Journal of Computational and Applied Mathematics 308 (2016) 263–275

Contents lists available at ScienceDirect

Journal of Computational and Applied Mathematics

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

On solving an isospectral flow

Amandeep Kaur

Department of Applied Mathematics and Theoretical Physics, Centre for Mathematical Sciences, University of Cambridge, Wilberforce Road, Cambridge CB3 0WA, England, United Kingdom

ARTICLE INFO

Article history:
Received 26 May 2015
Received in revised form 12 May 2016

Keywords:
Isospectral flow
Eigenvalues
Magnus expansion
Lie algebra
Binary trees

ABSTRACT

In this paper we expand the solution of the matrix ordinary differential system, originally due to Bloch and Iserles, of the form $X' = [N, X^2]$, $t \geq 0$, $X(0) = X_0 \in \mathrm{Sym}(n)$, $N \in \mathfrak{so}(n)$, where $\mathrm{Sym}(n)$ denotes the space of real $n \times n$ symmetric matrices and $\mathfrak{so}(n)$ denotes the Lie algebra of real $n \times n$ skew-symmetric matrices. The flow is solved using explicit Magnus expansion, which respects the isospectrality of the system. We represent the terms of expansion as binary rooted trees and deduce an explicit formalism to construct the trees recursively.

1. Introduction

Isospectral flows are matrix systems of ordinary differential equations of the form

$$X' = [B(X), X], \quad t \geq 0, \qquad X(0) = X_0 \in \mathrm{Sym}(n), \tag{1}$$

where $B(X) : \mathrm{Sym}(n) \to \mathfrak{so}(n)$. Their main structural feature is that they preserve the eigenvalues of the solution matrix. Isospectral flows occur in many important applications. First and the best known example is the Toda lattice, a one-dimensional lattice of particles whose motion is described by a nearest-neighbour interaction of an exponential type. It can be used to model a wide range of particle systems, ranging from the hard-sphere limit to the atomic case [1,2]. Another important example is the QR flow. The QR method for finding the eigenvalues of a matrix can be executed as an isospectral flow at unit intervals. QR flow is the generalization of non-periodic Toda flow. Such flows were first investigated by Symes [3] and subsequently in [4–9] and elsewhere.

Other well known examples include *eigenvalue problems* and *inverse eigenvalue problems for symmetric Toeplitz matrices* [10].

Note that if we let $B(X) = [N, X]$, in (1) where $N \in \mathrm{Sym}(n)$ then it leads to the *double-bracket flows*. Double bracket flows are isospectral flows given by the equations

$$X' = [[N, X], X], \quad t \geq 0, \qquad X(0) = X_0 \in \mathrm{Sym}(n) \tag{2}$$

where $N \in \mathrm{Sym}(n)$. They were introduced by Brockett [11] and Chu and Driessel [12]. Double bracket flows were discretized and then solved by Iserles by the method of Magnus series [13] and were generalized for more parameters [14]. Also, methods based on Magnus expansion are proposed for the numerical integration of the double-bracket flow and a bound on the convergence domain is provided by F. Casas [15].

In this paper we are concerned with the discretization of the matrix differential equation

$$X' = [N, X^2], \quad t \geq 0, \qquad X(0) = X_0 \in \mathrm{Sym}(n), \quad N \in \mathfrak{so}(n). \tag{3}$$

E-mail address: A.kaur@damtp.cam.ac.uk.

http://dx.doi.org/10.1016/j.cam.2016.05.033

The system (3) is known as the Bloch–Iserles (BI) equations. It is isospectral (preserves the eigenvalues of $X(t)$), is endowed with a Poisson structure and is integrable as proved by Bloch and Iserles [16]. We discretize this system using a similar approach, for instance in [13] and [15]. However, solving BI is much more complicated since it contains X^2 in the expression.

The above system is of interest for a number of reasons. Firstly, we can easily verify that it can be written in the form

$$X' = [N, X]X + X[N, X], \quad t \geq 0, \qquad X(0) = X_0 \in \mathrm{Sym}(n). \tag{4}$$

For $N \in \mathfrak{so}(n)$ and $X \in \mathrm{Sym}(n)$, we have $[N, X] \in \mathrm{Sym}(n)$, therefore it is a special case of a *congruent flow*

$$X' = A(X)X + XA^T(X), \quad t \geq 0, \qquad X(0) = X_0 \in \mathrm{Sym}(n), \tag{5}$$

where $A : \mathrm{Sym}(n) \to M(n)$, where $M(n)$ is the set of real $n \times n$ matrices, is sufficiently smooth. It is easy to verify that $X(t) = V(t)X_0V^T(t)$, where $V' = A(VX_0V^T)V,\ V(0) = I$. That means the solution is an outcome of the *general linear group* $\mathrm{GL}(n)$ acting on $\mathrm{Sym}(n)$ by congruence. That proves that the *signature* of $X(t)$ is constant [17]. Another interesting aspect of the given set of equations is that they are dual to the *generalized rigid body equations*

$$M' = [\Omega, M], \quad t \geq 0, \qquad M(0) \in \mathfrak{so}(n),$$

where $M = \Omega J + J\Omega,\ J \in \mathrm{Sym}(n)$ therefore $\Omega \in \mathfrak{so}(n)$ [18].

Also, it is clear that (3) can be rewritten in the form

$$X' = [XN + NX, X], \quad t \geq 0, \qquad X(0) = X_0 \in \mathrm{Sym}(n), \quad N \in \mathfrak{so}(n).$$

Since $XN + NX \in \mathfrak{so}(n)$ for $X \in \mathrm{Sym}(n),\ N \in \mathfrak{so}(n)$, it follows that the system (3) is indeed isospectral.

It is obvious that we can discretize isospectral flows by traditional numerical methods (e.g. Runge–Kutta and multistep), but, once $n \geq 3$, these methods cannot respect the isospectrality of the system, i.e. the numerical solution changes the eigenvalues [19]. Isospectrality is essential for applications ranging from classical mechanics, like Toda flows and N-body systems, to linear algebra, like QR flows and inverse eigenvalue problems, so we need to solve (3) by a method that respects it [20,19,21].

