Quantum Algorithms for Solving PDEs

Xiantao Li

Department of Mathematics

Pennsylvania State University

SIAM NNP

Outline

- Motivation
- Simulating quantum dynamics
- From lattice dynamics to Schrödinger equation
- General non-unitary non-dissipative differential equations
- Extension to stochastic/steady state/nonlinear dynamics

The Quantum Computing Promises:

- **2025: The Year of Quantum:** significant milestone/accelerating interest/ investment in the quantum field.
- **Overcoming Classical Barriers:** potential to overcome the barrier faced by classical processors.
- **Natural Simulation:** natural fit to simulate quantum chemistry and quantum physics (**exponential speedup**).

QM models (TDSE)

- $\frac{d}{dt}|\psi\rangle = -iH|\psi\rangle$
- 1st quantization: $H = \sum_j -\frac{\nabla_j^2}{2} + V(x)$
- 2nd quantization:
$$H = \sum_{ij} t_{ij} a_i^\dagger a_j + \sum_{ijkl} V_{ijkl}\, a_i^\dagger a_j^\dagger a_k a_l$$
- Transverse-Field Ising
$$H = -J \sum_{ij} \sigma_i^z \sigma_j^z - \sum_i \sigma_i^x$$

Scientific computing tasks

Large-scale ODE/PDEs

- $u_t = \kappa u_{xx}, u_{tt} = c^2 u_{xx}$
- Nonlinearity
- Stochastic dynamics
- Feedback control
- Machine-learning models $x' = NN_\theta(x, t)$.
- Optimizations
- Sampling.

These models are very different from TDSE

Can quantum computers simulate classical dynamics?

Express them as Schrödinger equations!

Time-dependent Schrödinger equation

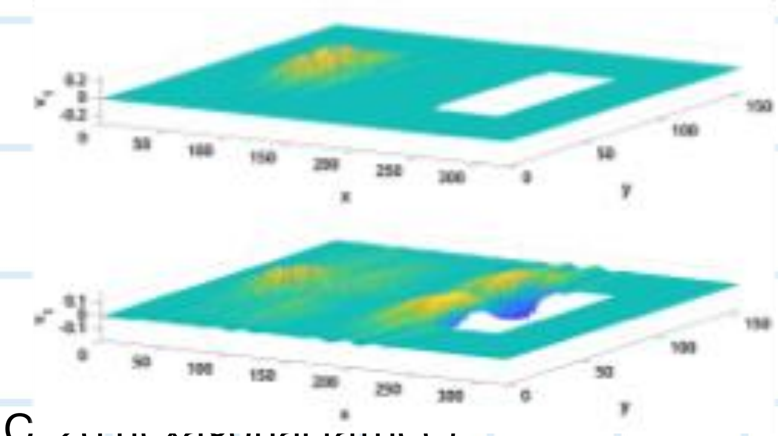
- TDSE: $\frac{d}{dt}|\psi\rangle = -iH|\psi\rangle$. H: self-adjoint (Hermitian) $H^\dagger = H$
- State-of-the-art: the evolution in $\mathbb{C}^N$ can be efficiently simulated: $\text{Cost} = T||H|| \log N \, polylog \frac{1}{\epsilon}$
- Algorithms:
  - Operator splitting (Trotter): $e^{-it(A+B)} \approx e^{-itA} e^{-itB}$. (Childs et al PRX 2021).
    - each exactly implemented by gate operations
  - Block encoding: $U_A = \frac{1}{\alpha}\begin{bmatrix} H & \cdot \\ \cdot & \cdot \end{bmatrix}$
  - QSVT: $U_A = \begin{bmatrix} H & \cdot \\ \cdot & \cdot \end{bmatrix} \to U_{p(A)} = \begin{bmatrix} p(H) & \cdot \\ \cdot & \cdot \end{bmatrix}$ (Gilyén *et al.* 2019)
  - LCU (Childs-Wiebe 2012).
  - Approximate diagonalizations
- ***Such a quantum speedup can be leveraged if the problem can be reduced to TDSE***

