

## TIME-AVERAGE ON THE NUMERICAL INTEGRATION OF NONAUTONOMOUS DIFFERENTIAL EQUATIONS\*

SERGIO BLANES†

**Abstract.** In this work we show how to numerically integrate nonautonomous differential equations by solving alternate time-averaged differential equations. Given a quadrature rule of order  $2s$  or higher for  $s = 1, 2, \dots$ , we show how to build a differential equation with an averaged vector field that is a polynomial function of degree  $s - 1$  in the independent variable,  $t$ , and whose solution after one time step agrees with the solution of the original differential equation up to order  $2s$ . Then, any numerical scheme can be used to solve this alternate averaged equation where the vector field is always evaluated at the chosen quadrature rule. We show how to use the Magnus series expansion, adapted to nonlinear problems, to build the formal solution, and this result is valid for any choice of the quadrature rule. This formal solution can be used to build new schemes that must agree with it up to the desired order. For example, we show how to build commutator-free methods from previous results in the literature. All methods can also be written in terms of moment integrals, and each integral can be computed using different quadrature rules. This procedure allows us to build tailored methods for different classes of problems. We illustrate the time-averaged procedure and its efficiency in solving several problems.

**Key words.** nonautonomous differential equations, time-averaged differential equations, quadrature rules, geometric integration

**AMS subject classifications.** 65L05, 65J10

**DOI.** 10.1137/17M1156150

### 1. Introduction. Nonautonomous differential equations

$$(1.1) \quad x' = f(t, x), \quad x(t_0) = x_0 \in \mathbb{C}^d, \quad t \in [t_0, T],$$

where  $' \equiv \frac{d}{dt}$ , frequently appear in practice. The time-dependency of the vector field can be originated from many different sources: due to external interactions, when a preliminary change of coordinates is considered, when solving boundary value problems using iterative methods, etc. Direct application of numerical integrators to solve (1.1) usually corresponds to using the scheme on the autonomous equation obtained by appending the independent variable,  $t$ , to the dependent variables:

$$(1.2) \quad \begin{pmatrix} x \\ t \end{pmatrix}' = \begin{pmatrix} f(t, x) \\ 1 \end{pmatrix}.$$

However, in most cases the explicit time-dependency appearing on the right-hand side of (1.1) plays an important role for the evolution of the system, so to take  $t$  as a dependent variable playing the same role as any component of  $x$  is, in many cases, not the most appropriate procedure. For example, the linear equation

$$(1.3) \quad x' = A(t)x, \quad x(t_0) = x_0$$

\*Received by the editors November 9, 2017; accepted for publication (in revised form) May 30, 2018; published electronically August 7, 2018.

<http://www.siam.org/journals/sinum/56-4/M115615.html>

**Funding:** This work was funded by Ministerio de Economía, Industria y Competitividad (Spain) through project MTM2016-77660-P (AEI/FEDER, UE).

†Instituto de Matemática Multidisciplinar, Universitat Politècnica de València, 46022 Valencia, Spain (serblaza@imm.upv.es).

with  $A \in \mathbb{C}^{d \times d}$  turns into an autonomous nonlinear differential equation

$$(1.4) \quad \begin{pmatrix} x \\ t \end{pmatrix}' = \begin{pmatrix} A(t)x \\ 1 \end{pmatrix}.$$

Then, it is not obvious how to use some methods (e.g., exponential integrators) that have been shown to be highly efficient for solving linear systems.

Given  $c_i$ ,  $i = 1, \dots, m$ , the nodes of a quadrature rule of order  $p \geq 2s$  (for  $s = 1, 2, 3, \dots$ , and choosing  $m$  accordingly), and a given time mesh,  $t_0 < t_1 < t_2 < \dots < t_N = T$ , with  $h_n = t_n - t_{n-1}$ ,  $n = 1, 2, \dots, N$ , the time steps, in this work we show how to build a vector field

$$(1.5) \quad \tilde{f}^{(2s)}(t, x, h_n) = f_0^{(2s)}(x, h_n) + \dots + t^{s-1} f_{s-1}^{(2s)}(x, h_n),$$

where each  $f_j^{(2s)}(x, h_n)$  is a linear combination of  $f(t_{n-1} + c_i h_n, y)$ ,  $i = 1, \dots, m$  (that changes on each time interval  $[t_{n-1}, t_n]$ ) such that

$$(1.6) \quad \|f(t, x) - \tilde{f}^{(2s)}(t, x, h_n)\| = \mathcal{O}(h_n^s), \quad t \in [t_{n-1}, t_n],$$

but the solution of the alternate averaged equation

$$(1.7) \quad y' = \tilde{f}^{(2s)}(t, y, h_n), \quad y(t_{n-1}) = x(t_{n-1}),$$

satisfies

$$(1.8) \quad \|x(t_n) - y(t_n)\| = \mathcal{O}(h_n^{2s+1}).$$

Notice that the vector field (1.5) is a polynomial function in  $t$  of degree  $s - 1$  and from (1.6) one would expect the solution to be accurate up to order  $s$  instead of  $2s$ .

For simplicity in the presentation and without loss of generality from now on we consider only the first time step and we take  $t_0 = 0$  and  $h_1 = h$ .

There are many cases in which the numerical solution of (1.7) is simpler or can be carried out more efficiently than the numerical solution of (1.1). Some examples follow:

- When the vector field,  $f(t, x)$ , is only known on a time mesh (e.g., on an equidistant mesh for  $t$ ). One can solve (1.7) using any desired scheme and the algorithms will only require one to evaluate  $f(t, x)$  at the chosen nodes. If, for example, collocation methods are used, in general, one ends up with lower-order implicit methods.
- If the evaluation of the time-dependent functions in  $f(t, x)$  is not well defined for certain schemes. For example, to solve (1.1) using a numerical method with complex coefficients would require one to evaluate  $f(t, x)$  at complex values of  $t$  where the functions could not be well defined. This will not be the case if such a method is used to solve (1.7).
- Since the vector field in (1.7) is a low-order polynomial in  $t$ , it considerably simplifies the analysis to obtain the formal solution up to a given order, or equivalently, to obtain an autonomous equation (time-independent along  $t \in [0, h]$ , but changing on each time interval)

$$(1.9) \quad y' = \hat{f}^{(2s)}(y, h), \quad y(0) = x_0$$

( $\hat{f}^{(2s)}(y, h)$  will be a more involved nonlinear autonomous function of  $f(c_i h, y)$ ,  $i = 1, \dots, s$ , and their derivatives) such that the solution of (1.9) at  $t = h$  also

satisfies (1.8). This formal solution makes it simpler to check for which cases the Lie algebra associated to the vector field has some cancellations, which allows one to use tailored methods to solve problems with such structure. This result can be very useful in some cases, for example:

- If an efficient scheme for the autonomous case is known, one can approximate the formal solution by a sequence of compositions of such a scheme using fractional time steps over properly averaged vector fields (see Example 5).
- It simplifies the process to build tailored methods for problems where the associated Lie algebra has some particular structure (see section 3.2).

Since linear combinations of the vector field at different instants preserve its algebraic structure, the solution  $y(h)$  will retain most qualitative properties of the exact solution,  $x(h)$ . If one solves (1.7) using a geometric integrator [3, 14, 22, 27], the numerical results will have qualitative properties similar to those of the solution obtained when solving (1.1) using the same scheme.

In the linear case (1.3) this procedure allows us to recover the results for Lie group methods [6, 18, 19, 20, 24, 26]. These results for linear problems were extended to nonlinear problems in [29] by considering a linearization of the nonlinear equation that required the use of implicit methods. The results obtained in the present work are different and do not necessarily need to consider implicit methods. In [13] the authors also consider numerical integrators based on modified differential equations, but the modified vector fields depend on the numerical scheme to be used, and this is irrespective of whether the problem is autonomous or nonautonomous.

**2. To obtain a differential equation for the Gauss–Legendre quadrature rule.** We first present how to obtain a differential equation when the nodes of the Gauss–Legendre (GL) quadrature rule,  $c_i$ ,  $i = 1, \dots, s$ , of order  $2s$  are chosen, and next we will extend the results to any other quadrature rule.

**THEOREM 2.1.** *Given  $c_i$ ,  $i = 1, \dots, s$ , the  $s$  nodes of the GL quadrature rule of order  $2s$ , and  $x(h)$ , the solution at  $t = h$  of (1.1), the solution of the differential equation*

$$(2.1) \quad y' = \tilde{f}^{(2s)}(t, y, h), \quad y(0) = x_0,$$

with

$$(2.2) \quad \tilde{f}^{(2s)}(t, y, h) = \sum_{i=1}^s \mathcal{L}_i\left(\frac{t}{h}\right) f(c_i h, y),$$

where  $\mathcal{L}_i(t)$  is the Lagrange polynomial

$$\mathcal{L}_i(s) = \prod_{j=1, j \neq i}^s \frac{s - c_j}{c_i - c_j}, \quad s \in [0, 1],$$

satisfies that

$$\|x(h) - y(h)\| = \mathcal{O}(h^{2s+1}).$$

*Proof.* Let us consider the Runge–Kutta–Gauss–Legendre collocation method of order  $2s$  [17] to solve (1.1) and (2.1). Since

$$\tilde{f}^{(2s)}(c_i h, y, h) = f(c_i h, y), \quad i = 1, \dots, s,$$

we have that in both cases the numerical results are exactly the same:

$$(2.3) \quad \begin{aligned} k_i &= f\left(c_i h, x_0 + h \sum_{j=1}^s a_{ij} k_j\right), \quad i = 1, \dots, s, \\ x_1 &= x_0 + h \sum_{i=1}^s b_i k_i. \end{aligned}$$

As a result we have that

$$(2.4) \quad \begin{aligned} \|x(h) - y(h)\| &= \|x(h) - x_1 + x_1 - y(h)\| \leq \|x(h) - x_1\| + \|x_1 - y(h)\| \\ &= \mathcal{O}(h^{2s+1}). \end{aligned} \quad \square$$

Notice that, since  $\tilde{f}^{(2s)}(t, x, h)$  is a polynomial function of  $t$  of degree  $s - 1$  that interpolates  $f(t, x)$ , then (1.6) is satisfied. But this result only guarantees that both solutions differ at order  $\mathcal{O}(h^{s+1})$  along the time interval, while we have shown that, at the end of the interval, they agree up to twice this order.