In this paper we solve the given isospectral flow using the method of Magnus series. We show that the solution of (3) can be represented in the form $X(t) = e^{\Omega(t)}X_0e^{-\Omega(t)}$, where instead of computing X at the first place, we obtain the Taylor expansion of Ω. Note that this ensures automatically that the numerical solution, being similar to X_0, is isospectral. We will see that the Taylor expansion of Ω can be formed algorithmically from X_0 and N and linear combinations of their commutators and anti-commutators. The goal of this paper is to determine the rules for finding the terms of Ω to an arbitrary accuracy. For the solution, first we convert the isospectral flow to a Lie-group flow and then translate it into a Lie-algebraic equation. This method preserves the isospectrality and gives the desired structure of the solution with large time steps. In Section 2 we solve the given system of differential equations using the Magnus expansion to obtain the Taylor expansion of Ω. Finally, in Section 3 the terms are represented by binary rooted trees and an algorithm is formed to construct the next tree by recursion and to calculate the coefficient of each tree. This lays the foundations to a more general setting, namely the explicit representation of the solution of (3) when $B(X)$ can be represented in a finite "alphabet". The representation as binary trees is very important because otherwise, as the number of terms in each iteration grows exponentially, the complexity of manual computation becomes prohibitive. By indexing the terms in the expansion with a subset of binary trees, it is convenient to derive explicit recurrence relations. Also, it is remarkable that the skew-symmetry and Jacobi identity obeyed by the commutator help us to reduce the number of terms by cancelling or writing certain terms as the linear combination of other terms.

2. An expansion of the solution

As stated above, the Bloch–Iserles system can be rewritten in the form

$$X' = [B(X), X], \quad t \geq 0, \qquad X(0) = X_0 \in \mathrm{Sym}(n)$$

with $B(X) = NX + XN$, where $B(X) : \mathrm{Sym}(n) \to \mathfrak{so}(n)$. This system is seen to be isospectral and it is standard to verify that

$$X(t) = Q(t)X_0Q^T(t), \quad t \geq 0, \tag{6}$$

where $Q(t) \in \mathrm{SO}(n)$ is the solution of

$$Q'(t) = (Q(t)X_0Q^T(t)N + NQ(t)X_0Q^T(t))Q(t), \qquad Q(0) = I. \tag{7}$$

In a similar way as Magnus [22] did for linear equations, our idea is to represent the solution of (7) in the form

$$Q(t) = e^{\Omega(t)},$$

where

$$\Omega' = \sum_{0}^{\infty} \frac{\mathrm{B}_r}{r!} \mathrm{ad}^r_{\Omega}(e^{\Omega}X_0e^{-\Omega}N + Ne^{\Omega}X_0e^{-\Omega}), \qquad \Omega(0) = 0. \tag{8}$$

Here $\mathrm{B}_m,\ m \in Z$ are Bernoulli numbers and ad^r_Ω is an iterated commutator defined by

$$\mathrm{ad}^0_\Omega A = A, \qquad \mathrm{ad}^1_\Omega A = [\Omega, A], \qquad \mathrm{ad}^2_\Omega A = [\Omega, [\Omega, A]], \dots, \qquad \mathrm{ad}^m_\Omega A = [\Omega, \mathrm{ad}^{m-1}_\Omega A],$$

where $[\Omega, A] = \Omega A - A\Omega$.

Now, taking $\Omega(t) = \sum_{m=0}^{\infty} \Omega_m t^m$ gives

$$\Omega'(t) = \sum_{m=0}^{\infty} (m+1)\Omega_{m+1} t^m \tag{9}$$

and this implies

$$\sum_{m=0}^{\infty} (m+1)\Omega_{m+1} t^m = \sum_{r=0}^{\infty} \frac{\mathrm{B}_r}{r!} \mathrm{ad}^r_{\Omega(t)} (e^{\Omega(t)} X_0 e^{-\Omega(t)} N + N e^{\Omega(t)} X_0 e^{-\Omega(t)}). \tag{10}$$

Comparing coefficients of $t^0, t^1, t^2 \dots$ we get the values of $\Omega_1, \Omega_2, \Omega_3 \dots$ as follows

$$\Omega_1 = NX_0 + X_0 N,$$
$$\Omega_2 = \frac{1}{2}(N[\Omega_1, X_0] + [\Omega_1, X_0]N),$$
$$\Omega_3 = \frac{1}{3}(N[\Omega_2, X_0] + [\Omega_2, X_0]N) + \frac{1}{6}(N[\Omega_1, [\Omega_1, X_0]] + [\Omega_1, [\Omega_1, X_0]]N) + \frac{1}{6}[\Omega_2, \Omega_1].$$

We denote $NX + XN = \{X\}$, thereby rewriting the above values of $\Omega_1, \Omega_2, \Omega_3 \dots$ in a more succinct manner as

$$\Omega_1 = \{X_0\}, \tag{11}$$
$$\Omega_2 = \frac{1}{2}\{[\Omega_1, X_0]\},$$
$$\Omega_3 = \frac{1}{3}\{[\Omega_2, X_0]\} + \frac{1}{6}\{[\Omega_1, [\Omega_1, X_0]]\} + \frac{1}{6}[\Omega_2, \Omega_1], \text{ etc.}$$

Note that $X \in \mathrm{Sym}(n),\ N \in \mathfrak{so}(n)$ implies that $\{X\} \in \mathfrak{so}(n)$. Introducing the curly bracket, i.e. featuring N implicitly in $NX + XN = \{X\}$ is important mainly for two reasons. Firstly, this helps to understand the recurrence of the terms in the expansion, else $XN + NX$ gets convoluted with other terms (either by getting cancelled or by adding up). Later in Section 3, while defining the structure of binary rooted trees; it prevents from getting bicoloured leaves, for instance, bicoloured leaves in [13].

Thus

$$\Omega(t) = t\{X_0\} + \frac{1}{2}t^2\{[\{X_0\}, X_0]\}$$
$$+ t^3\left(\frac{1}{6}\{[[\{X_0\}, X_0]], X_0]\} + \frac{1}{6}\{[\{X_0\}, [\{X_0\}, X_0]]\} + \frac{1}{12}[\{[\{X_0\}, X_0]\}, \{X_0\}]\right) + \cdots. \tag{12}$$

In next term there are three nested commutators. These are getting increasingly complex with each iteration and it is clear that the number of such terms is growing exponentially. In this paper we will represent these terms in terms of rooted trees, similar to the case of double-bracket flows in [13]. This builds upon an idea of Iserles and Nørsett [23] to use binary rooted trees as a shorthand for expansion terms.

Before we follow the procedure to simplify the above expansion, preliminary error graph of this method and error graph for eigenvalues of this method, as compared to the MATLAB ode45 solver with built-in parameters, are presented in Figs. 1 and 2 respectively.

