

Quantum circuits for partial differential equations via Schrödingerisation

Junpeng Hu¹, Shi Jin^{1,2,4}, Nana Liu^{2,3,4}, and Lei Zhang^{1,2,4}

¹School of Mathematical Sciences, Shanghai Jiao Tong University, Shanghai, 200240, China

²Institute of Natural Sciences, Shanghai Jiao Tong University, Shanghai, 200240, China

³University of Michigan-Shanghai Jiao Tong University Joint Institute, Shanghai, 200240, China

⁴MOE-LSC, Shanghai Jiao Tong University, Shanghai, 200240, China

Quantum computing has emerged as a promising avenue for achieving significant speedup, particularly in large-scale PDE simulations, compared to classical computing. One of the main quantum approaches involves utilizing Hamiltonian simulation, which is directly applicable only to Schrödinger-type equations. To address this limitation, Schrödingerisation techniques have been developed, employing the warped transformation to convert general linear PDEs into Schrödinger-type equations. However, despite the development of Schrödingerisation techniques, the explicit implementation of the corresponding quantum circuit for solving general PDEs remains to be designed. In this paper, we present a detailed implementation of a quantum algorithm for general PDEs using Schrödingerisation techniques. We provide examples of the heat equation, and the advection equation approximated by the upwind scheme, to demonstrate the effectiveness of our approach. Complexity analysis is also carried out to demonstrate the quantum advantages of these algorithms in high dimensions over their classical counter-

parts. Several numerical experiments demonstrate the validity of these proposed quantum circuits.

1 Introduction

Partial differential equations (PDEs) models are essential tools to investigate dynamic behaviors of many physical systems, arise in applications such as reservoir modeling, atmospheric and ocean circulation, and high-frequency scattering. Simulating complex dynamical systems within feasible computational time is of utmost importance. Despite the significant progress achieved through the utilization of high-performance computing clusters, the computational overhead remains a major bottleneck, particularly for large systems, high dimensional and multi-scale problems.

In recent years, quantum computing [1, 2] has emerged as a promising avenue for achieving significantly faster computation compared to classical computing. Despite the current limitations in hardware scalability and noise resistance, there has been remarkable progress in developing quantum algorithms for scientific and engineering computing, which will be useful when quantum computers become available one day, or can also help to guide the design of analog quantum computing [3]. Among many potential applications, one particularly promising area is the utilization of quantum computers as solvers for both quan-

Junpeng Hu: hjp3268@sjtu.edu.cn

Shi Jin: shijin-m@sjtu.edu.cn

Nana Liu: nana.liu@quantumlah.org

Lei Zhang: lzhang2012@sjtu.edu.cn

tum and classical partial differential equations (PDEs).

Lloyd’s groundbreaking work [4] introduced the first explicit quantum simulation algorithm, which allowed for the simulation of Hamiltonians containing local interaction terms. Building upon this, Aharonov and Ta-Shma [5] proposed an efficient simulation algorithm capable of handling a broader class of sparse Hamiltonians. Subsequent research endeavors [6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16] have further refined and improved these simulation techniques. In their work [14], Berry et al. combined the quantum walk approach [9, 10] with the fractional-query approach [13]. This innovative fusion resulted in a significant reduction in query complexity, achieving an algorithmic complexity of $O\left(\tau \frac{\log(\tau/\varepsilon)}{\log \log(\tau/\varepsilon)}\right)$, where $\tau := s |H|_{\max} t$ and ε is the desired precision. Notably, this approach exhibited a near-linear dependence on both the sparsity s and the evolution time t , while offering exponential speedup over ε . Moreover, Berry et al. established a lower bound, demonstrating the near-optimality of this result with respect to scaling in either τ or ε independently.

Achieving quantum speedup for ordinary or partial differential equations (ODEs/PDEs) that are not of the Schrödinger equation type is of significant importance in scientific and engineering applications. A commonly employed strategy is to discretize the spatial and temporal domains, thereby transforming linear partial differential equations (PDEs) into a system of linear algebraic equations. Subsequently, quantum algorithms are utilized to solve this transformed system [17, 18, 19]. The application of quantum algorithms in solving PDEs has the potential to yield polynomial or super-polynomial speedups [20, 21, 22]. Jin et al. conducted a comprehensive study on the time complexity of quantum difference methods for linear and nonlinear high-dimensional and multiscale PDEs within the framework of Asymptotic-Preserving schemes [23, 24, 25]. In their work [26, 27], they proposed a recent approach known as “Schrödingerisation”

for solving general linear PDEs. This approach involves converting linear PDEs into a system of Schrödinger equations in one higher dimension, using a simple transformation called the warped phase transformation. “Schrödingerisation” enables quantum simulations for general linear ordinary differential equations (ODEs) and PDEs [28], and iterative linear algebra solvers [29]. An alternative approach is presented in [30].

While significant progress has been made in the field, the explicit design of quantum circuits for solving partial differential equations (PDEs) remains open. A recent study by Sato et al. [31] has introduced a novel approach that utilizes the Bell basis to construct scalable quantum circuits specifically for wave or Schrödinger-type PDEs. In our paper, we construct quantum circuits for general PDEs—which not necessarily follow unitary dynamics—by using the Schrödingerisation technique.

Contribution We present a detailed circuit implementation of the approximating unitary matrix V , as depicted in Figure 5. This explicit implementation has not been reported in previous literature. To demonstrate the effectiveness of our approach, we provide two examples: the heat equation, and the advection equation with the upwind scheme which was not studied in [31]. Complexity analysis is performed to demonstrate the quantum advantages of these algorithms in high dimensions over their classical counterparts. Numerical experiments for both the heat equation and the advection equation were performed using the *Qiskit* package [32]. The results, obtained either through statevectors or observations from the quantum circuits, closely align with those derived from classical computational methods.

Organization The paper is structured as follows: In Section 2, we introduce the basic notations. The quantum circuit implementation for the heat equation is introduced in Section 3, followed by the implementation of the upwind scheme for the advection equation

in Section 4. To showcase the scalability of the proposed quantum circuits, we conduct a complexity analysis in Section 5. Several numerical experiments are presented to demonstrate the validity of these proposed quantum circuits in Section 6. Finally, we present our conclusions in Section 7.

Notation In the context of complexity analysis, we use the big O notation to describe the upper bound, and use $\tilde{O}$ to denote O where logarithmic terms are ignored. The symbols X, Y, Z represent the Pauli matrices.

2 Representation of finite difference operator

We begin by considering the one-dimensional case in order to establish the definition of the finite difference operator and its representation in quantum states.

Given a one-dimensional domain $\Omega := [0, L]$, which is uniformly subdivided into $N_x = 2^{n_x}$ intervals of length $h = \frac{L}{N_x}$, one can express a discrete function u with u_i defined on $x_i = ih$, for $i = 0, \dots, N_x - 1$, as a quantum state $|\mathbf{u}\rangle = \mathbf{u} / \|\mathbf{u}\|$, $\mathbf{u} := \sum_{j=0}^{2^{n_x}-1} u_j |j\rangle$.

The shift operators can be defined as follows:

$$\begin{aligned} S^- &:= \sum_{j=1}^{2^{n_x}-1} |j-1\rangle\langle j| \\ &= \sum_{j=1}^{n_x} I^{\otimes(n_x-j)} \otimes \sigma_{01} \otimes \sigma_{10}^{\otimes(j-1)} \\ &\triangleq \sum_{j=1}^{n_x} s_j^-, \\ S^+ &:= (S^-)^\dagger = \sum_{j=1}^{2^{n_x}-1} |j\rangle\langle j-1| \\ &= \sum_{j=1}^{n_x} I^{\otimes(n_x-j)} \otimes \sigma_{10} \otimes \sigma_{01}^{\otimes(j-1)} \\ &\triangleq \sum_{j=1}^{n_x} s_j^+. \end{aligned} \tag{2.1}$$

with basic 2×2 matrices

$$\begin{aligned} \sigma_{01} &:= |0\rangle\langle 1|, & \sigma_{10} &:= |1\rangle\langle 0|, \\ \sigma_{00} &:= |0\rangle\langle 0|, & \sigma_{11} &:= |1\rangle\langle 1|. \end{aligned} \tag{2.2}$$

The forward and backward finite difference operators are,

$$(D^+ \mathbf{u})_j = \frac{u_{j+1} - u_j}{h}, \quad (D^- \mathbf{u})_j = \frac{u_j - u_{j-1}}{h}, \tag{2.3}$$

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

Thus, the forward (D_P^+), backward (D_P^-), central ($D_P^\pm$) finite difference operators with respect to the first order derivative, and the central finite difference operator (D_P^Δ) with respect to the second order derivative, combined with the periodic boundary conditions, can be expressed as follows,

$$\begin{aligned} D_P^+ &= \frac{S^- - I^{\otimes n_x} + \sigma_{10}^{\otimes n_x}}{h}, \\ D_P^- &= \frac{I^{\otimes n_x} - S^+ - \sigma_{01}^{\otimes n_x}}{h}, \\ D_P^\pm &= \frac{S^- - S^+ - \sigma_{01}^{\otimes n_x} + \sigma_{10}^{\otimes n_x}}{2h}, \\ D_P^\Delta &= \frac{S^- + S^+ - 2I^{\otimes n_x} + \sigma_{01}^{\otimes n_x} + \sigma_{10}^{\otimes n_x}}{h^2}. \end{aligned} \tag{2.4}$$

Similarly, the difference operator with Dirichlet boundary conditions for the left or right side can be expressed as

$$\begin{aligned} D_D^+ &= \frac{S^- - I^{\otimes n_x}}{h}, & (\text{right side}), \\ D_D^- &= \frac{I^{\otimes n_x} - S^+}{h}, & (\text{left side}). \end{aligned} \tag{2.5}$$

Considering the Dirichlet boundary conditions for both sides, i.e., $u_0 = u_{N_x} = 0$, it suffices to solve the numerical solution $\mathbf{u} := [u_1; \dots, u_{N_x-1}]$. In this case, we choose $N_x = 2^{n_x} + 1$ and define $|\mathbf{u}\rangle = \mathbf{u} / \|\mathbf{u}\|$, $\mathbf{u} := \sum_{j=0}^{2^{n_x}-1} u_{j+1} |j\rangle$. Therefore, the difference operators with Dirichlet boundary conditions for

both sides are

$$\begin{aligned} D_D^+ &= \frac{S^- - I^{\otimes n_x}}{h}, \\ D_D^- &= \frac{I^{\otimes n_x} - S^+}{h}, \\ D_D^\pm &= \frac{S^- - S^+}{2h}, \\ D_D^\Delta &= \frac{S^- + S^+ - 2I^{\otimes n_x}}{h^2}. \end{aligned} \quad (2.6)$$

We will employ the following techniques described in [31], which will be utilized multiple times throughout the paper.

Lemma 1. [31] *Given an operator of the form $S = |a\rangle\langle b| + |b\rangle\langle a|$ with $|a\rangle, |b\rangle \in \mathbb{C}^{2^n}$ and $\langle a|b\rangle = 0$, it can be decomposed into*

$$\begin{aligned} S &= \frac{|a\rangle + |b\rangle}{\sqrt{2}} \frac{\langle a| + \langle b|}{\sqrt{2}} - \frac{|a\rangle - |b\rangle}{\sqrt{2}} \frac{\langle a| - \langle b|}{\sqrt{2}} \\ &\triangleq |a'\rangle\langle a'| - |b'\rangle\langle b'|, \end{aligned} \quad (2.7)$$

where $\langle a'|b'\rangle = 0$. Then, there exists a unitary matrix B such that the following relation holds true,

$$B|0\rangle|1\rangle^{\otimes(n-1)} = |a'\rangle, \quad B|1\rangle^{\otimes n} = |b'\rangle, \quad (2.8)$$

and S can be written as

$$S = B \left(Z \otimes |1\rangle\langle 1|^{\otimes(n-1)} \right) B^\dagger. \quad (2.9)$$

The time evolution operator $\exp(iSt)$ can be implemented as

$$\exp(iSt) = B \cdot CRZ_n^{1,\dots,n-1}(-2t) \cdot B^\dagger, \quad (2.10)$$

where $CRZ_n^{1,\dots,n-1}(\theta)$ is the multi-controlled RZ gate ($e^{-i\theta Z/2}$) acting on the n -th qubit controlled by $1, \dots, n-1$ -th qubits.

Remark 1. By rotating $|b\rangle$ with the phase $e^{-i\lambda}$, similar results hold for $S = e^{i\lambda}|a\rangle\langle b| + e^{-i\lambda}|b\rangle\langle a|$. In particular, when $|a\rangle$ and $|b\rangle$ are basis states, we can explicitly construct the unitary matrix B using H gates (Hadamard gates), Phase gates, Pauli gates, and CNOT gates, such as in Lemma 2, Equation (4.15), Equation (4.16).

Remark 2. As reformulated in Equation (2.9), it follows that $\|S\| = 1$ immediately.

We introduce a useful Lemma to facilitate the implementation of the differential operator's time evolution. The proof can be found in Appendix A.1.

Lemma 2. [31] *The time evolution operator formulated as*

$$\exp\left(i\gamma\tau(e^{i\lambda}s_j^- + e^{-i\lambda}s_j^+)\right) \quad (2.11)$$

can be implemented explicitly by

$$I^{\otimes(n_x-j)} \otimes W_j(\gamma\tau, \lambda), \quad (2.12)$$

with

$$\begin{aligned} W_j(\gamma\tau, \lambda) &:= B_j(\lambda) CRZ_j^{1,\dots,j-1}(-2\gamma\tau) B_j(\lambda)^\dagger, \\ B_j(\lambda) &:= \left(\prod_{m=1}^{j-1} CNOT_m^j \right) P_j(-\lambda) H_j, \end{aligned} \quad (2.13)$$

where H_j is the Hadamard gate acting on the j -th qubit, $P_j(\lambda)$ is the Phase gate acting on the j -th qubit as

$$P_j(\lambda) := \begin{bmatrix} 1 & 0 \\ 0 & e^{i\lambda} \end{bmatrix}, \quad (2.14)$$

$CNOT_m^j$ is the CNOT gate acting on the m -th qubit controlled by the j -th qubit.

3 The heat equation

We begin by considering a scalar field u governed by the d -dimensional heat equation.

$$\begin{cases} \partial_t u(t, x) = a\Delta u(t, x), & x \in \Omega \subset \mathbb{R}^d, \\ u(0, x) = f(x), \end{cases} \quad (3.1)$$

with the Dirichlet boundary condition, where $a > 0$ is a constant, the d -dimensional heat equation can be solved using the central difference method in classical simulation as follows:

$$d\mathbf{u}(t)/dt = aD_D^\Delta \mathbf{u}(t), \quad \mathbf{u}(0) = \mathbf{u}_0, \quad (3.2)$$

where $\mathbf{u} := [u_{j_1, \dots, j_d}]_{1 \leq j_i \leq N_x - 1}$.

3.1 Schrödingerisation

The heat equation, although of dissipation type and not directly simulatable using Hamiltonian simulation, can be transformed into an equivalent Schrödinger-type equation using the Schrödingerization technique. This transformation enables the application of quantum algorithms designed for Schrödinger-type equations to solve the heat equation in one-higher dimension.

3.1.1 Continuous formulation

To be more precise, we can apply the warped phase transformation $v(t, x, p) = e^{-p}u(t, x)$, where $p \geq 0$, to the heat equation,

$$\partial_t v(t, x, p) = a\Delta v(t, x, p) = -a\partial_p \Delta v(t, x, p). \quad (3.3)$$

By extending to $p < 0$ with initial data $v(0, x, p) = e^{-|p|}u(0, x)$, and applying $\mathcal{F}_p$, the Fourier transform over p , to Equation (3.3), one obtains

$$\partial_t \hat{v}(t, x, \eta) = i\eta a \Delta \hat{v}(t, x, \eta), \quad (3.4)$$

where $\hat{v}(t, x, \eta) := \mathcal{F}_p v(t, x, p)$. Since the operator $\eta a \Delta$ is self-adjoint under the Dirichlet BC, Equation (3.4) is exactly the Schrödinger equation.

