Adaptive Online Learning of Quantum States

arXiv:2206.00220

joint work with Xinyi Chen, Elad Hazan, Zhou Lu, Xinzhao Wang, and Rui Yang

Tongyang Li

Center on Frontiers of Computing Studies, Peking University

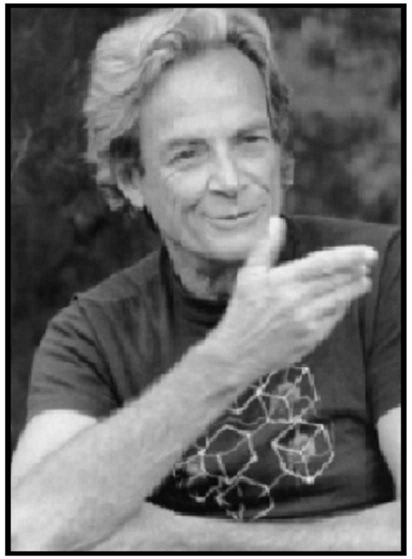
Quantum Computer Systems Lecture, August 11, 2022

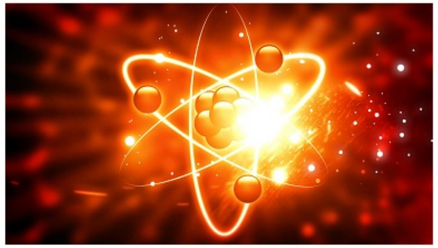
北京大学前沿计算研究中心
Center on Frontiers of Computing Studies, Peking University

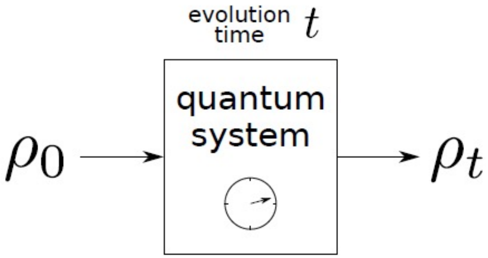
PRINCETON UNIVERSITY

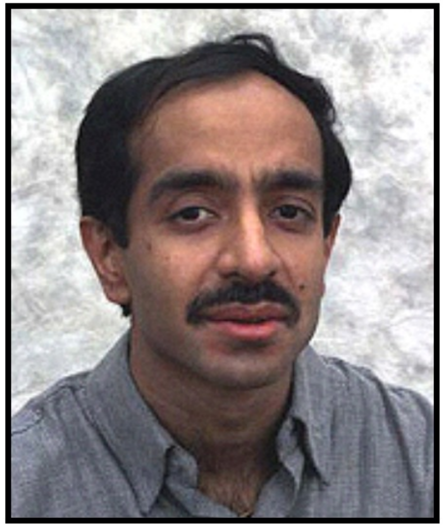
Why should we care about quantum computing?

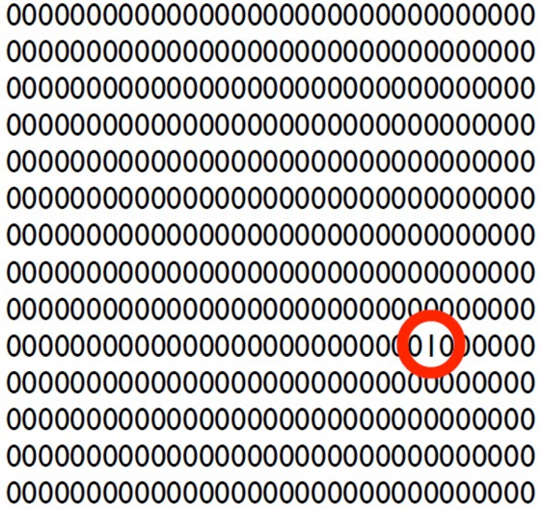
Fast algorithms for classically hard problems











3107418240490043721350750035888567
9300373460228427275457201619488232
0644051808150455634682967172328678
2437916272838033415471073108501919
5485290073377248227835257423864540
14691736602477652346609
=
163473364580925384844313388386509
085984178367003309231218111085238
93331001045081512121181675111579
×
190087128166482211312685157393541
397547189678996851549366663853908
8027103802104498957191261465571

- Linear systems
- Graph problems (minimum spanning tree, connectivity, shortest path, triangle finding, etc.)
- Formula evaluation
- Decomposing groups (abelian, dihedral, etc.)
- ……

Current quantum algorithm research: mostly Boolean and integer domains. Widely open even for **basic questions** in optimization and machine learning.

