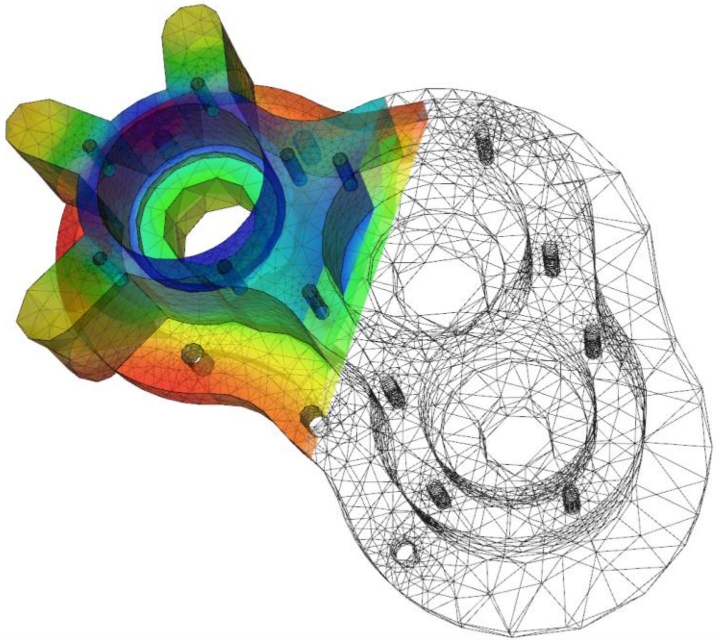
Quantum algorithms for time-dependent differential equations

DOMINIC W. BERRY

PEDRO COSTA

MACQUARIE University

2212.03544



Motivation

- Large systems of ODEs important for classical physics.
- Quantum computer can solve by encoding solution in amplitudes and applying linear equation solver [1] (Pedro's talk tomorrow).
- Our prior work showed how to solve time-*independent* ODEs [2].
- Our prior work also gave time-dependent Hamiltonian simulation with $\log(1/\epsilon)$ complexity [3].
- Here we provide an analogous result for time-dependent ODEs.

[1] P. C. S. Costa, D. An, Y. R. Sanders, Y. Su, R. Babbush, D. W. Berry, PRX Quantum **3**, 040303 (2022).
[2] D. W. Berry, A. M. Childs, A. Ostrander, and G. Wang, Communications in Mathematical Physics **356**, 1057 (2017).
[3] M. Kieferová, A. Scherer, D. W. Berry, Physical Review A **99**, 042314 (2019).

Formulating the problem

- General time-dependent ODE is of the form

$$\dot{\boldsymbol{x}}(t) = A(t)\boldsymbol{x}(t) + \boldsymbol{b}(t)$$

- $\boldsymbol{x}(t)$ and $\boldsymbol{b}(t)$ are vectors of length N
- $A(t)$ is $N \times N$ matrix

- We encode vector as amplitudes of quantum state

$$\propto \sum_{j=1}^{N} x_j(t)|j\rangle$$

- We have preparation of an initial $\boldsymbol{x}(t_0)$ and aim for solution at time t.

Dyson solution without driving

- Solution of $\dot{\boldsymbol{x}}(t) = A(t)\boldsymbol{x}(t)$ is $\boldsymbol{x}(t) = W(t,t_0)\boldsymbol{x}(t_0)$ with

$$W(t,t_0) = \sum_{k=0}^{\infty} \frac{1}{k!} \int_{t_0}^{t} dt_1 \int_{t_0}^{t} dt_2 \cdots \int_{t_0}^{t} dt_k \, \mathcal{T} A(t_1)A(t_2)\cdots A(t_k)$$

with time ordering, or

$$W(t,t_0) = \sum_{k=0}^{\infty} \int_{t_0}^{t} dt_1 \int_{t_1}^{t} dt_2 \cdots \int_{t_{k-1}}^{t} dt_k \, A(t_1)A(t_2)\cdots A(t_k)$$

- Similar to Hamiltonian evolution but not unitary.