3.1.2 Discretization

The next step involves discretizing the problem. For instance, in the one-dimensional case, the variables v and $\hat{v}$ are discretized as $\mathbf{v}(t) := [v_{j,k}(t)]_{j,k}$ and $\hat{\mathbf{v}}(t) := [\hat{v}_{j,k}(t)]_{j,k}$ with $v_{j,k}(t) := v(t, x_j, p_k)$, $\hat{v}_{j,k}(t) := v(t, x_j, \eta_k)$ respectively. The initial conditions can be represented as follows:

$$\begin{aligned} \mathbf{v}(0) &:= \mathbf{u}(0) \otimes \mathbf{p} \\ \hat{\mathbf{v}}(0) &:= \mathcal{F}_p \mathbf{v}(0) = \mathbf{u}(0) \otimes \eta \\ \mathbf{p} &:= \left[e^{-|p_0|}; e^{-|p_1|}; \dots; e^{-|p_{N_p-1}|} \right], \\ \eta &:= \left[\frac{2}{\eta_0^2 + 1}; \frac{2}{\eta_1^2 + 1}; \dots; \frac{2}{\eta_{N_p-1}^2 + 1} \right], \end{aligned} \quad (3.5)$$

where $-\pi R = p_0 < \dots < p_{N_p} = \pi R$ with mesh size $\Delta p = 2\pi R/N_p$, $N_p = 2^{n_p}$ and

the Fourier variable $\eta_k = (k - \frac{N_p}{2})/R$, $k = 0, 1, \dots, N_p - 1$. The Hamiltonian $\eta a \partial_{xx}$ is discretized using the central difference method as follows,

$$\begin{aligned} \mathbf{H}_{\text{heat}} &= a D_D^\Delta \otimes D_\eta \\ &= \frac{a}{h^2} (S^- + S^+ - 2I^{\otimes n_x}) \otimes \text{diag}(\eta_0, \dots, \eta_{N_p-1}) \\ &= \sum_{k=0}^{N_p-1} \left(k - \frac{N_p}{2} \right) \mathbf{H}_0 \otimes |k\rangle\langle k|, \end{aligned} \quad (3.6)$$

with

$$\mathbf{H}_0 := \gamma_0 \left[\sum_{j=1}^{n_x} (s_j^- + s_j^+) - 2I^{\otimes n_x} \right], \quad \gamma_0 = \frac{a}{h^2 R}. \quad (3.7)$$

In the d -dimensional case, one has

$$\begin{aligned} \mathbf{H}_{\text{heat}} &= a \sum_{\alpha=1}^d (D_D^\Delta)_\alpha \otimes D_\eta \\ &= \sum_{k=0}^{N_p-1} \left(k - \frac{N_p}{2} \right) \sum_{\alpha=1}^d (\mathbf{H}_0)_\alpha \otimes |k\rangle\langle k|, \end{aligned} \quad (3.8)$$

where $(\bullet)_\alpha := I^{\otimes(d-\alpha)n_x} \otimes \bullet \otimes I^{\otimes(\alpha-1)n_x}$.

3.2 Quantum circuit

We move forward to develop a detailed quantum circuit designed to solve the heat equation. To simplify the process and enhance the comprehension of the construction steps, we will adopt the notations listed below,

$$\begin{aligned} U_0(\tau) &:= \exp(i\mathbf{H}_0\tau), \\ \tilde{U}_0(\tau) &:= \prod_{\alpha=1}^d (U_0(\tau))_\alpha, \\ U_{\text{heat}}(\tau) &:= \exp(i\mathbf{H}_{\text{heat}}\tau). \end{aligned} \quad (3.9)$$

Then we can obtain

$$U_{\text{heat}}(\tau) = \sum_{k=0}^{N_p-1} \tilde{U}_0^{k-N_p/2}(\tau) \otimes |k\rangle\langle k|. \quad (3.10)$$

The proof of this is provided in Appendix A.3.

We then employ $V_0, \tilde{V}_0, V_{\text{heat}}$ as the approximations of $U_0, \tilde{U}_0, U_{\text{heat}}$, respectively. Here,

V_0 is derived by applying the first-order Lie-Trotter-Suzuki decomposition to U_0 , specifically as

$$\begin{aligned} U_0(\tau) &= \exp \left(i\gamma_0\tau \sum_{j=1}^{n_x} (s_j^- + s_j^+) - 2i\gamma_0\tau I^{\otimes n_x} \right) \\ &\approx \exp(-2i\gamma_0\tau) \prod_{j=1}^{n_x} \exp \left(i\gamma_0\tau (s_j^- + s_j^+) \right) \\ &\triangleq V_0(\tau). \end{aligned} \quad (3.11)$$

Recalling Lemma 2 and setting $\gamma = \gamma_0, \lambda = 0$, V_0 can be implemented by

$$V_0(\tau) = \text{Ph}(-2\gamma_0\tau) \prod_{j=1}^{n_x} I^{\otimes(n_x-j)} \otimes W_j(\gamma_0\tau, 0), \quad (3.12)$$

where $\text{Ph}(\theta) := e^{i\theta} I^{\otimes n_x}$ is the global phase gate. Following that, $\tilde{V}_0$ and V_{heat} can be expressed as

$$\begin{aligned} \tilde{V}_0(\tau) &:= \prod_{\alpha=1}^d (V_0(\tau))_{\alpha}, \\ V_{\text{heat}}(\tau) &:= \sum_{k=0}^{N_p-1} \tilde{V}_0^{k-N_p/2}(\tau) \otimes |k\rangle \langle k|, \\ &= \left(\tilde{V}_0^{-N_p/2}(\tau) \otimes I^{\otimes n_p} \right) \sum_{k=0}^{N_p-1} \tilde{V}_0^k(\tau) \otimes |k\rangle \langle k|. \end{aligned} \quad (3.13)$$

To effectively realize V_{heat} , we employ the subsequent method.

Lemma 3. *The unitary operator formulated as*

$$Q = \sum_{k=0}^{N_p-1} Q_0^k \otimes |k\rangle \langle k|, \quad (3.14)$$

can be realized by performing the controlled operator $c - Q_0^{2^m}$ on the first register, controlled by the m -th qubit of the second register.

Remark 3. *In the most favorable scenario, when $Q_0^k(\tau)$ can be implemented with cost independent on k —such as $Q_0^k(\tau) = Q_0(k\tau)$ —this strategy is significantly more efficient compared to implementing $Q(\tau)$ directly, as the efficiency is exponential. Nevertheless, in this study, we find that $\tilde{V}_0^k(\tau) \neq$*

$\tilde{V}_0(k\tau)$, resulting in $O(N_p)$ operations which offer a quadratic speed-up relative to directly executing V_{heat} .

The proof for Lemma 3 can be found in Appendix A.2. Detailed quantum circuit designs for the unitary matrix $W_j(\gamma\tau, \lambda)$, the operators $V_0(\tau)$, $\tilde{V}_0(\tau)$, and $V_{\text{heat}}(\tau)$ are depicted in Figure 1, Figure 2, Figure 3, and Figure 4, respectively.

Given $T = r\tau$, the operator $V_{\text{heat}}(\tau)$ is iterated r times to give the state $|\hat{\mathbf{v}}_D(T)\rangle := V_{\text{heat}}^r(\tau) |\hat{\mathbf{v}}(0)\rangle$. An inverse quantum Fourier transform can then be conducted to retrieve $|\mathbf{v}_D(T)\rangle$. Subsequently, a measurement

$$M_{\geq 0} = \sum_{p_k \geq 0} M_k := \sum_{p_k \geq 0} I^{\otimes n_x d} \otimes |k\rangle \langle k|, \quad (3.15)$$

is taken to select only the $|k\rangle, p_k \geq 0$ part of the state $|\mathbf{v}_D(T)\rangle$. Since $v(t, x, p) = e^{-p}u(t, x)$ for $p \geq 0$, it follows that

$$\begin{aligned} \langle \mathbf{v}_D(T) | M_k | \mathbf{v}_D(T) \rangle &\approx \frac{e^{-2p_k} \|\mathbf{u}(T)\|^2}{\|\mathbf{v}_D(T)\|^2} \\ &= \frac{e^{-2p_k} \|\mathbf{u}(T)\|^2}{\|\mathbf{v}(0)\|^2} = \frac{e^{-2p_k} \|\mathbf{u}(T)\|^2}{\|\mathbf{p}\|^2 \|\mathbf{u}(0)\|^2}. \end{aligned} \quad (3.16)$$

After the measurement, the output is an approximation of $|\mathbf{u}(T)\rangle |k\rangle$ with probability $e^{-2p_k} \|\mathbf{u}(T)\|^2 / \|\mathbf{p}\|^2 \|\mathbf{u}(0)\|^2$. The full circuit to implement the Schrödingerisation method is shown in Figure 5, where $\mathcal{QFT}$ ($\mathcal{IQFT}$) denotes the (inverse) quantum Fourier transform.

4 The advection equation

For the second illustrative example, let us consider the scalar field u governed by the d -dimensional advection equation,

$$\partial_t u(t, x) = \mathbf{a} \cdot \nabla u(t, x) = \sum_{\alpha=1}^d a_{\alpha} \partial_{x_{\alpha}} u(t, x), \quad (4.1)$$

with periodic boundary conditions, where $\mathbf{a}$ is a constant vector field. For simplicity, we assume that $a_{\alpha} > 0$ for $\alpha = 1, \dots, d_0$, and $a_{\alpha} < 0$ for $\alpha = d_0 + 1, \dots, d$. In classical

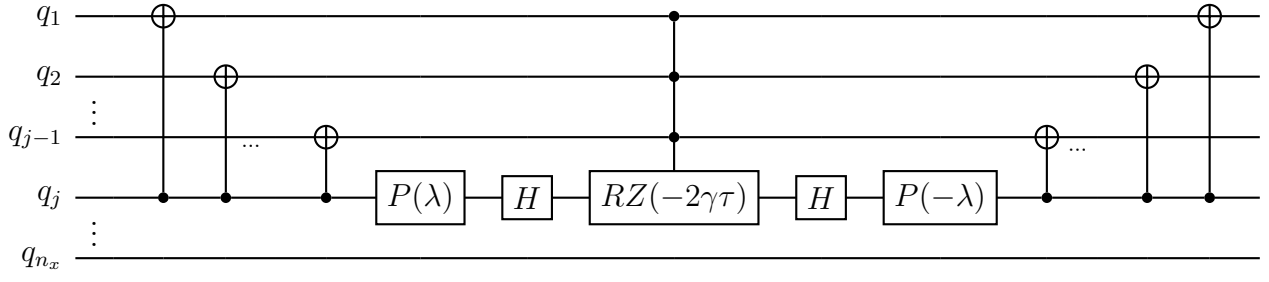


Figure 1: Quantum circuit for $W_j(\gamma\tau, \lambda)$.

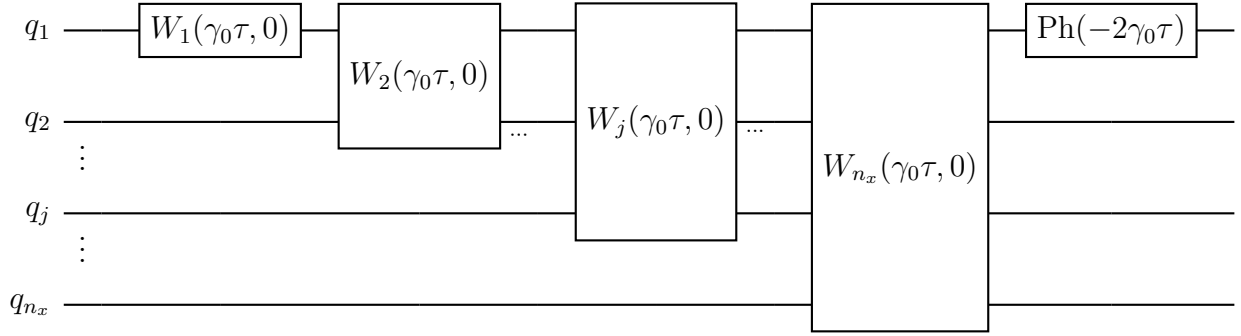


Figure 2: Quantum circuit for $V_0(\tau)$.

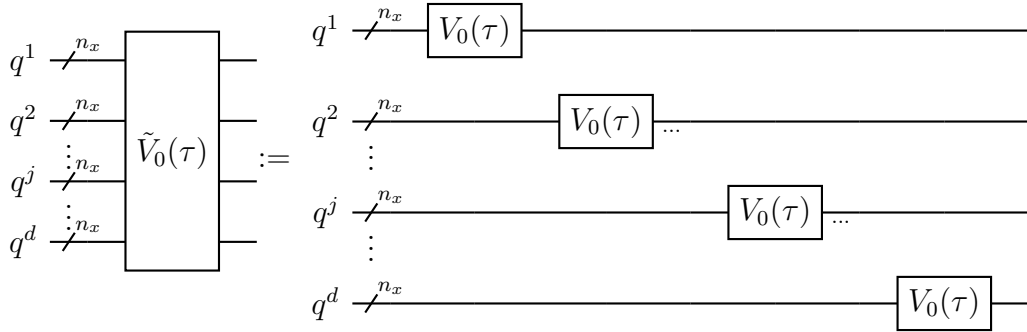


Figure 3: Quantum circuit for $\tilde{V}_0(\tau)$.

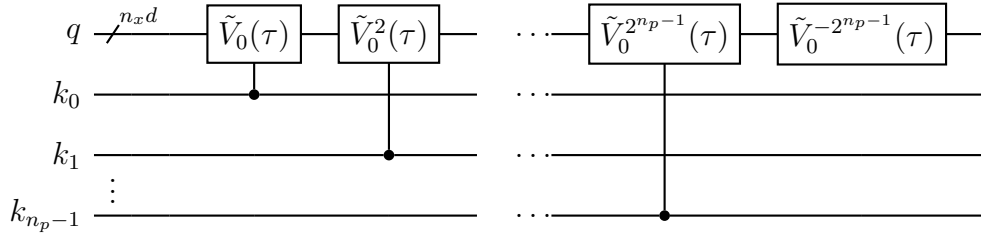


Figure 4: Quantum circuit for $V_{\text{heat}}(\tau)$.

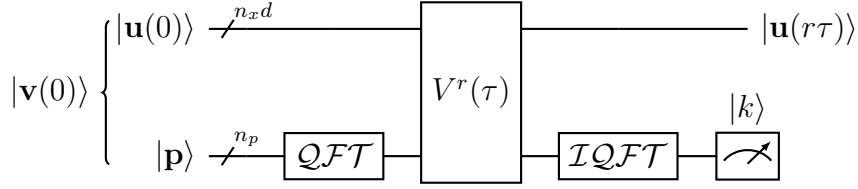


Figure 5: Quantum circuit for the Schrödingerisation method, where the measurement requires only projection onto $|k\rangle$, $p_k > 0$, and QFT ($IQFT$) denotes the (inverse) quantum Fourier transform.

simulation, Equation (4.1) can be solved using the upwind method as follows,

$$\begin{aligned} & d\mathbf{u}(t)/dt \\ &= \left(\sum_{\alpha=1}^{d_0} a_{\alpha} (D_P^+)_{\alpha} + \sum_{\alpha=d_0+1}^d a_{\alpha} (D_P^-)_{\alpha} \right) \mathbf{u}(t) \\ &\triangleq \mathbf{A}\mathbf{u}(t). \end{aligned} \quad (4.2)$$

In [31], the central finite difference method was employed to solve this equation while aiming to preserve its Schrödinger structure. However, it is important to note that the central finite difference method introduces significant dispersion errors, leading to spurious numerical oscillations across discontinuities, which was shown in the numerical examples in [31], thus is not the preferred approach for the advection equation when the solution becomes discontinuous. The center difference is natural for quantum simulation since its discrete spectral is purely imaginary, corresponding to unitary dynamics. The upwind scheme, on the other hand, is dissipative, thus one cannot directly solve it via quantum simulation. Therefore we will utilize the Schrödingerization technique.