In Fig. 1 we display error in the solution of (3) in the interval [0, 1] for a range of different step sizes Δt. The error plot is generated by truncating the expansion up to order three and is compared against the theoretically expected error of $O((\Delta t)^3)$. The experiments were performed on random 25×25 matrices. Note that in Fig. 1 we are interested in the case when $\Delta t \to 0$ (asymptotic limit), which corresponds to the left part of the figure. In Fig. 2 we calculate absolute error of eigenvalues of the two methods on a logarithmic scale. It is clearly seen that our method preserves the correct eigenvalues to machine accuracy (we note here that the machine epsilon 10^{-16} corresponds to relative error; however, we are calculating the absolute error in the graph). Despite a large time-step in the Magnus method, the error in eigenvalues stays very close to machine precision while the solution obtained using ode45 quickly strays away in terms of eigenvalues as time increases. This favourable behaviour of the Magnus method is to be expected from the principles underlying our approach.

Also, we have computed numerically the solution of the system for random 3×3 matrices using Lie group method using Magnus expansion. In Fig. 3, the phase portraits $(X_{1,2}, X_{k,l})$ are displayed for $(k, l) = (1, 1), (1, 3), (2, 2), (2, 3), (3, 3)$, with random initial condition. It has been proved in [16] that the system is Lie–Poisson and in [24] and [25] that it is integrable. So, it is clear that the behaviour of the solution displays the regularity and the solution curve lies on invariant tori. This is an indication of integrable Lie–Poisson structure. Similar behaviour is obtained for other randomly calculated matrices as well.

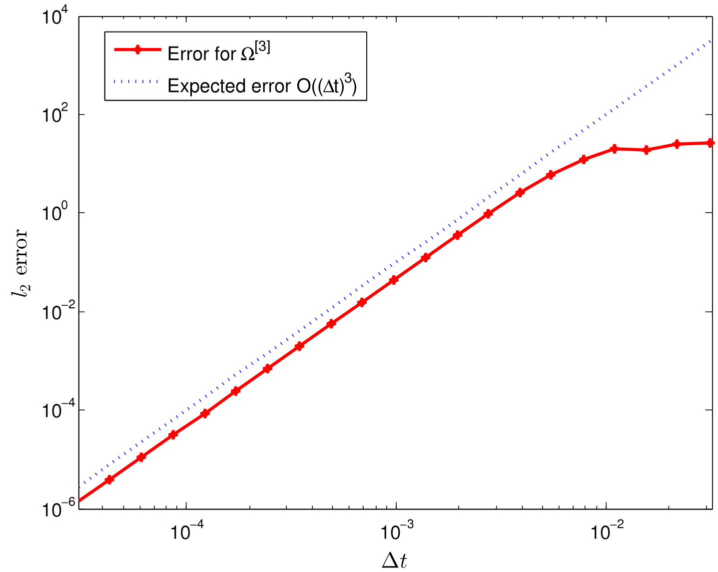


Fig. 1. Global error on logarithmic scale across an interval [0, 1] with different time steps, after truncating the Taylor expansion up to third order terms.

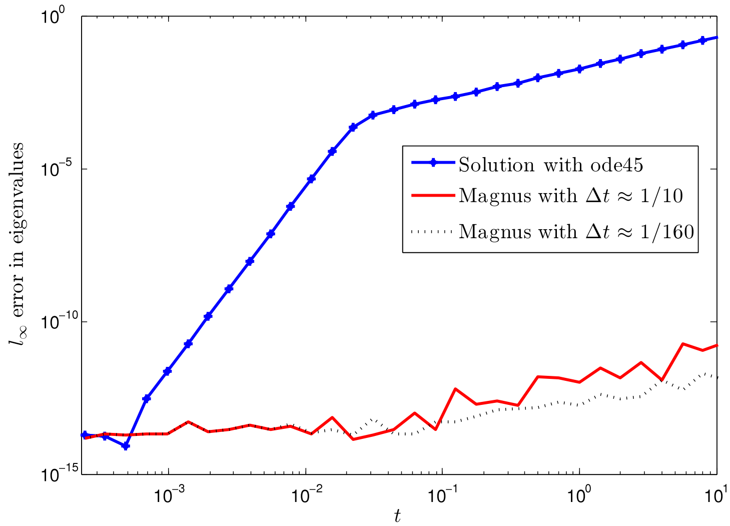


Fig. 2. Error plot showing absolute error of eigenvalues of the two methods on a logarithmic scale.

3. Representation by binary trees

Once we look at the expansion Ω, we note that each term is written in just two 'letters' X_0 and N, hence belongs to a free structure generated by them: The matrix N features implicitly in the bracket $\{Z\} = NZ + ZN$. Thus we can say that each term is made of commutators and curly brackets with some expression Z, which itself has been formed from X_0 and N. We attempt to find the expansion of the solution in Taylor series of the form

$$\Omega(t) = \sum_{r=1}^{\infty} \sum_{\tau \in \mathbb{T}_r} \alpha(\tau) H_\tau \tag{13}$$

where $\mathbb{T}_r$ is the set of all binary trees of power r: A tree τ is of power $r \geq 1$ if r is the greatest integer such that $H_\tau = O(t^r)$, H_τ is an expression constructed from X_0 and N according to rules implicit in the structure of the tree τ which will be explained next, and α is a scalar constant. Using rooted trees as a shorthand for expansion terms is an approach introduced by [23], that leads to a framework that elucidates the structure of individual terms and their relationship. While constructing trees, we commence by assigning X_0 to a single node, i.e. a *trivial tree*,

$$\bullet \rightsquigarrow X_0.$$

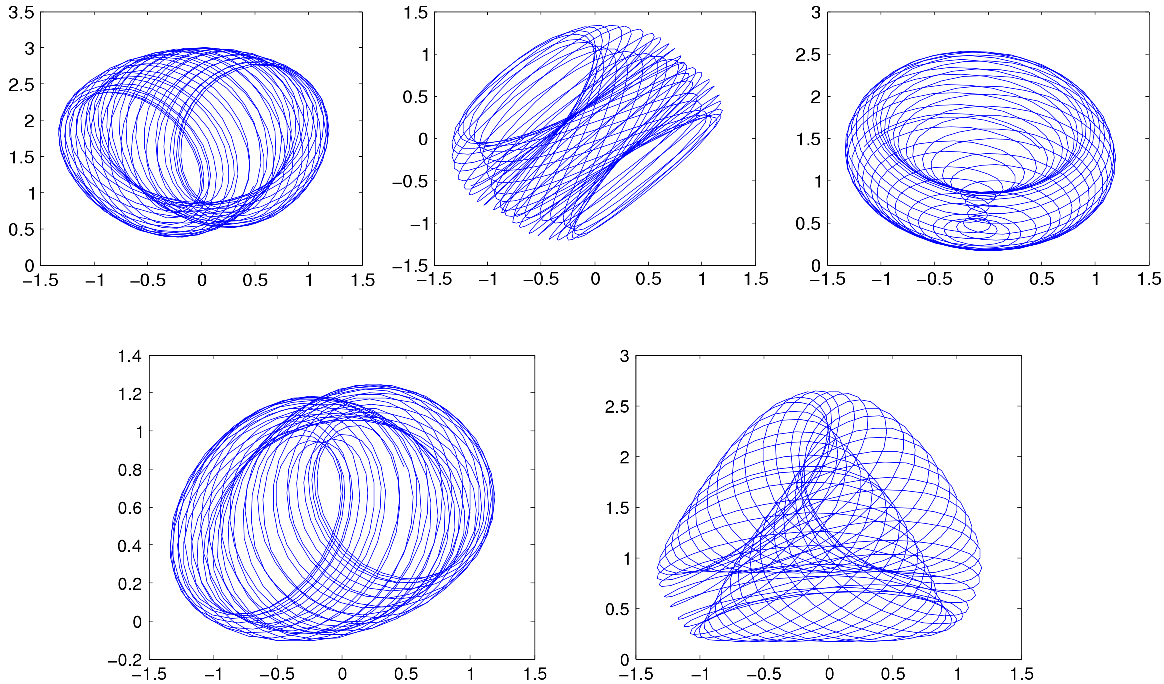
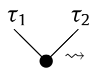


