Symplectic time-average propagators for the Schrödinger equation with a time-dependent Hamiltonian

Sergio Blanes, Fernando Casas, and Ander Murua

Citation: The Journal of Chemical Physics **146**, 114109 (2017); doi: 10.1063/1.4978410

View online: http://dx.doi.org/10.1063/1.4978410

View Table of Contents: http://aip.scitation.org/toc/jcp/146/11

Published by the American Institute of Physics

Symplectic time-average propagators for the Schrödinger equation with a time-dependent Hamiltonian

Sergio Blanes,[1,a)] Fernando Casas,[2,b)] and Ander Murua[3,c)]
[1]*Instituto de Matemática Multidisciplinar, Universitat Politècnica de València, E-46022 Valencia, Spain*
[2]*Departament de Matemàtiques and IMAC, Universitat Jaume I, E-12071 Castellón, Spain*
[3]*Konputazio Zientziak eta Adimen Artifiziala Saila, Informatika Fakultatea, EHU/UPV, Donostia/San Sebastián, Spain*

(Received 28 December 2016; accepted 27 February 2017; published online 21 March 2017)

Several symplectic splitting methods of orders four and six are presented for the step-by-step time numerical integration of the Schrödinger equation when the Hamiltonian is a general explicitly time-dependent real operator. They involve linear combinations of the Hamiltonian evaluated at some intermediate points. We provide the algorithm and the coefficients of the methods, as well as some numerical examples showing their superior performance with respect to other available schemes. *Published by AIP Publishing.* [http://dx.doi.org/10.1063/1.4978410]

I. INTRODUCTION

Many quantum mechanical problems of physical interest involve time-dependent potentials. One has to deal then with the time-dependent Schrödinger equation ($\hbar = 1$)

$$i\frac{\partial}{\partial t}\psi(x,t) = \hat{H}(t)\psi(x,t), \qquad \psi(x,0) = \psi_0(x), \tag{1}$$

where the Hamiltonian is given by

$$\hat{H}(t) = \hat{T} + \hat{V}(t) = -\Delta/(2\mu) + \hat{V}(t), \tag{2}$$

where μ is the reduced mass, Δ is the Laplacian operator, and $\hat{V}(t)$ depends explicitly on time. If the potential changes slowly with time, a usual technique for getting approximate numerical solutions consists of subdividing the time integration interval in a number of sufficiently short subintervals of length τ so that $\hat{V}$ is approximately constant in each subinterval and then solve (1) and (2) with the time-varying potential $\hat{V}(t)$ replaced by

$$\overline{V}(t) = \overline{V}_k \quad \text{for} \quad t \in [(k-1)\tau, k\tau] \tag{3}$$

with time-independent potentials $\overline{V}_k$. In this way, one gets $\psi_k \approx \psi(\cdot, t_k)$ at times $t_k = k\tau, k = 1, 2, 3, \ldots$ and eventually reaches the desired final time. The time-independent potential $\overline{V}_k$ can be chosen in several ways: as the average on the subinterval, $\overline{V}_k = (1/\tau)\int_{t_{k-1}}^{t_k} \hat{V}(t)dt$, as the value at the midpoint, $\overline{V}_k = \hat{V}(t_{k-1} + \tau/2)$, etc.[1–3]

The procedure typically requires discretizing first the Schrödinger equation (1) in space, for which several techniques are widely used: finite differences, spectral methods based on collocation with trigonometric polynomials, Galerkin method with a Hermite basis, etc, both in one or several dimensions (see Ref. 4 and references therein). Once this process has been carried out, one is led to a linear ordinary differential equation (ODE) of the form

$$i\frac{d}{dt}u(t) = H(t)\,u(t), \qquad u(0) = u_0 \in \mathbb{C}^N, \tag{4}$$

where $H(t)$ is a Hermitian piecewise constant $N \times N$ matrix,

$$H(t) = H_k \quad \text{for} \quad t \in [(k-1)\tau, k\tau], \tag{5}$$

and each $H_k = T + V_k$ is a discretized counterpart of the operator $\overline{H}_k \equiv \hat{T} + \overline{V}_k$. Its solution $u(t) \in \mathbb{C}^N$ represents a fully discretized version of the wave function $\psi(x,t)$ at N space grid points, with N usually a large number. The values $u_k = u(k\tau)$ of the solution $u(t)$ of (4) evaluated at the time grid points $t = k\,\tau$ ($k = 1, 2, \ldots$) can be computed as

$$u_k = \mathrm{e}^{-i\,\tau\,H_k}\,u_{k-1}, \tag{6}$$

but instead of evaluating exactly the matrix exponentials in (6) for determining u_k, two alternatives are widely used: (i) the exponential is approximated by the so-called unitary split operator algorithms, i.e., compositions of the form

$$\mathrm{e}^{-i\,\tau\,H_k} \approx \mathrm{e}^{-ib_m\tau V_k}\,\mathrm{e}^{-ia_m\tau T}\cdots\mathrm{e}^{-ib_1\tau V_k}\,\mathrm{e}^{-ia_1\tau T}, \tag{7}$$

where $\{a_i, b_i\}$ are appropriately chosen real coefficients,[5–8] and (ii) the action of the exponential on the vector u_{k-1} is approximated as

$$\mathrm{e}^{-i\,\tau\,H_k}u_{k-1} \approx P_m(\tau\,H_k)u_{k-1}, \tag{8}$$

where $P_m(y)$ is a polynomial of degree m in y that approximates the exponential $\mathrm{e}^{-i\,y}$. Standard choices for such $P_m(y)$ are truncated Taylor or Chebyshev series expansion of $\mathrm{e}^{-i\,y}$ for an appropriate real interval of y or a Lanczos approximation.[9]

When H_k is an arbitrary real symmetric matrix, the use of symplectic splitting methods has allowed the present authors to develop an algorithm that approximately computes $\mathrm{e}^{-i\,\tau\,H_k}v$ for given $\tau \in \mathbb{R}$ and $v \in \mathbb{C}^N$ within a prescribed error tolerance with a minimum number of matrix-vector products $H_k v$. The algorithm is more efficient than Chebyshev methods under the same conditions for all tolerances and time integration intervals.[10]

If the Hamiltonian does not evolve so slowly with time, the previous procedure is not particularly appropriate,

a)Electronic mail: serblaza@imm.upv.es.
b)Electronic mail: Fernando.Casas@uji.es.
c)Electronic mail: Ander.Murua@ehu.es.

however. Notice that either averaging $\hat{H}$ on each subinterval or taking each value at the midpoint produces only a second order of approximation in τ, whereas both the Chebyshev methods and the algorithm based on symplectic splitting methods have been designed to be used with large time steps. In consequence, the performance of such methods for approximately computing $\mathrm{e}^{-i\tau H_k}v$ degrades with a reduced τ, whereas large values of τ produce results with low accuracy due to a poor approximation of the original time-dependent potential $\hat{V}(t)$ by the piecewise constant potential (3). In that case, it would be more appropriate to use the numerical integrators designed to solve a linear system of the form (4) with $H(t)$, a continuously time-dependent matrix.

Many algorithms specifically designed for Equation (4) exist indeed in the literature. We mention in particular split-operator methods,[11] Runge–Kutta[12] and symplectic partitioned Runge–Kutta methods,[13] and a combination of a 4th-order Magnus method with the Lanczos algorithm[2] and the so-called (t, t') method.[14] In the paper,[2] an exhaustive comparison is carried out of these schemes. Polynomial approximations to the propagator based on Taylor[15] and Chebyshev polynomials[16–19] have also been proposed to deal with this problem. The review[19] provides a thorough presentation of one such class of polynomial approximations which are in principle also valid for nonlinear versions of the Schrödinger equation.

