

QUANTUM SIMULATION FOR QUANTUM DYNAMICS WITH  
ARTIFICIAL BOUNDARY CONDITIONS\*SHI JIN<sup>†</sup>, XIANTAO LI<sup>‡</sup>, NANA LIU<sup>§</sup>, AND YUE YU<sup>¶</sup>

**Abstract.** Quantum dynamics, typically expressed in the form of a time-dependent Schrödinger equation with a Hermitian Hamiltonian, is a natural application for quantum computing. However, when simulating quantum dynamics that involves the emission of electrons, it is necessary to use artificial boundary conditions (ABCs) to confine the computation within a fixed domain. The introduction of ABCs alters the Hamiltonian structure of the dynamics, and existing quantum algorithms cannot be directly applied since the evolution is no longer unitary. The current paper utilizes a recently introduced Schrödingerization method that converts non-Hermitian dynamics into a Schrödinger form for the artificial boundary problems [S. Jin, N. Liu, and Y. Yu, *Quantum Simulation of Partial Differential Equations via Schrödingerisation*, preprint, arXiv:2212.13969, 2022], [S. Jin, N. Liu, and Y. Yu, *Phys. Rev. A*, 108 (2023), 032603]. We implement this method for three types of ABCs, including the complex absorbing potential technique, perfectly matched layer methods, and Dirichlet-to-Neumann approach. We analyze the query complexity of these algorithms and perform numerical experiments to demonstrate the validity of this approach. This helps to bridge the gap between available quantum algorithms and computational models for quantum dynamics in unbounded domains.

**Key words.** quantum computing, artificial boundary conditions, time-dependent Schrödinger equation

**MSC codes.** 68Q12, 81P68, 35J10

**DOI.** 10.1137/23M1563451

**1. Introduction.** Quantum computing is an emerging technology that harnesses the laws of quantum mechanics to deliver unprecedented computational power [Hid19, NC02, Pre18, RP11]. Quantum algorithms operate on an  $n$ -qubit Hilbert space with dimension  $2^n$ , offering vast multidimensional spaces for computational

---

\*Submitted to the journal's Software, High-Performance Computing, and Computational Science and Engineering section April 3, 2023; accepted for publication (in revised form) April 3, 2024; published electronically July 10, 2024.

<https://doi.org/10.1137/23M1563451>

**Funding:** The first and third authors are also supported by NSFC grant 12341104, the Shanghai Jiao Tong University 2030 Initiative. The first author was partially supported by NSFC grant 12031013, the Shanghai Municipal Science and Technology Major Project (2021SHZDZX0102), and the Innovation Program of Shanghai Municipal Education Commission (2021-01-07-00-02-E00087). The second author was supported by a Seed Grant from the Institute of Computational and Data Science (ICDS) at Penn State. The third author acknowledges funding from the Science and Technology Program of Shanghai, China (21JC1402900). The fourth author was partially supported by China Postdoctoral Science Foundation (2022M712080) and the National Science Foundation for Young Scientists of China (12301561).

<sup>†</sup>School of Mathematical Sciences, Institute of Natural Sciences, MOE-LSC, Shanghai Jiao Tong University, Shanghai, 200240, People's Republic of China, and Shanghai Artificial Intelligence Laboratory, Shanghai, China (shijin-m@sjtu.edu.cn).

<sup>‡</sup>Department of Mathematics, The Pennsylvania State University, University Park, PA 16802 USA (xxl12@psu.edu).

<sup>§</sup>School of Mathematical Sciences, Institute of Natural Sciences, MOE-LSC, Shanghai Jiao Tong University, Shanghai, 200240, People's Republic of China, and Shanghai Artificial Intelligence Laboratory, Shanghai, China; and University of Michigan-Shanghai Jiao Tong University Joint Institute, Shanghai, 200240, China (nana.liu@quantumlah.org).

<sup>¶</sup>School of Mathematics and Computational Science Hunan Key Laboratory for Computation and Simulation in Science and Engineering, Key Laboratory of Intelligent Computing and Information Processing of Ministry of Education, Xiangtan University, Xiangtan, Hunan 411105, People's Republic of China (terenceyuyue@xtu.edu.cn).

models. Hence, it has a unique capability to handle large-scale scientific computing problems. A natural application is that of partial differential equations (PDEs) from time-dependent Schrödinger equations (TDSE), which follow unitary evolutions, and hence the wave functions can be coherently represented on quantum computers. Known as Hamiltonian simulations, a variety of efficient algorithms have been developed toward this end [BCC<sup>+</sup>15, BCK15, BCC<sup>+</sup>17, LC17, LC19, CGJ19, KSB19, AFL21, JLY23b, AFL22, FLT23].

While the TDSE is formulated in the entire physical space, in practice the computation has to be done in a bounded domain, typically in locations where the electron density is concentrated. For situations when the electrons are being emitted outside the computational domain, such as the photoionization process, or when they are being drawn away, such as the ionization process, this approach, however, can result in an extraordinarily large computational domain. An artificial boundary condition (ABC) that may absorb outgoing wave packets and, more importantly, keep the size of the computational domain to a minimum, is frequently used to solve such problems. A correctly set up ABC will yield the same result as if the computation were done in the infinite domain; e.g., see the survey paper [AAB<sup>+</sup>08].

Due to the introduction of the ABCs, the computer simulation is now following a dynamics that is no longer unitary. As a result, existing Hamiltonian simulation techniques cannot be directly applied. The goal of this paper is to bridge this emerging gap by mapping the non-Hamiltonian dynamics back to a Schrödinger equation or Hamiltonian system to enable the immediate Hamiltonian simulation capability. This is accomplished by using the recently introduced Schrödingerization technique [JLY22, JLY23b]. It is a generic framework for converting *any* linear partial (and ordinary) differential equation or dynamical system into a system of Schrödinger equations or a Hamiltonian system by going to the space which is one dimension higher, via the warped phase transformation. A key advantage of the Schrödingerization approach is its simplicity and generality, where the general Hamiltonian under which the system has a simple form. When the boundary value problem is considered, resulting in an inhomogeneous system that is not Hamiltonian, one just needs to introduce one auxiliary variable to make the system homogeneous, and then the Schrödingerization process still applies [JLY22, JLY23b].

In this paper, we explore three popular types of ABCs, including the complex absorbing potential method, perfectly matched layers method, and Dirichlet-to-Neumann approach, to facilitate the quantum implementation utilizing the Schrödingerization approach.

Another potential alternative to implementing the ABCs is to use quantum linear solvers (QLS) [HHL09, CKS17]. This can be done by first performing spatial discretization that reduces the models to a system of linear ordinary differential equations (ODEs), followed by time-marching schemes. Runge–Kutta methods, spectral (collocation) methods [CL20], multistep methods [BCOW17], and Dyson series [Kro23] are some of the important existing methods. In some earlier works [CL20], the analysis for this type of algorithm assumes that the matrix  $A$  in the linear ODE system is diagonalizable, and the final complexity involves a condition number of the eigenvector matrix. This was later improved by Krovi [Kro23], where the condition number is replaced by the following bound:

$$C(A) = \sup_{t \in [0, T]} \|\exp(At)\|.$$

The QLS-based approach was recently improved by An and Lin [AL22] with a better dependence on the condition number. In the context of ABCs, due to their dissipative

nature, the matrix  $A$  is typically stable, in that the real parts of the eigenvalues are nonpositive. Indeed, such a stability property is key to the development and analysis of ABCs [EM79]. The stability implies that  $C(A) = 1$ , thus completely removing the dependence of the complexity on the condition number.

However, QLS-based methods still require more involved steps, such as a truncation of Taylor series expansions or the construction of a unitary operation associated with the inverse of a matrix. The latter is often realized through the block-encoding formalism [GSLW19]. Such a QLS-based approach has been further improved to optimal dependence on the condition number [CAS<sup>+</sup>22]. In particular, it has been combined with the Dyson series expansion [BC22] to solve time-dependent differential equations. While these results might give similar query and gate complexities to the Schrödingerization approach, it is not straightforward to actually show explicitly how the inverse is obtained in the block-encoding formalism, which makes it difficult to perform in practice. This is in contrast to the Schrödingerization method, which is a much simpler scheme where the Hamiltonian to be simulated is *explicitly given*, which places the problem directly in the realm of Hamiltonian simulation. The cost in the Schrödingerization method is also *independent of the condition number*. Furthermore, Schrödingerization can be seamlessly integrated into new methods for time-dependent Hamiltonian simulation without using Taylor or Dyson series, only requires one extra dimension, and remains time continuous [CJL23]. This greatly simplifies techniques for quantum simulation of ODEs and PDEs with time-dependent coefficients in a way that is not possible with block-encoding.

We note that there are also very recent alternatives to QLS for simulating amplitude-encoded solutions of differential equations, for instance, schemes based on block-encoding [ALWZ22] or linear combination of unitaries (LCU) [ALL23]. The application of these methods to ABCs could provide a complementary viewpoint to our methods here.

