_{1\times(N+1)} \end{pmatrix},$$

i.e., they only contain one and two non-empty rows and their norms are proportional to $\mathcal{O}(h^3)$ and $\mathcal{O}(h^{j+k})$, respectively.

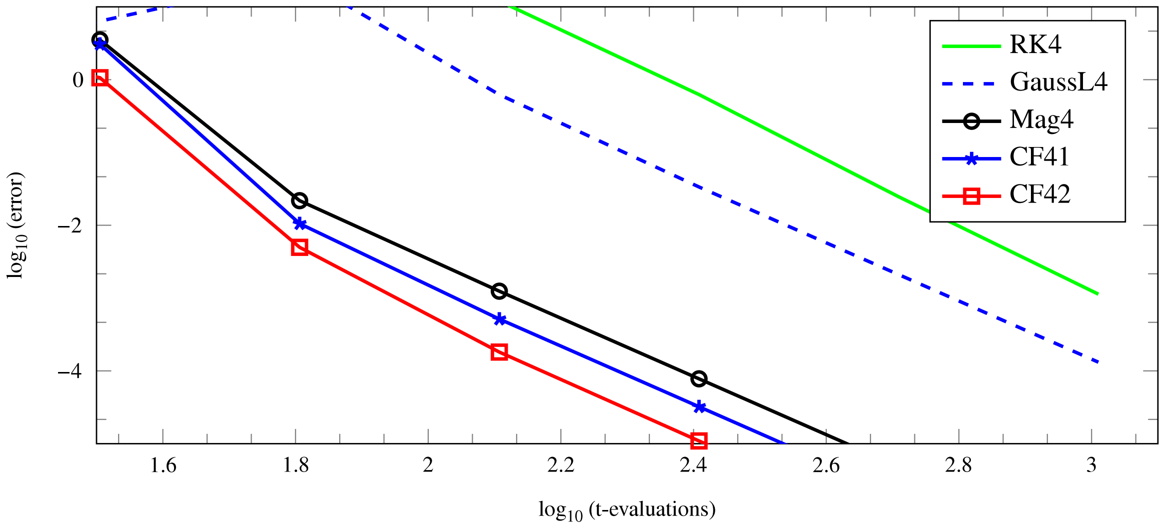

Fig. 1. Error in norm of the fundamental matrix solution, $\|\Phi(T,0)-\Phi_{ap}(T,0)\|$, where Φ_{ap} denotes the numerical solution for a given method, versus the number of time-dependent function evaluations for the problem in Example 1.

The goal is to build composition methods with as few exponentials involving the element α_1 as possible at a given order while leaving the remaining exponentials with cheaply computable matrices.

In our analysis, we only consider time-symmetric methods, i.e., maps $S(h)$ such that $S^{-1}(h) = S(-h)$. This approach simplifies the construction of methods and the methods share this property with the exact solution. In general, the most efficient composition methods in the literature have this symmetry. In order to consider a symmetric composition we proceed as follows, given $C_1(h), C_2(h)$ odd and even functions of h, i.e., $C_1(-h) = -C_1(h),\ C_2(-h) = C_2(h)$, then if $S_k(h)$ is a symmetric composition the following composition

$$S_{k+1}(h) = \exp\Big(C_1(h) + C_2(h)\Big)\ S_k(h)\ \exp\Big(C_1(h) - C_2(h)\Big)$$

is also time-symmetric.

We have studied the following sixth-order schemes and group them according to the number of appearances of α_1.