Motivated by the excellent performance exhibited by symplectic splitting methods for the computation of $\mathrm{e}^{-i\tau H}v$ for real symmetric matrices H, our purpose here is to adapt that methodology to the more general case of a time-dependent matrix $H(t)$ by using some techniques introduced in Refs. 20 and 21. The resulting algorithms only involve matrix-vector products of the form $H(t)v$ at different values of t. Here we mainly focus on the application to linear ordinary differential equation (4) obtained by a space discretization of a Hamiltonian operator $\hat{H}(t)$ in (1) and (2) with a time-dependent potential, although the integration schemes presented in this paper are valid for arbitrary linear ODE systems of the form (4) with a time-dependent real matrix $H(t)$, provided that $H(t)$ is reasonably smooth in t.

II. SYMPLECTIC SPLITTING METHODS

Before presenting the proposed numerical integrators for systems of the form (4), we first recall the relatively simpler case of a constant real matrix $H(t) \equiv \widetilde{H}$ and the existing schemes within the same family designed to approximate $u(t) = \mathrm{e}^{-it\widetilde{H}}u_0$.

A. Time-independent Hamiltonian

The class of symplectic splitting methods developed in Refs. 22 and 23 as an alternative to Chebyshev polynomial approximations of $\mathrm{e}^{-i\tau\widetilde{H}}$ is obtained by applying special purpose integrators to the equation

$$i\frac{d}{dt}u = \widetilde{H}u, \qquad u(0) = u_0 \tag{9}$$

expressed in terms of the real and imaginary part of the vector $u \in \mathbb{C}^N$, that is,

$$\begin{pmatrix} q \\ p \end{pmatrix} = \begin{pmatrix} \mathrm{Re}(u) \\ \mathrm{Im}(u) \end{pmatrix} \in \mathbb{R}^{2N},$$

so that Eq. (9) is equivalent to

$$\frac{d}{dt}\begin{pmatrix} q \\ p \end{pmatrix} = \begin{pmatrix} 0 & \widetilde{H} \\ 0 & 0 \end{pmatrix}\begin{pmatrix} q \\ p \end{pmatrix} + \begin{pmatrix} 0 & 0 \\ -\widetilde{H} & 0 \end{pmatrix}\begin{pmatrix} q \\ p \end{pmatrix}. \tag{10}$$

Application of a splitting scheme with appropriate coefficients

$$a_1, b_1, a_2, \ldots, a_m, b_m, a_{m+1} \in \mathbb{R} \tag{11}$$

leads (see Refs. 10 and 24–26) to an approximation of the solution operator of (10) of the form

$$\begin{pmatrix} I & \tau a_{m+1}\widetilde{H} \\ 0 & I \end{pmatrix}\begin{pmatrix} I & 0 \\ -\tau b_m\widetilde{H} & I \end{pmatrix}\cdots\begin{pmatrix} I & 0 \\ -\tau b_1\widetilde{H} & I \end{pmatrix}\begin{pmatrix} I & \tau a_1\widetilde{H} \\ 0 & I \end{pmatrix}. \tag{12}$$

If this procedure is applied to get a numerical approximation of $u(t)$ in the time interval $t \in [0, t_f]$ using M steps with time step $\tau = t_f/M$, then the following algorithm results:

$$\begin{aligned}
&v = \widetilde{H}\,\mathrm{Im}(u_0)\\
&\textbf{do}\ \ k = 0, M-1\\
&\quad q_0 = \mathrm{Re}(u_k), \quad p_0 = \mathrm{Im}(u_k)\\
&\quad q_1 = q_0 + \tau\, a_1 v\\
&\quad \textbf{do}\ \ i = 1, m\\
&\qquad p_i = p_{i-1} - \tau\, b_i\widetilde{H}q_i\\
&\qquad v = \widetilde{H}p_i\\
&\qquad q_{i+1} = q_i + \tau\, a_{i+1}v\\
&\quad \textbf{enddo}\\
&\quad u_k = q_{m+1} + ip_m\\
&\textbf{enddo}.
\end{aligned} \tag{13}$$

The coefficients of method (12) usually verify $a_{m+2-j} = a_j$ and $b_{m+1-j} = b_j$. In other words, composition (12) is left-right palindromic so that the scheme preserves time-symmetry. Although it is neither unitary nor unconditionally stable (as is the case of method (7)), it is symplectic and conjugate to a unitary scheme so that neither the average error in energy nor the norm of the approximate solution grows with time. Symplectic integrators preserve most qualitative properties of the exact solution and belong to a class of geometric numerical integrators[27–29] showing, in general, a more favorable error propagation than standard (e.g., Runge-Kutta, Taylor, multistep, etc.) integrators.

The error analysis carried out in Ref. 26, in particular the sharp error estimates obtained there, provides appropriate criteria to construct splitting methods of the form (12) possessing a large stability interval and especially adapted to different regularity conditions by optimizing the main error terms. This analysis has allowed the present authors to construct a set of symplectic splitting methods designed to be used under different precision requirements and time intervals, as well as a technique to select among them the most efficient scheme to approximate $\mathrm{e}^{-i\tau\widetilde{H}}v$ with the desired accuracy and a reduced computational cost. The resulting algorithm is presented in Ref. 10, where its performance is also compared with methods based on Chebyshev polynomials.

B. Time-dependent Hamiltonian

In the following, we address the more general problem (4) with a generic time-dependent real matrix $H(t)$ and show how the previous splitting scheme (12) can be generalized to this setting. Here we only provide a brief summary of the main steps involved and refer the reader to Refs. 20 and 21 for a more detailed treatment.

We again express $u \in \mathbb{C}^N$ in terms of the real and imaginary part so that Eq. (4) is equivalently written as

$$\frac{d}{dt}\begin{pmatrix} q \\ p \end{pmatrix} = \begin{pmatrix} 0 & H(t) \\ 0 & 0 \end{pmatrix}\begin{pmatrix} q \\ p \end{pmatrix} + \begin{pmatrix} 0 & 0 \\ -H(t) & 0 \end{pmatrix}\begin{pmatrix} q \\ p \end{pmatrix}. \tag{14}$$

Let $K(t,\tau)$ denote the solution operator so that any solution $(q(t), p(t))$ of (14) is formally expressed as

$$\begin{pmatrix} q(t+\tau) \\ p(t+\tau) \end{pmatrix} = K(t,\tau)\begin{pmatrix} q(t) \\ p(t) \end{pmatrix}$$

for all $t, \tau \in \mathbb{R}$.