**2.1. Differential equations in terms of moment integrals.** Given the moment integrals associated to the vector field  $f(t, x)$ ,

$$(2.5) \quad f_h^{(i)}(x) = \int_0^h \left(\frac{t - h/2}{h}\right)^i f(t, x) dt, \quad i = 0, 1, \dots, s - 1,$$

we show how to obtain the differential equation (1.7) where each  $f_j^{(2s)}(x, h_n)$  in (1.5) is a linear combination of the moment integrals.

For this purpose, we take into account that the interpolating polynomial vector field  $\tilde{f}^{(2s)}(t, x, h)$  is a polynomial function of degree  $s - 1$  in  $t$  and that it is a linear combination of the vector fields  $f_i(x) = f(c_i h, x)$ ,  $i = 1, \dots, s$ . In addition, following the idea proposed in [26], we see that

$$h f_i(x) = \mathcal{O}(h), \quad h(f_i(x) - f_j(x)) = \mathcal{O}(h^2), \quad \dots, \quad h \sum_{i=1}^s d_i f_i(x) = \mathcal{O}(h^s)$$

for appropriate choices of coefficients  $d_i$ . This result suggests to write  $\tilde{f}^{(2s)}(t, x, h)$  in terms of powers of  $h$ . For example, due to the time-symmetry of the solution and the symmetry of the nodes ( $c_{s+1-i} = 1 - c_i$ ,  $i = 1, 2, \dots$ ) we suggest to write it as a Taylor expansion about the midpoint, i.e., to take

$$(2.6) \quad t = \frac{h}{2} + \tau, \quad \tau \in \left[-\frac{h}{2}, \frac{h}{2}\right],$$

and to take the Taylor expansion about  $\tau = 0$  as follows:

$$(2.7) \quad \begin{aligned} \tilde{f}^{(2s)}(t, x, h) &= \tilde{f}^{(2s)}\left(\frac{h}{2} + \tau, x, h\right) = \frac{1}{h} \sum_{i=1}^s \left(\frac{\tau}{h}\right)^{i-1} F_i, \\ F_i &= h^i \frac{1}{(i-1)!} \frac{\partial^{i-1}}{\partial \tau^{i-1}} \tilde{f}^{(2s)}\left(\frac{h}{2} + \tau, x, h\right) \Big|_{\tau=0}. \end{aligned}$$

$F_i(x)$ ,  $i = 1, \dots, s$ , are linear combinations of  $f_j(x)$ ,  $j = 1, \dots, s$ , and  $F_i(x) = \mathcal{O}(h^i)$ ,  $F_i(x)F_j(x) = \mathcal{O}(h^{i+j})$ , etc. In addition, if we denote  $F_i(h, x) = F_i(x)$  to incorporate

the dependency of  $F_i$  with  $h$ , then, since the quadrature rule is symmetric, we have that

$$F_i(-h, x) = (-1)^i F_i(h, x),$$

and this property will be very useful in getting the formal solution with a reduced number of terms, which simplifies the process of building new methods.

We can easily check that the following hold:

- For  $s = 1$  we have  $c_1 = \frac{1}{2}$  and

$$(2.8) \quad y' = \frac{1}{h} F_1(y) = f\left(\frac{h}{2}, y\right),$$

whose solution provides a second-order approximation to the solution of (1.1) at  $t = h$ , i.e.,  $y(h) = x(h) + \mathcal{O}(h^3)$ .

- For  $s = 2$  we have  $(c_1, c_2) = (\frac{1}{2} - \frac{\sqrt{3}}{6}, \frac{1}{2} + \frac{\sqrt{3}}{6})$  and

$$(2.9) \quad \mathcal{L}_1\left(\frac{t}{h}\right) = -\sqrt{3}\left(\frac{t}{h} - \left(\frac{1}{2} + \frac{\sqrt{3}}{6}\right)\right), \quad \mathcal{L}_2\left(\frac{t}{h}\right) = \sqrt{3}\left(\frac{t}{h} - \left(\frac{1}{2} - \frac{\sqrt{3}}{6}\right)\right),$$

$$y' = \frac{1}{h} F_1(y) + \frac{1}{h^2} F_2(y) \tau,$$

$$F_1(y) = \frac{h}{2}(f(c_1 h, y) + f(c_2 h, y)), \quad F_2(y) = h\sqrt{3}(f(c_2 h, y) - f(c_1 h, y)),$$

with  $\tau = t - h/2$  and  $y(h) = x(h) + \mathcal{O}(h^5)$ .

- For  $s = 3$  we have  $(c_1, c_2, c_3) = (\frac{1}{2} - \frac{\sqrt{15}}{10}, \frac{1}{2}, \frac{1}{2} + \frac{\sqrt{15}}{10})$ , and we easily get

$$(2.10) \quad y' = \frac{1}{h} F_1(y) + \frac{1}{h^2} F_2(y) \tau + \frac{1}{h^3} F_3(y) \tau^2$$

with

$$F_1(y) = hf(c_2 h, y), \quad F_2(y) = \frac{\sqrt{15}h}{3}(f(c_3 h, y) - f(c_1 h, y)),$$

$$F_3(y) = \frac{10h}{3}(f(c_1 h, y) - 2f(c_2 h, y) + f(c_3 h, y)),$$

and  $y(h) = x(h) + \mathcal{O}(h^7)$ .

Let us now consider the moment integrals associated to the interpolating vector field (2.2):

$$(2.11) \quad \tilde{f}_h^{(i)}(x) = \int_0^h \left(\frac{t-h/2}{h}\right)^i \tilde{f}^{(2s)}(t, x, h) dt, \quad i = 0, 1, \dots, s-1.$$

Since  $\tilde{f}^{(2s)}(t, x, h)$  is a polynomial function in  $t$ , the integrals can be evaluated analytically, and taking into account (2.7) it is easy to check that

$$(2.12) \quad \tilde{f}_h^{(i)}(x) = \int_0^h \left(\frac{t-h/2}{h}\right)^i \frac{1}{h} \sum_{j=1}^s \left(\frac{t-h/2}{h}\right)^{j-1} F_j(x) dt = \sum_{j=1}^s \underbrace{\frac{1 - (-1)^{i+j}}{(i+j)2^{i+j}}}_{T_{i+1,j}^{(2s)}} F_j(x).$$

Notice that (2.12) is valid for any function  $f(t, x)$  and one can choose a function where  $\tilde{f}_h^{(i)}(x)$ ,  $i = 0, 1, \dots, s-1$ , and  $F_j(x)$ ,  $j = 1, \dots, s$ , correspond to two sets of

independent functions, so  $T_{ij}^{(2s)}$  must be the coefficients of an invertible matrix. Then, we can write

$$(2.13) \quad F_i(x) = \sum_{j=1}^s R_{ij}^{(2s)} \tilde{f}_h^{(j-1)}(x),$$

where  $R^{(2s)} = (T^{(2s)})^{-1}$ , and then substituting into (2.7) we have

$$(2.14) \quad \tilde{f}^{(2s)}(t, x, h) = \frac{1}{h} \sum_{i,j=1}^s \left( \frac{t-h/2}{h} \right)^{i-1} R_{ij}^{(2s)} \tilde{f}_h^{(j-1)}(x).$$

For example, taking  $2s = 2, 4, 6$  we have

$$(2.15) \quad R^{(2)} = 1, \quad R^{(4)} = \begin{pmatrix} 1 & 0 \\ 0 & 12 \end{pmatrix}, \quad R^{(6)} = \begin{pmatrix} \frac{9}{4} & 0 & -15 \\ 0 & 12 & 0 \\ -15 & 0 & 180 \end{pmatrix},$$

and then, (2.8)–(2.10) can be written in terms of the moment integrals associated to the interpolating polynomial as follows:

$$(2.16) \quad y' = \frac{1}{h} \tilde{f}_h^{(0)}(y),$$

$$(2.17) \quad y' = \frac{1}{h} \tilde{f}_h^{(0)}(y) + \frac{12}{h^2} \tilde{f}_h^{(1)}(y) \tau,$$

$$(2.18) \quad y' = \frac{1}{h} \left( \frac{9}{4} \tilde{f}_h^{(0)}(y) - 15 \tilde{f}_h^{(2)}(y) \right) + \frac{12}{h^2} \tilde{f}_h^{(1)}(y) \tau + \frac{1}{h^3} \left( -15 \tilde{f}_h^{(0)}(y) + 180 \tilde{f}_h^{(2)}(y) \right) \tau^2.$$

We are interested, however, in writing the differential equation in terms of the moment integrals (2.5) that are associated to the vector field in (1.1), and we present the following result.