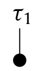
Fig. 3. The phase portraits $(X_{1,2}, X_{k,l})$ for $(k, l) = (1, 1), (1, 3), (2, 2), (2, 3), (3, 3)$, respectively, with a random initial condition. Here by $X_{k,l}$ we mean the kl^{th} element of the matrix X.

We define a function $\tau \rightarrow H_\tau$ from $\mathbb{T} = \bigcup_{r=1}^{\infty} \mathbb{T}_r$, a subset of binary rooted trees into $n \times n$ matrix functions by letting $H_\bullet = X_0$ and, by induction,



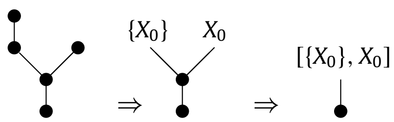
$\rightsquigarrow [H_{\tau_1}, H_{\tau_2}]$,

and



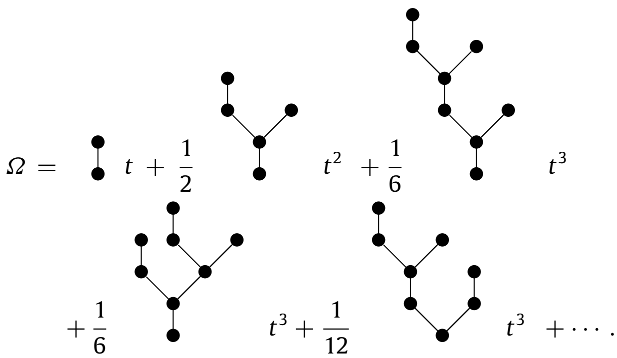
$\rightsquigarrow \{H_{\tau_1}\}$

where H_{τ_1} and H_{τ_2} are already constructed expansion terms. For example,



$\Rightarrow \{[\{X_0\}, X_0]\} \Rightarrow \{[\Omega_1, X_0]\}$.

In particular (12) can be written as



(14)

To explore the general rules underlying this correspondence between trees and expansion terms, let us have again a look at the equation

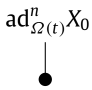
$$\sum_{m=0}^{\infty} (m+1)\Omega_{m+1} t^m = \sum_{r=0}^{\infty} \frac{\mathrm{B}_r}{r!} \mathrm{ad}^r_{\Omega(t)} \left(e^{\Omega(t)} X_0 e^{-\Omega(t)} N + N e^{\Omega(t)} X_0 e^{-\Omega(t)} \right).$$

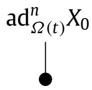
We know that

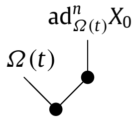
$$e^{\Omega}X_0e^{-\Omega} = \mathrm{Ad}_{\Omega}X_0 = e^{\mathrm{ad}_{\Omega}}X_0 = \sum_{n=0}^{\infty}\frac{1}{n!}\mathrm{ad}_{\Omega}^{n}X_0.$$

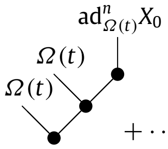
Thus

$$\sum_{m=0}^{\infty}(m+1)\Omega_{m+1}t^m = \sum_{r=0}^{\infty}\frac{\mathrm{B}_r}{r!}\mathrm{ad}_{\Omega(t)}^{r}\left(N\sum_{n=0}^{\infty}\frac{1}{n!}\mathrm{ad}_{\Omega(t)}^{n}X_0 + \sum_{n=0}^{\infty}\frac{1}{n!}\mathrm{ad}_{\Omega(t)}^{n}X_0N\right)$$

$$= \sum_{r=0}^{\infty}\frac{\mathrm{B}_r}{r!}\mathrm{ad}_{\Omega(t)}^{r}\sum_{n=0}^{\infty}\frac{1}{n!}$$ 

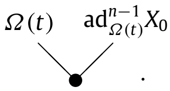
$$= \frac{\mathrm{B}_0}{0!}\sum_{n=0}^{\infty}\frac{1}{n!}$$ 

$$+ \frac{\mathrm{B}_1}{1!}\sum_{n=0}^{\infty}\frac{1}{n!}$$ 

$$+ \frac{\mathrm{B}_2}{2!}\sum_{n=0}^{\infty}\frac{1}{n!}$$ 

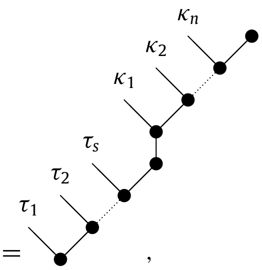
$$+\cdots. \tag{15}$$

where

$$\mathrm{ad}_{\Omega(t)}^{n}X_0 =$$ 

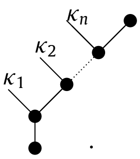
.