The remainder of the paper is structured as follows. We first give a concise overview of some existing absorbing boundary conditions in section 2 and highlight the mathematical form of the corresponding quantum dynamics. In section 3, we elaborate on how each type of absorbing boundary condition can be converted back into a Schrödinger form. Section 4 is dedicated to exhibiting a series of numerical demonstrations, which, while limited to low-dimensional examples on classical devices, serve to verify the mathematical validity of the proposed methods. In section 5, we discuss potential Hamiltonian simulation algorithms suitable for the newly formulated Schrödinger equations and provide preliminary cost evaluations for these algorithms.

**2. A brief review of existing boundary conditions.** We first review several types of artificial boundary conditions that are commonly used in practice. The goal of these numerical techniques is to be able to simulate the dynamics in the entire space  $\mathbb{R}^d$ , i.e.,

$$(1) \quad i \frac{\partial}{\partial t} \psi(\mathbf{x}, t) = \hat{H} \psi(\mathbf{x}, t), \quad \mathbf{x} \in \mathbb{R}^d,$$

where  $\hat{H}$  is a Hamiltonian operator expressed in reduced units,

$$(2) \quad \hat{H} = -\frac{\nabla^2}{2} + V(\mathbf{x}, t).$$

We assume that the initial wave function  $\psi(\mathbf{x}, 0)$  is compactly supported in a subdomain, here denoted by  $\Omega \subset \mathbb{R}^d$ . On the other hand, due to the initial momentum, or

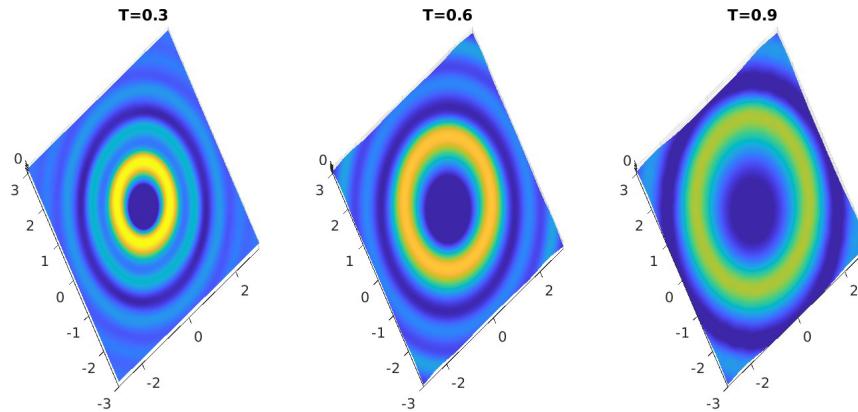


FIG. 1. Waves propagating outside the computational domain  $\Omega = [-3, 3] \times [-3, 3]$ .

the influence of the external potential, the wave function can spread well beyond  $\Omega$ , as shown in Figure 1. To avoid the need for continuously expanding the computational domain, it is necessary to develop suitable boundary conditions that can propagate wave packets beyond the domain's boundaries. These conditions, known as artificial boundary conditions (ABCs), are intended to prevent boundary reflections and hence interference with the dynamics inside the domain.

**2.1. Complex absorbing potentials.** The idea of the complex absorbing potential (CAP) is to introduce an artificial potential with a negative imaginary part to the TDSE. This modifies the TDSE (1) to

$$(3) \quad i \frac{\partial}{\partial t} \psi(\mathbf{x}, t) = \hat{H}\psi(\mathbf{x}, t) + W(\mathbf{x})\psi(\mathbf{x}, t), \quad \mathbf{x} \in D.$$

Here  $D \supset \Omega$  is a bigger domain that offers an absorbing layer,  $D \setminus \Omega$ , surrounding the computational domain  $\Omega$ . The complex potential  $W(\mathbf{x})$  is selected so that it is zero in the computational domain  $\Omega$  in order to keep the original dynamics unaltered, while in a surrounding absorbing layer, it has a negative imaginary part with magnitude slowly increasing away from the boundary  $\partial\Omega$  (see Figure 2 for an example). The hope is that the wave function at the boundary of the absorbing layer is sufficiently damped, at which point it can be simply set to zero. As popular techniques in computational chemistry, a wide variety of parametric forms for the imaginary potential have been proposed and extensively tested [MPNE04, YE18]. Some of the choices may include a nonzero real part in  $W(\mathbf{x})$  to offer more flexibility.

From the computational perspective, the CAP method only introduces a modification to the diagonal elements of  $\hat{H}$ . The implementation is quite straightforward, and the boundary conditions outside the absorbing layer can be chosen as the homogeneous Dirichlet boundary conditions.

**2.2. Perfectly matched layers methods.** Another classical strategy is the perfectly matched layer (PML) [Ber94], where one first constructs a buffer layer so that the outgoing waves in the computational domain are exactly preserved (perfect matching). Although conceptually similar to CAP, the PML, at least at the continuous level, can theoretically absorb the outgoing waves. The most common mathematical approach to ensure the perfectly matching property is to introduce a complex stretching of the spatial coordinate to derive a modified equation in the buffer layer, and then the resulting models are discretized simultaneously in the implementation.

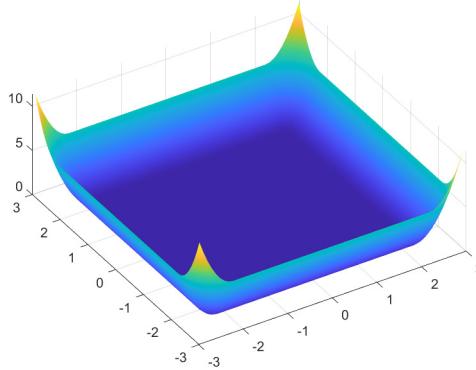


FIG. 2. An example of the complex absorbing potential  $W(x, y)$  for a two-dimensional domain (the amplitude is for  $-\text{Im}W$ ).  $\Omega = [-2.2, 2.2] \times [-2.2, 2.2]$  and  $D = [-3, 3] \times [-3, 3]$ . Motivated by the study in [RM96], we choose a polynomial form  $W(x, y) = -10i(|x| - 2.2)_+^2 - 10i(|y| - 2.2)_+^2$ .

PML has applications in many areas of science and engineering, such as electromagnetics, acoustics, and fluid dynamics, where accurate simulation of wave propagation is critical for understanding and predicting physical phenomena. For the TDSE (1), the PML has been developed by Zheng [Zhe07] and Nissen and Kreiss [NK11]. We follow the derivation in [NK11], and to briefly demonstrate the formulation, we first consider the one-dimensional Schrödinger equation,

$$(4) \quad i\partial_t\psi = -\frac{1}{2}\partial_x^2\psi + V(x).$$

A derivation of a PML starts by replacing the time derivative with multiplication by  $-i\omega$ , with  $\omega$  being the frequency variable. This changes the equation to

$$(5) \quad \omega\psi = -\frac{1}{2}\partial_x^2\psi + V(x).$$

In the second step, one performs a coordinate stretching,

$$\partial_x \longrightarrow \frac{1}{1 + i\sigma(x)/\omega}\partial_x,$$

which reduces (5) to

$$\begin{aligned} \omega\psi + i\sigma(x)\psi &= -\frac{1}{2}\partial_x \left( \frac{1}{1 + i\sigma(x)/\omega} \partial_x\psi \right) + (1 + i\sigma(x)/\omega)V(x), \\ &= -\frac{\partial_x^2}{2}\psi + V(x) + \frac{1}{2}\partial_x \left( \frac{i\sigma(x)/\omega}{1 + i\sigma(x)/\omega} \partial_x\psi \right) + \frac{i\sigma(x)}{\omega}V(x). \end{aligned}$$

The selection of the function  $\sigma$  may follow a similar recipe in the CAP method. Since we assume that  $V = 0$  outside the domain  $\Omega$ , we can drop the last term. We continue by defining an auxiliary function,

$$\chi = \frac{\sigma(x)/\omega}{1 + i\sigma(x)/\omega}\partial_x\psi.$$

Finally, by converting  $-i\omega$  back to the time derivative, we obtain a system of equations,

$$(6) \quad \begin{aligned} i\partial_t\psi &= \hat{H}\psi - i\sigma(x)\psi + \frac{i}{2}\partial_x\chi, \\ i\partial_t\chi &= \sigma(x)\partial_x\psi - i\sigma(x)\chi. \end{aligned}$$

We should point out that this idea of stretching the coordinate shows a close resemblance to the exterior coordinate scaling method [SE00, HRB07]. But the corresponding modified equations are quite different from the equations that we derive here.

For a two-dimensional problem, one can repeat the same steps in the second coordinate and generalize the model as

$$(7) \quad \begin{cases} i\partial_t\psi = \hat{H}\psi - i\sigma(x)\psi - i\sigma(y)\psi + \frac{i}{2}\partial_x\chi + \frac{i}{2}\partial_y\phi, \\ i\partial_t\chi = \sigma(x)\partial_x\psi - i\sigma(x)\chi, \\ i\partial_t\phi = \sigma(y)\partial_y\psi - i\sigma(y)\phi, \end{cases} \quad (x, y) \in D.$$

In the interior of  $\Omega$ , we have  $\sigma \equiv 0$ , and thus the new equations (7) will be reduced to the TDSE (1). Meanwhile, outside the buffer layer, i.e., at  $\partial D$ , the wave function can be set to zero. Notice that the perfectly matching property no longer holds after the numerical discretization. Therefore, a suitable choice of  $\sigma$ , together with a sufficiently large buffer layer, is crucial to the performance of a PML.

