Explicit Magnus expansions for nonlinear equations

Fernando Casas[a] **and Arieh Iserles**[b]§

[a] Departament de Matemàtiques, Universitat Jaume I, 12071-Castellón, Spain
[b] Department of Applied Mathematics and Theoretical Physics, University of Cambridge, Wilberforce Rd., Cambridge CB3 0WA, United Kingdom

E-mail: Fernando.Casas@mat.uji.es, ai@damtp.cam.ac.uk

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 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.

AMS classification scheme numbers: 65L05, 41A55, 22E60

§ To whom correspondence should be addressed

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

$$Y' = A(t)Y, \qquad Y(0) = Y_0, \tag{1}$$

where A is a $n \times n$ 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 [21]. In particular, the first terms read explicitly

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

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 [18] by using graph theory, whereas in [20] 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 NMR 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 [18] 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 discretising differential equations which preserve their known qualitative features, such as invariant quantities and geometric structure [13]. 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. On the one hand, 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 [18]. On the other hand, 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 generalise 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 [27] which in general cannot compete in efficiency with other classes of geometric integrators such as splitting and composition methods. An interesting alternative have 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

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

where $\mathcal{G}$ is a matrix Lie group, $A : \mathbb{R}_+ \times \mathcal{G} \longrightarrow \mathfrak{g}$ 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 ($\mathcal{G} \equiv \mathrm{SO}(n)$), Hamiltonian dynamics ($\mathcal{G} \equiv \mathrm{Sp}(n)$) and Quantum Mechanics ($\mathcal{G} \equiv \mathrm{SU}(n)$). 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) [26]. 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 eq. (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 [24, 17].

Here new Lie group solvers up to order four for (2) are presented. The new schemes are explicit by design and are expressed in terms of integrals which can be replaced by different quadrature rules. It is also possible to change the step size and even the order at each integration step, so that although their computational effort per step exceeds some other algorithms, with an optimal implementation one can get a more efficient Lie group solver for certain nonlinear problems.

The plan of the paper is as follows. In section 2 an 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 on several numerical examples, 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, just 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 our formalism is to represent the solution of (2) in the form

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

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

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

Here

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

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

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

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

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

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

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

The next step to get explicit approximations is to truncate appropriately the $d\exp^{-1}$ operator in the above expansion. Roughly speaking, when the whole series for $d\exp^{-1}$ 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

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

it follows at once that

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

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 $d\exp^{-1}$ at the $k=0$ term and take

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

With greater generality, we let

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

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 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 generalises 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

$$\Omega(t) = \sum_{l=1}^{\infty} t^l\, \omega_l$$

with $\omega_1 = A(Y_0)$, $\omega_2 = \frac{1}{2}G_1$ and, for $l \geq 3$,

$$l\,\omega_l = G_{l-1} + \sum_{j=1}^{l-1} \frac{B_j}{j!} \sum_{\substack{k_1+\cdots+k_j=l-1 \\ k_1\geq 1,\ldots,k_j\geq 1}} \mathrm{ad}_{\omega_{k_1}} \cdots \mathrm{ad}_{\omega_{k_j}} A(Y_0)$$

$$+\sum_{j=1}^{l-2}\left(\sum_{s=0}^{j}\frac{B_s}{s!}\sum_{\substack{k_1+\cdots+k_s=j\\ k_1\geq 1,\ldots,k_s\geq 1}}\mathrm{ad}_{\omega_{k_1}}\cdots\mathrm{ad}_{\omega_{k_s}}\right)G_{l-1-j} \tag{6}$$

$$+\sum_{j=1}^{l-2}\frac{B_j}{j!}\sum_{\substack{k_1+\cdots+k_j=l-2\\ k_1\geq 1,\ldots,k_j\geq 1}}\mathrm{ad}_{\omega_{k_1}}\cdots\mathrm{ad}_{\omega_{k_j}}G_1.$$