Clearly, in (15) we see that each tree here can be represented in the form

$$T_{s,n} \ni \tau =$$ 

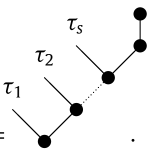
, (16)

where $s \in \{0, 1, 2, 3, \ldots\}$, $n \in \{0, 1, 2, 3, \ldots\}$. Here, the trees $\tau_1, \tau_2, \ldots \tau_s, \kappa_1, \kappa_2, \ldots \kappa_n$ have been featured earlier in the expansion, where $\tau_i \in \mathbb{T}_{p_i}$, and $\kappa_j \in \mathbb{T}_{q_j}$, $i = 1, 2, \ldots, s$, $j = 1, 2, \ldots, n$ and $p_1 + p_2 + \cdots + p_s + q_1 + q_2 + \cdots + q_n + 1 = r$. For $s = 0$ (for example, in the first four trees in (14)), the structure of the trees becomes

$$T_{0,n} \ni \tau =$$ 

. (17)

For $n = 0$ (for example, the fifth tree), this becomes

$$T_{s,0} \ni \tau =$$ 

. (18)

It is also possible to deduce the explicit form of the constant α by substituting the value of Ω from (13) in the form of $\alpha(\tau)$ and H_τ in (15) and simplifying it for $\alpha(\tau)$. Set

$$\alpha\Big($$  $$\Big) = 1. \tag{19}$$

Let $\tau \in \mathbb{T}_r$, $r \in \mathbb{N}$ and suppose that $\alpha(\tau_i)$ and $\alpha(\kappa_j)$ are known for $i = 1, 2, \ldots, s, j = 1, 2, \ldots, n$. Then

$$\alpha(\tau) = \frac{1}{r}\frac{B_s}{s!}\frac{1}{n!}\prod_{i=1}^{s}\alpha(\tau_i)\prod_{j=i}^{n}\alpha(\kappa_j), \quad s, n \in \mathbb{N}, \tag{20}$$

where B_s is the sth Bernoulli number.

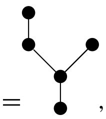
Now we have the general pattern for recursion. Suppose that $\mathbb{T}_1, \mathbb{T}_2, \ldots, \mathbb{T}_{r-1}$ are known and also the coefficient α in these sets is known. To construct $\mathbb{T}_r$ we note that every τ is of the form (16) for some $s \in \{0, 1, 2, 3, \ldots, r-1\}$ and $n \in \{0, 1, 2, 3, \ldots, r-1\}$. For every such s and n we consider all the partitions $p_1+p_2+\cdots+p_s+q_1+q_2+\cdots+q_n+1 = r$. For every partition we construct the tree τ in (16) and use (20) to determine the coefficient α. The trees which correspond to zero terms would be eliminated. Moreover, some trees can be replaced by linear combinations of other trees. Let us start from $\mathbb{T}_1$.

1. $\mathbb{T}_1$: For $s = 0, n = 0$, we have

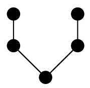
$\tau_1^1 =$ , $\quad \alpha(\tau_1^1) = \frac{1}{1}\cdot 1 \cdot \frac{1}{0!} = 1.$

2. $\mathbb{T}_2$:

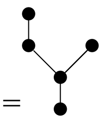
(a) For $s = 0, n = 1$, we can have only one possibility, $\kappa_1 =$ 

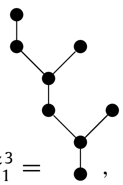
$\tau_1^2 =$ , $\quad \alpha(\tau_1^2) = \frac{1}{2}\cdot 1 \cdot \frac{1}{1!} = \frac{1}{2};$

(b) For $s = 1, n = 0 : \tau_1 =$ 

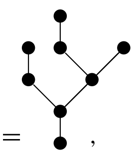
$\tau_2^2 =$ , vanishing tree, discard.

3. $\mathbb{T}_3$:

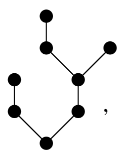
(a) $s = 0, n = 1$: $\kappa_1 =$ 

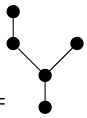
$\tilde{\tau}_1^3 =$ , $\quad \alpha(\tilde{\tau}_1^3) = \frac{1}{3}\cdot 1 \cdot \frac{1}{1!}\cdot\frac{1}{2} = \frac{1}{6};$

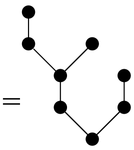
(b) $s = 0, n = 2$: $\kappa_1 = \kappa_2 =$ 

$\tilde{\tau}_2^3 =$ , $\quad \alpha(\tilde{\tau}_2^3) = \frac{1}{3}\cdot 1 \cdot \frac{1}{2!}\cdot 1 \cdot 1 = \frac{1}{6};$

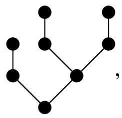
(c) $s = 1, n = 1$: $\kappa_1 = \tau_1 =$ 

$\tilde{\tau}_3^3 =$ , $\quad \alpha(\tilde{\tau}_3^3) = \frac{1}{3}\cdot\left(-\frac{1}{2}\right)\cdot\frac{1}{1!}\cdot 1 \cdot 1 = -\frac{1}{6};$

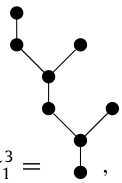
(d) $s = 1, n = 0$: $\tau_1 =$ 

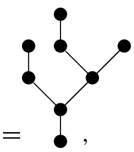
$\tilde{\tau}_4^3 =$  , $\alpha(\tilde{\tau}_4^3) = \frac{1}{3} \cdot \left(-\frac{1}{2}\right) \cdot \frac{1}{0!} \cdot 1 \cdot \frac{1}{2} = -\frac{1}{12}$;

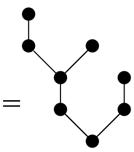
(e) $s = 2, n = 0$: $\tau_1 = \tau_2 =$ 

$\tilde{\tau}_5^3 =$  , vanishing tree, discard.

Before we proceed further, let us clean up the set $\mathbb{T}_3$. We clearly see that $\tilde{\tau}_3^3$ is nothing but $\tilde{\tau}_4^3$ with opposite sign. The two trees can be aggregated into $\tilde{\tau}_4^3$ say, with the coefficient replaced by $\alpha(\tilde{\tau}_4^3) - \alpha(\tilde{\tau}_3^3)$. After trivial rotations (corresponding to commutation) we obtain three trees in the set $\mathbb{T}_3$,

$\tau_1^3 =$  , $\alpha(\tau_1^3) = \frac{1}{6}$;

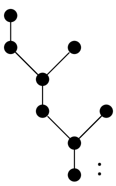
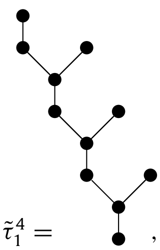
$\tau_2^3 =$  , $\alpha(\tau_2^3) = \frac{1}{6}$;

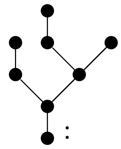
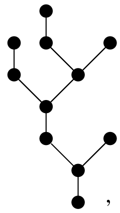
$\tau_3^3 =$  , $\alpha(\tau_3^3) = \frac{1}{12}$.

