

Explicit Magnus expansions for nonlinear equations

Fernando Casas¹ and Arieh Iserles²

¹ Departament de Matemàtiques, Universitat Jaume I, 12071-Castellón, Spain

² Department of Applied Mathematics and Theoretical Physics, University of Cambridge, Wilberforce Rd., Cambridge CB3 0WA, UK

E-mail: Fernando.Casas@mat.uji.es and ai@damp.cam.ac.uk

Received 19 August 2005, in final form 6 December 2005

Published 24 April 2006

Online at stacks.iop.org/JPhysA/39/5445

Abstract

In this paper we develop and analyse new explicit Magnus expansions for the nonlinear equation $Y' = A(t, Y)Y$ defined in a matrix Lie group. In particular, integration methods up to order four are presented in terms of integrals which can be either evaluated exactly or replaced by conveniently adapted quadrature rules. The structure of the algorithm allows us to change the step size and even the order during the integration process, thus improving its efficiency. Several examples are considered, including isospectral flows and highly oscillatory nonlinear differential equations.

PACS numbers: 02.60.-x, 02.60.Lj, 02.30.Hq

Mathematics Subject Classification: 65L05, 41A55, 22E60

1. Introduction

Nowadays the so-called Magnus expansion constitutes a widespread tool to construct approximate solutions of non-autonomous systems of linear ordinary differential equations. As is well known, the basic idea is to represent the solution of

$$Y' = A(t)Y, \quad Y(0) = Y_0, \quad (1)$$

where A is an $n \times n$ matrix, in the form $Y = \exp(\Omega(t))Y_0$ and express Ω as an infinite series $\Omega(t) = \sum_{k=1}^{\infty} \Omega_k(t)$, whose terms are linear combinations of integrals and nested commutators involving the matrix A at different times [22]. In particular, the first terms read explicitly

$$\Omega_1(t) = \int_0^t A(t_1) dt_1, \quad \Omega_2(t) = \frac{1}{2} \int_0^t dt_1 \int_0^{t_1} dt_2 [A_1, A_2],$$

where $A_i \equiv A(t_i)$ and $[X, Y] \equiv XY - YX$ is the commutator of X and Y . Explicit formulae for Ω_k of all orders have been given in [19] by using graph theory, whereas in [21] a recursive procedure for the generation of Ω_k was proposed. Different approximations to the solution of (1) are obtained when the series of Ω is truncated. This procedure has the very attractive

property of ensuring preservation of important qualitative properties of the exact solution at any order of truncation.

Since the 1960s, the Magnus expansion has been successfully applied as an analytic tool in numerous areas of physics and chemistry, from nuclear, atomic and molecular physics to nuclear magnetic resonance and quantum electrodynamics (see [1] for a list of references). Also a convergence proof for the series defining Ω has been obtained [1].

In recent years, Iserles and Nørsett [19] used rooted trees to analyse the expansion terms, leading to a recursive procedure to generate Ω and constructing practical algorithms for the numerical integration of equation (1). The resulting schemes are prototypical examples of *geometric integrators*: numerical methods for discretizing differential equations which preserve their known qualitative features, such as invariant quantities and geometric structure [14]. By sharing such properties with the exact solution, these methods provide numerical approximations which are more accurate and more stable for important classes of differential equations, such as those evolving on Lie groups. In addition, several integrators based on the Magnus expansion have proved to be highly competitive with other, more conventional numerical schemes with respect to accuracy and computational effort [2, 3].

In this respect, there are two important factors involved in the process of rendering Magnus expansion as a class of numerical integrators. Firstly, the structure of the Magnus series is such that the number of matrix evaluations required to compute all the multivariate integrals in the expansion to a given order is the same as the cost of the single quadrature formula for Ω_1 [19]. Secondly, an optimization procedure can be designed to reduce a great deal the number of commutators required by the scheme [3].

It is perhaps for these reasons that, although these algorithms have been primarily designed for linear problems, where the matrix function A depends on time only, several attempts have been made to generalize the formalism when $A = A(t, Y)$. In that case, though, multivariate integrals depend also on the value of the (unknown) variable Y at quadrature points. This leads to implicit methods and nonlinear algebraic equations in every step of the integration [30] which in general cannot compete in efficiency with other classes of geometric integrators such as splitting and composition methods. An interesting alternative has been proposed by Blanes and Moan [4]: they use a conveniently modified version of the Magnus expansion to construct a new class of splitting methods for non-autonomous Hamiltonian dynamical systems.

In this paper we try to overcome some of the aforementioned difficulties and develop new explicit Magnus expansions for the nonlinear equation

$$Y' = A(t, Y)Y, \quad Y(0) = Y_0 \in \mathcal{G}, \quad (2)$$

where \mathcal{G} is a matrix Lie group, $A : \mathbb{R}_+ \times \mathcal{G} \rightarrow \mathfrak{g}$ and \mathfrak{g} denotes the corresponding Lie algebra (the tangent space at the identity of \mathcal{G}). Equation (2) appears in relevant physical fields such as rigid mechanics and in the calculation of Lyapunov exponents ($\mathcal{G} \equiv \text{SO}(n)$), Hamiltonian dynamics ($\mathcal{G} \equiv \text{Sp}(n)$) and quantum mechanics ($\mathcal{G} \equiv \text{SU}(n)$). In fact, it can be shown that every differential equation evolving on a matrix Lie group \mathcal{G} can be written in the form (2). Moreover, the analysis of generic differential equations defined in homogeneous spaces can be reduced to the Lie-group equation (2) [28]. It is therefore of the greatest interest to design numerical geometric integration schemes for the system which are computationally as efficient as possible.

One common technique to solve (2) whilst preserving its Lie-group structure is to lift $Y(t)$ from \mathcal{G} to the underlying Lie algebra \mathfrak{g} (usually with the exponential map), then formulate and solve there an associated differential equation and finally map the solution back to \mathcal{G} . In this way the discretization procedure works in a linear space rather than in the Lie group. In particular, the idea of Munthe-Kaas is to approximate the solution of the associated differential

equation in the Lie algebra \mathfrak{g} by means of a classical Runge–Kutta method, thus obtaining the so-called Runge–Kutta–Munthe-Kaas (RKMK) class of schemes [18, 25, 26].

In this paper a general procedure for devising Magnus expansions for the nonlinear equation (2) is introduced. It is based on applying Picard’s iteration on the associated differential equation in the Lie algebra and retaining in each iteration the terms necessary to increase the order while maintaining the explicit character of the expansion. The resulting methods are thus explicit by design and are expressed in terms of integrals. They provide continuous approximations to the exact solution, i.e., they are valid for all values of t , not just at discrete time steps. These approximations can be treated in different ways depending on how the integrals are computed. If they are replaced by standard quadrature rules, new numerical integrators arise which can be expressed as RKMK schemes, although they generally involve more computational effort per step. Nevertheless, their structure allows quite naturally to design algorithms with variable approximation order during the integration process.

For certain types of nonlinear problems, such as those whose solution is highly oscillatory, the integrals appearing in the formalism can be approximated by specially adapted quadrature, thus rendering new integration methods specifically oriented to this class of systems.

Since the Magnus expansion we elaborate here yields explicit continuous approximate solutions, it might constitute a useful device in perturbation theory of nonlinear differential equations depending on a certain parameter ε [29]: the integrals could be conveniently replaced by expressions containing the parameter and it would be possible to analyse the ε -dependence of the resulting approximations. Also the continuous solutions provided by the formalism could be employed (analogously to continuous variants of Runge–Kutta methods) in delay-differential equations [15].