**2.3. The Dirichlet-to-Neumann approach.** The third type of ABC is based on a Dirichlet-to-Neumann (DtN) map. One can first apply a semidiscrete scheme to the TDSE (1), e.g., a standard finite difference scheme for the kinetic energy. This still yields an infinite system of ODEs. The next step is a domain decomposition. Toward this end, we define  $\Omega_I$  to be the grid points in  $\Omega$  and, similarly, let  $\Omega_{II}$  be the set of grid points outside the domain  $\Omega$ .

The semidiscrete model can be written in the following compact form:

$$(8) \quad \begin{cases} i\dot{\psi}_I(t) = H_{I,I}(t)\psi_I(t) + H_{I,II}(t)\psi_{II}(t), \\ i\dot{\psi}_{II}(t) = H_{II,I}(t)\psi_I(t) + H_{II,II}(t)\psi_{II}(t), \end{cases}$$

where the components of the wave functions are defined as  $\psi_I = [\psi(\mathbf{x}_k)]_{k \in \Omega_I}$  and  $\psi_{II} = [\psi(\mathbf{x}_k)]_{k \in \Omega_{II}}$ .  $\psi_I \in \mathbb{C}^{n_I}$  and  $\psi_{II} \in \mathbb{C}^{n_{II}}$ . We denote the discretization of the Hamiltonian operator in a partitioned form,

$$(9) \quad H = \begin{bmatrix} H_{I,I} & H_{I,II} \\ H_{II,I} & H_{II,II} \end{bmatrix}.$$

Since the Schrödinger equation (1) has been discretized in space, we use  $\cdot$  to denote the time derivatives hereafter. In addition, since we assumed  $V = 0$  in  $\Omega_{II}$ , the operator  $H_{II,II}$  only contains the kinetic energy (Laplacian).

For the grid points in the computational domain  $\Omega_I$ , one can separate out the grid points next to the boundary. Denote the set of those grid points by  $\Gamma$ :

$$\Gamma = \{\mathbf{x}_j \in \Omega_I : \text{if there exists } \mathbf{x}_k \in \Omega_{II} \text{ such that } H_{jk} \neq 0\}.$$

By reordering the grid points, one can arrange  $\psi_I$  as follows:

$$(10) \quad \psi_I = \begin{bmatrix} \psi_\Gamma \\ \psi_{I \setminus \Gamma} \end{bmatrix}.$$

As a result, the off-diagonal block of the Hamiltonian may be written as

$$H_{II,I} = [H_{II,\Gamma} \ H_{II,I \setminus \Gamma}] = [H_{II,\Gamma} \ \mathbf{0}].$$

By using Laplace transform, Wu and Li (see section II in [WL20]) derived the following exact boundary condition:

$$(11) \quad \begin{cases} \dot{\psi}_I(t) = -iH_{I,I}\psi_I(t) - iE^T\phi_\Gamma(t), \\ \phi_\Gamma(t) = \int_0^t \kappa(t-\tau)\psi_\Gamma(\tau)d\tau. \end{cases}$$

Here  $E$  is a restriction operator for extracting the components of a wave function that correspond to grid points at the boundary  $\Gamma$  from a function defined in  $\Omega_I$ , i.e.,

$$\psi_\Gamma = E\psi_I.$$

Furthermore,  $\phi_\Gamma$  represents the influence of the wave functions in  $\Omega_{II}$  on the wave functions in  $\Omega_I$ ,

$$\phi_\Gamma = H_{\Gamma,II}\psi_{II}(t).$$

But this connection is not needed in the implementation of (11).

The second equation in (11) is regarded as a discrete DtN map, where  $\kappa(t)$  is the real-time kernel function which corresponds to  $K(s)$  in the Laplace domain with

$$K(s) = -H_{\Gamma,II}(H_{II,II} - isI)^{-1}H_{II,\Gamma}.$$

Such ABCs can also be derived without the spatial discretization [HH04, AAB<sup>+</sup>08]. For example, for one-dimensional problems, the integral term can be expressed as a fractional derivative [Arn98, AAB<sup>+</sup>08]. But a discretization would be needed afterward.

Up to this point, the formulation is exact. In practice,  $K(s)$  is often treated using a Padé approximation. For instance, a zeroth-order approximation involves selecting an  $s_0$  as the interpolation point and letting

$$(12) \quad R = K(s_0) = -H_{\Gamma,II}(H_{II,II} - is_0I)^{-1}H_{II,\Gamma}.$$

Combined with (11), the ABC becomes

$$(13) \quad \dot{\psi}_I(t) = -iH_{I,I}\psi_I(t) - iE^T RE\psi_I(t).$$

This approximation also adds a potential, with the anti-Hermitian part being negative definite [WL20]. Thus it can be considered as a CAP method; but the Hermitian part is not necessarily zero, and the anti-Hermitian part is usually not diagonal.

From the first-order Padé approximation, one obtains the ABC as follows:

$$(14) \quad \begin{cases} \frac{d}{dt}\psi_I(t) = -iH_{I,I}\psi_I(t) - iE^T\phi_\Gamma(t), \\ \frac{d}{dt}\phi_\Gamma(t) = B\phi_\Gamma(t) + AE\psi_I(t). \end{cases}$$

By a straightforward interpolation at  $s = s_0$ , one has (see the proof of Theorem 3.2 in [WL20]),

$$(15) \quad A = -iH_{\Gamma,II}H_{II,\Gamma}, \quad B = s_0I - AR^{-1}.$$

We have presented the DtN approach as if  $H_{II,II}$  were a finite matrix. One can take an infinite volume limit by imposing a zero far-field condition. Another important observation is that the matrix  $R$  in (12) involves the operator  $H_{II,II}$  in the exterior, but the DtN map can be computed from a discrete boundary element equation [MR09, Li12], which only involves the wave function at the boundary. Such equations can be further simplified by seeking a sparse approximation of  $R$  using least-squares [Li09].

**3. Quantum simulations via Schrödingerization.** From the proceeding section, we observe that the TDSE under the three types of ABCs can be expressed in the following general form:

$$(16) \quad \frac{d}{dt}|\phi(t)\rangle = -iH_0|\phi(t)\rangle - H_1|\phi(t)\rangle.$$

Here  $|\phi(t)\rangle$  is a quantum state that encodes the wave function, and possibly the auxiliary functions, in the subdomain.  $H_0$  and  $H_1$  are both Hermitian, and  $H_0$  is a slight modification of the original Hamiltonian.

The Schrödingerization approach was first proposed in [JLY23b]. The key step is the warped phase transformation  $\mathbf{v}(t, p) = e^{-p}\psi(t)$ , defined for  $p > 0$  and symmetrically extending the initial data to  $p < 0$ , where  $\psi(t) := |\phi(t)\rangle$ . This transformation converts (16) into a system of linear convection equations,

$$(17) \quad \begin{cases} \frac{d}{dt}\mathbf{v}(t, p) = -iH_0\mathbf{v} + H_1\partial_p\mathbf{v}, \\ \mathbf{v}(0, p) = e^{-|p|}\psi_0. \end{cases}$$

For numerical implementation, it is natural and convenient to introduce  $\alpha = \alpha(p)$  in the initial data of (17) for  $p < 0$  as follows:

$$(18) \quad \begin{cases} \frac{d}{dt}\mathbf{v}(t, p) = A\mathbf{v}(t, p) = -H_1\partial_p\mathbf{v} + IH_2\mathbf{v}, \\ \mathbf{v}(0, p) = e^{-\alpha|p|}\psi_0. \end{cases}$$

To match the exact solution,  $\alpha(p) = 1$  is necessary for the region  $p > 0$ . In the  $p > 0$ -domain, we will truncate the domain at  $p = R$ , where  $R$  is sufficiently large such that  $e^{-R} \approx 0$ . We will choose a large  $\alpha$  for  $p < 0$  so the solution (see Figure 3) will have a support within a relatively small domain. Since the wave  $\tilde{\mathbf{v}}_j$  moves to the left, we choose an artificial boundary at  $p = L < 0$ , for  $|L|$  large enough such that  $\tilde{\mathbf{v}}_j$ , initially almost compact at  $[L_0, R]$ , will not reach the point  $p = L$  during the duration of the computation. This will allow us to use a periodic boundary condition in  $p$  in the spectral approximation.