My research: Bridge between quantum computing and optimization/machine learning.

Theory: Quantum speedups of optimization and machine learning

- Unstructured problems: general function evaluations
  - Convex optimization [QIP'19]
  - Volume estimation [QIP'20]
  - Escaping from saddle points [QIP'21] [NeurIPS'21]
- Structured problems: data stored as matrices
  - Semidefinite programs [QIP'19]
  - Classification [ICML'19]
  - Matrix games [AAAI'21]
  - Quantum-inspired classical algorithms [QIP'20] [STOC'20]

[Brandao, Kalev, L., Lin, Svore, Wu, QIP 2019] arXiv:1710.02581
[Chakrabarti, Childs, L., Wu, QIP 2019] arXiv:1809.01731
[L., Chakrabarti, Wu, ICML 2019] arXiv:1904.02276
[Chakrabarti, Childs, Hung, L., Wang, Wu, QIP 2020] arXiv:1908.03903
[Chia, Gilyen, L., Lin, Tang, Wang, QIP 2020, STOC 2020] arXiv:1910.06151
[Zhang, Leng, L., QIP 2021, Quantum 5:529 (2021)] arXiv:2007.10253
[L., Wang, Chakrabarti, Wu, AAAI 2021] arXiv:2012.06519
[Zhang, L., NeurIPS 2021] arXiv:2111.14069

Question: How can quantum algorithm research be relevant in near term?

My effort

- Study quantum algorithms for practical problems. This talk: Learn quantum states.
- Basic problem in quantum computing, while machine learning can play a major role.

In experiments, we can generate copies of a quantum state ρ. But what is it?

If we are preparing for a specific state, is the prepared state closed to our goal?

Quantum state tomography: How many copies of a d-dimensional density matrix ρ are necessary to output a state ρ' with trace distance error ε to ρ?

Haah et al., O'Donnell and Wright, both in STOC'16: $\Theta(d^2/\varepsilon^2)$ (up to poly-log factors)

For n-qubit quantum states, this gives a 2^{2n} factor, intractable in general. Current experiments can only do ~10 qubits.

Shadow Tomography

An alternative goal: Given a group of measurement operators $E_1, \ldots, E_m$, construct a state ρ' such that Tr[ρ'E_i] is ε-close to Tr[ρE_i] for all i.

Originally proposed by Aaronson, state-of-the-art $\tilde{O}((\log^2 m)(\log d)/\varepsilon^4)$ by Badescu and O'Donnell in STOC'21. Exponentially better in d.

However, important practical concerns still exist:

1. Shadow tomography algorithms require a joint measurement on copies of ρ.
2. In general, measurement operators E_i can be given sequentially.
3. Quantum states have fluctuations, can change over time.

Observation:

- ➢Points 1 + 2 aligns well with online learning.
- ➢Point 3 essentially looks for adaptivity.

Aaronson, Chen, Hazan, Kale, and Nayak,
Online Learning of Quantum States [NeurIPS'18]

Adaptive Online Learning of Quantum States

Denote the set of n-qubit density matrices:

$$C_n := \{M \in \mathbb{C}^{2^n \times 2^n}, M = M^\dagger, M \succeq 0, \mathrm{Tr}(M) = 1\}.$$

Standard (static) regret:

$$\text{Regret} = \sum_{t=1}^{T} \ell_t(x_t) - \min_{\varphi \in C_n} \sum_{t=1}^{T} \ell_t(\varphi).$$

The loss function can be $\ell_t(x) = |\mathrm{Tr}(E_t x_t) - \mathrm{Tr}(E_t \rho_t)|$ or $\ell_t(x) = (\mathrm{Tr}(E_t x_t) - \mathrm{Tr}(E_t \rho_t))^2$ for example; $\rho_t = \rho$ for all t for the static case. The loss functions are L-Lipschitz.

Dynamic regret:

$$\text{Regret}_{T,\mathcal{P}}^{\mathcal{A}} = \sum_{t=1}^{T} \ell_t\left(\mathrm{Tr}(E_t x_t^{\mathcal{A}})\right) - \min_{\varphi_t \in C_n} \sum_{t=1}^{T} \ell_t\left(\mathrm{Tr}(E_t \varphi_t)\right), \text{ where } \mathcal{P} = \sum_{t=1}^{T-1} \|\varphi_t - \varphi_{t+1}\|_*.$$

Adaptive Online Learning of Quantum States

The $\mathcal{P}$ here is called the path length, and $\|\cdot\|_*$ is the ℓ_1 norm of singular values.