The plan of the paper is as follows. In section 2 the explicit Magnus expansion for equation (2) is presented and analysed, in general and for the particular yet highly important case of isospectral flows. In section 3 we construct some numerical schemes based on the new expansion and illustrate their features with a numerical example, comparing them with the class of RKMK methods. In section 4 we show how the expansion can be implemented to integrate highly oscillatory nonlinear ODEs, by choosing the right quadrature rules on a modified version of the algorithm. Finally, section 5 contains some conclusions.

2. Magnus expansion

2.1. General case

As usual, the starting point in the formalism is to represent the solution of (2) in the form

$$Y(t) = e^{\Omega(t)} Y_0. \tag{3}$$

Then one obtains after trivial algebra the differential equation satisfied by Ω :

$$\Omega' = d \exp_{\Omega}^{-1}(A(t, e^{\Omega} Y_0)), \quad \Omega(0) = O. \tag{4}$$

Here

$$d \exp_{\Omega}^{-1}(C) = \sum_{k=0}^{\infty} \frac{B_k}{k!} \text{ad}_{\Omega}^k C,$$

$\{B_k\}_{m \in \mathbb{Z}_+}$ are the Bernoulli numbers and ad^m is a shorthand for an iterated commutator,

$$\text{ad}_{\Omega}^0 A = A, \quad \text{ad}_{\Omega}^{m+1} A = [\Omega, \text{ad}_{\Omega}^m A], \quad m \geq 0.$$

In the linear case, i.e. when A depends on time only, the Magnus series for Ω can be obtained by Picard's iteration,

$$\begin{aligned}\Omega^{[0]}(t) &\equiv O \\ \Omega^{[m+1]}(t) &= \int_0^t \mathbf{d} \exp_{\Omega^{[m]}(s)}^{-1} A(s) \, ds = \int_0^t \sum_{k=0}^{\infty} \frac{B_k}{k!} \text{ad}_{\Omega^{[m]}(s)}^k A(s) \, ds, \quad m \geq 0.\end{aligned}$$

The same formal procedure can also be applied to equation (4), giving instead

$$\begin{aligned}\Omega^{[m+1]}(t) &= \int_0^t \mathbf{d} \exp_{\Omega^{[m]}(s)}^{-1} A(s, e^{\Omega^{[m]}(s)} Y_0) \, ds \\ &= \int_0^t \sum_{k=0}^{\infty} \frac{B_k}{k!} \text{ad}_{\Omega^{[m]}(s)}^k A(s, e^{\Omega^{[m]}(s)} Y_0) \, ds, \quad m \geq 0.\end{aligned}$$

The next step in getting explicit approximations is to truncate appropriately the $\mathbf{d} \exp^{-1}$ operator in the above expansion. Roughly speaking, when the whole series for $\mathbf{d} \exp^{-1}$ is considered, the power series expansion of the iterate function $\Omega^{[k]}(t)$, $k \geq 1$, only reproduces the expansion of the solution $\Omega(t)$ up to certain order, say $\mathcal{O}(t^m)$. In consequence, the (infinite) power series of $\Omega^{[k]}(t)$ and $\Omega^{[k+1]}(t)$ differ in terms of order $\mathcal{O}(t^{m+1})$. The idea is then to discard in $\Omega^{[k]}(t)$ all terms of order greater than $\mathcal{O}(t^m)$. This of course requires careful analysis of each term in the expansion. For instance, $\Omega^{[0]} = O$ implies that $(\Omega^{[1]})' = A(t, Y_0)$ and therefore

$$\Omega^{[1]}(t) = \int_0^t A(s, Y_0) \, ds = \Omega(t) + \mathcal{O}(t^2).$$

Since

$$A(s, e^{\Omega^{[1]}(s)} Y_0) = A(0, Y_0) + \mathcal{O}(s)$$

it follows at once that

$$-\frac{1}{2} \int_0^t [\Omega^{[1]}(s), A(s, e^{\Omega^{[1]}(s)} Y_0)] \, ds = \mathcal{O}(t^3).$$

When this second term in $\Omega^{[2]}(t)$ is included and $\Omega^{[3]}$ is computed, it turns out that $\Omega^{[3]}$ reproduces correctly the expression of $\Omega^{[2]}$ up to $\mathcal{O}(t^2)$. Therefore we truncate $\mathbf{d} \exp^{-1}$ at the $k = 0$ term and take

$$\Omega^{[2]}(t) = \int_0^t A(s, e^{\Omega^{[1]}(s)} Y_0) \, ds.$$

With greater generality, we let

$$\begin{aligned}\Omega^{[1]}(t) &= \int_0^t A(s, Y_0) \, ds \\ \Omega^{[m]}(t) &= \sum_{k=0}^{m-2} \frac{B_k}{k!} \int_0^t \text{ad}_{\Omega^{[m-1]}(s)}^k A(s, e^{\Omega^{[m-1]}(s)} Y_0) \, ds, \quad m \geq 2\end{aligned}\tag{5}$$

and take the approximation $\Omega(t) \approx \Omega^{[m]}(t)$. This results in an explicit approximate solution that involves a linear combination of multiple integrals of nested commutators, so that $\Omega^{[m]}(t) \in \mathfrak{g}$ for all $m \geq 1$. In addition, it is a trivial exercise to show that $\Omega^{[m]}(t)$ reproduces exactly the sum of the first m terms in the Ω series of the usual Magnus expansion for the linear equation $Y' = A(t)Y$. It makes sense, then, to regard the scheme (5) as an explicit Magnus expansion for the nonlinear equation (2).

The actual order of approximation is provided by the following result (which as a matter of fact generalizes the cases $m = 1$ and $m = 2$ studied before):

Theorem 2.1. *Let $\Omega(t)$ be the exact solution of the initial value problem (4) and $\Omega^{[m]}(t)$ the iterate given by scheme (5). Then it is true that*

$$\Omega(t) - \Omega^{[m]}(t) = \mathcal{O}(t^{m+1}).$$

In other words, $\Omega^{[m]}(t)$, once inserted in (3), provides an explicit approximation $Y^{[m]}(t)$ for the solution of (2) that is correct up to order $\mathcal{O}(t^{m+1})$.

Sketch of the proof: To simplify matters, let us consider the autonomous case, i.e., $Y' = A(Y)Y$. The extension to the general situation is straightforward.

In this case a long but simple calculation shows that the exact solution of (4) can be written as the infinite series

$$\begin{aligned} \Omega(t) &= \sum_{l=1}^{\infty} t^l \omega_l \\ \text{with } \omega_1 &= A(Y_0), \omega_2 = \frac{1}{2}G_1 \text{ and, for } l \geq 3, \\ l\omega_l &= G_{l-1} + \sum_{j=1}^{l-1} \frac{B_j}{j!} \sum_{\substack{k_1+\dots+k_j=l-1 \\ k_1 \geq 1, \dots, k_j \geq 1}} \text{ad}_{\omega_{k_1}} \cdots \text{ad}_{\omega_{k_j}} A(Y_0) \\ &+ \sum_{j=1}^{l-2} \left(\sum_{s=0}^j \frac{B_s}{s!} \sum_{\substack{k_1+\dots+k_s=j \\ k_1 \geq 1, \dots, k_s \geq 1}} \text{ad}_{\omega_{k_1}} \cdots \text{ad}_{\omega_{k_s}} \right) G_{l-1-j} \\ &+ \sum_{j=1}^{l-2} \frac{B_j}{j!} \sum_{\substack{k_1+\dots+k_j=l-2 \\ k_1 \geq 1, \dots, k_j \geq 1}} \text{ad}_{\omega_{k_1}} \cdots \text{ad}_{\omega_{k_j}} G_1. \end{aligned} \tag{6}$$

Here G_k is a function which depends on $Y_0, \omega_1, \dots, \omega_k$,

$$G_k = G_k(Y_0; \omega_1, \omega_2, \dots, \omega_k), \quad k \geq 1.$$

On the other hand, if we discard all terms of order exceeding $\mathcal{O}(t^m)$ in $\Omega^{[m]}(t)$ given by (5), then

$$\Omega^{[m]}(t) = \sum_{l=1}^m t^l \hat{\omega}_l,$$

where $\hat{\omega}_1 = A(Y_0)$ and $\hat{\omega}_l, 2 \leq l \leq m$, is given by the same expression (6) with the substitutions

$$\omega_k \mapsto \hat{\omega}_k, \quad G_k \mapsto \hat{G}_k,$$

but now $\hat{G}_k = \hat{G}_k(Y_0; \hat{\omega}_1, \hat{\omega}_2, \dots, \hat{\omega}_{m-1}), k = 1, \dots, m$.