To determine the solution $u(t)$ of the linear ODE system (14) at times $t = t_k$ $(k = 1, 2, 3, \ldots)$, we will obtain approximations $q_k \approx q(t_k) = \mathrm{Re}(u(t_k))$, $p_k \approx p(t_k) = \mathrm{Im}(u(t_k))$ for $k = 0, 1, 2, \ldots$ as

$$\begin{pmatrix} q_{k+1} \\ p_{k+1} \end{pmatrix} = \widetilde{K}(t_k,\tau)\begin{pmatrix} q_k \\ p_k \end{pmatrix}$$

(starting from $q_0 = \mathrm{Re}(u_0)$, $p_0 = \mathrm{Im}(u_0)$), where $\widetilde{K}(t,\tau)$ is some approximation of the solution operator $K(t,\tau)$. More specifically, our proposal is to take $\widetilde{K}(t,\tau)$ as

$$\widetilde{K}(t,\tau) = \begin{pmatrix} I & \widetilde{H}^A_{m+1} \\ 0 & I \end{pmatrix}\begin{pmatrix} I & 0 \\ -\widetilde{H}^B_m & I \end{pmatrix}\cdots\begin{pmatrix} I & 0 \\ -\widetilde{H}^B_1 & I \end{pmatrix}\begin{pmatrix} I & \widetilde{H}^A_1 \\ 0 & I \end{pmatrix}, \tag{15}$$

where

$$\widetilde{H}^A_i = \tau\sum_{j=1}^{n} a_{i,j}H(t+c_j\tau), \quad \widetilde{H}^B_i = \tau\sum_{j=1}^{n} b_{i,j}H(t+c_j\tau), \tag{16}$$

for some n and appropriately chosen coefficients $a_{i,j}, b_{i,j}, c_j \in \mathbb{R}$. Notice that the composition (15) is similar in form to (12), but now the matrix $\widetilde{H}$ in each factor in (15) is replaced by a different (weighted) time-average of $H(t)$. The approximation (15) is said to have m stages.

When schemes (15) and (16) with $n = 3$ are considered as the approximate solution operator, then the following algorithm results for determining the approximations $u_k \approx u(t_k)$, $k = 1, 2, 3, \ldots$

$$\begin{aligned}
&H^A = 0, \quad u_k = u_0 \\
&\textbf{do } k = 0, M-1 \\
&\quad t_k = t_0 + k\,\tau, \quad q_0 = \mathrm{Re}(u_k), \quad p_0 = \mathrm{Im}(u_k) \\
&\quad H_1 = H(t_k + c_1\tau), \quad H_2 = H(t_k + c_2\tau), \quad H_3 = H(t_k + c_3\tau) \\
&\quad\quad q_1 = q_0 + \tau\left(H^A + a_{1,1}H_1 + a_{1,2}H_2 + a_{1,3}H_3\right)p_0 \\
&\quad \textbf{do } i = 1, m-1 \\
&\quad\quad p_i = p_{i-1} - \tau\left(b_{i,1}H_1 + b_{i,2}H_2 + b_{i,3}H_3\right)q_i \\
&\quad\quad q_{i+1} = q_i + \tau\left(a_{i+1,1}H_1 + a_{i+1,2}H_2 + a_{i+1,3}H_3\right)p_i \\
&\quad \textbf{enddo} \\
&\quad p_m = p_{m-1} - \tau\left(b_{m,1}H_1 + b_{m,2}H_2 + b_{m,3}H_3\right)q_m \\
&\quad\quad H^A = a_{m+1,1}H_1 + a_{m+1,2}H_2 + a_{m+1,3}H_3 \\
&\textbf{enddo} \\
&q_{m+1} = q_m + \tau H^A p_m \\
&u_K = q_{m+1} + i p_m
\end{aligned} \tag{17}$$

Recall that the exact solution operator is time-symmetric, i.e., it satisfies $K(t+\tau, -\tau) \equiv K(t,\tau)^{-1}$. For this reason, we will focus only on schemes (15) and (16) satisfying the same symmetry, that is, $\widetilde{K}(t+\tau,-\tau) = \widetilde{K}(t,\tau)^{-1}$. This is automatically achieved if the coefficients verify the following conditions:

$$c_{k-j+1} = 1 - c_j, \quad a_{m+2-i,k+1-j} = a_{i,j}, \quad b_{m+1-i,k+1-j} = b_{i,j} \tag{18}$$

for $j = 1, \ldots, k$, $i = 1, 2, \ldots, m$. The integration schemes (15) and (16) are of order r if

$$\widetilde{K}(t,\tau) - K(t,\tau) = \mathcal{O}(\tau^{r+1}) \quad \text{as} \quad \tau \to 0. \tag{19}$$

As is well known, time-symmetric integration schemes are of even order.

C. New schemes of order 4 and 6

We present next several time-symmetric schemes (15) and (16) of orders 4 and 6. In all of them, $n = 3$ and the coefficients $c_j, j = 1, 2, 3$, correspond to nodes of the Gauss-Legendre (GL) quadrature rule of order 6, that is,

$$c_1 = \frac{1}{2} - \frac{\sqrt{15}}{10}, \qquad c_2 = \frac{1}{2}, \qquad c_3 = \frac{1}{2} + \frac{\sqrt{15}}{10}. \tag{20}$$

The remaining coefficients $\{a_{ij}, b_{ij}\}$ must be determined to guarantee that (19) holds for $r = 6$. As a matter of fact, condition (19) translates into a set of polynomial equations on the coefficients (the so-called *order conditions*), whose solutions provide actual integration schemes. These equations have been obtained in Ref. 20 and are collected, for completeness, in a more simplified but equivalent form in Subsection 1 of the Appendix.

The process of obtaining appropriate values for the coefficients a_{ij}, b_{ij} is made simpler by noticing that the schemes (15) and (16) reduce in fact to (12) when applied to a constant matrix $H(t) = \widetilde{H}$ as long as

$$a_i = \sum_{j=1}^{n} a_{i,j}, \qquad b_i = \sum_{j=1}^{n} b_{i,j}. \tag{21}$$

It is then clear that a necessary condition for schemes (15) and (16), to be of order r for arbitrary smooth matrices $H(t)$, is that (12) with coefficients (21) be an approximation of order r to the solution operator of system (10). In the particular case of 6th-order time-symmetric methods, such conditions are the first six order conditions listed in Subsection 1 of the Appendix, which only depend on the coefficients a_i, b_i. Hence, a convenient starting point in the construction process of 6th-order integrators of the form (15) and (16) for Equation (14) is to design previously a symplectic splitting method of the form (12) for the corresponding autonomous problem (10). In other words, to determine appropriate values for the coefficients (11) satisfying the first six order conditions given in Subsection 1 of the Appendix.

Once this is done, one has to solve the remaining 18 order conditions and (21) for the coefficients $a_{i,j}, b_{i,j}$. We have explored compositions (12) ranging from 8 to 24 stages and different orders, and tried to obtain time-symmetric schemes (15) and (16) based on them. In many cases, the remaining order conditions cannot be solved for real valued coefficients $a_{i,j}$ and $b_{i,j}$ satisfying (21). In particular, all the 6th-order schemes (12)

with $m = 10$ stages we have analyzed lead to complex coefficients $a_{i,j}$ and $b_{i,j}$. The most efficient schemes we have found are time-symmetric compositions of order 6 involving $m = 11$ stages. We have chosen as a representative the following:

- An 11-stage scheme, denoted in the sequel as $\mathrm{SM}_{11}^{[6]}$.
- A second 11-stage scheme whose coefficients provide in the autonomous case an 8th-order approximation. It will be denoted as $\mathrm{SM}_{11}^{[8]}$.

We have constructed other compositions with more stages, but they turn out to be clearly less efficient than $\mathrm{SM}_{11}^{[6]}$ and $\mathrm{SM}_{11}^{[8]}$ on all the numerical tests (not reported here) we have carried out with them. For completeness, we have also considered 8-stage schemes (15) and (16) of order 4 that is of order 6 for constant $H(t) = \widetilde{H}$, which we denote as $\mathrm{SM}_{8}^{[4]}$. The coefficients of the three proposed integrators are collected in Subsection 2 of the Appendix. The same coefficients with more significant digits as well as their implementation in several examples can be found in Ref. 30.