**THEOREM 2.2.** *The solution of the differential equation*

$$(2.19) \quad z' = f^{(2s)}(t, z, h), \quad z(0) = x_0,$$

with

$$(2.20) \quad f^{(2s)}(t, z, h) = \frac{1}{h} \sum_{i,j=1}^s \left( \frac{t-h/2}{h} \right)^{i-1} R_{ij}^{(2s)} f_h^{(j-1)}(z),$$

satisfies that

$$\|x(h) - z(h)\| = \mathcal{O}(h^{2s+1}).$$

*Proof.* Taking into account that  $b_i, c_i$ ,  $i = 1, \dots, s$ , are the weights and nodes of the GL quadrature rule of order  $2s$ , and that this quadrature rule provides the exact solution for the integration of any polynomial function of degree lower than  $2s$ , we have that the solution of (2.11) is given by (2.12), but also by

$$(2.21) \quad \tilde{f}_h^{(i)}(y) = h \sum_{j=1}^s b_j \left( c_j - \frac{1}{2} \right)^i f(c_j h, y).$$

Then, if we use the same quadrature rule to integrate (2.5), we have that

$$f_h^{(i)}(y) = h \sum_{j=1}^s b_j \left(c_j - \frac{1}{2}\right)^i f(c_j h, y) + \mathcal{O}(h^{2s+1}),$$

so

$$(2.22) \quad f_h^{(i)}(y) = \tilde{f}_h^{(i)}(y) + \mathcal{O}(h^{2s+1}).$$

Then, we have that

$$(2.23) \quad \begin{aligned} \|x(h) - z(h)\| &= \|x(h) - y(h) + y(h) - z(h)\| \\ &\leq \|x(h) - y(h)\| + \|y(h) - z(h)\| = \mathcal{O}(h^{2s+1}). \end{aligned} \quad \square$$

As a result, the solutions of the differential equations

$$(2.24) \quad z' = \frac{1}{h} f_h^{(0)}(z),$$

$$(2.25) \quad z' = \frac{1}{h} f_h^{(0)}(z) + \frac{12}{h^2} f_h^{(1)}(z) \tau,$$

$$(2.26) \quad z' = \frac{1}{h} \left( \frac{9}{4} f_h^{(0)}(z) - 15 f_h^{(2)}(z) \right) + \frac{12}{h^2} f_h^{(1)}(z) \tau + \frac{1}{h^3} \left( -15 f_h^{(0)}(z) + 180 f_h^{(2)}(z) \right) \tau^2$$

agree with the solution of (1.1) up to second, fourth, and sixth order, respectively.

An alternate proof that does not consider Runge–Kutta (RK) methods is the following one.<sup>1</sup> The vector field (2.20) can be written as

$$(2.27) \quad f^{(2s)}(t, z, h) = \int_0^1 \sum_{i,j=1}^s \left(\frac{t}{h} - \frac{1}{2}\right)^{i-1} R_{ij}^{(2s)} \left(\sigma - \frac{1}{2}\right)^{j-1} f(\sigma h, z) d\sigma,$$

which is a particular case of the more general function

$$(2.28) \quad f^{(2s)}(t, z, h) = \int_0^1 P_{s-1}\left(\frac{t}{h}, \sigma\right) f(\sigma h, z) d\sigma$$

with  $P_{s-1}(\frac{t}{h}, \sigma)$  a polynomial function of degree  $s-1$  in both  $t/h$  and  $\sigma$ . If we consider the Taylor expansion about the origin of

$$(2.29) \quad f(t, x) - f^{(2s)}(t, x, h) = \sum_{j \geq 0} p_j \left(\frac{t}{h}\right) \frac{h^j}{j!} \frac{\partial^j}{\partial t^j} f(t, x) \Big|_{t=0},$$

where (taking  $\xi = t/h$ )  $p_j(\xi) = \xi^j + q_{s-1}(\xi)$  with  $q_{s-1}$  a polynomial of degree  $s-1$ , then if  $P_{s-1}(\xi, \sigma)$  is chosen such that

$$p_j(\xi) = 0, \quad j = 0, 1, \dots, s-1,$$

<sup>1</sup>I would like to thank Ernst Hairer for suggesting this alternate proof.

and  $p_j(\xi)$ ,  $j = s, s+1, \dots, 2s-1$ , are orthogonal to all polynomials of degree  $0 \leq k \leq 2s-j-1$ , or equivalently,

$$\int_0^1 p_j(\sigma)\sigma^k d\sigma = 0 \quad \text{for } 0 \leq k \leq 2s-j-1 \quad \text{and } j = s, \dots, 2s-1,$$

then the solution of (2.19) agrees with the solution of (1.1) up to order  $2s$ . The choice in (2.20) satisfies these conditions, and the proof of the theorem could be done avoiding the use of RK methods.

For example, for  $s = 2$  we have from (2.27), (2.28) that

$$P_1(\xi, \sigma) = (4 - 6\sigma) + (12\sigma - 6)\xi$$

and

$$p_0(\sigma) = 0, \quad p_1(\sigma) = 0, \quad p_2(\sigma) = \sigma^2 - \sigma + \frac{1}{6}, \quad p_3(\sigma) = \sigma^3 - \frac{9}{10}\sigma + \frac{1}{5}.$$

If we approximate the integrals using the GL quadrature rule, we recover the results from Theorem 2.1. Obviously, one can use any other quadrature rule, and we next provide a simple procedure to directly change from the GL quadrature rule to any other quadrature rule.

**2.2. To extend the result to other quadrature rules.** Given another quadrature rule of order  $p \geq 2s$  with nodes  $\hat{c}_i$ ,  $i = 1, \dots, m$ , we could repeat the same procedure as before by considering, for example, an RK scheme of order  $2s$  that shares the same values for the coefficients  $\hat{c}_i$ . However, the interpolating polynomial would be a polynomial of degree  $m-1$  in  $t$  with  $m > s$ .

In the following, we present a simpler rule to change from one quadrature rule to another quadrature rule that does not increase the degree of the polynomial. The following result indicates how to obtain another differential equation by replacing each  $f(c_i h, x)$ ,  $i = 1, \dots, s$ , in  $\tilde{f}^{(2s)}(t, x, h)$  by a linear combination of  $f(\hat{c}_j, x)$ ,  $j = 1, \dots, m$ , i.e., to replace  $f(c_i, x)$  by  $\sum_{j=1}^m a_{ij} f(\hat{c}_j, x)$  with proper coefficients  $a_{ij}$  such that the new vector field (with this change it is still a polynomial of degree  $s-1$  in  $t$ ) satisfies a relation similar to (1.6) and the solution of the associated equation at the end of the step is still of order  $2s$  as in (1.8).

**THEOREM 2.3.** *Given the weights and nodes  $b_i, c_j$ ,  $j = 1, \dots, s$ , of the GL quadrature rule of order  $2s$ , and the weights and nodes  $\hat{b}_i, \hat{c}_j$ ,  $j = 1, \dots, m$ , of a quadrature rule of order  $2s$  or higher, the differential equation*

$$(2.30) \quad z' = \hat{f}^{(2s)}(t, z, h), \quad z(0) = x_0,$$

where

$$(2.31) \quad \hat{f}^{(2s)}(t, x, h) = \frac{1}{h} \sum_{i=1}^s \left(\frac{\tau}{h}\right)^{i-1} \hat{F}_i(z),$$

and  $\hat{F}_i(z)$  is given by  $F_i(z)$  in (2.7), replacing each  $f(c_i h, z)$  by  $\sum_{j=1}^m a_{ij} f(\hat{c}_j h, y)$ , where

$$(2.32) \quad a_{ij} = \left( (Q^{(2s)})^{-1} \hat{Q}^{(2s,m)} \right)_{ij}, \quad Q_{ij}^{(2s)} = b_j \left( c_j - \frac{1}{2} \right)^{i-1}, \quad \hat{Q}_{ik}^{(2s,m)} = \hat{b}_k \left( \hat{c}_k - \frac{1}{2} \right)^{i-1}$$



with  $Q^{(2s)} \in \mathbb{R}^{s \times s}$ ,  $\widehat{Q}^{(2s,m)} \in \mathbb{R}^{s \times m}$ , satisfies that

$$(2.33) \quad \|x(h) - z(h)\| = \mathcal{O}(h^{2s+1}).$$

*Proof.* Taking into account (2.22), we then have that

$$\begin{aligned} h \sum_{j=1}^s b_j \underbrace{\left(c_j - \frac{1}{2}\right)^i}_{Q_{i+1,j}^{(2s)}} f(c_j h, y) &= \widetilde{f}_h^{(i)}(y) = f_h^{(i)}(y) + \mathcal{O}(h^{2s+1}) \\ &= h \sum_{j=1}^m \underbrace{\widehat{b}_j \left(\widehat{c}_j - \frac{1}{2}\right)^i}_{\widehat{Q}_{i+1,j}^{(2s,m)}} f(\widehat{c}_j h, y) + \mathcal{O}(h^{2s+1}). \end{aligned}$$

Then, in (2.8)–(2.10) we can safely make the substitutions

$$f(c_i h, y) \quad \longrightarrow \quad \sum_{j=1}^m a_{ij} f(\widehat{c}_j h, y),$$

where the coefficients  $a_{ij}$  are given by (2.32) (since  $Q^{(2s)}$  is an invertible matrix) and the solution of the new differential equation is still accurate up to order  $2s$  since the vector fields differ at this order.  $\square$

*Remark.* We have written the differential equation in terms of the moment integrals (2.5) associated to the vector field in (1.1). In some cases the integrals can be solved exactly. In addition, if some time-dependent functions evolve very smoothly (adiabatically) in  $t$  and are costly to compute, one could approximate the integrals using a lower-order quadrature rule with few evaluations per step, where, in general, the cost will be reduced while keeping the accuracy.

**2.3. Examples.**

*Example 1.* We illustrate how to build differential equations that provide solutions that are accurate to second and fourth order, respectively, when the vector field is known on an equidistant time mesh.