Since $\hat{\omega}_1 = \omega_1$, then $\hat{G}_1 = G_1$ and, by induction,

$$\hat{\omega}_l = \omega_l, \quad \hat{G}_l = G_l \quad \text{for } l = 1, \dots, m - 1,$$

but $\hat{G}_m = \hat{G}_m(Y_0; \hat{\omega}_1, \hat{\omega}_2, \dots, \hat{\omega}_{m-1})$, whereas $G_m = G_m(Y_0; \omega_1, \omega_2, \dots, \omega_m)$, so that $\hat{G}_m \neq G_m$. In consequence

$$\Omega'(t) - (\Omega^{[m]}(t))' = t^m(G_m - \hat{G}_m) + \mathcal{O}(t^{m+1})$$

and thus $\Omega(t) - \Omega^{[m]}(t) = \mathcal{O}(t^{m+1})$. □

2.2. Isospectral flows

The Magnus expansion introduced before can be easily adapted to construct an exponential representation of the solution for the differential system

$$Y' = [A(t, Y), Y], \quad Y(0) = Y_0 \in \text{Sym}(n). \quad (7)$$

Here $\text{Sym}(n)$ stands for the set of $n \times n$ symmetric real matrices and the (sufficiently smooth) function A maps $\mathbb{R}_+ \times \text{Sym}(n)$ into $\mathfrak{so}(n)$, the Lie algebra of $n \times n$ real skew-symmetric matrices. It is well known that the solution itself remains in $\text{Sym}(n)$ for all $t \geq 0$. Furthermore, the eigenvalues of $Y(t)$ are independent of time, i.e., $Y(t)$ has the same eigenvalues as Y_0 . This remarkable qualitative feature of the system (7) is the reason why it is called an *isospectral flow*. Such flows have several interesting applications in physics and applied mathematics, from molecular dynamics to micromagnetics to linear algebra [9].

Since $Y(t)$ and $Y(0)$ share the same spectrum, there exists a matrix function $Q(t) \in \text{SO}(n)$ (the Lie group of all $n \times n$ real orthogonal matrices with unit determinant), such that $Y(t)Q(t) = Q(t)Y(0)$, or equivalently,

$$Y(t) = Q(t)Y_0Q^T(t). \quad (8)$$

Then, by inserting (8) into (7), it is clear that the time evolution of $Q(t)$ is described by

$$Q' = A(t, QY_0Q^T)Q, \quad Q(0) = I, \quad (9)$$

i.e., an equation of type (2). Yet there is another possibility: if we seek the orthogonal matrix solution of (9) as $Q(t) = \exp(\Omega(t))$ with Ω being skew symmetric,

$$Y(t) = e^{\Omega(t)}Y_0e^{-\Omega(t)}, \quad t \geq 0, \quad \Omega(t) \in \mathfrak{so}(n), \quad (10)$$

then the corresponding equation for Ω reads

$$\Omega' = d \exp_{\Omega}^{-1}(A(t, e^{\Omega}Y_0e^{-\Omega})), \quad \Omega(0) = O. \quad (11)$$

In a similar way as for equation (4), we apply Picard's iteration to (11) and truncate the $d \exp^{-1}$ series at $k = m - 2$. Now we can also truncate consistently the operator

$$\text{Ad}_{\Omega}Y_0 \equiv e^{\Omega}Y_0e^{-\Omega} = e^{\text{ad}_{\Omega}}Y_0$$

and the outcome still lies in $\mathfrak{so}(n)$. By doing so, we replace the computation of one matrix exponential by several commutators.

In the end, the scheme reads

$$\begin{aligned} \Omega^{[1]}(t) &= \int_0^t A(s, Y_0) ds \\ \Theta_{m-1}(t) &= \sum_{l=0}^{m-1} \frac{1}{l!} \text{ad}_{\Omega^{[l]}(t)}^l Y_0 \\ \Omega^{[m]}(t) &= \sum_{k=0}^{m-2} \frac{B_k}{k!} \int_0^t \text{ad}_{\Omega^{[m-1]}(s)}^k A(s, \Theta_{m-1}(s)) ds, \quad m \geq 2 \end{aligned} \quad (12)$$

and, as before, one has $\Omega(t) = \Omega^{[m]}(t) + \mathcal{O}(t^{m+1})$. Thus

$$\begin{aligned} \Theta_1(t) &= Y_0 + [\Omega^{[1]}(t), Y_0] \\ \Omega^{[2]}(t) &= \int_0^t A(s, \Theta_1(s)) ds \\ \Theta_2(t) &= Y_0 + [\Omega^{[2]}(t), Y_0] + \frac{1}{2}[\Omega^{[2]}(t), [\Omega^{[2]}(t), Y_0]] \\ \Omega^{[3]}(t) &= \int_0^t A(s, \Theta_2(s)) ds - \frac{1}{2} \int_0^t [\Omega^{[2]}(s), A(s, \Theta_2(s))] ds \end{aligned}$$

and so on. Observe that this procedure preserves the isospectrality of the flow since the approximation $\Omega^{[m]}(t)$ lies in $\mathfrak{so}(n)$ for all $m \geq 1$ and $t \geq 0$. It is also equally possible to develop a formalism based on rooted trees in this case, in a similar way as for the standard Magnus expansion.

An important subclass of systems is formed by the so-called *quasilinear* isospectral flows. We say that the system (7) is quasilinear if A is a linear function in the entries of Y , i.e., $A(t, \alpha_1 Y_1 + \alpha_2 Y_2) = \alpha_1 A(t, Y_1) + \alpha_2 A(t, Y_2)$. In the autonomous case, some relevant examples include the double bracket flow, the periodic Toda lattice, the Toeplitz annihilator defined by

$$A_{k,l}(Y) = \begin{cases} Y_{k+1,l} - Y_{k,l-1}, & 1 \leq k < l \leq n, \\ 0, & 1 \leq k = l \leq n, \\ Y_{k-1,l} - Y_{k,l+1}, & 1 \leq l < k \leq n. \end{cases} \tag{13}$$

[12] and certain classes of Lie–Poisson flows [6, 7]. The isospectral flow (7) with a matrix A given by (13) can be used to find a symmetric Toeplitz matrix with a prescribed set of real numbers as its eigenvalues. The corresponding flow generally converges to an asymptotic state, so that in this context it is very useful to have explicit approximations [30].

