High Order Efficient Splittings for the Semiclassical Time–Dependent Schrödinger Equation

Sergio Blanes · Vasile Gradinaru

Abstract Standard numerical schemes with time–step h deteriorate (e.g. like $\varepsilon^{-2}h^2$) in the presence of a small semiclassical parameter ε in the time–dependent Schrödinger equation. The recently introduced semiclassical splitting was shown to be of order $\mathcal{O}\left(\varepsilon h^2\right)$. We present now an algorithm that is of order $\mathcal{O}\left(\varepsilon h^7 + \varepsilon^2 h^6 + \varepsilon^3 h^4\right)$ at the expense of roughly three times the computational effort of the semiclassical splitting and another that is of order $\mathcal{O}\left(\varepsilon h^6 + \varepsilon^2 h^4\right)$ at the *same* expense of the computational effort of the semiclassical splitting.

Keywords semiclassical · time–dependent Schrödinger equation · splitting · wavepackets

PACS 02.60.Jh · 02.60.Cb 02.70.Hm · 03.65.Sq

Mathematics Subject Classification (2000) 65M12 · 65Z05 · 65P10 · 81-08 · 81Q05 · 81Q20

The work of SB has been funded by Ministerio de Economía, Industria y Competitividad (Spain) through project MTM2016-77660-P (AEI/FEDER, UE)

1 Introduction

The standard techniques for the numerical solution of the time–dependent Schrödinger equation in its semiclassical formulation[1]

$$i\,\varepsilon^2\,\partial_t\psi \;=\; H(\varepsilon)\,\psi \tag{1}$$

were known to be able to give only convergence of order $\mathcal{O}\left(h^k/\varepsilon^2\right)$, see [3,2, 12], which forces the use of small time-steps h. However, significant progress has been made recently in improving the pessimistic behaviour in ε^2, see [1] and [14]. Here, $\psi = \psi(x,\,t)$ is the wave function that depends on the spatial variables $x = (x_1, \ldots, x_d) \in \mathbb{R}^d$ and the time variable $t \in \mathbb{R}$. The Hamiltonian

$$H(\varepsilon) = -\frac{\varepsilon^4}{2}\,\Delta_x \,+\, V(x)$$

involves the Laplace operator Δ_x and a smooth real potential V. Some applications use physical observables whose values doesn't necessarily need the precise solution and allow techniques that avoid the direct solution of (1), e.g. [19], but other applications such as scattering [16,15], non-adiabatic transitions [9], photodissociation [11,10], vibronic spectra in quantum control [22] need precise knowledge about the wavefunction values in presence of small but fixed parameter ε. In physical chemistry one often use the phrase "exact quantum calculations" for a numerical approximation with Fourier basis in space and a Strang splitting in time, which may be extremely expensive and inefficient at the same time. Hagedorn wavepackets provide a spectral approximation in space with a time-dependent set of basis functions that give the exact solution for the Schrödinger equation with the potential locally approximated by a quadratic function. We show that, when the actual solution is localised and the approximation in space can be done accurately by Hagedorn wavepackets, there are time integrators that are extremely efficient and do not deteriorate for small values of the model parameter ε. In this paper we focus only on the time integration and hence we set $d = 1$, but the computational efficiency is even more evident for higher d as seen in [13].

When the behaviour of the exact solution is semiclassical, i.e. as long as it stays localized in space or in frequency domain, good approximations in terms of ε are possible: concrete assumptions and results are in [17]. This insight has been recently turned into practical numerical methods: [13] sets the foundations of the algorithms and takes advantage of the spectral approximation with Hagedorn wavepackets in $\mathbb{R}^d$ in order to attack higher dimensional problems; [14] defines the semiclassical splitting, which is rigorously proven to be of order $\mathcal{O}\left(\varepsilon\, h^2\right)$. The order of convergence was improved in the same work to $\varepsilon\, h^4$ via a combination of the semiclassical splitting with a Magnus integration step, which triples the computational cost per time step.

[1] Some authors use ε or $\hbar$ where we use ε^2; we prefer this notation in order to avoid fractional powers in the convergence rates and in order to stay consistent with the notation used by George Hagedorn, whose wavepackets are used here for the approximation in space.

While the pessimist behaviour in ε has been reversed, the accuracy needed by applications in physics and chemistry requests a systematic way of raising the convergence order. Theorem 7.2 in [17] stimulates us to search for a numerical method of higher order in ε, too. The combination of the insight given by Lemma 3 and Lemma 4 from [14] with the splitting methods for perturbed systems from [20] and [5] gives a systematic way of improving the semiclassical splitting algorithm, not only in terms of powers of the time-step h, but also in terms of the semiclassical parameter ε. Numerical experiments with our favourite algorithm below give results of order

$$\mathcal{O}\left(\varepsilon h^7 + \varepsilon^2 h^6 + \varepsilon^3 h^4\right) ,$$

while the computational time is roughly three times that of the semiclassical splitting. If we can afford only the computational effort of the semiclassical splitting, then we should use the last algorithm in this paper; it gives results of order

$$\mathcal{O}\left(\varepsilon h^6 + \varepsilon^2 h^4\right)$$

only at the computational cost of the semiclassical splitting. The main idea in order to achieve this efficiency is to use perturbation aware splittings. The fundamental observation is that the right hand side can be seen as the perturbation of an operator whose role in the algorithm is essential but harmless in terms of computational effort.

The next section reviews the ideas of the semiclassical splitting and spots light on the fundamental idea. Then we review some essential facts from the general splitting methods. The processed methods with modified potentials for perturbed systems and the case of the non-autonomous systems for the semiclassical Schrödinger equation are discussed. Finally, we present the two most important arising algorithms together with simulation results for two model problems.

2 Semiclassical splitting

The Hagedorn (semiclassical) wavepackets depending on the parameters $\Pi(t) = (q(t), p(t), Q(t), P(t)) \in \mathbb{R}^d \times \mathbb{R}^d \times \mathbb{C}^{d\times d} \times \mathbb{C}^{d\times d}$ and denoted $\varphi_k^\varepsilon[q, p, Q, P]$, $k = 0, 1, \ldots$ form an orthonormal basis of $L^2(\mathbb{R}^d)$. Since we focus on the space dimension $d = 1$, we can simplify the notations a lot, but the presented techniques are valid for general d. The basis is constructed in [17] with appropriate raising and lowering operators starting from the Gaussian

$$\varphi_0^\varepsilon[q, p, Q, P](x) \;=\; (\pi)^{-\frac{1}{4}} \left(\varepsilon\, Q\right)^{-\frac{1}{2}} \,\exp\left(\frac{i}{2\,\varepsilon^2} P Q^{-1} (x-q)^2 + \frac{i}{\varepsilon^2} p (x-q)\right) ,$$

and has the property that each state $\varphi_k[\Pi](x) = \varphi_k^\varepsilon[q, p, Q, P](x)$ is concentrated in position near q and in momentum near p with uncertainties