4.1 Schrödingerisation

In this case, $-i\mathbf{A}$ is not Hermitian, so we decompose $\mathbf{A}$ into $\mathbf{A} = \mathbf{A}_1 + i\mathbf{A}_2$ with

$$\begin{aligned} \mathbf{A}_1 &= \frac{\mathbf{A} + \mathbf{A}^\dagger}{2}, \quad \mathbf{A}_2 = \frac{\mathbf{A} - \mathbf{A}^\dagger}{2i}, \\ \mathbf{A}_1 &= \mathbf{A}_1^\dagger, \quad \mathbf{A}_2 = \mathbf{A}_2^\dagger. \end{aligned} \quad (4.3)$$

4.1.1 Continuous formulation

Applying the Schrödingerisation method, the warped phase transformation $\mathbf{v}(t, p) = e^{-p}\mathbf{u}(t)$, $p \geq 0$, satisfies

$$\begin{aligned} \partial_t \mathbf{v}(t, p) &= \mathbf{A}_1 \mathbf{v}(t, p) + i\mathbf{A}_2 \mathbf{v}(t, p) \\ &= -\mathbf{A}_1 \partial_p \mathbf{v}(t, p) + i\mathbf{A}_2 \mathbf{v}(t, p), \end{aligned} \quad (4.4)$$

and this equation is extended to $p < 0$ with initial data $\mathbf{v}(0, p) = e^{-|p|}\mathbf{u}(0)$. After performing the Fourier transform with respect to p , one obtains

$$\partial_t \hat{\mathbf{v}}(t, \eta) = i(\eta \mathbf{A}_1 + \mathbf{A}_2) \hat{\mathbf{v}}(t, \eta) \triangleq i\mathbf{H} \hat{\mathbf{v}}(t, \eta), \quad (4.5)$$

with $\mathbf{H}$ being Hermitian.

4.1.2 Discretization

Similarly to the heat equation, we discretize p as $-\pi R = p_0 < \dots < p_{N_p} = \pi R$ with a mesh size of $\Delta p = 2\pi R/N_p$. We also define $\eta_k = (k - \frac{N_p}{2})/R$ for $k = 0, 1, \dots, N_p - 1$. The initial conditions can be represented as,

$$\begin{aligned} \mathbf{v}(0) &:= \mathbf{u}(0) \otimes \mathbf{p}, \\ \hat{\mathbf{v}}(0) &:= \mathcal{F}_p \mathbf{v}(0) = \mathbf{u}(0) \otimes \eta, \end{aligned} \quad (4.6)$$

with $\mathbf{u}(0) := [u_0(0); u_1(0); \dots; u_{N_x-1}(0)]$ and $\mathbf{p}, \eta$ defined as in (3.5). Noting that

$$\begin{aligned} \mathbf{A}_1 &= \sum_{\alpha=1}^d \frac{|a_{\alpha}|h}{2} (D_P^{\Delta})_{\alpha}, \\ \mathbf{A}_2 &= \sum_{\alpha=1}^d \frac{a_{\alpha}}{i} (D_P^{\pm})_{\alpha}. \end{aligned} \quad (4.7)$$

The Hamiltonian $\mathbf{H}_{\text{adv}}$ is discretized as

$$\begin{aligned}\mathbf{H}_{\text{adv}} &= \mathbf{A}_1 \otimes D_\eta + \mathbf{A}_2 \otimes I^{\otimes n_p} \\ &= \sum_{k=0}^{N_p-1} \left(k - \frac{N_p}{2} \right) \sum_{\alpha=1}^d |a_\alpha| (\mathbf{H}_1)_\alpha \otimes |k\rangle \langle k| \\ &\quad + \sum_{\alpha=1}^d a_\alpha (\mathbf{H}_2)_\alpha \otimes I^{\otimes n_p}.\end{aligned}\tag{4.8}$$

with

$$\begin{aligned}\mathbf{H}_1 &= \mathbf{H}_1^{(1)} + \mathbf{H}_1^{(2)}, \quad \gamma_1 = \frac{1}{2hR}, \\ \mathbf{H}_1^{(1)} &:= \gamma_1 (\sigma_{01}^{\otimes n_x} + \sigma_{10}^{\otimes n_x}), \\ \mathbf{H}_1^{(2)} &:= \gamma_1 \sum_{j=1}^{n_x} (s_j^- + s_j^+) - 2\gamma_1 I^{\otimes n_x},\end{aligned}\tag{4.9}$$

$$\begin{aligned}\mathbf{H}_2 &= \mathbf{H}_2^{(1)} + \mathbf{H}_2^{(2)}, \quad \gamma_2 = \frac{1}{2h}, \\ \mathbf{H}_2^{(1)} &:= -i\gamma_2 (-\sigma_{01}^{\otimes n_x} + \sigma_{10}^{\otimes n_x}), \\ \mathbf{H}_2^{(2)} &:= -i\gamma_2 \sum_{j=1}^{n_x} (s_j^- - s_j^+).\end{aligned}\tag{4.10}$$

4.2 Quantum circuit

Similar to Section 3.2, we introduce the following notations to clarify the construction procedure,

$$\begin{aligned}U_1(\tau) &:= \exp(i\mathbf{H}_1\tau), \quad \tilde{U}_1(\tau) := \prod_{\alpha=1}^d (U_1(|a_\alpha|\tau))_\alpha \\ U_2(\tau) &:= \exp(i\mathbf{H}_2\tau), \quad \tilde{U}_2(\tau) := \prod_{\alpha=1}^d (U_2(a_\alpha\tau))_\alpha \\ U_{\text{adv}}(\tau) &:= \exp(i\mathbf{H}_{\text{adv}}\tau).\end{aligned}\tag{4.11}$$

We then obtain the following result, with the proof provided in Appendix A.4.

$$\begin{aligned}U_{\text{adv}}(\tau) &\approx U_*(\tau) := \\ &\left(\tilde{U}_2(\tau) \otimes I^{\otimes n_p} \right) \sum_{k=0}^{N_p-1} \tilde{U}_1^{k-N_p/2}(\tau) \otimes |k\rangle \langle k|,\end{aligned}\tag{4.12}$$

the proof of which is presented in Appendix A.4. Additionally, We denote $U_1^{(1)}, U_1^{(2)}, U_1^{(1)}, U_1^{(2)}$ as the Hamiltonian simulation operators corresponding to

$\mathbf{H}_1^{(1)}, \mathbf{H}_1^{(2)}, \mathbf{H}_2^{(1)}, \mathbf{H}_2^{(2)}$, respectively. The approximations of these operators are denoted as the corresponding ‘V’ operators.

The first-order Lie-Trotter-Suzuki decomposition of U_1 and U_2 gives

$$\begin{aligned}U_1(\tau) &= \exp(i(\mathbf{H}_1^{(1)} + \mathbf{H}_1^{(2)})\tau) \\ &\approx U_1^{(1)}(\tau)U_1^{(2)}(\tau), \\ U_2(\tau) &= \exp(i(\mathbf{H}_2^{(1)} + \mathbf{H}_2^{(2)})\tau) \\ &\approx U_2^{(1)}(\tau)U_2^{(2)}(\tau).\end{aligned}\tag{4.13}$$

Similar to Equation (3.11) and Equation (3.12), $U_1^{(2)}, U_2^{(2)}$ are approximated by

$$\begin{aligned}U_1^{(2)}(\tau) &\approx V_1^{(2)}(\tau) := \\ &\text{Ph}(-2\gamma_1\tau) \prod_{j=1}^{n_x} I^{\otimes(n_x-j)} \otimes W_j(\gamma_1\tau, 0), \\ U_2^{(2)}(\tau) &\approx V_2^{(2)}(\tau) := \\ &\prod_{j=1}^{n_x} I^{\otimes(n_x-j)} \otimes W_j\left(\gamma_2\tau, -\frac{\pi}{2}\right),\end{aligned}\tag{4.14}$$

by taking $\gamma = \gamma_1, \lambda = 0$ and $\gamma = \gamma_2, \lambda = -\pi/2$ in Lemma 2.

The quantum circuits for $U_1^{(1)}(\tau)$ and $U_2^{(1)}(\tau)$ can be explicitly constructed following Lemma 1 as

$$\begin{aligned}U_1^{(1)}(\tau) &= \exp\left(i\gamma_1\tau \left(\sigma_{01}^{\otimes n_x} + \sigma_{10}^{\otimes n_x}\right)\right) \\ &= B^{(1)} \cdot \text{CRZ}_{n_x}^{1, \dots, n_x-1}(-2\gamma_1\tau) \cdot \left(B^{(1)}\right)^\dagger, \\ B^{(1)} &:= \prod_{m=1}^{n_x-1} \text{CNOT}_m^{n_x} H_{n_x} \prod_{m=1}^{n_x-1} X_m \\ &= B_{n_x}(0) \prod_{m=1}^{n_x-1} X_m.\end{aligned}\tag{4.15}$$

$$\begin{aligned}U_2^{(1)}(\tau) &= \exp\left(i\gamma_2\tau \left(i\sigma_{01}^{\otimes n_x} - i\sigma_{10}^{\otimes n_x}\right)\right) \\ &= B^{(2)} \cdot \text{CRZ}_{n_x}^{1, \dots, n_x-1}(-2\gamma_2\tau) \cdot \left(B^{(2)}\right)^\dagger, \\ B^{(2)} &:= \prod_{m=1}^{n_x-1} \text{CNOT}_m^{n_x} P_{n_x} \left(-\frac{\pi}{2}\right) H_{n_x} \prod_{m=1}^{n_x-1} X_m \\ &= B_{n_x} \left(\frac{\pi}{2}\right) \prod_{m=1}^{n_x-1} X_m.\end{aligned}\tag{4.16}$$

Given $U_1^{(1)}, U_2^{(1)}, V_1^{(2)}, V_2^{(2)}$ defined in Equation (4.15), Equation (4.16) and Equation

(4.14), U_1 , U_2 , $\tilde{U}_1$ and $\tilde{U}_2$ are approximated by

$$\begin{aligned} U_1(\tau) &\approx V_1(\tau) := U_1^{(1)}(\tau)V_1^{(2)}(\tau), \\ U_2(\tau) &\approx V_2(\tau) := U_2^{(1)}(\tau)V_2^{(2)}(\tau), \\ \tilde{U}_1(\tau) &\approx \tilde{V}_1(\tau) := \prod_{\alpha=1}^d (V_1(|a_\alpha|\tau))_\alpha, \\ \tilde{U}_2(\tau) &\approx \tilde{V}_2(\tau) := \prod_{\alpha=1}^d (V_2(a_\alpha\tau))_\alpha. \end{aligned} \quad (4.17)$$

Ultimately, $U_{\text{adv}}(\tau)$ is approximated by

$$\begin{aligned} U_{\text{adv}}(\tau) &\approx V_{\text{adv}}(\tau) := \\ &(\tilde{V}_2(\tau) \otimes I^{\otimes n_p}) \sum_{k=0}^{N_p-1} \tilde{V}_1^{k-N_p/2}(\tau) \otimes |k\rangle\langle k|, \end{aligned} \quad (4.18)$$

The efficient implementation technique described in Lemma 3 is also applicable here. The detailed quantum circuits for the approximating unitary matrices $U_1^{(1)}(\tau)$, $U_2^{(1)}(\tau)$, $V_1(\tau)$, $V_2(\tau)$, $\tilde{V}_1(\tau)$, $\tilde{V}_2(\tau)$ and $V_{\text{adv}}(\tau)$ are shown in Figure 6-12. Given $V_{\text{adv}}(\tau)$, the circuit for implementing the Schrödingerisation method remains the same as in Figure 5.

5 Complexity analysis

In this subsection, we estimate the complexity of the quantum circuits constructed in the previous sections and demonstrate their scalability.

Typical quantum algorithms for solving partial differential equations often depend on oracles for general matrices, but the cost of preparing these oracles is often not considered, which can make it challenging to assess their feasibility accurately. In our study, we provide complete algorithms for the heat equation and the advection equation (with constant coefficients) that does not require any oracles, making it more practical. The complexity results we present here are based on basic gates, i.e., single-qubit gates and CNOT gates, ensuring clarity and accuracy.

5.1 The heat equation

Lemma 4. Consider the Schrödinger equation $d|\mathbf{u}(t)\rangle/dt = i\mathbf{H}_{\text{heat}}|\mathbf{u}(t)\rangle$, where the Hamiltonian $\mathbf{H}_{\text{heat}}$ is given by Equation (3.8). The time evolution operator $U_{\text{heat}}(\tau) = \exp(i\mathbf{H}_{\text{heat}}\tau)$ with a time increment τ can be approximated by the unitary $V_{\text{heat}}(\tau)$ in Equation (3.13), and its explicit circuit implementation is depicted in Figure 1-4. Furthermore, the approximation error in terms of the operator norm is upper-bounded by

$$\|U_{\text{heat}}(\tau) - V_{\text{heat}}(\tau)\| \leq \frac{dN_p\gamma_0^2\tau^2(n_x - 1)}{4}, \quad (5.1)$$

and

$$\|U_{\text{heat}}(T) - V_{\text{heat}}^r(\tau)\| \leq \frac{dN_p\gamma_0^2T^2(n_x - 1)}{4r}, \quad (5.2)$$

where $r = T/\tau$, d denotes the spatial dimension, $N_p = 2^{n_p}$ and $N_x = 2^{n_x}$ represent the number of grid points for the variables p and x (per dimension), respectively. Additionally, γ_0 is defined in Equation (3.7).

Proof. According to the theory of the Trotter splitting error with commutator scaling [33] and the analysis in [31], we have

$$\begin{aligned} &\|U_0(\tau) - V_0(\tau)\| \\ &\leq \frac{\gamma_0^2\tau^2}{2} \sum_{j=1}^{n_x} \sum_{j'=j+1}^{n_x} \left\| [(s_j^- + s_j^+), (s_{j'}^- + s_{j'}^+)] \right\| \\ &\leq \frac{\gamma_0^2\tau^2(n_x - 1)}{2}, \end{aligned} \quad (5.3)$$

where we have applied the commutator results presented in Appendix B.1, Equation (B.5). From Equation (B.9), we have

$$\begin{aligned} &\|U_{\text{heat}}(\tau) - V_{\text{heat}}(\tau)\| \\ &\leq \frac{dN_p}{2} \|U_0(\tau) - V_0(\tau)\| \\ &\leq \frac{dN_p\gamma_0^2\tau^2(n_x - 1)}{4}. \end{aligned} \quad (5.4)$$

$$\begin{aligned} &\|U_{\text{heat}}(T) - V_{\text{heat}}^r(\tau)\| \\ &\leq r \|U_{\text{heat}}(\tau) - V_{\text{heat}}(\tau)\| \\ &\leq \frac{dN_p\gamma_0^2T^2(n_x - 1)}{4r}. \end{aligned} \quad (5.5)$$

□

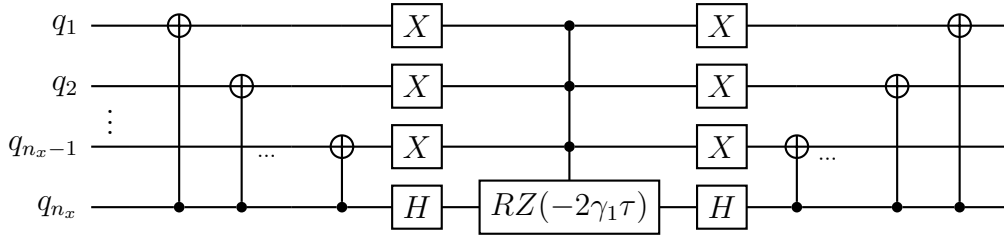


Figure 6: Quantum circuit for $U_1^{(1)}(\tau)$.