When the iterative scheme (12) is applied to a time-independent quasilinear flow one gets the expression

$$\Omega^{[m]}(t) = \sum_{l=1}^m t^l \omega_l,$$

where the coefficients ω_l are constructed recursively (as in the proof of theorem 2.1, but now the functions G_k are determined explicitly):

$$\omega_1 = A(Y_0)$$

$$2\omega_2 = A(\text{ad}_{\omega_1} Y_0)$$

$$\begin{aligned} l\omega_l = & \sum_{j=1}^{l-1} \frac{1}{j!} \sum_{\substack{k_1+\dots+k_j=l-1 \\ k_1 \geq 1, \dots, k_j \geq 1}} A(\text{ad}_{\omega_{k_1}} \dots \text{ad}_{\omega_{k_j}} Y_0) + \sum_{j=1}^{l-1} \frac{B_j}{j!} \sum_{\substack{k_1+\dots+k_j=l-1 \\ k_1 \geq 1, \dots, k_j \geq 1}} \text{ad}_{\omega_{k_1}} \dots \text{ad}_{\omega_{k_j}} A(Y_0) \\ & + \sum_{j=2}^{l-1} \left(\sum_{s=1}^{j-1} \frac{B_s}{s!} \sum_{\substack{k_1+\dots+k_s=j-1 \\ k_1 \geq 1, \dots, k_s \geq 1}} \text{ad}_{\omega_{k_1}} \dots \text{ad}_{\omega_{k_s}} \right) \\ & \left(\sum_{p=1}^{l-j} \frac{1}{p!} \sum_{\substack{k_1+\dots+k_p=l-j \\ k_1 \geq 1, \dots, k_p \geq 1}} A(\text{ad}_{\omega_{k_1}} \dots \text{ad}_{\omega_{k_p}} Y_0) \right) \quad l \geq 3. \end{aligned} \tag{14}$$

In this case it is even possible to obtain a domain of convergence of the procedure when $m \rightarrow \infty$ by applying the same techniques as in [10]. Specifically, let us consider a norm in $\mathfrak{so}(n)$ and a number $\mu > 0$ satisfying

$$\|[X, Y]\| \leq \mu \|X\| \|Y\|$$

for all X, Y in $\mathfrak{so}(n)$ and suppose that A is a matrix such that $\|A(Y)\| \leq K \|Y\|$ for a certain constant K . (A discussion of an important case when $\mu < 2$ can be found in [5].) Then the series

$$\sum_{l=1}^{\infty} t^l \|\omega_l\|$$

converges for $0 \leq t < t_c$, where $t_c = \frac{\xi}{\mu K \|Y_0\|}$ and

$$\xi = \int_0^{2\pi} \frac{e^{-x}}{2 + \frac{x}{2}(1 - \cot \frac{x}{2})} dx \simeq 0.688\,776\dots$$

Example: the double bracket flow. The double bracket equation

$$Y' = [[Y, N], Y], \quad Y(0) = Y_0 \in \text{Sym}(n) \quad (15)$$

was introduced by Brockett [8] and Chu and Driessel [12] to solve certain standard problems in applied mathematics, although similar equations also appear in the formulation of physical theories such as micromagnetics [24]. Here N is a constant matrix in $\text{Sym}(n)$. As mentioned before, it constitutes an example of a quasilinear isospectral flow with $A(Y) \equiv [Y, N]$. Then, clearly, $\|A(Y)\| \leq K \|Y\|$ with $K = \mu \|N\|$. With these substitutions, (14) reproduces exactly the expansion obtained in [16] with the convergence domain established in [10].

3. Numerical integrators based on the Magnus expansion

3.1. The new methods

To build a practical numerical integrator based on the nonlinear Magnus expansion (5) (or (12)), the integrals appearing in the formalism have to be conveniently evaluated. For most problems, though, only the first-order approximation $\Omega^{[1]}(t)$ can be exactly computed, and so the remaining integrals needed at higher orders must be replaced by affordable quadratures, depending on the particular problem. Also the existence of several commutators and matrix exponential evaluations at the intermediate stages requires a detailed treatment to reduce the computational complexity and render practical integration schemes.

To illustrate the different issues involved in the construction of numerical methods from (5) we consider here schemes of order 2 and 3, whereas a fourth-order method is presented in the appendix. In all cases, we choose quadrature rules with equispaced points along the interval $[t_n, t_n + h]$.

Order 2. This case corresponds to $m = 2$ in (5), so that

$$\Omega^{[1]}(t) = \int_0^t A(s, Y_0) ds \quad (16)$$

$$\Omega^{[2]}(t) = \int_0^t A(s, e^{\Omega^{[1]}(s)} Y_0) ds. \quad (17)$$

If A is such that the integral (16) can be exactly computed, all that is required to get a second-order integrator is to replace the integral (17) with a quadrature rule of order 2. For instance, if we discretize $\Omega^{[2]}$ with the trapezoidal rule, then

$$\Omega^{[2]}(h) = \frac{h}{2} (A(0, Y_0) + A(h, e^{\Omega^{[1]}(h)} Y_0)) + \mathcal{O}(h^3). \quad (18)$$

In fact, to build a second-order scheme it is not necessary to evaluate exactly the integral (16), but only a first-order approximation. If, for instance, we use Euler's method, $\Omega^{[1]}(h) = hA(0, Y_0) + \mathcal{O}(h^2)$ and this results in a new explicit second-order scheme

$$\begin{aligned} v_1 &\equiv \frac{h}{2} (A(0, Y_0) + A(h, e^{hA(0, Y_0)} Y_0)) = \Omega^{[2]}(h) + \mathcal{O}(h^3) \\ Y_1 &= e^{v_1} Y_0, \end{aligned} \quad (19)$$

which is precisely the two-stage Runge–Kutta–Munthe-Kaas (RKMK) method with Butcher tableau

$$\begin{array}{c|cc} 0 & & \\ 1 & 1 & \\ \hline & \frac{1}{2} & \frac{1}{2} \end{array}.$$

Needless to say, method (19) is different from (18) when matrix A depends explicitly on time, although both are second-order approximations to $\Omega(t)$.

If $\Omega^{[2]}$ is discretized with the midpoint rule instead,

$$\begin{aligned} v_2 &\equiv hA\left(\frac{h}{2}, e^{\frac{h}{2}A(0,Y_0)}Y_0\right) = \Omega^{[2]}(h) + \mathcal{O}(h^3) \\ Y_1 &= e^{v_2}Y_0, \end{aligned} \tag{20}$$

we retrieve exactly the RKMK Heun method [18, p 355].

We see that, at order 2, the well-known explicit RKMK schemes are reproduced by the Magnus expansion as soon as $\Omega^{[1]}$ is discretized with a first-order quadrature. New schemes arise, however, when $\Omega^{[1]}$ is evaluated exactly.

Order 3. In addition to equations (16) and (17) we have to work with

$$\Omega^{[3]}(t) = \int_0^t \left(A_2(s) - \frac{1}{2}[\Omega^{[2]}(s), A_2(s)] \right) ds, \tag{21}$$

where $A_2(s) \equiv A(s, e^{\Omega^{[2]}(s)}Y_0)$. If we use Simpson’s rule to approximate (21), then

$$\begin{aligned} \Omega^{[3]}(h) &= \frac{h}{6}(A(0, Y_0) + 4A_2(h/2) + A_2(h)) \\ &\quad - \frac{h}{3}[\Omega^{[2]}(h/2), A_2(h/2)] - \frac{h}{12}[\Omega^{[2]}(h), A_2(h)] + \mathcal{O}(h^4). \end{aligned}$$