$\varepsilon|Q|\sqrt{k+\frac{1}{2}}$ and $\varepsilon|P|\sqrt{k+\frac{1}{2}}$, respectively. The initial condition for (1) is written as a linear combination of Hagedorn wavepackets

$$\psi(0) \;=\; e^{iS(0)/\varepsilon^2} \sum_{k=0}^{K-1} c_k(0,\,\varepsilon)\, \varphi_k[\Pi(0)]\,, \tag{2}$$

and a numerical approximation of the solution is searched in the time-dependent space $\operatorname{span}\{\varphi_0[\Pi(t)],\ldots,\varphi_{K-1}[\Pi(t)]\}$, i.e. of the form

$$e^{iS(t)/\varepsilon^2} \sum_{k=0}^{K-1} c_k(t,\,\varepsilon)\, \varphi_k[\Pi(t)]\,.$$

We decompose the Hamiltonian during one time-step of length h in two parts

$$H = A + W$$

with

$$A = T + U = -\,\frac{\varepsilon^4}{2}\,\Delta_x + U(q(t),x)\,,$$

where $T = -\,\frac{\varepsilon^4}{2}\,\Delta_x$ is the kinetic energy operator, $U(q(t),x)$ is the quadratic Taylor expansion of $V(x)$ around the time dependent classical position $q(t)$ and $W(q(t),x)$ is the corresponding remainder:

$$V(x) \;=\; U(q,x)+W(q,x) \;=\; V(q)+V'(q)(x-q)+\frac{1}{2}\,V''(q)(x-q)^2+W(q,x)\,.$$

The semiclassical splitting in [14] is

$$\exp\left(-\frac{i}{\varepsilon^2}\frac{h}{2}A\right)\exp\left(-\frac{i}{\varepsilon^2}hW(q(\frac{h}{2}))\right)\exp\left(-\frac{i}{\varepsilon^2}\frac{h}{2}A\right). \tag{3}$$

Splittings based on this decomposition approximate the solution of the Schrödinger equation with the time-independent right hand side (1) by a smart composition of the solutions of some equations involving different time-dependent operators on their right hand side: $A(q(t))$ and $W(q(t))$.

In the discretisation setting of the Hagedorn wavepackets, the propagation with A can be done fast (and most important, cheap) via an accurate splitting in its components T and U, since each part can be propagated exactly, see [13]. If we denote $\delta t = h/N$ the time step used for this (internal) splitting (with a sufficiently large number N of internal time steps as it will be indicated later), we have here an order $\mathcal{O}\left((\delta t)^r\right)$. Indeed, the propagation with A involves only solving ordinary differential equations not depending on ε for the parameters Π and S:

$$\begin{aligned}
\dot q(t) &= p(t)\\
\dot Q(t) &= P(t)\\
\dot S(t) &= \frac{1}{2}\,p(t)^2 \;-\; U(q(t))\\
\dot p(t) &= -\,U'(q(t))\\
\dot P(t) &= -\,U''(q(t))\,Q(t)\,.
\end{aligned} \tag{4}$$

We see the vector field $\Pi = (q,p,Q,P)^T$ as the sum of two vector fields $(q,0,Q,0)^T$ and $(0,p,0,P)^T$. In the following equations, one of the vector fields is used to advance q, Q and S while keeping the other constant; the other one is used to advance p, P and S.

While higher order methods would be possible, we used here a method of order $r = 8$ for $\exp\left(-\frac{i}{\varepsilon^2}\frac{h}{2}A\right)$; the propagated parameters are then of order $\mathcal{O}\left((\delta t)^8 \frac{h}{2}\right)$. For the sake of completeness, we give here the propagation step with T and with U, respectively:

$$\begin{aligned} q(\delta t) &= q(0) + \delta t\, p(0) & p(\delta t) &= p(0) - \delta t\, \nabla U(q(0)) \\ Q(\delta t) &= Q(0) + \delta t\, P(0) & P(\delta t) &= P(0) - \delta t\, \nabla^2 U(q(0))Q(0) \\ S(\delta t) &= S(0) + \delta t\, \frac{1}{2} p(0)^T p(0) & S(\delta t) &= S(0) - \delta t\, U(q(0)) \end{aligned}$$

for an internal time-step δt.

The propagation with the time-dependent remainder $W(q(t))$ involves only the propagation of the coefficients c, given the evolution of the parameters Π, and it is the most expensive part of the algorithm. The rest of this section recall the key ideas in the proof of the convergence of the semiclassical splitting; they gave the main hint for the new algorithms.

Two functions that play the role of intermediate approximations are used in the proof of the convergence rate in [14]. Let Ψ and $\tilde{u}$ denote the exact and the numerical solutions of (1), respectively. Define $u(t)$ the Hagedorn wavepacket with the exact parameters $\Pi(t)$ and $S(t)$ from (4) and with the exact coefficients c given by the solution to the linear system of ordinary differential equations

$$i\,\varepsilon^2\,\dot{c} \;=\; F\left[\Pi(t)\right]\,c\,, \tag{5}$$

where the $K \times K$ matrix $F[\Pi]$ has entries

$$F_{j,k}\left[\Pi(t)\right] \;=\; \int \overline{\varphi_j[\Pi(t)](x)}\; W(q(t),x)\; \varphi_k[\Pi(t)](x)\, dx\,.$$

We can write the coefficients $c(t)$ as

$$c(t) = \exp\left(-\frac{i}{\varepsilon^2}\int_0^t F\left[\Pi(s)\right] ds\right) c(0)\,.$$

If we keep the exact Π and S, but approximate the coefficients $c(t)$ e.g. via the BCH-formula or a Magnus scheme involving a quadrature formula that corresponds to what is used in the construction of the numerical solution u, we get a function u_1 which is used as an intermediate solution in the proof of the error estimates. Let us shortly review the steps of the proof in [14]:

The Theorem 2 in [14] gives

$$\|\Psi(t) - u(t)\| \le \mathcal{C}\;\varepsilon^N\,,$$

with suitable $N \ge 1$ depending on the smoothness of V.

Lemma 4 in [14] ensures that the error in wavepackets is of order $\mathcal{O}\left((\delta t)^8 h/\varepsilon^2\right)$ which leads to the same order in the difference between an intermediate solution and the numerical solution, as in Theorem 4 in [14]:

$$\| u_1(h) - \tilde{u}(h) \| \;\leq\; \mathcal{C}\,\frac{(\delta t)^8}{\varepsilon^2}\,h\,.$$

Hence the local error

$$\begin{aligned}\| u(h) - \tilde{u}(h) \| \;&\leq\; \| u_1(h) - \tilde{u}(h) \| \; + \| u(h) - u_1(h) \| \\ &\leq\; \mathcal{C}\,\frac{(\delta t)^8}{\varepsilon^2}\,h + \; \| u(h) - u_1(h) \|\end{aligned}$$

is dominated by the difference between the numerical solution u and the intermediate solution u_1, if we choose the internal time-step δt accordingly.

Lemma 3 in [14] shows that the remainder W manifests itself as a perturbation of order $\mathcal{O}\left(\varepsilon^3\right)$ on the solution described solely by the parameters Π and S. Here is the new idea of this article: instead of the semiclassical splitting, it is more efficient and natural to use a splitting method which is specialized for perturbed systems.

3 Higher order splitting methods

The first step in the process to choose an appropriate method (or to build a new one) for numerically solving a given problem is to analyze the algebraic structure of the problem as well as to identify the costly parts when considering different ways to solve it.