Wave equations

- Wave equation. $\partial_t^2 u = c^2 \nabla^2 u$
- Factorization: $-\boldsymbol{\nabla}_h^2 = \boldsymbol{Q}^T \boldsymbol{Q}, \boldsymbol{Q}$ **sparse − rectangle matrix**
- $\Rightarrow \frac{\boldsymbol{d}}{\boldsymbol{dt}} \boldsymbol{\psi} = -\boldsymbol{iH\psi},\ \ \boldsymbol{H} = -\begin{bmatrix} & \boldsymbol{Q}^T \\ \boldsymbol{Q} & \end{bmatrix}$. Costa et al 2019. Babbush, et al. 2023.
- Vector-valued discretized wave equation: $\ddot{u}_j = -\sum_{k=1}^{N} D_{j-k} u_k$. Or $\ddot{u} = -Du$.
- Dispersion relation. $\widehat{D}(k) = \sum_j D_j\, e^{-ik\cdot j}$.
- Trigonometric factorization: $D_j = \frac{1}{|B|} \int_B \widehat{D}(k) e^{ik\cdot j} dk = \frac{1}{|B|} \int_B \hat{Q}(k)^\dagger \hat{Q}(k) \quad e^{ik\cdot j} dk$
- Exact factorization (Fejer-Rietz factorization). $\widehat{D}(k) = \sum_j D_j\, z^j, \hat{Q}(k) = \sum_j Q_j\, z^j$
- $\widehat{D}(k) = \left|\hat{Q}(k)\right|^2 \Rightarrow\ D = Q^\dagger Q$ (matrix multiplication by convolution)

From wave equations to Schrödinger (Li, PRL)

- $\ddot{u} = -Du$ (Lattice waves/finite difference for acoustic and elastic wave equations)
- $D = Q^\dagger Q \Rightarrow \frac{d}{dt}\psi = -iH\psi,\ H = -\begin{bmatrix} & Q^T \\ Q & \end{bmatrix}$
- Example: $\ddot{u}_j = -\frac{1}{6}u_{j-2} + u_{j-1} - \frac{5}{3}u_j + u_{j+1} - \frac{1}{6}u_{j+2}$.
- $\widehat{D}(k) = 2(1-\cos k) - \frac{1}{3}(1-\cos 2k)$. $\hat{Q} = q_0 + q_1 e^{-ik} + q_2 e^{-i2k}$.
- $Q = \begin{bmatrix} q_0 & & & \\ q_1 & q_0 & & \\ q_2 & q_1 & \ddots & \\ & q_2 & \ddots & \\ & & \ddots & \end{bmatrix}_{(N+2)\times N}$



- In general, $Q = \sum_j A_j \otimes Q_j$. A_j: binary entries labelling neighbors. (3d FCC, 2d hexagonal lattice)
- The optimal Hamiltonian simulation algorithms apply.
- Overall quantum simulation complexity: logarithmic in system size and precision, linear in time * Debye frequency
- Linear wave equations are quantum easy.
- ***What about general ODEs ? (dissipative, stable, unstable, non-autonomous, ODEs).***

Linear dissipative ODEs/PDEs

- Linear PDEs $u_t = Lu$. E.g., $u_t = \kappa u_{xx}, u_{tt} = c^2 u_{xx}, iu_t = -\frac{\nabla^2}{2}u + V(x)u - i\Sigma u$
- Spatial discretization ➔ : $x' = Ax, \quad A = -iA_0 + A_1$
  - A_0 and A_1 Hermitian, $A_0^\dagger = A_0, A_1^\dagger = A_1, A_1 \le 0 \Rightarrow \frac{d}{dt}||x(t)|| \le 0.$
- **Schrödingerization**: Jin-Liu-Yu, PRL 2024.
  - $\psi(t,p) = e^p x(t), \mathrm{p} \ge 0.$
  - $\partial_t \psi = -iA_0\psi + A_1 \partial_p \psi = -iH\psi,$
  - With the right BCs, H is Hermitian. (∂_p can be turned into a skew Hermitian operator)
- Example: the heat equation
  - $\partial_t u = \kappa \partial_{xx} u. A_0 = 0, A_1 = \kappa \partial_{xx}$
  - After Schrödingerization: $\partial_t \psi(t,x,p) = \kappa \partial_{pxx}\psi.$
  - To recover the solution: $u(t,x) = e^{-p_*}\psi(t,x,p_*).$
- Linear combination of Hamiltonian evolution An-Liu-Lin, PRL 2024.
  - $e^{-iA_0 t - A_1 t} = \int \frac{1}{\pi(1+k^2)} e^{-itA_0 - itkA_1} dk$