One-α_1-exponential method. We start from the fourth-order method (22) and suppose we use a sixth-order quadrature rule. Then, α_3 can be added to the symmetric composition and it can be used for optimization purposes, i.e.,

$$\Phi_3^{[4]} = \exp\left(\frac{1}{12}\alpha_2 + z_2\alpha_3\right)\ \exp\left(\alpha_1 + z_1\alpha_3\right)\ \exp\left(-\frac{1}{12}\alpha_2 + z_2\alpha_3\right). \tag{24}$$

This scheme has two parameters, z_1, z_2. Using the Baker–Campbell–Hausdorff (BCH) formula and equating to (14), we find that, by consistency, $z_1 + 2z_2 = 1/12$, and this leaves us with a free parameter which can be used to reduce the error. Since z_1, z_2 multiply α_3, they will appear linearly on the leading error terms at order 5 in the commutators [23] and [113]. If we consider that the commutator [113] (which contains two operators α_1) is more relevant for the error of the method, we can use the free parameter to cancel this term.

On the other hand, since a commutator contains only two non-empty rows, we could add to the first and last exponential a linear combination of the commutators [12], [13] and [23]. Since [12], [23] are odd operators in h we will include them distributed symmetrically. The operator [13] is even in h and will thus be distributed skew-symmetrically. In this way, we obtain the following symmetric composition scheme which contains six free parameters to solve the order conditions

$$\begin{aligned}\Phi_1^{[6]} &= \exp\Big(z_2\alpha_2 + z_3\alpha_3 + [\alpha_1 + z_4\alpha_2, z_5\alpha_1 + z_6\alpha_3]\Big)\ \exp\Big(\alpha_1 + z_1\alpha_3\Big)\\ &\quad\times \exp\Big(-z_2\alpha_2 + z_3\alpha_3 + [-\alpha_1 + z_4\alpha_2, z_5\alpha_1 + z_6\alpha_3]\Big).\end{aligned} \tag{25}$$

Apparently, there is the same number of order conditions as parameters, however, we found that there is a free parameter which we can use to reduce some of the error terms at leading order. In a similar way to the optimization process mentioned for the fourth-order method, we choose this parameter to cancel the coefficient which multiplies [11113]. The solution obtained is:

$$z_1 = \frac{1}{28},\qquad z_2 = \frac{1}{10},\qquad z_3 = \frac{1}{42},\qquad z_4 = -\frac{3}{4},\qquad z_5 = \frac{1}{90},\qquad z_6 = \frac{1}{840}.$$

The next question is: can we obtain an eight-order method adding new terms of similar complexity to the previous scheme?

If we use an eight-order quadrature rule, α_4 has to be included in the scheme and the truncated Magnus expansion becomes

$$\Omega^{[8]} = \Omega^{[6]} - \frac{1}{80}[14] + \mathcal{L}\Big([34], [124], [223], [313], [412], [1114], [1123], [1312], [2113], [2212], [11113], [11212], [21112], [111112]\Big),$$

where $\Omega^{[6]}$ is given in (14) and $\mathcal{L}$ denotes a linear combination of the elements which are of order $\mathcal{O}(h^7)$. The composition we can build is:

$$\begin{aligned}\tilde{\Phi}_1^{[6]} &= \exp\left(z_2\alpha_2 + z_3\alpha_3 + [\alpha_1 + z_4\alpha_2, z_5\alpha_1 + z_6\alpha_3] + a_1\alpha_4 + a_2[14] + a_3[24] + a_4[34]\right)\\ &\quad \times \exp\left(\alpha_1 + z_1\alpha_3\right) \exp\left(-z_2\alpha_2 + z_3\alpha_3 + [-\alpha_1 + z_4\alpha_2, z_5\alpha_1 + z_6\alpha_3] - a_1\alpha_4 + a_2[14] - a_3[24] + a_4[34]\right). \qquad (26)\end{aligned}$$

We have four new parameters, a_1, a_2, a_3, a_4 multiplying terms with α_4 which must satisfy $a_1 + 2z_5a_3 = 1/80$ in order to match the condition for [14] which is necessary for the method to be of order six. The three remaining parameters are used to reduce the error from the terms [34], [124], [412], [1114]. To reach order eight, 11 parameters are needed, but only 10 independent terms are available including all combinations with double commutators. This has not been explored in detail.