- Second-order method: we take  $s = 1$  and the trapezoidal rule,  $(\widehat{b}_1, \widehat{b}_2) = (\frac{1}{2}, \frac{1}{2})$ ,  $(\widehat{c}_1, \widehat{c}_2) = (0, 1)$ , so

$$(2.34) \quad \begin{aligned} Q^{(2)} &= 1, \quad \widehat{Q}^{(2,2)} = \begin{pmatrix} 1 & 1 \\ 2 & 2 \end{pmatrix} \Rightarrow a = (Q^{(2)})^{-1} \widehat{Q}^{(2,2)} = \begin{pmatrix} 1 & 1 \\ 2 & 2 \end{pmatrix}, \\ z' &= \frac{1}{h} \widehat{F}_1(z), \quad \widehat{F}_1(y) = \frac{h}{2} (f(\widehat{c}_1 h, z) + f(\widehat{c}_2 h, z)), \end{aligned}$$

which corresponds to the well-known trapezoidal-averaged method, i.e.,

$$z' = \frac{1}{2} (f(0, z) + f(h, z)),$$

whose solution satisfies  $z(h) = x(h) + \mathcal{O}(h^3)$ .

- Fourth-order method: we take  $s = 2$  with the Simpson rule  $(\widehat{b}_1, \widehat{b}_2, \widehat{b}_3) = (\frac{1}{6}, \frac{4}{6}, \frac{1}{6})$ ,  $(\widehat{c}_1, \widehat{c}_2, \widehat{c}_3) = (0, \frac{1}{2}, 1)$ , so

$$(2.35) \quad \begin{aligned} (Q^{(4)})^{-1} \widehat{Q}^{(4,3)} &= \frac{1}{6} \begin{pmatrix} 1 + \sqrt{3} & 4 & 1 - \sqrt{3} \\ 1 - \sqrt{3} & 4 & 1 + \sqrt{3} \end{pmatrix}, \\ z' &= \frac{1}{h} \widehat{F}_1(z) + \frac{1}{h^2} \widehat{F}_2(z) \tau, \end{aligned}$$

with

$$\begin{aligned}\widehat{F}_1(z) &= \frac{h}{2} \sum_{j=1}^3 (a_{1,j} + a_{2,j}) f(\hat{c}_j h, z) = \frac{h}{6} (f(\hat{c}_1 h, z) + 4f(\hat{c}_2 h, z) + f(\hat{c}_3 h, z)), \\ \widehat{F}_2(z) &= h\sqrt{3} \sum_{j=1}^3 (a_{2,j} - a_{1,j}) f(\hat{c}_j h, z) = h(f(\hat{c}_3 h, z) - f(\hat{c}_1 h, z));\end{aligned}$$

i.e., the solution of the differential equation

$$(2.36) \quad z' = \frac{1}{6} \left( f(0, z) + 4f\left(\frac{h}{2}, z\right) + f(h, z) \right) + (f(h, z) - f(0, z)) \frac{\tau}{h}$$

satisfies  $z(h) = x(h) + \mathcal{O}(h^5)$ .

*Example 2.* Let us consider the fourth-order Simpson rule and the alternate differential equations

$$(2.37) \quad y' = \tilde{f}^{(4)}(t, y, h), \quad y(0) = x_0,$$

with

$$(2.38) \quad \begin{aligned}\tilde{f}^{(4)}(t, y, h) &= \sum_{i=1}^3 \mathcal{L}_i\left(\frac{t}{h}\right) f(c_i h, y) \\ &= f(0, y) + \left( -3f(0, y) + 4f\left(\frac{h}{2}, y\right) - f(h, y) \right) \frac{t}{h} \\ &\quad + \left( 2f(0, y) - 4f\left(\frac{h}{2}, y\right) + 2f(h, y) \right) \frac{t^2}{h^2},\end{aligned}$$

where  $\tilde{f}^{(4)}(t, y, h)$  is the interpolating polynomial. The standard fourth-order RK method to solve (1.1) and (2.37) provides the same numerical result, so the solution of (2.37) at  $t = h$  provides a fourth-order approximation. However, the vector field in (2.37) is quadratic in  $t$  instead of linear as in (2.36).

*Example 3.* Let us consider the scalar differential equation

$$(2.39) \quad x' = \sum_{j=1}^k \alpha_j(t) f_j(x), \quad x(0) = x_0.$$

We build a differential equation that only requires the evaluation of  $\alpha_{j,i} = \alpha_j(c_i h)$ ,  $i = 1, 2$  (evaluated at the fourth-order GL quadrature rule), and whose solution at the end of each step provides a fourth-order approximation.

We take  $s = 2$  and evaluate the constants

$$a_{j,1} = \frac{h}{2} (\alpha_{j,1} + \alpha_{j,2}), \quad a_{j,2} = h\sqrt{3} (\alpha_{j,2} - \alpha_{j,1}),$$

and finally the equation to be solved is

$$(2.40) \quad y' = \sum_{j=1}^k \left( \frac{1}{h} a_{j,1} + \frac{1}{h^2} a_{j,2} \tau \right) f_j(y),$$

with  $\tau = t - h/2$ , which is only linear in  $t$ . Notice that one can use a given numerical scheme (see, e.g., [15, 17] and references therein) to solve both (2.39) and (2.40), but in the last case such a scheme will only require one to evaluate the functions  $\alpha_j(t)$  at the fourth-order GL quadrature rule. Furthermore, in the case where  $\alpha_j(t)$  evolve very smoothly in  $t$ , one could use a higher-order scheme to solve (2.40) that could provide more accurate results (in many cases this occurs at a moderate additional computational cost). Notice that if one uses a Lie group method [19] to solve (2.39), the same Lie group method can be used to solve (2.40), and in some cases this can be achieved more efficiently in the second case.

*Example 4.* Let us now consider the Kepler problem with a time-dependent mass given by the Hamiltonian

$$(2.41) \quad H(t, q, p) = \frac{1}{2}p^T p - \mu(t)\frac{1}{r}$$

with  $q, p \in \mathbb{R}^3$ ,  $r = \|q\| = \sqrt{q^T q}$ , with Hamilton equations

$$(2.42) \quad \begin{aligned} q' &= p, \\ p' &= -\mu(t)\frac{q}{r^{3/2}}. \end{aligned}$$

If one takes  $t$  as a coordinate, say  $q_t = t$ , and its associated momentum,  $p_t$ , the Hamilton equations associated to the autonomous Hamiltonian

$$K(q_t, q, p_t, p) = \left( \frac{1}{2}p^T p + p_t \right) - \mu(q_t)\frac{1}{r}$$

have the same solution for  $q, p$ , and symplectic splitting methods can be used. Let  $\{a_i, b_i\}_{i=1}^m$  be the coefficients of a splitting method; then the scheme to advance one time step, from  $(q_n, p_n)$  to  $(q_{n+1}, p_{n+1})$ , is given by

$$(2.43) \quad \begin{aligned} &Q_0 = q_n, \quad P_0 = p_n \\ &\mathbf{for} \ i = 1 : m \\ &\quad Q_i = Q_{i-1} + a_i h P_{i-1}, \\ &\quad P_i = P_{i-1} - b_i h \mu(d_i h) \frac{Q_i}{R_i^{3/2}} \\ &\mathbf{end} \\ &q_{n+1} = Q_m, \quad p_{n+1} = P_m \end{aligned}$$

with  $R_i = \sqrt{Q_i^T Q_i}$ ,  $d_i = \sum_{j=1}^i a_j$ .

If we take  $s = 2$ , the fourth-order Gaussian quadrature rule, we compute  $\mu_i = \mu(c_i h)$ ,  $i = 1, 2$ , and evaluate

$$M_1 = \frac{h}{2}(\mu_1 + \mu_2), \quad M_2 = h\sqrt{3}(\mu_2 - \mu_1),$$

then the differential equation given by

$$(2.44) \quad \begin{aligned} q' &= p, \\ p' &= -\left( \frac{1}{h}M_1 + \frac{1}{h^2}M_2 \tau \right) \frac{q}{r^{3/2}} \end{aligned}$$

provides a fourth-order accurate solution.

If, however, we use the Simpson quadrature rule, we should replace  $M_1, M_2$  by

$$M_1 = \frac{h}{6} \left( \mu(0) + 4\mu\left(\frac{h}{2}\right) + \mu(h) \right), \quad M_2 = h(\mu(h) - \mu(0)).$$

Efficient fourth-order splitting methods can be found in [11] with  $m = 6$ , so six evaluations of  $\mu(t)$  per step are required in the algorithm (2.43), but the same splitting method will only require two evaluations per step to solve (2.44) for both choices of the quadrature rule. In some other cases, the scheme requires one to evaluate  $\mu(t)$  at values where it is not well defined or it can be difficult to evaluate as, for example, if one uses splitting methods with complex coefficients [4, 12, 16] (these methods are useful, e.g., when forward integrations are either necessary or highly convenient), but this would not be the case if one solves instead (2.44) with such schemes since  $\mu(t)$  has been previously evaluated at the nodes of the chosen quadrature rule.

**3. To construct the formal solution.** Let us consider the linear equation (1.3) with formal solution at  $t = h$  given by  $x(h) = \Phi_h x_0$ . It is known that for  $h$  small enough,  $\Phi_h$  is close to the identity and it can be written as  $\Phi_h = e^{\Omega_h}$ , with  $\Omega_h$  a constant matrix. Then, the solution of the autonomous linear equation

$$y' = \frac{1}{h} \Omega_h y, \quad y(0) = x_0$$

satisfies, in general, that  $y(t) \neq x(t)$  for  $t \in (0, h)$ , but  $y(h) = x(h)$ , and  $e^{\Omega_h}$  provides the formal solution at  $t = h$  that can be used to build numerical schemes. For example, from a properly truncated Magnus series expansion [5, 20, 23] where the nested integrals are properly approximated making use of the matrix  $A(t)$  evaluated on a given quadrature rule, one can build a matrix,  $M_h$ , such that  $M_h = \Omega_h + \mathcal{O}(h^{p+1})$  and the solution of

$$z' = \frac{1}{h} M_h z, \quad z(0) = x_0$$

satisfies  $z(h) = x(h) + \mathcal{O}(h^{p+1})$ .