Now $\Omega^{[1]}$ can be approximated with Euler and $\Omega^{[2]}(h)$ with the midpoint rule, eq.(20), whereas

$$\Omega^{[2]}\left(\frac{h}{2}\right) = \frac{h}{4}\left(A(0, Y_0) + \frac{h}{4}A\left(\frac{h}{2}, e^{\frac{h}{2}A(0,Y_0)}Y_0\right)\right) + \mathcal{O}(h^3)$$

to get a third-order scheme. The algorithm can be formulated *à la* RKMK as

$$\begin{aligned} u_1 &= 0 \\ k_1 &= hA(0, Y_0) \\ u_2 &= \frac{1}{2}k_1 \\ k_2 &= hA(h/2, e^{u_2}Y_0) \\ u_3 &= \frac{1}{4}(k_1 + k_2) \\ k_3 &= hA(h/2, e^{u_3}Y_0) \\ u_4 &= k_2 \\ k_4 &= hA(h, e^{u_4}Y_0) \\ v_3 &= \frac{1}{6}(k_1 + 4k_3 + k_4) - \frac{1}{3}[u_3, k_3] - \frac{1}{12}[u_4, k_4] \\ Y_1 &= e^{v_3}Y_0. \end{aligned} \tag{22}$$

This method closely resembles the RKMK scheme based on the Butcher tableau

$$\begin{array}{c|ccc} 0 & & & \\ \frac{1}{2} & \frac{1}{2} & & \\ 1 & -1 & 2 & \\ \hline & \frac{1}{6} & \frac{2}{3} & \frac{1}{6} \end{array}. \tag{23}$$

Table 1. Computational cost of different integrators for equation (2).

Order	Method	A evaluations	Commutators	Exponentials
2	RKMK	2	0	2
	Magnus	2	0	2
3	RKMK	3	1	3
	Magnus	4	1	4
4	RKMK	4	2	4
	Magnus	6	2	6

As a matter of fact, the technique developed in [27] for RKMK methods can also be applied here to reduce the number of commutators. The idea is to introduce transformed variables

$$Q_i = \sum_{j=1}^i V_{ij} k_j = \mathcal{O}(h^{q_i}),$$

where the constants V_{ij} are chosen in such a way that the resulting integers q_i are as large as possible. Then it is clear that

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

which allows us to discard terms of order higher than the method itself. Thus, for the integrator (22) we use instead

$$\begin{aligned} Q_1 &= k_1 = \mathcal{O}(h) & Q_2 &= k_2 - k_1 = \mathcal{O}(h^2) \\ Q_3 &= k_3 - k_2 = \mathcal{O}(h^3) & Q_4 &= k_4 - 2k_2 + k_1 = \mathcal{O}(h^3) \end{aligned} \quad (24)$$

and thus

$$\begin{aligned} u_1 &= 0 & u_2 &= \frac{1}{2} Q_1 \\ u_3 &= \frac{1}{2} Q_1 + \frac{1}{4} Q_2 & u_4 &= Q_1 + Q_2 \\ u_3 &= Q_1 + Q_2 + \frac{2}{3} Q_3 + \frac{1}{6} Q_4 - \frac{1}{6} [Q_1, Q_2]. \end{aligned} \quad (25)$$

The resulting algorithm involves four A evaluations, one commutator and three matrix exponentials per time step. It is therefore computationally more expensive than the corresponding RKMK scheme based on (23), as shown in table 1.

Order 4. With $m = 4$ we can use Simpson's rule to approximate $\Omega^{[4]}(h)$. The computations already done to find $\Omega^{[3]}(h)$ can be reused here, but it is still necessary to calculate $\Omega^{[3]}(h/2)$ up to order $\mathcal{O}(h^3)$ and perform two new A evaluations. The resulting algorithm is presented in the appendix and requires six A evaluations, two commutators and six matrix exponentials per time step.

By following this strategy new explicit numerical integrators for equation (2) of arbitrary order can be constructed. They are written in the same format as the RKMK class of schemes when $\Omega^{[1]}$ is discretized with a first-order quadrature. One could say, then, that the nonlinear Magnus expansion generates specific RKMK methods when the integrals are replaced by appropriate quadratures. But they are different in the following sense. Due to its iterative character, in a Magnus method of order p , unlike a conventional RKMK scheme, the internal stages provide approximations to the exact solution up to order $p - 1$. For this reason the new methods require, in general, more computational effort per time step, but on the other hand, variable step size and order techniques can be incorporated in a natural way into the algorithm [15, p 233] thus improving its overall efficiency.

In table 1 we display the computational cost of the numerical integrators based on the nonlinear Magnus expansion (5) obtained in this section in terms of A evaluations, number of commutators and matrix exponentials needed. For comparison we also include RKMK schemes of the same order based on standard Runge–Kutta methods (the explicit third-order Magnus method developed by Zanna [30] by relaxed collocation requires the same computational effort as the corresponding RKMK scheme).

With respect to the isospectral flow, it is possible to design a new family of numerical integrators from (12) with the same type of numerical quadratures. Essentially, the main difference with, say, algorithm (22), is that the action $e^{u_i} Y_0 e^{-u_i}$ is replaced by an appropriate truncation. As a consequence, the methods require only the evaluation of a single matrix exponential per step at the price of more commutators involved in the procedure. In particular, we have constructed a fourth-order scheme requiring twelve commutators.

3.2. Numerical examples

As an illustration of the algorithms proposed here we consider the isospectral flow $Y' = [A(t, Y), Y]$ governed by the skew-symmetric 3×3 matrix A whose upper-diagonal elements are

$$\begin{aligned} A_{12}(t, Y) &= \alpha_1(t)(Y_{22} - Y_{11}) \\ A_{13}(t, Y) &= \alpha_2(t)(Y_{23} - Y_{12}) \\ A_{23}(t, Y) &= \alpha_3(t)(Y_{33} - Y_{22}), \end{aligned} \tag{26}$$

with $\alpha_i(t) = \cos \omega_i t$. Observe that the case $\omega_i = 0$ corresponds to the Toeplitz annihilator problem (13) with $n = 3$.

Our purpose, rather than providing a complete characterization of the numerical integrators based on the Magnus expansion, is to show how they behave in practice in comparison with other integration schemes. In particular, we check the order of approximation, the computational efficiency and the qualitative description of the exact flow. To this end we only consider the general scheme presented in the appendix (adapted to the isospectral case) and the fourth-order algorithm built from (12) mentioned before, requiring 12 commutators and only one matrix exponential. They are compared with the Runge–Kutta–Munthe-Kaas integrator (RKMK4) based on ‘the’ Runge–Kutta method with Butcher tableau

$$\begin{array}{c|cccc} 0 & & & & \\ \frac{1}{2} & \frac{1}{2} & & & \\ \frac{1}{2} & 0 & \frac{1}{2} & & \\ 1 & 0 & 0 & 1 & \\ \hline & \frac{1}{6} & \frac{1}{3} & \frac{1}{3} & \frac{1}{6} \end{array} . \tag{27}$$

First, we carry out a numerical integration in the interval $t \in [0, 20]$ for several (constant) values of the step size h with initial condition $Y_0 = Q \operatorname{diag}(\eta) Q^T$. Here $\eta = [1, 2, 3]^T$ and Q is a randomly chosen orthogonal matrix. To study the efficiency of the methods, the error is determined at the final time by computing the Frobenius norm of the difference between the approximate and the exact solution matrices (we take as exact solution the outcome of integrating the problem with a higher-order approximation). Then this error is represented as a function of the computational effort measured in terms of floating point operations (evaluated using Matlab 5.3 built-in routine `flops`). The evaluation of matrix exponentials is done with the built-in function `expm`, although also the Rodrigues formula can be used to get an explicit expression for $\exp u$, $u \in \mathfrak{so}(3)$ [23, p 261]. The corresponding efficiency curves are plotted in figure 1.