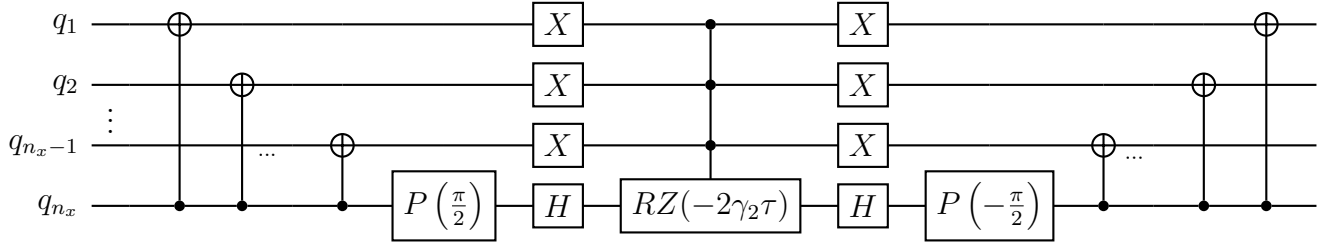


Figure 7: Quantum circuit for $U_2^{(1)}(\tau)$.

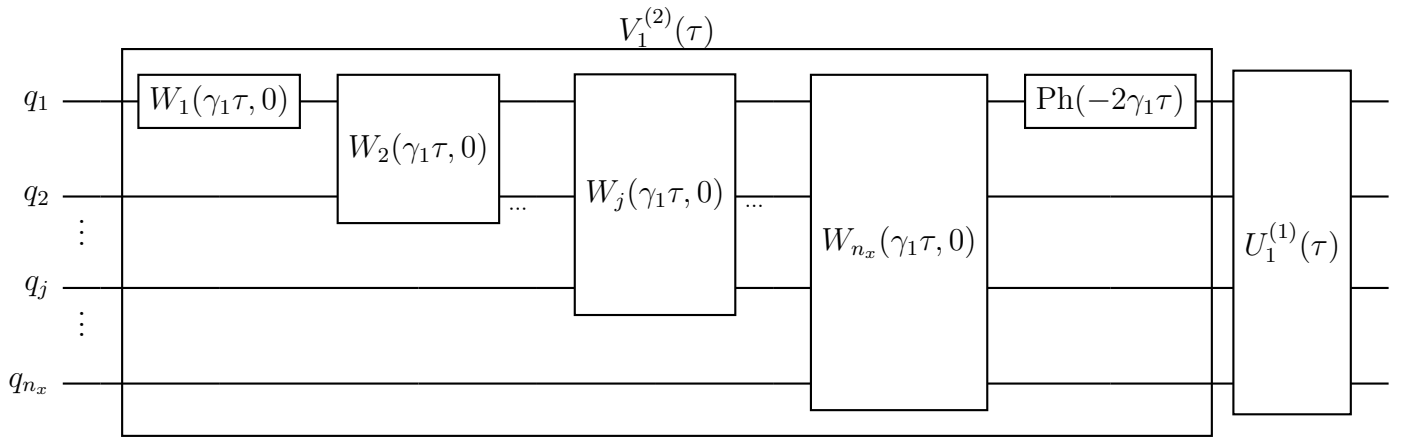


Figure 8: Quantum circuit for $V_1(\tau)$.

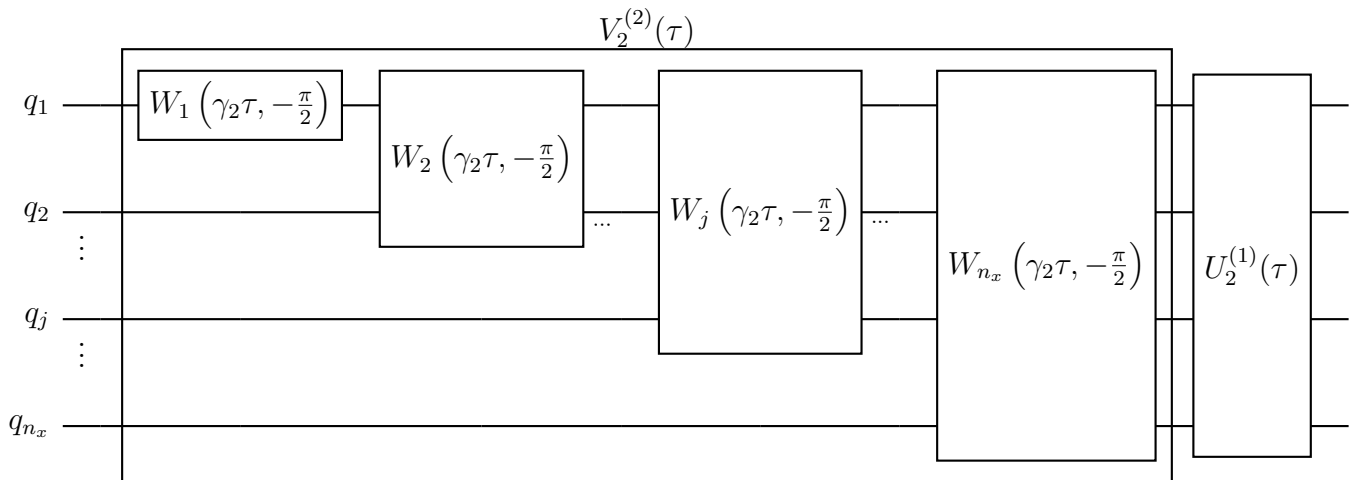


Figure 9: Quantum circuit for $V_2(\tau)$.

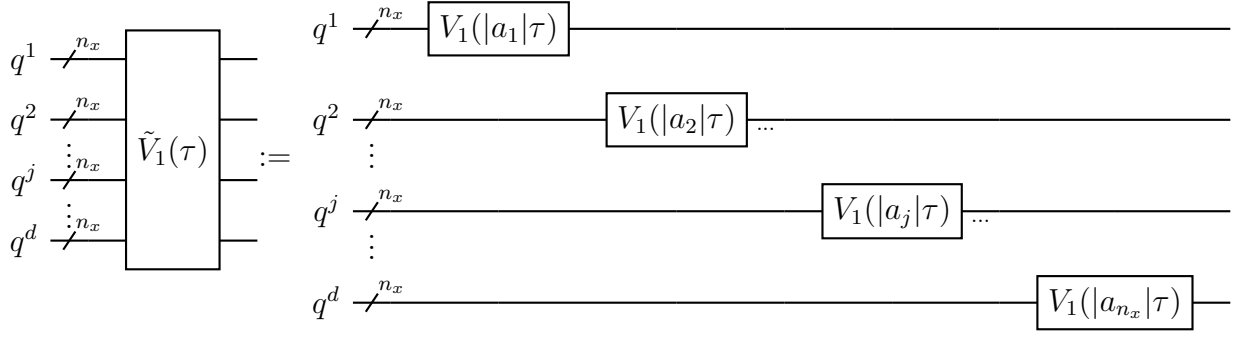


Figure 10: Quantum circuit for $\tilde{V}_1(\tau)$.

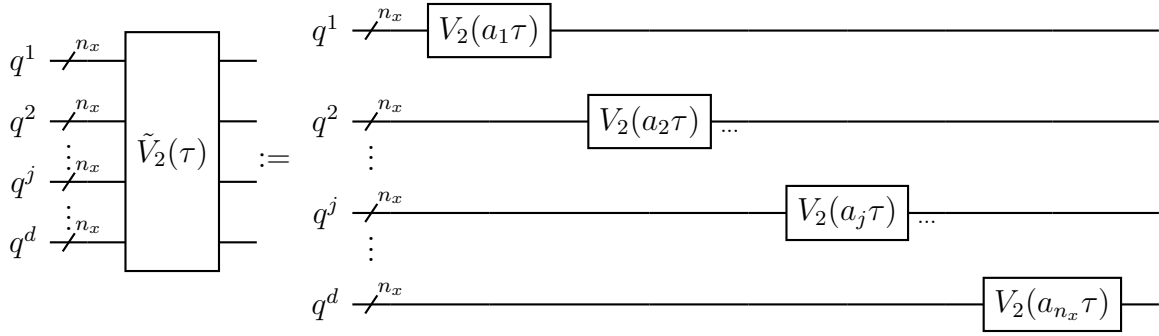


Figure 11: Quantum circuit for $\tilde{V}_2(\tau)$.

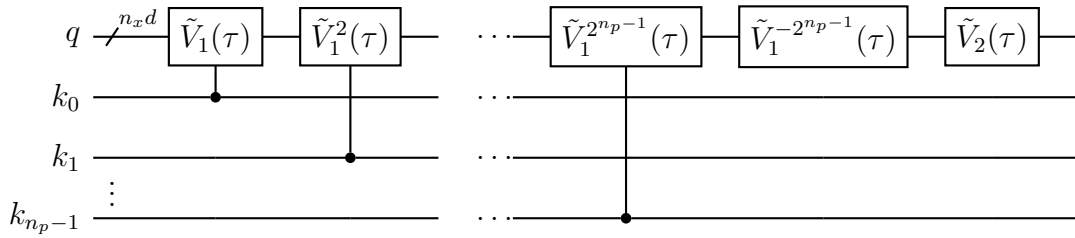


Figure 12: Quantum circuit for $V_{\text{adv}}(\tau)$.

Lemma 5. *The approximated time evolution operator $V_{\text{heat}}(\tau)$ in Equation (3.13) can be implemented using $O(dN_p n_x)$ single-qubit gates and at most $\mathcal{Q}_{V_{\text{heat}}} = O(dN_p n_x^2)$ CNOT gates for $n_x \geq 3$.*

Proof. The implementation of $V_{\text{heat}}(\tau)$ involves a maximum of $2^{n_p-1} \tilde{V}_0(-\tau)$ gates and $\sum_{m=0}^{n_p-1} 2^m = 2^{n_p} - 1$ controlled $\tilde{V}_0(\tau)$ gates. Each $\tilde{V}_0$ gate consists of d V_0 gates. Furthermore, V_0 consists of a phase gate Ph and $W_j, j = 1, \dots, n$. W_j can be decomposed into a multi-controlled RZ gate (a controlled RZ gate for $j = 2$), 2 Hadmard gates and $2(j-1)$ CNOT gates in the case $\lambda = 0$. Hence the number of single-qubit gates is $d2^{n_p-1}(2n_x + 1) = O(dN_p n_x)$. The total number of CNOT gates in the circuit implementation of V_{heat} is determined by

$$\mathcal{Q}_{V_{\text{heat}}} = d2^{n_p-1}\mathcal{Q}_{V_0} + d(2^{n_p} - 1)\mathcal{Q}_{c-V_0}. \quad (5.6)$$

According to the decomposition techniques presented in [31, 34], it is known that a multi-controlled RZ or RX gate with $(j-1)$ control qubits can be decomposed into single-qubit gates and at most $16j-40$ CNOT gates. Hence

$$\begin{aligned} \mathcal{Q}_{V_0} &= 2 + \sum_{j=3}^{n_x} (16j - 40) + \sum_{j=2}^{n_x} 2(j-1) \\ &= 9n_x^2 - 33n_x + 34. \end{aligned} \quad (5.7)$$

The controlled V_0 gate consists of a controlled phase gate and controlled $W_j, j = 1, \dots, n$. $c - W_j$ can be decomposed into a multi-controlled RZ gate (a controlled RZ gate for $j = 1$), two controlled H gates, and $2(j-1)$ controlled CNOT gates. Therefore, the total number of gates in the circuit implementation of the controlled V_0 gate can be upper-bounded by

$$\begin{aligned} &\mathcal{Q}_{c-V_0} \\ &= 2 + \sum_{j=3}^{n_x+1} (16j - 40) + 2n_x + \sum_{j=2}^{n_x} 2(j-1) * 8 \\ &= 16n_x^2 - 22n_x + 10. \end{aligned} \quad (5.8)$$

Then, V_{heat} can be implemented using single-qubit gates and at most

$$\begin{aligned} \mathcal{Q}_{V_{\text{heat}}} &= d2^{n_p-1}\mathcal{Q}_{V_0} + d(2^{n_p} - 1)\mathcal{Q}_{c-V_0} \\ &= O(dN_p n_x^2) \end{aligned} \quad (5.9)$$

CNOT gates. $\square$

Lemma 6. *Let $\mathbf{H}_{\text{heat}}$ be the Hamiltonian defined in Equation (3.8). The time evolution operator $U_{\text{heat}}(T) = \exp(i\mathbf{H}_{\text{heat}}T)$ for a time duration T can be implemented on a $(dn_x + n_p)$ -qubit system using quantum circuits with $O(d^2 n_x^2 N_p^2 \gamma_0^2 T^2 / \varepsilon)$ single-qubit gates and $O(d^2 n_x^3 N_p^2 \gamma_0^2 T^2 / \varepsilon)$ non-local gates, within an additive error of ε . The quantum circuit for $U_{\text{heat}}(T)$ involves repetitive applications of the one-time step unitary $V_{\text{heat}}(\tau)$ depicted in Figure 4.*

Proof. To suppress the error in the simulation over the total time T to a small value ε , it is sufficient to divide the total time T into r intervals, where $r = T/\tau$, such that

$$\|U_{\text{heat}}^r(\tau) - V_{\text{heat}}^r(\tau)\| \leq \frac{dN_p \gamma_0^2 T^2 (n_x - 1)}{4r} \leq \varepsilon, \quad (5.10)$$

which can be rearranged as

$$r \geq \frac{dN_p \gamma_0^2 T^2 (n_x - 1)}{4\varepsilon}. \quad (5.11)$$

According to Lemma 5, $V_{\text{heat}}(\tau)$ can be implemented using $O(dN_p n_x)$ single-qubit gates and at most $\mathcal{Q}_{V_{\text{heat}}} = O(dN_p n_x^2)$ CNOT gates. Therefore, $V_{\text{heat}}^r(\tau)$ can be implemented using $O(r\mathcal{Q}_{V_{\text{heat}}}) = O(d^2 n_x^3 N_p^2 \gamma_0^2 T^2 / \varepsilon)$ non-local gates and $O(d^2 n_x^2 N_p^2 \gamma_0^2 T^2 / \varepsilon)$ single-qubit gates. $\square$

Theorem 1. *Given the heat equation (3.1), the state $|\mathbf{u}(t)\rangle$, where $\mathbf{u}(t)$ is the solution of Equation (3.2) with a mesh size h , can be prepared with the precision ε using the Schrödingerization method depicted in Figure 4 and Figure 5. This preparation can be achieved using at most $\tilde{O}(d^2 T^2 \|\mathbf{u}(0)\|^3 / (\|\mathbf{u}(T)\|^3 h^4 \varepsilon^3))$ single-qubit gates and CNOT gates.*

Proof. As depicted in Figure 5, the quantum circuit involves a quantum Fourier transform, an inverse quantum Fourier transform, a unitary $V_{\text{heat}}^r(\tau)$ operation, and several measurements with $M_{\geq 0}$. The (inverse) quantum Fourier transform can be implemented with $O(n_p^2)$ CR gates, which corresponds to $O(n_p^2)$ CNOT gates [2, 34, 35].

The complexity analysis of simulating the unitary $V_{\text{heat}}^r(\tau)$ as concluded in Lemma 6 can be applied with $n_x = O(\log(L/h))$ and $\gamma_0 = a/(h^2 R)$. This results in a computational cost of $O(d^2 T^2 N_p^2 \log^3(L/h)/(h^4 \varepsilon' R^2))$, where ε' represents the desired precision $|U_{\text{heat}}(T) - V_{\text{heat}}^r(\tau)| \leq \varepsilon'$.