We now analyze how to obtain the formal solution of (1.1) that can be considered as the solution of an autonomous equation<sup>2</sup> where the exact solution at  $t = h$  coincides with the desired solution up to a given order.

For this purpose we consider the Magnus expansion for nonlinear differential equations [9, 25]. The formalism used in this work considers a polynomial of degree  $s - 1$  in  $t$  that allows one to obtain the formal solution up to order  $2s$  in the simplest way, and that can be adapted to be used with any desired quadrature rule.

This can be achieved by writing the nonlinear differential equation in terms of linear Lie operators, which we briefly review.

**3.1. Lie algebra of Lie operators.** Let  $\mathcal{A}$  be the set of all infinitely differentiable vector functions defined in a given region,  $f : D \subset \mathbb{R} \times \mathbb{R}^d \rightarrow \mathbb{R}^d$ . This set, with the addition and multiplication by a scalar and the Lie bracket as the internal product, forms a Lie algebra of vector fields. Given  $f \in \mathcal{A}$ , we define the Lie operator, or Lie derivative, associated to  $f = (f_1(t, x), \dots, f_d(t, x))$  and denoted by  $L_f$ , as the operator  $L_f : \mathcal{A} \rightarrow \mathcal{A}$ , given by

$$(3.1) \quad L_f = \sum_{i=1}^d f_i(t, x) \frac{\partial}{\partial x_i}$$

<sup>2</sup>The associated vector field is autonomous on each time interval,  $[t_n, t_{n+1}]$ , but it changes on each time step.

(derivatives with respect to coordinates, but not with respect to  $t$ ), which acts on  $g \in \mathcal{A}$  as

$$(3.2) \quad L_f g(t, x) = w(t, x), \quad \text{where} \quad w_i(t, x) = \sum_{j=1}^d f_j(t, x) \frac{\partial g_i(t, x)}{\partial x_j}.$$

Since  $L_f$  acts on each component of the vector field,  $g$ , we will use it acting both on vector functions as well as on scalar functions. For simplicity in the presentation, we write

$$L_f \text{Id}(x) = L_f x = f(t, x).$$

We denote by  $\hat{\mathcal{A}}$  the set of Lie operators that also have the structure of a vector space with the addition and product by a scalar

$$(3.3) \quad (L_f + L_g)w = L_{f+g}w, \quad \alpha L_f g = L_{\alpha f}g, \quad \alpha \in \mathbb{R}.$$

Let us now introduce the internal product,  $[\cdot, \cdot] : \hat{\mathcal{A}} \times \hat{\mathcal{A}} \rightarrow \hat{\mathcal{A}}$ , as follows:

$$(3.4) \quad [L_f, L_g] = L_f L_g - L_g L_f,$$

i.e., the commutator of Lie operators. This internal product is bilinear and skew-symmetric, and, given  $f, g, k \in \mathcal{A}$ ,

$$(3.5) \quad [L_f, L_g]k = L_{(f,g)}k,$$

where  $(f, g)$  denotes a vector field obtained from the Lie bracket of the vector fields whose components are given by

$$(f, g)_i = \sum_{j=1}^d \left( f_j \frac{\partial g_i}{\partial x_j} - g_j \frac{\partial f_i}{\partial x_j} \right).$$

Then, the set  $\hat{\mathcal{A}}$  with the commutator as the internal product is a Lie algebra that is isomorphic to the Lie algebra of vector fields with the Lie bracket as the internal product.

DEFINITION 3.1 (Lie transform). *Given  $f(t, x) \in \mathcal{A}$ , we define the associated Lie transform,  $e^{L_f} : \mathcal{A} \rightarrow \mathcal{A}$ , as*

$$e^{L_f} = \sum_{k=0}^{\infty} \frac{L_f^k}{k!}.$$

Here  $L_f^i \psi = L_f(L_f^{i-1} \psi)$  and  $L_f^0 \psi = \psi$ , where  $\psi$  can be either a scalar or a vector function.

Let  $x = \psi_t(x_0)$  be the solution of the autonomous equation  $x' = f(x)$ . This map can be written as the action of an operator on the identity function as follows:

$$x = \psi_t(x_0) = \Psi_t \text{Id}(x_0).$$

The equation to be satisfied by the operator  $\Psi_t$  is

$$\begin{aligned} \frac{dx}{dt} &= \frac{d}{dt} \Psi_t \text{Id}(x_0), \\ f(x) &= L_{f(x)} \text{Id}(x) = L_{f(\Psi_t \circ \text{Id}(x_0))} \text{Id}(\Psi_t \circ \text{Id}(x_0)) = \Psi_t L_{f(x_0)} \text{Id}(x_0), \end{aligned}$$

so the operator differential equation is

$$(3.6) \quad \frac{d\Psi_t}{dt} = \Psi_t L_{f(x_0)},$$

and the solution can formally be written as

$$x = \psi_t(x_0) = e^{tL_{f(x_0)}} \text{Id}(x_0)$$

or, in short,  $x = e^{tL_{f(x_0)}} x_0$ , where the derivatives act on  $x_0$ , which can be seen as an independent set of coordinates (a transformation from  $x_0$  to  $x$ ).

Similarly, the solution of the nonautonomous equation (1.1) at  $t = h$  can formally be written as a Lie transform of the (unknown) vector field  $f_\Omega(h, x_0)$ , i.e.,

$$(3.7) \quad \Psi_h = \exp(hL_{f_\Omega(h, x_0)}).$$

The equation to be satisfied by  $\Psi_t$  is now

$$(3.8) \quad \frac{d\Psi_t}{dt} = \Psi_t L_{f(t, x_0)}.$$

The linear operator  $L_{f(t, x_0)}$  appears on the right side of  $\Psi_t$  instead of the left side, so we can use the property

$$0 = \frac{d}{dt}(\Psi_t^{-1}\Psi_t) = \frac{d\Psi_t^{-1}}{dt}\Psi_t + \Psi_t^{-1}\frac{d\Psi_t}{dt} = \frac{d\Psi_t^{-1}}{dt}\Psi_t + \Psi_t^{-1}\Psi_t L_{f(t, x_0)}.$$

Thus

$$(3.9) \quad \frac{d\Psi_t^{-1}}{dt} = -L_{f(t, x_0)}\Psi_t^{-1},$$

and, since this is a linear differential equation, we can solve it using the Magnus series expansion similarly to the matrix case.

Given the linear equation (1.3), if we write the formal solution at  $t = h$  as  $x(h) = e^{\Omega_h} x_0$ , with  $\Omega_h$  given by the Magnus series expansion [5, 20, 23],  $\Omega_h = \sum_{k=1}^{\infty} \Omega_{h,k}$ , we have that

$$\Omega_{h,1} = \int_0^h A(t_1) dt_1, \quad \Omega_{h,2} = \frac{1}{2} \int_0^h \int_0^{t_1} [A(t_1), A(t_2)] dt_2 dt_1, \quad \dots,$$

where  $[P, Q] = PQ - QP$ . Then, the solution of (3.9) can formally be written as the exponential of the series (the series for nonlinear problems usually does not converge, but a proper truncation of it usually provides accurate solutions)

$$\Psi_h^{-1} = \exp\left(h \sum_{k=1}^{\infty} L_{f_{\Omega_k}}\right).$$

Notice that from (3.7) we have that  $\Psi_h^{-1} = \exp(-hL_{f_\Omega(h, x_0)})$ , and then

$$-hL_{f_\Omega(h, x_0)} = \int_0^h -L_{f(t_1, x_0)} dt_1 + \frac{1}{2} \int_0^h \int_0^{t_1} [-L_{f(t_1, x_0)}, -L_{f(t_2, x_0)}] dt_2 dt_1 + \dots$$

Then taking into account the properties (3.3) and (3.5), we find that  $f_\Omega(h, x_0)$  is given by

$$(3.10) \quad hf_\Omega(h, x_0) = \int_0^h f(t_1, x_0) dt_1 + \frac{1}{2} \int_0^h \int_0^{t_1} (f(t_2, x_0), f(t_1, x_0)) dt_2 dt_1 + \dots$$

We observe that the nested Lie brackets with even number of vector fields have the opposite sign of similar terms obtained when considering the Magnus series for solving the linear problem. This is due to the following property: given the linear vector fields  $f(x) = Fx$ ,  $g(x) = Gx$  with  $F, G \in \mathbb{C}^{d \times d}$ , the Lie bracket is given by

$$(f(x), g(x)) = (Fx, Gx) = [G, F]x.$$

In a similar way, given the Lie transforms

$$e^{tL_{f(x_0)}} x_0 = \psi_t^f(x_0), \quad e^{tL_{g(x_0)}} x_0 = \psi_t^g(x_0),$$

we have that<sup>3</sup>

$$e^{tL_{g(x_0)}} e^{tL_{f(x_0)}} x_0 = e^{tL_{g(x_0)}} \psi_t^f(x_0) = \psi_t^f(e^{tL_{g(x_0)}} x_0) = \psi_t^f \circ \psi_t^g(x_0).$$

Since (2.1) provides an approximation to order  $2s$  to the solution  $x(h)$  and  $\tilde{f}^{(2s)}(t, x, h)$  is a polynomial function in  $t$  of degree  $s - 1$ , we can solve (2.1) using the Magnus expansion where the integrals can be exactly evaluated.