Although only schemes with coefficients c_j, $j = 1, 2, 3$ corresponding to the nodes (20) of the 6th-order Gauss-Legendre quadrature rule, are considered here, other choices are of course possible. In Subsection 3 of the Appendix, we provide the necessary transformations to adapt the present schemes to any other quadrature rule of order $r \geq 6$.

It is worth remarking that the m-stage splitting schemes presented here are particularly advantageous with respect to other time integrators (such as those considered in Section III) when the computational cost of evaluating the product $H(t_1)u$ is essentially the same as the cost of the product $(H(t_1) + H(t_2))u$. This is certainly the case when the Hamiltonian is of the form

$$H(t) = \sum_{i=1}^{s} f_i(t) H_i$$

with H_i constant and $s \ll N$, as occurs in many applications. Notice that the algorithm requires $2m$ products with real vectors, and this is equivalent to m products with complex vectors (and so roughly the same cost as approximating the action of the exponential by a polynomial of degree m). In addition, only one new real vector, v, needs to be kept in memory.

III. NUMERICAL ILLUSTRATIONS

We now illustrate the performance of the proposed methods on a pair of academic examples, namely, a high dimensional generalization of the Rosen–Zener model and the well-known Walker–Preston of a diatomic molecule. Although admittedly simple, they nevertheless represent adequately many typical applications and thus may serve as an indicative test bench of the methods proposed in this paper.

In our experiments, we compare with other well known schemes adapted to the linear problem (14). In particular, and guided by the conclusions of the analysis carried out in Ref. 2, we consider some highly efficient symplectic partitioned Runge-Kutta (splitting) methods. For our first example, we also include the efficiency diagrams obtained with a pair of explicit Runge–Kutta schemes as a standard reference. Specifically, the following integrators are taken for the numerical experiments:

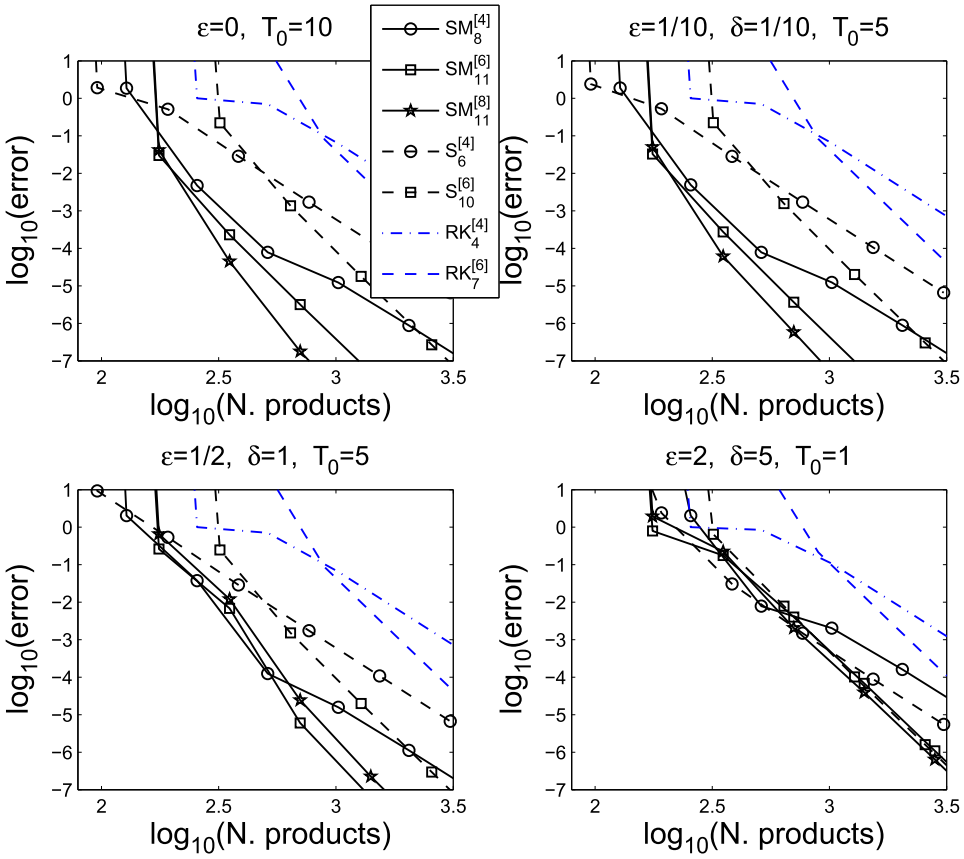

FIG. 1. Two-norm error in the evolution operator at the final time for the high dimensional Rosen–Zener model (23) in double logarithmic scale.

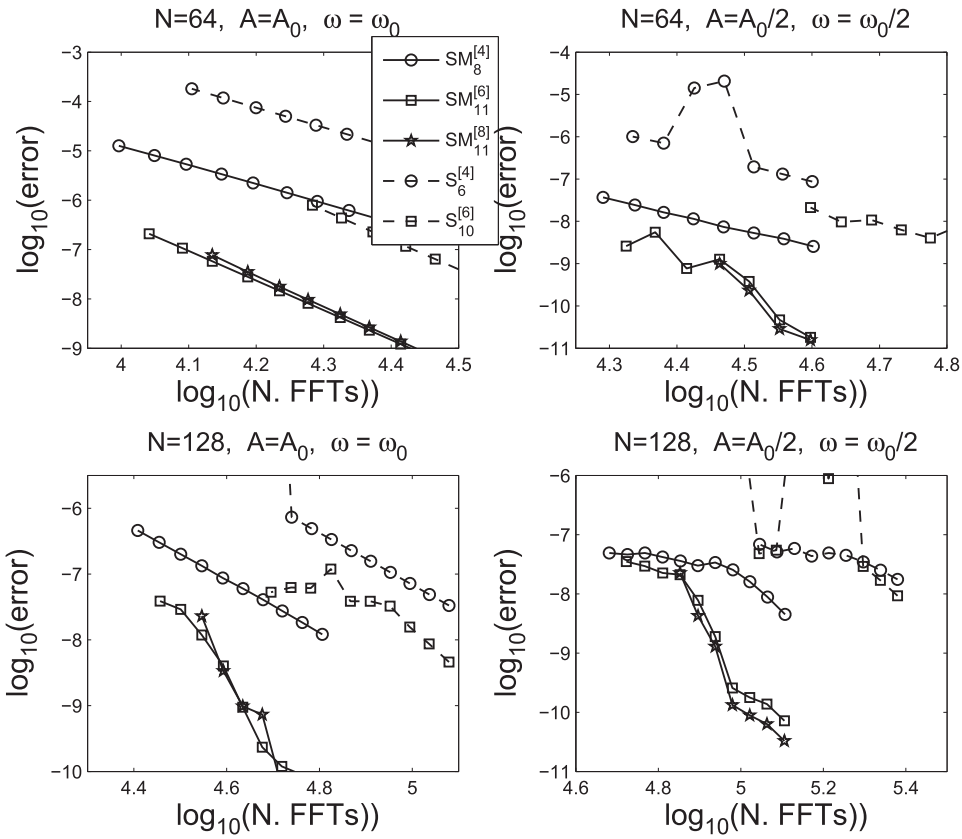

FIG. 2. Two-norm error in the vector solution of the discretized one-dimensional SE (24) versus the number of FFT calls in double logarithmic scale.