We now discuss how to restore the solution  $\psi(t)$ . This can be restored by a simple integration,

$$(19) \quad \psi(t) = \int_0^\infty \mathbf{v}(t, p) dp.$$

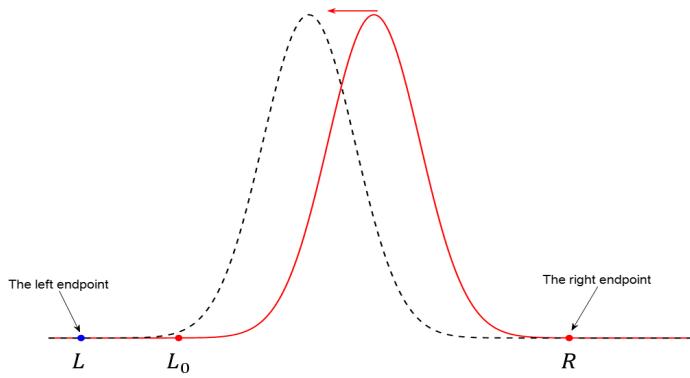


FIG. 3. Schematic diagram for the computational domain of  $p$ .

A more direct strategy, in light of the relation  $\psi(t) = e^p \mathbf{v}(t, p)$  for all  $p > 0$ , is to simply choose any  $p_{k_*} > 0$  and let

$$(20) \quad \psi(t) = e^{p_{k_*} t} \mathbf{v}(t, p_{k_*}).$$

A more intuitive view is to discretize the  $p$  dimension and concatenate the corresponding function for each  $p$ . Toward this end, we choose uniform mesh size  $\Delta p = (R - L)/N_p$  for the auxiliary variable with  $N_p$  being an even number, with the grid points denoted by  $a = p_0 < p_1 < \dots < p_{N_p} = b$ . To compute  $\mathbf{v}(t, p)$ , let the vector  $\mathbf{w}$  be the collection of the function  $\mathbf{v}$  at these grid points, defined more precisely as,

$$\mathbf{w} = [\mathbf{w}_1; \mathbf{w}_2; \dots; \mathbf{w}_n],$$

with “;” indicating the straightening of  $\{\mathbf{w}_i\}_{i \geq 1}$  into a column vector. This can also be expressed as a superposition state using  $|k\rangle$  as a new basis,

$$\mathbf{w}_i = \sum_k \mathbf{v}_i(t, p_k) |k\rangle,$$

where  $\mathbf{v}_i$  is the  $i$ th entry of  $\mathbf{v}$ .

By applying the discrete Fourier transformation in the  $p$  direction, one arrives at

$$(21) \quad \frac{d}{dt} \mathbf{w}(t) = -i(H_0 \otimes I)\mathbf{w} + i(H_1 \otimes P_\mu)\mathbf{w}.$$

Here,  $P_\mu$  is the matrix expression of the momentum operator  $-i\partial_p$ , given by

$$P_\mu = \Phi D_\mu \Phi^{-1}, \quad D_\mu = \text{diag}(\mu_{-N}, \dots, \mu_{N-1}),$$

where  $\mu_l = 2\pi l/(b - a)$  are the Fourier modes, and

$$\Phi = (\phi_{jl})_{M \times M} = (\phi_l(x_j))_{M \times M}, \quad \phi_l(x) = e^{i\mu_l(x-a)}$$

is the matrix representation of the discrete Fourier transform. At this point, we have successfully mapped the dynamics back to a Hamiltonian system. By a change of variables  $\tilde{\mathbf{w}} = (I_u \otimes \Phi^{-1})\mathbf{w}$ , one has

$$(22) \quad \frac{d}{dt} \tilde{\mathbf{w}}(t) = -i(H_0 \otimes I)\tilde{\mathbf{w}} + i(H_1 \otimes D_\mu)\tilde{\mathbf{w}}.$$

This is more amenable to an approximation by a quantum algorithm. In particular, if  $H_0$  and  $H_1$  are sparse, then (22) is a Schrödinger equation with the Hamiltonian  $H = H_0 \otimes I - H_1 \otimes D_\mu$  that inherits the sparsity.

With the state vector encoding  $\tilde{\mathbf{w}}$ , one can first apply the quantum Fourier transform on  $p$  to get back to  $\mathbf{w}$  in (21), which can be written as the tensor-product form

$$\mathbf{w} = \sum_{i,k} w_{ik} |i\rangle |k\rangle, \quad w_{ik} \equiv \mathbf{w}_i(p_k),$$

and then restore the solution  $\psi$  via the integral in (19) or the simple relation in (20). Here we outline some additional detailed steps:

- For the integral in (19), we introduce the index function  $\chi(p)$  satisfying  $\chi(p) = 0$  for  $p < 0$  and  $\chi(p) = 1$  for  $p \geq 0$  and get the  $i$ th entry of  $\psi$  by the following numerical integration:

$$\begin{aligned} \psi_i &= \int_0^\infty \mathbf{v}_i(p) dp \approx \int_L^R \chi(p) \mathbf{v}_i(p) dp \\ &\approx c_0 \chi(p_0) \mathbf{v}_i(p_0) + \dots + c_{M-1} \chi(p_{M-1}) \mathbf{v}_i(p_{M-1}) \\ &=: \tilde{c}_0 \mathbf{v}_i(p_0) + \dots + \tilde{c}_{M-1} \mathbf{v}_i(p_{M-1}), \end{aligned}$$

where  $c_0, \dots, c_{M-1}$  are the weights for numerical integration, and  $\tilde{c}_j = c_j \chi(p_j)$ . This reduces to the case considered in [JL22, JLY23a], where details are presented on how to retrieve the observables induced by the numerical integration.

- If the direct formula (20) is used, one can apply quantum measurement to the  $p$ -register in the computational basis. The state vector is then collapsed to

$$|\psi_*\rangle \equiv \left( \sum_i w_{ik_*} |i\rangle \right) \otimes |k_*\rangle$$

for some  $k_*$ . Since  $|L| \ll R$  is valid in the implementation, we get  $|\psi_*\rangle$  satisfying  $p_{k_*} > 0$  with high probability. Noting that all  $w_{ik_*}$  share the same multiplicative factor  $e^{-p_{k_*}}$  for such an index  $k_*$ , we have actually obtained the desired state  $|\psi\rangle$  on the first register. This approach does not require extra cost to retrieve the quantum state with amplitudes proportional to  $\psi$ .

Costa, Jordan, and Ostrander [CJO19] devised a method to transform the acoustic wave equation into a form of the Schrödinger equation, thereby facilitating the use of quantum algorithms for simulating wave propagation. This approach incorporates spatial discretization and accounts for both Dirichlet and Neumann physical boundary conditions at the boundaries of a finite domain. But it is important to recognize that the Schrödingerization procedure is applicable to general dynamics problems. In addition, the absorbing boundary conditions presented here are designed to simulate dynamics on unbounded domains.

**3.1. Schrödingerization for the CAP method.** As an example, we consider the CAP method (3) applied to the TDSE (1) in two dimensions ( $\mathbf{x} = (x, y)$ ), although the extension to higher dimensions is straightforward. As an example, the five-point finite difference is used to discretize the Laplacian on the grid points,

$$x_0 < x_1 < \dots < x_N < x_{N+1}, \quad y_0 < y_1 < \dots < y_N < y_{N+1}.$$

After a spatial discretization of (3), the semidiscrete system for the unknowns  $\psi(x_i, y_j, t)$  for  $1 \leq i, j \leq N$  has the following form:

$$(23) \quad \begin{cases} \frac{d}{dt} \psi(t) = -iH_0 \psi(t) - H_1 \psi(t), \\ \psi(0) = \psi_0, \end{cases}$$

where  $H_0$  is a real symmetric matrix corresponding to the discretization of  $\hat{H}$ , together with the real parts of the absorbing potential  $W(x, y)$ , and  $H_1 = -\text{diag}(\text{Im}\mathbf{W})$  is a diagonal matrix given by the imaginary part  $W$  of the complex potential, where  $\mathbf{W} = \sum_{ij} W(x_i, y_j) |i, j\rangle$ . In fact, let  $D_{xx}$  be the one-dimensional difference matrices for the second-order derivatives ( $\partial_{xx}$ ) under the homogeneous boundary conditions. Then

$$(24) \quad H_0 = -\frac{1}{2}(D_{xx} \otimes I + I \otimes D_{xx}) + \mathbf{V} + \text{diag}(\text{Re}\mathbf{W}), \quad \mathbf{V} = \text{diag}(V(x_i, y_j)),$$

where we have assumed the same partitions along the  $x$  and  $y$  directions.

