A splitting approach for the magnetic Schrödinger equation

M. Caliari*

Dipartimento di Informatica, Università di Verona, Italy

A. Ostermann, C. Piazzola

Institut für Mathematik, Universität Innsbruck, Austria

Abstract

The Schrödinger equation in the presence of an external electromagnetic field is an important problem in computational quantum mechanics. It also provides a nice example of a differential equation whose flow can be split with benefit into three parts. After presenting a splitting approach for three operators with two of them being unbounded, we exemplarily prove first-order convergence of Lie splitting in this framework. The result is then applied to the magnetic Schrödinger equation, which is split into its potential, kinetic and advective parts. The latter requires special treatment in order not to lose the conservation properties of the scheme. We discuss several options. Numerical examples in one, two and three space dimensions show that the method of characteristics coupled with a nonequispaced fast Fourier transform (NFFT) provides a fast and reliable technique for achieving mass conservation at the discrete level.

Keywords: magnetic Schrödinger equation, exponential splitting methods, convergence, Fourier techniques, nonequispaced fast Fourier transform

1. Introduction

In quantum mechanics a lot of phenomena occur under the influence of an external electromagnetic field. Typical examples include the Zeeman effect, Landau levels and superconductivity. So, quite a few problems in computational solid state physics and quantum chemistry require the solution of the

*Corresponding author

Email addresses: marco.caliari@univr.it (M. Caliari), alexander.ostermann@uibk.ac.at (A. Ostermann), chiara.piazzola@uibk.ac.at (C. Piazzola)

Schrödinger equation in the presence of an electromagnetic field

$$\begin{aligned} \mathrm{i}\varepsilon\partial_t u &= \frac{1}{2}(\mathrm{i}\varepsilon\nabla + A)^2 u + Vu, \quad t \geq 0,\ x \in \mathbb{R}^d, \\ u(0,x) &= u_0(x). \end{aligned} \tag{1}$$

Here, the unknown $u = u(t,x) \in \mathbb{C}$ is the quantum mechanical wave function, $V(t,x) \in \mathbb{R}$ is the scalar potential and $A(t,x) = (A_1(t,x), \dots, A_d(t,x))^\mathsf{T} \in \mathbb{R}^d$ is the vector potential. In addition $\varepsilon \in (0,1]$ denotes the small semi-classical parameter which is the scaled Planck constant. The equation is considered subject to vanishing boundary conditions, i.e., $\lim_{|x|\to\infty} u(t,x) = 0$. We recall that mass is a conserved quantity of this equation.

Exponential splitting schemes constitute a well-established class of methods for the numerical solution of Schrödinger equations (see, e.g., [1, 4, 7, 8, 11]). In this approach, the kinetic part is solved in Fourier space, which gives spectral accuracy in space, whereas the multiplicative potential is integrated pointwise in physical space. The transformation between Fourier and physical space is carried out using the fast Fourier transform, which results in an overall fast algorithm. In our situation, however, when the vector potential depends on the position, we get an additional advection term, which cannot be handled efficiently with Fourier techniques.

Thus, the structure of problem (1) suggests to split the equation into three subproblems: a potential step which collects the scalar terms of the potentials (which are pointwise multiplications), a kinetic step which involves the Laplacian, and an advection step which results from the vector potential. For carrying out a time step, each of these steps is solved separately and their solutions are recombined to define the numerical approximation. This is the underlying idea of exponential splitting schemes (see [14, 5, 12]). In this paper we analyse a first-order method, the so-called Lie splitting. Note, however, that higher-order methods can be analysed in exactly the same way, if the underlying problem has enough spatial smoothness, see [6, 7].

Splitting the magnetic Schrödinger equation for the purpose of its numerical solution into three subproblems is not a new idea. In their recent paper [9], Jin and Zhou proposed such a scheme. For the solution of the advection step, they considered a semi-Lagrangian approach. Such an approach has been used in many other fields as well (see, e.g., [13, 2, 3]).

Our present paper differs from [9] mainly in the following aspects: we give a framework for carrying out an abstract convergence proof for exponential splitting methods applied to (1), and we give a detailed error analysis for the Lie splitting scheme by identifying the required smoothness assumptions on the data. Moreover, we address conservation properties of the scheme and identify an alternative to Lagrange interpolation, as the latter does not conserve mass.

The outline of this paper is as follows. We start in section 2 with an abstract convergence result for splitting into three subproblems. Guided by the properties of the magnetic Schrödinger equation, we present an analytic framework that allows us to prove convergence for exponential splitting schemes. We exemplify this by proving that Lie splitting applied to (1) has order of convergence one, as expected.

In section 3 we apply a gauge transformation to the magnetic Schrödinger equation to obtain the equivalent formulation (17) with a divergence-free vector potential. This formulation is used in (18) to define the employed splitting. In the following section we show how to compute the solution of the kinetic step in spectral space and that of the potential step in physical space. For the advection step we use the method of characteristics. However, since the characteristic curves do not cut the previous time horizon at grid points, in general, special care has to be taken. We compare three different possibilities, namely discrete Fourier series evaluation, local polynomial interpolation and Fourier series evaluation by a nonequispaced fast Fourier transform (NFFT), see [10]. The latter allows us to evaluate a Fourier series at an arbitrary set of points in a fast way. To our knowledge, this transform was not yet applied in the present context.

In section 7 we present some numerical results. Our main goal is the comparison of the different approximations used in the advection step. In particular, we study how well the considered numerical algorithms preserve mass, and how they compare in terms of computational efficiency.

2. Splitting into three operators

For the numerical solution of (1), we propose a splitting approach. Motivated by the particular form of the vector field, which is the sum of a kinetic, a potential and an advective part, we consider a splitting into three terms. For this purpose, we formulate (1) as an abstract initial value problem

$$\begin{aligned} \partial_t u &= (\mathcal{A}+\mathcal{B}+\mathcal{C})u, \quad 0 \le t \le T, \\ u(0) &= u_0 \end{aligned} \tag{2}$$

in a Banach space X with norm $\|\cdot\|$. Below, we will state an analytic framework for these operators $\mathcal{A}$, $\mathcal{B}$ and $\mathcal{C}$ that, on the one hand, is sufficiently general to include the magnetic Schrödinger equation as an example and, on the other hand, allows us to carry out an abstract convergence proof for (exponential) splitting methods. We will illustrate our approach by analysing in detail the Lie splitting scheme[1]

$$u_{n+1} = \mathrm{e}^{\tau\mathcal{C}}\mathrm{e}^{\tau\mathcal{A}}\mathrm{e}^{\tau\mathcal{B}}u_n, \tag{3}$$

where τ denotes the step size and u_n is the numerical approximation to the true solution $u(t) = \mathrm{e}^{t(\mathcal{A}+\mathcal{B}+\mathcal{C})}u(0)$ at time $t = t_n = n\tau$. We will show below that the Lie splitting scheme is first-order convergent. Let us stress, however, that exactly the same ideas can be used to analyse exponential splitting methods of higher order.