The number of measurements and the error of the output state can be discussed in the context of the given expressions $|\hat{\mathbf{v}}(T)\rangle = U(T)|\hat{\mathbf{v}}(0)\rangle$ and $|\hat{\mathbf{v}}_D(T)\rangle = V_{\text{heat}}^r(\tau)|\hat{\mathbf{v}}(0)\rangle$. To retrieve $|\mathbf{v}(T)\rangle$ and $|\mathbf{v}_D(T)\rangle$, an inverse quantum Fourier transform can be performed on these states. After the projection with $M_{\geq 0}$, one can observe that

$$\frac{\|M_{\geq 0}\mathbf{v}(T) - \mathbf{u}(T) \otimes \mathbf{p}_{\geq 0}\|}{\|\mathbf{u}(T)\| \|\mathbf{p}_{\geq 0}\|} = O\left(\frac{\pi R}{N_p} + e^{-\pi R}\right) \quad (5.12)$$

which represents the (relative) discretization error over the p variable, $\mathbf{p}_{\geq 0} := \sum_{p_k \geq 0} e^{-p_k} |k\rangle$. On the other hand,

$$\begin{aligned} & \| |M_{\geq 0} |\mathbf{v}_D(T)\rangle\rangle - |M_{\geq 0} |\mathbf{v}(T)\rangle\rangle \| \\ &= O\left(\frac{\|M_{\geq 0} (|\mathbf{v}_D(T)\rangle - |\mathbf{v}(T)\rangle)\|}{\|M_{\geq 0} |\mathbf{v}(T)\rangle\|}\right) \\ &= O\left(\frac{\|\mathbf{v}(0)\|}{\|\mathbf{u}(T)\| \|\mathbf{p}_{\geq 0}\|} \|U(T) - V_{\text{heat}}^r(\tau)\|\right) \\ &= O\left(\frac{\|\mathbf{u}(0)\|}{\|\mathbf{u}(T)\|} \varepsilon'\right), \end{aligned} \quad (5.13)$$

where we have used the fact that $\|\mathbf{p}_{\geq 0}\| = O(\|\mathbf{p}\|)$. To ensure the precision of the output state, we require that $R = O(\log(1/\varepsilon))$, $N_p = O(R/\varepsilon) = \tilde{O}(1/\varepsilon)$ and $\varepsilon' = O(\|\mathbf{u}(T)\| \varepsilon / \|\mathbf{u}(0)\|)$. Note that the probability of getting the desired state is $O(\|\mathbf{u}(T)\|^2 / \|\mathbf{u}(0)\|^2)$, hence $O(\|\mathbf{u}(0)\|^2 / \|\mathbf{u}(T)\|^2)$ measurements are

needed.

Combining all the steps, the Schrödingerization method for solving the heat equation utilizes at most $\tilde{O}\left(d^2 T^2 \frac{\|\mathbf{u}(0)\|^3}{\|\mathbf{u}(T)\|^3 h^4 \varepsilon^3}\right)$ single-qubit gates and CNOT gates. $\square$

Remark 4. In the classical implementation, the application of the difference operator requires $O(s2^{n_{xd}})$ arithmetic operations, where s denotes the sparsity of the difference operator and $s(D_D^\Delta) = 3$ in the case of the heat equation. Using the forward Euler scheme, the classical simulation up to time T within the additive error ε requires $O(T^2/\varepsilon)$ steps, which results in $O(s2^{n_{xd}}(T^2/\varepsilon))$ arithmetic operations. In addition, the Courant-Friedrichs-Lewy (CFL) condition requires the time step $\tau = O(h^2)$. As a result, classical simulation up to time T within the additive error ε requires $O(s2^{n_{xd}}(T^2/\varepsilon + T/h^2)) = O(s(T^2/h^d \varepsilon + T/h^{d+2}))$ arithmetic operations. Therefore, quantum advantage is achievable in higher-dimensional (for example, $d > 5$ if $h = O(\varepsilon)$) cases, where the quantum algorithm demands only a logarithmic number of operations compared to classical methods.

Remark 5. The discretization over p is essentially a Fourier spectral method applied to the function $e^{-|p|}$. The observed first-order convergence in p arises from the limited regularity of $e^{-|p|}$, which is continuous but not differentiable (C^1). To address this, we can consider a more general initial data $v(0, x, p) = g(p)u(0, x)$, where $g(p)$ is defined on $\mathbb{R}$ and satisfies:

$$g(p) = \begin{cases} h(p), & p \in (-\infty, 0], \\ e^{-p}, & p \in (0, +\infty). \end{cases} \quad (5.14)$$

The convergence rate can be enhanced by selecting a smoother $g(p)$, such as higher-order polynomials [36]. Assuming $g \in C^k(\mathbb{R})$, the discretization error over the p variable improves to $O(\Delta p^{k+1} + e^{-\pi R})$. Consequently, it suffices to choose $N_p = O(R/\varepsilon^{\frac{1}{k+1}}) = \tilde{O}(1/\varepsilon^{\frac{1}{k+1}})$. Overall, the number of single-qubit gates and CNOT gates required for

solving the heat equation can be reduced to $\tilde{O}\left(d^2 T^2 \|\mathbf{u}(0)\|^3 / \left(\|\mathbf{u}(T)\|^3 h^4 \varepsilon^{1+\frac{2}{k+1}}\right)\right)$.

5.2 The advection equation

In this section, we conduct a complexity analysis of the quantum circuit presented in Section 4 for the upwind scheme of the advection equation.

Lemma 7. *Consider the Schrödinger equation $d|\mathbf{u}(t)\rangle/dt = i\mathbf{H}_{adv}|\mathbf{u}(t)\rangle$, where the Hamiltonian $\mathbf{H}_{adv}$ is given by Equation (4.8). The time evolution operator $U_{adv}(\tau) = \exp(i\mathbf{H}_{adv}\tau)$ with a time increment τ can be approximated by the unitary $V_{adv}(\tau)$ in Equation (4.18), and its explicit circuit implementation is depicted in Figure 6-12. Furthermore, the approximation error in terms of the operator norm is upper-bounded by*

$$\begin{aligned} & \|U_{adv}(\tau) - V_{adv}(\tau)\| \\ & \leq \frac{\tau^2 n_x (N_p \gamma_2^2 + 2N_p \gamma_1 \gamma_2 + 2\gamma_1^2)}{4} \sum_{\alpha=1}^d a_\alpha^2, \end{aligned} \quad (5.15)$$

and

$$\begin{aligned} & \|U_{adv}(T) - V_{adv}^r(\tau)\| \\ & \leq \frac{T^2 n_x (N_p \gamma_2^2 + 2N_p \gamma_1 \gamma_2 + 2\gamma_1^2)}{4r} \sum_{\alpha=1}^d a_\alpha^2, \end{aligned} \quad (5.16)$$

where $r = T/\tau$, d denotes the spatial dimension, $N_p = 2^{n_p}$ and $N_x = 2^{n_x}$ represent the number of grid points for the variables p and x , respectively. Additionally, γ_1, γ_2 are defined in Equation (4.9), Equation (4.10).

Proof. Similarly to the proof of Lemma 4, the Trotter splitting error is upper-bounded by the norm of commutators. We utilize several commutator results here. For detailed calculations, please refer to Appendix B. Recalling the definition of $U_1(\tau)$, $U_2(\tau)$, $V_1(\tau)$ and $V_2(\tau)$ in Equation (4.13) and Equation (4.17), we have

$$\begin{aligned} \|U_1(\tau) - V_1(\tau)\| & \leq \frac{\gamma_1^2 \tau^2 n_x}{2}, \\ \|U_2(\tau) - V_2(\tau)\| & \leq \frac{\gamma_2^2 \tau^2 n_x}{2}, \end{aligned} \quad (5.17)$$

where the details are presented in Equation (B.10) and Equation (B.11). This implies that

$$\begin{aligned} \|\tilde{U}_1(\tau) - \tilde{V}_1(\tau)\| & \leq \frac{\gamma_1^2 \tau^2 n_x}{2} \sum_{\alpha=1}^d a_\alpha^2, \\ \|\tilde{U}_2(\tau) - \tilde{V}_2(\tau)\| & \leq \frac{\gamma_2^2 \tau^2 n_x}{2} \sum_{\alpha=1}^d a_\alpha^2. \end{aligned} \quad (5.18)$$

In all, we obtain

$$\begin{aligned} & \|U_{adv}(\tau) - V_{adv}(\tau)\| \\ & \leq \|U_{adv}(\tau) - U_*(\tau)\| + \|U_*(\tau) - V_{adv}(\tau)\| \\ & \leq \frac{\tau^2}{2} \left\| [\mathbf{A}_1 \otimes D_\eta, \mathbf{A}_2 \otimes I^{\otimes n_p}] \right\| \\ & \quad + \frac{N_p}{2} \|\tilde{U}_1(\tau) - \tilde{V}_1(\tau)\| + \|\tilde{U}_2(\tau) - \tilde{V}_2(\tau)\| \\ & \leq \frac{\tau^2 N_p \gamma_1 \gamma_2 n_x}{2} \sum_{\alpha=1}^d a_\alpha^2 \\ & \quad + \frac{N_p \gamma_1^2 \tau^2 n_x}{4} \sum_{\alpha=1}^d a_\alpha^2 + \frac{\gamma_1^2 \tau^2 n_x}{2} \sum_{\alpha=1}^d a_\alpha^2 \\ & \leq \frac{\tau^2 n_x (N_p \gamma_1^2 + 2N_p \gamma_1 \gamma_2 + 2\gamma_2^2)}{4} \sum_{\alpha=1}^d a_\alpha^2, \end{aligned} \quad (5.19)$$

where we have put Equation (B.8) in. Therefore

$$\begin{aligned} & \|U_{adv}(T) - V_{adv}^r(\tau)\| \\ & \leq r \|U_{adv}(\tau) - V_{adv}(\tau)\| \\ & \leq \frac{T^2 n_x (N_p \gamma_1^2 + 2N_p \gamma_1 \gamma_2 + 2\gamma_2^2)}{4r} \sum_{\alpha=1}^d a_\alpha^2. \end{aligned} \quad (5.20)$$

□

Lemma 8. *The approximated time evolution operator $V_{adv}(\tau)$ in Equation (4.18) can be implemented using $O(dN_p n_x)$ single-qubit gates and at most $\mathcal{Q}_{V_{adv}} = O(dN_p n_x^2)$ CNOT gates for $n_x \geq 3$.*

Proof. The proof is similar to the proof of Lemma 5. The implementation of $V_{adv}(\tau)$ involves a maximum of a $\tilde{V}_2(\tau)$ gate, $2^{n_p-1} \tilde{V}_1(-\tau)$ gates and $\sum_{m=0}^{n_p-1} 2^m = 2^{n_p} - 1$ controlled $\tilde{V}_1(\tau)$ gates. Each $\tilde{V}_1$ ($\tilde{V}_2$) gate consists of d V_1 (V_2) gates. Furthermore, V_1 and V_2 are similar to V_0 but incorporate additional

gates $U_1^{(1)}$ and $U_2^{(1)}$, each of which can be decomposed into a multi-controlled RZ gate, 2 Hadamard gates, 2 phase gates and $2(n_x - 1)$ X gates and $2(n_x - 1)$ CNOT gates. The controlled $U_1^{(1)}$ requires $2(n_x - 1)$ extra controlled X gates. Hence the number of single-qubit gates is $d(4n_x + 2 + 2 + 2(n_x - 1)) + d2^{n_p-1}(2n_x + 2 + 2(n_x - 1) + 1) = O(dN_p n_x)$.

The total number of CNOT gates in the circuit implementation of V_{adv} is determined by

$$\mathcal{Q}_{V_{\text{adv}}} = d\mathcal{Q}_{V_2} + d2^{n_p-1}\mathcal{Q}_{V_1} + d(2^{n_p} - 1)\mathcal{Q}_{c-V_1}, \quad (5.21)$$

with

$$\begin{aligned} \mathcal{Q}_{V_2} &= \mathcal{Q}_{V_1} = \mathcal{Q}_{V_0} + (16n_x - 40) + 2(n_x - 1) \\ &= 9n_x^2 - 15n_x - 8. \end{aligned} \quad (5.22)$$

$$\begin{aligned} \mathcal{Q}_{c-V_1} &= \mathcal{Q}_{c-V_0} + 16(n_x + 1) - 40 + 2n_x + 16(n_x - 1) \\ &= 16n_x^2 - 2n_x - 30. \end{aligned} \quad (5.23)$$

Then, V_{adv} can be implemented using $O(dN_p n_x)$ single-qubit gates and at most

$$\begin{aligned} \mathcal{Q}_{V_{\text{adv}}} &= d\mathcal{Q}_{V_2} + d2^{n_p-1}\mathcal{Q}_{V_1} + d(2^{n_p} - 1)\mathcal{Q}_{c-V_1} \\ &= O(dN_p n_x^2), \end{aligned} \quad (5.24)$$

CNOT gates.

□

Lemma 9. *Let $\mathbf{H}_{\text{adv}}$ be the Hamiltonian defined in Equation (4.8). The time evolution operator $U_{\text{adv}}(T) = \exp(i\mathbf{H}_{\text{adv}}T)$ for a time duration T can be implemented on a $(dn_x + n_p)$ -qubit system using quantum circuits with $O(dn_x^2 T^2 N_p (N_p \gamma_1^2 + 2N_p \gamma_1 \gamma_2 + 2\gamma_2^2) \sum_{\alpha=1}^d a_\alpha^2 / \varepsilon)$ single-qubit gates and $O(dn_x^3 T^2 N_p (N_p \gamma_1^2 + 2N_p \gamma_1 \gamma_2 + 2\gamma_2^2) \sum_{\alpha=1}^d a_\alpha^2 / \varepsilon)$ non-local gates, within an additive error of ε . The quantum circuit for $U_{\text{adv}}(T)$ involves repetitive applications of the one-time step unitary $V_{\text{adv}}(\tau)$ depicted in Figure 12.*

Proof. To suppress the error in the simulation over the total time T to a small value ε , it is sufficient to divide the total time T into r

intervals, where $r = T/\tau$, such that

$$\begin{aligned} &\|U_{\text{adv}}^r(\tau) - V_{\text{adv}}^r(\tau)\| \\ &\leq \frac{T^2 n_x (N_p \gamma_1^2 + 2N_p \gamma_1 \gamma_2 + 2\gamma_2^2)}{4r} \sum_{\alpha=1}^d a_\alpha^2 \\ &\leq \varepsilon, \end{aligned} \quad (5.25)$$

which can be rearranged as

$$r \geq \frac{T^2 n_x (N_p \gamma_1^2 + 2N_p \gamma_1 \gamma_2 + 2\gamma_2^2)}{4\varepsilon} \sum_{\alpha=1}^d a_\alpha^2. \quad (5.26)$$

According to Lemma 8, $V_{\text{adv}}(\tau)$ can be implemented using $O(dN_p n_x)$ single-qubit gates and at most $\mathcal{Q}_{V_{\text{adv}}} = O(dN_p n_x^2)$ CNOT gates. Therefore, $V_{\text{adv}}^r(\tau)$ can be implemented using $O(r\mathcal{Q}_{V_{\text{adv}}}) = O(dn_x^3 T^2 N_p (N_p \gamma_1^2 + 2N_p \gamma_1 \gamma_2 + 2\gamma_2^2) \sum_{\alpha=1}^d a_\alpha^2 / \varepsilon)$ non-local gates and $O(dn_x^2 T^2 N_p (N_p \gamma_1^2 + 2N_p \gamma_1 \gamma_2 + 2\gamma_2^2) \sum_{\alpha=1}^d a_\alpha^2 / \varepsilon)$ single-qubit gates. □

Theorem 2. *Given the advection equation (4.1), the state $|\mathbf{u}(t)\rangle$, where $\mathbf{u}(t)$ is the solution of Equation (4.2) with a mesh size h , can be prepared up to precision ε using the Schrödingerization method depicted in Figure 12 and Figure 5. This preparation can be achieved using at most $\tilde{O}(dT^2 \sum_{\alpha=1}^d a_\alpha^2 \|\mathbf{u}(0)\|^3 / (\|\mathbf{u}(T)\|^3 h^2 \varepsilon^3))$ single-qubit gates and CNOT gates.*

Proof. The complexity analysis of simulating the unitary $V_{\text{adv}}^r(\tau)$ as concluded in Lemma 9 can be applied with $n_x = O(\log(L/h))$ and $\gamma_1 = 1/(2hR)$, $\gamma_2 = 1/(2h)$. This results in a computational cost of $O(d^2 T^2 N_p^2 \log^3(L/h) \sum_{\alpha=1}^d a_\alpha^2 / (h^2 \varepsilon' R^2))$, with $\varepsilon' = \|\mathbf{u}(T)\| \varepsilon / \|\mathbf{u}(0)\|$.