- $\mathrm{RK}_4^{[4]}$: The standard 4-stage 4th-order explicit Runge–Kutta method.
- $\mathrm{RK}_7^{[6]}$: The 7-stage 6th-order explicit RK method that uses the Lobatto quadrature rule at internal stages to advance the time.
- $\mathrm{S}_6^{[4]}$ and $\mathrm{S}_{10}^{[6]}$: The 6-stage 4th-order and the 10-stage 6th-order splitting methods, respectively, designed in Ref. 31 for general separable systems and adapted to the present problem by considering the time as two new dependent variables as proposed in Ref. 13. In this way, the linear problem (14) is transformed into a nonlinear autonomous system. One step of these schemes is evaluated as follows:

$$\begin{aligned} &\textbf{do } i = 1, m \\ &\quad q_i = q_{i-1} + \tau\, a_i H(t_k + e_i\tau) \\ &\quad p_i = p_{i-1} - \tau\, b_i H(t_k + d_i\tau) q_i \\ &\textbf{enddo} \end{aligned} \tag{22}$$

  with $d_i = \sum_{j=1}^{i} a_j$, $e_i = \sum_{j=0}^{i-1} b_j$ and $b_0 = 0$. These methods are independent of the Hamiltonian structure and have been tested in Ref. 2, showing a high performance.

- The new schemes $\mathrm{SM}_8^{[4]}$ (order 4), $\mathrm{SM}_{11}^{[6]}$, and $\mathrm{SM}_{11}^{[8]}$ (order 6).

In all cases, the computational cost of the algorithms is estimated as the number of products required to produce the final result.

A. A high dimensional Rosen–Zener model

For our first illustration, we consider a generalization of the well known Rosen–Zenner model for a quantum system of two levels[32] closely related to the model studied in Ref. 33. Specifically, the system is directly formulated as (4) with

$$H(t) = \omega(t)\, \sigma_3 \otimes I_k + V(t)\, \sigma_1 \otimes D^{[k]}. \tag{23}$$

Here I_k is the $k \times k$ identity matrix, $D^{[k]}$ is a tridiagonal symmetric matrix with entries $D^{[k]}_{i,i} = 0$, $D^{[k]}_{i+1,i} = D^{[k]}_{i,i+1} = 1$, $i = 1, \ldots, k-1$, $D^{[k]}_{k,k} = 0$, $\omega(t)$ is a time-dependent function, and σ_i are the Pauli matrices.

We take $N = 2k = 20$, $\omega(t) = w_0 + \varepsilon \cos(\delta t)$, $V(t) = V_0/\cosh(t/T_0)$, and $w_0 = 5$, $V_0 = 1/2$ and integrate along the interval $t \in [t_0, t_f]$ with $t_0 = -2$ and $t_f = t_0 + 8\pi$. We study the performance of the methods in the following cases:

(i) $\varepsilon = 0$, $T_0 = 10$.
(ii) $\varepsilon = \frac{1}{10}$, $\delta = 1/10$, $T_0 = 5$.
(iii) $\varepsilon = \frac{1}{2}$, $\delta = 1$, $T_0 = 5$.
(iv) $\varepsilon = 2$, $\delta = 5$, $T_0 = 1$.

The results obtained are shown in Figure 1. We observe that the new methods show the best performance and their superiority increases when the explicitly time-dependent functions are smooth. In that case, the main error originates from terms associated with the splitting method for the autonomous case, and the new methods are optimized for this case.

B. The Walker–Preston model

This constitutes a standard model for a diatomic molecule in a strong laser field.[34] The system is described by the one-dimensional Schrödinger equation (in units such that $\hbar = 1$)

$$i\frac{\partial}{\partial t}\psi(x,t) = \left(-\frac{1}{2\mu}\frac{\partial^2}{\partial x^2} + V(x) + f(t)x\right)\psi(x,t) \tag{24}$$

with $\psi(x,0)=\psi_0(x)$. Here $V(x)=D(1-\mathrm{e}^{-\alpha x})^2$ is the Morse potential and $f(t)x=A\cos(\omega(t))x$ accounts for the laser field.

As explained in the Introduction, the usual procedure in its numerical treatment consists of defining the wave function ψ in a certain domain $x\in[x_0,x_N]$ that is subdivided into N parts of length $\Delta x=(x_N-x_0)/N$ with $x_i=x_0+i\Delta x$, and then the vector $u(t)\in\mathbb{C}^N$ with components $u_i=(\Delta x)^{1/2}\psi(x_{i-1},t)$, $i=1,\ldots,N$ is formed. Notice that the 2-norm $\|u(t)\|$ remains constant for all t and, in order to use the Fast Fourier Transform (FFT) algorithm, periodic boundary conditions $\psi(x_0,t)=\psi(x_N,t)$ are assumed.

Here we take the interval $x\in[-0.8,4.32]$, subdivided into $N=64$ and $N=128$ parts of length $\Delta x=0.08$ and $\Delta x=0.04$, respectively, and the parameters are chosen as $\mu=1745$ a.u., $D=0.2251$ a.u., and $\alpha=1.1741$ a.u. (corresponding to the HF molecule). For each choice of N, we take for time-dependent interaction $A=A_0=0.011\,025$ a.u. and laser frequency $\omega=\omega_0=0.017\,87$. The numerical experiments are then repeated with $A=A_0/2$ and $\omega=\omega_0/2$, corresponding to a reduction in the intensity and frequency of the laser. As initial conditions we take the ground state of the Morse potential,

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

with $\gamma=2D/w_0$, $w_0=\alpha\sqrt{2D/\mu}$, and σ is a normalizing constant, and integrate over the time interval $t\in[0,10\,T_0]$ with $T_0=2\pi/\omega$ (the choice $\omega=\omega_0/2$ requires a twice longer time integration).

To check accuracy, we measure the two-norm error in the solution at the final time, $u(10\,T_0)$. As usual, the exact solution is accurately approximated using a sufficiently small time step.

Figure 2 shows the efficiency plots for the methods. The largest time step (i.e., the smaller number of FFTs) corresponds to the stability limit of the method (an overflow or an exceedingly large error appears when the time step is slightly increased). We observe that the relative performance of the new methods increases both when taking a finer mesh or when the time-dependent interaction is weaker and smoother. We observe that in such limit, the 6th-order method that corresponds to an 8th-order one in the autonomous case shows the best performance when high accuracies are desired. The 4th-order method shows less accuracy but allows for longer time steps and more frequent outputs.

IV. CONCLUDING REMARKS

We have presented several symplectic splitting methods for the numerical integration in time of the Schrödinger equation previously discretized in space when the Hamiltonian is an explicitly time-dependent real operator. Our approach differs from others existing in the literature in that we do not need to impose beforehand a slow time-dependence in the potential so that it can be taken as approximately constant in each time subinterval along the integration. It is based on the fact that the semidiscretized equation can be formulated as a classical linear Hamiltonian system for which the solution operator can be approximated as a product of matrices involving linear combinations of the (semidiscretized) time-dependent Hamiltonian on quadrature nodes at each time step. Here we have considered methods of orders four and six based on the Gauss–Legendre quadrature rule of order six, but the extension to other quadrature rules (as shown in Subsection 3 of the Appendix) is rather straightforward. Although originally intended for the numerical integration of the Schrödinger equation with a time-dependent potential, the new methods are still valid for arbitrary linear ODE systems of the form $\dot u=-iH(t)u$ for an arbitrary real matrix $H(t)$ provided its time dependence on t is reasonably smooth. Moreover, the proposed method could also be used to solve systems like $x'=M(t)y$, $y'=N(t)x$ arising, e.g., in elastic waves analysis.[35]

ACKNOWLEDGMENTS

The authors acknowledge Ministerio de Economía y Competitividad (Spain) for financial support through Project Nos. MTM2013-46553-C3 and MTM2016-77660-P (AEI/FEDER, UE). Additionally, A.M. has been partially supported by the Basque Government (Consolidated Research Group No. IT649-13).