In our case, the problem to be solved can be written as a separable perturbed problem as follows

$$\dot{u} = (A(t) + \varepsilon B(t))u \tag{6}$$

where we denote $W = \varepsilon B$ with ε the small parameter in order to highlight that it is a perturbed system. According to the considerations in the previous section, the non-autonomous equation

$$\dot{u} = A(t)u \tag{7}$$

can be efficiently and cheaply solved to high accuracy, while the solution of

$$\dot{u} = \varepsilon B(t)u \tag{8}$$

involves the costly part. However, if the explicit dependence on time is frozen in $B(t)$, the numerical solution of (8) can be carried out directly.

It seems natural to solve the problem using splitting methods, yet to choose the most appropriate one is not obvious. There are hundreds of splitting methods in the literature at different orders and tailored to different classes of problems (see [4,18,20] and references therein) and the performance achieved strongly depends on the choice of the method and its implementation in the appropriate way.

One can split a system in many different ways and to split into the correct form jointly with the use of the appropriate method makes an enormous difference as we will see.

For convenience of the reader, we briefly review splitting methods and focus on some subclasses of methods in which our problem belongs and for which there are tailored methods with improved performance.

3.1 Splitting an autonomous problem into two parts

Assume that the differential equation can be written as follows

$$\dot{x} = f^{[1]}(x) + f^{[2]}(x) = Ax + Bx, \qquad x(0) = x_0 \in \mathbb{R}^d \tag{9}$$

where

$$A = \sum_{i=1}^{d} f_i^{[1]} \frac{\partial}{\partial x_i}, \qquad B = \sum_{i=1}^{d} f_i^{[2]} \frac{\partial}{\partial x_i}$$

and such that each sub-problem

$$\dot{x} = Ax, \qquad \dot{x} = Bx,$$

can be integrated exactly (or more generally, it is simpler to integrate than the original system).

An s-stage splitting method is given by the composition

$$\Psi(h) = \mathrm{e}^{b_s hB}\, \mathrm{e}^{a_s hA} \cdots \mathrm{e}^{b_1 hB}\, \mathrm{e}^{a_1 hA} = \mathrm{e}^{F(h)}$$

with

$$F(h) = hv_a A + hv_b B + h^2 v_{ab}[A,B] + h^3 v_{aab}[A,[A,B]] + h^3 v_{bab}[B,[A,B]] + \mathcal{O}(h^4) \tag{10}$$

and $v_a, v_b, \ldots$ are polynomials in a_i, b_i. Since the exact solution is $x(h) = \mathrm{e}^{h(A+B)} x_0$ we compare $F(h)$ with $h(A+B)$ and we have to impose $v_a = v_b = 1$ to have a consistent scheme. If in addition $v_{ab} = 0$, then the resulting method is order two, etc. so the method is of order r if

$$\mathrm{e}^{F(h)} = \mathrm{e}^{h(A+B)} + \mathcal{O}(h^{r+1}).$$

A set of efficient methods up to order ten can be found in [20, 4, 18] and references therein.

3.2 Perturbed systems

Consider now that the equation takes the form

$$\dot{x} = f(x) = f^{[1]}(x) + \varepsilon f^{[2]}(x), \tag{11}$$

with $|\varepsilon| \ll 1$, where a consistent splitting method takes now the form

$$\Psi(h) = \mathrm{e}^{b_s h\varepsilon B}\, \mathrm{e}^{a_s hA} \cdots \mathrm{e}^{b_1 h\varepsilon B}\, \mathrm{e}^{a_1 hA} = \mathrm{e}^{h(A+\varepsilon B+E(h,\varepsilon))}, \tag{12}$$

with

$$\begin{aligned} E(h,\varepsilon) = {} & h\,\varepsilon\, v_{ab}[A,B] + h^2\,\varepsilon\, v_{aab}[A,[A,B]] + h^2\,\varepsilon^2\, v_{bab}[B,[A,B]] \\ & + h^3\,\varepsilon\, v_{aaab}[A,[A,[A,B]]] + h^3\varepsilon^2 v_{baab}[B,[A,[A,B]]] + h^3\varepsilon^3\, v_{bbab}[B,[B,[A,B]]] + \mathcal{O}(h^4)\ . \end{aligned} \tag{13}$$

Since one typically deals with small values of ε, we have to examine the local error as $\varepsilon \to 0$. This can be done of course by analyzing the difference between $\Psi(h)$ and $\mathrm{e}^{h(A+\varepsilon B)}$ or directly $E(h,\varepsilon)$.

According to [20], the method is said to be of *generalized order* $(r_1, r_2, \dots, r_m)$ (where $r_1 \geq r_2 \geq \cdots \geq r_m$) if the remainder in (14) is such that

$$hE(h,\varepsilon) = \mathcal{O}(\varepsilon h^{r_1+1} + \varepsilon^2 h^{r_2+1} + \cdots + \varepsilon^m h^{r_m+1}).$$

As a result, one can use an scheme for perturbed problems that provides similar accuracy as a general splitting method, but at a reduced cost.

The following methods from [20] have shown to be highly efficient and were also considered and implemented for this work:

The 2-stage symmetric method of generalized order (4,2) given by the composition

$$\Psi_{(4,2)} = \mathrm{e}^{a_1 hA}\, \mathrm{e}^{b_1 h\varepsilon B}\, \mathrm{e}^{a_2 hA} \mathrm{e}^{b_1 h\varepsilon B}\, \mathrm{e}^{a_1 hA} \tag{14}$$

with $a_1 = (3-\sqrt{3})/6,\ a_2 = 1-2a_1,\ b_1 = 1/2$.

The 5-stage symmetric method of generalized order (8,4)

$$\Psi_{(8,4)} = \mathrm{e}^{a_1 hA}\, \mathrm{e}^{b_1 h\varepsilon B}\, \mathrm{e}^{a_2 hA}\, \mathrm{e}^{b_2 h\varepsilon B}\, \mathrm{e}^{a_3 hA}\, \mathrm{e}^{b_3 h\varepsilon B}\, \mathrm{e}^{a_3 hA}\, \mathrm{e}^{b_2 h\varepsilon B}\, \mathrm{e}^{a_2 hA}\, \mathrm{e}^{b_1 h\varepsilon B}\, \mathrm{e}^{a_1 hA} \tag{15}$$

with

$a_1 = 0.0753469602698929$	$a_2 = 0.5179168546882568$	$a_3 = 1/2 - (a_1 + a_2)$
$b_1 = 0.1902259393736766$	$b_2 = 0.8465240704435263$	$b_3 = 1 - 2(b_1 + b_2)$

3.3 Processed methods for perturbed systems

The number of order conditions for a given splitting scheme grows in general very rapidly with the order and this is true even for perturbed problems. Different strategies have been proposed along the years to reduce the number and complexity of the order conditions, thus ensuring that the resulting schemes require less evaluations than conventional methods.

