

Splitting methods in the numerical integration of non-autonomous dynamical systems

Sergio Blanes · Fernando Casas · Ander Murua

Received: 11 February 2011 / Accepted: 28 February 2011 / Published online: 20 March 2011
© Springer-Verlag 2011

Abstract We present a procedure leading to efficient splitting schemes for the time integration of explicitly time dependent partitioned linear differential equations arising when certain partial differential equations are previously discretized in space. In the first stage we analyze the order conditions of the corresponding autonomous problem and construct new 6th-order methods. In the second stage, by following a procedure previously designed by the authors, we generalize the methods to the time dependent case in such a way that no order reduction is present. The resulting schemes compare favorably with other integrators previously available.

Keywords Splitting methods · Time-dependent problems · Geometric integrators

Mathematics Subject Classification (2000) 65L05 · 65P10 · 65Z05

1 Introduction

The evolution of many physical systems is usually described by an ordinary differential equation (ODE)

$$x' = f(x), \quad x(t_0) = x_0 \in \mathbb{R}^d, \quad (1)$$

S. Blanes
Instituto de Matemática Multidisciplinar, Universitat Politècnica de València,
46022 Valencia, Spain
e-mail: serblaza@imm.upv.es

F. Casas (✉)
Departament de Matemàtiques, Institut de Matemàtiques i Aplicacions de Castelló,
Universitat Jaume I, 12071 Castellón, Spain
e-mail: Fernando.Casas@uji.es

A. Murua
Konputazio Zientziak eta A.A. saila, Informatika Fakultatea,
EHU/UPV, Donostia/San Sebastián, Spain
e-mail: Ander.Murua@ehu.es

whose formal solution can be written as

$$x(t) = \varphi_t(x_0) = \exp((t - t_0) D_f)x_0.$$

Here D_f stands for the Lie derivative associated with $f(x)$, i.e. $D_f \equiv f(x) \cdot \nabla$. Describing the physical problem at hand requires then to formulate an appropriate mathematical model in this setting. In other words, a suitable function $f(x)$ such that (1) can reproduce the most salient features of the real system. In this respect, since the real system often involves one or more symmetries and these symmetries can be mathematically formulated in terms of Lie groups, a necessary condition is that the Lie derivative D_f possesses the appropriate Lie algebraic structure.

Of course, one is also interested in solving (1), but with the exception of a very few simple cases, only numerical approximations are usually obtained. Generally speaking, standard numerical integrators (Runge–Kutta formulae, linear multistep methods) produce approximate solutions that do not take into account the special algebraic structure of f and therefore do not preserve the corresponding symmetries. In consequence, much effort has been devoted during the last two decades to the design of numerical integrators preserving those qualitative (geometric) properties of the exact solution. Examples of such algorithms include symplectic integrators, volume preserving methods, Lie group integrators, variational methods in mechanics, etc. [13, 14, 20]. All of them are now put into the more general category of *geometric numerical integrators*. In geometric integration, in fact, it is crucial to identify significant (geometric) properties of the dynamical system (1) and construct numerical integration algorithms that preserve those features. In addition, of course, one is interested in building efficient methods with the usual properties of accuracy and stability.

Although research in geometric numerical integrators for differential equations has experienced a tremendous boost during the last decades, it is fair to say that this has been mainly restricted to autonomous problems, whereas nonlinear systems of the form

$$x' = f(x, t), \quad x(t_0) = x_0 \in \mathbb{R}^d, \tag{2}$$

i.e., when time appears explicitly in the formulation of the problem, have been up to some point disregarded. In the linear case $X' = A(t)X$, with A and X $n \times n$ matrices, several options have been widely explored and indeed the Magnus expansion has shown to be an extremely useful device to get analytical as well as numerical approximations [7].

A usual procedure in the numerical analysis of explicitly time-dependent problems consists in transforming (2) into an autonomous differential equation by the introduction of a new variable,

$$\begin{cases} x' = f(x, x_t) \\ x'_t = 1 \end{cases} \tag{3}$$

or, equivalently,

$$y' = F(y), \quad y(t_0) = (x_0, t_0) \in \mathbb{R}^{d+1}, \tag{4}$$

and $F(y) = (f(x, x_t), 1)$. Notice that formulation (3) introduces the auxiliary variable x_t aimed at eliminating the explicit time dependence, so that numerical integrators designed for (1) can, in principle, be used in this setting. This process exhibits, nevertheless, several drawbacks. First, the algebraic structure of D_f and D_F may differ, so that methods specifically designed for (1) cannot simply be used for the integration of the new enlarged system (4). Second, even when they can be applied, very often their efficiency reduces considerably. For this reason, during the last years, the authors have analyzed different procedures

to adapt effective splitting schemes when the system is explicitly time-dependent without losing their efficiency (see, e.g., [1,3,9]). In the particular but important case of the linear differential equations arising from the space discretization of the Schrödinger equation, it is shown in [3] how to generalize methods previously designed in [11] when an explicit time dependency is introduced. The results achieved in [3] motivates the search of new and more powerful integration methods for time independent systems especially designed to be used in the non-autonomous case. The goal of the present paper is precisely this: to carry out a systematic analysis of the order conditions to be satisfied by splitting methods when integrating partitioned linear systems to which the techniques exposed in [3] are subsequently applied to render schemes also valid in the time dependent case.

The plan of the paper is as follows. In Sect. 2 we briefly review some of the available techniques to adapt splitting methods designed for autonomous problems to the explicitly time dependent case. In Sect. 3 we focus the treatment on partitioned linear systems of differential equations arising, in particular, in the time integration of the Schrödinger equation previously discretized in space. There we present a procedure to get and solve the resulting order conditions for methods of order ≤ 6 to which the approach presented in [3] is applied to construct efficient splitting schemes in the time dependent case. The validity of the treatment is then illustrated in Sect. 4, where the performance of a new 6th-order method constructed along these lines is compared with other standard integrators.

2 Splitting methods and their generalization to non-autonomous systems

2.1 General treatment

Although splitting methods have been used for a long time in the numerical treatment of differential equations, they have experienced a revival with the advent of geometric integration. In fact, a good deal of geometric integrators are based on the idea of splitting. The idea is fairly simple: suppose f in Eq. (1) can be decomposed as $f(x) = f^{[A]}(x) + f^{[B]}(x)$ in such a way that systems

$$x' = f^{[A]}(x), \quad x' = f^{[B]}(x) \tag{5}$$

can either be solved in closed form or accurately integrated. Then one combines these partial solutions into an approximate solution for (1), often of high accuracy. To make this sentence precise, let us denote by

$$\varphi_t^{[A]} = \exp((t - t_0)D_A), \quad \varphi_t^{[B]} = \exp((t - t_0)D_B)$$

the flows corresponding to Eqs. (5), where D_A and D_B represent the Lie derivatives associated with $f^{[A]}(x)$ and $f^{[B]}(x)$, respectively. Then one considers the composition

$$\psi_h \equiv \varphi_{a_{m+1}h}^{[A]} \circ \varphi_{b_mh}^{[B]} \circ \varphi_{a_mh}^{[A]} \circ \varphi_{b_{m-1}h}^{[B]} \circ \dots \circ \varphi_{b_1h}^{[B]} \circ \varphi_{a_1h}^{[A]} \tag{6}$$

with appropriately chosen real coefficients a_i, b_i to ensure that ψ_h is an approximation to the exact solution φ_h up to order $\mathcal{O}(h^p)$ with respect to the time step h , i.e. $\psi_h = \varphi_h + \mathcal{O}(h^{p+1})$. A great deal of methods of this class exists in the literature, of different orders and tailored for different structures of the vector field: general separable problems, systems arising from second order differential equations $x'' = g(x)$, near-integrable systems, etc. (see [5,13,15,17] and references therein). One could say that the performance of the different splitting methods strongly depends on the particular problem at hand, so that a previous analysis is highly

recommended to select the most appropriate scheme for its numerical treatment [5]. The non-autonomous separable problem

$$x' = f^{[A]}(x, t) + f^{[B]}(x, t) \tag{7}$$

is indeed a case in point.