Another common situation: The dynamics is from M possible dynamical models $\{\Phi^{(1)}, \ldots, \Phi^{(M)}\}$, each of which being a quantum channel.

Example: $\Phi(\rho) = e^{iHt}\rho e^{-iHt}$, where H is a background Hamiltonian that evolves the quantum state following the Schrödinger equation.

$$\mathcal{P}' = \min_{\Phi^{(i)} \in \{\Phi^{(1)},\ldots,\Phi^{(M)}\}} \sum_{t=1}^{T-1} \|\varphi_{t+1} - \Phi^{(i)}(\varphi_t)\|_*.$$

Dynamic Regret

$$\text{Regret}_{T,\mathcal{P}}^{\mathcal{A}} = \sum_{t=1}^{T} \ell_t \left(\text{Tr}(E_t x_t^{\mathcal{A}})\right) - \min_{\varphi_t \in C_n} \sum_{t=1}^{T} \ell_t \left(\text{Tr}(E_t \varphi_t)\right).$$

Result 1: For the path-length setting,

$$\text{Regret}_{T,\mathcal{P}}^{\mathcal{A}} = O\left(L\sqrt{T(n+\log(T))\mathcal{P}}\right).$$

Result 2: For the M dynamical model setting,

$$\text{Regret}_{T,\mathcal{P}'}^{\mathcal{A}} = O\left(L\sqrt{T(n+\log(T))\mathcal{P}'} + \sqrt{T\log(M\log(T))}\right).$$

Result 3: If the dynamical models are ℓ-local quantum channels (influence $\leq \ell$ qubits),

$$\text{Regret}_{T,\mathcal{P}'}^{\mathcal{A}} = \tilde{O}\left(L\sqrt{T(n+\log(T))\mathcal{P}'} + 2^{6\ell}\sqrt{T}\right).$$

Adaptive Regret

$$\text{SA-Regret}_T^{\mathcal{A}}(\tau) = \max_{I\subseteq[T],|I|=\tau} \left(\sum_{t\in I} \ell_t\left(\mathrm{Tr}(E_t x_t^{\mathcal{A}})\right) - \min_{\varphi\in C_n} \sum_{t\in I} \ell_t\left(\mathrm{Tr}(E_t\varphi)\right) \right).$$

Standard regret takes $\tau = 1$. We have:

Result 4: $\text{SA-Regret}_T^{\mathcal{A}}(\tau) = O(L\sqrt{n\tau\log(T)})$.

Result 5: Consider the k-shift setting where we allow ρ to change at most k times. We have $R_{k\text{-shift}}^{\mathcal{A}} = O(L\sqrt{knT\log(T)})$ where

$$R_{k\text{-shift}}^{\mathcal{A}} = \sum_{t=1}^{T} \ell_t(E_t x_t^{\mathcal{A}}) - \min_{1=t_1\le t_2\le\cdots\le t_{k+1}=T+1}\ \min_{\varphi_1,\varphi_2,\ldots,\varphi_k\in C_n} \sum_{j=1}^{k}\sum_{t=t_j}^{t_{j+1}-1} \ell_t(E_t\varphi_j).$$

Main Results

Regret	Reference	Setting	O-Bound
Dynamic	Theorem 1	path length	$L\sqrt{T(n+\log(T))\mathcal{P}}$
Dynamic	Corollary 2	Family of M dynamical models	$L\sqrt{T(n+\log(T))\mathcal{P}'}$ $+\sqrt{T\log(M\log(T))}$
Dynamic	Corollary 3	l-local quantum channels	$L\sqrt{T(n+\log(T))\mathcal{P}'}+2^{6l}\sqrt{T}$
Adaptive	Theorem 2	arbitrary interval with length τ	$L\sqrt{n\tau\log(T)}$
Adaptive	Corollary 4	k-shift	$L\sqrt{knT\log(T)}$

Table 1: Adaptive and dynamic regret bounds of online learning of quantum states. Here n is the number of qubits, L is the Lipschitzness of loss functions, T is the total number of iterations, $\mathcal{P}$ is the path length defined in (1), $\mathcal{P}'$ in Corollary 2 is a variant of path length defined in (5) and $\mathcal{P}'$ in Corollary 3 is defined in (14). l-local quantum channels are explained in Section 3.