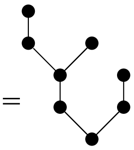
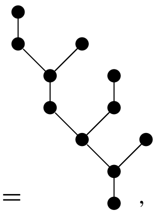
Using the tree formalism, we are now constructing the next "generation" of terms.

1. $\mathbb{T}_4$:

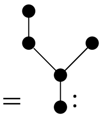
   (a) $s = 0, n = 1$:

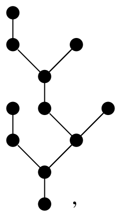
   i. $\kappa_1 =$  : $\tilde{\tau}_1^4 =$  , $\alpha(\tilde{\tau}_1^4) = \frac{1}{24}$;

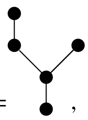
ii. $\kappa_1 =$  : $\tilde{\tau}_2^4 =$  , $\alpha(\tilde{\tau}_2^4) = \frac{1}{24}$;

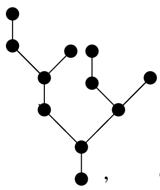
iii. $\kappa_1 =$  : $\tilde{\tau}_3^4 =$  , $\alpha(\tilde{\tau}_3^4) = \frac{1}{48}$;

(b) $s = 0, n = 2$:

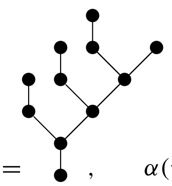
i. $\kappa_1 =$  $\kappa_2 =$  :

$\tilde{\tau}_4^4 =$  , $\alpha(\tilde{\tau}_4^4) = \dfrac{1}{16}$;

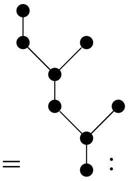
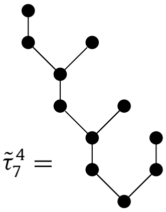
ii. $\kappa_1 =$  , $\kappa_2 =$  :

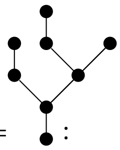
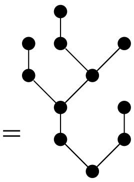
$\tilde{\tau}_5^4 =$  , $\alpha(\tilde{\tau}_5^4) = \dfrac{1}{16}$;

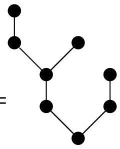
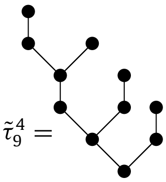
(c) $s = 0, n = 3$: $\kappa_1 = \kappa_2 = \kappa_3 =$ 

$\tilde{\tau}_6^4 =$  , $\alpha(\tilde{\tau}_6^4) = \dfrac{1}{24}$;

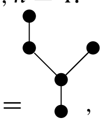
(d) $s = 1, n = 0$:

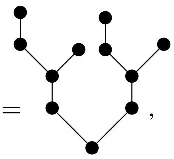
i. $\tau_1 =$  : $\tilde{\tau}_7^4 =$  , $\alpha(\tilde{\tau}_7^4) = -\frac{1}{48}$;

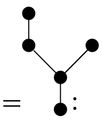
ii. $\tau_1 =$  : $\tilde{\tau}_8^4 =$  , $\alpha(\tilde{\tau}_8^4) = -\frac{1}{48}$;

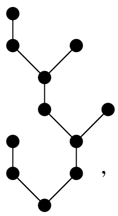
iii. $\tau_1 =$  : $\tilde{\tau}_9^4 =$  , $\alpha(\tilde{\tau}_9^4) = -\frac{1}{96}$;

(e) $s = 1, n = 1$:

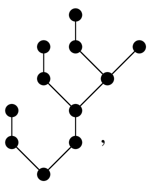
i. $\tau_1 =$  , $\kappa_1 =$  :

$\tilde{\tau}_{10}^4 =$  , vanishing tree, discard.

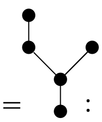
ii. $\tau_1 =$  , $\kappa_1 =$  :

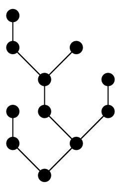
$\tilde{\tau}_{11}^4 =$  , $\alpha(\tilde{\tau}_{11}^4) = -\dfrac{1}{16}$;

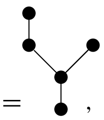
(f) $s = 1, n = 2$: $\kappa_1 = \kappa_2 = \tau_1 =$  :

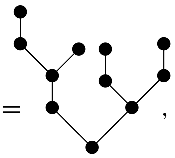
$\tilde{\tau}_{12}^4 =$  , $\alpha(\tilde{\tau}_{12}^4) = -\dfrac{1}{16}$;

(g) $s = 2, n = 0$:

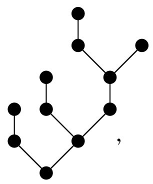
i. $\tau_1 =$  , $\tau_2 =$  :

$\tilde{\tau}_{13}^4 =$  , $\alpha(\tilde{\tau}_{13}^4) = \dfrac{1}{96}$;

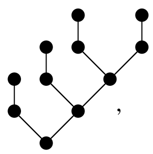
ii. $\tau_1 =$  , $\tau_2 =$ :

$\tilde{\tau}_{14}^4 =$  , vanishing tree, discard.

(h) $s = 2, n = 1$: $\kappa_1 = \tau_1 = \tau_2 =$ :

$\tilde{\tau}_{15}^4 =$  , $\alpha(\tilde{\tau}_{15}^4) = \frac{1}{48}$;

(i) $s = 3, n = 0$: $\tau_1 = \tau_2 = \tau_3 =$ :

$\tilde{\tau}_{16}^4 =$  , $\alpha(\tilde{\tau}_{16}^4) = 0$; vanishing tree, discard.

Three trees are vanishing here. Also, with trivial rotations, few trees get cancelled or merge into other trees obeying the anti symmetry, for example, $\tilde{\tau}_9^4$ is $-\tilde{\tau}_{13}^4$ which is also equal to $-(-\tilde{\tau}_{15}^4)$, similarly $\tilde{\tau}_7^4$ is nothing but $\tilde{\tau}_{11}^4$ with opposite sign and also $\tilde{\tau}_8^4 = -\tilde{\tau}_{12}^4$ and the trees $\tilde{\tau}_3^4$, $\tilde{\tau}_4^4$ and $\tilde{\tau}_5^4$ satisfy Jacobi identity. Tiding up $\mathbb{T}_4$ and translating every tree in terms of commutators and curly brackets, the Taylor expansion of $\Omega(t)$ becomes