Particular solution

- Solution of $\dot{\boldsymbol{x}}(t) = A(t)\boldsymbol{x}(t) + \boldsymbol{b}(t)$ with $\boldsymbol{x}(t_0) = 0$

$$\boldsymbol{v}(t, t_0) = \sum_{k=1}^{\infty} \frac{1}{k!} \int_{t_0}^{t} dt_1 \int_{t_0}^{t} dt_2 \cdots \int_{t_0}^{t} dt_k \, \mathcal{T} A(t_1) A(t_2) \cdots A(t_{k-1}) \boldsymbol{b}(t_k)$$

with $\boldsymbol{b}$ on right, or

$$\boldsymbol{v}(t, t_0) = \sum_{k=1}^{\infty} \int_{t_0}^{t} dt_1 \int_{t_1}^{t} dt_2 \cdots \int_{t_{k-1}}^{t} dt_k \, A(t_1) A(t_2) \cdots A(t_k) \boldsymbol{b}(t_k)$$

- Complete solution

$$\boldsymbol{x}(t) = W(t, t_0)\boldsymbol{x}(t_0) + \boldsymbol{v}(t, t_0)$$

Solution over long time as matrix

- Break into r time intervals:

$$\boldsymbol{x}(0) = \boldsymbol{x}_0$$
$$\boldsymbol{x}(\Delta t) = W(\Delta t, 0)\boldsymbol{x}(0) + \boldsymbol{v}(\Delta t, 0)$$
$$\boldsymbol{x}(2\Delta t) = W(2\Delta t, \Delta t)\boldsymbol{x}(\Delta t) + \boldsymbol{v}(2\Delta t, \Delta t)$$
$$\boldsymbol{x}(3\Delta t) = W(3\Delta t, 2\Delta t)\boldsymbol{x}(2\Delta t) + \boldsymbol{v}(3\Delta t, 2\Delta t)$$

etc...

- Rewrite as matrix

$$\begin{bmatrix} \mathbb{1} & 0 & 0 & 0 \\ -W(\Delta t, 0) & \mathbb{1} & 0 & 0 \\ 0 & -W(2\Delta t, \Delta t) & \mathbb{1} & 0 \\ 0 & 0 & -W(3\Delta t, 2\Delta t) & \mathbb{1} \end{bmatrix} \begin{bmatrix} \boldsymbol{x}(0) \\ \boldsymbol{x}(\Delta t) \\ \boldsymbol{x}(2\Delta t) \\ \boldsymbol{x}(3\Delta t) \end{bmatrix} = \begin{bmatrix} \boldsymbol{x}_0 \\ \boldsymbol{v}(\Delta t, 0) \\ \boldsymbol{v}(2\Delta t, \Delta t) \\ \boldsymbol{v}(3\Delta t, 2\Delta t) \end{bmatrix}$$

Quantum linear equation solver

- General form of linear equations

$$\boldsymbol{\mathcal{A}x} = \boldsymbol{b}$$

- Solve using quantum walk and filtering.

- Complexity $\propto \kappa \log(1/\epsilon)$

- Here solution gives

$$\sum_{m=0}^{r} |m\rangle \otimes \sum_{j=1}^{N} x_j(m\Delta t)|j\rangle$$

P. C. S. Costa, D. An, Y. R. Sanders, Y. Su, R. Babbush, D. W. Berry, PRX Quantum **3**, 040303 (2022).

Solution at final time

- Pad out matrix:

$$\begin{bmatrix} \mathbb{1} & 0 & 0 & 0 & 0 & 0 & 0 \\ -W(\Delta t, 0) & \mathbb{1} & 0 & 0 & 0 & 0 & 0 \\ 0 & -W(2\Delta t, \Delta t) & \mathbb{1} & 0 & 0 & 0 & 0 \\ 0 & 0 & -W(3\Delta t, 2\Delta t) & \mathbb{1} & 0 & 0 & 0 \\ 0 & 0 & 0 & -\mathbb{1} & \mathbb{1} & 0 & 0 \\ 0 & 0 & 0 & 0 & -\mathbb{1} & \mathbb{1} & 0 \\ 0 & 0 & 0 & 0 & 0 & -\mathbb{1} & \mathbb{1} \end{bmatrix}$$