One of such strategies is the use of a *processor* or *corrector*. The idea is: given an integrator ψ_h (the *kernel*), one tries to find a (near-identity) parametric map $\pi_h : \mathbb{R}^d \longrightarrow \mathbb{R}^d$ (the *pre-processor*) such that the new scheme

$$\hat{\psi}_h = \pi_h^{-1} \circ \psi_h \circ \pi_h \tag{16}$$

is more accurate than ψ_h. Application of n steps of the new integrator $\hat{\psi}_h$ leads to $\hat{\psi}_h^n = \pi_h^{-1} \circ \psi_h^n \circ \pi_h$.

The method ψ_h is of *effective order* r if a pre-processor π_h exists for which $\hat{\psi}_h$ is of (conventional) order r, that is, if $\pi_h^{-1} \circ \psi_h \circ \pi_h = v + \mathcal{O}(h^{r+1})$, where v is here the exact solution.

There are processed methods that are addressed to perturbed problems and we choose the following composition that provides a method of generalized order (7,6,4) with a 3-stage kernel given by[2]:

$$\begin{aligned}
\Psi^{[P]}_{(7,6,4)} &= \pi_6^{-1} \circ \psi_3 \circ \pi_6 \\
&= \left(\varphi^{[A]}_{-z_1 h} \circ \varphi^{[\varepsilon B]}_{-y_1 h} \circ \cdots \circ \varphi^{[A]}_{-z_6 h} \circ \varphi^{[\varepsilon B]}_{-y_6 h} \right) \\
&\quad \circ \left(\varphi^{[A]}_{a_1 h} \circ \varphi^{[\varepsilon B]}_{b_2 h} \circ \varphi^{[A]}_{a_2 h} \circ \varphi^{[\varepsilon B]}_{b_2 h} \circ \varphi^{[A]}_{a_2 h} \circ \varphi^{[\varepsilon B]}_{b_1 h} \circ \varphi^{[A]}_{a_1 h} \right) \\
&\quad \circ \left(\varphi^{[\varepsilon B]}_{y_6 h} \circ \varphi^{[A]}_{z_6 h} \circ \cdots \circ \varphi^{[\varepsilon B]}_{y_1 h} \circ \varphi^{[A]}_{z_1 h} \right),
\end{aligned} \tag{17}$$

with coefficients given in Table 1. It is possible to build more elaborated pre- and post-processors leading to methods of generalized order $(s, 6, 4)$ with s as large as desired.

3.4 Processed methods with modified potentials for perturbed systems

Let us now look closer at the semiclassical Schrödinger equation in the context of the above splittings:

$$\partial_t \psi = \left(\left(i\frac{\varepsilon^2}{2} \Delta_x - i\frac{1}{\varepsilon^2} U(t,x) \right) - i\frac{1}{\varepsilon^2} W(t,x) \right) \psi = (A+B)\psi$$

[2] We write the scheme as a composition of maps because the exponentials of Lie operators act in the reverse order as they are written. This is not a problem for symmetric compositions, but processed methods are not symmetric compositions and we want to avoid any confusion in the application of the methods.

Table 1 Coefficients for the (7,6,4) processed splitting method.

$a_1 = 0.5600879810924619$	$a_2 = 1/2 - a_1$	
$b_1 = 1.5171479707207228$	$b_2 = 1 - 2b_1$	
$z_1 = -0.3346222298730800$	$z_2 = 1.0975679907321640$	$z_3 = -1.0380887460967830$
$z_4 = 0.6234776317921379$	$z_5 = -1.1027532063031910$	$z_6 = -0.0141183222088869$
$y_1 = -1.6218101180868010$	$y_2 = 0.0061709468110142$	$y_3 = 0.8348493592472594$
$y_4 = -0.0511253369989315$	$y_5 = 0.5633782670698199$	$y_6 = -0.5$

where W is the remainder in the quadratic Taylor expansion of the potential V as in Section 2:

$$W(q,x) = V(x) - U(q,x) = V(x) - V(q) - V'(q)(x-q) - \frac{1}{2}V''(q)(x-q)^2 \,. \quad (18)$$

The commutator ($[A,B] = (AB - BA)$) acting on the wave function is given by

$$\begin{aligned}[A,B]\psi &= [i\frac{\varepsilon^2}{2}\Delta_x - i\frac{1}{\varepsilon^2}U(t,x), -i\frac{1}{\varepsilon^2}W(x,t)]\psi = [i\frac{\varepsilon^2}{2}\Delta_x, -i\frac{1}{\varepsilon^2}W(t,x)]\psi \\ &= \frac{1}{2}[\Delta_x, W]\psi = \frac{1}{2}\Delta_x W\psi + \nabla_x W \cdot \nabla_x \psi.\end{aligned}$$

We used here that U an W commute, so we obtained

$$[A,B] = \frac{1}{2}\Delta_x W + \nabla_x W \cdot \nabla_x ,$$

that we use further in

$$\begin{aligned}[B,[A,B]]\psi &= [-i\frac{1}{\varepsilon^2}W(t,x), \frac{1}{2}\Delta_x W + \nabla_x W \cdot \nabla_x]\psi = -i\frac{1}{\varepsilon^2}[W(t,x), \nabla_x W \cdot \nabla_x]\psi \\ &= i\frac{1}{\varepsilon^2}\nabla_x W \cdot \nabla_x W\psi.\end{aligned}$$

As a result we have that

$$[B,[A,B]] = -i\frac{1}{\varepsilon^2}\left(-\nabla_x W \cdot \nabla_x W\right),$$

i.e., it depends on coordinates and commutes with B. In addition, (18) gives

$$\nabla_x W = V'(x) - V'(q) - V''(q)(x-q)\,,$$

so $\nabla_x W \cdot \nabla_x W$ can be computed from the previous evaluations of W. Hence, the evaluation of $e^{B+[B,[A,B]]}$ has similar complexity and computational cost as the evaluation of e^B.

This property allows us to use a processed method in which the kernel is a small modification of the semiclassical splitting *at no extra cost* corresponding

Table 2 Coefficients for the (6,4) processed splitting method.

$y_1 = -0.1659120515409654$	$y_2 = -0.1237659000825160$	$y_3 = 0.0250397323738759$
$y_4 = 0.2269372219010943$		
$z_1 = -0.9125829692505096$	$z_2 = -0.3605243318856133$	$z_3 = 0.7354063037876117$
$z_4 = 0.5$		

to the following scheme of effective generalized order (6,4) from [6] whose coefficients are collected in Table 2

$$\begin{aligned}\Psi^{[P]}_{(6,4)} &= \pi_4^{-1} \circ \psi_{1,1} \circ \pi_4\\ &= \left(\varphi^{[B]}_{-y_1 h} \circ \varphi^{[A]}_{-z_1 h} \circ \varphi^{[B]}_{-y_2 h} \circ \varphi^{[A]}_{-z_2 h} \circ \varphi^{[B]}_{-y_3 h} \circ \varphi^{[A]}_{-z_3 h} \circ \varphi^{[B]}_{-y_4 h} \circ \varphi^{[A]}_{-z_4 h}\right)\\ &\quad \circ\left(\varphi^{[A]}_{h/2} \circ \widetilde{\varphi}^{[Bm]}_{h,h/24} \circ \varphi^{[A]}_{h/2}\right) \qquad (19)\\ &\quad \circ\left(\varphi^{[A]}_{z_4 h} \circ \varphi^{[B]}_{y_4 h} \circ \varphi^{[A]}_{z_3 h} \circ \varphi^{[B]}_{y_3 h} \circ \varphi^{[A]}_{z_2 h} \circ \varphi^{[B]}_{y_2 h} \circ \varphi^{[A]}_{z_1 h} \circ \varphi^{[B]}_{y_1 h}\right),\end{aligned}$$

with

$$\begin{aligned}\widetilde{\varphi}^{[Bm]}_{h,h/24} &\equiv e^{hB+\frac{h^3}{24}[B,[A,B]]}\\ &= \exp\left(-i\frac{1}{\varepsilon^2} h\left(W(q(\frac{h}{2})) - \frac{h^2}{24}\nabla_x W(q(\frac{h}{2}))\cdot\nabla_x W(q(\frac{h}{2}))\right)\right).\end{aligned}$$