**3.2. Schrödingerization for the PML method.** The PML in (7) involves both first- and second-order derivatives. By applying the central differences to these derivatives, we obtain a straightforward discretization,

$$\begin{cases} \partial_t \psi = -iH_V \psi - (I \otimes \boldsymbol{\sigma}_x + \boldsymbol{\sigma}_y \otimes I) \psi + \frac{1}{2}(I \otimes D_x) \boldsymbol{\chi} + \frac{1}{2}(D_y \otimes I) \boldsymbol{\phi}, \\ \partial_t \boldsymbol{\chi} = -i(I \otimes (\boldsymbol{\sigma}_x D_x)) \psi - (I \otimes \boldsymbol{\sigma}_y) \boldsymbol{\chi}, \\ \partial_t \boldsymbol{\phi} = -i((\boldsymbol{\sigma}_y D_y) \otimes I) \psi - (\boldsymbol{\sigma}_y \otimes I) \boldsymbol{\phi}, \end{cases}$$

where  $D_x$  and  $D_{xx}$  are the one-dimensional difference matrices for finite-difference approximations of the first- and second-order derivatives ( $\partial_x$  and  $\partial_{xx}$ ) under homogeneous boundary conditions,  $\boldsymbol{\sigma}_x = \text{diag}(\sigma(x_1), \dots, \sigma(x_N))$  is a diagonal matrix, and

$$(25) \quad H_V = -\frac{1}{2}(D_{xx} \otimes I + I \otimes D_{xx}) + \mathbf{V}, \quad \mathbf{V} = \text{diag}(V(x_i, y_j)).$$

Let  $\mathbf{u}_h(t) = [\psi; \chi; \phi]$ . Then the above system can be written in matrix form,

$$\frac{d}{dt} \mathbf{u}_h = L_h \mathbf{u}_h,$$

where

$$(26) \quad L_h = \begin{bmatrix} -iH_V - (I \otimes \boldsymbol{\sigma}_x + \boldsymbol{\sigma}_y \otimes I) & \frac{1}{2}(I \otimes D_x) & \frac{1}{2}(D_y \otimes I) \\ -i(I \otimes (\boldsymbol{\sigma}_x D_x)) & -(I \otimes \boldsymbol{\sigma}_x) & O \\ -i((\boldsymbol{\sigma}_y D_y) \otimes I) & O & -(\boldsymbol{\sigma}_y \otimes I) \end{bmatrix}.$$

In accordance with (23), one has  $H_0 = -(L_h - L_h^\dagger)/(2i)$  and  $H_1 = -(L_h + L_h^\dagger)/2$ .

**3.3. Schrödingerization for the DtN approach.** Before we apply the Schrödingerization procedure for the DtN method (14), we first introduce a change of variables by defining

$$(27) \quad \psi_\Gamma(t) = (H_{\Gamma,\Pi} H_{\Pi,\Gamma})^{-1/2} \phi_\Gamma(t).$$

The invertibility of the matrix can be seen from the linear independence of the rows of the matrix  $H_{\Gamma,\Pi}$ .

Direct substitutions into (14) yield

$$(28) \quad \begin{cases} \frac{d}{dt} \psi_1(t) = -iH_{\Pi,\Gamma} \psi_1(t) - i\Sigma_{\Gamma,\Gamma} \psi_\Gamma(t), \\ \frac{d}{dt} \psi_\Gamma(t) = -i\Sigma_{\Gamma,\Gamma}^T \psi_1(t) + \Sigma_{\Gamma,\Gamma} \psi_\Gamma(t). \end{cases}$$

Here we have defined matrices

$$(29) \quad \begin{aligned} S &= H_{\Gamma,\Pi} H_{\Pi,\Gamma}, \\ \Sigma_{\Gamma,\Gamma} &= E^T S^{1/2}, \\ \Sigma_{\Gamma,\Gamma} &= s_0 I + iS^{1/2} R^{-1} S^{1/2}. \end{aligned}$$

This notation is motivated by the notions of self-energy in electron transport [BMO<sup>+</sup>02].

More explicit expressions for these matrices can be obtained using the formula (12):

$$\begin{aligned} R &= -H_{\Gamma,\Pi} (H_{\Pi,\Pi} - is_0 I)^{-1} H_{\Pi,\Gamma} \\ &= -H_{\Gamma,\Pi} (H_{\Pi,\Pi}^2 + s_0^2 I)^{-1} (H_{\Pi,\Pi} + is_0 I) H_{\Pi,\Gamma} \\ &= -G_0 - is_0 G_1, \end{aligned}$$

Here, to arrive at the second line, we have used the trivial fact that  $H_{\Pi,\Pi}$  commutes with  $I$ . In the third line, we have defined the matrices

$$(30) \quad G_0 = H_{\Gamma,\Pi} (H_{\Pi,\Pi}^2 + s_0^2 I)^{-1} H_{\Pi,\Pi} H_{\Pi,\Gamma}, \quad G_1 = H_{\Gamma,\Pi} (H_{\Pi,\Pi}^2 + s_0^2 I)^{-1} H_{\Pi,\Gamma}.$$

To separate out the Hermitian and skew-Hermitian parts of  $\Sigma_{\Gamma,\Gamma}$ , we notice that

$$\frac{1}{2} \left( R^{-1} + (R^{-1})^\dagger \right) = \frac{1}{2} R^{-1} (R^\dagger + R) (R^{-1})^\dagger = -R^{-1} G_0 (R^{-1})^\dagger.$$

Similarly, we have

$$\frac{1}{2i} \left( R^{-1} - (R^{-1})^\dagger \right) = \frac{1}{2i} R^{-1} (R^\dagger - R) (R^{-1})^\dagger = -R^{-1} G_1 (R^{-1})^\dagger.$$

Now we can apply the Schrödingerization procedure with the Hermitian and skew-Hermitian parts given by

$$(31) \quad H_0 = \begin{bmatrix} H_{\text{I},\text{I}} & \Sigma_{\text{I},\Gamma} \\ \Sigma_{\text{I},\Gamma}^T & -S^{\frac{1}{2}} R^{-1} G_0 (R^{-1})^\dagger S^{\frac{1}{2}} \end{bmatrix}, \quad H_1 = \begin{bmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{0} & -s_0 I + S^{\frac{1}{2}} R^{-1} G_1 (R^{-1})^\dagger S^{\frac{1}{2}} \end{bmatrix}.$$

**4. Numerical results.** We now present results from some numerical experiments. As a concrete example, we consider the TDSE in two dimensions and choose a square domain  $\Omega := [-3, 3] \times [-3, 3]$ . Following the example in [Zhe07], we choose the initial condition as follows:

$$\psi_0(x, y) = \begin{cases} 1 + \cos(\pi r) + i(\cos(2\pi r) - 1), & r \leq 1, \\ 0 & \text{otherwise.} \end{cases}$$

Here  $r = \sqrt{x^2 + y^2}$ . For the potential, we choose  $V(x, y) = \sin(2\pi r)$  in the unit disc and extend it to zero outside.

In the numerical experiments, we solve (22) using the following matrix exponential:

$$\tilde{\mathbf{w}}(t) = e^{-iHt} \tilde{\mathbf{w}}(0).$$

This is implemented on a desktop computer using the MATLAB built-in function `expm.m`. In the implementation, we also take  $N_x = N_p = 64$  and  $p \in [L, R] = [-5, 5]$ . It is important, however, to note that the purpose of these experiments is to demonstrate that (22) in the Schrödingerization captures the dynamics under the ABCs. A quantum implementation of (22) is much more feasible than execution on a traditional computer, due to the lack of dependence on the dimension.

We first examine the CAP method. In these experiments, the imaginary potential  $W(x, y)$  is chosen as the sum of  $w_1(x, y)$  and  $w_2(x, y)$ , with

$$w_1(x, y) = \begin{cases} -10i(|x| - 2.2)^2, & |x| > 2.2, \\ 0 & \text{otherwise,} \end{cases} \quad w_2(x, y) = \begin{cases} -10i(|y| - 2.2)^2, & |y| > 2.2, \\ 0 & \text{otherwise,} \end{cases}$$

as shown in Figure 2. The results from quantum simulations of (3) and (22) at  $t = 0.3, 0.6$  and  $t = 0.9$  are displayed in Figure 4(d)–(f), from which we observe results that are very similar to those in Figure 4(a)–(c) for the original form (3). This verifies the correctness of the protocol of the Schrödingerization approach.

Now we turn to the PML method. We implemented (7) on the same test problem, and the results are depicted in Figure 5. Here the function  $\sigma$  is chosen as

$$\sigma(x) = \begin{cases} 10(x - 2.2)^2, & x > 2.2, \\ 10(-2.2 - x)^2, & x < -2.2. \end{cases}$$

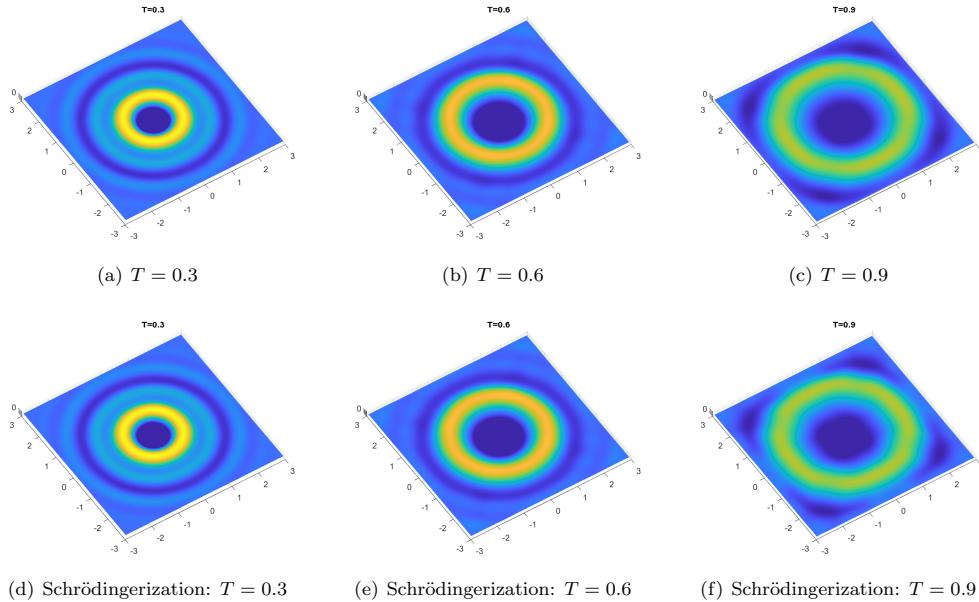


FIG. 4. Snapshots of the real parts of  $\psi$  computed using the CAP method. Top: a direct implementation of (3). Bottom: implementation of the Schrödingerization form (22).