[1] Throughout the paper $\mathrm{e}^{\tau\mathcal{L}}u_0$ will denote the exact solution at time τ of the abstract (linear) differential equation $\partial_t u = \mathcal{L}u$ with initial value $u(0) = u_0$.

In a first step, we will study the local error $\|\mathrm{e}^{\tau\mathcal{C}}\mathrm{e}^{\tau\mathcal{A}}\mathrm{e}^{\tau\mathcal{B}}u(t)-u(t+\tau)\|$ of Lie splitting along the exact solution. For this purpose, we employ the following assumption.

Assumption 1. Let $\mathcal{B}$ be a bounded operator, and let $\mathcal{A}$, $\mathcal{C}$, and $\mathcal{A}+\mathcal{C}$ generate strongly continuous semigroups $\mathrm{e}^{t\mathcal{A}}$, $\mathrm{e}^{t\mathcal{C}}$, and $\mathrm{e}^{t(\mathcal{A}+\mathcal{C})}$ on X. We assume that the following bounds hold for $0\le t\le T$ along the exact solution

$$\|[\mathcal{A},\mathcal{C}]\mathrm{e}^{s\mathcal{A}}u(t)\|\le c_1, \tag{4a}$$

$$\|\mathcal{C}\mathrm{e}^{s\mathcal{A}}\mathcal{B}u(t)\|\le c_2, \tag{4b}$$

$$\|\mathcal{C}^2\mathrm{e}^{s\mathcal{A}}u(t)\|\le c_3, \tag{4c}$$

$$\|\mathcal{C}\mathrm{e}^{\sigma\mathcal{A}}\mathcal{C}\mathrm{e}^{s(\mathcal{A}+\mathcal{C})}u(t)\|\le c_4, \tag{4d}$$

$$\|[\mathcal{A}+\mathcal{C},\mathcal{B}]\mathrm{e}^{s(\mathcal{A}+\mathcal{C})}u(t)\|\le c_5 \tag{4e}$$

with some constants c_1, c_2, c_3, c_4, and c_5 that do not depend on $0\le\sigma,s\le T$.

Next, we recall the definition of the φ_k functions, which play some role in our analysis. For complex z and integer $k\ge 1$, we set

$$\varphi_k(z)=\int_0^1 \mathrm{e}^{(1-\theta)z}\frac{\theta^{k-1}}{(k-1)!}\,\mathrm{d}\theta. \tag{5}$$

These functions are uniformly bounded in the complex half-plane $\operatorname{Re} z\le 0$ and analytic in $\mathbb{C}$. Let $\mathcal{E}$ be the generator of a strongly continuous semigroup. Then, for all $k\ge 1$, the following identity holds in the domain of $\mathcal{E}^k$

$$\mathrm{e}^{\tau\mathcal{E}}=\sum_{j=0}^{k-1}\frac{\tau^j}{j!}\mathcal{E}^j+\tau^k\mathcal{E}^k\varphi_k(\tau\mathcal{E}). \tag{6}$$

We are now in the position to state the local error bound.

Theorem 1 (Local error bound). *Under Assumption 1, the following bound for the local error holds*

$$\|\mathrm{e}^{\tau\mathcal{C}}\mathrm{e}^{\tau\mathcal{A}}\mathrm{e}^{\tau\mathcal{B}}u(t)-u(t+\tau)\|\le C\tau^2, \qquad t\in[0,T-\tau] \tag{7}$$

with a constant C that does not depend on t and τ.

Proof. Our proof uses ideas developed in [7]. Since $\mathcal{B}$ is bounded, the numerical solution can be expanded in the following way

$$\begin{aligned}\mathrm{e}^{\tau\mathcal{C}}\mathrm{e}^{\tau\mathcal{A}}\mathrm{e}^{\tau\mathcal{B}}u(t)&=\mathrm{e}^{\tau\mathcal{C}}\mathrm{e}^{\tau\mathcal{A}}\left(I+\tau\mathcal{B}+\mathcal{O}(\tau^2)\right)u(t)\\&=\underbrace{\mathrm{e}^{\tau\mathcal{C}}\mathrm{e}^{\tau\mathcal{A}}u(t)}_{P_1}+\underbrace{\tau\mathrm{e}^{\tau\mathcal{C}}\mathrm{e}^{\tau\mathcal{A}}\mathcal{B}u(t)}_{Q_1}+\mathcal{O}(\tau^2).\end{aligned} \tag{8}$$

The exact solution, on the other hand, is expanded with the help of the variation-of-constants formula. Applying this formula twice yields the representation

$$\begin{aligned}\mathrm{e}^{\tau(\mathcal{A}+\mathcal{B}+\mathcal{C})}u(t) &= \mathrm{e}^{\tau(\mathcal{A}+\mathcal{C})}u(t) + \int_0^\tau \mathrm{e}^{s(\mathcal{A}+\mathcal{C})}\mathcal{B}\mathrm{e}^{(\tau-s)(\mathcal{A}+\mathcal{B}+\mathcal{C})}u(t)\mathrm{d}s \\ &= \underbrace{\mathrm{e}^{\tau(\mathcal{A}+\mathcal{C})}u(t)}_{P_2} + \underbrace{\int_0^\tau \mathrm{e}^{s(\mathcal{A}+\mathcal{C})}\mathcal{B}\mathrm{e}^{(\tau-s)(\mathcal{A}+\mathcal{C})}u(t)\mathrm{d}s}_{Q_2} + \mathcal{O}(\tau^2).\end{aligned} \tag{9}$$

Collecting all the terms, we can rewrite the local error as

$$\mathrm{e}^{\tau\mathcal{C}}\mathrm{e}^{\tau\mathcal{A}}\mathrm{e}^{\tau\mathcal{B}}u(t) - u(t+\tau) = P + Q + \mathcal{O}(\tau^2), \tag{10}$$

where $P = P_1 - P_2$ and $Q = Q_1 - Q_2$.