3.5 Splitting methods for non-autonomous systems

Notice that the splitting methods we have considered until now are valid for the numerical integration of autonomous equations. However, our splitting introduces a time-dependence. The question we analyze next is whether the same schemes can be used when there is an explicit time dependence in the equation to integrate. The answer is positive if the schemes are used properly.

Let us consider a non-autonomous perturbed system given by

$$\dot{x} = f(t,x) = f^{[1]}(t,x) + \varepsilon f^{[2]}(t,x), \qquad x(0) = x_0. \qquad (20)$$

Then we can take t as a new coordinate and transform (20) into an equivalent autonomous equation to which splitting algorithms for perturbed problems are subsequently applied. More specifically, equation (20) is equivalent to the enlarged system

$$\frac{d}{dt}\begin{pmatrix} x\\ x_t\end{pmatrix} = \underbrace{\begin{pmatrix} f^{[1]}(x_t,x)\\ 1\end{pmatrix}}_{\hat{f}^{[1]}} + \varepsilon \underbrace{\begin{pmatrix} f^{[2]}(x_t,x)\\ 0\end{pmatrix}}_{\hat{f}^{[2]}} \qquad (21)$$

with $x_t \in \mathbb{R}$ leading to the following (autonomous) equations

$$\dot{y} = \hat{f}^{[1]}(y), \qquad \dot{y} = \varepsilon\hat{f}^{[2]}(y)$$

with $y = (x, x_t)$. Since the equation $\dot{x} = 1$ has been appended to the dominant part, the equation still retains the structure of a perturbed problem and then we can use the previous splitting methods for such class of problems [7].

This splitting requires the exact solution of the non-autonomous problem

$$\dot{x} = f^{[1]}(t, x), \tag{22}$$

or to numerically solve it to high accuracy. Since this is the cheap part of our problem, one can safely use a high-order method to solve it at a low computational cost.

On the other hand, since the equation to be solved in the extended phase space is autonomous, processed methods can be used and it only remains to check if the scheme with modified potential can still be used as well as to analyze how it must be adapted to the non-autonomous case.

To this purpose we write the equation to solve as follows

$$\begin{aligned}
\partial_t \begin{pmatrix} \psi \\ u_t \end{pmatrix} &= \begin{pmatrix} \left(\left(i\frac{\varepsilon^2}{2}\Delta_x - i\frac{1}{\varepsilon^2}U(u_t, x) \right) - i\frac{1}{\varepsilon^2}W(u_t, x) \right) \psi \\ 1 \end{pmatrix} \\
&= \begin{pmatrix} \left(i\frac{\varepsilon^2}{2}\Delta_x - i\frac{1}{\varepsilon^2}U(u_t, x) \right) \psi \\ 1 \end{pmatrix} + \begin{pmatrix} -i\frac{1}{\varepsilon^2}W(u_t, x)\psi \\ 0 \end{pmatrix} \\
&= (\widetilde{A} + \widetilde{B}) \begin{pmatrix} \psi \\ u_t \end{pmatrix}
\end{aligned}$$

where now $\widetilde{A}, \widetilde{B}$ are Lie operators defined as follows

$$\begin{aligned}
\widetilde{A} &\equiv \left(i\frac{\varepsilon^2}{2}\Delta_x - i\frac{1}{\varepsilon^2}U(u_t, x) \right) \frac{\delta}{\delta\psi} + 1 \cdot \frac{\delta}{\delta u_t}, \\
\widetilde{B} &\equiv -i\frac{1}{\varepsilon^2}W(u_t, x)\frac{\delta}{\delta\psi}\,.
\end{aligned}$$

We can easily check that

$$[\widetilde{B}, [\widetilde{A}, \widetilde{B}]] \begin{pmatrix} \psi \\ u_t \end{pmatrix} = \begin{pmatrix} i\frac{1}{\varepsilon^2}\nabla_x W \cdot \nabla_x W\psi \\ 0 \end{pmatrix}.$$

To sum up, we can use splitting methods for perturbed problems with processing and modified potentials. This requires to solve either exactly or numerically to high accuracy, the non-autonomous problem

$$\partial_t\psi = \left(i\frac{\varepsilon^2}{2}\Delta_x - i\frac{1}{\varepsilon^2}U(t, x) \right) \psi$$

as well as to solve the autonomous problem (the time is frozen at each stage)

$$\partial_t\psi = -i\frac{1}{\varepsilon^2}W\psi$$

or, for the kernel (that contains a modified potential)

$$\partial_t \psi = -i\frac{1}{\varepsilon^2}\left(W - \frac{h^2}{24}\nabla_x W \cdot \nabla_x W\right)\psi.$$

As a result, we have a numerical scheme that has similar stability and computational cost as the semiclassical splitting, but provides a considerably improved accuracy.

4 Numerical results

Let us summarise first the main results of the previous section. We saw that the splitting methods for perturbed systems $A + W = A + \varepsilon B$ are the natural choice to increase the convergence order. They all look like

$$\prod_j \exp\left(-\frac{i}{\varepsilon^2} b_j h W(q(b_j h))\right) \exp\left(-\frac{i}{\varepsilon^2} a_j h A\right) \tag{23}$$

with constants a_j and b_j properly chosen. Since the expensive part is that involving W, only AWA-schemes in the sense of [20] come into play. Moreover, since the splitting is time-dependent, we have to deal in each step with equations that depend explicitly on time; we make them autonomous by adding an unknown for the time. A crucial remark is that the propagation of the time-variable must be done together with the propagation with A, in order not to loose the advantage that W acts as a perturbation of order $\mathcal{O}(\varepsilon)$. The Theorem 3 in [14] brings up the result on the error from the order of the involved quadrature rule.

The global convergence rates and the costs in terms of the most expensive propagation (that involving W) are summarised in the Table 3. The most efficient method is the preprocessor splitting method (19) with the modified potential, while a compromise between work and accuracy gives the 3-stages $(7, 6, 4)$-method (17).

The exact implementation details are in the python library WaveBlocks [8] that is publicly available. We give below the complete description of the propagation algorithm based on the $(7, 6, 4)$-scheme. In IntSplit we used the propagation with A via the order 8 splitting of Kahan and Li, e.g. [18] page 157, and the exact propagation with U and T as in (5).