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

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

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

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

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

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

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

but $\hat{G}_m = \hat{G}_m(Y_0;\hat{\omega}_1,\hat{\omega}_2,\ldots,\hat{\omega}_{m-1})$, whereas $G_m = G_m(Y_0;\omega_1,\omega_2,\ldots,\omega_m)$, 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 a exponential representation of the solution for the differential system

$$Y' = [A(Y),Y], \qquad Y(0)=Y_0\in \mathrm{Sym}(n). \tag{7}$$

Here $\mathrm{Sym}(n)$ stands for the set of $n\times n$ symmetric real matrices and the (sufficiently smooth) function A maps $\mathrm{Sym}(n)$ 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 $\mathrm{Sym}(n)$ 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 and 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 $Q(t) \in \mathrm{SO}(n)$ (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,

$$Y(t) = Q(t)Y_0Q^T(t). \tag{8}$$

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

$$Q' = A(t, QY_0Q^T)\, Q, \qquad Q(0) = I, \tag{9}$$

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

$$Y(t) = \mathrm{e}^{\Omega(t)} Y_0\, \mathrm{e}^{-\Omega(t)}, \qquad t \geq 0, \qquad \Omega(t) \in \mathfrak{so}(n), \tag{10}$$

then the corresponding equation for Ω reads

$$\Omega' = d\exp_\Omega^{-1}\left(A(\mathrm{e}^{\Omega} Y_0 \mathrm{e}^{-\Omega})\right), \qquad \Omega(0) = O. \tag{11}$$

In a similar way as for eq. (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

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

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

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

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

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

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., $A(\alpha_1 Y_1 + \alpha_2 Y_2) = \alpha_1 A(Y_1) + \alpha_2 A(Y_2)$. Some relevant examples include the double bracket flow, the periodic Toda lattice (to be introduced later on), the Toeplitz annihilator defined by

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

[11] 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 [27].

When the iterative scheme (12) is applied to a 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):

$$\begin{aligned} \omega_1 &= A(Y_0) \\ 2\,\omega_2 &= A(\operatorname{ad}_{\omega_1} Y_0) \\ l\,\omega_l &= \sum_{j=1}^{l-1} \frac{1}{j!} \sum_{\substack{k_1+\cdots+k_j=l-1 \\ k_1\ge 1,\ldots,k_j\ge 1}} A(\operatorname{ad}_{\omega_{k_1}} \cdots \operatorname{ad}_{\omega_{k_j}} Y_0) \\ &\quad + \sum_{j=1}^{l-1} \frac{B_j}{j!} \sum_{\substack{k_1+\cdots+k_j=l-1 \\ k_1\ge 1,\ldots,k_j\ge 1}} \operatorname{ad}_{\omega_{k_1}} \cdots \operatorname{ad}_{\omega_{k_j}} A(Y_0) \\ &\quad + \sum_{j=2}^{l-1} \left(\sum_{s=1}^{j-1} \frac{B_s}{s!} \sum_{\substack{k_1+\cdots+k_s=j-1 \\ k_1\ge 1,\ldots,k_s\ge 1}} \operatorname{ad}_{\omega_{k_1}} \cdots \operatorname{ad}_{\omega_{k_s}} \right) \\ &\qquad \left(\sum_{p=1}^{l-j} \frac{1}{p!} \sum_{\substack{k_1+\cdots+k_p=l-j \\ k_1\ge 1,\ldots,k_p\ge 1}} A(\operatorname{ad}_{\omega_{k_1}} \cdots \operatorname{ad}_{\omega_{k_p}} Y_0) \right) \qquad l \ge 3 \end{aligned} \tag{14}$$

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

$$\|[X, Y\| \le \mu \|X\| \, \|Y\|$$

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 $t_c = \dfrac{\xi}{\mu K \|Y_0\|}$ and

$$\xi = \int_0^{2\pi} \frac{\mathrm{e}^{-x}}{2 + \frac{x}{2}(1 - \cot \frac{x}{2})} dx \simeq 0.688776\ldots$$

Example: The double bracket flow. The double bracket equation

$$Y' = [[Y, N], Y], \qquad Y(0) = Y_0 \in \mathrm{Sym}(n) \tag{15}$$

was introduced by Brocket [8] and Chu & Driessel [11] to solve certain standard problems in applied mathematics, although similar equations also appear in the formulation of physical theories such as micromagnetics [23]. Here N is a constant matrix in $\mathrm{Sym}(n)$. 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 [15] with the convergence domain established in [10]. □

3. Numerical integrators based on the Magnus expansion

3.1. The new methods

Usually, the integrals appearing in the nonlinear Magnus expansion (5) (or (12)) cannot be evaluated in practice for a given matrix A. Hence, unless they are replaced by affordable quadratures, the overall scheme is of little value as a numerical algorithm. 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.