For expanding P_1 we employ the φ_2 function (see (6)) to get

$$\mathrm{e}^{\tau\mathcal{C}}\mathrm{e}^{\tau\mathcal{A}}u(t) = \mathrm{e}^{\tau\mathcal{A}}u(t) + \tau\mathcal{C}\mathrm{e}^{\tau\mathcal{A}}u(t) + \tau^2\mathcal{C}^2\varphi_2(\tau\mathcal{C})\mathrm{e}^{\tau\mathcal{A}}u(t).$$

Using the variation-of-constants formula twice, we can rewrite P_2 as

$$\begin{aligned}\mathrm{e}^{\tau(\mathcal{A}+\mathcal{C})}u(t) = \mathrm{e}^{\tau\mathcal{A}}u(t) &+ \int_0^\tau \mathrm{e}^{s\mathcal{A}}\mathcal{C}\mathrm{e}^{(\tau-s)\mathcal{A}}u(t)\,\mathrm{d}s \\ &+ \int_0^\tau \mathrm{e}^{s\mathcal{A}}\mathcal{C}\int_0^{\tau-s} \mathrm{e}^{\sigma\mathcal{A}}\mathcal{C}\mathrm{e}^{(\tau-s-\sigma)(\mathcal{A}+\mathcal{C})}u(t)\,\mathrm{d}\sigma\mathrm{d}s.\end{aligned}$$

Thus, to bound P, we need first to estimate

$$\tau\mathcal{C}\mathrm{e}^{\tau\mathcal{A}}u(t) - \int_0^\tau \mathrm{e}^{s\mathcal{A}}\mathcal{C}\mathrm{e}^{(\tau-s)\mathcal{A}}u(t)\mathrm{d}s, \tag{11}$$

and then to bound the remaining terms. Let $f(s) = \mathrm{e}^{s\mathcal{A}}\mathcal{C}\mathrm{e}^{(\tau-s)\mathcal{A}}u(t)$. Then, the expression (11) becomes

$$\tau f(0) - \int_0^\tau f(s)\mathrm{d}s = \tau f(0) - \int_0^\tau \left(f(0) + \int_0^s f'(\sigma)\mathrm{d}\sigma\right)\mathrm{d}s = -\int_0^\tau\int_0^s f'(\sigma)\mathrm{d}\sigma\mathrm{d}s,$$

and can be bounded with assumption (4a)

$$\left\|\int_0^\tau\int_0^s \mathrm{e}^{\sigma\mathcal{A}}[\mathcal{A},\mathcal{C}]\mathrm{e}^{(\tau-\sigma)\mathcal{A}}u(t)\mathrm{d}\sigma\mathrm{d}s\right\| \le c\tau^2. \tag{12}$$

Furthermore, by employing assumptions (4c) and (4d), the remaining terms in P can be estimated as

$$\|\tau^2\mathcal{C}^2\varphi_2(\tau\mathcal{C})\mathrm{e}^{\tau\mathcal{A}}u(t)\| = \|\tau^2\varphi_2(\tau\mathcal{C})\mathcal{C}^2\mathrm{e}^{\tau\mathcal{A}}u(t)\| \le c\tau^2 \tag{13}$$

and

$$\left\|\int_0^\tau \mathrm{e}^{s\mathcal{A}}\mathcal{C}\int_0^{\tau-s} \mathrm{e}^{\sigma\mathcal{A}}\mathcal{C}\mathrm{e}^{(\tau-s-\sigma)(\mathcal{A}+\mathcal{C})}u(t)\mathrm{d}\sigma\mathrm{d}s\right\| \le c\tau^2. \tag{14}$$

Taking all together, we have shown that $P = \mathcal{O}(\tau^2)$.

As regards Q, by setting $g(s) = \mathrm{e}^{s(\mathcal{A}+\mathcal{C})}\mathcal{B}\mathrm{e}^{(\tau-s)(\mathcal{A}+\mathcal{C})}u(t)$ and proceeding in the same way as for (11) we obtain

$$\begin{aligned} Q &= \tau \mathrm{e}^{\tau\mathcal{C}}\mathrm{e}^{\tau\mathcal{A}}\mathcal{B}u(t) - \int_0^\tau g(s)\,\mathrm{d}s \\ &= \tau \mathrm{e}^{\tau\mathcal{C}}\mathrm{e}^{\tau\mathcal{A}}\mathcal{B}u(t) - \tau g(\tau) - \int_0^\tau\int_\tau^s g'(\sigma)\,\mathrm{d}\sigma\mathrm{d}s \\ &= \tau \mathrm{e}^{\tau\mathcal{C}}\mathrm{e}^{\tau\mathcal{A}}\mathcal{B}u(t) - \tau\mathrm{e}^{\tau(\mathcal{A}+\mathcal{C})}\mathcal{B}u(t) \\ &\quad - \int_0^\tau\int_\tau^s \mathrm{e}^{\sigma(\mathcal{A}+\mathcal{C})}[\mathcal{A}+\mathcal{C},\mathcal{B}]\mathrm{e}^{(\tau-\sigma)(\mathcal{A}+\mathcal{C})}u(t)\,\mathrm{d}\sigma\mathrm{d}s. \end{aligned}$$

The double integral is bounded with the help of assumption (4e) by $c\tau^2$. For the remaining two terms, we use that

$$\mathrm{e}^{\tau\mathcal{C}}\mathrm{e}^{\tau\mathcal{A}}\mathcal{B}u(t) = \mathrm{e}^{\tau\mathcal{A}}\mathcal{B}u(t) + \tau\mathcal{C}\varphi_1(\tau\mathcal{C})\mathrm{e}^{\tau\mathcal{A}}\mathcal{B}u(t)$$

and employ once more the variation-of-constants formula

$$\mathrm{e}^{\tau(\mathcal{A}+\mathcal{C})}\mathcal{B}u(t) = \mathrm{e}^{\tau\mathcal{A}}\mathcal{B}u(t) + \int_0^\tau \mathrm{e}^{s\mathcal{A}}\mathcal{C}\mathrm{e}^{(\tau-s)(\mathcal{A}+\mathcal{C})}\mathcal{B}u(t)\,\mathrm{d}s.$$

Assumption (4b) shows that their difference is again bounded by $c\tau^2$. From this we conclude the assertion. □

Assumption 1 guarantees that the semigroups, generated by $\mathcal{A}$, $\mathcal{B}$, and $\mathcal{C}$ satisfy the bounds

$$\|\mathrm{e}^{t\mathcal{A}}\| \le M_1\mathrm{e}^{t\omega_1}, \quad \|\mathrm{e}^{t\mathcal{B}}\| \le \mathrm{e}^{t\omega_2}, \quad \|\mathrm{e}^{t\mathcal{C}}\| \le M_3\mathrm{e}^{t\omega_3}, \qquad t \ge 0$$

for some constants $M_1 \ge 1$, $M_3 \ge 1$, ω_1, ω_2, and ω_3. Moreover, it is possible to choose an equivalent norm $\|\cdot\|_*$ on X such that $\|\mathrm{e}^{t\mathcal{A}}\|_* \le \mathrm{e}^{t\omega_{\mathcal{A}}}$. Unfortunately, this is still not enough to prove stability, in general. Therefore, we impose an additional assumption.

Assumption 2. There is a constant ω_C such that $\|\mathrm{e}^{t\mathcal{C}}\|_* \le \mathrm{e}^{t\omega_\mathcal{C}}$ for all $t \ge 0$.

Under this additional assumption, it is easy to show stability.