$$\begin{aligned}
\Omega(t) = {} & t\{X_0\} \\
& + \frac{1}{2}t^2\{[\{X_0\}, X_0]\} \\
& + t^3\left(\frac{1}{6}\{[\{[\{X_0\}, X_0]\}, X_0]\} + \frac{1}{6}\{[\{X_0\}, [\{X_0\}, X_0]]\}\right. \\
& \left. + \frac{1}{12}[\{[\{X_0\}, X_0]\}, \{X_0\}]\right) \\
& + t^4\left(\frac{1}{24}\{[\{[\{[\{X_0\}, X_0]\}, X_0]\}, X_0]\}\right. \\
& + \frac{1}{24}\{[\{[\{X_0\}, [\{X_0\}, X_0]]\}, X_0]\} + \frac{1}{24}\{[\{X_0\}, [\{[\{X_0\}, X_0]\}, X_0]]\} \\
& + \frac{1}{12}\{[\{[\{X_0\}, X_0]\}, [\{X_0\}, X_0]]\} + \frac{1}{24}\{[\{X_0\}, [\{X_0\}, [\{X_0\}, X_0]]]\} \\
& \left. + \frac{1}{24}[\{[\{[\{X_0\}, X_0]\}, X_0]\}, \{X_0\}] + \frac{1}{24}[\{[\{X_0\}, [\{X_0\}, X_0]]\}, \{X_0\}]\right) \\
& + \cdots.
\end{aligned}$$

In explanation of the simplification exercise of the above terms: As we stated that the terms get simplified obeying the anti symmetry and Jacobi identity; by anti symmetry we mean $[A, B] = -[B, A]$, which is an easy exercise to verify for the reader, whereas A, B, C are said to satisfy the Jacobi identity if $[A, [B, C]] + [B, [C, A]] + [C, [A, B]] = 0$. As we translate from the trees,

$$\begin{aligned}
&\tilde{\tau}_3^4 \to \{[[\{[\{X_0\}, X_0]\}, \{X_0\}], X_0]\} \to \{[[\Omega_2, \Omega_1], X_0]\}, \\
&\tilde{\tau}_4^4 \to \{[\{X_0\}, [\{[\{X_0\}, X_0]\}, X_0]]\} \to \{[\Omega_1, [\Omega_2, X_0]]\}, \\
&\tilde{\tau}_5^4 \to \{[\{[\{X_0\}, X_0]\}, [\{X_0\}, X_0]]\} \to \{[\Omega_2, [\Omega_1, X_0]]\}
\end{aligned}$$

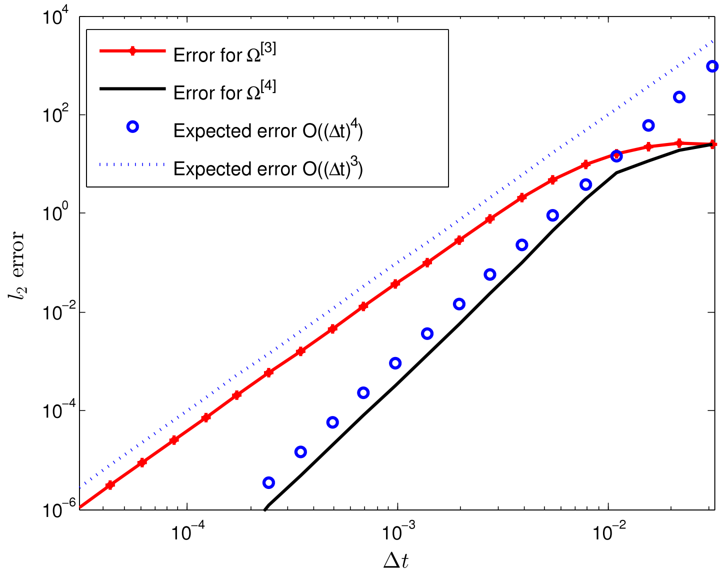
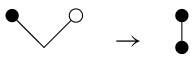


Fig. 4. Global error on logarithmic scale across an interval [0, 1] with different time steps, after truncating the Taylor expansion up to fourth order terms.

and hence use the Jacobi identity for simplification. Also, one may observe that by co-incidence the tree formation seems identical with [13] by replacing


.

It seems quite tempting to introduce such a correspondence between the brackets $\{X\}$ and $[., N]$ for the whole tree exercise. With this replacement we get correct terms up to third order but it gives different values when we reach next generation terms. The reason for this fact is that the difference between $\{X\}$ and $[., N]$ is not only the brackets but the different signs, $+$ and $-$, which appear implicitly in the expression and surely make difference as the complexity increases. Moreover, the number of independent fourth order terms in [13] is 8, whereas the number of independent terms in Ω_4 in BI system is 7. This explains that the fourth order terms obtained from the two approaches are different. Therefore, no such correspondence for the tree structure has been found between the two.

We see that once a tree formulation is obtained, translating it into commutators and curly brackets costs less labour as compared to the complexity of manual computation. After translating the trees into mathematical expressions, we plot the error graph. Truncating the expansion up to fourth order terms, the error graph of the solution as compared to the MATLAB ode45 solver with built-in parameters, is shown in Fig. 4. The error plot is generated by comparing against the theoretically expected error of $O((\Delta t)^3)$ and $O((\Delta t)^4)$. The experiments were performed on random 25×25 matrices. Clearly, this plot shows that the Magnus method is a fourth order method.

In Fig. 5, we generate the error graphs of $\Omega^{[3]}$ and $\Omega^{[4]}$ obtained by translating the trees and comparing against the error in explicit Magnus expansion in the appendix (see remark iii) of [26]. In the graph, global error on logarithmic scale across an interval [0, 1] with different time steps and the solutions are compared against MATLAB ode45 solver with built-in parameters. There is remarkable overlap in the accuracy between the two approaches. Also, we plot the error curve displaying the difference between the solution obtained by tree algorithm and explicit Magnus expansion in the appendix (see remark iii) of [26]. Again, note that in Fig. 5, we are interested in the case when $\Delta t \to 0$ (asymptotic limit), which corresponds to the left part of the figure. It is observed that no variability is seen in the solution of both methods. Similar dynamics has been captured by the solution with the two approaches, i.e. tree algorithm and explicit Magnus expansion in the appendix (see remark iii) of [26]. Although both methods are of the same accuracy, it is less expensive to use the tree algorithm. We see that the number of terms in each iteration grows exponentially and the complexity of manual computation becomes prohibitive. Using the tree algorithm we reduce the computational cost and deduce the explicit formalism of the solution.

One thus observes that the BI equations have a number of remarkable properties which motivate us to expand the equations. It has been seen in Fig. 3 that these equations have Lie–Poisson structure. Fig. 2 shows that the discretization of the BI equations using Magnus expansion preserves the eigenvalues of the solution matrix. Also, it is observed in Fig. 4 that the Lie group method using Magnus expansion is a fourth order method, here by fourth order we mean the truncation of $\Omega(t)$ up to the fourth power of t. By employing the shorthand of binary rooted trees for expansion terms, the computation is made affordable. This also lays a foundation to the explicit representation of the solution of the BI system.