APPENDIX: ON THE CONSTRUCTION OF METHODS

1. Order conditions

To derive the order conditions for the schemes (15) and (16) with $n=3$ and the coefficients c_j given by (20), it is advantageous to first write (15) in terms of a new set of variables in order to avoid redundancies and limit the presence of contributions at orders higher than six. Specifically, instead of considering the matrices $H_i\equiv H(t+c_i\tau)$, $i=1,2,3$, so that $\tau H_i=\mathcal{O}(\tau)$, we introduce the variables

$$\begin{aligned}\gamma_1&\equiv\tau H_2=\mathcal{O}(\tau),\\ \gamma_2&\equiv\frac{\sqrt{15}}{3}\tau(H_3-H_1)=\mathcal{O}(\tau^2),\\ \gamma_3&\equiv\frac{10}{3}\tau(H_1-2H_2+H_3)=\mathcal{O}(\tau^3).\end{aligned}\tag{A1}$$

As a matter of fact, the γ_i are the $(i-1)$th derivatives, evaluated at the midpoint $t+\tau/2$ and divided by $(i-1)!$, of a matrix that interpolates $H(t)$ at the nodes. Then, schemes (15) and (16) can be expressed as

$$\begin{aligned}\widetilde K(t_k,\tau)&=\prod_{i=1}^{m+1}\begin{pmatrix}I&0\\-\tau(b_{i,1}H_1+b_{i,2}H_2+b_{i,3}H_3)&I\end{pmatrix}\begin{pmatrix}I&\tau(a_{i,1}H_1+a_{i,2}H_2+a_{i,3}H_3)\\0&I\end{pmatrix}\\&=\prod_{i=1}^{m+1}\begin{pmatrix}I&0\\-(b_i\gamma_1+b_i'\gamma_2+b_i''\gamma_3)&I\end{pmatrix}\begin{pmatrix}I&(a_i\gamma_1+a_i'\gamma_2+a_i''\gamma_3)\\0&I\end{pmatrix},\end{aligned}\tag{A2}$$

with $b_{m+1,j} = b_{m+1} = b'_{m+1} = b'_{m+1} = 0$, $j = 1,2,3$. In the autonomous case, we have that $\gamma_1 = \tau\widetilde{H}$, $\gamma_2 = \gamma_3 = 0$ and the composition (12) is recovered. If the composition satisfies the symmetry conditions (18), the scheme will be of order 6 when applied to a system of the form (14) whenever the following conditions are satisfied (see Ref. 20 for the derivation of equivalent conditions for symmetric schemes (12) of order 6):

$$\sum_{i=1}^{m+1} a_i = 1, \qquad \sum_{i=1}^{m} b_i = 1,$$
$$\sum_{m+1\geq i_3>i_2>i_1\geq 1} a_{i_3} b_{i_2} a_{i_1} = \frac{1}{6}, \qquad \sum_{m\geq i_3>i_2>i_1\geq 1} b_{i_3} a_{i_2} b_{i_1} = \frac{1}{6},$$
$$\sum_{m+1\geq i_5>i_4>i_3>i_2>i_1\geq 1} a_{i_5} b_{i_4} a_{i_3} b_{i_2} a_{i_1} = \frac{1}{120},$$
$$\sum_{m\geq i_5>i_4>i_3>i_2>i_1\geq 1} b_{i_5} a_{i_4} b_{i_3} a_{i_2} b_{i_1} = \frac{1}{120},$$

$$\sum_{i=1}^{m+1} a''_i = \frac{1}{12}, \qquad \sum_{i=1}^{m} b''_i = \frac{1}{12},$$
$$\sum_{m+1\geq i_2>i_1\geq 1} a'_{i_2} b_{i_1} = \frac{1}{12}, \qquad \sum_{m\geq i_2>i_1\geq 1} b'_{i_2} a_{i_1} = \frac{1}{12},$$

$$\sum_{m+1\geq i_2>i_1\geq 1} a''_{i_2} b'_{i_1} = -\frac{1}{240}, \qquad \sum_{m\geq i_2>i_1\geq 1} b''_{i_2} a'_{i_1} = -\frac{1}{240},$$
$$\sum_{m+1\geq i_3>i_2>i_1\geq 1} a_{i_3} b''_{i_2} a_{i_1} = \frac{1}{120}, \qquad \sum_{m\geq i_3>i_2>i_1\geq 1} b_{i_3} a''_{i_2} b_{i_1} = \frac{1}{120},$$
$$\sum_{m+1\geq i_3>i_2>i_1\geq 1} a_{i_3} b_{i_2} a''_{i_1} = \frac{1}{60}, \qquad \sum_{m\geq i_3>i_2>i_1\geq 1} b_{i_3} a_{i_2} b''_{i_1} = \frac{1}{60},$$
$$\sum_{m+1\geq i_3>i_2>i_1\geq 1} a_{i_3} b'_{i_2} a'_{i_1} = \frac{1}{240}, \qquad \sum_{m\geq i_3>i_2>i_1\geq 1} b_{i_3} a'_{i_2} b'_{i_1} = \frac{1}{240},$$
$$\sum_{m+1\geq i_3>i_2>i_1\geq 1} a'_{i_3} b_{i_2} a'_{i_1} = -\frac{1}{120}, \qquad \sum_{m\geq i_3>i_2>i_1\geq 1} b'_{i_3} a_{i_2} b'_{i_1} = -\frac{1}{120},$$
$$\sum_{m+1\geq i_3>i_2>i_1\geq 1} a_{i_4} b_{i_3} a_{i_2} b'_{i_1} = -\frac{1}{80}, \qquad \sum_{m\geq i_4>i_3>i_2>i_1\geq 1} b_{i_4} a_{i_3} b_{i_2} a'_{i_1} = -\frac{1}{80},$$
$$\sum_{m+1\geq i_3>i_2>i_1\geq 1} a_{i_4} b_{i_3} a'_{i_2} b_{i_1} = -\frac{1}{240}, \qquad \sum_{m\geq i_4>i_3>i_2>i_1\geq 1} b_{i_4} a_{i_3} b'_{i_2} a_{i_1} = \frac{1}{240}.$$

Here, from (A1) and (A2), one has

$$\begin{pmatrix} a_{i,1} & b_{i,1} \\ a_{i,2} & b_{i,2} \\ a_{i,3} & b_{i,3} \end{pmatrix} = \begin{pmatrix} 0 & -\sqrt{\frac{5}{3}} & \frac{10}{3} \\ 1 & 0 & -\frac{20}{3} \\ 0 & \sqrt{\frac{5}{3}} & \frac{10}{3} \end{pmatrix} \begin{pmatrix} a_i & b_i \\ a'_i & b'_i \\ a''_i & b''_i \end{pmatrix}. \tag{A3}$$

2. Coefficients of the methods

Next we collect the coefficients of the symplectic splitting methods proposed in this work for the numerical integration of Equation (14). For the sake of clarity, we write them in matrix form. The same coefficients, with more digits, are collected in Ref. 30. All these coefficients have to be supplemented by the nodes c_j given in (20).