Theorem 2 (Stability). *Under Assumptions 1 and 2, Lie splitting is stable, i.e., there is a constant C such that*

$$\left\|\left(\mathrm{e}^{\tau\mathcal{C}}\mathrm{e}^{\tau\mathcal{A}}\mathrm{e}^{\tau\mathcal{B}}\right)^j\right\| \le C \tag{15}$$

for all $j \in \mathbb{N}$ and $\tau \ge 0$ satisfying $0 \le j\tau \le T$. □

Proof. Our assumptions imply that $\left\|\mathrm{e}^{\tau\mathcal{C}}\mathrm{e}^{\tau\mathcal{A}}\mathrm{e}^{\tau\mathcal{B}}\right\|_* \le \mathrm{e}^{\tau(\omega_\mathcal{A}+\omega_\mathcal{B}+\omega_\mathcal{C})}$ from which the assertion follows. □

From consistency and stability, convergence follows in a standard way.

Theorem 3 (Global error bound). *Under Assumptions 1 and 2, the Lie splitting discretization* (3) *of the initial value problem* (2) *is convergent of order 1, i.e., there exists a constant C such that*

$$\|u_n - u(t_n)\| \le C\tau,$$

for all $n \in \mathbb{N}$ and $\tau > 0$ satisfying $0 \le n\tau = t_n \le T$.

Proof. We express the global error with the help of a telescopic sum

$$\begin{aligned} u_n - u(t_n) &= \Big(\big(\mathrm{e}^{\tau\mathcal{C}} \mathrm{e}^{\tau\mathcal{A}} \mathrm{e}^{\tau\mathcal{B}} \big)^n - \mathrm{e}^{n\tau(\mathcal{A}+\mathcal{B}+\mathcal{C})} \Big) u(0) \\ &= \sum_{j=0}^{n-1} \big(\mathrm{e}^{\tau\mathcal{C}} \mathrm{e}^{\tau\mathcal{A}} \mathrm{e}^{\tau\mathcal{B}} \big)^{n-j-1} \big(\mathrm{e}^{\tau\mathcal{C}} \mathrm{e}^{\tau\mathcal{A}} \mathrm{e}^{\tau\mathcal{B}} u(t_j) - u(t_{j+1}) \big) \end{aligned}$$

and use the estimates (7) and (15). □

3. Example: the magnetic Schrödinger equation

The electromagnetic field in $\mathbb{R}^3$ is the combination of the electric field E and the magnetic field B. Both fields depend on time and space, in general. Mathematically, they are given by a scalar potential V and a vector potential A, respectively

$$E = -\nabla V - \frac{\partial A}{\partial t}, \qquad B = \nabla \times A.$$

Making use of the fact that we can impose conditions on the potentials as long as we do not affect the resulting fields, we will apply the following transformations

$$\begin{aligned} \tilde{u}(t,x) &= u(t,x)\,\mathrm{e}^{\mathrm{i}\lambda(t,x)}, \\ \tilde{A}(t,x) &= A(t,x) + \varepsilon\nabla\lambda(t,x), \\ \tilde{V}(t,x) &= V(t,x) - \varepsilon\partial_t\lambda(t,x). \end{aligned} \tag{16}$$

One natural choice is to impose a so-called Coulomb gauge, i.e., to select λ in such a way that $\nabla \cdot \tilde{A} = 0$. Consequently, this gauge λ has to satisfy the Poisson equation $\varepsilon\Delta\lambda = -\nabla \cdot A$.

Applying (16) to the Schrödinger equation (1) and dropping right away the tildes, we obtain the following problem

$$\begin{aligned} &\mathrm{i}\varepsilon\partial_t u = -\frac{\varepsilon^2}{2}\Delta u + \mathrm{i}\varepsilon A \cdot \nabla u + \frac{1}{2}|A|^2 u + Vu, \quad t \in [0,T], \\ &u(0,x) = u_0(x) \end{aligned} \tag{17}$$

with a divergence-free vector potential A.

We are now in the position to give a precise formulation of the three subproblems that are used in our splitting. Henceforth, they will be called potential, kinetic and advection step, respectively:

$$\partial_t u = \mathcal{B}u = -\frac{\mathrm{i}}{\varepsilon}\left(\frac{1}{2}|A|^2 + V\right)u, \tag{18a}$$

$$\partial_t u = \mathcal{A}u = \frac{\mathrm{i}\varepsilon}{2}\Delta u, \tag{18b}$$

$$\partial_t u = \mathcal{C}u = A\cdot\nabla u, \qquad \nabla\cdot A = 0. \tag{18c}$$

The kinetic step (18b) can be handled analytically in Fourier space, whereas the potential step (18a) is easily performed in physical space. For the advection step (18c) we will present three modifications of a semi-Lagrangian method in section 5 below.

An important feature of (1) and (17) is the conservation of mass $m = \|u(t,\cdot)\|_{L^2}^2$, i.e., $\frac{\partial}{\partial t}\|u(t,\cdot)\|_{L^2}^2 = 0$. The split step solution based on (18) is mass conserving as well. Indeed, the kinetic step preserves the L^2 norm due to Parseval's identity. The modulus of the solution of the potential step is preserved, and we are also able to show that the advection step conserves the mass. This is seen by multiplying (18c) by $\overline{u}$

$$\overline{u}\,\partial_t u - \overline{u}\,A\cdot\nabla u = 0$$

and adding this equation to its complex conjugate, which results in

$$\partial_t|u|^2 = A\cdot\nabla|u|^2.$$

Integrating this last equation by parts shows

$$\partial_t\|u\|_{L^2}^2 = \int \partial_t|u|^2\,\mathrm{d}x = \int A\cdot\nabla|u|^2\,\mathrm{d}x = -\int |u|^2\nabla\cdot A\,\mathrm{d}x = 0,$$

where the last identity follows from the Coulomb gauge.

Henceforth, we consider (17) and (18) on the hyperrectangle $\Omega = \Pi_{i=1}^d[a_i, b_i)$, subject to periodic boundary conditions. In particular, the potentials V and A are assumed to be periodic functions on Ω. Then, all what has been said in this section remains valid.

We finally remark that our splitting approach also works if A is not divergence-free. In this case the potential operator is given by

$$\mathcal{B} = -\frac{\mathrm{i}}{\varepsilon}\left(\frac{1}{2}|A|^2 + V\right) + \frac{1}{2}\nabla\cdot A,$$

whereas the other two operators stay the same. However, in this case, we will lose the conservation of mass for the potential step.

4. Space discretization, potential and kinetic step

We discretize the hyperrectangle $\Omega = \prod_{i=1}^d [a_i, b_i)$ by a regular grid. For $1 \le i \le d$, let $N_i \ge 2$ be an even integer and let

$$I_N = \mathbb{Z}^d \cap \prod_{i=1}^{d} \left[-\tfrac{N_i}{2}, \tfrac{N_i}{2} \right). \tag{19}$$

For $j = (j_1, \dots, j_d) \in I_N$ we consider the grid points x^j with components

$$x_i^j = \frac{a_i + b_i}{2} + \frac{j_i}{N_i}(b_i - a_i), \qquad 1 \le i \le d.$$

For performing the potential step, we solve the ordinary differential equation (18a) at each grid point x^j. More precisely, starting with an initial value v at time t_n, we solve

$$\dot{w}(s) = -\frac{\mathrm{i}}{\varepsilon} \left(\frac{1}{2} \left| A(t_n + s, x^j) \right|^2 + V(t_n + s, x^j) \right) w(s), \quad w(0) = v(x^j)$$

to obtain

$$\left(\mathrm{e}^{\tau \mathcal{B}} v \right) (x^j) = w(\tau).$$

If the potentials A and V are time-independent, the analytic solution is readily available. Otherwise, a quadrature method (up to machine precision) can be employed.