The number of measurements and the error analysis of the output state are exactly the same as in the proof of Theorem 1, leading to $R = O(\log(1/\varepsilon))$, $N_p = O(R/\varepsilon) = \tilde{O}(1/\varepsilon)$. Combining all the steps, the Schrödingerization method for the upwind scheme of the advection equation utilizes at most $\tilde{O}(dT^2 \sum_{\alpha=1}^d a_\alpha^2 \frac{\|\mathbf{u}(0)\|^3}{\|\mathbf{u}(T)\|^3 h^2 \varepsilon^3})$ single-qubit gates and CNOT gates. □

Remark 6. As discussed in Remark 4, classical simulation up to time T within the additive error ε requires $O(s2^{n_x d}(T^2/\varepsilon + T/h)) = O(s(T^2/h^d\varepsilon + T/h^{d+1}))$ arithmetic operations since $\tau = O(h)$ for the advection equation. Therefore, quantum advantage is achievable in higher-dimensional (for example, $d > 4$ if $h = O(\varepsilon)$) cases. Compared to the quantum circuit constructed in [31], where the central difference operator is used to obtain a Hermitian matrix, the parameter N_p^2 in the gate complexity comes from the largest value of the Fourier variable η . However, our method is applicable to general non-Hermitian operators.

Remark 7. The improvements in the discretization error over the p variable can be achieved as discussed in Remark 5. By selecting $g \in C^k(\mathbb{R})$, the number of single-qubit gates and CNOT gates required for solving the advection equation can be reduced to $\tilde{O}\left(dT^2 \sum_{\alpha=1}^d a_\alpha^2 \|\mathbf{u}(0)\|^3 / (\|\mathbf{u}(T)\|^3 h^2 \varepsilon^{1+\frac{2}{k+1}})\right)$.

6 Numerical experiments

This section presents the numerical experiments conducted to validate the Schrödingerisation circuits introduced in Sections 3 and 4. The experiments are performed using the *Qiskit* package [32], which facilitates the construction and execution of the quantum circuits on a local simulator. Throughout the implementation phase, the *Statevector* simulator is utilized to retrieve the complete solution, while the *Estimator* function is employed for the measurement of observables. To contextualize the performance of the quantum simulations, comparisons are made against two reference points: a classical implementation of the Schrödingerisation method executed using the *Scipy* package [37], and the direct application of the matrix exponential operator $e^{\mathbf{A}t}$ to the initial condition vector $\mathbf{u}(0)$, achieved through conventional matrix-vector multiplication.

6.1 The heat equation

We first conduct a Hamiltonian simulation to solve the one-dimensional heat equation subject to Dirichlet boundary conditions, as explicated in Section 3,

$$\begin{cases} \partial_t u(t, x) = a \partial_{xx} u(t, x), & x \in \Omega = [0, L]. \\ u(0, x) = f(x), \end{cases} \quad (6.1)$$

Given the coefficient a and the initial condition $f(x)$ as

$$a = L/\pi^2, \quad f(x) = \sin(\pi x/L), \quad (6.2)$$

the exact solution is

$$u(t, x) = e^{-t/L} \sin(\pi x/L). \quad (6.3)$$

Employing the central difference approximation in conjunction with the Schrödingerisation procedure in a sequential manner, the discretized heat equation is governed by the following ODEs

$$\frac{d\mathbf{u}(t)}{dt} = \mathbf{A}\mathbf{u}(t) := aD_D^\Delta \mathbf{u}(t), \quad (6.4)$$

$$\frac{d\hat{\mathbf{v}}(t)}{dt} = i\mathbf{H}_{\text{heat}}\hat{\mathbf{v}}(t) := i(\mathbf{A} \otimes D_\eta)\hat{\mathbf{v}}(t), \quad (6.5)$$

where $\mathbf{u}(t) \in \mathbb{R}^{2^{n_x}}$, $\hat{\mathbf{v}}(t) \in \mathbb{R}^{2^{n_x+n_p}}$. To formulate the problem, we set $L = 17$, $n_x = 4$ for the central difference approximation, and set $n_p = 3, 5, 7$, $R = 4$ for the Schrödingerisation method. As shown in Figure 5, $V(\tau)$ is implemented $r = T/\tau$ times with the time increment parameter $\tau = 0.005$.

The numerical results at time $T = 5$ are illustrated in Figure 13 and Figure 14. As shown in Figure 13, there is a clear concordance between the numerical solutions derived from the quantum circuits and those obtained via the classical implementation of the Schrödingerisation method, regardless of the value of n_p . Furthermore, as the number of ancilla qubits n_p increases, the solutions produced by the Schrödingerisation method progressively converge towards the solution computed directly using the matrix exponential operator $e^{\mathbf{A}t}$. This convergence underscores

the Schrödingerisation method's capability to effectively approximate the solution. The discrepancy with the exact solution is due to the too small number of mesh points used (in both x and p) which can be reduced if more grid points are used, which is not done here due to our limited quantum computing resource.

Figure 14 illustrates the energy $\|\mathbf{u}\|^2$ obtained from the circuits using 10000 shots. This result is achieved using two approaches:

- Measure the last qubit in register p to obtain the probability $P(1)$ of it being in state $|1\rangle$. This corresponds to the projection of the state $|\mathbf{v}(T)\rangle$ onto $|\mathbf{u}(T)\rangle|k\rangle$ for $k \geq \frac{N_p}{2}$, i.e., $p_k \geq 0$. Thus, $P(1)$ provides an estimate of the magnitude of $M_{\geq 0}|\mathbf{v}(T)\rangle$, and $\|\mathbf{u}(T)\|$ is obtained by:

$$\|\mathbf{u}(T)\|^2 \approx \frac{P(1) \|\mathbf{u}(0)\|^2 \|\mathbf{p}\|^2}{\|\mathbf{p}_{\geq 0}\|^2}. \quad (6.6)$$

- Measure the quantum state $|\mathbf{v}\rangle$ with the observable

$$\text{obs} := I^{\otimes n_x} \otimes |1\rangle\langle 1| \otimes |0\rangle\langle 0|^{\otimes (n_p-1)}, \quad (6.7)$$

which projects the state $|\mathbf{v}(T)\rangle$ to $|\mathbf{u}(T)\rangle \left| \frac{N_p}{2} \right\rangle$, $p_{N_p/2} = 0$. Note that in this case $|\mathbf{u}(T)\rangle$ is recovered with $p = 0$ since $\mathbf{A}$ is negative definite. For general ODE systems, Jin, Liu and Ma systematically studied the important issue of recovering the original variables in [38].

The numerical results validate that both estimates reliably capture the energy measurements. The code is available at [GitHub](#) [39].

6.2 The advection equation

Next, we consider the one-dimensional advection equation subject to periodic boundary conditions, as explicated in Section 4,

$$\begin{cases} \partial_t u(t, x) = a \partial_x u(t, x), & x \in \Omega = [0, L]. \\ u(0, x) = f(x), \end{cases} \quad (6.8)$$

Given the coefficient a and the initial condition $f(x)$ as

$$a = 1, \quad f(x) = \begin{cases} 0, & x \in [kL, kL + L/2), \\ 1, & x \in [kL + L/2, kL), \end{cases} \quad (6.9)$$

the exact solution is

$$u(t, x) = f(x + at). \quad (6.10)$$

As discussed above, we use the upwind scheme to avoid numerical oscillations across discontinuities, and then the Schrödingerisation procedure is applied:

$$\frac{d\mathbf{u}(t)}{dt} = \mathbf{A}\mathbf{u}(t) := aD_P^+ \mathbf{u}(t), \quad (6.11)$$

$$\begin{aligned} \frac{d\hat{\mathbf{v}}(t)}{dt} &= i \left(\mathbf{A}_1 \otimes D_\eta + \mathbf{A}_2 \otimes I^{\otimes n_p} \right) \hat{\mathbf{v}}(t) \\ &\triangleq i\mathbf{H}_{\text{adv}} \hat{\mathbf{v}}(t), \end{aligned} \quad (6.12)$$

where $\mathbf{u}(t) \in \mathbb{R}^{2^{n_x}}$, $\hat{\mathbf{v}}(t) \in \mathbb{R}^{2^{n_x+n_p}}$. To formulate the problem, we set $L = 16$, $n_x = 4$ for the upwind scheme, $n_p = 3, 5, 7$, $R = 4$ for the Schrödingerisation method and the time increment parameter $\tau = 0.005$ for the quantum circuits.

The numerical results at time $T = 3$ are illustrated in Figure 15 and Figure 16. As observed previously, the numerical solutions generated from the quantum circuits agree well with those acquired through the classical Schrödingerisation implementation, and demonstrate convergence towards the solution computed directly using $e^{\mathbf{A}t}$ as n_p increases. In contrast to the numerical results obtained via the central difference method described in [31], the solutions derived through this approach are non-oscillatory due to the dissipative nature of the upwind scheme, rather than oscillatory for the central scheme in [31], across discontinuities.

Figure 16 illustrates the energy $\|\mathbf{u}\|^2$ observed from the circuits using 10000 shots. This is achieved by measuring the last qubit in register p and measuring the quantum state $|\mathbf{v}\rangle$ with the observable (6.7). The numerical results closely approximate those derived

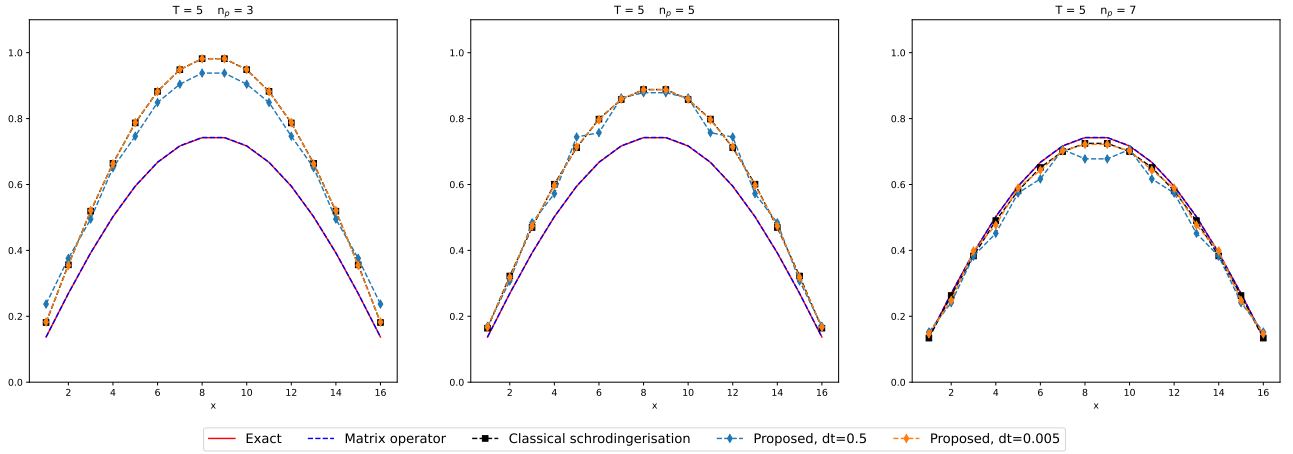


Figure 13: Numerical solutions u of the heat equation using the matrix exponential operator e^{AT} , the classical implementation and the proposed quantum circuits ($n_p = 3, 5, 7$, respectively) of the Schrödingerisation method.

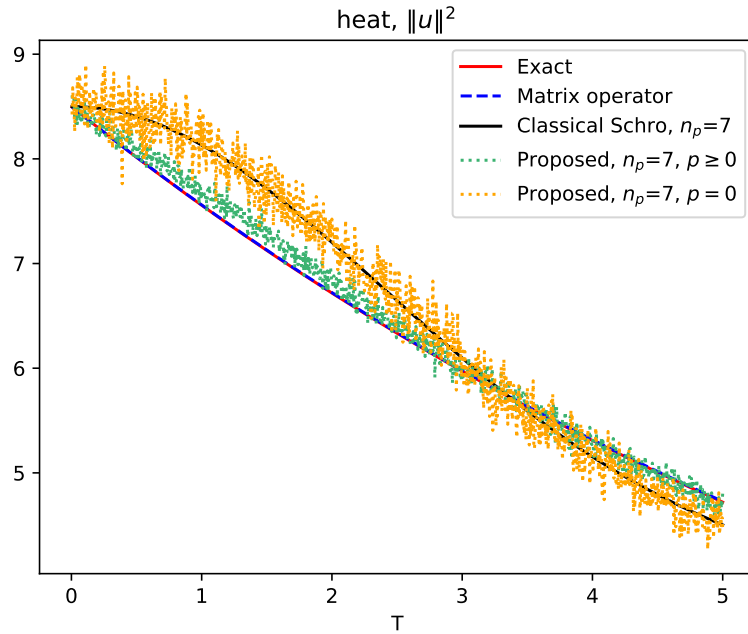


Figure 14: The energy $\|u\|^2$ of the heat equation using the matrix exponential operator e^{AT} , the classical implementation of the Schrödingerisation method and observations of the proposed quantum circuits ($n_p = 7$) using 10000 shots.

from the matrix exponential operator and exhibit dissipative characteristics attributable to the properties of the upwind scheme.

7 Conclusion

We present a practical implementation of quantum circuits for the Schrödingerization algorithm, which enables the solution of general partial differential equations (PDEs). To illustrate the approach, we demonstrate its application to two simple examples: the heat equation and the advection equation with the upwind scheme. Detailed quantum circuits are given and their complexities are analyzed, showing quantum advantages in high dimensions over their classical counterparts. The numerical experiments illustrate that the results obtained from these quantum circuits align closely with those achieved through the classical implementation of the Schrödingerisation method, as well as with the exact solution of the ordinary differential equation (ODE) discretized from the original partial differential equation (PDE).

In our future research, we aim to expand upon this work by developing scalable quantum circuits to accommodate higher order schemes, diverse boundary and interface conditions, and more challenging PDEs such as Maxwell and radiative transfer equations. Furthermore, we plan to implement these circuits on real quantum computers, thus bridging the gap between theory and practical quantum computing applications.

Acknowledgement

SJ, NL, and LZ were supported by NSFC grant No. 12341104, the Shanghai Jiao Tong University 2030 Initiative, and the Fundamental Research Funds for the Central Universities. SJ was also supported by the Shanghai Municipal Science and Technology Major Project (2021SHZDZX0102), the Innovation Program of Shanghai Municipal Education Commission (No. 2021-01-07-00-02-E00087).

NL was also supported by NSFC grant No. 12471411 and the Science and Technology Program of Shanghai, China (21JC1402900). LZ was also supported by NSFC grant No. 12271360, Shanghai Municipal Science and Technology Project (22JC1401600).