"Follow the Leader"

In online learning of minimizing $\text{Regret} = \sum_{t=1}^{T} \ell_t(x_t) - \min_{\varphi \in C} \sum_{t=1}^{T} \ell_t(\varphi),$

the most straightforward approach is to use at any time the optimal decision:

$$x_{t+1} = \arg\min_{\varphi \in C} \sum_{\tau=1}^{t} \ell_\tau(\varphi).$$

This flavor of strategy is known as "fictitious play" in economics, and has been named "Follow the Leader" in machine learning. However, it can fail.

Assume $C = [-1, 1]$, $f_1(x) = -0.5x$, $f_2(x) = x$, $f_3(x) = -x, \ldots$

$\text{Regret} = \Theta(T).$

$x_2 = \min_{x \in C} f_1(x) = 1$, $x_3 = \min_{x \in C} f_1(x) + f_2(x) = -1, \ldots$

Regularized Follow the Leader

People fix this by adding a regularizer, which increases stability:

$$x_{t+1} = \arg\min_{\varphi \in C} \left(\eta \sum_{\tau=1}^{t} \nabla \ell_\tau(\varphi) + R(\varphi) \right).$$

In terms of gradient descent methods, this lifts GD to MD (mirror descent).

In our paper, we use OMD with von Neumann entropy being the regularizer.

Online Mirror Descent for Learning Quantum States

Algorithm 2: OMD for Quantum Tomography

1: **Input:** domain $\mathcal{K} = (1 - \frac{1}{T})\mathcal{C}_n + \frac{1}{T2^n} I$, step size $\eta < \frac{1}{2L}$.
2: Define $R(x) = \sum_{k=1}^{2^n} \lambda_k(x) \log \lambda_k(x)$, $\nabla R(x) := I + \log(x)$, and let B_R denote the Bregman divergence defined by R.
3: Set $x_1 = 2^{-n} I$, and y_1 to satisfy $\nabla R(y_1) = \mathbf{0}$.
4: **for** $t = 1, \ldots, T$ **do**
5:     Predict x_t and receive loss $\ell_t(\mathrm{Tr}(E_t x_t))$.
6:     Define $\nabla_t = \ell_t'(\mathrm{Tr}(E_t x_t)) E_t$, where $\ell_t'(y)$ is a subderivative of ℓ_t with respect to y.
7:     Update y_{t+1} such that $\nabla R(y_{t+1}) = \nabla R(x_t) - \eta \nabla_t$.
8:     Update $x_{t+1} = \mathrm{argmin}_{x \in \mathcal{K}} B_R(x || y_{t+1})$.
9: **end for**

Line 1: A small shift to make the gradients bounded (derivative of xlog x is 1+log x).

Line 2: Von Neumann entropy as the regularizer. Advantage: avoid 2^n dependence

Line 5: make prediction and obtain the loss function

Line 6-8: the standard gadget for mirror descent

Remained Question: How to choose the step size?

Multiplicative Weight Update

We don’t know the path length P ahead of time, but the learning rate is a function of P.

Fix: We instantiate copies of OMD algorithms with different learning rates as experts, and uses the multiplicative weight (MW) algorithm to select the best expert.

Algorithm 1: Dynamic Regret for Quantum Tomography

1: **Input:** a candidate set of η, $S = \{2^{-k-1} \mid 1 \leq k \leq \log T\}$, constant α.
2: Initialize k experts $E_1, \ldots, E_k$, where E_k is an instance of Algorithm 2 with $\eta = 2^{-k-1}$.
3: Set initial weights $w_1(k) = \frac{1}{\log T}$.
4: **for** $t = 1, \ldots, T$ **do**
5: Predict $x_t = \frac{\sum_k w_t(k) x_t(k)}{\sum_k w_t(k)}$, where $x_t(k)$ is the output of the k-th expert.
6: Observe loss function $\ell_t(\cdot)$.
7: Update the weights as

$$w_{t+1}(k) = w_t(k) e^{-\alpha \ell_t(x_t(k))}.$$

8: Send gradients $\nabla \ell_t(x_t(k))$ to each expert E_k.
9: **end for**

Line 5-7: MW update, a standard technique in online learning (the smaller loss, the larger weight).

Technical Contributions

- Compared to classical state-of-the-art online learning algorithm for dynamic regret, we circumvent 2^n dependence by OMD with von Neumann entropy regularizer.
- Compared to the NeurIPS'18 online learning of quantum states paper, our framework OMD + MW also works with dynamic regret.