The kinetic step is approximated in Fourier space. For a given function

$$v \colon \prod_{i=1}^{d} [a_i, b_i) \to \mathbb{C},$$

let $\hat{v}_k$ denote its Fourier coefficients, i.e.

$$v(x) = \sum_{k \in I_N} \hat{v}_k E_k(x), \qquad E_k(x) = \prod_{i=1}^{d} \frac{\mathrm{e}^{2\pi \mathrm{i} k_i (x_i - a_i)/(b_i - a_i)}}{\sqrt{b_i - a_i}},$$

where $x = (x_1, \dots, x_d)$. Further, let

$$\lambda_i = \frac{\mathrm{i}\varepsilon}{2} \left(\frac{2\pi k_i}{b_i - a_i} \right)^2.$$

The Fourier coefficients of $\mathrm{e}^{\tau \mathcal{A}} v$ are then given by $\mathrm{e}^{\tau \lambda_i} \hat{v}_{k_i}$. The transformation between physical and Fourier space is usually carried out with the fast Fourier transform.

5. Advection step

In this section we describe the solution of the advection step

$$\begin{cases} \partial_t v(t,x) = A(x)\cdot\nabla v(t,x), \quad t\in[0,\tau], \\ v(0,x) = v_0(x). \end{cases} \tag{20}$$

Our approach is based on the method of characteristics, i.e., we make use of the curves $s\mapsto x(s)\in\mathbb{R}^d$ satisfying the d-dimensional system of ordinary differential equations

$$\dot{x}(s) = -A(x(s)).$$

Since the solution of the advection equation (20) is constant along characteristics, we have $v(\tau,x^j) = v(0,x^j(0))$ for each grid point x^j, $j\in I_N$, where $x^j(0)$ denotes the solution of

$$\begin{cases} \dot{x}^j(s) = -A(x^j(s)), \quad s\in[0,\tau], \\ x^j(\tau) = x^j \end{cases} \tag{21}$$

at $s=0$. This system can be solved once and for all for each grid point with an explicit method at high precision, if the time step τ is kept constant. However, since $x^j(0)$ is not a grid point, in general, the value $v(0,x^j(0)) = v_0(x^j(0))$ has to be recovered. We describe here three different procedures for achieving the evaluation of $v_0(x^j(0))$ at the set of $\prod_i N_i$ points $\{x^j(0)\}_j$. For the sake of simplicity, we only describe the one-dimensional case in detail. However, we also report the overall computational cost for the general d-dimensional case.

We remark that the same approach can be used for time dependent potentials $A(t,x)$. Instead of (21) one has to solve the non-autonomous problem

$$\begin{cases} \dot{x}^j(s) = -A(t_n+s,x^j(s)), \quad s\in[0,\tau], \\ x^j(\tau) = x^j. \end{cases} \tag{22}$$

Its numerical solution at $s=0$ is again used to define the sought-after approximation $v(\tau,x^j) = v(0,x^j(0))$.

5.1. Direct Fourier series evaluation

Since the initial value $v_0(x)$ of the advection step is the result of the solution of the kinetic step, the function v_0 is known through its Fourier coefficients $\{\hat{v}_k\}_k$. It is therefore possible to directly evaluate

$$v_0(x^j(0)) = \sum_{k\in I_N} \hat{v}_k E_k(x^j(0)). \tag{23}$$

In the d-dimensional case, the $\prod_i N_i^2$ values $E_k(x^j(0))$ can be precomputed once and for all, if the time step τ is constant. The evaluation cost of (23) at the point set $\{x^j(0)\}_j$ is then $\mathcal{O}(\prod_i N_i^2)$ at each time step.

5.2. Local polynomial interpolation

Another possibility (see, for instance, [13, 9, 3]) is local polynomial interpolation. It is possible to evaluate $v_0(x)$ at the grid points $\{x^j\}_j$ with an inverse fast Fourier transform of cost $\mathcal{O}\left(N_1 \cdot \ldots \cdot N_d \cdot (\log N_1 + \ldots + \log N_d)\right)$. An approximation of the values $v_0(x^j(0))$ can then be obtained by local polynomial interpolation

$$v_0(x^j(0)) \approx \sum_{k \in I_p} v_0(x^{j+k}) L_{j+k}(x^j(0)). \tag{24}$$

Here $\{x^{j+k}\}_k$ is the set of the p grid points, p even, satisfying

$$x^{j-p/2} < \ldots < x^{j-1} \leq x^j(0) < x^j < \ldots < x^{j+p/2-1},$$

and L_{j+k} denotes the elementary Lagrange polynomial of degree $p-1$ that takes the value one at x^{j+k} and zero at all the other $p-1$ points. Of course, the points x^{j+k} and the corresponding values $v_0(x^{j+k})$ have to be taken by periodicity if necessary.

In the d-dimensional case, for a constant time step τ it is possible to precompute once and for all the elementary Lagrange polynomials at the points $\{x^j(0)\}_j$ (for a total amount of $p^d \prod_i N_i$ values). Then, the evaluation of (24) at each time step requires $\mathcal{O}(p^d \prod_i N_i)$ operations.

5.3. Fourier series evaluation by NFFT

The third explored possibility is the evaluation of (23) by means of an approximate fast Fourier transform. Among others, we tested the nonequispaced fast Fourier transform (NFFT) by Keiner, Kunis and Potts [10]. The computational cost of such an approach is $\mathcal{O}\big(N_1 \cdot \ldots \cdot N_d \cdot (\log N_1 + \ldots + \log N_d + \log|\epsilon|)\big)$, where ϵ is the desired accuracy.