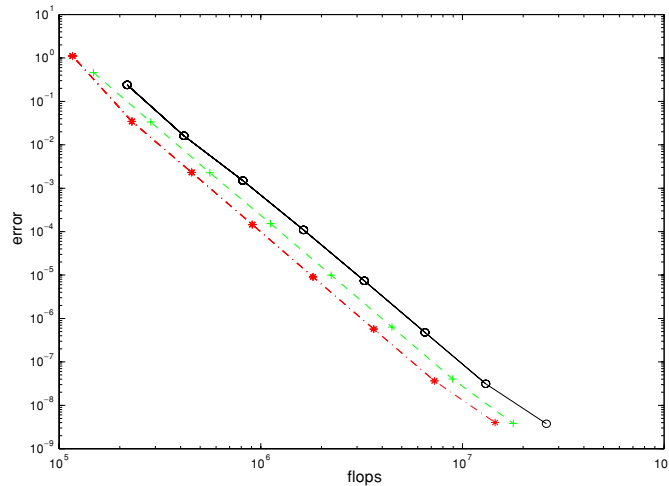


Figure 1. Error versus floating point operations obtained with the fourth-order integrators based on Magnus (solid line with circles), RKMK4 (dashed line with +) and a Magnus method adapted to isospectral flows involving 12 commutators and 1 matrix exponential per time step (dash-dotted line with *).

Observe that, in agreement with the theoretical estimates of table 1, RKMK4 is more efficient than the algorithm (A.1) based on the general Magnus expansion, whereas the efficiency shown by the especially adapted Magnus scheme is slightly better. This is quite remarkable in view of the more complex structure of the algorithms based on the nonlinear Magnus expansion.

Next we take $\omega_i = 0$ and the initial condition $Y_0 = \text{diag}(2, 5, 9)$. In that case, solving the corresponding problem (7) allows us to find a symmetric 3×3 Toeplitz matrix that possesses the same sequence of eigenvalues as Y_0 , i.e., we can solve the inverse eigenvalue problem [11, 18]. Note that $A(Y) = O$ if Y is a Toeplitz matrix (hence the name of the Toeplitz annihilator), so that a solution of the isospectral problem defined by A is a fixed point of the flow.

When the problem is discretized with a standard integrator (such as multistep or Runge–Kutta schemes), the numerical flow fails to converge to the correct Toeplitz matrix, since they do not preserve the eigenvalues along the evolution, whereas isospectral methods tend to a Toeplitz matrix with the right eigenvalues [18]. This can be clearly observed in figure 2, where the ‘Toeplitz error’ $\log \|A(Y_n)\|_F$ is plotted as a function of time when Y_n is obtained with RKMK4 and the new fourth-order isospectral Magnus method, both with step size $h = 1/6$. As t increases, the Toeplitz error becomes progressively smaller and eventually Y_n converges to a symmetric Toeplitz matrix up to machine accuracy, whereas the eigenvalues are preserved by construction. It is worth noticing, however, that the approximation obtained by Magnus attains the numerical limit with fewer iterations than RKMK4.

4. Application to highly oscillatory nonlinear ODEs

Suppose now that we are given the ODE system

$$\mathbf{y}' = A(t, \mathbf{y})\mathbf{y}, \quad \mathbf{y}(0) = \mathbf{y}_0 \in \mathcal{M}, \quad (28)$$

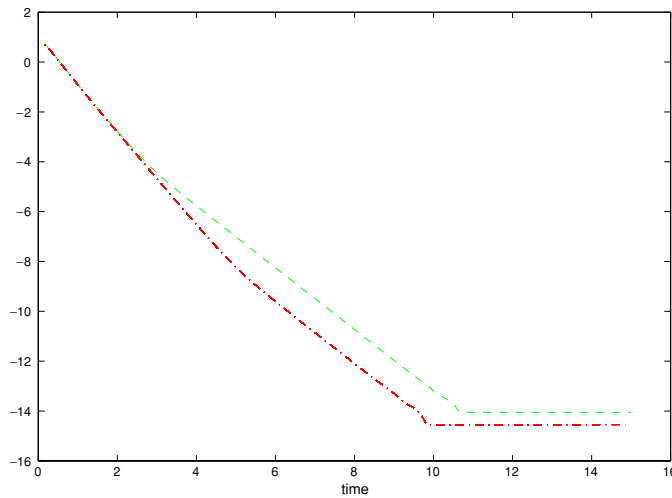


Figure 2. Toeplitz error $\log \|A(Y_n)\|_F$ versus time for a 3×3 symmetric Toeplitz inverse eigenvalue problem when solved by RKMK4 (dashed line) and the new isospectral integrator based on Magnus expansion (dash-dotted line).

where $\mathcal{M} \subset \mathbb{R}^d$ is a homogeneous space acted upon by the Lie group \mathcal{G} and $A : \mathbb{R}_+ \times \mathcal{M} \rightarrow \mathfrak{g}$. In that case $\mathbf{y}(t)$ evolves in \mathcal{M} , as well as the approximations obtained with the explicit Magnus expansion (5).

Let us assume in addition that the solution of (28) oscillates rapidly. In the linear case a conveniently chosen transformation prior to the application of the Magnus expansion allows us to get very accurate results [17]. We generalize this approach to the nonlinear setting.

Suppose that we have computed $\mathbf{y}_n \approx \mathbf{y}(t_n)$ and wish to advance to $t_{n+1} = t_n + h$. The idea is to consider a new variable $\mathbf{z}(x)$ such that

$$\mathbf{y}(t_n + x) = e^{xA(t_n, \mathbf{y}_n)} \mathbf{z}(x). \tag{29}$$

Then

$$\frac{d\mathbf{z}}{dx} = B(x, \mathbf{z}(x))\mathbf{z}, \quad \mathbf{z}(0) = \mathbf{y}_n \tag{30}$$

with

$$B(x, \mathbf{z}(x)) = F^{-1}(x)[A(t_n + x, F(x)\mathbf{z}(x)) - A(t_n, \mathbf{y}_n)]F(x) \tag{31}$$

and $F(x) = \exp[xA(t_n, \mathbf{y}_n)]$. We note for future use that $B(0, \mathbf{z}(0)) = O$.

Observe that the new variable $\mathbf{z}(x)$ may also be seen as a correction to the solution provided by the first-order term $\Omega^{[1]}$ of the Magnus expansion (discretized with Euler’s method). For this reason one expects that if the system (30) is solved with the nonlinear Magnus expansion the error in the corresponding approximations will be significantly smaller than with the standard algorithm, even when the same quadrature rules are used [17]. But in the highly oscillatory case other especially tailored quadratures exist which provide excellent results [20].

To illustrate the main features of the nonlinear modified Magnus expansion applied to the highly oscillatory system (28), let us consider equations of the form

$$y'' + a(t, y, y')y = 0, \quad y(0) = y_0, \quad y'(0) = y'_0, \tag{32}$$

where it is assumed that $a(t, y, y') \gg 1$. Particular examples are the Emden–Fowler ($a = ty^2$), the Lane–Emden ($a = (y/t)^{n-1}$) and the Thomas–Fermi ($a = -(y/t)^{1/2}$) equations [31].