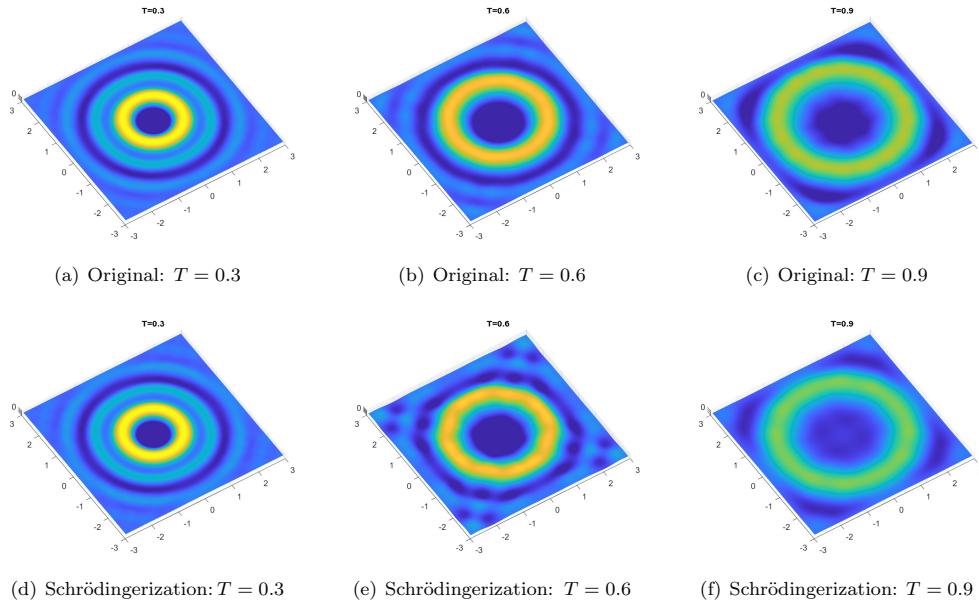


FIG. 5. Snapshots of the real parts of  $\psi$  computed using the PML method. (a)–(c): the original form. (d)–(f): the Schrödingerization form.

To implement the DtN approach, one can solve a discrete boundary element equation [WL20] to determine the matrix  $R$  in (12), which then determines the self-energy in (29). Since our purpose is to test the Schrödingerization approach, we compute  $R$  from (12) by choosing a relatively large domain  $D \supset \Omega$  and simply set  $\Omega_{\text{II}} = D \setminus \Omega_{\text{I}}$ . For the test example, it suffices to choose  $D = [-6, 6]^2$ .

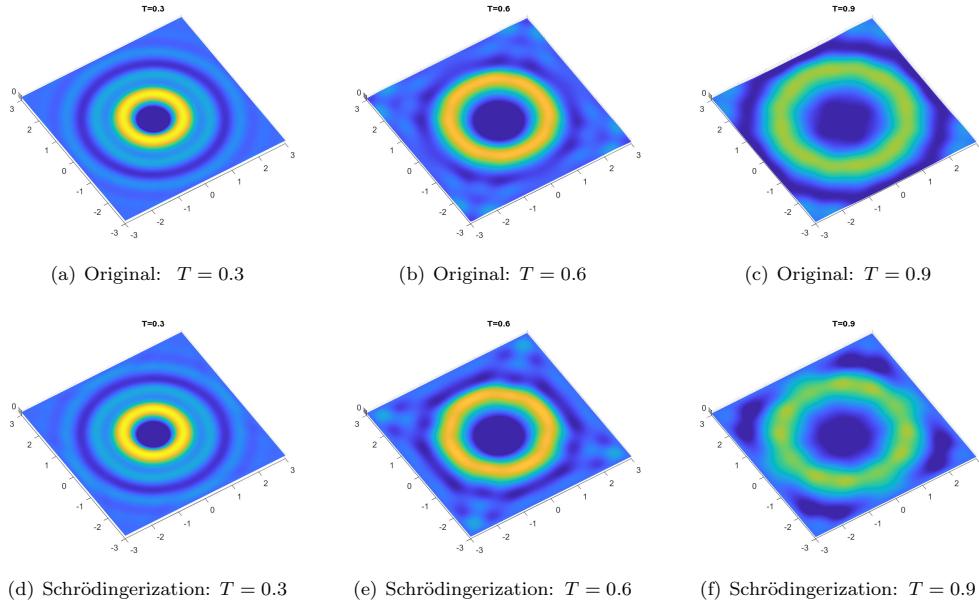


FIG. 6. Snapshots of the real parts of  $\psi$  computed using the DtN method. (a)–(c): the original form. (d)–(f): the Schrödingerization form.

Since all the matrices ( $H_{\text{I,I}}$ ,  $H_{\text{I,II}}$ , and  $H_{\text{II,II}}$ ) can be directly extracted from the Hamiltonian matrix on  $D$ , we can compute the matrices in (29) by direct matrix inversion and multiplications. In (15), we take  $s_0 = 1$ . The direct simulation with the DtN ABC can be done by solving the ODEs (28). The results are displayed in the top row of Figure 6. Meanwhile, a quantum algorithm would solve (22) with the two matrices  $H_0$  and  $H_1$  from (31). We observe that the classical algorithm is able to maintain the propagating pattern of the wavefront without boundary reflections. The solution based on the Schrödingerization has a similar performance. The slight difference can be attributed to the discretization in the  $p$  space.

**5. The implementation complexity of the quantum algorithms.** Once the quantum dynamics with ABCs is turned into a Hamiltonian system (22), one can apply a Hamiltonian simulation algorithm to produce the wave function  $|\psi(T)\rangle$ . Due to the possible time-dependence potential  $V(\mathbf{x}, t)$ , a Hamiltonian simulation algorithm for time-dependent Hamiltonian  $H(t)$  should be considered. To assess the algorithm complexity associated with the implementation of the ABCs, we use the recent results by Berry et al. [BCS<sup>+</sup>20], Theorem 10], although the other algorithms can also be used for the assessment. Here we simply highlight the query complexity.

**THEOREM 5.1.** *The TDSE (1) with an  $s$ -sparse Hamiltonian  $H(t)$  can be simulated from  $t = 0$  to  $t = T$  within error  $\epsilon$  with query complexity*

$$(32) \quad \mathcal{O}\left(s\|H\|_{\max,1} \frac{\log(s\|H\|_{\max,1}/\epsilon)}{\log\log(\|H\|_{\max,1}/\epsilon)}\right).$$

Here the norm is defined as

$$(33) \quad \|H\|_{\max,1} = \int_0^T \|H\|_{\max}(t) dt, \quad \|H\|_{\max}(t) := \max_{i,j} |H_{i,j}(t)|.$$

We now discuss the complexity of implementing the time-dependent Schrödinger equation (22) obtained from the Schrödingerization of the three types of ABCs. Since the discretization error for (17) is  $\mathcal{O}(\Delta p)$  (since  $e^{-|p|}$  is only continuous and not continuously differentiable in  $p$ ) and the discretization in real space has error  $\mathcal{O}(\Delta x^2)$ , we set

$$(34) \quad \Delta p = \mathcal{O}(\Delta x^2)$$

to make the discretization error comparable. We observe that the presence of a singularity  $\exp(-|p|)$  has a direct impact on the accuracy. In fact, a spectral method applied to the  $p$  space will only have a first-order accuracy, which will further affect the overall complexity, as shown below. Meanwhile, if the initial condition does not involve a kink, e.g., Gaussian profile, the method will still hold up to small error. For instance, as outlined in [JL23], the  $\exp(-|p|)$  profile can be replaced by a Gaussian profile, and the fidelity between the two cases is very close to 1. The interested reader is referred to the complete complexity analysis and error propagation found in [JL23, section IV and Appendix A].