If an eight-order quadrature rule is used, the following relations must be used [21]

$$\begin{aligned}\alpha_1 &= \frac{9}{4}A^{(0)} - 15A^{(2)}, \qquad \alpha_2 = 15(5A^{(1)} - 28A^{(3)}),\\ \alpha_3 &= -15A^{(0)} + 180A^{(2)}, \qquad \alpha_4 = -140(3A^{(1)} - 20A^{(3)}).\end{aligned}$$

Two-α_1-exponential method. Next, we explore the following scheme with seven parameters to solve six order conditions leaving a free parameter:

$$\begin{aligned}\Phi_2^{[6]} &= \exp\Big(z_3\alpha_2 + z_4\alpha_3 + [\alpha_1 + z_5\alpha_2, z_6\alpha_1 + z_7\alpha_3]\Big)\\ &\quad \times \exp\Big(\alpha_1/2 + z_1\alpha_2 + z_2\alpha_3\Big)\exp\Big(\alpha_1/2 - z_1\alpha_2 + z_2\alpha_3\Big)\\ &\quad \times \exp\Big(-z_3\alpha_2 + z_4\alpha_3 + [-\alpha_1 + z_5\alpha_2, z_6\alpha_1 + z_7\alpha_3]\Big). \qquad (27)\end{aligned}$$

There is a free parameter that, as in the previous case, is used to cancel the coefficient at order seven which multiplies [11113]. The solution obtained is:

$$z_1 = \frac{1}{10}, \quad z_2 = \frac{89}{4536}, \quad z_3 = \frac{3}{80}, \quad z_4 = \frac{25}{1134}, \quad z_5 = -\frac{51}{976}, \quad z_6 = \frac{61}{1530}, \quad z_7 = \frac{61}{68040}.$$

Three-α_1-exponential method. A third α_1-exponential opens the possibility of a negative coefficient multiplying α_1. We propose the following new commutator-free method:

$$\begin{aligned}\Phi_3^{[6]} &= \exp\Big(z_6\alpha_2 + z_7\alpha_3\Big)\exp\Big(z_3\alpha_1 + z_4\alpha_2 + z_5\alpha_3\Big)\exp\Big(z_1\alpha_1 + z_2\alpha_3\Big)\exp\Big(z_3\alpha_1 - z_4\alpha_2 + z_5\alpha_3\Big)\\ &\quad \times \exp\Big(-z_6\alpha_2 + z_7\alpha_3\Big).\end{aligned}$$

Two real solutions are obtained, one with $z_1 < 0$ and the other with $z_3 < 0$. The solution with $z_1 < 0$ is:

$$\begin{aligned}z_1 &= -0.13408143773095485 5148833, & z_2 &= -0.01266912945062494 9118909,\\ z_3 &= 0.56704071886547742 7574417, & z_4 &= 0.15679795546721757 2935920,\\ z_5 &= 0.03255502814109521 1662211, & z_6 &= 0.01544620325088392 9563910,\\ z_7 &= 0.01544620325088392 9563910. & & \qquad (28)\end{aligned}$$

At least three exponentials including α_1 are necessary in order to solve the order conditions associated with the elements of the algebra α_1, [12] and [1112] without using commutators. This was not possible with the one- and two-exponential methods.

4. Numerical examples

In this section we analyze the performance of the following methods:

- The 7-stage sixth-order explicit RK method (RK6) with coefficients given in [22, p. 177] which requires 5 new evaluations of the matrix $M(t)$ per step.