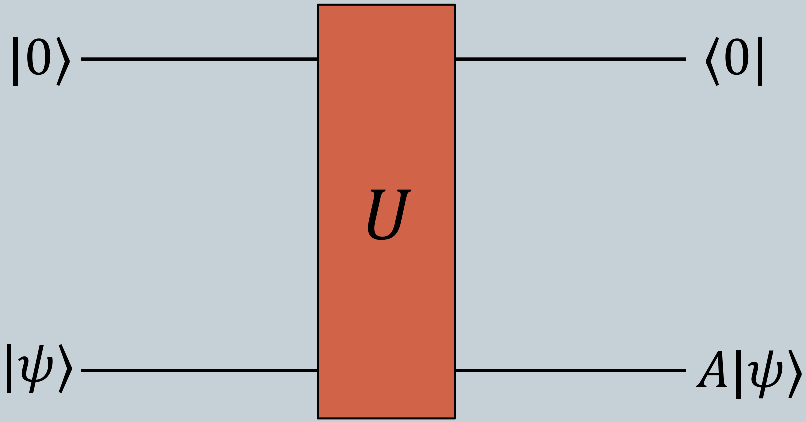
- Solution of form, with final $T = r\Delta t$

$$\sum_{m=0}^{r} |m\rangle \otimes \sum_{j=1}^{N} x_j(m\Delta t)|j\rangle + \sum_{m=r+1}^{2r} |m\rangle \otimes \sum_{j=1}^{N} x_j(T)|j\rangle$$

Block encoding

- Block encoding of form

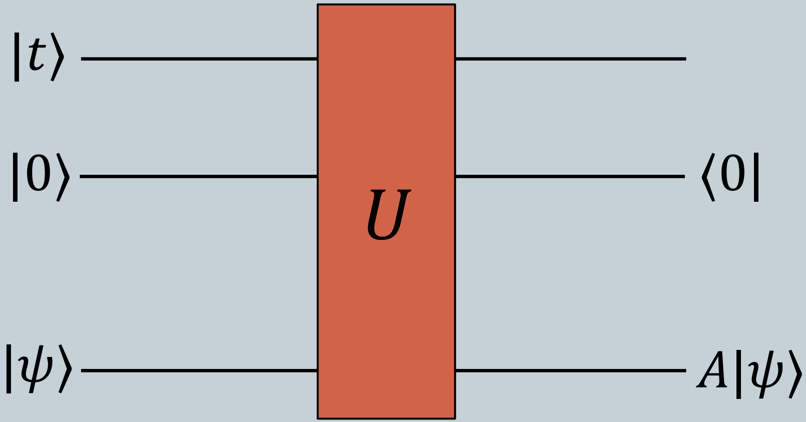
$$\langle 0|U|0\rangle = \frac{1}{\lambda}A$$



Block encoding

- Block encoding has time input:

$$\langle 0|U|0\rangle|t\rangle = \frac{1}{\lambda}A(t)|t\rangle$$



Block encoding matrix

- Truncate Dyson series:

$$W(t,t_0) \approx \sum_{k=0}^{K} \frac{1}{k!} \int_{t_0}^{t} dt_1 \int_{t_0}^{t} dt_2 \cdots \int_{t_0}^{t} dt_k \, \mathcal{T} A(t_1) A(t_2) \cdots A(t_k)$$

- Approximate integrals by sums

$$W(t,t_0) \approx \sum_{k=0}^{K} \frac{1}{k!} \sum_{j_1=0}^{M} \sum_{j_2=0}^{M} \cdots \sum_{j_k=0}^{M} \mathcal{T} A(t_{j_1}) A(t_{j_1}) \cdots A(t_{j_k})$$

1. First prepare superposition over $|k\rangle$.
2. Prepare equal superposition of $j_1, j_2, \dots, j_k$.
3. Quantum sort time registers.
4. Apply block encodings of $A(t)$.