In analysis:

- Generalized Pythagorean Theorem for Hermitian matrices
- A new derivation of the Bregman divergence in analyzing OMD (complex variables)
- Bound the difference of the von Neumann entropy between two quantum states

Adaptive regret

Use a meta-algorithm called Coin Betting for Changing Environment (CBCE) proposed by Jun et al. [AISTATS' 17] with its black-box algorithm being the regularized follow-the-leader algorithm.

Mistake Bound

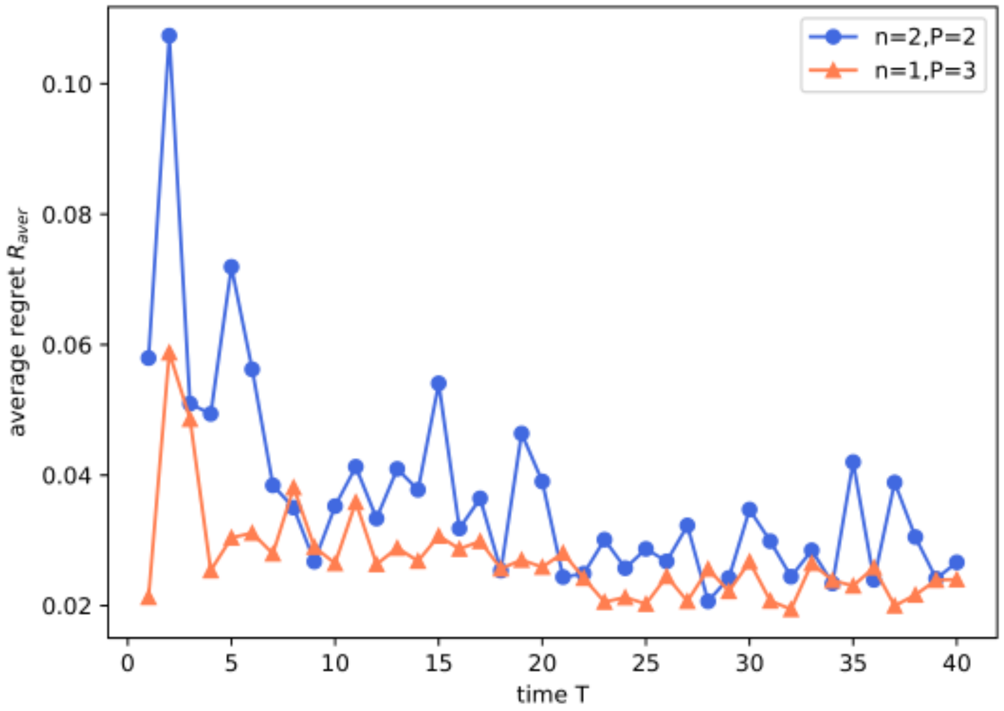
Corollary 1 (Path length setting). *Let $\{\rho_t\}$ be a sequence of n-qubit mixed state whose path length*

$$P_T = \sum_{t=1}^{T-1} \|\rho_{t+1} - \rho_t\|_* \leq \mathcal{P},$$

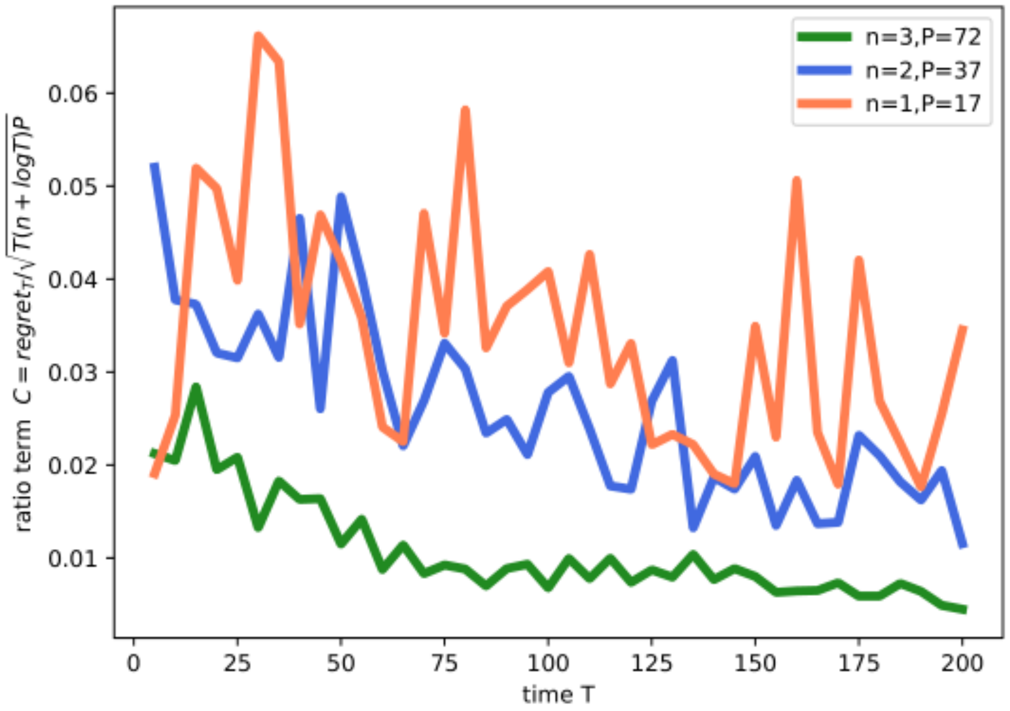
and let $E_1, E_2, \ldots$ be a sequence of two-outcome operators revealed to the learner one by one, each followed by a value $b_t \in [0,1]$ such that $|\operatorname{Tr}(E_t\rho_t) - b_t| \leq \frac{1}{3}\epsilon$. Then there is an explicit strategy for outputting hypotheses $x_1, x_2, \ldots$ such that $|\operatorname{Tr}(E_t x_t) - \operatorname{Tr}(E_t\rho_t)| > \epsilon$ for at most $\tilde{O}(\frac{L^2 n\mathcal{P}}{\epsilon^2})$ times.