An EEE framework (ArXiv:2507.10285) for $u_t = \mathcal{L}(t)u$

- $x' = -iH(t)x + K(t)x, , H^\dagger = H, K^\dagger = K$
- A general dilation scheme

$(l| \otimes I$ Evaluation ($(l|$: linear functional

$\mathcal{T}e^{-i\int_0^t I_a \otimes H(s) + iF_a \otimes K(s)ds}$ *Evolution*

$F_a^\dagger = -F_a$, in $\mathcal{H}_a$

$\mathcal{H}_a$ dense in $\mathbb{X}$

$|r) \otimes |x_0\rangle$ *Encoding*

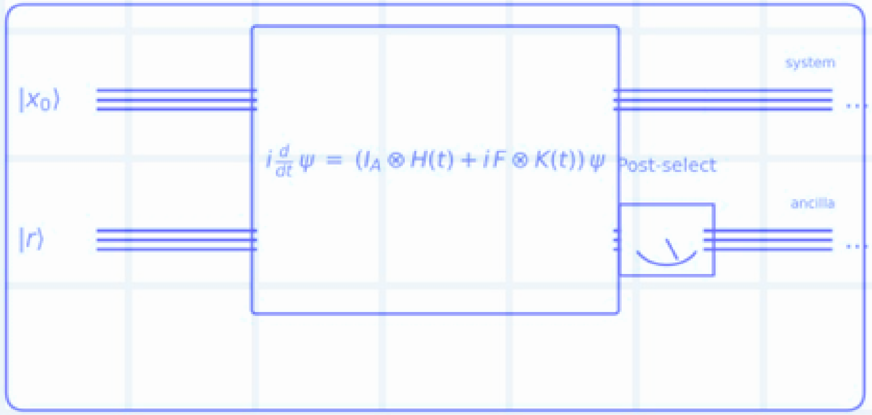
$|r) = f(p) \in \mathbb{X}$

= $\mathcal{T}e^{\int_0^t A(s)ds}|x_0\rangle$ *Exact ODE Evolution*

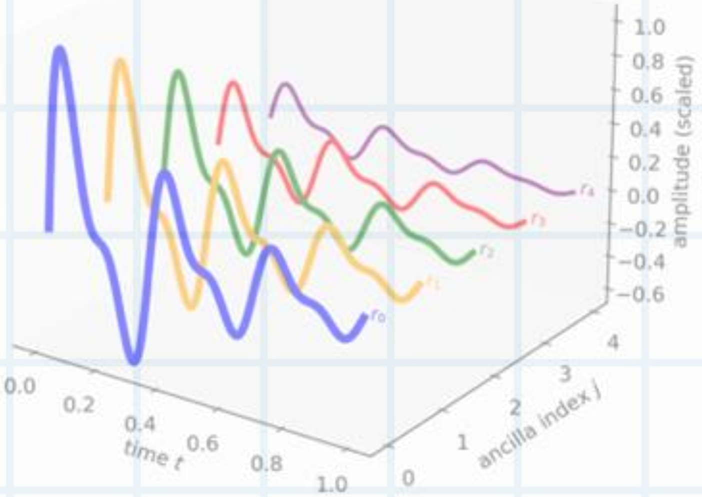
- **Theorem:** $(l| \otimes I\, \mathcal{T}e^{-i\int_0^t I_a \otimes H(s) + iF_a \otimes K(s)ds}|r) \otimes I = \mathcal{T}e^{\int_0^t A(s)ds}$ if the moment conditions $(l|F_a^k|r) = 1, \forall k \geq 0,$ are satisfied.
- Example: $(l|r) = 1$ and $F_a|r) = |r)$.
- ROM perspective: engineering a reservoir that reproduces $\mathcal{T}e^{\int_0^t A(s)ds}$ as an input/output map

EEE workflow

- The overall circuit



- The final quantum state: $\sum_j r_j |j\rangle \otimes |x(t)\rangle$.