We may choose δt such that we let the error term $(\delta t)^8 h/\varepsilon^2$ to match any of the above orders. We took the number of the internal steps of length $\delta t = h/N$ to be $N = 1 + [(h^{8-\beta}/\varepsilon^{2+\alpha})^{1/8}]$ fitting the overall error $\varepsilon^\alpha h^\beta$.

The benchmark problem is the same as that used in [14]: the initial value

$$\psi(0) = \varphi[1, 0, 1, i] \tag{24}$$

is propagated from $t = 0$ to the time $\mathcal{T} = 2$ using 24 wavepackets into the torsional potential $V(x) = 1 - \cos(x)$. All error plots use the same legends which are displayed in Fig. 1. The error plots for various values of the parameter ε for the modified $(6, 4)$ perturbed method (19) and the semiclassical $(7, 6, 4)$

Table 3 Comparison of the high order splitting methods.

Method	Type	Cost	Order
(3)	semiclassical	1	εh^2
(14)	perturbed	2	$\varepsilon h^4 + \varepsilon^2 h^2$
(15)	perturbed	5	$\varepsilon h^8 + \varepsilon^2 h^4$
(19)	perturbed, processor, modification	1	$\varepsilon h^6 + \varepsilon^2 h^4$
(17)	perturbed, processor	3	$\varepsilon h^7 + \varepsilon^2 h^6 + \varepsilon^3 h^4$

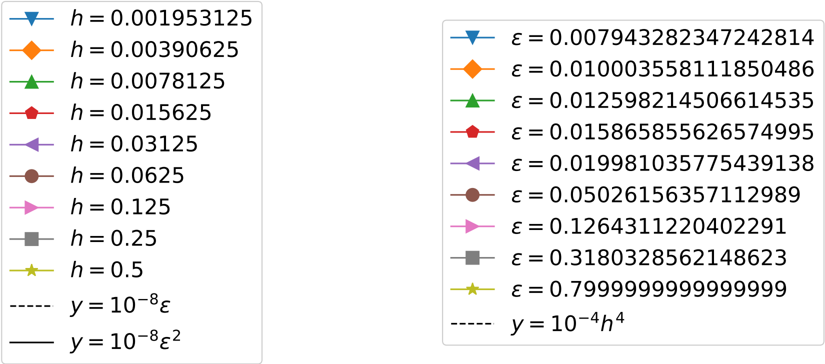
$h = 0.001953125$
$h = 0.00390625$
$h = 0.0078125$
$h = 0.015625$
$h = 0.03125$
$h = 0.0625$
$h = 0.125$
$h = 0.25$
$h = 0.5$
$y = 10^{-8}\varepsilon$
$y = 10^{-8}\varepsilon^2$
$\varepsilon = 0.007943282347242814$
$\varepsilon = 0.010003558111850486$
$\varepsilon = 0.012598214506614535$
$\varepsilon = 0.015865855626574995$
$\varepsilon = 0.019981035775439138$
$\varepsilon = 0.05026156357112989$
$\varepsilon = 0.1264311220402291$
$\varepsilon = 0.3180328562148623$
$\varepsilon = 0.7999999999999999$
$y = 10^{-4}h^4$

Fig. 1 The legends for all the experiments.

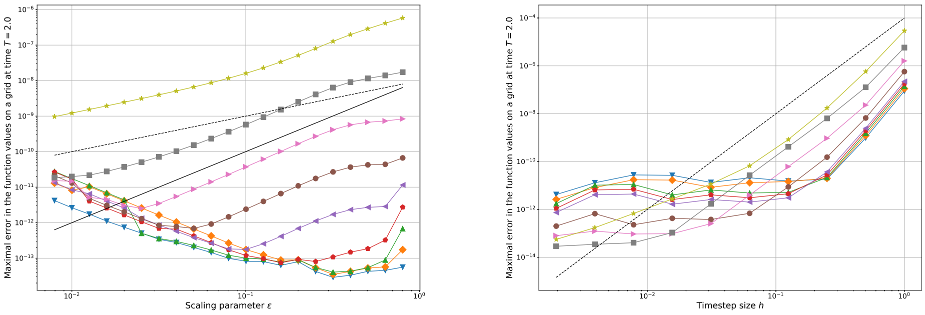

Fig. 2 The error dependence on ε and on h: semiclassical modified $(6, 4)$ processed method, torsional potential.

method are in Figs. 2, 3 and 4 and should be compared with the left part of Figures 2 and 3 in [14].

In Figs. 2 and 3, the results for various parameters are compared to reference computations with the same algorithm with the timestep $h = 10^{-12}$.

In Fig. 4 we used the semiclassical $(7, 6, 4)$-method with the timestep $h = 10^{-12}$ as the reference solution.

In Fig. 5 we display the dependence on ε of the L^2-norm of the error estimated on a space grid of both the modified $(6, 4)$ perturbed method and the semiclassical $(7, 6, 4)$-method when the reference solution is based on a

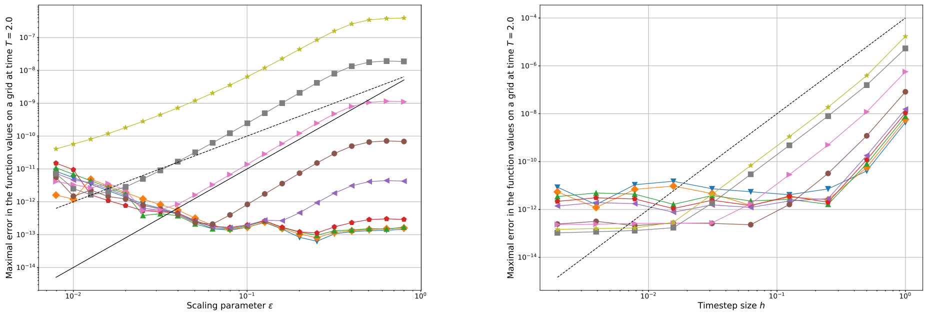

Fig. 3 The error dependence on ε and on h: semiclassical $(7, 6, 4)$-method, torsional potential.

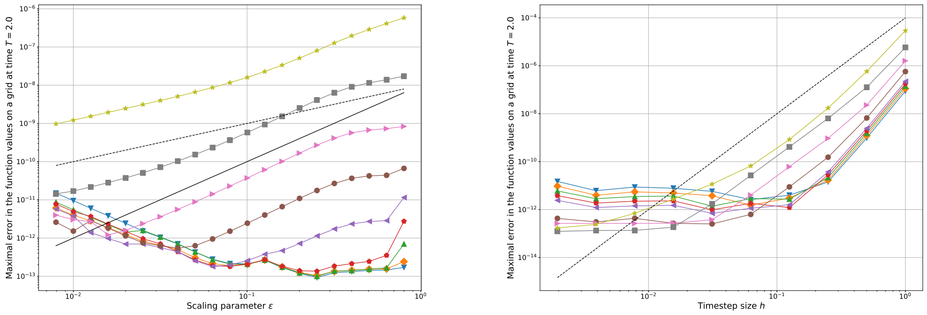

Fig. 4 The error dependence on ε and on h: semiclassical modified $(6, 4)$ processed against a reference solution by the semiclassical $(7, 6, 4)$-method, torsional potential.