One might think of two procedures to adapt scheme (6) in this setting. The first one consists in replacing the maps $\varphi_{a_i h}^{[A]}, \varphi_{b_i h}^{[B]}$ by the maps associated to the exact flow defined by the equations

$$x' = f^{[A]}(x, t), \quad t \in [t_0 + c_i h, t_0 + (c_i + a_i)h] \tag{8}$$

$$x' = f^{[B]}(x, t), \quad t \in [t_0 + d_i h, t_0 + (d_i + b_i)h]. \tag{9}$$

Here $c_i = \sum_{j=0}^{i-1} a_j, d_i = \sum_{j=0}^{i-1} b_j, a_0 = 0, b_0 = 0$, and the initial conditions are given by the solution obtained from the previous flow. This approach can be considered as a time-average on each stage of the composition. Obviously, obtaining the exact solution of the non-autonomous equations (8) and (9) is by no means trivial due to the explicit time-dependence. In any case, the formal solution can be obtained by using the Magnus expansion, as shown in [7].

The second procedure is perhaps simpler. It consists in taking the maps $\varphi_{a_i h}^{[A]}, \varphi_{b_i h}^{[B]}$ in (6) as the $(a_i h)$ -flow and $(b_i h)$ -flow associated respectively to the autonomous equations

$$x' = f^{[A]}(x, t_0 + d_i h), \quad t \in [t_0 + c_i h, t_0 + (c_i + a_i)h] \tag{10}$$

$$x' = f^{[B]}(x, t_0 + c_i h), \quad t \in [t_0 + d_i h, t_0 + (d_i + b_i)h]. \tag{11}$$

Notice that the coefficients c_i, d_i appear interchanged in the vector fields with respect to (8) and (9).

These two strategies, which could be dubbed ‘averaging’ and ‘frozen’ techniques, respectively, may differ considerably both in the accuracy reached by the methods and also in their computational cost. Let us illustrate them on a simple but important example arising in applications.

2.2 Linear non-autonomous separable systems

When discretizing in space the time dependent Schrödinger equation involving a time dependent potential $V(t)$ (for instance, with a pseudospectral method), the following system of ODEs arises:

$$iu' = H(t)u, \tag{12}$$

where $u \in \mathbb{C}^N$ and H is a real symmetric matrix. If the real and imaginary parts in u are considered, $u = q + ip$, the N -dimensional linear complex system (12) can be written as the $2N$ -dimensional real system

$$q' = H(t)p, \quad p' = -H(t)q. \tag{13}$$

These, in fact, can be interpreted as the classical Hamilton equations corresponding to the Hamiltonian

$$\mathcal{H}(q, p, t) = \frac{1}{2}p^T H(t)p + \frac{1}{2}q^T H(t)q. \tag{14}$$

Although we limit ourselves to this problem, most of the discussion is also valid with minor modifications for the more general system

$$q' = M(t)p, \quad p' = N(t)q, \tag{15}$$

for $q \in \mathbb{R}^{d_1}$, $p \in \mathbb{R}^{d_2}$. Equation (15) arises, in particular, when the Maxwell equations are discretized in space [19].

Equations (13) can be written in the compact form

$$z' = (A(t) + B(t))z, \tag{16}$$

where $z = (q, p)^T$ and

$$A(t) = \begin{pmatrix} 0 & H(t) \\ 0 & 0 \end{pmatrix}, \quad B(t) = \begin{pmatrix} 0 & 0 \\ -H(t) & 0 \end{pmatrix}. \tag{17}$$

Let us consider first the autonomous problem, i.e., when H does not depend explicitly on t , in which case the corresponding equations

$$q' = Hp, \quad p' = -Hq \tag{18}$$

possess the formal solution

$$z(t) = e^{t(A+B)}z(0). \tag{19}$$

Typically, as a result of the discretization in space, the value of N is large, and thus the exact computation of the exponential is exceedingly costly. In consequence, it makes sense to construct approximations requiring a much reduced computational effort. This can be achieved when the scheme only involves the computation of Hq and Hp in a particular sequence and only a few times per step. But this is precisely what splitting methods effectively do, as it is evident if one writes the composition (6) for this particular problem:

$$\begin{aligned} K(h) &\equiv e^{ha_{m+1}A} e^{hb_m B} e^{ha_m A} \dots e^{hb_1 B} e^{ha_1 A} \\ &= \begin{pmatrix} I & ha_{m+1}H \\ 0 & I \end{pmatrix} \begin{pmatrix} I & 0 \\ -hb_m H & I \end{pmatrix} \dots \begin{pmatrix} I & 0 \\ -hb_1 H & I \end{pmatrix} \begin{pmatrix} I & ha_1 H \\ 0 & I \end{pmatrix}. \end{aligned} \tag{20}$$

The scheme requires m matrix-vector products Hq and Hp (the last product at each step can be reused in the first stage at the following step) and is referred as an m -stage method. Although any of the splitting methods for separable systems collected in [13, 15, 17, 18] can be used for carrying out numerical integrations here, Eq. (16) in the autonomous case possesses the following crucial simplifying property:

$$[A, [A, [A, B]]] = [B, [B, [B, A]]] = 0, \tag{21}$$

where $[\cdot, \cdot]$ stands for the usual commutator: $[A, B] = AB - BA$. This feature allows one to build very efficient methods indeed [2, 3, 11]. Moreover, it has been shown that any non-symmetric method for this problem is conjugate to a symmetric method [4], so that one may restrict the analysis to symmetric compositions, i.e., when $a_{m+2-i} = a_i$, $b_{m+1-i} = b_i$. The resulting scheme is sometimes referred to as an ABA composition. Since the role of A and B can be naturally interchanged, BAB compositions are not separately studied.

Let us turn our attention now to the time dependent case. As is well known, the Magnus expansion allows one to formally write the solution of (16) as

$$z(t) = e^{\Omega(t)}z(0), \tag{22}$$

where $\Omega(t)$ is given by an infinite series involving $A(t)$, $B(t)$, multivariate integrals and nested commutators with a finite radius of convergence [7, 16]. Although it is indeed possible to derive numerical integration algorithms from the Magnus expansion [8], they still require the computation of the exponential of a full matrix of high dimension involving iterate integrals and commutators. We apply instead the two procedures pointed out in Sect. 2.1.

The first one (the ‘averaging’ technique) uses composition (6) with maps corresponding to the exact solutions of (8)–(9), which for this particular problem read (taking $t_0 = 0$)

$$\begin{aligned} \varphi_{a_i h}^{[A_1]}(q, p) &= \left(q + \int_{t+c_i h}^{t+c_{i+1} h} H(\tau) d\tau p, p \right) \\ \varphi_{b_i h}^{[B_1]}(q, p) &= \left(q, p - \int_{t+d_i h}^{t+d_{i+1} h} H(\tau) d\tau q \right). \end{aligned} \quad (23)$$

Notice that the resulting scheme can be seen as the composition method (6) applied to the autonomous Hamiltonian

$$\mathcal{H}(q, q_1, q_2, p, p_1, p_2) = \left(\frac{1}{2} p^T H(q_1) p + p_1 \right) + \left(\frac{1}{2} q^T H(q_2) q + p_2 \right) \equiv A_1 + B_1, \quad (24)$$

where we have considered time as *two* different additional coordinates.