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

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

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

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 discretise $\Omega^{[2]}$ with the trapezoidal rule, then

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

In fact, 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

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

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

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

If, on the other hand, $\Omega^{[1]}$ is discretised with Euler and $\Omega^{[2]}$ with the midpoint rule,

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

we retrieve exactly the RKMK Heun method [17, p. 355].

Not all explicit RKMK methods can be recovered in this way and, moreover, there are some interesting differences. In particular, RKMK methods always require to discretise $\Omega^{[1]}$ with a first-order quadrature, something not necessary for schemes based on the preceding nonlinear Magnus expansion.

Order 3. In addition to eqs. (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 $A_2(s) \equiv A(s, \mathrm{e}^{\Omega^{[2]}(s)}Y_0)$. If we use Simpson's rule to approximate (21), then

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

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

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

to get a 3th-order scheme. The algorithm can be formulated *à la* RKMK as

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

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

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

As a matter of fact, the technique developed in [25] 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

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

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

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

and thus

$$\begin{aligned}
u_1 &= 0 & \qquad u_2 &= \frac{1}{2}Q_1 \\
u_3 &= \frac{1}{2}Q_1 + \frac{1}{4}Q_2 & \qquad u_4 &= Q_1 + Q_2 \\
v_3 &= Q_1 + Q_2 + \frac{2}{3}Q_3 + \frac{1}{6}Q_4 - \frac{1}{6}[Q_1, Q_2]
\end{aligned} \tag{25}$$

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

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

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 two new A evaluations. The resulting algorithm is collected in the appendix and requires six A evaluations, two commutators and six matrix exponentials per time step.

In Table 1 we display the computational cost of the numerical integrators based on the nonlinear Magnus expansion (5) obtained before in terms of A evaluations, number of commutators and matrix exponentials needed. For comparison we also include RKMK schemes of the same order (the explicit third-order Magnus method developed by Zanna [27] by relaxed collocation requires the same computational effort as the corresponding RKMK scheme).

As it is evident from the table, the new methods require more computational effort per time step than the corresponding RKMK schemes of the same order. This is due to the fact that it is necessary to compute lower order approximations to Ω at the internal stages of the method. But this allows in a natural way to use local extrapolation for controlling the step size at each iteration. Moreover, even a variable order technique could be incorporated into the algorithm [14, p. 233], thus improving a great deal its overall efficiency. In addition, for particular types of problems some other especially adapted quadrature rules may be employed and/or some of the integrals could also be exactly evaluated. This is in contrast with the RKMK class of methods.

With respect to the isospectral flow $Y' = [A(Y), Y]$, it is possible to design a new family of numerical integrators from (12). Essentially, the main difference with, say, algorithm (22), is that the action $\mathrm{e}^{u_i} Y_0 \mathrm{e}^{-u_i}$ is replaced by an appropriate truncation. As a consequence, the methods require only the evaluation of a single matrix exponential per step but the number of commutators involved is much larger. For instance, we have constructed a fourth-order scheme requiring twelve commutators.

3.2. Methods for quasilinear isospectral flows

When the isospectral flow is quasilinear, the explicit solution obtained in section 2.2 for any term in the Magnus expansion (eq. (14)) can be used to construct especially adapted numerical integrators requiring much less computational effort per iteration than the previous algorithms. Let us illustrate this point by considering, in particular, methods up to order four.