*The CAP method.* For the CAP method, in light of (24), we obtain directly that

$$\begin{aligned} \|H_0(t)\|_{\max} &\leq \frac{2}{\Delta x^2} + \max_{i,j} |V(x_i, y_j, t)| + \max_{i,j} |\text{Re}W(x_i, y_j)|, \\ \|H_1\|_{\max} &\leq \max_{i,j} |\text{Im}W(x_i, y_j)|. \end{aligned}$$

Thus we can conclude that

$$\|H\|_{\max} = \|H_0 \otimes I - H_1 \otimes D_\mu\|_{\max} \leq \frac{2}{\Delta x^2} + \|V\|_{\max,1} + \frac{\|W\|_{\max}}{\Delta x^2},$$

where we have used  $\|D_\mu\| = \mathcal{O}(1/\Delta p)$  together with (34). Meanwhile, the sparsity of  $H$  is  $s(H) = \mathcal{O}(1)$  (for  $d$  dimensions, it is  $\mathcal{O}(d)$ ). Combining these estimates gives the total complexity, as shown in the first row of Table 1.

*The PML method.* We examine the magnitude of the Hamiltonian in (22) from (26) and (25),

$$\begin{aligned} \|H_0(t)\|_{\max} &= \left\| \frac{1}{2} (L_h - L_h^\dagger) \right\|_{\max} \\ &\leq \|H_V\|_{\max} + (\|\sigma_x\|_{\max} + \|\sigma_y\|_{\max}) (\|D_x\|_{\max} + \|D_y\|_{\max}) \\ &\leq \frac{1}{\Delta x^2} + \max_{i,j} |V(x_i, y_j, t)| + 4 \max_i |\sigma(x_i)| \frac{1}{\Delta x}. \end{aligned}$$

TABLE 1

*Summary of the complexity with implementing the time-dependent Schrödinger equation (22) obtained from the Schrödingerization of the three types of ABCs. The  $\tilde{\mathcal{O}}$  notation rules out logarithmic factors.*

Artificial Boundary Condition	Query Complexity
Complex Absorbing Potential (CAP)	$\tilde{\mathcal{O}}\left(\frac{T}{\Delta x^2} + \ V\ _{\max,1} + \frac{T\ W\ _{\max}}{\Delta x^2}\right)$
Perfectly Matched Layers (PML)	$\tilde{\mathcal{O}}\left(\frac{T}{\Delta x^2} + \ V\ _{\max,1} + \frac{T\ \sigma\ _{\max}}{\Delta x^3}\right)$
Dirichlet-to-Neumann Map (DtN)	$\tilde{\mathcal{O}}\left(s_\Sigma\left(\frac{T}{\Delta x^2} + \ V\ _{\max,1}\right)\right)$

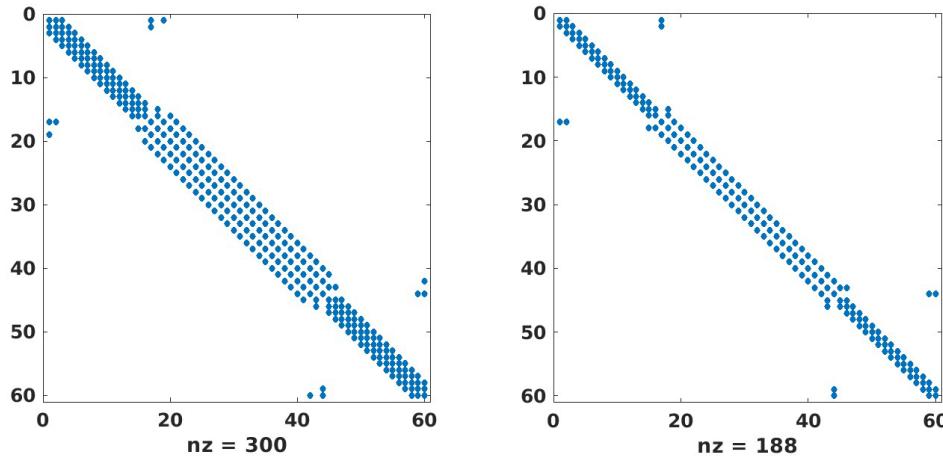


FIG. 7. The effective sparsity of  $\Sigma_{\Gamma,\Gamma}$ . Left: entries satisfying  $|\Sigma_{ij}| > 0.01\|\Sigma_{\Gamma,\Gamma}\|_{\max}$  (one-percent cut). Right: entries satisfying  $|b_{ij}| > 0.02\|\Sigma_{\Gamma,\Gamma}\|_{\max}$  (two-percent cut).  $\|\Sigma_{\Gamma,\Gamma}\|_{\max} := \max_{i,j} |\Sigma_{ij}|$ .

Similarly, we have

$$\|H_1\|_{\max} = \left\| \frac{1}{2}(L_h + L_h^\dagger) \right\|_{\max} \leq (\|\sigma_x\| + \|\sigma_y\|)(\|D_x\| + \|D_y\|) \leq 4 \max_i |\sigma(x_i)| \frac{1}{\Delta x}.$$

These two estimates give

$$\|H\|_{\max} \lesssim \frac{1}{\Delta x^2} + \|V\|_{\max} + \|\sigma\|_{\max} \frac{1}{\Delta x \Delta p}.$$

Finally, one can also deduce from (26) and (25) that the sparsity  $s(H) = \mathcal{O}(1)$ . The total complexity is shown in the second row of Table 1.

*The DtN method.* The query complexity with the DtN method is provided in the third row of Table 1. To prove this estimate, we follow (29) and (31). First, we notice that  $S = H_{\Gamma,\Pi}H_{\Pi,\Gamma}$  in (29) is sparse and the elements are of the order  $\frac{1}{\Delta x^4}$ . Therefore,  $\|\Sigma_{\Gamma,\Gamma}\|_{\max} = \mathcal{O}(\frac{1}{\Delta x^2})$ . Following the same observation, we deduce that  $\|R\|_{\max} = \mathcal{O}(\frac{1}{\Delta x^2})$ . Using the formulas (30) that we derived in the previous section, we find that  $\|G_0\|_{\max} = \mathcal{O}(\frac{1}{\Delta x^2})$ , but  $\|G_0\|_{\max} = \mathcal{O}(1)$ .

Inserting these estimates into (31), we have

$$\|H(t)\|_{\max} = \mathcal{O}\left(\frac{1}{\Delta x^2} + \|V\|_{\max} + \frac{1}{\Delta p}\right).$$

The other factor that contributes to the complexity is the sparsity of the matrices in (29). Numerical observations of these matrices suggest that they are close to identity matrices. For example, for the matrix  $\Sigma_{\Gamma,\Gamma}$  in (29), we plot its entries in Figure 7 by removing entries that are much smaller than its maximum norm. We let  $s_\Sigma$  be such an effective sparsity, and the overall complexity is linear in  $s_\Sigma$ .

**6. Conclusion.** In this article, we use the Schrödingerization approach recently introduced in [JLY22, JLY23b] for quantum dynamics with artificial boundary conditions (ABCs). While a quantum dynamics with ABCs is no longer a Hermitian Hamiltonian system, which is most natural for quantum simulation, the Schrödingerization

approach makes it so in a simple fashion. We give detailed implementation and estimate the computational complexities for three representative ABCs, including the complex absorbing potential method, perfectly matched layers method, and Dirichlet-to-Neumann approach. Numerical experiments validate our approach, demonstrating that the Schrödingerized systems yield the same results as the original dynamics with ABCs.

Our approach provides a simple quantum simulation algorithm for quantum dynamics in unbounded domains. As pointed out in [JLY22, JLY23b], the Schrödingerization approach is not only applicable to quantum dynamics with ABCs but also to general linear partial differential equations with physical boundary conditions. It can also be applied to such equations with interface conditions. These will be the subjects of our future research.