On the other hand, with the ‘frozen’ technique (10) and (11) one has

$$\varphi_{a_i h}^{[A_2]}(q, p) = (q + H(d_i h) p, p), \quad \varphi_{b_i h}^{[B_2]}(q, p) = (q, p - H(c_i h) q), \quad (25)$$

and the corresponding scheme, as before, is nothing but composition (6) applied to the Hamiltonian

$$\mathcal{H}(q, q_1, q_2, p, p_1, p_2) = \left(\frac{1}{2} p^T H(q_2) p + p_1 \right) + \left(\frac{1}{2} q^T H(q_1) q + p_2 \right) \equiv A_2 + B_2. \quad (26)$$

Now the Hamilton equations are no longer linear and, moreover,

$$\{A_i, \{A_i, \{A_i, B_i\}\}\} \neq 0, \quad \{B_i, \{B_i, \{B_i, A_i\}\}\} \neq 0, \quad i = 1, 2$$

in terms of the Poisson bracket. In consequence, the highly efficient schemes of type (20) designed for systems verifying (21) lose their appealing accomplishments when applied in the non-autonomous case.

Another possibility is suggested in [9]. One might consider a combination of (24) and (26) in the form

$$\mathcal{H}(q, q_1, q_2, p, p_1, p_2) = \left(\frac{1}{2} p^T H(q_1) p + p_1 \right) + \left(\frac{1}{2} q^T H(q_1) q + p_2 \right) \quad (27)$$

or equivalently

$$\mathcal{H}(q, q_1, p, p_1) = \left(\frac{1}{2} p^T H(q_1) p + p_1 \right) + \left(\frac{1}{2} q^T H(q_1) q \right) \equiv A_3 + B_3, \quad (28)$$

with associated maps

$$\varphi_{a_i h}^{[A_3]}(q, p) = \left(q + \int_{t+c_i h}^{t+c_{i+1} h} H(\tau) d\tau p, p \right), \quad \varphi_{b_i h}^{[B_3]}(q, p) = (q, p - H(c_i h)q). \quad (29)$$

Now $\{B_3, \{B_3, \{B_3, A_3\}\}\} = 0$ so that Runge–Kutta–Nyström methods can be used (even with modified potentials) [5], and thus significant improvements in the efficiency with respect to the previous choices can be achieved.

3 A new class of splitting methods for non-autonomous linear systems

Our purpose here is to generalize the ‘averaging’ technique (23) by using the Magnus expansion and formulate directly splitting methods of the form (20) in terms of the new maps in such a way that the resulting schemes do not suffer from a degradation in their performance. More specifically, the new methods have the form (for a time step of size h)

$$z(t+h) \approx e^{\tilde{A}_{m+1}} e^{\tilde{B}_m} e^{\tilde{A}_m} \dots e^{\tilde{B}_1} e^{\tilde{A}_1} z(t), \quad (30)$$

where the matrices $\tilde{A}_i \equiv A_i(t, h)$ and $\tilde{B}_i \equiv B_i(t, h)$ are taken as

$$\tilde{A}_i = h \int_{-1/2}^{1/2} p_i^a(\tau) A(t_{1/2} + h\tau) d\tau, \quad \tilde{B}_i = h \int_{-1/2}^{1/2} p_i^b(\tau) B(t_{1/2} + h\tau) d\tau. \quad (31)$$

Here $t_{1/2} = t + \frac{h}{2}$ and $p_i^a(\tau), p_i^b(\tau)$ are *filters* (scalar functions). As we will see in the sequel, to get integrators of even order $n = 2s$, it is enough to consider p_i^a, p_i^b as polynomials of degree $s - 1$, i.e.,

$$p_i^a(\tau) = a_i^{[0]} + a_i^{[1]}\tau + \dots + a_i^{[s-1]}\tau^{s-1}, \quad p_i^b(\tau) = b_i^{[0]} + b_i^{[1]}\tau + \dots + b_i^{[s-1]}\tau^{s-1}.$$

Since in our case

$$e^{\tilde{A}_i} = \begin{pmatrix} I & \tilde{H}_i^A \\ 0 & I \end{pmatrix}, \quad e^{\tilde{B}_i} = \begin{pmatrix} I & 0 \\ -\tilde{H}_i^B & I \end{pmatrix},$$

then (30) can be written as $z(t+h) \approx K(t, h)z(t)$, where

$$\begin{aligned} K(t, h) &= e^{\tilde{A}_{m+1}} e^{\tilde{B}_m} e^{\tilde{A}_m} \dots e^{\tilde{B}_1} e^{\tilde{A}_1} \\ &= \begin{pmatrix} I & \tilde{H}_{m+1}^A \\ 0 & I \end{pmatrix} \begin{pmatrix} I & 0 \\ -\tilde{H}_m^B & I \end{pmatrix} \begin{pmatrix} I & \tilde{H}_m^A \\ 0 & I \end{pmatrix} \dots \begin{pmatrix} I & 0 \\ -\tilde{H}_1^B & I \end{pmatrix} \begin{pmatrix} I & \tilde{H}_1^A \\ 0 & I \end{pmatrix}, \end{aligned} \quad (32)$$

and

$$\tilde{H}_i^A = h \int_{-1/2}^{1/2} p_i^a(\tau) H(t_{1/2} + h\tau) d\tau, \quad \tilde{H}_i^B = h \int_{-1/2}^{1/2} p_i^b(\tau) H(t_{1/2} + h\tau) d\tau. \quad (33)$$

These integrals can either be computed analytically or numerically approximated by using some appropriate quadrature rule. If the method (32) is of order $2s$, then a quadrature rule of order $2s$ or higher must be used to retain the original order.

The numerical scheme is then determined by the values of the coefficients $a_i^{[k]}, b_i^{[k]}$ ($k = 0, 1, \dots, s - 1, i = 1, \dots, m$). The problem of designing efficient methods of a prescribed

order $2s$ is then equivalent to determining coefficients such that the composition (30) achieves the desired order of accuracy and, at a given cost, provides the most accurate results among a number of possible choices. The method is of order $2s$ if the coefficients $a_i^{[k]}, b_i^{[k]}$ satisfy a system of algebraic equations (the order conditions), which have to be first formulated and then solved (usually by numerical tools). A subset of such order conditions corresponds precisely to the particular case where $H(t)$ actually does not depend on t , so that the matrix (32) reduces to the matrix $K(h)$ in (20) with a_i, b_i related to $a_i^{[k]}, b_i^{[k]}$. When constructing a method of order $2s$, we proceed as follows: (i) we first determine the values of the coefficients a_i, b_i in such a way that (20) gives a good method of order $2s$ for the autonomous case, (ii) and then choose the coefficients $a_i^{[k]}, b_i^{[k]}$, so that the remaining order conditions hold (once the relation between the coefficients a_i, b_i and $a_i^{[k]}, b_i^{[k]}$ is taken into account).

3.1 Order conditions for the autonomous case

When considering the matrix (20) used to propagate the numerical solution in the autonomous case (18), one observes that diagonalizing the matrix H with an appropriate linear change of variables transforms the system into N uncoupled harmonic oscillators with frequencies $\omega_1, \dots, \omega_N$. Although in practice one wants to avoid diagonalizing the matrix H , numerically solving the system (18) by a splitting method is mathematically equivalent to applying the splitting method to each of such harmonic oscillators (and then rewriting the result in the original variables). Clearly, the exact solution of each individual harmonic oscillator with frequency ω is propagated by the 2×2 matrix $O(\omega h)$, where

$$O(x) = \begin{pmatrix} \cos x & \sin x \\ -\sin x & \cos x \end{pmatrix}. \tag{34}$$

As for the numerical solution, it is propagated by a matrix $K(\omega h)$ defined as

$$K(x) = \begin{pmatrix} 1 & a_{m+1}x \\ 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 \\ -b_mx & 1 \end{pmatrix} \begin{pmatrix} 1 & a_mx \\ 0 & 1 \end{pmatrix} \cdots \begin{pmatrix} 1 & 0 \\ -b_1x & 1 \end{pmatrix} \begin{pmatrix} 1 & a_1x \\ 0 & 1 \end{pmatrix}.$$

It is straightforward to check that

$$K(x) = \begin{pmatrix} K_1(x) & K_2(x) \\ K_3(x) & K_4(x) \end{pmatrix} = \begin{pmatrix} 1 + \sum_{i=1}^m k_{1,i}x^{2i} & \sum_{i=1}^{m+1} k_{2,i}x^{2i-1} \\ \sum_{i=1}^m k_{3,i}x^{2i-1} & 1 + \sum_{i=1}^m k_{4,i}x^{2i} \end{pmatrix}, \tag{35}$$

where $k_{i,j}$ are homogeneous polynomials in the parameters a_i, b_i .