Preparing state

- Similarly truncate and discretise $\boldsymbol{v}$:

$$\boldsymbol{v}(t, t_0) \approx \sum_{k=1}^{K} \frac{1}{k!} \sum_{j_1=0}^{M} \sum_{j_2=0}^{M} \cdots \sum_{j_k=0}^{M} \mathcal{T} A(t_{j_1}) A(t_{j_1}) \cdots A\left(t_{j_{k-1}}\right) \boldsymbol{b}(t_{j_k})$$

- We need to prepare complete state of the form

$$|0\rangle|x(0)\rangle + \sum_{m=1}^{r} |m\rangle|\boldsymbol{v}_m\rangle$$

- Difficulties arise if norm of $\boldsymbol{v}_m$ is small due to cancellations in time variation.

Complexity – condition number

- Complexity proportional to condition number κ.
- Norm of matrix is order of a constant.
- Inverse of matrix has simple form

$$\begin{bmatrix} \mathbb{1} & 0 & 0 & 0 & 0 & 0 & 0 \\ W(\Delta t,0) & \mathbb{1} & 0 & 0 & 0 & 0 & 0 \\ W(2\Delta t,0) & W(2\Delta t,\Delta t) & \mathbb{1} & 0 & 0 & 0 & 0 \\ W(3\Delta t,0) & W(3\Delta t,\Delta t) & W(3\Delta t,2\Delta t) & \mathbb{1} & 0 & 0 & 0 \\ W(3\Delta t,0) & W(3\Delta t,\Delta t) & W(3\Delta t,2\Delta t) & \mathbb{1} & \mathbb{1} & 0 & 0 \\ W(3\Delta t,0) & W(3\Delta t,\Delta t) & W(3\Delta t,2\Delta t) & \mathbb{1} & \mathbb{1} & \mathbb{1} & 0 \\ W(3\Delta t,0) & W(3\Delta t,\Delta t) & W(3\Delta t,2\Delta t) & \mathbb{1} & \mathbb{1} & \mathbb{1} & \mathbb{1} \end{bmatrix}$$

- Norm of inverse is proportional to number of steps r, so

$$\kappa \propto r$$

Complexity – choice of r

- For Dyson series we want $\lambda \Delta t \leq 1$.
- For total time T we need $r = T/\Delta t \propto \lambda T$.
- Complexity of solving linear equations is then
$$\kappa \log(1/\epsilon) \propto \lambda T \log(1/\epsilon)$$
- To amplify solution at final time need factor of
$$\frac{\max_t \|\boldsymbol{x}(t)\|}{\|\boldsymbol{x}(T)\|}$$
- Further factor comes from state preparation.

Final complexity

- For each step we need K calls to A, giving complexity proportional to

$$\lambda T \log\left(\frac{1}{\epsilon}\right) \log\left(\frac{\lambda T}{\epsilon}\right)$$

- For total gate complexity, main contribution is cost of sorting time registers

$$\lambda T \log\left(\frac{1}{\epsilon}\right) \log\left(\frac{\lambda T}{\epsilon}\right) M$$
$$\rightarrow \lambda T \log\left(\frac{1}{\epsilon}\right) \log\left(\frac{\lambda T}{\epsilon}\right) \log\left(\frac{TD}{\lambda \epsilon}\right)$$

- D depends on derivatives of A and $\boldsymbol{b}$.

Conclusions

- Complexity near-linear in time, logarithmic in allowable error:

$$\lambda T \log\left(\frac{1}{\epsilon}\right) \log\left(\frac{\lambda T}{\epsilon}\right)$$

- Replicates excellent scaling from Hamiltonian simulation.

- Approach also simplifies simulation in time-independent case.

- Difficulty of approach is ensuring state preparation works.

D. W. Berry, P. C. S. Costa, arXiv: 2212.03544.