To take the sufficient number of terms in the Magnus series to obtain an approximation to order  $2s$ , we consider the following property [6, 21, 26] (see also [5] and references therein): each term in  $hL_{f_{\Omega_k}}$  is an odd function of  $h$ , and in particular

$$\begin{aligned} hL_{f_{\Omega_1}} &= \mathcal{O}(h), & hL_{f_{\Omega_2}} &= \mathcal{O}(h^3), \\ hL_{f_{\Omega_{2m-1}}} &= \mathcal{O}(h^{2m+1}), & hL_{f_{\Omega_{2m}}} &= \mathcal{O}(h^{2m+1}), \quad m > 1. \end{aligned}$$

Let us denote by

$$y(h) = \exp(hL_{\tilde{f}_{\Omega}^{(2s)}(h, x_0)})x_0$$

the solution of (2.1) when considering the interpolating polynomial; then the first two terms in the series (3.10) are given by

$$(3.11) \quad hf_{\Omega_1}^{(2s)}(h, x_0) = \int_0^h \tilde{f}_h^{(2s)}(t, x_0) dt = \int_{-h/2}^{h/2} \frac{1}{h} \sum_{i=1}^s \left(\frac{\tau}{h}\right)^i F_i(x_0) d\tau$$

$$(3.12) \quad = \sum_{i=1}^s \frac{1 - (-1)^i}{i 2^i} F_i(x_0),$$

$$hf_{\Omega_2}^{(2s)}(h, x_0) = \frac{1}{2} \int_{-h/2}^{h/2} \int_{-h/2}^{\sigma_1} (\tilde{f}_h^{(2s)}(\sigma_2, x_0), \tilde{f}_h^{(2s)}(\sigma_1, x_0)) d\sigma_2 d\sigma_1.$$

Then, for  $2s = 2, 4, 6$  we have the following:

- A second-order method is obtained taking  $2s = 2$  and truncating the Magnus expansion to the first term for the vector field given by (2.8)

$$hf_{\Omega}^{(2)} = hf_{\Omega_1}^{(2)} = \int_0^h \frac{1}{h} F_1(x_0) dt = F_1(x_0).$$

<sup>3</sup>See the appendix of [3] (and references therein) for a review on the Lie algebra of vector fields, the Lie algebra of the associated Lie operators, and the order in which they appear.

- A fourth-order method is obtained taking  $2s = 4$  and truncating the Magnus expansion to the second term for the vector field given by (2.9). By introducing the change of variable in the integral  $\tau = t - \frac{h}{2}$  we have

$$\begin{aligned}
 (3.13) \quad h\tilde{f}_{\Omega}^{(4)}(y) &= \int_{-h/2}^{h/2} \left( \frac{1}{h}F_1(y) + \frac{1}{h^2}F_2(y) \tau \right) d\tau \\
 &\quad + \frac{1}{2} \int_{-h/2}^{h/2} \int_{-h/2}^{s_1} \left( \frac{1}{h}F_1(y) + \frac{1}{h^2}F_2(y) s_2, \frac{1}{h}F_1(y) + \frac{1}{h^2}F_2(y) s_1 \right) ds_2 ds_1 \\
 &= F_1(y) + \frac{1}{12}(F_1(y), F_2(y)).
 \end{aligned}$$

- A sixth-order method is obtained taking  $f_{\tilde{\Omega}_1^{(6)}}, \dots, f_{\tilde{\Omega}_4^{(6)}}$  and (2.10)

$$\begin{aligned}
 h\tilde{f}_{\Omega_1}^{(6)}(y) &= F_1(y) + \frac{1}{12}F_3(y), \\
 h\tilde{f}_{\Omega_2}^{(6)}(y) &= \frac{1}{12}(12) - \frac{1}{240}(23), \\
 h\tilde{f}_{\Omega_3}^{(6)}(y) &= \frac{1}{360}(113) - \frac{1}{240}(212) + \mathcal{O}(h^7), \\
 h\tilde{f}_{\Omega_4}^{(6)}(y) &= -\frac{1}{720}(1112) + \mathcal{O}(h^7),
 \end{aligned}$$

so

$$\begin{aligned}
 h\tilde{f}_{\Omega}^{(6)}(y) &= F_1(y) + \frac{1}{12}F_3(y) + \frac{1}{12}(12) - \frac{1}{240}(23) + \frac{1}{360}(113) \\
 &\quad - \frac{1}{240}(212) - \frac{1}{720}(1112) + \mathcal{O}(h^7),
 \end{aligned}$$

where  $(ij \dots kl)$  represents the nested Lie bracket

$$(F_i(y), (F_j(y), (\dots, (F_k(y), F_l(y)) \dots))).$$

In general, the evaluation of Lie brackets of vector fields is a very costly and cumbersome computation. However, the formal solution written in this form allows us to build new schemes at different orders of accuracy that are simpler to evaluate and agree with the formal solution up to the desired order.

For example, the differential equation with the vector field given by (3.13) that provides a fourth-order approximation corresponds to the Lie transform

$$\exp\left(L_{F_1} + \frac{1}{12}[L_{F_1}, L_{F_2}]\right),$$

where  $L_{F_i} = \mathcal{O}(h^i)$ , but this transformation can be approximated by a composition of maps that do not involve commutators (or Lie brackets) following the same decomposition techniques as those used to build commutator-free (quasi-)Magnus integrators [1, 7, 11, 28]. For instance, we have

$$\exp\left(L_{F_1} + \frac{1}{12}[L_{F_1}, L_{F_2}]\right) = \exp\left(\frac{1}{2}L_{F_1} - \frac{1}{6}L_{F_2}\right) \exp\left(\frac{1}{2}L_{F_1} + \frac{1}{6}L_{F_2}\right) + \mathcal{O}(h^5),$$



or, in terms of vector fields and taking into account that the Lie transform acts in the reverse order, this is equivalent to sequentially solving

$$\begin{aligned} z' &= \frac{1}{h} \left( \frac{1}{2}F_1(z) - \frac{1}{6}F_2(z) \right), & z(0) &= x_0, \\ y' &= \frac{1}{h} \left( \frac{1}{2}F_1(y) + \frac{1}{6}F_2(y) \right), & y(0) &= z(h), \end{aligned}$$

or equivalently,

$$(3.14) \quad z' = (a_1 f(c_1 h, z) + a_2 f(c_2 h, z)), \quad z(0) = x_0,$$

$$(3.15) \quad y' = (a_2 f(c_1 h, y) + a_1 f(c_2 h, y)), \quad y(0) = z(h),$$

with

$$(3.16) \quad a_1 = \frac{1}{4} + \frac{\sqrt{3}}{6}, \quad a_2 = \frac{1}{4} - \frac{\sqrt{3}}{6}.$$

*Example 5.* Let us consider the nonautonomous Kepler problem (2.41). The evolution for the autonomous case (with  $\mu$  a constant parameter) is given by the map

$$(3.17) \quad \begin{aligned} (q(t), p(t)) &= \Phi(q_0, p_0, t, \mu) \\ &= (f q_0 + g p_0, f_p q_0 + g_p p_0), \end{aligned}$$

where  $f, f_p, g, g_p$  are computed as follows (see [3] and references therein):

$$\begin{aligned} r_0 &= \|q_0\|, \quad u = q_0^T p_0, \quad E = \frac{1}{2} p_0^T p_0 - \mu \frac{1}{r_0}, \quad a = -\frac{\mu}{2E}, \quad w = \sqrt{\frac{\mu}{a^3}}, \quad \sigma = 1 - \frac{r_0}{a}, \\ \psi &= \frac{u}{wa^2}, \quad wt = x - \sigma \sin x + \psi(1 - \cos x), \\ f &= 1 + \frac{(\cos x - 1)a}{r_0}, \quad g = t + \frac{\sin x - x}{w}, \\ f_p &= -\frac{aw \sin x}{r_0(1 - \sigma \cos x + \psi \sin x)}, \quad g_p = 1 + \frac{\cos x - 1}{1 - \sigma \cos x + \psi \sin x}, \end{aligned}$$

where  $x$  has to be evaluated numerically.

A second-order scheme is given by the midpoint rule

$$(3.18) \quad (q_1, p_1) = \Phi\left(q_0, p_0, h, \mu\left(\frac{h}{2}\right)\right),$$

while the fourth-order scheme (3.14), (3.15) is given by the sequence of compositions

$$(3.19) \quad (q_{1/2}, p_{1/2}) = \Phi(q_0, p_0, (a_1 + a_2)h, M_1), \quad M_1 = \frac{a_1 \mu(c_1 h) + a_2 \mu(c_2 h)}{a_1 + a_2},$$

$$(3.20) \quad (q_1, p_1) = \Phi(q_{1/2}, p_{1/2}, (a_2 + a_1)h, M_2), \quad M_2 = \frac{a_2 \mu(c_1 h) + a_1 \mu(c_2 h)}{a_1 + a_2}.$$

Since  $a_1 + a_2 = \frac{1}{2}$ , this can be considered as the composition of a half step with one method followed by a half step with its adjoint such that the composition provides a fourth-order approximation.

The results are also valid for any quadrature rule just following the same changes as in Theorem 2.3, and they can also be written in terms of moment integrals. For example, using the Simpson rule the method reads

$$\begin{aligned}(q_{1/2}, p_{1/2}) &= \Phi\left(q_0, p_0, \frac{h}{2}, M_1\right), & M_1 &= \frac{1}{6}\left(3\mu(0) + 4\mu\left(\frac{h}{2}\right) - \mu(h)\right), \\ (q_1, p_1) &= \Phi\left(q_{1/2}, p_{1/2}, \frac{h}{2}, M_2\right), & M_2 &= \frac{1}{6}\left(-\mu(0) + 4\mu\left(\frac{h}{2}\right) + 3\mu(h)\right).\end{aligned}$$

**3.2. Tailored methods for particular problems.** As we have mentioned, working with the graded algebra allows us to easily identify in many classes of problems some additional simplifications or elements of the algebra that can be incorporated into the schemes to improve their performance. In this section we just present an example as an illustration.