For the readers' convenience, we briefly sketch the NFFT algorithm in one dimension, using the original notation of [10]. Given some coefficients $\{\hat{f}_k\}_{k \in I_N}$, N even, and a set of *arbitrary* points $\{x^j\}_j \subset \left[-\frac{1}{2}, \frac{1}{2}\right)$, the aim is a fast evaluation of the one-periodic trigonometric polynomial

$$f(x) = \sum_{k \in I_N} \hat{f}_k \mathrm{e}^{-2\pi \mathrm{i} k x} \tag{25}$$

at the points $\{x^j\}_j$. In the first step, $f(x)$ is replaced with the ansatz

$$s_1(x) = \sum_{\ell \in I_n} g_\ell \, \tilde{\varphi}\left(x - \tfrac{\ell}{n}\right), \quad \sigma \geq 2, \quad n = \sigma N \text{ even},$$

where $\{g_\ell\}_\ell$ are some coefficients to be defined later and

$$\tilde{\varphi}(x) = \sum_{r \in \mathbb{Z}} \varphi(x + r)$$

is the one-periodic version of a *window* function $\varphi\colon \mathbb{R} \to \mathbb{R}$. The window function φ is chosen in such a way that $\tilde{\varphi}$ has a uniformly convergent Fourier series

$$\tilde{\varphi}(x) = \sum_{k\in\mathbb{Z}} c_k(\tilde{\varphi})\mathrm{e}^{-2\pi\mathrm{i}kx}.$$

The default window function used by NFFT is the so called *Keiser–Bessel function*

$$\varphi(x) = \frac{1}{\pi}\begin{cases} \dfrac{\sinh(\beta\sqrt{m^2-n^2x^2})}{\sqrt{m^2-n^2x^2}} & \text{for } |x| < \dfrac{m}{n}, \\ \dfrac{\sin(\beta\sqrt{n^2x^2-m^2})}{\sqrt{n^2x^2-m^2}} & \text{for } |x| > \dfrac{m}{n}, \\ \beta & \text{for } |x| = \dfrac{m}{n} \end{cases}$$

with the *shape* parameter $\beta = \pi(2 - 1/\sigma)$. The value of m depends on the desired accuracy ϵ and is chosen $m = 8$ for double precision. The *oversampling* factor σ is defined by

$$\sigma = \frac{2^{\lceil \log_2 2N \rceil}}{N}.$$

That is, $n = \sigma N$ is the smallest power of two with $2 \le \sigma < 4$. Now we plug the Fourier series expansion of $\tilde{\varphi}(x)$ into $s_1(x)$ in order to get

$$\begin{aligned} s_1(x) &= \sum_{\ell\in I_n} g_\ell\, \tilde{\varphi}\left(x - \tfrac{\ell}{n}\right) \\ &= \sum_{\ell\in I_n} g_\ell \sum_{k\in\mathbb{Z}} c_k(\tilde{\varphi})\mathrm{e}^{-2\pi\mathrm{i}k\left(x-\frac{\ell}{n}\right)} \\ &= \sum_{k\in\mathbb{Z}} \left(\sum_{\ell\in I_n} g_\ell \mathrm{e}^{2\pi\mathrm{i}k\frac{\ell}{n}}\right) c_k(\tilde{\varphi})\mathrm{e}^{-2\pi\mathrm{i}kx} \end{aligned}$$

and apply a cutoff in the frequency domain

$$s_1(x) \approx \sum_{k\in I_n} \left(\sum_{\ell\in I_n} g_\ell \mathrm{e}^{2\pi\mathrm{i}k\frac{\ell}{n}}\right) c_k(\tilde{\varphi})\mathrm{e}^{-2\pi\mathrm{i}kx} = \sum_{k\in I_n} \hat{g}_k c_k(\tilde{\varphi})\mathrm{e}^{-2\pi\mathrm{i}kx}. \qquad (26)$$

Comparing now equations (25) and (26), we see that the coefficients $\{\hat{g}_k\}_k$ are simply given by

$$\hat{g}_k = \begin{cases} \dfrac{\hat{f}_k}{c_k(\tilde{\varphi})}, & k \in I_N, \\ 0, & k \in I_n \setminus I_N, \end{cases}$$

and the values $\{g_\ell\}_\ell$ can be recovered by a fast Fourier transform of length n. The parameter m is then used as a *cutoff* to approximate in practice the window function $\varphi(x)$ with

$$\psi(x) = \varphi(x)\chi_{\left[-\frac{m}{n},\frac{m}{n}\right]}(x).$$

In this way, $s_1(x)$ is further approximated by

$$s_1(x) \approx s(x) = \sum_{\ell \in I_n} g_\ell \, \tilde{\psi}\left(x - \tfrac{\ell}{n}\right).$$

Now we use that $\tilde{\psi}$ vanishes outside of $-\frac{m}{n} \le x - \frac{\ell}{n} \le \frac{m}{n}$. Thus, for fixed x^j, the above sum contains at most $2m+1$ terms different from zero. Finally, $s(x)$ is evaluated at the set $\{x^j\}_j$, providing the desired approximation of $\{f(x^j)\}_j$.

6. Application to the magnetic Schrödinger equation

In this section we exemplify the assumptions of Theorem 3 for the magnetic Schrödinger equation (17). For this purpose, we choose $X = L^2(\Omega)$ with $\Omega = \prod_{i=1}^d [a_i, b_i)$ and assume that the potentials A and V are sufficiently smooth. Note that the potential operator $\mathcal{B}$ is bounded, whereas the kinetic operator $\mathcal{A}$ and the advection operator $\mathcal{C}$ are both unbounded. We start with the verification of Assumption 1.

◇ Condition (4a): Since $\mathrm{e}^{t\mathcal{A}} w$ is the exact solution of the problem $\partial_t u = \mathcal{A}u$, $u(0) = w$, it preserves the smoothness of the initial data. Further, the commutator $[\mathcal{A}, \mathcal{C}]$ is a second-order differential operator

$$\begin{aligned}[\mathcal{A}, \mathcal{C}]u &= \frac{\mathrm{i}\varepsilon}{2}[\Delta, A \cdot \nabla]u \\ &= \frac{\mathrm{i}\varepsilon}{2}\Big(\Delta(A \cdot \nabla u) - A \cdot \nabla(\Delta u)\Big).\end{aligned}$$

So, we need to assume that the initial data are twice differentiable.

◇ Conditions (4b), (4c), and (4d): As $\mathcal{C}$ is a first-order differential operator, it is again sufficient to require that the initial data are twice differentiable.

◇ Condition (4e): The commutator is a second-order differential operator

$$[\mathcal{A} + \mathcal{C}, \mathcal{B}]u = \left[\frac{\mathrm{i}\varepsilon}{2}\Delta + A \cdot \nabla, -\frac{\mathrm{i}}{\varepsilon}\left(\frac{1}{2}|A|^2 + V\right)\right]u,$$

so the same smoothness as before is required.

Stability is easily verified. From the conservation of mass discussed at the end of section 3, we get $\|\mathrm{e}^{\tau\mathcal{A}}\|_{L^2} = 1$ and $\|\mathrm{e}^{\tau\mathcal{C}}\|_{L^2} = 1$.

Note that the above bound for the advection semigroup only holds in the Coulomb gauge setting. However, by the method of characteristics, the solution of the advection step is of the form $u(t, x(t)) = u_0(x(0))$, where $x(t) = x(0) + tA(x(0)) + \mathcal{O}(t^2)$. Setting $\xi = x(0)$, we have

$$\|u\|_{L^2}^2 = \int_\Omega |u(x)|^2 \mathrm{d}x = \int_\Omega |u_0(\xi)|^2 \mathrm{d}x = \int_\Omega |u_0(\xi)|^2 \left|\det\big(I + tA'(\xi) + \mathcal{O}(t^2)\big)\right| \mathrm{d}\xi.$$

Under the assumption that the partial derivatives of A are bounded, we have

$$\|u\|_{L^2}^2 \le \|u_0\|_{L^2}^2 + Ct\|u_0\|_{L^2}^2 \le (1 + Ct)\|u_0\|_{L^2}^2 \le \mathrm{e}^{2t\omega_{\mathcal{C}}}\|u_0\|_{L^2}^2,$$

which is exactly the weaker bound required in Assumption 2.