- Algorithms for the unitary evolution
  - Operator-splitting, Dyson series, Magnus expansion, etc.
  - Qubitization, Q singular value transform, linear combination of unitaries.
  - All these algorithms lead to $O(\log N)$ complexity

Fulfilling the moment conditions

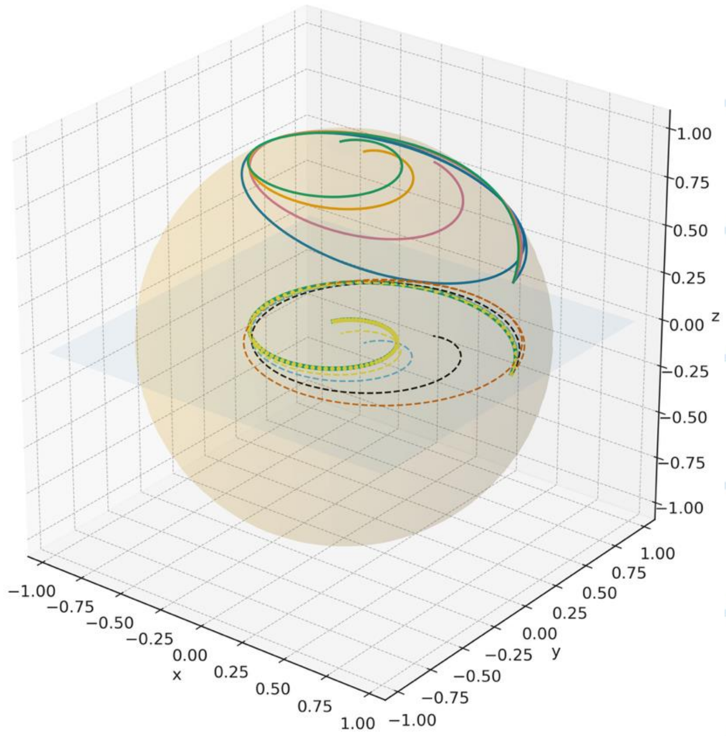
$$(l|F_a^k|r) = 1, \forall k \geq 0$$

- **Schrödingerization**:
  - $F = -\frac{\partial}{\partial p}$, skew Hermitian on $[0, +\infty)$ if $f(0) = 0.$
  - $|r) = e^{-p}, (l|f = e^{p_*} f(p_*).$ Moment conditions are satisfied.
  - $|r) \notin \mathcal{H}_a$. Numerical issue: minimize the boundary effect.
- Invariance under unitary transformation: $(l|U^{-1}, UF_aU^{-1}, U|r)$ still satisfy the moment condition.
- Let U be Fourier transform.
  - $U|r) = \frac{1}{k+i}, \; U \circ F \circ U^{-1} = ik, (l|U^{-1}f = Res_{k=-i}\hat{f}(k)$
  - This is the LCHS
- ***Are there other choices? Many***

Space H_A	Dilation operators	Generator F	Right vector $\|r)$	Evaluation $(l\|$
$H^1(0,1)$	Differential	$\theta\left(p\partial_p + \frac{1}{2}\right)$	$p^{1/\theta - 1/2}$	$(l\|f = 2^{1/\theta-1/2} f(\frac{1}{2})$
$L^2(\mathbb{R})$	Integral	$(Ff)(p) = \int_{\mathbb{R}} pe^{-\theta\|p-q\|} f(q)dq$	$e^{a(\theta)p}$	$(l\|f = f(0)$
$L^2(\mathbb{R})$	Pseudo-differential	$-i(-\Delta)^\theta$	$e^{i\xi_0 x}, \xi_0 = e^{i\pi/(4\theta)}$	$(l\|f = f(0)$
$\mathscr{B}$	Bargmann–Fock	$\theta(a^\dagger - a)$	$\exp\left(\frac{z^2}{2} - \frac{z}{\theta}\right)$	$(l\|f = f(0)$
ℓ^2	Difference	$(Ff)_n = \theta(f_n - f_{n-1})$	$\{\lambda_\theta^n\}_{n\geq 0}, \lambda = \frac{\theta}{1+\theta}$	$(l\|f = f_0$