References

- [1] Richard P Feynman. “Simulating physics with computers”. In Feynman and computation. Pages 133–153. CRC Press (2018).
- [2] Michael A Nielsen and Isaac L Chuang. “Quantum computation and quantum information”. Cambridge university press. (2010).
- [3] Shi Jin and Nana Liu. “Analog quantum simulation of partial differential equations”. *Quantum Science and Technology* **9**, 035047 (2024).
- [4] Seth Lloyd. “Universal quantum simulators”. *Science* **273**, 1073–1078 (1996).
- [5] Dorit Aharonov and Amnon Ta-Shma. “Adiabatic quantum state generation and statistical zero knowledge”. In Proceedings of the thirty-fifth annual ACM symposium on Theory of computing. Pages 20–29. (2003).
- [6] Andrew Macgregor Childs. “Quantum information processing in continuous time”. PhD thesis. Massachusetts Institute of Technology. (2004). url: <http://hdl.handle.net/1721.1/16663>.
- [7] Dominic W Berry, Graeme Ahokas, Richard Cleve, and Barry C Sanders. “Efficient quantum algorithms for simulating sparse hamiltonians”. *Communications in Mathematical Physics* **270**, 359–371 (2007).
- [8] Nathan Wiebe, Dominic W Berry, Peter Høyer, and Barry C Sanders. “Simulating quantum dynamics on a quantum computer”. *Journal of Physics A: Mathematical and Theoretical* **44**, 445308 (2011).

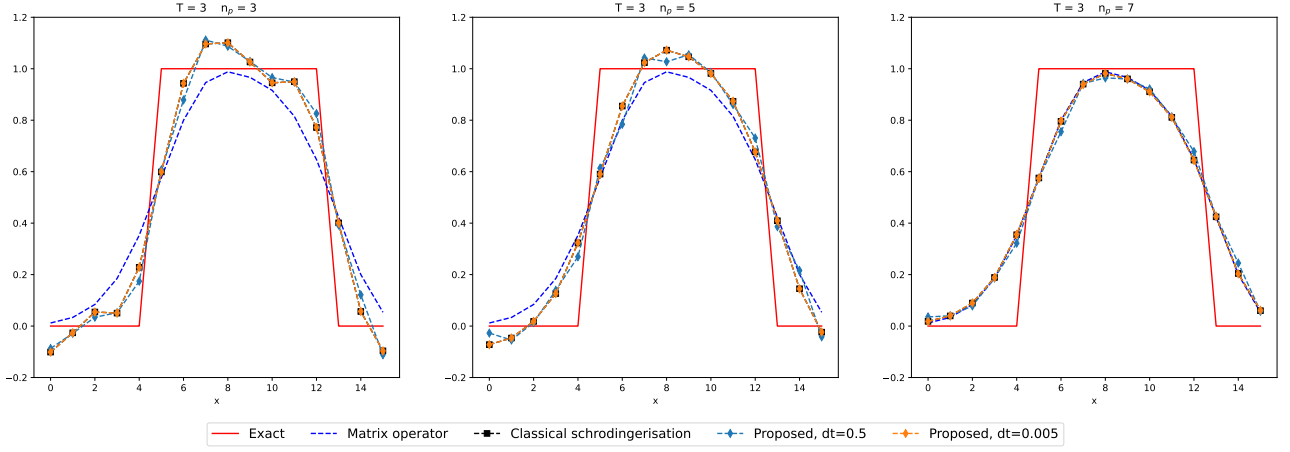


Figure 15: Numerical solutions u of the advection equation using the matrix exponential operator e^{AT} , the classical implementation and the proposed quantum circuits ($n_p = 3, 5, 7$, respectively) of the Schrödingerisation method.

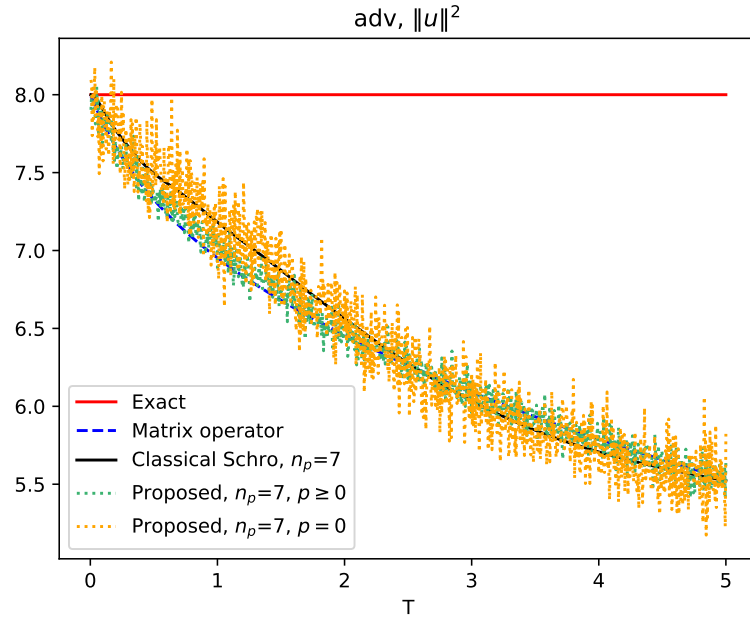


Figure 16: The energy $\|u\|^2$ of the advection equation using the matrix exponential operator e^{AT} , the classical implementation of the Schrödingerisation method and observations of the proposed quantum circuits ($n_p = 7$) using 10000 shots.

- [9] Andrew M Childs. “On the relationship between continuous-and discrete-time quantum walk”. *Communications in Mathematical Physics* **294**, 581–603 (2010).
- [10] Dominic W Berry and Andrew M Childs. “Black-box hamiltonian simulation and unitary implementation”. *Quantum Information & Computation* **12**, 29–62 (2012). url: <https://dl.acm.org/doi/abs/10.5555/2231036.2231040>.
- [11] Andrew M Childs and Nathan Wiebe. “Hamiltonian simulation using linear combinations of unitary operations”. *Quantum Information & Computation* **12**, 901–924 (2012). url: <https://dl.acm.org/doi/abs/10.5555/2481569.2481570>.
- [12] Dominic W Berry, Andrew M Childs, Richard Cleve, Robin Kothari, and Rolando D Somma. “Simulating hamiltonian dynamics with a truncated taylor series”. *Physical review letters* **114**, 090502 (2015).
- [13] Dominic W Berry, Andrew M Childs, Richard Cleve, Robin Kothari, and Rolando D Somma. “Exponential improvement in precision for simulating sparse hamiltonians”. In *Proceedings of the forty-sixth annual ACM symposium on Theory of computing*. Pages 283–292. (2014).
- [14] Dominic W Berry, Andrew M Childs, and Robin Kothari. “Hamiltonian simulation with nearly optimal dependence on all parameters”. In *2015 IEEE 56th annual symposium on foundations of computer science*. Pages 792–809. IEEE (2015).
- [15] Dominic W Berry, Andrew M Childs, Yuan Su, Xin Wang, and Nathan Wiebe. “Time-dependent hamiltonian simulation with l^1 -norm scaling”. *Quantum* **4**, 254 (2020).
- [16] Qi Zhao, You Zhou, Alexander F Shaw, Tongyang Li, and Andrew M Childs. “Hamiltonian simulation with random inputs”. *Physical Review Letters* **129**, 270502 (2022).
- [17] Aram W Harrow, Avinatan Hassidim, and Seth Lloyd. “Quantum algorithm for linear systems of equations”. *Physical review letters* **103**, 150502 (2009).
- [18] Yudong Cao, Anmer Daskin, Steven Frankel, and Sabre Kais. “Quantum circuit design for solving linear systems of equations”. *Molecular Physics* **110**, 1675–1680 (2012).
- [19] Andrew M Childs, Robin Kothari, and Rolando D Somma. “Quantum algorithm for systems of linear equations with exponentially improved dependence on precision”. *SIAM Journal on Computing* **46**, 1920–1950 (2017).
- [20] Dominic W Berry. “High-order quantum algorithm for solving linear differential equations”. *Journal of Physics A: Mathematical and Theoretical* **47**, 105301 (2014).
- [21] Dominic W Berry, Andrew M Childs, Aaron Ostrander, and Guoming Wang. “Quantum algorithm for linear differential equations with exponentially improved dependence on precision”. *Communications in Mathematical Physics* **356**, 1057–1081 (2017).
- [22] Andrew M Childs, Jin-Peng Liu, and Aaron Ostrander. “High-precision quantum algorithms for partial differential equations”. *Quantum* **5**, 574 (2021).
- [23] Shi Jin, Nana Liu, and Yue Yu. “Time complexity analysis of quantum difference methods for linear high dimensional and multiscale partial differential equations”. *Journal of Computational Physics* **471**, 111641 (2022).
- [24] Shi Jin and Nana Liu. “Quantum algorithms for nonlinear partial differential equations”. *Bulletin des Sciences Mathématiques* **194**, 103457 (2024).

- [25] Shi Jin, Nana Liu, and Yue Yu. “Time complexity analysis of quantum algorithms via linear representations for non-linear ordinary and partial differential equations”. *Journal of Computational Physics* **487**, 112149 (2023).
- [26] Shi Jin, Nana Liu, and Yue Yu. “Quantum simulation of partial differential equations via schrodingerisation” (2022). [arXiv:2212.13969](#).
- [27] Shi Jin, Nana Liu, and Yue Yu. “Quantum simulation of partial differential equations: Applications and detailed analysis”. *Physical Review A* **108**, 032603 (2023).
- [28] Zhen Lu and Yue Yang. “Quantum computing of reacting flows via hamiltonian simulation”. *Proceedings of the Combustion Institute* **40**, 105440 (2024).
- [29] Shi Jin and Nana Liu. “Quantum simulation of discrete linear dynamical systems and simple iterative methods in linear algebra”. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences* **480**, 20230370 (2024).
- [30] Dong An, Jin-Peng Liu, and Lin Lin. “Linear combination of hamiltonian simulation for nonunitary dynamics with optimal state preparation cost”. *Physical Review Letters* **131**, 150603 (2023).
- [31] Yuki Sato, Ruho Kondo, Ikko Hamamura, Tamiya Onodera, and Naoki Yamamoto. “Hamiltonian simulation for hyperbolic partial differential equations by scalable quantum circuits”. *Physical Review Research* **6**, 033246 (2024).
- [32] Qiskit contributors. “Qiskit: An open-source framework for quantum computing” (2023).
- [33] Andrew M Childs, Yuan Su, Minh C Tran, Nathan Wiebe, and Shuchen Zhu. “Theory of trotter error with commutator scaling”. *Physical Review X* **11**, 011020 (2021).
- [34] Rafaella Vale, Thiago Melo D Azevedo, Ismael CS Araújo, Israel F Araujo, and Adenilton J da Silva. “Circuit decomposition of multi-controlled special unitary single-qubit gates”. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* (2023).
- [35] Lin Lin. “Lecture notes on quantum algorithms for scientific computation” (2022). [arXiv:2201.08309](#).
- [36] Shi Jin, Nana Liu, and Chuwen Ma. “Schrödingerisation based computationally stable algorithms for ill-posed problems in partial differential equations” (2024). [arXiv:2403.19123](#).
- [37] Pauli Virtanen, Ralf Gommers, Travis E. Oliphant, Matt Haberland, Tyler Reddy, David Cournapeau, Evgeni Burovski, Pearu Peterson, Warren Weckesser, Jonathan Bright, Stéfan J. van der Walt, Matthew Brett, Joshua Wilson, K. Jarrod Millman, Nikolay Mayorov, Andrew R. J. Nelson, Eric Jones, Robert Kern, Eric Larson, C J Carey, İlhan Polat, Yu Feng, Eric W. Moore, Jake VanderPlas, Denis Laxalde, Josef Perktold, Robert Cimrman, Ian Henriksen, E. A. Quintero, Charles R. Harris, Anne M. Archibald, Antônio H. Ribeiro, Fabian Pedregosa, Paul van Mulbregt, and SciPy 1.0 Contributors. “SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python”. *Nature Methods* **17**, 261–272 (2020).
- [38] Shi Jin, Nana Liu, and Chuwen Ma. “On schrödingerization based quantum algorithms for linear dynamical systems with inhomogeneous terms” (2024). [arXiv:2402.14696](#).
- [39] Junpeng Hu. “Quantum circuits for partial differential equations via schrodingerisation”. GitHub Repository (2024). url: <https://github.com/hjp3268/Schrodingerisation-Circuits>.

A Several proofs related to the derivations of the circuits.

A.1 Proof of Lemma 2

Direct calculation of $e^{i\lambda}s_j^- + e^{-i\lambda}s_j^+$ yields:

$$e^{i\lambda}s_j^- + e^{-i\lambda}s_j^+ = I^{\otimes(n_x-j)} \otimes \left(e^{i\lambda} |a_j\rangle \langle b_j| + e^{-i\lambda} |b_j\rangle \langle a_j| \right), \quad (\text{A.1})$$

with

$$|a_j\rangle := |0\rangle|1\rangle^{\otimes(j-1)}, \quad |b_j\rangle := |1\rangle|0\rangle^{\otimes(j-1)}. \quad (\text{A.2})$$

Recalling Lemma 1, one can define the unitary matrix $B_j(\lambda)$ such that

$$B_j(\lambda) |0\rangle |1\rangle^{\otimes(j-1)} = \frac{|a\rangle + e^{-i\lambda} |b\rangle}{\sqrt{2}}, \quad B_j(\lambda) |1\rangle |1\rangle^{\otimes(j-1)} = \frac{|a\rangle - e^{-i\lambda} |b\rangle}{\sqrt{2}}, \quad (\text{A.3})$$

which are called the Bell basis in [31]. And $B_j(\lambda)$ is constructed by

$$B_j(\lambda) := \left(\prod_{m=1}^{j-1} \text{CNOT}_m^j \right) P_j(-\lambda) H_j, \quad (\text{A.4})$$

where H_j is the Hadamard gate acting on the j -th qubit, $P_j(\lambda)$ is the Phase gate acting on the j -th qubit as

$$P_j(\lambda) := \begin{bmatrix} 1 & 0 \\ 0 & e^{i\lambda} \end{bmatrix}, \quad (\text{A.5})$$

and CNOT_m^j is the CNOT gate acting on the m -th qubit controlled by the j -th qubit. Then Equation (A.1) can be simplified as

$$e^{i\lambda}s_j^- + e^{-i\lambda}s_j^+ = I^{\otimes(n_x-j)} \otimes B_j(\lambda) \left(Z \otimes |1\rangle\langle 1|^{\otimes(j-1)} \right) B_j(\lambda)^\dagger. \quad (\text{A.6})$$

The time evolution operator is formulated as

$$\begin{aligned} \exp \left(i\gamma\tau (e^{i\lambda}s_j^- + e^{-i\lambda}s_j^+) \right) &= I^{\otimes(n_x-j)} \otimes B_j(\lambda) \text{CRZ}_j^{1,\dots,j-1}(-2\gamma\tau) B_j(\lambda)^\dagger \\ &\triangleq I^{\otimes(n_x-j)} \otimes W_j(\gamma\tau, \lambda), \end{aligned} \quad (\text{A.7})$$

where $\text{CRZ}_j^{1,\dots,j-1}(\theta)$ is the multi-controlled RZ gate as defined in Lemma 1.

A.2 Proof of Lemma 3

We utilize the binary representation of integers $k = (k_{n_p-1} \cdots k_0) = \sum_{m=0}^{n_p-1} k_m 2^m$. This allows us to express $Q_0^k(\tau)$ as $\prod_{m=0}^{n_p-1} Q_0^{k_m 2^m}(\tau)$, resulting in a reduced number of gates:

$$\begin{aligned} Q &= \sum_{k=0}^{N_p-1} Q_0^k(\tau) \otimes |k\rangle\langle k| \\ &= \sum_{k_{n_p-1}, \dots, k_0} \prod_{m=0}^{n_p-1} Q_0^{k_m 2^m}(\tau) \otimes \left(|k_{n_p-1}\rangle\langle k_{n_p-1}| \right) \otimes \cdots \otimes \left(|k_0\rangle\langle k_0| \right) \\ &= \prod_{m=0}^{n_p-1} \left(Q_0^{k_m 2^m}(\tau) \otimes \sum_{k_m} |k_m\rangle\langle k_m| \right) \\ &= \prod_{m=0}^{n_p-1} \left(Q_0^{2^m}(\tau) \otimes |1\rangle\langle 1| + I^{\otimes n_x} \otimes |0\rangle\langle 0| \right), \end{aligned} \quad (\text{A.8})$$

where the primed product $\prod'$ denotes the regular matrices product for the first register (consisting of n_x qubits) and the tensor product for the second register (consisting of n_p qubits).