The integrator (35) typically will be stable for $|h\omega| < x_*$ for some value x_* (that we call stability threshold) depending on the coefficients a_i, b_i . It has been shown [4] that if for a given splitting method $x_* > 0$, then the method applied to (18), for H a constant matrix, is conjugate, for $h\rho(H) < x_*$, to the solution of a modified system

$$q' = \tilde{H}(h)p, \quad p' = -\tilde{H}(h)q, \tag{36}$$

where

$$h\tilde{H}(h) = hH + \phi_{2n+1}(hH)^{2n+1} + \phi_{2n+3}(hH)^{2n+3} + \dots \tag{37}$$

for some constants $\phi_{2i+1}, i = 1, 2, \dots$, provided that the method is of order $2n$ for the harmonic oscillator (see [4] for more details). Here $\rho(H)$ denotes the spectral radius of H .

We thus intend to construct accurate symmetric schemes with large stability intervals $(-x_*, x_*)$. Notice that for a fair comparison of the stability interval for splitting methods

with different number of stages, one must consider the relative stability threshold x_*/m . For this class of schemes, the elements of the stability matrix have to satisfy

$$K_1(x) = K_4(x) \approx \cos x, \tag{38}$$

$$K_2(x) \approx \sin x, \quad K_3(x) \approx -\sin x, \tag{39}$$

$$K_1(x)^2 - K_2(x)K_3(x) = 1. \tag{40}$$

Since we are dealing with symmetric compositions, we found more appropriate (due to the ill conditioned equations to be numerically solved) to consider the decomposition

$$K(x) = U(-x)^{-1}U(x), \quad \text{where } U(x) = \begin{pmatrix} U_1(x) & U_2(x) \\ U_3(x) & U_4(x) \end{pmatrix} \tag{41}$$

with U_1, U_4 even polynomial functions and U_2, U_3 odd polynomial functions. Since we are interested in matrices K, U to be decomposed as products (35), then they must satisfy $\det(K) = \det(U) = 1$. Clearly, conditions (38)–(40) are equivalent to $O(x) \approx U(-x)^{-1}U(x)$ together with

$$U_1(x)U_4(x) - U_2(x)U_3(x) \equiv 1. \tag{42}$$

Whence, one has an approximation of order $2s$ if

$$\begin{aligned} U_3(x) \cos(x/2) + U_4(x) \sin(x/2) &= \mathcal{O}(x^{2n+1}) \\ U_1(x) \sin(x/2) - U_2(x) \cos(x/2) &= \mathcal{O}(x^{2n+1}). \end{aligned} \tag{43}$$

Observe that to obtain a method of order $2n$ one needs a composition with $m \geq 2n - 1$ stages, as already noticed in [11]. The matrix U has $4n - 2$ parameters that can be used to solve the required system of $4n - 2$ equations: indeed, equations (43) originate $2n$ linear equations and condition (42) gives $2n - 2$ quadratic equations. However, the methods with minimal number of stages obtained by solving these $4n - 2$ equations have stability thresholds $x_* \approx \pi$, and thus the relative stability threshold $x_*/m \approx \pi/(4n - 2)$ becomes very small for high order methods.

A simple trick to get methods with larger relative stability threshold is to add the condition

$$O(j\pi) = K(j\pi), \quad j = 1, \dots, l \tag{44}$$

for some positive integer l . For moderate values of l relative to s , this typically gives a method with stability threshold $x_* > l\pi$. In addition to improving stability, due to its interpolatory nature, condition (44) contributes to improve also the precision of the method when applied to (18).

In terms of the matrix $U(x)$, condition (44) reads

$$U_1((2j - 1)\pi) = U_4((2j - 1)\pi) = 0, \quad U_1(j\pi) = U_3(2j\pi) = 0, \quad j \leq l/2. \tag{45}$$

Given positive integers s, l , we impose conditions (42), (43), and (45) to the matrix $U(x)$, which gives a system of $4(n+l) - 2$ linear and quadratic equations in terms of the coefficients of the polynomials $U_j(x)$, $j = 1, 2, 3, 4$. The required number of free parameters can be obtained by considering

$$\begin{aligned} d(U_1) &= 2(n + l - 1), & d(U_4) &= 2(n + l), \\ d(U_2) &= 2(n + l) - 1, & d(U_3) &= 2(n + l) - 1, \end{aligned}$$

where $d(P(x))$ denotes the degree of the polynomial $P(x)$. For a given matrix $U(x)$ satisfying the required conditions, if there exists a splitting method associated to the matrix (41) (if it exists, is unique [4]), then in general will have $m = 2(n + l) - 1$ stages.

We have obtained (with the help of the software *Mathematica*) all the solutions of the equations corresponding to moderate values of n and l ($n + l \leq 6$). For each n and l , we choose among all the real solutions of the corresponding system of polynomial equations the best methods with respect to suitable criteria based on the rigorous error estimates (for the application of (18)) derived in [6]. Once an appropriate matrix $U(x)$ is chosen for given n and l , we compute the coefficients $\{a_1, b_1, a_2, b_2, \dots\}$ of the splitting scheme corresponding to $K(x) = U(-x)^{-1}U(x)$ by following the algorithm presented in [4]. We collect in Table 1 the coefficients of two of the best methods obtained in this way with $m = 11$ stages with $n = l = 3$ and $n = 4, l = 2$ (in the last case, the method is of order eight at the cost of being slightly less stable due to the smaller value of l).

3.2 Determining the coefficients for the non-autonomous case

The order conditions to be satisfied by the coefficients $a_i^{[k]}, b_i^{[k]}, i, k = 0, 1, \dots, s - 1$ for the polynomials in the scheme (33) to give a method of order $p = 2s$ can be determined by following the approach of [3]. First one considers the formal solution of Eqs. (16) at time h as furnished by the Magnus expansion,

$$z(t + h) = e^{\Omega(t,h)}z(t), \tag{46}$$

where $\Omega(t, h) = \sum_{k=1}^{\infty} \Omega_k(t, h)$ and each $\Omega_k(t, h)$ is a multiple integral of combinations of nested commutators containing k matrices $A(t)$ and $B(t)$ [7, 16]. The explicit expression of $\Omega_k(t, h)$ can be obtained by inserting into the recurrence defining the Magnus expansion a Taylor series of the matrices $A(t)$ and $B(t)$ around the midpoint $t + h/2$ (to take advantage of the time-symmetry property of the solution), i.e.,

$$A(t + \frac{h}{2} + \tau) = \alpha_1 + \alpha_2\tau + \alpha_3\tau^2 + \dots, \quad B(t + \frac{h}{2} + \tau) = \beta_1 + \beta_2\tau + \beta_3\tau^2 + \dots$$

Then, the Baker–Campbell–Hausdorff (BCH) formula is repeatedly applied to (30), so that $K(t, h)$ is expressed as the exponential of only one operator,

$$K(t, h) = \exp(\tilde{\Omega}(t, h)),$$

depending on $\tilde{A}_i(t, h), \tilde{B}_i(t, h)$ ($i = 1, \dots, m$) and nested commutators of these matrices. In consequence, the numerical scheme is of order p if $\tilde{\Omega}(t, h) - \Omega(t, h) = \mathcal{O}(h^{p+1})$ as $h \rightarrow 0$.