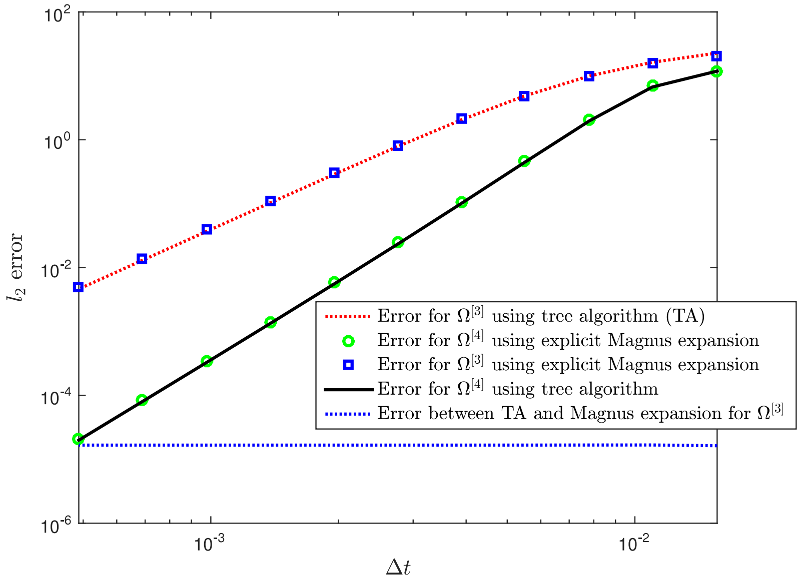


Fig. 5. Comparison error graphs: tree algorithm and explicit Magnus expansion.

Acknowledgements

Special thanks to my supervisor professor Arieh Iserles for his valuable suggestions. I am also grateful to SGPC-Cambridge Commonwealth Trust to support the research with the funds.

References

[1] M. Toda, Theory of Nonlinear Lattices, Springer-Verlag, Berlin, 1981.
[2] M. Toda, Nonlinear Waves and Solitons, in: Mathematics and its Applications (Japanese Series), 1989.
[3] W.W. Symes, The QR algorithm and scattering for the finite nonperiodic Toda lattice, Physica D 4 (1982) 275–280.
[4] T. Nanda, Isospectral flows on band matrices (Ph.D. thesis), New York University, 1982.
[5] P. Deift, T. Nanda, C. Tomei, Ordinary differential equations and the symmetric eigenvalue problem, SIAM J. Numer. Anal. 20 (1) (1983) 1–22. URL http://epubs.siam.org/doi/abs/10.1137/0720001.
[6] P. Deift, S. Rivera, C. Tomei, D. Watkins, A monotonicity property for Toda-type flows, SIAM J. Matrix Anal. Appl. 12 (3) (1991) 463–468. URL http://epubs.siam.org/doi/abs/10.1137/0612033.
[7] M. Chu, The generalized Toda flow, the QR algorithm and the center manifold theory, SIAM J. Algebr. Discrete Methods 5 (2) (1984) 187–201. URL http://epubs.siam.org/doi/abs/10.1137/0605020.
[8] M. Chu, K.R. Driessel, Can Real Symmetric Toeplitz Matrices have Arbitrary Real Spectra? Tech. Rep., Idaho State University, 1989.
[9] J. Lagarias, Monotonicity properties of the Toda flow, the QR-flow, and subspace iteration, SIAM J. Matrix Anal. Appl. 12 (3) (1991) 449–462. URL http://epubs.siam.org/doi/abs/10.1137/0612032.
[10] S. Friedland, J. Nocedal, M. Overton, The formulation and analysis of numerical methods for inverse eigenvalue problems, SIAM J. Numer. Anal. 24 (3) (1987) 634–667. URL http://epubs.siam.org/doi/abs/10.1137/0724043.
[11] R. Brockett, Dynamical systems that sort lists, diagonalize matrices, and solve linear programming problems, Linear Algebra Appl. 146 (1991) 79–91.
[12] M. Chu, K. Driessel, The projected gradient method for least squares matrix approximations with spectral constraints, SIAM J. Numer. Anal. 27 (4) (1990) 1050–1060.
[13] A. Iserles, On the discretization of double-bracket flows, Found. Comput. Math. 2 (3) (2002) 305–329.
[14] A.M. Bloch, A. Iserles, Aspects of generalized double-bracket flows, in: Proc. Centre de Recherche Montreal, American Mathematical Society, 2005, pp. 65–76.
[15] F. Casas, Numerical integration methods for the double-bracket flow, J. Comput. Appl. Math. 166 (2) (2004) 477–495. URL http://www.sciencedirect.com/science/article/pii/S0377042703007714.
[16] A. Bloch, A. Iserles, On an isospectral Lie–Poisson system and its Lie-algebra, Found. Comput. Math. (2006) 121–144.
[17] R. Horn, C. Johnson, Topics in Matrix Analysis, Cambridge University Press, Cambridge, 1991.
[18] S. Manakov, Note on the integration of Euler's equations of the dynamics of an n-dimensional rigid body, Funct. Anal. Appl. 10 (1976) 328–329.
[19] M. Calvo, A. Iserles, A. Zanna, Conservative methods for the Toda lattice equations, IMA J. Numer. Anal. 19 (4) (1999) 509–523.
[20] M. Calvo, A. Iserles, A. Zanna, Numerical solution of isospectral flows, Math. Comput. 66 (220) (1997) 1461–1486.
[21] A. Zanna, On the numerical solution of isospectral flows (Ph.D. thesis), University of Cambridge, 1998.
[22] W. Magnus, On the exponential solution of differential equations for a linear operator, Comm. Pure Appl. Math. 7 (1954) 649–673.
[23] A. Iserles, S. Nørsett, On the solution of linear differential equations in Lie groups, Phil. Trans. R. Soc. A 357 (1999) 983–1019.
[24] A.M. Bloch, V. Brînzanescu, A. Iserles, J.E. Marsden, T.S. Ratiu, A class of integrable flows on the space of symmetric matrices, Comm. Math. Phys. 290 (2009) 399–435.
[25] L.-C. Li, C. Tomei, C.: The complete integrability of a Lie—Poisson system proposed by Bloch and Iserles, Int. Math. Res. Not. IMRN (2006) 64949.
[26] F. Casas, A. Iserles, Explicit Magnus expansions for nonlinear equations, J. Phys. A: Math. Gen. 39 (19) (2006) 5445.