A.3 Proof of Equation (3.10)

By definition,

$$U_{\text{heat}}(\tau) := \exp(i\mathbf{H}_{\text{heat}}\tau) = \exp\left(i\tau \sum_{k=0}^{N_p-1} \left(k - \frac{N_p}{2}\right) \sum_{\alpha=1}^d (\mathbf{H}_0)_\alpha \otimes |k\rangle\langle k|\right). \quad (\text{A.9})$$

Combining with the following properties,

$$\begin{aligned} \exp\left(\sum_k A_k \otimes |k\rangle\langle k|\right) &= \sum_k \exp(A_k) \otimes |k\rangle\langle k|, \\ \exp\left(\sum_\alpha (A)_\alpha\right) &= \prod_\alpha (\exp(A))_\alpha, \end{aligned} \quad (\text{A.10})$$

we obtain

$$\begin{aligned} U_{\text{heat}}(\tau) &= \sum_{k=0}^{N_p-1} \exp\left(i\tau \left(k - \frac{N_p}{2}\right) \sum_{\alpha=1}^d (\mathbf{H}_0)_\alpha\right) \otimes |k\rangle\langle k| \\ &= \sum_{k=0}^{N_p-1} \left(\exp\left(i\tau \sum_{\alpha=1}^d (\mathbf{H}_0)_\alpha\right)\right)^{k-N_p/2} \otimes |k\rangle\langle k| \\ &= \sum_{k=0}^{N_p-1} \left(\prod_{\alpha=1}^d \exp(i\tau (\mathbf{H}_0)_\alpha)\right)^{k-N_p/2} \otimes |k\rangle\langle k| \\ &= \sum_{k=0}^{N_p-1} \left(\prod_{\alpha=1}^d (U_0(\tau))_\alpha\right)^{k-N_p/2} \otimes |k\rangle\langle k| \\ &= \sum_{k=0}^{N_p-1} \tilde{U}_0^{k-N_p/2}(\tau) \otimes |k\rangle\langle k|. \end{aligned} \quad (\text{A.11})$$

A.4 Proof of Equation (4.12)

By employing the first-order Lie-Trotter-Suzuki decomposition, we can approximate the time evolution operator $U_{\text{adv}}(\tau) := \exp(i\mathbf{H}_{\text{adv}}\tau)$ as follows,

$$\begin{aligned} U_{\text{adv}}(\tau) &= \exp\left(i\tau \left(\sum_{k=0}^{N_p-1} \left(k - \frac{N_p}{2}\right) \sum_{\alpha=1}^d |a_\alpha| (\mathbf{H}_1)_\alpha \otimes |k\rangle\langle k| + \sum_{\alpha=1}^d a_\alpha (\mathbf{H}_2)_\alpha \otimes I^{\otimes n_p}\right)\right) \\ &\approx \exp\left(i\tau \sum_{\alpha=1}^d a_\alpha (\mathbf{H}_2)_\alpha \otimes I^{\otimes n_p}\right) \exp\left(i\tau \sum_{k=0}^{N_p-1} \left(k - \frac{N_p}{2}\right) \sum_{\alpha=1}^d |a_\alpha| (\mathbf{H}_1)_\alpha \otimes |k\rangle\langle k|\right) \\ &= \left(\prod_{\alpha=1}^d \exp(ia_\alpha\tau (\mathbf{H}_2)_\alpha) \otimes I^{\otimes n_p}\right) \sum_{k=0}^{N_p-1} \left(\prod_{\alpha=1}^d \exp(i|a_\alpha|\tau (\mathbf{H}_1)_\alpha)\right)^{k-N_p/2} \otimes |k\rangle\langle k| \\ &= \left(\prod_{\alpha=1}^d (U_2(a_\alpha\tau))_\alpha \otimes I^{\otimes n_p}\right) \sum_{k=0}^{N_p-1} \left(\prod_{\alpha=1}^d (U_1(|a_\alpha|\tau))_\alpha\right)^{k-N_p/2} \otimes |k\rangle\langle k| \\ &= \left(\tilde{U}_2(\tau) \otimes I^{\otimes n_p}\right) \sum_{k=0}^{N_p-1} \tilde{U}_1^{k-N_p/2}(\tau) \otimes |k\rangle\langle k|. \end{aligned} \quad (\text{A.12})$$

B Several calculations related to the complexity analysis

B.1 Calculation of commutators

In this section, we calculate several commutators and the upper bound of their norms, which are used in the proof of Lemma 4 and Lemma 7.

Note that $\sigma_{01}^2 = \sigma_{10}^2 = 0$, $\sigma_{01}\sigma_{10} = \sigma_{00}$, $\sigma_{10}\sigma_{01} = \sigma_{11}$. Direct calculation gives

$$\begin{aligned}
 [\sigma_{01}^{\otimes n_x}, \sigma_{10}^{\otimes n_x}] &= \sigma_{00}^{\otimes n_x} - \sigma_{11}^{\otimes n_x}, \\
 [s_j^-, \sigma_{01}^{\otimes n_x}] &= 0, \quad [s_j^-, \sigma_{10}^{\otimes n_x}] = (\sigma_{10}^{\otimes(n_x-1)} \otimes Z) \delta_{j1}, \\
 [s_j^+, \sigma_{10}^{\otimes n_x}] &= 0, \quad [s_j^+, \sigma_{01}^{\otimes n_x}] = (-\sigma_{01}^{\otimes(n_x-1)} \otimes Z) \delta_{j1}, \\
 [s_j^-, s_{j'}^+] &= I^{\otimes(n-j)} \otimes (\sigma_{00} \otimes \sigma_{11}^{\otimes(j-1)} - \sigma_{11} \otimes \sigma_{00}^{\otimes(j-1)}) \delta_{jj'}, \\
 [s_j^-, s_{j'}^-] &= 0, \quad j \geq j' > 1, \quad [s_1^-, s_1^-] = 0, \\
 [s_j^-, s_1^-] &= -I^{\otimes(n_x-j)} \otimes \sigma_{01} \otimes \sigma_{10}^{\otimes(j-2)} \otimes Z, \quad j > 1, \\
 [s_j^+, s_{j'}^+] &= 0, \quad j \geq j' > 1, \quad [s_1^+, s_1^+] = 0, \\
 [s_j^+, s_1^+] &= I^{\otimes(n_x-j)} \otimes \sigma_{10} \otimes \sigma_{01}^{\otimes(j-2)} \otimes Z, \quad j > 1.
 \end{aligned} \tag{B.1}$$

As shown in Lemma 1 and Remark 2, it follows that

$$\begin{aligned}
 \|[\sigma_{01}^{\otimes n_x}, \sigma_{10}^{\otimes n_x}]\| &= \|[s_j^-, s_j^+]\| = 1, \\
 \|[s_1^-, \sigma_{10}^{\otimes n_x}] + [s_1^+, \sigma_{01}^{\otimes n_x}]\| &= \|(\sigma_{10}^{\otimes(n_x-1)} - \sigma_{01}^{\otimes(n_x-1)}) \otimes Z\| = 1, \\
 \|[s_1^-, \sigma_{10}^{\otimes n_x}] - [s_1^+, \sigma_{01}^{\otimes n_x}]\| &= \|(\sigma_{10}^{\otimes(n_x-1)} + \sigma_{01}^{\otimes(n_x-1)}) \otimes Z\| = 1, \\
 \|[s_j^-, s_1^-] + [s_j^+, s_1^+]\| &= \|I^{\otimes(n_x-j)} \otimes (\sigma_{10} \otimes \sigma_{01}^{\otimes(j-2)} - \sigma_{01} \otimes \sigma_{10}^{\otimes(j-2)}) \otimes Z\| = 1, \quad j > 1, \\
 \|[s_{n_x}^-, s_{n_x}^+] - [\sigma_{01}^{\otimes n_x}, \sigma_{10}^{\otimes n_x}]\| &= \|I \otimes (\sigma_{11}^{\otimes(n_x-1)} - \sigma_{00}^{\otimes(n_x-1)})\| = 1.
 \end{aligned} \tag{B.2}$$

Therefore we have the following results

$$\left\| \left[\sum_{j=1}^{n_x} (s_j^- + s_j^+), \sigma_{01}^{\otimes n_x} + \sigma_{10}^{\otimes n_x} \right] \right\| = \|[s_1^-, \sigma_{10}^{\otimes n_x}] + [s_1^+, \sigma_{01}^{\otimes n_x}]\| = 1. \tag{B.3}$$

$$\left\| \left[\sum_{j=1}^{n_x} (s_j^- - s_j^+), \sigma_{01}^{\otimes n_x} - \sigma_{10}^{\otimes n_x} \right] \right\| = \|[s_1^-, \sigma_{10}^{\otimes n_x}] - [s_1^+, \sigma_{01}^{\otimes n_x}]\| = 1. \tag{B.4}$$

$$\sum_{j=1}^{n_x} \sum_{j'=j+1}^{n_x} \|[s_j^- + s_j^+, (s_{j'}^- + s_{j'}^+)]\| = \sum_{j'=2}^{n_x} \|[s_1^-, s_{j'}^-] + [s_1^+, s_{j'}^+]\| = n_x - 1. \tag{B.5}$$

$$\sum_{j=1}^{n_x} \sum_{j'=j+1}^{n_x} \|[s_j^- - s_j^+, (s_{j'}^- - s_{j'}^+)]\| = \sum_{j'=2}^{n_x} \|[s_1^-, s_{j'}^-] - [s_1^+, s_{j'}^+]\| = n_x - 1. \tag{B.6}$$

$$\begin{aligned}
\|[\mathbf{H}_1, \mathbf{H}_2]\| &= \gamma_1 \gamma_2 \left\| -2 \left[\sum_{j=1}^{n_x} s_j^-, \sum_{j=1}^{n_x} s_j^+ \right] + 2 [\sigma_{01}^{\otimes n_x}, \sigma_{10}^{\otimes n_x}] \right\| \\
&= \gamma_1 \gamma_2 \left\| -2 \sum_{j=1}^{n_x} [s_j^-, s_j^+] + 2 [\sigma_{01}^{\otimes n_x}, \sigma_{10}^{\otimes n_x}] \right\| \\
&\leq 2\gamma_1 \gamma_2 \left(\sum_{j=1}^{n_x-1} \| [s_j^-, s_j^+] \| + \| [s_{n_x}^-, s_{n_x}^+] - [\sigma_{01}^{\otimes n_x}, \sigma_{10}^{\otimes n_x}] \| \right) \\
&= 2\gamma_1 \gamma_2 n_x.
\end{aligned} \tag{B.7}$$

$$\begin{aligned}
&\| [\mathbf{A}_1 \otimes D_\eta, \mathbf{A}_2 \otimes I^{\otimes n_p}] \| \\
&= \left\| \left[\sum_{k=0}^{N_p-1} \left(k - \frac{N_p}{2} \right) \sum_{\alpha=1}^d |a_\alpha| (\mathbf{H}_1)_\alpha \otimes |k\rangle \langle k|, \sum_{\alpha=1}^d a_\alpha (\mathbf{H}_2)_\alpha \otimes I^{\otimes n_p} \right] \right\| \\
&= \left\| \sum_{k=0}^{N_p-1} \left(k - \frac{N_p}{2} \right) \left[\sum_{\alpha=1}^d |a_\alpha| (\mathbf{H}_1)_\alpha, \sum_{\alpha=1}^d a_\alpha (\mathbf{H}_2)_\alpha \right] \otimes |k\rangle \langle k| \right\| \\
&= \left\| \sum_{k=0}^{N_p-1} \left(k - \frac{N_p}{2} \right) \sum_{\alpha=1}^d |a_\alpha| a_\alpha [(\mathbf{H}_1)_\alpha, (\mathbf{H}_2)_\alpha] \otimes |k\rangle \langle k| \right\| \\
&\leq \frac{N_p \sum_{\alpha=1}^d a_\alpha^2}{2} \|[\mathbf{H}_1, \mathbf{H}_2]\| \\
&\leq N_p \gamma_1 \gamma_2 n_x \sum_{\alpha=1}^d a_\alpha^2.
\end{aligned} \tag{B.8}$$

B.2 Approximation of operators

$$\begin{aligned}
&\|U_{\text{heat}}(\tau) - V_{\text{heat}}(\tau)\| \\
&= \left\| \sum_{k=0}^{N_p-1} \left[\left(\prod_{\alpha=1}^d (U_0(\tau))_\alpha \right)^{k-N_p/2} - \left(\prod_{\alpha=1}^d (V_0(\tau))_\alpha \right)^{k-N_p/2} \right] \otimes |k\rangle \langle k| \right\| \\
&= \max_{0 \leq k \leq N_p-1} \left\| \left(\prod_{\alpha=1}^d (U_0(\tau))_\alpha \right)^{k-N_p/2} - \left(\prod_{\alpha=1}^d (V_0(\tau))_\alpha \right)^{k-N_p/2} \right\| \\
&\leq \max_{0 \leq k \leq N_p-1} \left| k - \frac{N_p}{2} \right| \left\| \prod_{\alpha=1}^d (U_0(\tau))_\alpha - \prod_{\alpha=1}^d (V_0(\tau))_\alpha \right\| \\
&\leq \frac{dN_p}{2} \|U_0(\tau) - V_0(\tau)\| \\
&\leq \frac{dN_p \gamma_0^2 \tau^2 (n_x - 1)}{4}.
\end{aligned} \tag{B.9}$$

$$\begin{aligned}
\|U_1(\tau) - V_1(\tau)\| &\leq \|U_1(\tau) - U_1^{(1)}(\tau)U_1^{(2)}(\tau)\| + \|U_1^{(1)}(\tau)U_1^{(2)}(\tau) - V_1(\tau)\| \\
&\leq \frac{\gamma_1^2 \tau^2}{2} \left\| \left[\sum_{j=1}^{n_x} (s_j^- + s_j^+), \sigma_{01}^{\otimes n_x} + \sigma_{10}^{\otimes n_x} \right] \right\| + \|U_1^{(2)}(\tau) - V_1^{(2)}(\tau)\| \\
&\leq \frac{\gamma_1^2 \tau^2}{2} + \frac{\gamma_1^2 \tau^2 (n_x - 1)}{2} = \frac{\gamma_1^2 \tau^2 n_x}{2},
\end{aligned} \tag{B.10}$$

$$\begin{aligned}
\|U_2(\tau) - V_2(\tau)\| &\leq \|U_2(\tau) - U_2^{(1)}(\tau)U_2^{(2)}(\tau)\| + \|U_2^{(1)}(\tau)U_2^{(2)}(\tau) - V_2(\tau)\| \\
&\leq \frac{\gamma_2^2\tau^2}{2} \left\| \left[\sum_{j=1}^{n_x} (s_j^- - s_j^+), \sigma_{01}^{\otimes n_x} - \sigma_{10}^{\otimes n_x} \right] \right\| + \|U_2^{(2)}(\tau) - V_2^{(2)}(\tau)\| \\
&\leq \frac{\gamma_2^2\tau^2}{2} + \frac{\gamma_2^2\tau^2(n_x - 1)}{2} = \frac{\gamma_2^2\tau^2 n_x}{2},
\end{aligned} \tag{B.11}$$

where the results from Equation (B.3), Equation (B.4) and Equation (5.3) are applied.