The analysis is simplified by imposing time symmetry to the composition (32): $K(t + h, -h) = K(t, h)^{-1}$, or equivalently, $\tilde{\Omega}(t + h - h) = -\tilde{\Omega}(t, h)$. This is automatically satisfied if

$$\tilde{A}_{m+2-i}(t + h, -h) = -\tilde{A}_i(t, h), \quad \tilde{B}_{m+1-i}(t + h, -h) = -\tilde{B}_i(t, h), \quad B_{m+1}(t, h) = 0 \tag{47}$$

for $i = 1, 2, \dots, m$. Expanding $\tilde{A}_i(t, h), \tilde{B}_i(t, h)$ in (31) in terms of α_j, β_j ,

$$\tilde{A}_i(t, h) = \sum_{n \geq 1} h^n a_i^{(n)} \alpha_n, \quad \tilde{B}_i(t, h) = \sum_{n \geq 1} h^n b_i^{(n)} \beta_n, \tag{48}$$

it is clear that conditions (47) are fulfilled as soon as

Table 1 Coefficients $a_i^{(j)}, b_i^{(j)}$ for the 11-stage sixth-order method with $s = 3$ and $n = 3, l = 3$ (SM₁₁6), and $n = 4, l = 2$ (SM₁₁86). The schemes are written as ABA time-reversible compositions. The corresponding coefficients $a_i^{[j]}, b_i^{[j]}$ can be obtained from (58) for $s = 3$, and then using the coefficients from $r_{i,j}^{(3)}$ given in (56)

SM ₁₁ 6		
$a_1^{(1)} = 0.04648745479086313086$	$a_1^{(2)} = 0.02193266201422243554$	$a_1^{(3)} = 0.01077435319621614570$
$b_1^{(1)} = 0.18433048350266556347$	$b_1^{(2)} = 0.07626534298581334973$	$b_1^{(3)} = 0.03014212501832794093$
$a_2^{(1)} = -0.06069167116564293530$	$a_2^{(2)} = -0.02039856592080283195$	$a_2^{(3)} = -0.00542713028889977442$
$b_2^{(1)} = -0.04105690329771146237$	$b_2^{(2)} = -0.01259689047991468431$	$b_2^{(3)} = -0.00437211775744595277$
$a_3^{(1)} = 0.21846652646340681047$	$a_3^{(2)} = 0.08285246661957294702$	$a_3^{(3)} = 0.02955403630534304458$
$b_3^{(1)} = 0.13375567966675033070$	$b_3^{(2)} = 0.02732023273818340957$	$b_3^{(3)} = 0.01331421499301463305$
$a_4^{(1)} = 0.16805357948309270304$	$a_4^{(2)} = 0.03387453196733517440$	$a_4^{(3)} = 0.00676540745400725080$
$b_4^{(1)} = 0.20376454713235473820$	$b_4^{(2)} = 0.05751859844471906953$	$b_4^{(3)} = 0.00258244441277004544$
$a_5^{(1)} = 0.31439236417035348674$	$a_5^{(2)} = -0.00275088080153402384$	$a_5^{(3)} = 0$
$b_5^{(1)} = -0.01176016691496004372$	$b_5^{(2)} = 1/100$	$b_5^{(3)} = 0$
$a_6^{(1)} = \frac{1}{2} - (a_1^{(1)} + \dots + a_5^{(1)})$	$a_6^{(2)} = 0.00270234526088928730$	$a_6^{(3)} = 0$
$b_6^{(1)} = 1 - 2(b_1^{(1)} + \dots + b_5^{(1)})$	$b_6^{(2)} = 0$	$b_6^{(3)} = 0$
$a_{13-i}^{(1)} = a_i^{(1)}$	$a_{13-i}^{(2)} = -a_i^{(2)}$	$a_{13-i}^{(3)} = a_i^{(3)}, i = 1, \dots, 6$
$b_{12-i}^{(1)} = b_i^{(1)}$	$b_{12-i}^{(2)} = -b_i^{(2)}$	$b_{12-i}^{(3)} = b_i^{(3)}, i = 1, \dots, 5$
SM ₁₁ 86		
$a_1^{(1)} = 0.03973124988532299637$	$a_1^{(2)} = 0.02400794642202757534$	$a_1^{(3)} = 0.01426347688561003458$
$b_1^{(1)} = 0.14287701840991454594$	$b_1^{(2)} = 0.0569485755287938004$	$b_1^{(3)} = 0.02419546407969782444$
$a_2^{(1)} = -0.21448946380256107573$	$a_2^{(2)} = -0.05636883329464990245$	$a_2^{(3)} = -0.00827632423527062146$
$b_2^{(1)} = -0.00914109419622051067$	$b_2^{(2)} = -0.00288348177686401417$	$b_2^{(3)} = -0.00078756602133509741$
$a_3^{(1)} = 0.36819766483105942846$	$a_3^{(2)} = 0.10704989071362334460$	$a_3^{(3)} = 0.02378542863893405247$
$b_3^{(1)} = 0.12498111985066240923$	$b_3^{(2)} = 0.04926847474936858693$	$b_3^{(3)} = 0.01500328192941108314$
$a_4^{(1)} = 0.16749724928405616467$	$a_4^{(2)} = 0.03981179966317880912$	$a_4^{(3)} = 0.00949421566450808647$
$b_4^{(1)} = 0.22721131708027567509$	$b_4^{(2)} = 0.01651693314599112332$	$b_4^{(3)} = 0.00388912208303570946$
$a_5^{(1)} = 0.39616080151130058710$	$a_5^{(2)} = -0.0348720903488828659$	$a_5^{(3)} = 0.01581584517797694018$
$b_5^{(1)} = -0.00876131570709263239$	$b_5^{(2)} = 0$	$b_5^{(3)} = -0.00003594151650905306$
$a_6^{(1)} = \frac{1}{2} - (a_1^{(1)} + \dots + a_5^{(1)})$	$a_6^{(2)} = 0.06436174254164758857$	$a_6^{(3)} = -0.01341597546509182559$
$b_6^{(1)} = 1 - 2(b_1^{(1)} + \dots + b_5^{(1)})$	$b_6^{(2)} = 0$	$b_6^{(3)} = -0.0011953877526759980$
$a_{13-i}^{(1)} = a_i^{(1)}$	$a_{13-i}^{(2)} = -a_i^{(2)}$	$a_{13-i}^{(3)} = a_i^{(3)}, i = 1, \dots, 6$
$b_{12-i}^{(1)} = b_i^{(1)}$	$b_{12-i}^{(2)} = -b_i^{(2)}$	$b_{12-i}^{(3)} = b_i^{(3)}, i = 1, \dots, 5$

$$a_{m+1-i}^{(n)} = (-1)^{n+1} a_i^{(n)}, \quad b_{m-i}^{(n)} = (-1)^{n+1} b_i^{(n)}, \quad b_m^{(n)} = 0, \quad (49)$$

for $n \geq 1, i = 1, 2, \dots, m$.

In the autonomous case one has $\alpha_1 = hA, \beta_1 = hB$ and $\alpha_j = \beta_j = 0, j > 1$ so that the scheme reduces to

$$K(h) = e^{a_{m+1}\alpha_1} e^{b_m^{(1)}\beta_1} e^{a_m^{(1)}\alpha_1} \dots e^{b_m^{(1)}\beta_1} e^{a_1^{(1)}\alpha_1}, \quad (50)$$

which corresponds to Eq. (20) with $a_i^{(1)} = a_i, b_i^{(1)} = b_i$. From (50) it is clear that $a_i^{(1)}, b_i^{(1)}$ can be taken separately from the remaining coefficients. For instance, from (31) we have

$$a_i^{(1)} = h \int_{-1/2}^{1/2} p_i^a(\tau) d\tau, \quad b_i^{(1)} = h \int_{-1/2}^{1/2} p_i^b(\tau) d\tau. \quad (51)$$

In any case, the actual choice of $a_i^{(1)}, b_i^{(1)}$ plays an essential role to get the coefficients $a_i^{[j]}, b_i^{[j]}$ leading to efficient methods, since the constant parts $\alpha_1 = hA(t_{1/2}), \beta_1 = hB(t_{1/2})$ usually represent the dominant contributions to the evolution of the system.