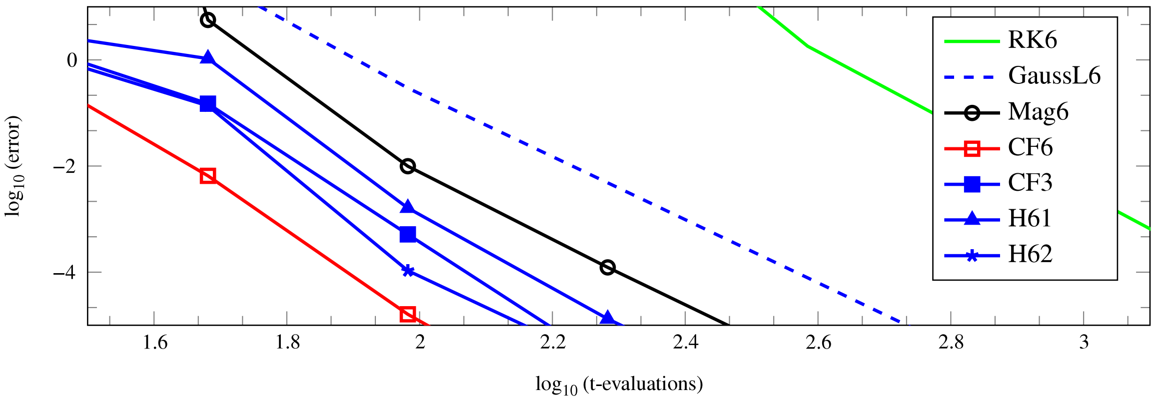

Fig. 2. Same as Fig. 1 for the sixth-order methods. Here, H61 and H62 denote the new sixth-order hybrid methods from (25) and (27), respectively. CF3 is the novel commutator-free method from (28).

- The 3-stage sixth-order implicit Runge–Kutta–Gauss–Legendre method (GaussL6) which requires 3 new evaluations of the matrix $M(t)$ per step.
- The one-exponential sixth-order Magnus integrator (Mag6) using the Gauss quadrature rule [21].
- The 6-exponential sixth-order CF method (CF6) from [20].
- The 1-α_1-exponential sixth-order Hybrid method (H61) given in (25).
- The 2-α_1-exponential sixth-order Hybrid method (H62) given in (27).
- The 3-α_1-exponential sixth-order CF method (CF3) given in (28).

As a first test, we repeat the numerical integration in Example 1. The results are shown in Fig. 2. We observe that all exponential methods are clearly superior to the explicit and implicit RK methods. The CF6 method is the most accurate, but the hybrid methods are cheaper to compute. Obviously, the relative computational cost between them will depend on each particular problem, but the cost of H61 and H62 could be as low as half the cost of the method CF6. In addition, these two methods have positive coefficients multiplying α_1 and this could be of interest for some problems.

As we have mentioned, the solution of this equation is oscillatory. When the coefficients of the equation oscillate with a frequency close to the frequency of the system, a parametric resonance can appear. This is the case, e.g. for the well known Mathieu equation $x'' + (w^2 + \epsilon \cos(t))x = 0$ for w close to the resonant values $w = 0, 1, 2, \ldots$.

Example 2. Let us now consider the same fourth-order non-homogeneous linear equation

$$x^{(4)} + f_2(t)x'' + f_0(t)x = g(t), \tag{29}$$

but with

$$f_0(t) = 4\,(1 + e\cos(\omega t))\,, \qquad f_2(t) = 5\,(1 + e\cos(\omega t))\,, \qquad g(t) = \exp(\cos(\omega t)), \tag{30}$$

that has parametric resonances for values of ω around $\omega = 1$ and $\omega = 2$. We take $\omega = 2$ and integrate the fundamental matrix solution until the final times $t_f = 10$ and $t_f = 100$. The solution is very sensitive to the parameter e and we repeat the computations for $e = 1/10$ and $e = 1/2$. The results are illustrated in Fig. 3.