From eq. (14) it is clear that

$$\begin{aligned} 3\omega_3 &= A\left(\mathrm{ad}_{\omega_2} Y_0 + \frac{1}{2}\mathrm{ad}^2_{\omega_1} Y_0\right) - \frac{1}{2}\mathrm{ad}_{\omega_1}\omega_2 \\ 4\omega_4 &= A\left(\mathrm{ad}_{\omega_3} Y_0 + \frac{1}{2}\mathrm{ad}_{\omega_1}\mathrm{ad}_{\omega_2} Y_0 + \frac{1}{2}\mathrm{ad}_{\omega_2}\mathrm{ad}_{\omega_1} Y_0 + \frac{1}{6}\mathrm{ad}^3_{\omega_1} Y_0\right) \\ &\quad - \mathrm{ad}_{\omega_1}\omega_3 - \frac{1}{6}\mathrm{ad}^2_{\omega_1}\omega_2. \end{aligned} \tag{26}$$

These expressions can be further simplified by grouping terms in the nested commutators to reduce its number to a minimum. As a result one has the following algorithm:

$$\begin{aligned} \theta_1 &= Y_0 \\ \omega_1 &= A(\theta_1) \\ d_1 &= [\omega_1, Y_0] \\ \theta_2 &= d_1 \\ \omega_2 &= \frac{1}{2}A(\theta_2); \quad \rightarrow \quad \Omega^{[2]}(h) = \omega_1 h + \omega_2 h^2 \\ d_2 &= [\omega_2, Y_0]; \qquad d_3 = [\omega_1, d_1]; \qquad d_4 = [\omega_1, \omega_2] \\ \theta_3 &= d_2 + \frac{1}{2}d_3 \\ \omega_3 &= \frac{1}{3}A(\theta_3) - \frac{1}{6}d_4 \quad \rightarrow \quad \Omega^{[3]}(h) = \Omega^{[2]}(h) + \omega_3 h^3 \\ d_5 &= [\omega_3 - d_4/2, Y_0]; \qquad d_6 = [\omega_1, d_2 + d_3/6]; \qquad d_7 = \frac{1}{3}[\omega_1, A(\theta_3)] \\ \theta_4 &= d_5 + d_6 \\ \omega_4 &= \frac{1}{4}(A(\theta_4) - d_7) \\ \Omega^{[4]}(h) &= \sum_{i=1}^{4} \omega_i h^i \end{aligned} \tag{27}$$

and finally

$$Y(t_k + h) = \mathrm{e}^{\Omega^{[m]}(h)}\, Y(t_k)\, \mathrm{e}^{-\Omega^{[m]}(h)},$$

where Y_0 has to be replaced by $Y(t_k)$ in the corresponding expression of $\Omega^{[m]}(h)$, $m = 2, 3, 4$. These methods generalise those obtained in [10] for the double bracket flow and require only one matrix exponential per time step. Also the number of evaluations of the A matrix has been reduced as shown in Table 2, where we also collect the number of commutators involved and compare this new family of methods with RKMK integrators up to order five for the same class of problems. The fifth-order RKMK method is based

Order	Method	A evaluations	Commutators	Exponentials
2	RKMK	2	0	2
	Magnus-QL	2	1	1
3	RKMK	3	1	3
	Magnus-QL	3	4	1
4	RKMK	4	2	4
	Magnus-QL	4	7	1
5	RKMK	7	6	6
	Magnus-QL	5	15	1

Table 2. Computational cost of RKMK methods and the new integrators based on the Magnus expansion for quasilinear isospectral flows (Magnus-QL).

on the well known DOPRI5(4) Runge–Kutta scheme, which has a total of seven stages (the seventh stage is used only for error estimation) [12].

Observe that, in general, an mth-order Magnus method only requires m evaluations of the matrix A. From the table one expects that, unless the system has a special structure allowing a fast computation of the exponential of the matrices involved, the new methods will be more efficient than the RKMK class of algorithms, even with a fixed step size implementation.

3.3. Numerical example: the periodic Toda lattice

As an illustration of the algorithms proposed in this section we consider a three-particle periodic Toda lattice, i.e., three particles on a ring governed by the Hamiltonian function

$$H(q,p) = \frac{1}{2}(p_1^2 + p_2^2 + p_3^2) + \mathrm{e}^{-(q_2-q_1)} + \mathrm{e}^{-(q_3-q_2)} + \mathrm{e}^{-(q_1-q_3)} - 3.$$