## REFERENCES

- [AAB<sup>+</sup>08] X. ANTOINE, A. ARNOLD, C. BESSE, M. EHRHARDT, AND A. SCHÄDLE, *A review of transparent and artificial boundary conditions techniques for linear and nonlinear Schrödinger equations*, Commun. Comput. Phys., 4 (2008), pp. 729–796.
- [AFL21] D. AN, DI. FANG, AND L. LIN, *Time-dependent unbounded Hamiltonian simulation with vector norm scaling*, Quantum, 5 (2021), 459.
- [AFL22] D. AN, DI. FANG, AND L. LIN, *Time-dependent Hamiltonian simulation of highly oscillatory dynamics and superconvergence for Schrödinger equation*, Quantum, 6 (2022), 690.
- [AL22] D. AN AND L. LIN, *Quantum linear system solver based on time-optimal adiabatic quantum computing and quantum approximate optimization algorithm*, ACM Trans. Quantum Comput., 3 (2022), pp. 1–28.
- [ALL23] D. AN, J.-P. LIU, AND L. LIN, *Linear combination of Hamiltonian simulation for non-unitary dynamics with optimal state preparation cost*, Phys. Rev. Lett., 131 (2023), 150603.
- [ALWZ22] D. AN, J.-P. LIU, D. WANG, AND Q. ZHAO, *A Theory of Quantum Differential Equation Solvers: Limitations and Fast-Forwarding*, preprint, arXiv:2211.05246, 2022.
- [Arn98] A. ARNOLD, *Numerically absorbing boundary conditions for quantum evolution equations*, VLSI Des., 6 (1998), pp. 313–319.
- [BC22] D. W. BERRY AND P. COSTA, *Quantum Algorithm for Time-Dependent Differential Equations Using Dyson Series*, preprint, arXiv:2212.03544, 2022.
- [BCC<sup>+</sup>15] D. W. BERRY, A. M. CHILDS, R. CLEVE, R. KOTHARI, AND R. D. SOMMA, *Simulating Hamiltonian dynamics with a truncated Taylor series*, Phys. Rev. Lett., 114 (2015), 090502.
- [BCC<sup>+</sup>17] D. W. BERRY, A. M. CHILDS, R. CLEVE, R. KOTHARI, AND R. D. SOMMA, *Exponential improvement in precision for simulating sparse Hamiltonians*, Forum Math. Sigma, 5 (2017), e8.
- [BCK15] D. W. BERRY, A. M. CHILDS, AND R. KOTHARI, *Hamiltonian simulation with nearly optimal dependence on all parameters*, in Proceedings of the 2015 IEEE 56th Annual Symposium on Foundations of Computer Science (FOCS 2015), Berkeley, CA, 2015, pp. 792–809.
- [BCOW17] D. W. BERRY, A. M. CHILDS, A. OSTRANDER, AND G. WANG, *Quantum algorithm for linear differential equations with exponentially improved dependence on precision*, Comm. Math. Phys., 356 (2017), pp. 1057–1081.
- [BCS<sup>+</sup>20] D. W. BERRY, A. M. CHILDS, Y. SU, X. WANG, AND N. WIEBE, *Time-dependent Hamiltonian simulation with  $L^1$ -norm scaling*, Quantum, 4 (2020), 254.
- [Ber94] J.-P. BERENGER, *A perfectly matched layer for the absorption of electromagnetic waves*, J. Comput. Phys., 114 (1994), pp. 185–200.
- [BMO<sup>+</sup>02] M. BRANDBYGE, J.-L. MOZOS, P. ORDEJÓN, J. TAYLOR, AND K. STOKBRO, *Density-functional method for nonequilibrium electron transport*, Phys. Rev. B, 65 (2002), 165401.
- [CAS<sup>+</sup>22] P. C. S. COSTA, D. AN, Y. R. SANDERS, Y. SU, R. BABBUSH, AND D. W. BERRY, *Optimal scaling quantum linear-systems solver via discrete adiabatic theorem*, PRX Quantum, 3 (2022), 040303.

- [CGJ19] S. CHAKRABORTY, A. GILYÉN, AND S. JEFFERY, *The power of block-encoded matrix powers: Improved regression techniques via faster Hamiltonian simulation*, in Proceedings of the 46th International Colloquium on Automata, Languages, and Programming (ICALP 2019), 132, 2019, pp. 33:1–33:14.
- [CJL23] Y. CAO, S. JIN, AND N. LIU, *Quantum Simulation for Time-Dependent Hamiltonians - with Applications to Non-Autonomous Ordinary and Partial Differential Equations*, arXiv:2312.02817, 2023.
- [CJO19] P. C. S. COSTA, S. JORDAN, AND A. OSTRANDER, *Quantum algorithm for simulating the wave equation*, Phys. Rev. A, 99 (2019), 012323.
- [CKS17] A. M. CHILDS, R. KOTHARI, AND R. D. SOMMA, *Quantum algorithm for systems of linear equations with exponentially improved dependence on precision*, SIAM J. Comput., 46 (2017), pp. 1920–1950, <https://doi.org/10.1137/16M1087072>.
- [CL20] A. M. CHILDS AND J.-P. LIU, *Quantum spectral methods for differential equations*, Comm. Math. Phys., 375 (2020), pp. 1427–1457.
- [EM79] B. ENGQUIST AND A. MAJDA, *Radiation boundary conditions for acoustic and elastic wave calculations*, Comm. Pure Appl. Math., 32 (1979), pp. 313–357.
- [FLT23] F. DI, L. LIN, AND Y. TONG, *Time-marching based quantum solvers for time-dependent linear differential equations*, Quantum, 7 (2023), 955.
- [GSLW19] A. GILYÉN, Y. SU, G. H. LOW, AND N. WIEBE, *Quantum singular value transformation and beyond: Exponential improvements for quantum matrix arithmetics*, in Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing, 2019, pp. 193–204.
- [HH04] H. HAN AND Z. HUANG, *Exact artificial boundary conditions for the Schrödinger equation in  $R^2$* , Commun. Math. Sci., 2 (2004), pp. 79–94.
- [HHL09] A. W. HARROW, A. HASSIDIM, AND S. LLOYD, *Quantum algorithm for solving linear systems of equations*, Phys. Rev. Lett., 103 (2009) 150502.
- [Hid19] J. D. HIDARY, *Quantum Computing: An Applied Approach*, Springer, 2019.
- [HRB07] F. HE, C. RUIZ, AND A. BECKER, *Absorbing boundaries in numerical solutions of the time-dependent Schrödinger equation on a grid using exterior complex scaling*, Phys. Rev. A, 75 (2007), 053407.
- [JL22] S. JIN AND N. LIU, *Quantum algorithms for computing observables of nonlinear partial differential equations*, Bulletin des Math. Sci., (2024), to appear.
- [JL23] S. JIN AND N. LIU, *Analog quantum simulation of partial differential equations*, Quantum Sci. Tech., accepted, <https://iopscience.iop.org/article/10.1088/2058-9565/ad49cf>.
- [JLY22] S. JIN, N. LIU, AND Y. YU, *Quantum Simulation of Partial Differential Equations via Schrödingerisation*, preprint, arXiv:2212.13969, 2022.
- [JLY23a] S. JIN, N. LIU, AND Y. YU, *Time complexity analysis of quantum algorithms via linear representations for nonlinear ordinary and partial differential equations*, J. Comput. Phys., 487 (2023), 112149.
- [JLY23b] S. JIN, N. LIU, AND Y. YU, *Quantum simulation of partial differential equations: Applications and detailed analysis*, Phys. Rev. A, 108 (2023), 032603.
- [Kro23] H. KROVI, *Improved quantum algorithms for linear and nonlinear differential equations*, Quantum, 7 (2023), 913.
- [KSB19] M. KIEFEROVÁ, A. SCHERER, AND D. W. BERRY, *Simulating the dynamics of time-dependent Hamiltonians with a truncated Dyson series*, Phys. Rev. A, 99 (2019), 042314.
- [LC17] G. H. LOW AND I. L. CHUANG, *Optimal Hamiltonian simulation by quantum signal processing*, Phys. Rev. Lett., 118 (2017), 010501.
- [LC19] G. H. LOW AND I. L. CHUANG, *Hamiltonian simulation by qubitization*, Quantum, 3 (2019), 163.
- [Li09] X. LI, *Efficient boundary conditions for molecular statics models of solids*, Phys. Rev. B, 80 (2009), 104112.
- [Li12] X. LI, *An atomistic-based boundary element method for the reduction of molecular statics models*, Comput. Methods Appl. Mech. Engrg., 225 (2012), pp. 1–13.
- [MPNE04] J. G. MUGA, J. P. PALAO, B. NAVARRO, AND I. L. EGUSQUIZA, *Complex absorbing potentials*, Phys. Rep., 395 (2004), pp. 357–426.
- [MR09] P.-G. MARTINSSON AND G. J. RODIN, *Boundary algebraic equations for lattice problems*, Proc. Roy. Soc. A Math. Phys. Eng. Sci., 465 (2009), pp. 2489–2503.
- [NC02] M. A. NIELSEN AND I. CHUANG, *Quantum Computation and Quantum Information*, American Association of Physics Teachers, 2002.

- [NK11] A. NISSEN AND G. KREISS, *An optimized perfectly matched layer for the Schrödinger equation*, Comm. Comput. Phys., 9 (2011), pp. 147–179.
- [Pre18] J. PRESKILL, *Quantum computing in the NISQ era and beyond*, Quantum, 2 (2018), 79.
- [RM96] U. V. RISS AND H.-D. MEYER, *Investigation on the reflection and transmission properties of complex absorbing potentials*, J. Chem. Phys., 105 (1996), pp. 1409–1419.
- [RP11] E. G. RIEFFEL AND W. H. POLAK, *Quantum Computing: A Gentle Introduction*, MIT Press, 2011.
- [SE00] E. Y. SIDKY AND B. D. ESRY, *Boundary-free propagation with the time-dependent Schrödinger equation*, Phys. Rev. Lett., 85 (2000), 5086.
- [WL20] X. WU AND X. LI, *Absorbing boundary conditions for the time-dependent Schrödinger-type equations in  $R^3$* , Phys. Rev. E, 101 (2020), 013304.
- [YE18] Y. YU AND B. D. ESRY, *An optimized absorbing potential for ultrafast, strong-field problems*, J. Phys. B Atomic Mol. Opt. Phys., 51 (2018), 095601.
- [Zhe07] C. ZHENG, *A perfectly matched layer approach to the nonlinear Schrödinger wave equations*, J. Comput. Phys., 227 (2007), pp. 537–556.