From moment fulfilling to universal approximation

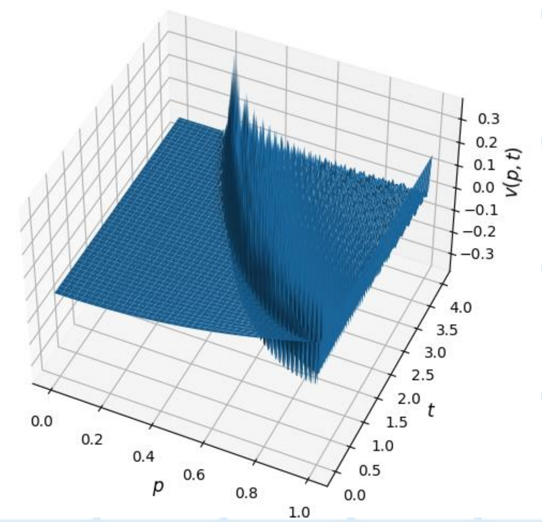
- These families of fulfilling operators ➔ Exact dilation methods
- $(l| \otimes I\, \mathcal{T}e^{-i\int_0^t I_a \otimes H(s) + iF_a \otimes K(s) ds} |r) \otimes I = \mathcal{T}e^{\int_0^t A(s) ds}$
- By ϵ approximating $|r)$ in the Hilbert space, we find ϵ approximations of $x(t)$ using TDSE
- ***How is this implemented for general ODEs?***



Differential operator on [0,1]

- $F_\theta = \theta\left(p\partial_p + \frac{1}{2}\right), |r) = p^{\frac{1}{\theta}-\frac{1}{2}}, (l|f = 2^{\frac{1}{\theta}-\frac{1}{2}} f\left(\frac{1}{2}\right)$. ($\theta$: tunable parameters to min complexity).
- Example: $u_t = au_x + u_{xx}$,
  - PBC on $[0,2\pi]$.
  - $A_0 = a\partial_x, A_1 = \partial_{xx}$.
  - Dilated system: $w(t,x,p)$: $\partial_t w = a\partial_x w + p\partial_{xxp} w + \frac{1}{2}\partial_{xx} w$. $w(0,x,p) = u(0,x)p^{\frac{1}{\theta}-\frac{1}{2}}, w(t,x,1) = 0$.
- Finite-difference with summation-by-parts (SBP) property
  - $F_\theta = \frac{\theta}{2}\{\partial_p, p\} \approx \theta F_h = \frac{\theta}{2}\{D_h, P\}$
  - Combined Hamiltonian $\widetilde{\mathrm{H}} = I \otimes H + i\theta F_h \otimes K \Rightarrow i\Psi = -i\,\widetilde{H}\Psi$
- Initial condition: $|r) \approx |r\rangle \propto \sum_j p_j^\beta |j\rangle, \beta = \frac{1}{\theta} - \frac{1}{2}$. $w(0,x,p) = |r\rangle \otimes |x_0\rangle$
- Boundary condition: $w(t,x,1) = 0$. To ensure that θF_h is skew Hermitian.

Finite speed of propagation property



- The transport equation from F: $u_t = -pu_p - \frac{1}{2}u,\ p \in (0,1)$.
- The error comes from the boundary effect.
- Method of characteristics: $\dot{p} = -p, \dot{u} = -\frac{1}{2}u$.
- Boundary effective arrives at a point p_* at time $t_* = \log\frac{1}{p_*}$.
- Finite speed propagation for the finite difference method.

Consider the dilated dynamics: $\psi_t = -i(I_A \otimes H + \theta F_h \otimes K)\psi,$ with initial condition supported at the boundary. Assume that $\eta = \frac{e\theta K_{max}t}{1-p_*} < 1.$ $h = \frac{1}{M}$ $and\ p_* = 1 - mh$. Then