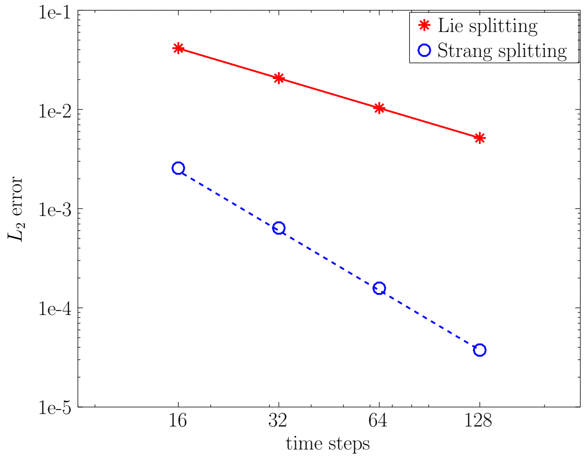

Figure 1: Temporal errors (stars, circles) for the Lie and Strang splitting methods and reference orders (lines) for the one-dimensional example.

7. Numerical experiments

The first numerical example is a variation of [9, Example 2]. The vector potential is chosen as $A(x) = \sin(2\pi x)/5 + 1/5$ and the scalar potential as $V(x) = \cos(2\pi x)/5 + 4/5$. The initial value is $u_0(x) = \sqrt{\rho_0(x)}\exp(\mathrm{i}S_0(x)/\varepsilon)$, where

$$\rho_0(x) = \mathrm{e}^{-50\left(x-\frac{1}{2}\right)^2}, \quad S_0(x) = -\frac{\log\left(\mathrm{e}^{5(x-\frac{1}{2})} + \mathrm{e}^{-5(x-\frac{1}{2})}\right)}{5}, \quad \varepsilon = \frac{1}{128}.$$

Note that this initial value is not periodic. However, due to the exponential decay of $\rho_0(x)$, the problem can be solved numerically up to time $T = 0.42$ in the space interval $[0, 1]$ by assuming periodic boundary conditions. The Coulomb gauge transformation yields

$$\lambda(x) = \frac{\cos(2\pi x)}{10\pi\varepsilon}.$$

In Figure 1 we plot the global errors of Lie splitting at the final time $T = 0.42$ for various time steps and $N = 2048$ spatial discretization points. The reference solution was obtained with 512 time steps. We include in this figure the error behavior of Strang splitting, defined by

$$u_{n+1} = \mathrm{e}^{\frac{\tau}{2}\mathcal{B}}\mathrm{e}^{\frac{\tau}{2}\mathcal{A}}\mathrm{e}^{\tau\mathcal{C}}\mathrm{e}^{\frac{\tau}{2}\mathcal{A}}\mathrm{e}^{\frac{\tau}{2}\mathcal{B}}u_n. \tag{27}$$

In this double logarithmic diagram, the errors of a method lie on a straight line of slope q, where q denotes the order of the method. Both, Lie and Strang splitting show their expected orders of convergence. Lie splitting has order one, as proved in Theorem 3, whereas Strang splitting converges with order two.

Note that the computationally most expensive task in the employed splitting approach is the advection step. Therefore, we order the steps in (27) in such a way that the advection equation is solved only once in each time step. In this way, Strang splitting provides much more accuracy without being significantly more expensive than Lie splitting.

Next, we compare the three different numerical realizations of the advection step, namely by local interpolation, by direct Fourier series evaluation (DFT) and by NFFT. In Table 1 we report the error in mass conservation and the required CPU time for various values of N. The number of time steps is fixed to $n = 128$. The error in mass conservation is measured as the maximum deviation from the initial mass on the discrete level (l^2 in space and l^∞ in time).

		interpolation				Fourier	
N		$p = 2$	$p = 4$	$p = 6$	$p = 8$	DFT	NFFT
128	mass	1.4e-01	1.8e-02	2.1e-03	2.8e-04	2.8e-15	8.6e-14
	CPU	0.13	0.12	0.12	0.12	0.10	0.16
256	mass	9.4e-02	2.7e-03	7.2e-05	2.5e-06	2.0e-15	1.0e-14
	CPU	0.13	0.13	0.13	0.14	0.19	0.17
512	mass	5.2e-02	2.9e-04	2.0e-06	1.8e-08	3.6e-15	1.7e-14
	CPU	0.16	0.19	0.17	0.16	0.27	0.19
1024	mass	1.6e-02	1.8e-05	3.0e-08	9.6e-11	4.0e-15	5.5e-14
	CPU	0.22	0.23	0.23	0.24	0.56	0.23
2048	mass	4.2e-03	1.1e-06	4.9e-10	3.8e-12	3.3e-15	1.3e-14
	CPU	0.36	0.37	0.37	0.37	1.42	0.33

Table 1: Error in mass conservation and CPU time (in seconds) for the one-dimensional numerical example.

Due to the compressive behavior of $S_0'(x)$, which acts as an initial velocity, the evolution develops caustics and the numerical solution requires a sufficiently large number N of Fourier modes in order to reproduce accurate physical observables. While DFT and NFFT always preserve the mass almost up to machine precision, the polynomial methods become comparable only with the largest tested value of N and at polynomial degree 7. For this degree, they are slightly more expensive than the NFFT approach.

The second numerical experiment is set in the two-dimensional domain

$[-5, 5]^2$ with

$$\begin{aligned} A_1(x,y) &= -3\sin\left(\tfrac{2\pi(y+5)}{10}\right), \\ A_2(x,y) &= 3\sin\left(\tfrac{2\pi(x+5)}{10}\right), \\ V(x,y) &= 20\cos\left(\tfrac{2\pi(x+5)}{10}\right) + 20\cos\left(\tfrac{2\pi(y+5)}{10}\right) + 40, \end{aligned}$$

and initial value

$$u_0(x,y) = \sqrt{\tfrac{\sqrt{10}}{\pi}}\exp\left(-\tfrac{\sqrt{10}}{2}\left((x-1)^2 + y^2\right)\right).$$

The semi-classical parameter is chosen $\varepsilon = 1$, the final time $T = 50$ and the number of time steps $n = 1000$. In Table 2 we compare the three methods that only differ in the treatment of the advection step. In particular, we compare the behavior of tensor interpolation at 4×4 and 6×6 points with direct Fourier series evaluation and NFFT with the default value $m = 8$ and the smaller values $m = 6$ and $m = 4$, respectively.