Once a set of values for $a_i^{(1)}, b_i^{(1)}, i = 1, \dots, m$, satisfying the symmetry condition (49) is chosen, we have to determine the coefficients, $a_i^{(n)}, b_i^{(n)}, n \geq 2, i = 1, \dots, m$, which satisfy the remaining order conditions. This can produce different methods with the same values for the coefficients $a_i^{(1)}, b_i^{(1)}$.

Let us now write \tilde{A}_i, \tilde{B}_i as follows:

$$\tilde{A}_i = \sum_{j=0}^{s-1} a_i^{[j]} A^{(j)}, \quad \tilde{B}_i = \sum_{j=0}^{s-1} b_i^{[j]} B^{(j)}, \quad (52)$$

where

$$A^{(j)} \equiv h \int_{-1/2}^{1/2} \tau^j A(t_{1/2} + h\tau) d\tau, \quad B^{(j)} \equiv h \int_{-1/2}^{1/2} \tau^j B(t_{1/2} + h\tau) d\tau \quad (53)$$

for $j = 0, \dots, s - 1$.

At this point the following remark is worth to be stated. Suppose that $\bar{b}_i, c_i (i = 1, \dots, k)$, are the weights and nodes of a particular quadrature rule for integrals. Then it is possible to approximate all the integrals $A^{(i)}$ (up to the required order) just by using only the evaluations A_i at the nodes c_i of the quadrature rule required to compute $A^{(0)}$:

$$A^{(i)} = h \sum_{j=1}^k \bar{b}_j \left(c_j - \frac{1}{2} \right)^i A_j, \quad i = 0, \dots, s - 1, \quad (54)$$

with $A_i \equiv A(t_n + c_i h)$. In particular, if fourth and sixth order Gauss–Legendre quadrature rules are considered, we have $s = k = 2$ and $\bar{b}_1 = \bar{b}_2 = 1/2, c_1 = 1/2 - \sqrt{3}/6, c_2 = 1/2 + \sqrt{3}/6$. To order six we have $s = k = 3$ and $\bar{b}_1 = \bar{b}_3 = 5/18, \bar{b}_2 = 4/9, c_1 = 1/2 - \sqrt{15}/10, c_2 = 1/2, c_3 = 1/2 + \sqrt{15}/10$.

Now, a simple relationship can be established between the coefficients $a_i^{[j]}, b_i^{[j]}$ for a given method and the coefficients $a_i^{(n)}, b_i^{(n)}$ by taking into account how the matrices $A^{(i)}, B^{(i)}$ and α_i, β_i are related. Specifically, one has (neglecting higher order terms)

$$A^{(i)} = \int_{-1/2}^{1/2} \sum_{j=1}^s h^j \alpha_j \tau^{i+j-1} d\tau = \sum_{j=1}^s (T^{(s)})_{ij} h^j \alpha_j \equiv \sum_{j=1}^s \frac{1 - (-1)^{i+j}}{(i+j)2^{i+j}} h^j \alpha_j, \quad (55)$$

$0 \leq i \leq s - 1$, and an analogous expression relating $B^{(i)}$ with β_i . If this relation is inverted (to order four, $s = 2$, and six, $s = 3$) we get

$$(r_{ij}^{(2)}) \equiv (T^{(2)})^{-1} = \begin{pmatrix} 1 & 0 \\ 0 & 12 \end{pmatrix}, \quad (r_{ij}^{(3)}) = \begin{pmatrix} \frac{9}{4} & 0 & -15 \\ 0 & 12 & 0 \\ -15 & 0 & 180 \end{pmatrix} \quad (56)$$

respectively. If we consider (48)

$$\begin{aligned}\tilde{A}_i &= \sum_{n=1}^s a_i^{(n)} h^n \alpha_n = \sum_{n=1}^s a_i^{(n)} \sum_{j=1}^s r_{nj}^{(s)} A^{(j-1)} \\ \tilde{B}_i &= \sum_{n=1}^s b_i^{(n)} h^n \alpha_n = \sum_{n=1}^s b_i^{(n)} \sum_{j=1}^s r_{nj}^{(s)} B^{(j-1)},\end{aligned}\tag{57}$$

and compare with (52) then

$$a_i^{[j]} = \sum_{n=1}^s a_i^{(n)} r_{n,j+1}^{(s)}, \quad b_i^{[j]} = \sum_{n=1}^s b_i^{(n)} r_{n,j+1}^{(s)}.\tag{58}$$

From this analysis, we proceed as follows. We first compute the coefficients a_i, b_i by applying the procedure shown before for the autonomous case. Then, we take $a_i^{(1)} = a_i, b_i^{(1)} = b_i$ and then we get the coefficients $a_i^{(k)}, b_i^{(k)}, k > 1$.

In this work we have considered symmetric methods of order six. For the autonomous case the schemes have necessarily $m \geq 5$ stages. In consequence, we have analyzed compositions with $m = 5, 7, 9, 11$ which corresponds to take $n = 3$ and $l = 0, 1, 2, 3$ in Sect. 3.1, but we have also analyzed the case $n = 4$ and $l = 0, 1, 2$. In each case we have taken the optimal choices according to the criterion considered before.

On the other hand, getting sixth-order methods for the non-autonomous case requires to solve a system of 10 nonlinear equations in the variables $a_i^{(2)}, b_i^{(2)}$ and an additional linear system of 8 equations to be solved in $a_i^{(3)}, b_i^{(3)}$, with the symmetry (49). To obtain real solutions for these equations it was necessary to consider methods with at least 11 stages. Here we have carried out an exhaustive search of solutions and examined how all of them behave in practice. This turns out to be the set of coefficients which minimize the sum of the absolute values of the coefficients. The corresponding sets are collected in Table 1. For future reference, we refer to these (optimal in the previous sense) methods as SM₁₁₆ and SM₁₁₈₆. The coefficients $a_i^{[j]}, b_i^{[j]}$ can be obtained from (58) for $s = 3$, and then using the coefficients from $(r_{i,j}^{(3)})$ given in (56). As a matter of fact, in [3] we included in the numerical examples the results achieved by SM₁₁₆, but without indicating how the scheme was obtained.

4 Numerical examples

Our purpose in this section is to illustrate the performance of the new specially adapted 11-stage sixth-order (SM₁₁₆ and SM₁₁₈₆) splitting methods for partitioned non-autonomous linear systems. To do so, we carry out some comparisons with some other well established general purpose geometric schemes. Given a basic symmetric second order method, we consider symmetric compositions of the basic method with fractional steps. The most efficient compositions of this family of orders 6 to 10 known by the authors (with real coefficients and without processor or corrector) correspond to the following methods: the 13-stage sixth-order (S₁₃₆), the 21-stage eighth-order (S₂₁₈), and the 35-stage tenth-order (S₃₅₁₀) methods obtained in [22]. The coefficients of S₃₅₁₀ are also collected, for instance, in [13, chapter V]. As basic second order scheme we take the well known leapfrog composition: $\psi_h^{[2]} \equiv \varphi_{h/2}^{[B]} \circ \varphi_h^{[A]} \circ \varphi_{h/2}^{[B]}$. The explicit time dependence is treated by taking the time as two new coordinates as shown in (10)–(11) or (25). We also consider the 11-stage sixth-order

Runge–Kutta–Nyström splitting method, RKN_{116} , given in [10] and adapted to explicitly time-dependent problems [9] similarly as given in (28)–(29), which gives similar performance to the scheme proposed in [1], but it is simpler to implement. This method is also implemented taking the time as two new coordinates, as in the $S_m p$ schemes, to illustrate the advantage of treating the time dependency appropriately (in this case it is referred as RKN_{1164}).