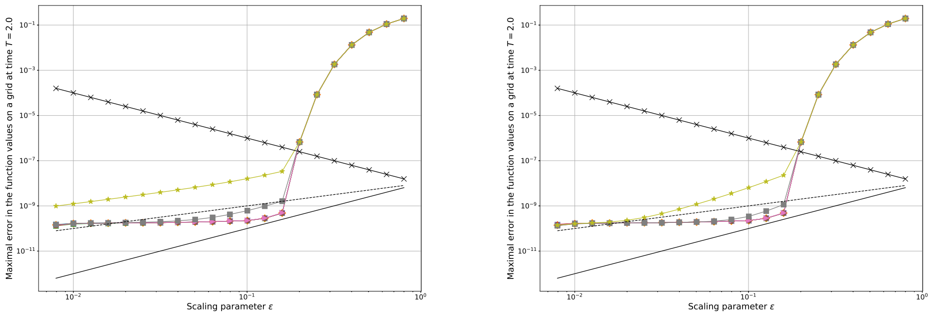

Fig. 5 The error dependence on ε of the L^2-norm of the error estimated on a space grid; the reference solution is given by an order 6 scheme in time with tiny h together with a very precise Fourier approximation in space: the modified $(6, 4)$ perturbed method (left) and the semiclassical $(7, 6, 4)$-method (right). The line with crosses reminds us the $1/\varepsilon^2$-deterioration of the standard schemes.

Fourier approximation with 2^{16} points in space and a symplectic Runge-Kutta-Nystrom method of order 6 in time with step $h = 10^{-16}$. We see that the solution is not accurate for large ε, which corresponds to models that are not semiclassical. The line with crosses reminds us the $1/\varepsilon^2$-deterioration of the standard schemes. We see the improvement of the quality of our solution for small ε; large time-steps can be used without loss in accuracy.

The methods are so precise that the strong round off in $\exp(-i\eta/\varepsilon^2) - \exp(-i(\eta + \mathrm{eps})/\varepsilon^2)$, with $\eta \neq 0$ and eps = machine precision, in the measurement of the error is not avoidable. At ε close to 1, the semiclassical approximation is not valid anymore. At ε small, the solution based on Fourier approximation and splitting is not accurate enough, so we used as reference solution that produced by our own method with a small time-step and large basis sets, which were observed to be enough for the propagation till endtime $\mathcal{T}$.

If the solution does not remain localized, more basis functions are needed, as we can see in Fig. 6; here we look at the convergence with respect to the time step and the choice of the basis size K for the physical parameters in case of the interaction potential in ${}^1\Sigma_g^+\mathrm{Hg}_2$ as found in [21]; the parameter set for the potential and the initial value $\psi(0) = \varphi[q_0, p_0, Q_0, P_0]$ as well as ε are given inside the top of the Fig. 6.

Algorithm 1 A full simulation from $t = 0$ to end time $t = Mh = \mathcal{T}$.

```
procedure SIMULATION($u[\Pi, \mathbf{c}]$, $h$, $M$)
    $N = \max\left(1, 1 + \left\lfloor \sqrt{h\,\varepsilon^{-\frac{3}{4}}} \right\rfloor\right)$    ▷ Number of local steps to be used with $A = T + U$
    $u[\Pi, \mathbf{c}]$ = PREPROPAGATE($u[\Pi, \mathbf{c}], h, N$)
    for $i = 1, \ldots, M$ do
        $u[\Pi, \mathbf{c}]$ = PROPAGATE($u[\Pi, \mathbf{c}], h, N$)
    end for
    $u[\Pi, \mathbf{c}]$ = POSTPROPAGATE($u[\Pi, \mathbf{c}], h, N$)
    return $u[\Pi, \mathbf{c}]$
end procedure
```

Algorithm 2 A single timestep of the kernel.

```
procedure PROPAGATE($u[\Pi, \mathbf{c}]$, $h$, $N$)
    for $j = 1, \ldots, k$ do                    ▷ $(7, 6, 4)$ has $k = 4$, while (19) has $k = 1$
        $h_2 = a_j h$                            ▷ Step with (in case of (19) modified) remainder $W$
        $\mathbf{F} \in \mathbb{C}^{K \times K}$; $\mathbf{F}_{r,c} :=< \varphi_r|W|\varphi_c >$
        $\mathbf{c} = \exp\left(-\imath \frac{h_2}{\varepsilon^2} \mathbf{F}\right) \mathbf{c}$
        $h_1 = b_j h$                                          ▷ Step with $A = T + U$
        $u[\Pi, \mathbf{c}]$ = INTSPLIT($u[\Pi, \mathbf{c}], h_1, N$)
    end for
    return $u[\Pi, \mathbf{c}]$
end procedure
```

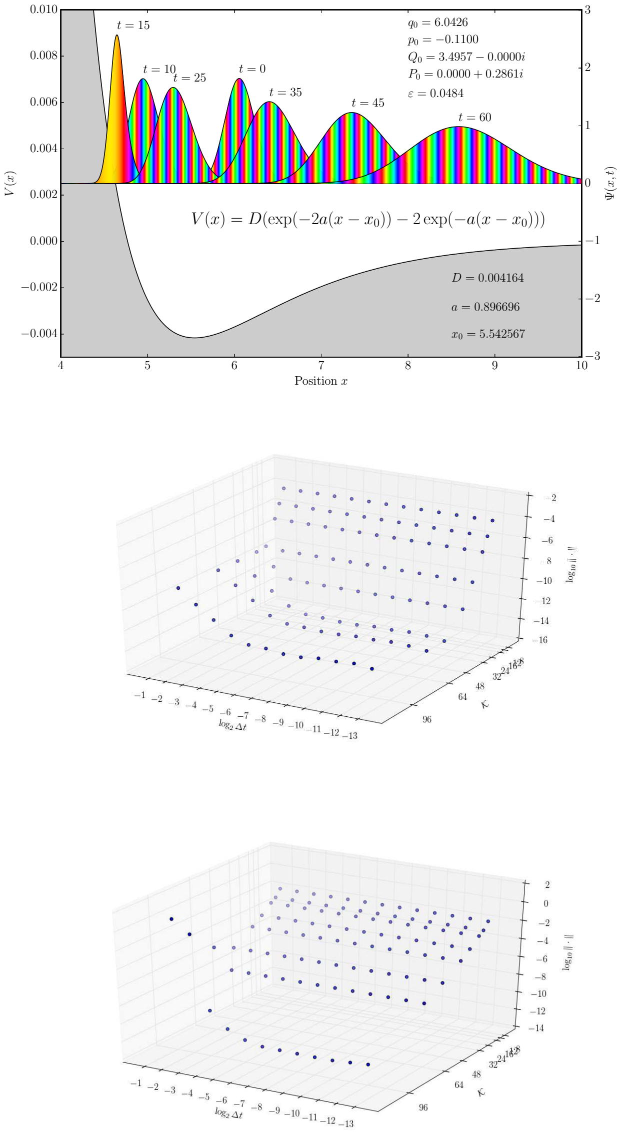

Fig. 6 Computational setting for ${}^1\Sigma_g^+\mathrm{Hg}_2$ (top), the error dependence on the time step h and on the size of the basis K in the semiclassical $(7, 6, 4)$-method at end time $T = 5$ (middle) and at $T = 35$ (bottom).