Let us consider the nonautonomous Hamiltonian

$$H(t, q, p) = \frac{1}{2}p^T p + V(t, q)$$

with associated vector field from Hamilton equations

$$f(t, q, p) = (p, g(t, q))$$

with  $g(t, q) = -\nabla V(t, q)$ . If we denote by  $G_i(q)$ ,  $i = 1, \dots, s$ , the graded vector fields obtained from  $g(t, q)$  similarly to  $F_i(q, p)$  obtained from  $f(t, q, p)$ , we easily observe that

$$F_1 = (hp, G_1),$$

but

$$F_i = (0, G_i), \quad i > 1,$$

so  $F_i$  for  $i > 1$  depend only on coordinates, and their Lie brackets vanish; i.e.,

$$(F_i, F_j) = 0 \quad \text{if } i, j > 1.$$

For methods of order greater than 4, this implies, for example, that  $(F_2, F_3) = 0$ , which could allow us to obtain methods of order 5 or higher at a reduced cost.

In addition, in this case we have that

$$(F_j, (F_1, F_i)) = (F_i, (F_1, F_j)) = R_{i+j+1}(q) \quad \text{if } i, j > 1;$$

i.e., they are vector fields of order  $h^{i+j+1}$  that depend only on coordinates and could be incorporated into the numerical schemes. For example, these elements were incorporated into the numerical integration of the matrix Hill equation in [2] showing a high performance, and it is worth analyzing whether these methods can also be used as integrators for this nonautonomous problem since both share the same algebraic structure when building the numerical schemes.

**4. Linear systems.** Let us now consider homogeneous linear systems (1.3). We can apply all the schemes obtained to the general nonlinear case where we must take into account that

$$(f(\tau_1, x), f(\tau_2, x)) = (A(\tau_1)x, A(\tau_2)x) = [A(\tau_2), A(\tau_1)]x,$$

and we recover the well-known Magnus series expansion for linear problems as well as the Magnus integrators obtained in the literature [5, 6, 19, 20, 21].

On the other hand, taking into account that

$$\exp(hL_{f(\tau_i, x_0)})x_0 = e^{hA(\tau_i)}x_0, \quad i = 1, 2,$$

it follows that

$$\exp(hL_{f(\tau_1, x_0)})\exp(hL_{f(\tau_2, x_0)})x_0 = e^{hA(\tau_2)}e^{hA(\tau_1)}x_0,$$

i.e., the exponentials appear in the reverse order, and we recover the schemes referred to as commutator-free (quasi-)Magnus integrators [1, 7, 8, 11].

The order of accuracy of most methods from the literature when considering time-averaged methods using quadrature rules other than the GL quadrature rule, even when correct, has not been properly proven. The present paper, however, provides a proof that guarantees that the order of accuracy of all such methods is correct.

**5. Additional examples.** The goal of this section is to provide additional examples to show how to use the time-averaged techniques presented in this work in order to adapt different one-step methods involving the vector field evaluated at a given quadrature rule. We first review the steps to build time-averaged methods:

- Choose the *order* of the method,  $2s$ .
- Consider the graded Lie algebra with generators  $F_1, \dots, F_s$  (in terms of the GL quadrature rule), and write the *equation* in terms of these generators (the exact solution corresponds to an approximate solution up to order  $2s$ ).
- Write the chosen *numerical scheme* in terms of  $F_i$ .
- Take the *desired quadrature rule* or quadrature rules, and replace  $F_i$  by  $\widehat{F}_i$  through the changes indicated in Theorem 2.3. One can also write the scheme in terms of the moment integrals.
- Numerically *solve* the equations appearing in the scheme.

*Example 6.* Given the linear equation

$$(5.1) \quad x' = A(t)x, \quad x(0) = x_0,$$

where  $A(t)$  is known in an equidistant mesh, numerically solve it using the fourth-order implicit Runge–Kutta Gauss–Legendre (RKGL4) method (we take only the first time step where we have  $A_1 = A(0), A_2 = A(\frac{h}{2}), A_3 = A(h)$ ).

For this problem we have the following:

- $2s = 4$ , and we choose the Simpson quadrature rule.
- The graded matrices (equivalent to the  $F_1, F_2$  vector fields) are

$$B_1 = \frac{h}{6}(A_1 + 4A_2 + A_3), \quad B_2 = h(A_3 - A_1).$$

- The equation to be solved is

$$(5.2) \quad y' = \left(\frac{1}{h}B_1 + \frac{1}{h^2}B_2\tau\right)y, \quad y(0) = x_0,$$

and the RKGL4 method applied to solve this problem corresponds to

$$\begin{aligned}
 k_1 &= \frac{1}{h} \left( B_1 - \frac{\sqrt{3}}{6} B_2 \right) \left( x_0 + h \left( \frac{1}{4} k_1 + \frac{3 - 2\sqrt{3}}{12} k_2 \right) \right), \\
 k_2 &= \frac{1}{h} \left( B_1 + \frac{\sqrt{3}}{6} B_2 \right) \left( x_0 + h \left( \frac{3 + 2\sqrt{3}}{12} k_1 + \frac{1}{4} k_2 \right) \right), \\
 (5.3) \quad y_1 &= x_0 + \frac{h}{2} (k_1 + k_2).
 \end{aligned}$$

- One has to solve the implicit equations to get the numerical solution that corresponds to a fourth-order approximation.

If we consider (2.37), (2.38), we obtain

$$(5.4) \quad y' = \tilde{A}(t)y, \quad y(0) = x_0,$$

with  $\tilde{A}(t)$  the interpolating polynomial

$$\tilde{A}(t) = A_1 + (-3A_1 + 4A_2 + A_3) \frac{t}{h} + (2A_1 - 4A_2 + 2A_3) \frac{t^2}{h^2}.$$

Then, the Magnus expansion applied to this equation can contain more terms than necessary to build methods up to fourth order. In this particular case, however, the RKGL4 method applied to solve (5.4) also corresponds to the scheme (5.3).

We notice that the choice of the method to solve the problem is not constrained by the fact that the matrix  $A(t)$  is only known at some given instants. In addition, one could use a higher-order method to solve (5.2) that could provide more accurate results, e.g., when  $A(t)$  evolves smoothly with  $t$ . Obviously, asymptotically this would correspond to a fourth-order method due to the choice of the Simpson rule.

*Example 7.* Let us consider the two-dimensional nonautonomous Kepler problem (2.41) with  $\mu(t)$  a decreasing function and with initial conditions

$$q_0 = (1 - e, 0), \quad p_0 = (0, \sqrt{(1 + e)/(1 - e)})$$

along the time interval,  $t \in [0, 20]$ , using only compositions of the map (3.17).

For this purpose, we can consider the extended autonomous Hamiltonian

$$K(q_t, q, p_t, p) = p_t + \left( \frac{1}{2} p^T p - \mu(q_t) \frac{1}{r} \right) = X + Y$$

and solve it using a splitting method of order  $p$  with coefficients  $\{a_i, b_i\}_{i=1}^m$  such that

$$\prod_{i=1}^{m+1} e^{ha_i X} e^{hb_i Y} = e^{h(X+Y)} + \mathcal{O}(h^{p+1}),$$

with  $b_{m+1} = 0$  and  $X, Y$  two noncommuting operators.

If we split the system (1.2) as

$$(5.5) \quad \begin{pmatrix} x \\ t \end{pmatrix}' = \begin{pmatrix} f(t, x) \\ 0 \end{pmatrix} + \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

and solve it using the splitting method to advance from  $t_n$  to  $t_{n+1}$ , then it corresponds to solving the sequence of autonomous equations

$$(5.6) \quad \begin{aligned} y'_1 &= b_1 f(t_n + d_1 h, y_1), & y_1(0) &= x_n, \\ y'_2 &= b_2 f(t_n + d_2 h, y_2), & y_2(0) &= y_1(h), \\ &\vdots \\ y'_m &= b_m f(t_n + d_m h, y_m), & y_m(0) &= y_{m-1}(h), \end{aligned}$$

with  $d_i = \sum_{j=1}^i a_j$  and  $x_{n+1} = y_m(h)$ . Then, the algorithm applied to the nonautonomous Kepler problem corresponds to

$$(5.7) \quad \begin{aligned} Q_0 &= q_n, & P_0 &= p_n, \\ (Q_1, P_1) &= \Phi(Q_0, P_0, b_1 h, \mu(t_n + d_1 h)) \\ &\vdots \\ (Q_m, P_m) &= \Phi(Q_{m-1}, P_{m-1}, b_m h, \mu(t_n + d_m h)) \\ q_{n+1} &= Q_m, & p_{n+1} &= P_m, & t_{n+1} &= t_n + d_{m+1} h. \end{aligned}$$

For instance, the three-stage fourth-order scheme (frequently referred to as Yoshida’s fourth-order method) corresponds to  $m = 3$  and coefficients

$$a_1 = a_4 = \frac{1}{2(2 - 2^{1/3})}, \quad a_2 = a_3 = 1/2 - a_1, \quad b_1 = b_3 = 2a_1, \quad b_2 = 1 - 2b_1.$$

Notice that  $a_2, b_2 < 0$  and this implies that the second map of the algorithm corresponds to a backward integration with an increase of the mass. This backward integration can be problematic for stiff or nonreversible problems. This trouble as well as the performance of this method can be improved using the optimized six-stage fourth-order method from [10] with coefficients

$$\begin{aligned} a_1 &= 0.0792036964311956, & a_2 &= 0.3531729060497740, & a_3 &= -0.0420650803577195, \\ a_4 &= 1 - 2(a_1 + a_2 + a_3), & a_5 &= a_3, a_6 = a_2, a_7 = a_1, & b_1 &= 0.2095151066133620, \\ b_2 &= -0.1438517731798181, & b_3 &= 1/2 - (b_1 + b_2), & b_4 &= b_3, b_5 = b_2, b_6 = b_1, \end{aligned}$$

where now  $d_i > 0$ ,  $i = 1, \dots, 7$ , but  $b_2, b_5 < 0$  and still involves backward time integrations.