We observe that the error grows with the final time as well as with the choice of the parameter e, and the new hybrid methods show an excellent performance since they are among the most accurate as well as the cheapest to compute (apart from the explicit RK method which show a very poor performance). Similar results are obtained for the numerical integration of the Mathieu equation for values of the parameters in the instability region. If one is interested in finding the stability regions for a given set of parameters of the equation, usually this is done with the numerical integration of the equations repeatedly for many different choices of the parameters and efficient methods need to be fast and accurate. The methods presented in this work are thus of great interest for such problems.

5. Conclusions

We have studied the numerical integration of high-order linear non-homogeneous differential equations written as first order homogeneous linear equations (which show a particular algebraic structure in terms of the companion matrix) using exponential methods. We have shown how to build new methods which are hybrid compositions between Magnus and commutator-free methods. The new methods can reach similar accuracy as standard exponential methods, but with a reduced complexity. Additional parameters can be included into the scheme for optimization purposes. We have shown how to obtain the order conditions to build sixth-order methods and several methods are obtained. The performance of the new methods has been tested on several numerical problems.

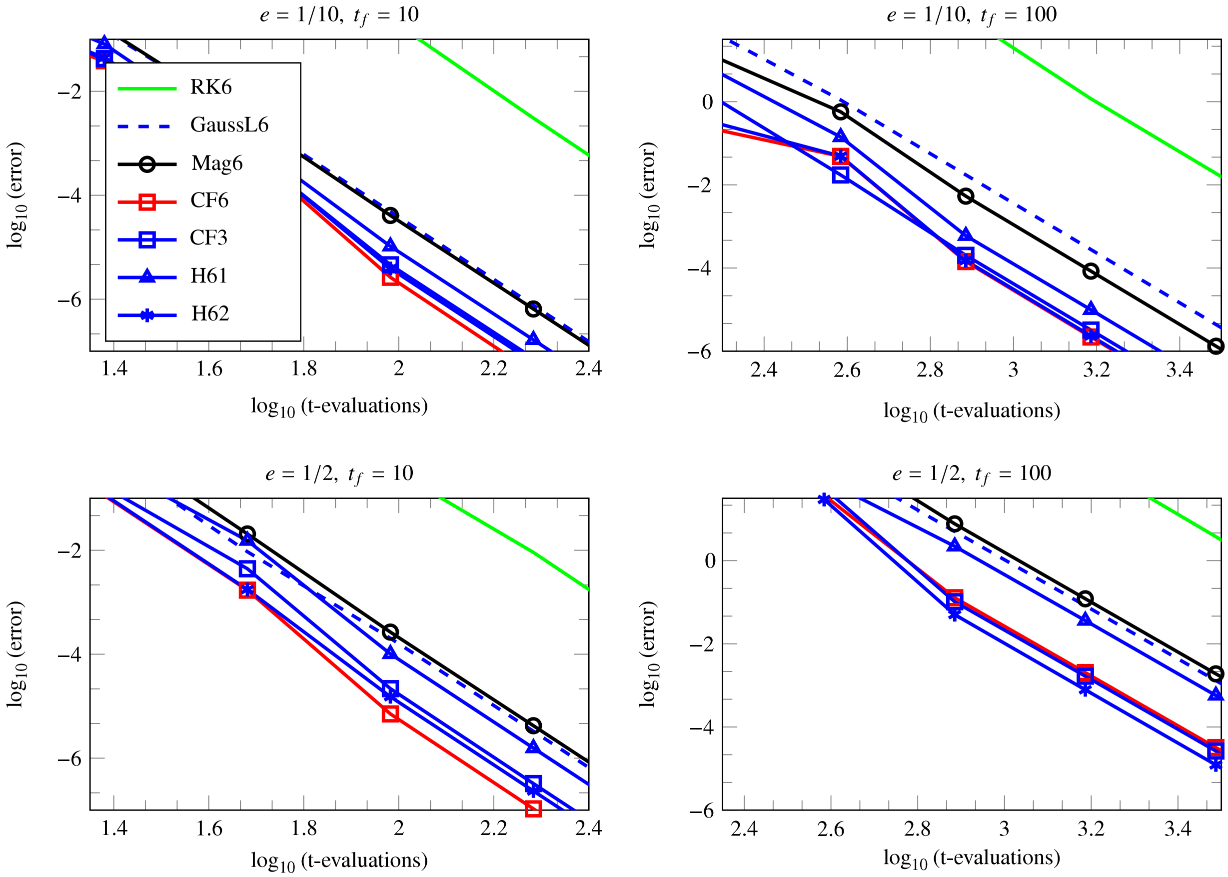

Fig. 3. The two-norm error in the fundamental matrix solution for the sixth-order methods applied to the problem (29)–(30) computed at the final times $t_f = 10$ (left figures) and $t_f = 100$ (right figures) for the choices $e = 1/10$ (top figures) and $e = 1/2$ (bottom figures).

As a further application of the results of the present work, we remark that homogeneous non-autonomous linear differential equations describe the evolution of many dynamical systems in classical and quantum mechanics (see [23,18] and references therein) as well as in biology [24] or engineering [25,5,26]. It is straightforward to extend the here presented analysis to the class of homogeneous non-autonomous equations