When (32) is written in a matrix form, we obtain (28) with

$$A(t, \mathbf{y}) = \begin{pmatrix} 0 & 1 \\ -a(t, \mathbf{y}) & 0 \end{pmatrix}$$

and $\mathbf{y} = (y, y')^T$. Denoting by $\theta_n \equiv \sqrt{a(t_n, \mathbf{y}_n)}$, it is clear that

$$F(x) = e^{xA(t_n, \mathbf{y}_n)} = \begin{pmatrix} \cos x\theta_n & \theta_n^{-1} \sin x\theta_n \\ -\theta_n \sin x\theta_n & \cos x\theta_n \end{pmatrix},$$

whereas for the new matrix B one gets, after some algebra,

$$B(x, \mathbf{z}(x)) = \frac{1}{4}(\theta_n^2 - \theta^2(x))[2M_1 + M_2 e^{2i\theta_n x} + M_3 e^{-2i\theta_n x}] \quad (33)$$

with $\theta^2(x) \equiv a(t_n + x, F(x)\mathbf{z}(x))$ and

$$M_1 = \begin{pmatrix} 0 & -\theta_n^{-2} \\ 1 & 0 \end{pmatrix}, \quad M_2 = \begin{pmatrix} i\theta_n^{-1} & \theta_n^{-2} \\ 1 & -i\theta_n^{-1} \end{pmatrix}, \quad M_3 = \begin{pmatrix} -i\theta_n^{-1} & \theta_n^{-2} \\ 1 & i\theta_n^{-1} \end{pmatrix}.$$

This is the expression required for applying the nonlinear Magnus expansion (5). The first term is given by

$$\begin{aligned} \Omega^{[1]}(x) &= \int_0^x B(\tau, \mathbf{y}_n) d\tau = \frac{1}{2} \int_0^x (\theta_n^2 - \theta_1^2(\tau)) d\tau M_1 \\ &\quad + \frac{1}{4} \int_0^x (\theta_n^2 - \theta_1^2(\tau)) e^{2i\theta_n \tau} d\tau M_2 + \frac{1}{4} \int_0^x (\theta_n^2 - \theta_1^2(\tau)) e^{-2i\theta_n \tau} d\tau M_3 \\ &\equiv I_0(x)M_1 + I_+(x)M_2 + I_-(x)M_3, \end{aligned}$$

where now $\theta_1^2(\tau) \equiv a(t_n + \tau, F(\tau)\mathbf{y}_n)$. Since $B(0, \mathbf{y}_n) = O$, any quadrature rule that uses only the values of $\theta_n^2 - \theta_1^2(\tau)$ at the endpoints requires only the value at x (the value at the origin is zero). For the non-oscillatory part $I_0(x)$ we can use the trapezoidal rule

$$I_0(x) \approx x\varphi_1(x), \quad \text{with} \quad \varphi_1(x) \equiv \frac{1}{4}(\theta_n^2 - \theta_1^2(x)).$$

For $I_{\pm}(x)$ it seems appropriate to apply Filon–Lobatto quadratures. With this class of methods one has in general

$$\int_0^x f(\tau) e^{\pm 2i\theta_n \tau} d\tau \approx b_1^{\pm}(\theta_n) f(0) + b_2^{\pm}(\theta_n) f(x)$$

with

$$b_2^{\pm}(\theta_n) = \frac{e^{\pm 2i\theta_n x} - 1}{\pm 2i\theta_n} + \frac{e^{\pm 2i\theta_n x} - 1}{4x\theta_n^2}.$$

Consequently, putting all the pieces together,

$$\Omega^{[1]}(x) = \varphi_1(x)(xM_1 + b_2^+(\theta_n)M_2 + b_2^-(\theta_n)M_3) \quad (34)$$

or equivalently

$$\Omega^{[1]}(x) = \varphi_1(x) \begin{pmatrix} \frac{\cos 2\theta_n x}{\theta_n^2} - \frac{\sin 2\theta_n x}{2\theta_n^3 x} & -\frac{x}{\theta_n^2} + \frac{\sin 2\theta_n x}{\theta_n^3} - \frac{1 - \cos 2\theta_n x}{2\theta_n^4 x} \\ x + \frac{\sin 2\theta_n x}{\theta_n} - \frac{1 - \cos 2\theta_n x}{2\theta_n^2 x} & -\frac{\cos 2\theta_n x}{\theta_n^2} + \frac{\sin 2\theta_n x}{2\theta_n^3 x} \end{pmatrix}.$$

For $\Omega^{[2]}(x) = \int_0^x B(\tau, e^{\Omega^{[1]}(\tau)}\mathbf{y}_n) d\tau$ one gets the same expression (34), but now $\varphi_1(x)$ has to be replaced by

$$\varphi_2(x) \equiv \frac{1}{4}(\theta_n^2 - \theta_2^2(x)), \quad \text{where} \quad \theta_2^2(x) = a(t_n + x, F(x)e^{\Omega^{[1]}(x)}\mathbf{y}_n).$$

Similar considerations apply to higher-order terms, although the analysis is obviously more involved. If the truncated Magnus solution of (30) is $\mathbf{z}(x) = \exp(\Omega^{[k]}(x))\mathbf{y}_n$, the approximation obtained in this way has the form

$$\mathbf{y}_{n+1} = e^{hA(t_n, \mathbf{y}_n)} e^{\Omega^{[k]}(h)} \mathbf{y}_n, \quad n \in \mathbb{Z}.$$

For the particular case of the Emden–Fowler equation, one has $\theta_n = \sqrt{t_n}y_n$ and

$$\theta_1^2(h) = (t_n + h)[(F(h)\mathbf{y}_n)_1]^2, \quad \theta_2^2(h) = (t_n + h)[(F(h)e^{\Omega^{[1]}(h)}\mathbf{y}_n)_1]^2.$$

5. Conclusions

The nonlinear Magnus expansion we propose in this paper can be considered as a natural generalization of the usual expansion for linear problems. As well as this, it provides explicit integrators in terms of integrals of nested commutators and it is amenable to standard procedures to reduce the total number of commutators. Although only methods up to order 4 have been presented here, the same strategy can in principle be applied to construct higher-order schemes preserving the main qualitative properties of the exact solution. The new integrators require in general more computational effort per time step than other well-known Lie-group methods, both in function evaluations and in computations of matrix exponentials. Nevertheless, for particular types of problems, such as isospectral flows, it is possible to adapt appropriately the procedure and reduce the computational cost per step. In fact, when applied to specific isospectral problems, their efficiency is similar to the Runge–Kutta–Munthe-Kaas class of Lie-group integrators. They also provide the correct asymptotic description when applied to solve the inverse eigenvalue problem of Toeplitz symmetric matrices.

The numerical schemes obtained from the Magnus expansion can be written in the same format as the RKMK explicit methods when the integrals are replaced by appropriate quadratures. Given a method of order p , its internal stages provide approximations to the exact solution from order 1 up to order $p - 1$, and so their computational complexity is generally higher for equations defined on Lie groups.

In any case, one should be aware that RKMK integrators, like all Runge–Kutta methods, are implicitly based on quadrature of interpolatory kind (e.g. Gauss, Lobatto, Radau, etc) and, as such, they are not very appropriate for highly oscillatory equations. With schemes based on the nonlinear Magnus expansion, however, one can use any quadrature rule one may wish, including Filon-type methods, and thus they constitute a natural option to integrate highly oscillatory differential equations.

We can conclude that the nonlinear Magnus expansion introduced here is a very flexible tool to analyse nonlinear equations defined on Lie groups and/or homogeneous spaces acted upon by Lie groups. It allows us to use different quadrature rules and even in some cases to work with exact integrals. At the same time, the procedure can be modified to cope with highly oscillatory systems of nonlinear ODEs in conjunction with especially adapted quadratures.

Although only numerical aspects of the expansion have been considered in this paper, the Magnus expansion could also be useful in perturbative analysis of nonlinear differential equations as a device providing continuous approximations depending on the parameters of the problem and even for certain delay-differential equations.

Acknowledgments

The work of FC has been partially supported by Ministerio de Educación y Ciencia (Spain) under project MTM2004-00535 (co-financed by the ERDF of the European Union)

and Generalitat Valenciana (GV2003-002), while on sabbatical leave at the University of Cambridge.