Corollary 5 (k-shift setting). *Let $\{\rho_t\}$ be a sequence of n-qubit mixed state which changes at most k times, and let $E_1, E_2, \ldots$ be a sequence of two-outcome measurements that are revealed to the learner one by one, each followed by a value $b_t \in [0,1]$ such that $|\operatorname{Tr}(E_t\rho_t) - b_t| \leq \frac{1}{3}\epsilon$. Then there is an explicit strategy for outputting hypothesis states $x_1, x_2, \ldots$ such that $|\operatorname{Tr}(E_t x_t) - \operatorname{Tr}(E_t\rho_t)| > \epsilon$ for at most $O\left(\frac{kn}{\epsilon^2}\log\left(\frac{kn}{\epsilon^2}\right)\right)$ times.*

Numerical Experiments



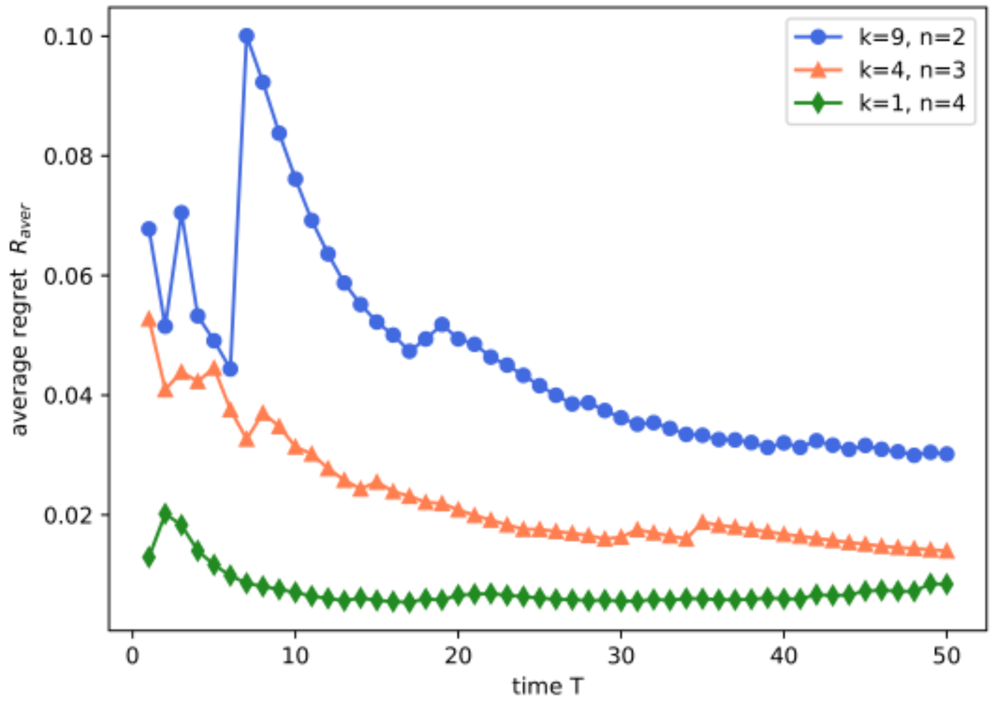
(a) Average Regret R_{aver}



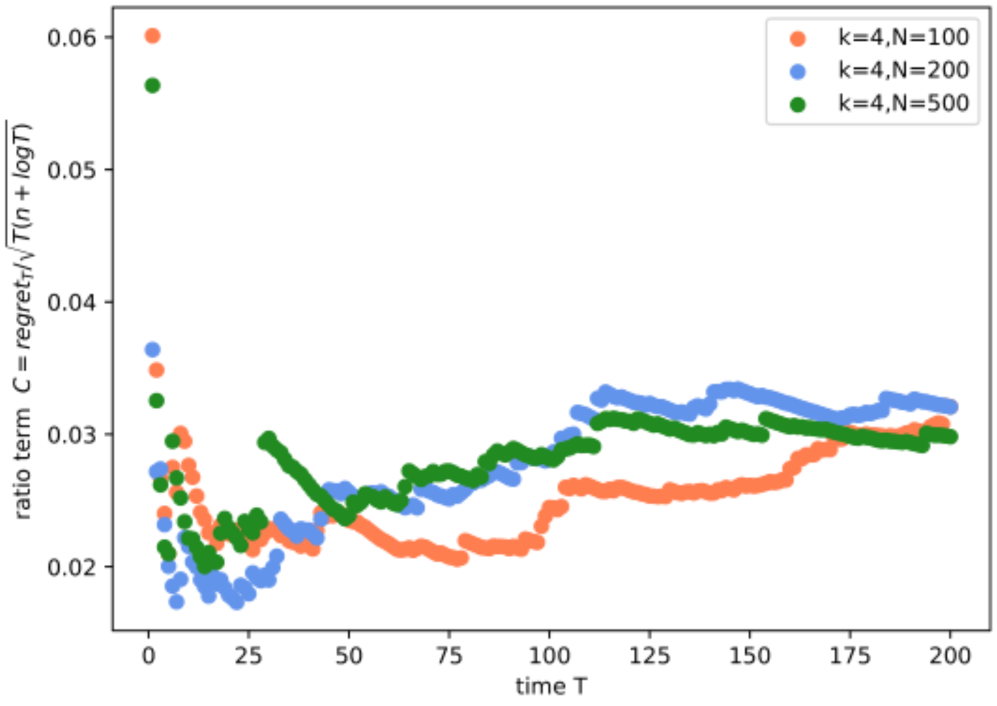
(b) Ratio $C = R_{\text{path}} / \sqrt{T(n + \log T)\mathcal{P}}$

Dynamic regret bound on the path length setting.

Numerical Experiments



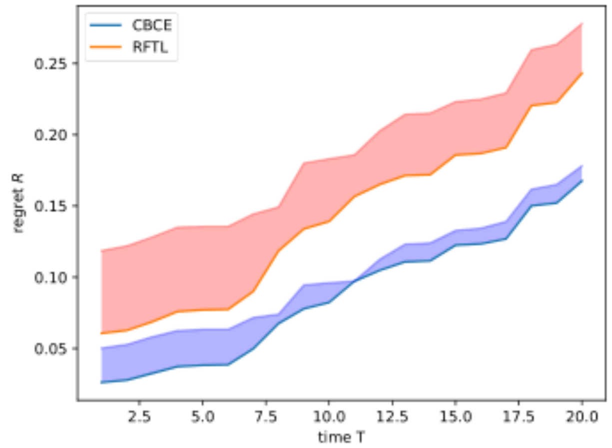
(a) Average Regret R_{aver}



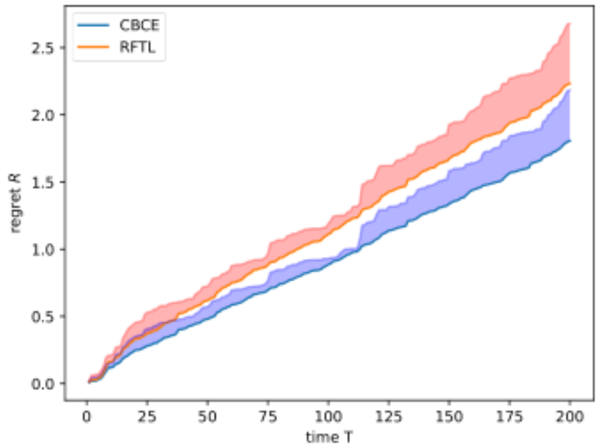
(b) Ratio $C = R_{k\text{-shift}}/\sqrt{T \log T}$

Adaptive regret bound on the k-shift setting.