It is well known that the corresponding equations of motion can be written in the form $Y' = [A(Y), Y]$ with

$$Y = \begin{pmatrix} \beta_1 & \alpha_1 & \alpha_3 \\ \alpha_1 & \beta_2 & \alpha_2 \\ \alpha_3 & \alpha_2 & \beta_3 \end{pmatrix}, \qquad A(Y) = \begin{pmatrix} 0 & -\alpha_1 & \alpha_3 \\ \alpha_1 & 0 & -\alpha_2 \\ -\alpha_3 & \alpha_2 & 0 \end{pmatrix}, \tag{28}$$

where the new variables α_j, β_j are

$$\alpha_j = \frac{1}{2}\,\mathrm{e}^{-(q_{j+1}-q_j)/2}, \qquad \beta_j = \frac{1}{2}p_j \qquad\qquad j = 1,2,3$$

($q_4 \equiv q_1$). As mentioned before this system is an example of a quasilinear isospectral flow, and thus the special methods introduced in section 3.2 can be applied directly.

Our purpose here, rather than providing a complete characterization of the numerical integrators based on the Magnus expansion, is to show how they behave in practice with respect to efficiency and preservation of qualitative properties in comparison with other integration schemes. To this end we only consider the general

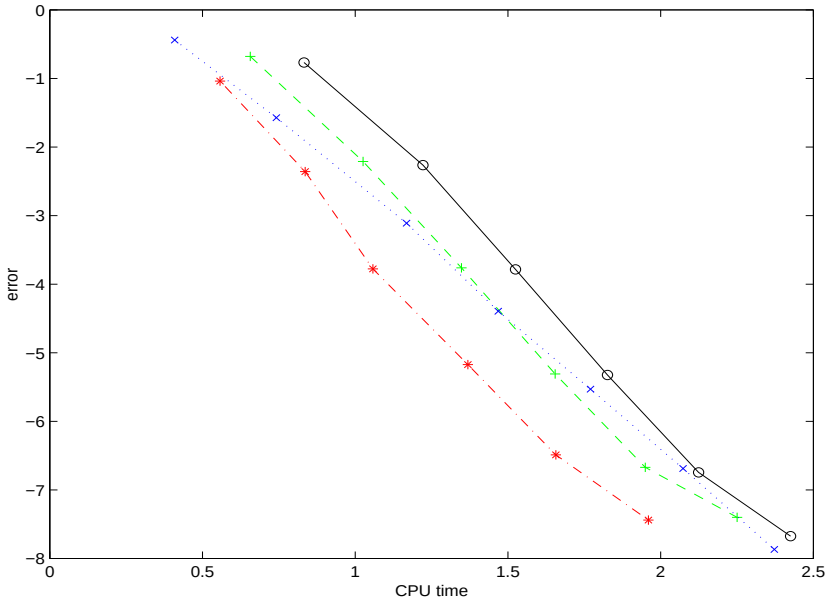

Figure 1. Error versus CPU time (in logarithmic scale) obtained with the fourth order integrators based on Magnus (solid line with circles), RKMK4 (dashed line with +), RK4 (dotted line with ×) and the special method (27) for quasilinear isospectral flows (dash-dotted line with ∗).

fourth-order scheme presented in the appendix (adapted to the isospectral case) and the algorithm (27) especially tailored to integrate quasilinear isospectral flows, and compare their performance with 'the' Runge–Kutta (RK4) method with the Butcher tableau

$$\begin{array}{c|cccc} 0 & & & & \\ \frac{1}{2} & \frac{1}{2} & & & \\ \frac{1}{2} & 0 & \frac{1}{2} & & \\ 1 & 0 & 0 & 1 & \\ \hline & \frac{1}{6} & \frac{1}{3} & \frac{1}{3} & \frac{1}{6} \end{array} \tag{29}$$

and the Runge–Kutta–Munthe-Kaas integrator (RKMK4) also based on the tableau (29).

We carry out a numerical integration in the interval $t \in [0, 3000]$ for several (constant) values of the step size h with initial condition $q_0 = (0, 0, 0)$, $p_0 = (1, 1, 0)$. 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. Then this error is represented as a function of the computational effort measured in terms of the CPU time used by the algorithms. The computation is done in Matlab and the command `expm` is employed to evaluate matrix exponentials (although also the Rodrigues formula can be used to get an explicit expression for $\exp u$, $u \in \mathfrak{so}(3)$ [22, p. 261]). The corresponding efficiency curves are plotted in Figure 1.