$$x' = M(t)x,$$

where $M(t) = A + B(t)$ such that the evaluation of $\exp(M(t))$ is computationally demanding but $\exp(B(t))$ can be trivially computed. One can either use the methods derived in this work or build new exponential methods for particular problems. This requires the analysis of the Lie algebra associated with the Matrix $M(t)$. If the analysis carried out on a family of problems indicates that some other elements of the Lie algebra can also be efficiently computed, these elements can also be considered in the scheme in a similar way as shown in this work.

On the other hand, it is well known that the computational cost of a given method strongly depends on the problem to be solved. As we have mentioned, an s-stage fully implicit RK method requires to compute the inverse of a large matrix which is about s^3 times the cost of the inverse of a $N \times N$ matrix, and the inverse of a matrix can be carried out at the cost of 4/3 the product of two matrices [10]. Then, for a sixth-order method with $s = 3$ the total cost is 3 evaluations of the $M(t)$ and $s^3\frac{4}{3} = 36$ products. On the other hand, one can approximate e^{hM} using a Padé approximation up to accuracy of order $\mathcal{O}(h^{10})$ with 3 products and one inverse, i.e. at the cost of only $(4 + 1/3)$ products (or using the Paterson–Stockmeyer scheme to compute the Taylor expansion up to $\mathcal{O}(h^9)$ with only 4 products [10]). Obviously, the cost of each matrix–matrix multiplication will depend on the sparsity of the matrix, and this is different for Magnus and commutator-free methods as well as for the new methods. This analysis has to be undertaken in order to determine the most efficient method for a given problem class or, for the design of new methods along the procedures of this work (possibly using extra parameters for optimization).

Acknowledgments

PB, SB and FC acknowledge the Ministerio de Economía y Competitividad (Spain) for financial support through the coordinated project MTM2013-46553-C3-3-P. PB also acknowledges the support through the FPU fellowship AP2009-1892. FC has been additionally supported by NPRP GRANT #5-674-1-114 from the Qatar National Research Fund. EP has been partially supported by the Universitat Politècnica de València under project 2087.