Algorithm 4 Postprocessing.

```
procedure PostPropagate($u[\Pi, \mathbf{c}]$, $h$, $N$)
    for $j = v, \ldots, 1$ do                                ▷ $(7, 6, 4)$ has $v = 6$, while $(19)$ has $v = 4$
        $h_2 = y_j h$                                            ▷ Step with remainder $W$
        $\mathbf{F} \in \mathbb{C}^{|\mathcal{K}| \times |\mathcal{K}|}$; $\mathbf{F}_{r,c} := < \varphi_r | W | \varphi_c >$
        $\mathbf{c} = \exp\left(-\imath \frac{h_2}{\varepsilon^2} \mathbf{F}\right) \mathbf{c}$
        $h_1 = z_j h$                                            ▷ Step with $A = T + U$
        $u[\Pi, \mathbf{c}] = $ IntSplit$(u[\Pi, \mathbf{c}], h_1, N)$
    end for
    return $u[\Pi, \mathbf{c}]$
end procedure
```

Algorithm 3 Preprocessing.

```
procedure PrePropagate($u[\Pi, \mathbf{c}]$, $h$, $N$)
    for $j = 1, \ldots, v$ do                                ▷ $(7, 6, 4)$ has $v = 6$, while $(19)$ has $v = 4$
        $h_1 = -z_j h$                                           ▷ Step with $A = T + W$
        $u[\Pi, \mathbf{c}] = $ IntSplit$(u[\Pi, \mathbf{c}], h_1, N)$
        $h_2 = -y_j h$                                           ▷ Step with remainder $W$
        $\mathbf{F} \in \mathbb{C}^{|\mathcal{K}| \times |\mathcal{K}|}$; $\mathbf{F}_{r,c} := < \varphi_r | W | \varphi_c >$
        $\mathbf{c} = \exp\left(-\imath \frac{h_2}{\varepsilon^2} \mathbf{F}\right) \mathbf{c}$
    end for
    return $u[\Pi, \mathbf{c}]$
end procedure
```

References

1. Bader, P., Iserles, A., Kropielnicka, K., Singh, P.: Effective approximation for the linear time-dependent Schrödinger equation. Foundations of Computational Mathematics **14**(4), 689–720 (2014)
2. Balakrishnan, N., Kalyanaraman, C., Sathyamurthy, N.: Time-dependent quantum mechanical approach to reactive scattering and related processes. Physics Reports **280**(2), 79–144 (1997). doi:10.1016/S0370-1573(96)00025-7
3. Bao, W., Jin, S., Markowich, P.A.: On time-splitting spectral approximations for the Schrödinger equation in the semiclassical regime. J. Comput. Phys. **175**(2), 487–524 (2002). doi:10.1006/jcph.2001.6956
4. Blanes, S., Casas, F.: A Concise Introduction to Geometric Numerical Integration. CRC Press, Boca Raton (2016)
5. Blanes, S., Casas, F., Ros, J.: Symplectic integration with processing: A general study. SIAM Journal on Scientific Computing **21**(2), 711–727 (1999). doi:10.1137/S1064827598332497
6. Blanes, S., Casas, F., Ros, J.: Processing symplectic methods for near-integrable hamiltonian systems. Celest. Mech. & Dyn. Astron. (77), 17–36 (2000)
7. Blanes, S., Diele, F., Maragni, C., Ragni, S.: Splitting and composition methods for explicit time dependence in separable dynamical systems. J. Comput. Appl. Math. pp. 646–659 (2010)
8. Bourquin, R., Gradinaru, V.: WaveBlocksND: Reusable building blocks for simulations with semiclassical wavepackets (2010-15). URL `https://github.com/raoulbq/WaveBlocksND`
9. Bourquin, R., Gradinaru, V., Hagedorn, G.: Non-adiabatic transitions near avoided crossings: theory and numerics. Journal of Mathematical Chemistry pp. 1–18 (2011). doi:10.1007/s10910-011-9827-2

10. Church, M., Hele., T., Ezra, G., Ananth, N.: Nonadiabatic semiclassical dynamics in the mixed quantum-classical initial value representation. The Journal of Chemical Physics **148**(10), 102,326 (2018). doi:10.1063/1.5005557
11. Coronado, E., Batista, V., Miller, W.: Nonadiabatic photodissociation dynamics of icn in the Ã continuum: A semiclassical initial value representation study. The Journal of Chemical Physics **112**(13), 5566–5575 (2000). doi:10.1063/1.481130
12. Descombes, S., Thalhammer, M.: An exact local error representation of exponential operator splitting methods for evolutionary problems and applications to linear Schrödinger equations in the semi-classical regime. BIT **50**(4), 729–749 (2010). doi:10.1007/s10543-010-0282-4
13. Faou, E., Gradinaru, V., Lubich, C.: Computing semiclassical quantum dynamics with Hagedorn wavepackets. SIAM J. Sci. Comput. **31**(4), 3027–3041 (2009). doi:10.1137/080729724
14. Gradinaru, V., Hagedorn, G.A.: Convergence of a semiclassical wavepacket based time-splitting for the Schrödinger equation. Numerische Mathematik **126**(1), 53–73 (2014). doi:10.1007/s00211-013-0560-6
15. Gradinaru, V., Hagedorn, G.A., Joye, A.: Exponentially accurate semiclassical tunneling wavefunctions in one dimension. Journal of Physics A **43**(47), 474,026, 13 (2010). doi:10.1088/1751-8113/43/47/474026
16. Gradinaru, V., Hagedorn, G.A., Joye, A.: Tunneling dynamics and spawning with adaptive semi-classical wave-packets. Journal of Chemical Physics **132** (2010). doi:10.1063/1.3429607
17. Hagedorn, G.A.: Raising and lowering operators for semiclassical wave packets. Ann. Phys. **269**(1), 77–104 (1998). doi:10.1006/aphy.1998.5843
18. Hairer, E., Lubich, C., Wanner, G.: Geometric numerical integration. Structure-preserving algorithms for ordinary differential equations, vol. 31, second edn. Springer Series in Computational Mathematics (2006)
19. Keller, J., Lasser, C.: Propagation of quantum expectations with Husimi functions. SIAM Journal of Applied Mathematics **73**, 1557–1581 (2012)
20. McLachlan, R.: Composition methods in the presence of small parameters. BIT Numerical Mathematics **35**(2), 258–268 (1995). doi:10.1007/BF01737165
21. Stefanov, B., Iordanov, O., Zarkova, L.: Interaction potential in ${}^1\Sigma_g{}^+Hg_2$: fit to the experimental data. Journal of Physics B: Atomic and Molecular Physics **15**(2), 239 (1982)
22. Wang, C.S., Curtis, J.C., Lester, B.J., Zhang, Y., Gao, Y.Y., Freeze, J., Batista, V.S., Vaccaro, P.H., Chuang, I.L., Frunzio, L., Jiang, L., Girvin, S.M., Schoelkopf, R.J.: Quantum simulation of molecular vibronic spectra on a superconducting bosonic processor (2019)