Since the main interest of this family of methods lies in the numerical integration of differential equations originated from space discretizations of PDEs, the computational cost of the methods is measured by the number of stages required. The integrals (31) appearing in the new scheme are approximated by the sixth-order Gauss–Legendre quadrature rule, as indicated in (52) and (54). Notice that using this quadrature rule only three evaluations for the time dependent functions are required per step, whereas 11 is the number used to count the cost of the algorithm.

Perturbed harmonic oscillators. We first consider as a simple test bench problem the Mathieu equation,

$$q'' + (\omega^2 + \epsilon \cos(t))q = 0,$$

with $q \in \mathbb{R}$, which corresponds to a time dependent linear harmonic oscillator with Hamiltonian

$$H = \frac{1}{2}p^2 + \frac{1}{2}(\omega^2 + \epsilon \cos(t))q^2.$$

We take as initial condition $q(0) = 1$, $p(0) = 1$, integrate up to $t = 200\pi/\omega$ and measure the average error in phase space (at $t = 2\pi/\omega, 4\pi/\omega, \dots, 200\pi/\omega$) in terms of the number of force evaluations for different time steps (in logarithmic scale). We compare the relative error for $\omega = 3/2$ with $\epsilon = 1/40$ and $\epsilon = 1/4$, and $\omega = 5$, with $\epsilon = 1/40$ and $\epsilon = 4$. The results are collected in Fig. 1. Since the kinetic part is time-independent, the RKN_{116} method can be used in this case by just taking the time as a new coordinate. The superiority of the new schemes is manifest for all accuracies of practical interest. We can also observe that this superiority is more relevant when the dominant contribution from the Hamiltonian originates from the constant part, since the new scheme is built to be highly efficient for small perturbations of the autonomous harmonic oscillator. In this setting the scheme SM_{1186} is superior and behaves as an eighth-order method in the case $\omega = 5$, $\epsilon = 1/40$. We observe that when the time-dependent functions are significant, the errors associated to the new method are dominated by the Magnus expansion treatment. This is an important observation to take into account to build new improved methods in this class.

The Schrödinger equation. Let us now consider the one-dimensional Schrödinger equation (in units where $\hbar = 1$)

$$i \frac{\partial}{\partial t} \psi(x, t) = \left(-\frac{1}{2\mu} \frac{\partial^2}{\partial x^2} + V(x) + f(t)x \right) \psi(x, t), \quad (59)$$

with $\psi(x, 0) = \psi_0(x)$. We take the Morse potential $V(x) = D(1 - e^{-\alpha x})^2$ in a laser field described by $f(t)x = A \cos(\omega t)x$. It corresponds to the Walker–Preston model of a diatomic molecule in a strong laser field [23]. This problem is used as a test bench for the numerical methods presented in [12, 21] and the same values for the parameters are taken (in atomic units): $\mu = 1,745$ a.u., $D = 0.2251$ a.u. and $\alpha = 1.1741$ a.u. (corresponding to the HF molecule), $A = 0.011025$ a.u. and laser frequency $\omega = 0.01787$. We assume that the system

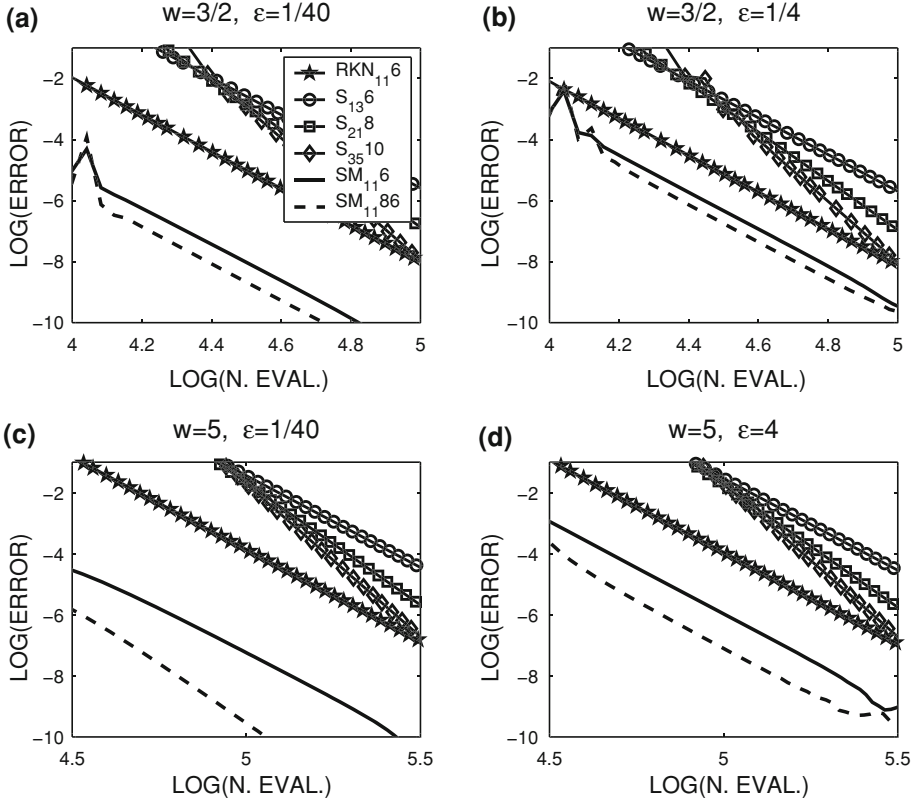


Fig. 1 Average error in phase space (at $t = 2\pi/\omega, 4\pi/\omega, \dots, 100\pi/\omega$) versus the number of force evaluations for different time steps (in logarithmic scale) for the Mathieu equation for different values of ω and ϵ .

is defined in the interval $x \in [-0.8, 4.32]$, which is split into $N = 64$ parts of length $\Delta x = 0.08$, and impose periodic boundary conditions.

After space discretization, Eq. (59) leads to the complex linear equation (12) with $u \in \mathbb{C}^N$ and $u_k(t) = \psi(x_k, t)(\Delta x)^{1/2}$, $k = 0, 1, \dots, N - 1$. Here $x_k = x_0 + k\Delta x$ and $H(t) = T + \hat{V}(t)$ is an Hermitian matrix (real and symmetric). As initial conditions we take the ground state of the Morse potential,

$$\phi(x) = \sigma \exp(-(\gamma - 1/2)\alpha x) \exp(-\gamma e^{-\alpha x}),$$

with $\gamma = 2D/w_0$, $w_0 = \alpha\sqrt{2D/\mu}$ and σ is a normalizing constant.

Notice that $\hat{V}(t)$ is a diagonal matrix with elements $\hat{V}_{jj} = V(x_j) + f(t)x_j$, and Tq, Tp can be efficiently computed using FFTs [2, 11]. Notice also that in (33) we now have

$$H_i^A = h \int_{-1/2}^{1/2} p_i^a(\tau) H(t_{1/2} + h\tau) d\tau = ha_i^{(1)} T + ha_i^{(1)} V + hX F_a(t, h) \quad (60)$$

and

$$H_i^B = h \int_{-1/2}^{1/2} p_i^b(\tau) H(t_{1/2} + h\tau) d\tau = hb_i^{(1)} T + hb_i^{(1)} V + hX F_b(t, h), \quad (61)$$

where $F_{a/b}(t, h) = h \int_{-1/2}^{1/2} p_i^{a/b}(\tau) f(t_{1/2} + h\tau) d\tau$. Here X is a diagonal matrix with diagonal elements $X_{jj} = x_j$. The products $H_i q$ and $H_i p$ only require one FFT and its inverse and thus an m -stage method requires $4m$ FFTs per step. The split (25) used for the general purpose methods $S_m p$ was already proposed in [21], showing a clear improvement with respect to the second order Magnus integrator (combined with a third order splitting scheme) given in [12]. This split is also used for the RKN_{116} method, but it produces a fourth-order scheme. This is corrected using the averaging (28)–(29) which only requires to replace the scalar function $f(t)$ by its corresponding integral along each fractional time-step in one part of the separable Hamiltonian system.