References

[1] T.M. Apostol, Explicit formulas for solutions of the second order matrix differential equation $Y = AY$, Amer. Math. Monthly 82 (1975) 159–162.
[2] I.I. Kolodner, On $\exp(tA)$ with A satisfying a polynomial, J. Math. Anal. Appl. 52 (1975) 514–524.
[3] G.I. Kalogeropoulos, A.D. Karageorgos, A.A. Pantelous, On the solution of higher order linear homogeneous complex $\sigma-\alpha$ descriptor matrix differential systems of Apostol–Kolodner type, J. Franklin Inst. 351 (3) (2014) 1756–1777.
[4] R.B. Taher, M. Rachidi, Linear matrix differential equations of higher-order and applications, Electron. J. Differential Equations 95 (2008) 1–12.
[5] U.M. Ascher, R.M. Mattheij, R.D. Russell, Numerical Solutions of Boundary Value Problems for Ordinary Differential Equations, SIAM, 1995.
[6] T.Y. Na, Computational Methods in Engineering Boundary Value Problems, in: Math. in Science and Engineering, vol. 145, Academic Press, New York, 1979.
[7] M.A. Noor, K.I. Noor, S.T. Mohyud-Din, Variational iteration method for solving sixth-order boundary value problems, Commun. Nonlinear Sci. Numer. Simul. 14 (2009) 2571–2580.
[8] Y. Wang, Y.B. Zhao, G.W. Wei, A note on the numerical solution of high-order differential equations, J. Comput. Appl. Math. 159 (2003) 387–398.
[9] S. Blanes, F. Diele, C. Marangi, S. Ragni, Splitting and composition methods for explicit time dependence in separable dynamical systems, J. Comput. Appl. Math. 235 (2010) 646–659.
[10] A.H. Al-Mohy, N.J. Higham, Computing the action of the matrix exponential, with an application to exponential integrators, SIAM J. Sci. Comput. 33 (2011) 488–511.
[11] N.J. Higham, A.H. Al-Mohy, Computing matrix functions, Acta Numer. 51 (2010) 159–208.
[12] C.B. Moler, C.F. Van Loan, Nineteen dubious ways to compute the exponential of a matrix, twenty-five years later, SIAM Rev. 45 (2003) 3–49.
[13] J. Niesen, W.M. Wright, Algorithm 919: a Krylov subspace algorithm for evaluating the φ-functions appearing in exponential integrators, ACM Trans. Math. Software 38 (2012) Article 22.
[14] R.B. Sidje, Expokit: a software package for computing matrix exponentials, ACM Trans. Math. Software 24 (1998) 130–156.
[15] W. Magnus, On the exponential solution of differential equations for a linear operator, Comm. Pure Appl. Math. 7 (1954) 649–673.
[16] A. Iserles, H.Z. Munthe-Kaas, S.P. Nørsett, A. Zanna, Lie group methods, Acta Numer. 9 (2000) 215–365.
[17] A. Iserles, S.P. Nørsett, On the solution of linear differential equations in Lie groups, Philos. Trans. R. Soc. Lond. Ser. A 357 (1999) 983–1019.
[18] S. Blanes, F. Casas, J.A. Oteo, J. Ros, The Magnus expansion and some of its applications, Phys. Rep. 470 (2009) 151–238.
[19] S. Blanes, P.C. Moan, Fourth- and sixth-order commutator-free Magnus integrators for linear and non-linear dynamical systems, Appl. Numer. Math. 56 (2006) 1519–1537.
[20] A. Alvermann, H. Fehske, High-order commutator-free exponential time-propagation of driven quantum systems, J. Comput. Phys. 230 (2011) 5930–5956.
[21] S. Blanes, F. Casas, J. Ros, Improved high order integrators based on the Magnus expansion, BIT 40 (2000) 434–450.
[22] J.C. Butcher, Numerical Methods for Ordinary Differential Equations, Wiley, Chichester, England, 2003.
[23] P. Bader, Geometric integrators for Schrödinger equations (Ph.D. thesis), Univ. Pol. Valencia, 2014.
[24] S. Ruan, G.S.K. Wolkowicz, J. Wu (Eds.), Dynamical Systems and their Applications in Biology, in: Fields Institute Communications, American Mathematical Society, USA, 2003.
[25] B.D.O. Anderson, J.B. Moore, Optimal Control: Linear Quadratic Methods, Dover, New York, 1990.
[26] S. Blanes, E. Ponsoda, Magnus integrators for solving linear-quadratic differential games, J. Comput. Appl. Math. 236 (2012) 3394–3408.