Observe that, in agreement with the theoretical estimates of Tables 1 and 2, RKMK4 is more efficient than the algorithm (A.1) based on the general Magnus

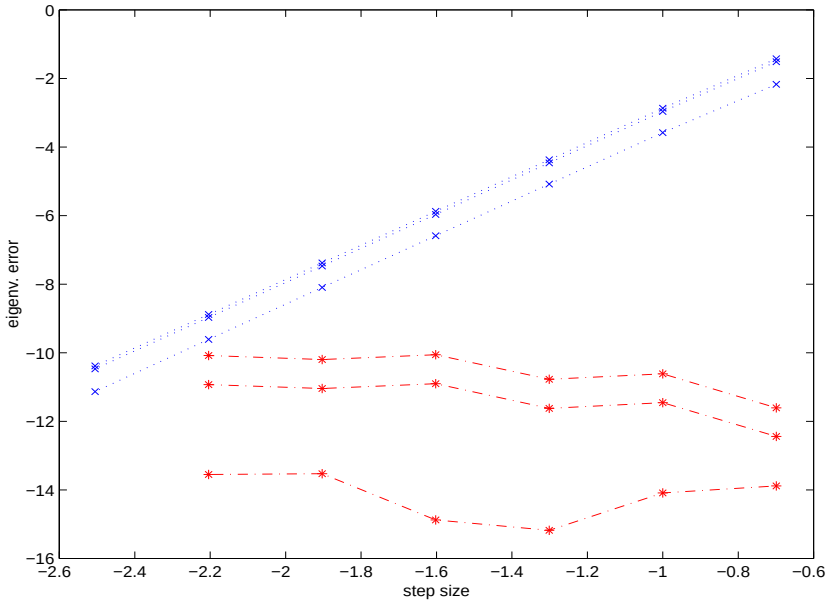

Figure 2. Difference between the approximate and exact eigenvalues obtained with RK4 (dotted line with $\times$) and the special method (27) (dash-dotted line with $*$). Both the step sizes and the error are represented in a logarithmic scale.

expansion, whereas the especially adapted scheme (27) outperforms both of them. It is also more efficient than RK4, even though the computational cost per iteration of the latter is lower (it does not require to evaluate matrix exponentials).

We remark once again that the numerical integrators based on the Magnus expansion can be implemented in a very natural way with variable step sizes, and thus their efficiency could be further improved. Additionally, they preserve the isospectral character of the flow (as does RKMK4), contrarily to RK4, as shown in Figure 2 in a logarithmic scale. There we plot the difference between the eigenvalues of the exact solution $Y(t)$, $\lambda_1 = \frac{1+\sqrt{3}}{2}$, $\lambda_2 = 0$, $\lambda_3 = \frac{1-\sqrt{3}}{2}$ and the eigenvalues of the approximations obtained by (27) and RK4 at $t = 3000$ when the integration is carried out with the step sizes considered in Figure 1.

4. Application to highly-oscillatory nonlinear ODEs

Suppose now that we are given the ODE system

$$\mathbf{y}' = A(t, \mathbf{y})\mathbf{y}, \qquad \mathbf{y}(0) = \mathbf{y}_0 \in \mathcal{M}, \tag{30}$$

where $\mathcal{M} \subset \mathbb{R}^d$ is a homogeneous space acted upon by the Lie group $\mathcal{G}$ and $A : \mathbb{R}_+ \times \mathcal{M} \longrightarrow \mathfrak{g}$. 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 (30) oscillates rapidly. In the linear case a conveniently chosen transformation previous to the application of the Magnus

expansion allows to get very accurate results [16]. 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

$$\mathbf{y}(t_n + x) = \mathrm{e}^{xA(t_n,\mathbf{y}_n)}\, \mathbf{z}(x). \tag{31}$$

Then

$$\frac{d\mathbf{z}}{dx} = B(x, \mathbf{z}(x))\mathbf{z}, \qquad \mathbf{z}(0) = \mathbf{y}_n \tag{32}$$

with

$$B(x, \mathbf{z}(x)) = F^{-1}(x)\left[A(t_n + x, F(x)\mathbf{z}(x)) - A(t_n, \mathbf{y}_n)\right] F(x) \tag{33}$$

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 (32) 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 [16]. But in the highly-oscillatory case other specially tailored quadratures exist which provide excellent results [19].