Appendix

In this appendix we present a fourth-order algorithm for the numerical integration of the general Lie equation $Y' = A(t, Y)Y$ based on the nonlinear Magnus expansion (5) for $t_{n+1} = t_n + h$:

$$\begin{aligned}
 u_1 &= 0 \\
 k_1 &= hA(t_n, Y_n), & Q_1 &= k_1 \\
 u_2 &= \frac{1}{2}Q_1 \\
 k_2 &= hA\left(t_n + \frac{h}{2}, e^{u_2}Y_0\right), & Q_2 &= k_2 - k_1 \\
 u_3 &= \frac{1}{2}Q_1 + \frac{1}{4}Q_2 \\
 k_3 &= hA\left(t_n + \frac{h}{2}, e^{u_3}Y_0\right), & Q_3 &= k_3 - k_2 \\
 u_4 &= Q_1 + Q_2 & & \tag{A.1} \\
 k_4 &= hA(t_n + h, e^{u_4}Y_0), & Q_4 &= k_4 - 2k_2 + k_1 \\
 u_5 &= \frac{1}{2}Q_1 + \frac{1}{4}Q_2 + \frac{1}{3}Q_3 - \frac{1}{24}Q_4 - \frac{1}{48}[Q_1, Q_2] \\
 k_5 &= hA\left(t_n + \frac{h}{2}, e^{u_5}Y_0\right), & Q_5 &= k_5 - k_2 \\
 u_6 &= Q_1 + Q_2 + \frac{2}{3}Q_3 + \frac{1}{6}Q_4 - \frac{1}{6}[Q_1, Q_2] \\
 k_6 &= hA(t_n + h, e^{u_6}Y_0), & Q_6 &= k_6 - 2k_2 + k_1 \\
 v &= Q_1 + Q_2 + \frac{2}{3}Q_5 + \frac{1}{6}Q_6 - \frac{1}{6}\left[Q_1, Q_2 - Q_3 + Q_5 + \frac{1}{2}Q_6\right] \\
 Y_{n+1} &= e^v Y_n.
 \end{aligned}$$

Remarks

- (i) The computation of u_5, k_5 is independent of u_6, k_6 and it is required only to obtain v (which differs from $\Omega^{[4]}(t_n + h)$ only in $\mathcal{O}(h^5)$ terms that have no effect on the order).
- (ii) The above algorithm comprises also lower-order methods: if we take $v = k_1$ we have a first-order scheme; if $v = k_2$ then a second-order method results; finally, by computing only up to u_6 (but not u_5, k_5, Q_5) we recover the third-order method (22). It might be therefore implemented so that not only the step size but also the order can be changed at each step, similarly to extrapolation methods [15, p 233].
- (iii) This algorithm can also be directly applied to the isospectral flow $Y' = [A(t, Y), Y]$ with the replacement of $e^{u_i}Y_n$ by the action $e^{u_i}Y_n e^{-u_i}$ in the computation of k_i and finally $Y_{n+1} = e^v Y_n e^{-v}$.

References

- [1] Blanes S, Casas F, Oteo J A and Ros J 1998 Magnus and Fer expansions for matrix differential equations: the convergence problem *J. Phys. A: Math. Gen.* **31** 259–68
- [2] Blanes S, Casas F and Ros J 2000 Improved high order integrators based on the Magnus expansion *BIT* **40** 434–50
- [3] Blanes S, Casas F and Ros J 2002 High order optimized geometric integrators for linear differential equations *BIT* **42** 262–84
- [4] Blanes S and Moan P C 2001 Splitting methods for non-autonomous Hamiltonian equations *J. Comput. Phys.* **170** 205–30
- [5] Bloch A M and Iserles A 2005 Commutators of skew-symmetric matrices *Int. J. Bifur. Chaos* **15** 793–801
- [6] Bloch A M and Iserles A 2005 On an isospectral Lie–Poisson system and its Lie algebra *University of Cambridge Technical Report DAMTP NA2005/01*
- [7] Bloch A M, Iserles A, Marsden J E and Ratiu T S A class of integrable geodesic flows on the symplectic group *Preprint ArXiv/math-ph/0512093*
- [8] Brockett R W 1991 Dynamical systems that sort lists, diagonalize matrices and solve linear programming problems *Linear Algebra Appl.* **146** 79–91
- [9] Calvo M P, Iserles A and Zanna A 1997 Numerical solution of isospectral flows *Math. Comput.* **66** 1461–86
- [10] Casas F 2004 Numerical integration methods for the double-bracket flow *J. Comput. Appl. Math.* **166** 477–95
- [11] Chu M T 1995 On the differential equation $\frac{dX}{dt} = [X, k(X)]$ where k is a Toeplitz annihilator *Technical Report* North Carolina State University
- [12] Chu M T and Driessell K R 1990 The projected gradient method for least squares matrix approximations with spectral constraints *SIAM J. Numer. Anal.* **27** 1050–60
- [13] Dormand J R 1996 *Numerical Methods for Differential Equations: A Computational Approach* (Boca Raton: CRC Press)
- [14] Hairer E, Lubich C and Wanner G 2002 *Geometric Numerical Integration. Structure-Preserving Algorithms for Ordinary Differential Equations (Springer Series Comput. Math. vol 31)* (Berlin: Springer)
- [15] Hairer E, Nørsett S P and Wanner G 1993 *Solving Ordinary Differential Equations I* 2nd edn (*Springer Series Comput. Math.* vol 8) (Berlin: Springer)
- [16] Iserles A 2002 On the discretization of double-bracket flows *Found. Comput. Math.* **2** 305–29
- [17] Iserles A 2002 On the global error of discretization methods for highly oscillatory ordinary differential equations *BIT* **42** 561–99
- [18] Iserles A, Munthe-Kaas H Z, Nørsett S P and Zanna A 2000 Lie group methods *Acta Numerica* **9** 215–365
- [19] Iserles A and Nørsett S P 1999 On the solution of linear differential equations in Lie groups *Philos. Trans. R. Soc. A* **357** 983–1019
- [20] Iserles A and Nørsett S P 2004 On quadrature methods for highly oscillatory integrals and their implementation *BIT* **44** 755–72
- [21] Klarsfeld S and Oteo J A 1989 Recursive generation of higher-order terms in the Magnus expansion *Phys. Rev. A* **39** 3270–3
- [22] Magnus W 1954 On the exponential solution of differential equations for a linear operator *Commun. Pure Appl. Math.* **7** 649–73
- [23] Marsden J E and Ratiu T S 1994 *Introduction to Mechanics and Symmetry* (New York: Springer)
- [24] Moore J B, Mahony R E and Helmke U 1994 Numerical gradient algorithms for eigenvalue and singular value calculations *SIAM J. Matrix Anal. Appl.* **15** 881–902
- [25] Munthe-Kaas H 1998 Runge–Kutta methods on Lie groups *BIT* **38** 92–111
- [26] Munthe-Kaas H 1999 High order Runge–Kutta methods on manifolds *Appl. Numer. Math.* **29** 115–27
- [27] Munthe-Kaas H and Owren B 1999 Computations in a free Lie algebra *Phil. Trans. R. Soc. A* **357** 957–81
- [28] Munthe-Kaas H and Zanna A 1997 Numerical integration of differential equations on homogeneous manifolds *Foundations of Computational Mathematics* ed F Cucker and M Shub (Berlin: Springer) pp 305–15
- [29] Nayfeh A 1973 *Perturbation Methods* (New York: Wiley)
- [30] Zanna A 1999 Collocation and relaxed collocation for the Fer and the Magnus expansion *SIAM J. Numer. Anal.* **36** 1145–82
- [31] Zwillinger D 1992 *Handbook of Differential Equations* 2nd edn (Boston: Academic)