a. Method $SM_8^{[4]}$

$$a = \begin{pmatrix} 0.056\,543\,643\,80 & 0.013\,657\,706\,809 & -0.034\,367\,547\,779 \\ 0.151\,876\,511\,53 & -0.066\,217\,362\,266 & 0.075\,469\,735\,351 \\ 0.074\,446\,942\,50 & 0.208\,318\,930\,216 & -0.042\,763\,740\,386 \\ -0.011\,712\,456\,09 & -0.002\,171\,489\,464 & 0.008\,284\,688\,848 \\ 0 & 0.137\,268\,873\,853 & 0 \\ a_{4,3} & a_{4,2} & a_{4,1} \\ a_{3,3} & a_{3,2} & a_{3,1} \\ a_{2,3} & a_{2,2} & a_{2,1} \\ a_{1,3} & a_{1,2} & a_{1,1} \end{pmatrix},$$

$$b = \begin{pmatrix} 0.109\,165\,185\,01 & 0.013\,444\,365\,00 & -0.019\,256\,622\,788 \\ 0.188\,194\,699\,07 & -0.039\,707\,697\,39 & 0.058\,395\,655\,885 \\ 0.141\,154\,416\,25 & 0.570\,642\,655\,82 & -0.043\,047\,573\,981 \\ -0.100\,060\,196\,70 & -0.322\,157\,101\,21 & -0.056\,767\,784\,980 \\ b_{4,3} & b_{4,2} & b_{4,1} \\ b_{3,3} & b_{3,2} & b_{3,1} \\ b_{2,3} & b_{2,2} & b_{2,1} \\ b_{1,3} & b_{1,2} & b_{1,1} \end{pmatrix}.$$

b. Method $SM_{11}^{[6]}$

$$a = \begin{pmatrix} 0.064\,229\,455\,56 & -0.025\,341\,566\,51 & 0.007\,599\,565\,74 \\ -0.044\,424\,869\,66 & -0.024\,510\,802\,57 & 0.008\,244\,001\,06 \\ 0.205\,475\,526\,18 & 0.021\,439\,623\,03 & -0.008\,448\,622\,75 \\ 0.066\,283\,190\,90 & 0.122\,950\,863\,12 & -0.021\,180\,474\,54 \\ -0.003\,551\,371\,84 & 0.314\,392\,364\,17 & -a_{4,1} \\ 0.003\,488\,712\,73 & -0.186\,708\,253\,74 & -a_{5,1} \\ a_{6,3} & a_{6,2} & a_{6,1} \\ a_{5,3} & a_{5,2} & a_{5,1} \\ a_{4,3} & a_{4,2} & a_{4,1} \\ a_{3,3} & a_{3,2} & a_{3,1} \\ a_{2,3} & a_{2,2} & a_{2,1} \\ a_{1,3} & a_{1,2} & a_{1,1} \end{pmatrix},$$

$$b = \begin{pmatrix} 0.198\,931\,884\,48 & -0.016\,617\,016\,61 & 0.002\,015\,615\,63 \\ -0.030\,836\,241\,53 & -0.011\,909\,451\,58 & 0.001\,688\,789\,82 \\ 0.079\,650\,985\,44 & 0.044\,994\,246\,37 & 0.009\,110\,447\,84 \\ 0.082\,864\,339\,33 & 0.186\,548\,251\,04 & -0.065\,648\,043\,24 \\ 0.012\,909\,944\,48 & -0.011\,760\,166\,91 & -b_{5,1} \\ 0 & 0.061\,932\,719\,82 & 0 \\ b_{5,3} & b_{5,2} & b_{5,1} \\ b_{4,3} & b_{4,2} & b_{4,1} \\ b_{3,3} & b_{3,2} & b_{3,1} \\ b_{2,3} & b_{2,2} & b_{2,1} \\ b_{1,3} & b_{1,2} & b_{1,1} \end{pmatrix}.$$

c. Method $SM_{11}^{[8]}$

$$a = \begin{pmatrix} 0.078\,539\,048\,50 & -0.055\,358\,596\,01 & 0.016\,550\,797\,39 \\ -0.100\,359\,598\,31 & -0.159\,313\,968\,90 & 0.045\,184\,103\,41 \\ 0.217\,485\,576\,77 & 0.209\,628\,140\,57 & -0.058\,916\,052\,51 \\ 0.083\,044\,197\,90 & 0.104\,202\,478\,18 & -0.019\,749\,426\,81 \\ 0.007\,699\,808\,87 & 0.290\,721\,833\,65 & 0.097\,739\,158\,98 \\ 0.038\,370\,734\,11 & -0.167\,657\,665\,27 & -0.127\,810\,570\,54 \\ a_{6,3} & a_{6,2} & a_{6,1} \\ a_{5,3} & a_{5,2} & a_{5,1} \\ a_{4,3} & a_{4,2} & a_{4,1} \\ a_{3,3} & a_{3,2} & a_{3,1} \\ a_{2,3} & a_{2,2} & a_{2,1} \\ a_{1,3} & a_{1,2} & a_{1,1} \end{pmatrix},$$

$$b = \begin{pmatrix} 0.154\,171\,842\,09 & -0.018\,426\,075\,45 & 0.007\,131\,251\,77 \\ -0.006\,347\,779\,03 & -0.003\,890\,654\,05 & 0.001\,097\,338\,89 \\ 0.113\,616\,267\,16 & 0.024\,959\,240\,32 & -0.013\,594\,387\,63 \\ 0.034\,287\,009\,27 & 0.201\,283\,836\,52 & -0.008\,359\,528\,72 \\ -0.000\,119\,805\,05 & -0.008\,521\,705\,59 & b_{5,1} \\ -0.003\,984\,625\,91 & 0.053\,635\,160\,96 & b_{6,1} \\ b_{5,3} & b_{5,2} & b_{5,1} \\ b_{4,3} & b_{4,2} & b_{4,1} \\ b_{3,3} & b_{3,2} & b_{3,1} \\ b_{2,3} & b_{2,2} & b_{2,1} \\ b_{1,3} & b_{1,2} & b_{1,1} \end{pmatrix}.$$

3. Schemes with nodes of an arbitrary quadrature rule

Methods (15) and (16) can be used with any other quadrature rule of order $r \geq 6$, say $\{\hat{w}_i, \hat{c}_i\}_{i=1}^k$. All that is required is to replace in (17) the nodes c_i, $i = 1, 2, 3$ by $\hat{c}_i$, $i = 1, \ldots, k$, and the coefficients $\{a_{i,j}, b_{i,j}\}$ by the new $\{\hat{a}_{i,j}, \hat{b}_{i,j}\}$, so that the scheme reads

$$\widehat{K}(t_k, \tau) = \prod_{i=1}^{m+1} \begin{pmatrix} I & 0 \\ -\tau \sum_{i=1}^k \hat{b}_{i,1} \hat{H}_i & I \end{pmatrix} \begin{pmatrix} I & \tau \sum_{i=1}^k \hat{a}_{i,1} \hat{H}_i \\ 0 & I \end{pmatrix}$$

with $\hat{H}_i = H(t + \hat{c}_i \tau)$, $i = 1, \ldots, k$. The new coefficients $\{\hat{a}_{i,j}, \hat{b}_{i,j}\}$ are chosen such that

$$\widehat{K}(t_k, \tau) = \widetilde{K}(t_k, \tau) + \mathcal{O}(\tau^7)$$

but this is satisfied if

$$\hat{a} = a\, Q^{-1} \widehat{Q}, \qquad \hat{b} = b\, Q^{-1} \widehat{Q}. \tag{A4}$$