				Fourier			
		interpolation			NFFT		
$N_1 = N_2$		$p = 4$	$p = 6$	DFT	$m = 8$	$m = 6$	$m = 4$
128	mass	1.0e-01	2.5e-03	9.9e-11	1.0e-10	2.4e-10	2.2e-07
	CPU	25.2	33.5	174.3	23.7	22.8	20.6
256	mass	6.9e-03	3.9e-05	1.3e-08	1.3e-08	2.0e-08	2.5e-02
	CPU	101.7	117.9	2254	99.6	85.6	87.7
512	mass	4.3e-04	6.2e-07	*	9.7e-11	2.5e-10	2.0e-07
	CPU	412.7	506.8	*	435.7	401.4	400.4
1024	mass	2.7e-05	9.6e-09	*	9.7e-11	2.5e-10	1.9e-07
	CPU	1796	2139	*	1948	1840	1709

Table 2: Error in mass conservation and CPU time (in seconds) for the two-dimensional numerical example.

We observe that, for this long-term simulation, the mass is always well conserved by the direct Fourier series evaluation and by NFFT with the default value $m = 8$. On the other hand, if m is halved, there is a significant degradation, especially with $N_1 = N_2 = 256$. The direct Fourier series evaluation is much more expensive than the other methods, being impracticable for $N_1 = N_2 \geq 512$. The interpolation methods roughly cost as much as the NFFT approach, but their mass preservation is by far worse.

The final numerical example is a three-dimensional variation of the previous one. In the domain $[-5,5]^3$, with $\varepsilon = 1$, we chose

$$\begin{aligned}
A_1(x,y,z) &= \sin\left(\tfrac{2\pi(y+5)}{10}\right) + \sin\left(\tfrac{2\pi(z+5)}{10}\right)\\
A_2(x,y,z) &= \sin\left(\tfrac{2\pi(x+5)}{10}\right) + \sin\left(\tfrac{2\pi(z+5)}{10}\right)\\
A_3(x,y,z) &= \sin\left(\tfrac{2\pi(x+5)}{10}\right) + \sin\left(\tfrac{2\pi(y+5)}{10}\right)\\
V(x,y,z) &= 20\cos\left(\tfrac{2\pi(x+5)}{10}\right) + 20\cos\left(\tfrac{2\pi(y+5)}{10}\right) + 20\cos\left(\tfrac{2\pi(z+5)}{10}\right) + 60,
\end{aligned}$$

and the initial value

$$u_0(x,y,z) = \tfrac{2^{3/8}}{\pi^{3/2}} \exp\left(-\tfrac{\sqrt{2}}{2}\left((x-1)^2 + y^2 + z^2\right)\right).$$

With this example, we also tested the option `PRE_FULL_PSI` of NFFT (see [10]).

		NFFT	
$N_1 = N_2 = N_3$		`PRE_PSI`	`PRE_FULL_PSI`
16	mass	6.1e-13	6.1e-13
	CPU	5.6	6.5
32	mass	8.2e-14	8.2e-14
	CPU	37.7	51.7
64	mass	7.1e-13	*
	CPU	396.5	*
128	mass	7.9e-09	*
	CPU	2976	*

Table 3: Error in mass conservation and CPU time (in seconds) for the three-dimensional example.

At the price of a full precomputation of the window functions, which requires a storage of $(2m+1)^3 \prod_i N_i$ double precision numbers, this option should allow an overall faster execution. In Table 3 we display the error of mass conservation and the CPU time for simulations up to $T = 5$ with 100 time steps. As expected, there is no difference in the mass conservation property between the two schemes. However, we never succeeded in getting the `PRE_FULL_PSI` version faster than the default one (named `PRE_PSI`). For $N_1 = N_2 = N_3 \geq 64$, it was even not possible to store the precomputed values in the RAM (8 GB). Nevertheless, the default implementation of NFFT, which requires a storage of $3(2m+1)\prod_i N_i$, works without any problem.

8. Conclusions

In this paper we considered the numerical solution of the linear Schrödinger equation with a vector potential. The structure of the problem suggested to use a splitting method involving three different parts, namely a multiplicative term coming from scalar potentials, the Laplacian, and the advective term due to the vector potential. After establishing convergence of Lie splitting for an abstract problem, we analysed the required assumptions in the specific case of the magnetic Schrödinger equation. For the advection step, the solution along the characteristic curves was approximated by a nonequispaced fast Fourier transform. It turned out to be as fast as local polynomial interpolation and as accurate as direct Fourier series evaluation in the mass conservation at discrete level. Therefore, it can be considered as a competitive tool in the solution of advection equations with the method of characteristics.

References

[1] W. Bao, S. Jin, and P. Markowich. On time-splitting spectral approximations for the Schrödinger equation in the semiclassical regime. *J. Comput. Phys.*, 175(2):487–524, 2002.

[2] L. Einkemmer and A. Ostermann. Convergence analysis of Strang splitting for Vlasov-type equations. *SIAM J. Numer. Anal.*, 52(1):140–155, 2014.

[3] L. Einkemmer and A. Ostermann. A splitting approach for the Kadomtsev–Petviashvili equation. *J. Comput. Phys.*, 299:716–730, 2015.

[4] E. Faou. *Geometric Numerical Integration and Schrödinger Equations.* European Mathematical Society (EMS), Zürich, 2012.

[5] E. Hairer, C. Lubich, and G. Wanner. *Geometric Numerical Integration. Structure-Preserving Algorithms for Ordinary Differential Equations.* Springer, Berlin, second edition, 2006.

[6] E. Hansen and A. Ostermann. Exponential splitting for unbounded operators. *Math. Comp.*, 78(267):1485–1496, 2009.

[7] T. Jahnke and C. Lubich. Error bounds for exponential operator splittings. *BIT*, 40(4):735–744, 2000.

[8] S. Jin, P. Markowich, and C. Sparber. Mathematical and computational methods for semiclassical Schrödinger equations. *Acta Numer.*, 20:121–209, 2011.

[9] S. Jin and Z. Zhou. A semi-Lagrangian time splitting method for the Schrödinger equation with vector potentials. *Commun. Inf. Syst.*, 13(3):247–289, 2013.

[10] J. Keiner, S. Kunis, and D. Potts. Using NFFT 3—A software library for various nonequispaced fast Fourier transforms. *ACM Trans. Math. Software*, 36(4):19:1–19:30, 2009.

[11] C. Lubich. *From Quantum to Classical Molecular Dynamics: Reduced Models and Numerical Analysis.* European Mathematical Society (EMS), Zürich, 2008.

[12] R.I. McLachlan and G.R.W. Quispel. Splitting methods. *Acta Numer.*, 11:341–434, 2002.

[13] E. Sonnendrücker, J. Roche, P. Bertrand, and A. Ghizzo. The semi-Lagrangian method for the numerical resolution of the Vlasov equation. *J. Comput. Phys.*, 149:201–220, 1999.

[14] G. Strang. On the construction and comparison of difference schemes. *SIAM J. Numer. Anal.*, 5(3):506–517, 1968.