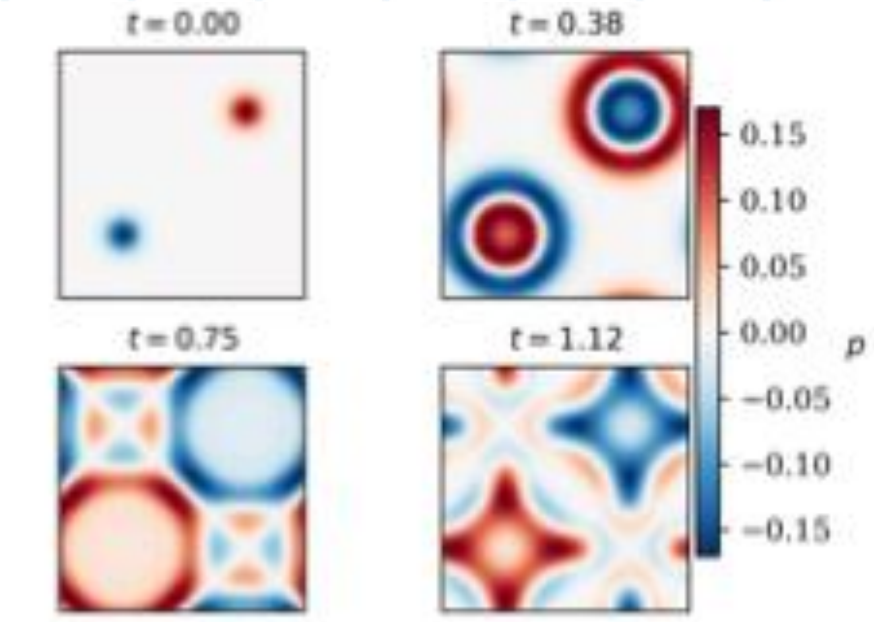
$$||\langle i| \otimes I\psi(t)|\psi_0\rangle|| < C\,\eta^m, \forall\ ih < p_*.$$

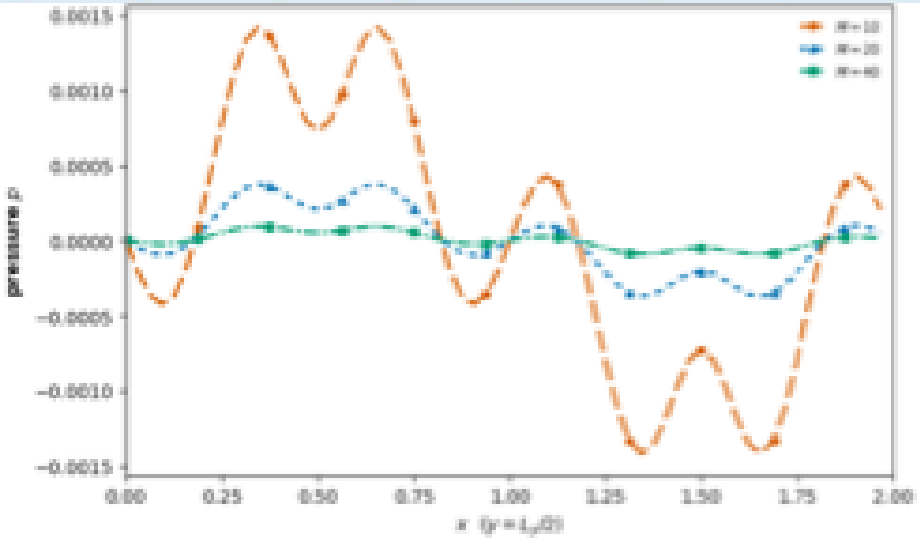
Therefore: $M = \Omega(\log\frac{1}{\epsilon})$ is enough to suppress the boundary effect

The boundary effect can be delayed by geometrically refined grids

Example

- Two-dimensional Maxwell Viscoelastic Wave equation
- Strain, momentum and stress (ε, p, σ) with viscous stress.





The convergence with the ancilla dimension.

Summary

- Linear differential equations are mostly quantum-easy (theoretically).
- The procedure is motivated by reduced-order modeling.
- Some nonlinear equations are quantum-easy. (smoothing/weakly nonlinear/no resonance)
- For general nonlinear equations: $e^{O(T)}O(\log N)$. Brüstle-Wiebe 2025.

Many remaining questions

- Implementation of dilation methods on near term device
- Integrate quantum algorithms with error mitigation schemes
- Improving the convergence radius for nonlinear problems.