Here, $a \in \mathbb{R}^{(m+1)\times 3}$, $b \in \mathbb{R}^{m\times 3}$, $\hat{a} \in \mathbb{R}^{(m+1)\times k}$, $\hat{b} \in \mathbb{R}^{m\times k}$ denote the matrices with coefficients $a_{i,j}, b_{i,j}, \hat{a}_{i,j}, \hat{b}_{i,j}$, respectively, and the elements of the matrices $Q \in \mathbb{R}^{3\times 3}$, $\widehat{Q} \in \mathbb{R}^{3\times k}$ are given by

$$Q_{i,j} = w_j \left(c_j - \frac{1}{2}\right)^{i-1}, \quad \widehat{Q}_{i,l} = \hat{w}_l \left(\hat{c}_k - \frac{1}{2}\right)^{i-1},$$

$i, j = 1, 2, 3$, $l = 1, \ldots, k$, with

$$(c_1, c_2, c_3) = \left(\frac{1}{2} - \frac{\sqrt{15}}{10}, \frac{1}{2}, \frac{1}{2} + \frac{\sqrt{15}}{10}\right),$$

$$(w_1, w_2, w_3) = \frac{1}{18}(5, 8, 5).$$

As an illustration, suppose $H(t)$ is known only at equidistant times. Then we can use the 6h-order Newton–Cotes quadrature rule with $\hat{c}_i = (i-1)/4$, $i = 1, \ldots, 5$,

$$(\hat{w}_1, \hat{w}_2, \hat{w}_3, \hat{w}_4, \hat{w}_5) = \frac{1}{90}(7, 32, 12, 32, 7),$$

so that

$$Q^{-1}\widehat{Q} = \begin{pmatrix} \frac{7}{150}(5+\sqrt{15}) & \frac{4}{75}(5+2\sqrt{15}) & 0 & \frac{4}{75}(5-2\sqrt{15}) & \frac{7}{150}(5-\sqrt{15}) \\ -\frac{7}{60} & \frac{7}{15} & \frac{3}{10} & \frac{7}{15} & -\frac{7}{60} \\ \frac{7}{150}(5-\sqrt{15}) & \frac{4}{75}(5-2\sqrt{15}) & 0 & \frac{4}{75}(5+2\sqrt{15}) & \frac{7}{150}(5+\sqrt{15}) \end{pmatrix}.$$

Then the corresponding coefficients for schemes $\mathrm{SM}_8^{[4]}$, $\mathrm{SM}_{11}^{[6]}$, and $\mathrm{SM}_{11}^{[8]}$ can be obtained (from those collected in Subsection 2 of the Appendix) by (A4).

We have carried out the experiments of Section III with the present symplectic splitting methods but using different quadrature rules. Our experiments (not reported here) show that the plots are essentially identical when considering the equidistant 6th-order Newton-Cotes or the 8th-order GL quadrature rule. If, on the other hand, the 4th-order GL quadrature rule is used instead, the new 4th-order method remains basically the same, but the 6th-order methods lose some accuracy, as expected.

[1] A. Castro, M. Marques, and A. Rubio, J. Chem. Phys. **121**, 3425 (2004).
[2] K. Kormann, S. Holmgren, and H. Karlsson, J. Chem. Phys. **128**, 184101 (2008).
[3] D. Poulin, A. Qarry, R. Somma, and F. Verstraete, Phys. Rev. Lett. **106**, 170501 (2011).
[4] C. Lubich, *From Quantum to Classical Molecular Dynamics: Reduced Models and Numerical Analysis* (European Mathematical Society, 2008).
[5] T. Jahnke and C. Lubich, BIT Numer. Math. **40**, 735 (2000).
[6] C. Neuhauser and M. Thalhammer, BIT Numer. Math. **49**, 199 (2009).
[7] M. Thalhammer, SIAM J. Numer. Anal. **46**, 2022 (2008).
[8] M. Thalhammer, SIAM. J. Numer. Anal. **50**, 3231 (2012).
[9] T. Park and J. Light, J. Chem. Phys. **85**, 5870 (1986).
[10] S. Blanes, F. Casas, and A. Murua, J. Comput. Phys. **303**, 396 (2015).
[11] M. Feit, J. A. Fleck, Jr., and A. Steiger, J. Comput. Phys. **47**, 412 (1982).
[12] J. Tremblay and T. Carrington, Jr., J. Chem. Phys. **121**, 11535 (2004).
[13] J. Sanz-Serna and A. Portillo, J. Chem. Phys. **104**, 2349 (1996).
[14] U. Peskin, R. Kosloff, and N. Moiseyev, J. Chem. Phys. **100**, 8849 (1994).
[15] D. Lauvergnat, S. Blasco, and X. Chapuisat, J. Chem. Phys. **126**, 204103 (2007).
[16] H. Tal-Ezer, R. Kosloff, and C. Cerjan, J. Comput. Phys. **100**, 179 (1992).
[17] M. Ndong, H. Tal-Ezer, R. Kosloff, and C. Koch, J. Chem. Phys. **132**, 064105 (2010).

[18]H. Tal-Ezer, R. Kosloff, and I. Schaefer, J. Sci. Comput. **53**, 211 (2012).
[19]I. Schaefer, H. Tal-Ezer, and R. Kosloff, preprint arXiv:1611.06707 (2016).
[20]S. Blanes, F. Casas, and A. Murua, Int. J. Comput. Math. **84**, 713 (2007).
[21]S. Blanes, F. Casas, and A. Murua, Rev. R. Acad. Cienc. Exactas, Fis. Nat. **106**, 49 (2012).
[22]S. Gray and D. Manolopoulos, J. Chem. Phys. **104**, 7099 (1996).
[23]S. Gray and J. Verosky, J. Chem. Phys. **100**, 5011 (1994).
[24]S. Blanes, F. Casas, and A. Murua, J. Chem. Phys. **124**, 234105 (2006).
[25]S. Blanes, F. Casas, and A. Murua, Found. Comput. Math. **8**, 357 (2008).
[26]S. Blanes, F. Casas, and A. Murua, SIAM J. Sci. Comput. **33**, 1525 (2011).
[27]S. Blanes and F. Casas, *A Concise Introduction to Geometric Numerical Integration, Monographs and Research Notes in Mathematics* (CRC Press, 2016).
[28]E. Hairer, C. Lubich, and G. Wanner, *Geometric Numerical Integration*, Structure-Preserving Algorithms for Ordinary Differential Equations, 2nd ed. (Springer-Verlag, 2006).
[29]J. M. Sanz-Serna and M. P. Calvo, *Numerical Hamiltonian Problems*, AMMC 7 (Chapman & Hall, 1994).
[30]See http://www.gicas.uji.es/Research/TD-propagators.html for Geometric integration Research Group.
[31]S. Blanes and P. Moan, J. Comput. Appl. Math. **142**, 313 (2002).
[32]N. Rosen and C. Zener, Phys. Rev. **40**, 502 (1932).
[33]E. Kyoseva, N. Vitanova, and B. Shore, J. Mod. Opt. **54**, 2237 (2007).
[34]R. Walker and K. Preston, J. Chem. Phys. **67**, 2017 (1977).
[35]X. Li, W. Wang, M. Lu, M. Zhang, and Y. Li, Geophys. J. Int. **188**, 1382 (2012).