First, we integrate the system in the time interval $t \in [0, 2\tau]$ with $\tau = 2\pi/\omega$. The exact solution, $u_{ex}(2\tau)$, at the final time is obtained numerically using a sufficiently small time step. We measure the error in the wave function, $\|u_{ex}(2\tau) - u(2\tau)\|$ versus the number of FFTs required for each method, and this is repeated for different values of the time step, starting with a very small time step and increasing it until reaching a time step close to the

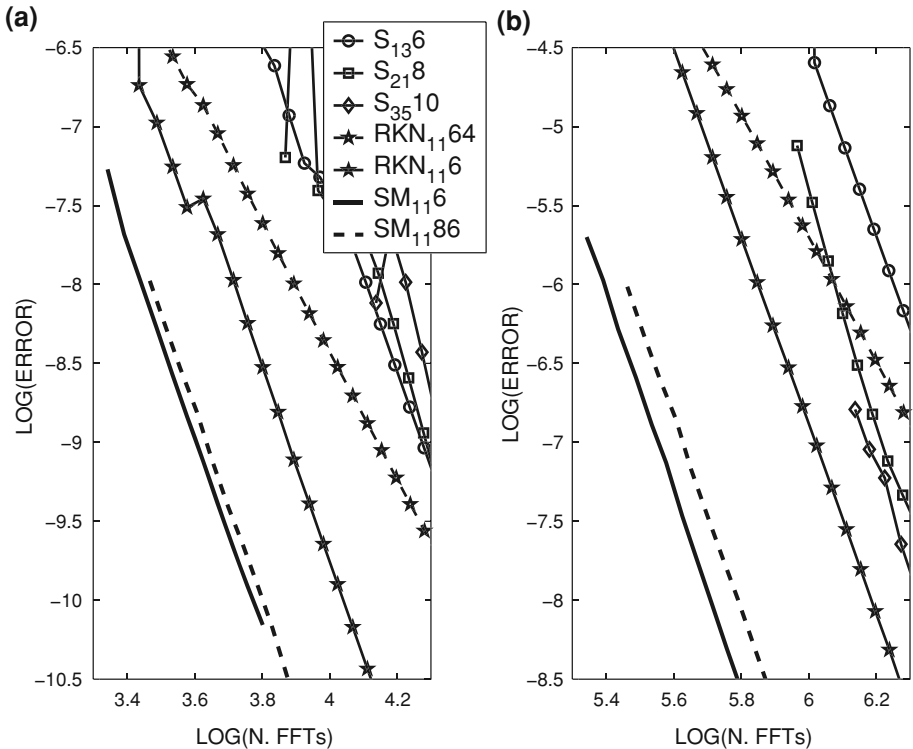


Fig. 2 Error in the wave function versus the number of FFTs for the one-dimensional Schrödinger equation (59) written in the form (13) after space discretization: **a** for the time integration $t \in [0, 2\tau]$, and **b** for $t \in [0, 200\tau]$ with $\tau = 2\pi/\omega$.

stability limit of the method (an overflow appears if the time step is slightly increased). We repeated the same experiment taking a larger time integration, $t \in [0, 200\tau]$.

Figure 2 shows the efficiency plots for the methods. The superiority of the new splitting methods is manifest both with respect to efficiency and the stability limit. In addition, we observe that this superiority increases when taking a longer time integration. This constitutes indeed an interesting property of the new methods which is currently under investigation [6]. Scheme SM₁₁6 is more stable (it was built with a larger value of l), but SM₁₁86 can be more efficient when a high accuracy is desired and the time dependent functions originate a small perturbation to the remaining autonomous problem, because in that limit the method behaves as an eighth-order method.

Acknowledgments This work has been supported by Ministerio de Ciencia e Innovación (Spain) under project MTM2007-61572 (co-financed by the ERDF of the European Union). SB also acknowledges financial support from Generalitat Valenciana through project GV/2009/032.

References

1. Blanes, S., Casas, F.: Splitting methods for non-autonomous separable dynamical systems. *J. Phys. A. Math. Gen.* **39**, 5405–5423 (2006)
2. Blanes, S., Casas, F., Murua, A.: Symplectic splitting operator methods tailored for the time-dependent Schrödinger equation. *J. Chem. Phys.* **124**, 234105 (2006)
3. Blanes, S., Casas, F., Murua, A.: Splitting methods for non-autonomous linear systems. *Int. J. Comput. Math.* **84**, 713–727 (2007)
4. Blanes, S., Casas, F., Murua, A.: On the linear stability of splitting methods. *Found. Comp. Math.* **8**, 357–393 (2008)
5. Blanes, S., Casas, F., Murua, A.: Splitting and composition methods in the numerical integration of differential equations. *Bol. Soc. Esp. Math. Apl.* **45**, 87–143 (2008)
6. Blanes, S., Casas, F., Murua, A.: Error analysis of splitting methods for the time dependent Schrödinger equation. arXiv:1001.1549 (2011)
7. Blanes, S., Casas, F., Oteo, J.A., Ros, J.: The Magnus expansion and some of its applications. *Phys. Rep.* **470**, 151–238 (2009)
8. Blanes, S., Casas, F., Ros, J.: Improved high order integrators based on Magnus expansion. *BIT* **40**, 434–450 (2000)
9. Blanes, S., Diele, F., Marangi, C., Ragni, S.: Splitting and composition methods for explicit time dependence in separable dynamical systems. *J. Comput. Appl. Math.* **235**, 646–659 (2010)
10. Blanes, S., Moan, P.C.: Practical symplectic partitioned Runge–Kutta and Runge–Kutta–Nyström methods. *J. Comput. Appl. Math.* **142**, 313–330 (2002)
11. Gray, S., Manolopoulos, D.E.: Symplectic integrators tailored to the time-dependent Schrödinger equation. *J. Chem. Phys.* **104**, 7099–7112 (1996)
12. Gray, S., Verosky, J.M.: Classical Hamiltonian structures in wave packet dynamics. *J. Chem. Phys.* **100**, 5011–5022 (1994)
13. Hairer, E., Lubich, C., Wanner, G.: Geometric numerical integration. Structure-preserving algorithms for ordinary differential equations, 2nd ed.. Springer, Berlin (2006)
14. Iserles, A., Munthe-Kaas, H.Z., Nørsett, S.P., Zanna, A.: Lie group methods. *Acta Numer.* **9**, 215–365 (2000)
15. Leimkuhler, B., Reich, S.: *Simulating Hamiltonian Dynamics*. Cambridge University Press, Cambridge (2004)
16. Magnus, W.: On the exponential solution of differential equations for a linear operator. *Commun. Pure Appl. Math.* **7**, 649–673 (1954)
17. McLachlan, R.I., Quispel, R.: Splitting methods. *Acta Numer.* **11**, 341–434 (2002)
18. McLachlan, R.I., Quispel, R.G.W.: Geometric integrators for ODEs. *J. Phys. A. Math. Gen.* **39**, 5251–5285 (2006)
19. Rieben, R., White, D., Rodrigue, G.: High-order symplectic integration methods for finite element solutions to time dependent Maxwell equations. *IEEE Trans. Antennas Propag.* **52**, 2190–2195 (2004)
20. Sanz-Serna, J.M., Calvo, M.P.: *Numerical Hamiltonian Problems*. Chapman & Hall, London (1994)

-
21. Sanz-Serna, J.M., Portillo, A.: Classical numerical integrators for wave-packet dynamics. *J. Chem. Phys.* **104**, 2349–2355 (1996)
 22. Sofroniou, M., Spaletta, G.: Derivation of symmetric composition constants for symmetric integrators. *Optim. Methods Softw.* **20**, 597–613 (2005)
 23. Walker, R.B., Preston, K.: Quantum versus classical dynamics in treatment of multiple photon excitation of anharmonic-oscillator. *J. Chem. Phys.* **67**, 2017–2028 (1977)