On the other hand, the new fourth-order time-averaged scheme (3.19), (3.20) requires only two forward compositions.

We take the initial conditions with  $e = 1/2$  and the following decreasing function for the time-dependent mass:

$$(5.8) \quad \mu(t) = 1 + \exp\left(-\frac{1}{5}\left(t + \frac{1}{4}\sin^2(4t)\right)\right),$$

where the solution in positions at  $t = 20$  is

$$q(20) = (0.108926658115, 0.735820545363).$$

We integrate the system using these three methods, and as a reference, we also show the results obtained using the second-order midpoint method (3.18). We measure the error at the end of the integration versus the number of times the map  $\Phi$  is evaluated for different choices of the time step. The results are shown in Figure 5.1, where we observe that the new time-averaged commutator-free method shows the best results for all accuracies.

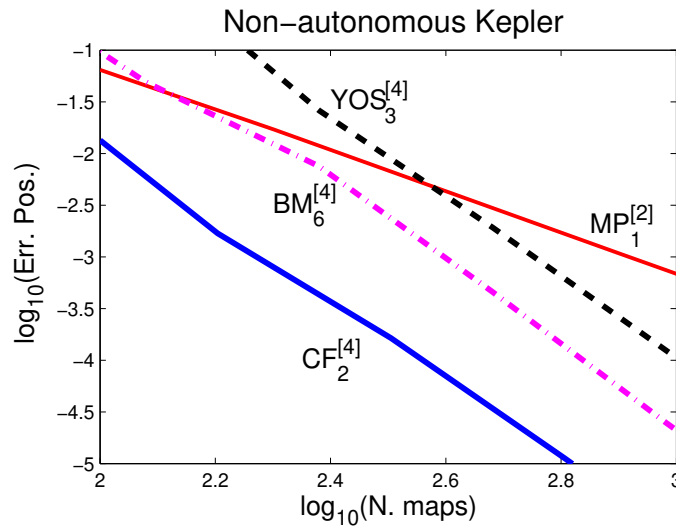


FIG. 5.1. Error in positions at  $t = 20$  versus the number of maps for different choices of the time step when integrating the nonautonomous Kepler problem with time-dependent mass given by (5.8) and taking  $e = 1/2$  for the choice of the initial conditions:  $MP_1^{[2]}$ , the one-map second-order midpoint method;  $YOS_3^{[4]}$ , the three-map fourth-order Yoshida's method;  $BM_6^{[4]}$ , the six-map fourth-order method from [10]; and  $CF_2^{[4]}$ , the two-map time-averaged (commutator-free) fourth-order method (3.19), (3.20).

**6. Conclusions.** In this work we have shown how to numerically integrate nonautonomous differential equations by solving alternate time-averaged differential equations where the vector field is evaluated at the nodes of a previously chosen quadrature rule. If the chosen quadrature rule is of order  $2s$  or higher, for  $s = 1, 2, \dots$ , we have shown how to build a differential equation with a new vector field that is only a polynomial function of degree  $s - 1$  in the independent variable,  $t$ , and whose solution after one time step agrees with the solution of the original differential equation up to order  $2s$ . This result is irrespective of the number of nodes of the quadrature rule. Then, any numerical scheme can be used to solve this alternate equation, and the vector field need only be evaluated at the chosen quadrature rule.

We have shown how to use the Magnus series expansion to build the formal solution in terms of a graded Lie algebra. This can be considered as the formal solution and allows us to develop new numerical schemes adapted for different classes of problems. For instance, the commutator-free (quasi-)Magnus integrators, obtained for linear systems and adapted in this work to nonlinear systems, can lead to very efficient schemes. All methods can also be written in terms of moment integrals, and each integral can be computed using different quadrature rules. We have also shown how the new methods tailored for different linear systems can also be used for solving nonlinear systems when they share similar properties for their associated Lie algebras.

**Acknowledgments.** I would like to thank the referees for their constructive suggestions that helped to improve the presentation of this work. I would also like to thank Ernst Hairer for suggesting the alternate proof of Theorem 2.2 without numerical integration methods.

## REFERENCES

- [1] A. ALVERMANN AND H. FEHSKE, *High-order commutator-free exponential time-propagation of driven quantum systems*, J. Comput. Phys., 230 (2011), pp. 5930–5956.
- [2] P. BADER, S. BLANES, E. PONSODA, AND M. SEYDAOGLU, *Symplectic integrators for the matrix Hill's equation and its applications to engineering models*, J. Comput. Appl. Math., 316 (2017), pp. 47–59.
- [3] S. BLANES AND F. CASAS, *A Concise Introduction to Geometric Numerical Integration*, CRC Press, Boca Raton, FL, 2016.
- [4] S. BLANES, F. CASAS, P. CHARTIER, AND A. MURUA, *Optimized high-order splitting methods for some classes of parabolic equations*, Math. Comp., 82 (2013), pp. 1559–1576.
- [5] S. BLANES, F. CASAS, J.A. OTEO, AND J. ROS, *The Magnus expansion and some of its applications*, Phys. Rep., 470 (2009), pp. 151–238.
- [6] S. BLANES, F. CASAS, AND J. ROS, *Improved high order integrators based on Magnus expansion*, BIT, 40 (2000), pp. 434–450.
- [7] S. BLANES, F. CASAS, AND M. THALHAMMER, *High-order commutator-free quasi-Magnus exponential integrators for non-autonomous linear evolution equations*, Comput. Phys. Commun., 220 (2017), pp. 243–262.
- [8] S. BLANES AND P.C. MOAN, *Splitting methods for the time-dependent Schrödinger equation*, Phys. Lett. A, 265 (2000), pp. 35–42.
- [9] S. BLANES AND P.C. MOAN, *Splitting methods for non-autonomous differential equations*, J. Comput. Phys., 170 (2001), pp. 205–230.
- [10] S. BLANES AND P.C. MOAN, *Practical symplectic partitioned Runge-Kutta and Runge-Kutta-Nyström methods*, J. Comput. Appl. Math., 142 (2002), pp. 313–330.
- [11] S. BLANES AND P.C. MOAN, *Fourth- and sixth-order commutator-free Magnus integrators for linear and non-linear dynamical systems*, Appl. Numer. Math., 56 (2006), pp. 1519–1537.
- [12] F. CASTELLA, P. CHARTIER, S. DESCOMBES, AND G. VILMART, *Splitting methods with complex times for parabolic equations*, BIT, 49 (2009), pp. 487–508.
- [13] P. CHARTIER, E. HAIRER, AND G. VILMART, *Numerical integrators based on modified differential equations*, Math. Comp., 76 (2007), pp. 1941–1953.
- [14] E. HAIRER, C. LUBICH, AND G. WANNER, *Geometric Numerical Integration. Structure-Preserving Algorithms for Ordinary Differential Equations*, 2nd ed., Springer Ser. Comput. Math. 31, Springer-Verlag, 2006.
- [15] E. HAIRER, S.P. NØRSETT, AND G. WANNER, *Solving Ordinary Differential Equations I, Nonstiff Problems*, 2nd revised ed., Springer-Verlag, 1993.
- [16] E. HANSEN AND A. OSTERMANN, *High order splitting methods for analytic semigroups exist*, BIT, 49 (2009), pp. 527–542.
- [17] A. ISERLES, *A First Course in the Numerical Analysis of Differential Equations*, 2nd ed., Cambridge University Press, Cambridge, UK, 2008.
- [18] A. ISERLES, A. MARTINSEN, AND S.P. NØRSETT, *On the implementation of the method of Magnus series for linear differential equations*, BIT, 39 (1999), pp. 281–304.
- [19] A. ISERLES, H.Z. MUNTHER-KAAS, S.P. NØRSETT, AND A. ZANNA, *Lie group methods*, Acta Numer., 9 (2000), pp. 215–365.
- [20] A. ISERLES AND S.P. NØRSETT, *On the solution of linear differential equations in Lie groups*, R. Soc. Lond. Philos. Trans. Ser. A Math. Phys. Eng. Sci., 357 (1999), pp. 983–1019.
- [21] A. ISERLES, S.P. NØRSETT, AND A.F. RASMUSSEN, *Time-symmetry and high-order Magnus methods*, Appl. Numer. Math., 39 (2001), pp. 379–401.
- [22] B. LEIMKÜHLER AND S. REICH, *Simulating Hamiltonian Dynamics*, Cambridge University Press, Cambridge, UK, 2004.
- [23] W. MAGNUS, *On the exponential solution of differential equations for a linear operator*, Comm. Pure Appl. Math., 7 (1954), pp. 649–673.
- [24] P.C. MOAN, *Efficient Approximation of Sturm–Liouville Problems Using Lie-Group Methods*, Technical Report 1998/NA11, Department of Applied Mathematics and Theoretical Physics, University of Cambridge, Cambridge, UK, 1998.
- [25] J.A. OTEO AND J. ROS, *The Magnus expansion for classical Hamiltonian systems*, J. Phys. A, 24 (1991), pp. 5751–5762.
- [26] H. MUNTHER-KAAS AND B. OWREN, *Computations in a free Lie algebra*, R. Soc. Lond. Philos. Trans. Ser. A Math. Phys. Eng. Sci., 357 (1999), pp. 957–981.

- [27] J.M. SANZ-SERNA AND M.P. CALVO, *Numerical Hamiltonian Problems*, Chapman & Hall, London, 1994.
- [28] M. THALHAMMER, *A fourth-order commutator-free exponential integrator for nonautonomous differential equations*, SIAM J. Numer. Anal., 44 (2006), pp. 851–864, <https://doi.org/10.1137/05063042>.
- [29] A. ZANNA, *Collocation and relaxed collocation for the Fer and the Magnus expansions*, SIAM J. Numer. Anal., 36 (1999), pp. 1145–1182, <https://doi.org/10.1137/S0036142997326616>.