To illustrate the main features of the nonlinear modified Magnus expansion applied to the highly-oscillatory system (30), let us consider equations of the form

$$y'' + a(t, y, y')y = 0, \qquad y(0) = y_0, \quad y'(0) = y_0', \tag{34}$$

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 [28].

When (34) is written in a matrix form, we obtain (30) with

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

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

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

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

$$B(x, \mathbf{z}(x)) = \frac{1}{4}(\theta^2(x) - \theta_n^2)\left[2M_1 + M_2\mathrm{e}^{2i\theta_n x} + M_3\mathrm{e}^{-2i\theta_n x}\right] \tag{35}$$

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

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

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

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

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 $\theta_1^2(\tau) - \theta_n^2$ 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

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

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

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

with

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

Consequently, putting all the pieces together,

$$\Omega^{[1]}(x) = \varphi_1(x)\left(xM_1 + b_2^+(\theta_n)M_2 + b_2^-(\theta_n)M_3\right) \tag{36}$$

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 $\Omega^{[2]}(x) = \int_0^x B(\tau, \mathrm{e}^{\Omega^{[1]}(\tau)}\mathbf{y}_n)d\tau$ one gets the same expression (36), but now $\varphi_1(x)$ has to be replaced by

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

Similar considerations apply to higher order terms, although the analysis is obviously more involved. If the truncated Magnus solution of (32) is $\mathbf{z}(x) = \exp(\Omega^{[k]}(x))\mathbf{y}_n$, the approximation obtained in this way has the form

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

5. Conclusions

The nonlinear Magnus expansion we have proposed 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 four 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 but on the other hand they can be easily implemented with variable step sizes. Also, for particular types of problems it is possible to adapt appropriately the procedure and build up very efficient methods even with a fixed step size implementation.

In summary, this nonlinear Magnus expansion constitutes a very flexible tool to analyse nonlinear equations defined on Lie groups and/or homogeneous spaces acted upon by Lie groups. It allows 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.

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 A.

In this appendix we present a 4th-order algorithm for the numerical integration of the general Lie equation $Y' = A(t, Y)$ based on the nonlinear Magnus expansion (5) for $t_{n+1} = t_n + h$:

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

$$
\begin{aligned}
u_6 &= Q_1 + Q_2 + \frac{2}{3}Q_3 + \frac{1}{6}Q_4 - \frac{1}{6}[Q_1, Q_2] \\
k_6 &= hA(t_n + h, \mathrm{e}^{u_6} Y_0); \qquad Q_6 = k_6 - 2k_2 + k_1 \\
v &= Q_1 + Q_2 + \frac{2}{3}Q_5 + \frac{1}{6}Q_6 - \frac{1}{6}[Q_1, Q_2 - Q_3 + Q_5 + \frac{1}{2}Q_6] \\
Y_{n+1} &= \mathrm{e}^{v} Y_n
\end{aligned}
$$

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 [14, page 233].

(iii) This algorithm can also be directly applied to the isospectral flow $Y' = [A(t, Y), Y]$ with the replacement of $\mathrm{e}^{u_i} Y_n$ by the action $\mathrm{e}^{u_i} Y_n \mathrm{e}^{-u_i}$ in the computation of k_i and finally $Y_{n+1} = \mathrm{e}^{v} Y_n \mathrm{e}^{-v}$.

References

[1] S. Blanes, F. Casas, J.A. Oteo and J. Ros, Magnus and Fer expansions for matrix differential equations: the convergence problem, *J. Phys. A: Math. Gen.* **31** (1998), pp. 259-268.

[2] S. Blanes, F. Casas and J. Ros, Improved high order integrators based on the Magnus expansion, *BIT* **40** (2000), pp. 434-450.

[3] S. Blanes, F. Casas and J. Ros, High order optimized geometric integrators for linear differential equations, *BIT* **42** (2002), pp. 262-284.

[4] S. Blanes and P.C. Moan, Splitting metehods for non-autonomous Hamiltonian equations, *J. Comput. Phys.* **170** (2001), pp. 205-230.

[5] A.M. Bloch and A. Iserles, Commutators of skew-symmetric matrices, *Intnl. J. Bifurcations & Chaos* **15** (2005), 793–801.

[6] A.M. Bloch and A. Iserles, On an isospectral Lie–Poisson system and its Lie algebra, University of Cambridge Tech. Rep. DAMTP NA2005/01 (2005).

[7] A.M. Bloch, A. Iserles, J.E. Marsden and T.S. Ratiu, A class of integrable geodesic flows on the symplectic group, in preparation.

[8] R.W. Brockett, Dynamical systems that sort lists, diagonalize matrices and solve linear programming problems, *Linear Algebra Appl.* **146** (1991), pp. 79-91.

[9] M.P. Calvo, A. Iserles and A. Zanna, Numerical solution of isospectral flows, *Math. Comput.* **66** (1997), pp. 1461-1486.

[10] F. Casas, Numerical integration methods for the double-bracket flow, *J. Comput. Appl. Math.* **166** (2004), pp. 477-495.

[11] M.T. Chu and K.R. Driessel, The projected gradient method for least squares matrix approximations with spectral constraints, *SIAM J. Numer. Anal.* **27** (1990), pp. 1050-1060.

[12] J.R. Dormand, *Numerical Methods for Differential Equations: A Computational Approach*, CRC Press, Boca Raton (1996).

[13] E. Hairer, C. Lubich, and G. Wanner, *Geometric Numerical Integration. Structure-Preserving Algorithms for Ordinary Differential Equations*, Springer Ser. Comput. Math. 31, Springer-Verlag, Berlin (2002).

[14] E. Hairer, S.P. Nørsett and G. Wanner, *Solving Ordinary Differential Equations I* 2nd Ed., Springer Ser. Comput. Math. 8, Springer-Verlag, Berlin (1993).

[15] A. Iserles, On the discretization of double-bracket flows, *Found. Comput. Math.* **2** (2002), pp. 305-329.

[16] A. Iserles, On the global error of discretization methods for highly-oscillatory ordinary differential equations, *BIT* **42** (2002), pp. 561-599.

[17] A. Iserles, H.Z. Munthe-Kaas, S.P. Nørsett and A. Zanna, Lie group methods, *Acta Numerica* **9** (2000), pp. 215-365.

[18] A. Iserles and S.P. Nørsett, On the solution of linear differential equations in Lie groups, *Philos. Trans. Royal Soc. London Ser. A* **357** (1999), pp. 983-1019.

[19] A. Iserles and S.P. Nørsett, On quadrature methods for highly oscillatory integrals and their implementation, *BIT* **44** (2004), pp. 755-772.

[20] S. Klarsfeld and J.A. Oteo, Recursive generation of higher-order terms in the Magnus expansion, *Phys. Rev. A* **39** (1989), pp. 3270-3273.

[21] W. Magnus, On the exponential solution of differential equations for a linear operator, *Commun. Pure Appl. Math.* **7** (1954), pp. 649-673.

[22] J.E. Marsden and T.S. Ratiu, *Introduction to Mechanics and Symmetry*, Springer-Verlag, New York (1994).

[23] J.B. Moore, R.E. Mahony and U. Helmke, Numerical gradient algorithms for eigenvalue and singular value calculations, *SIAM J. Matrix Anal. Appl.* **15** (1994), pp. 881-902.

[24] H. Munthe-Kaas, High order Runge–Kutta methods on manifolds, *Appl. Numer. Math.* **29** (1999), pp. 115-127.

[25] H. Munthe-Kaas and B. Owren, Computations in a free Lie algebra, *Philos. Trans. Royal Soc. London Ser. A* **357** (1999), pp. 957-981.

[26] H. Munthe-Kaas and A. Zanna, Numerical integration of differential equations on homogeneous manifolds. In F. Cucker and M. Shub, editors, *Foundations of Computational Mathematics*, Springer-Verlag, Berlin, 1997, pp. 305-315.

[27] A. Zanna, Collocation and relaxed collocation for the Fer and the Magnus expansion, *SIAM J. Numer. Anal.* **36** (1999), pp. 1145-1182.

[28] D. Zwillinger, *Handbook of Differential Equations* 2nd Ed., Academic Press, Boston (1992).