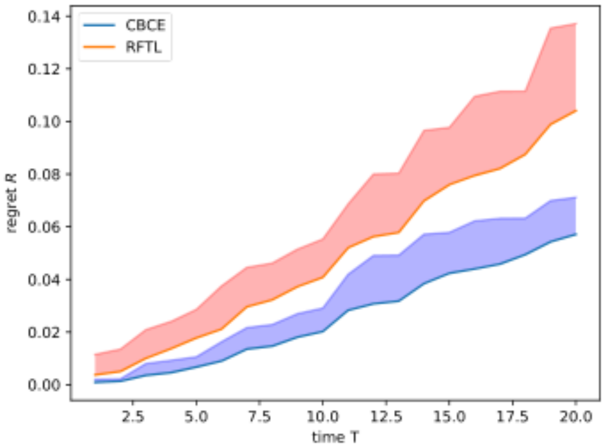
Numerical Experiments



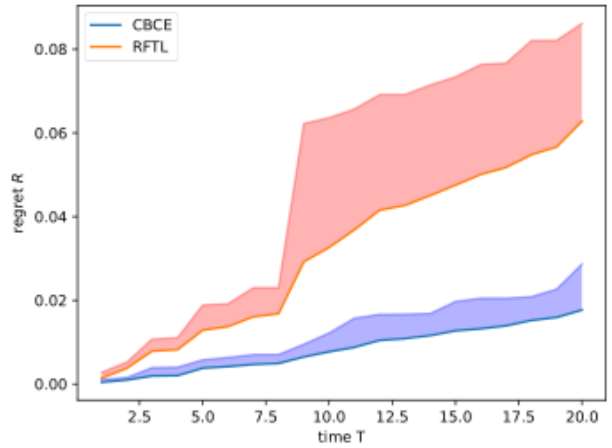
(a) k-shift setting, $k = 4$, $n = 2$.



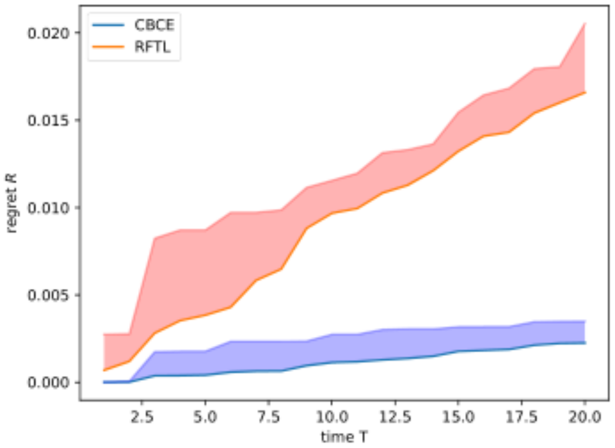
(b) k-shift setting, $k = 80$, $n = 2$.



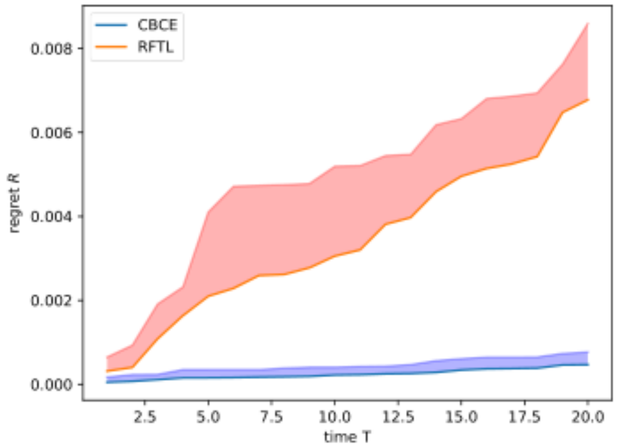
(c) k-shift setting, $k = 4$, $n = 3$.



(d) k-shift setting, $k = 4$, $n = 4$.



(e) k-shift setting, $k = 4$, $n = 5$.



(f) k-shift setting, $k = 4$, $n = 6$.

Comparison between our adaptive Algorithm 3 and the non-adaptive algorithm RFTL.

Deep Thinking

Online learning $\underset{\text{This direction?}}{\overset{\text{this work}}{\rightleftarrows}}$ **Quantum Computing**

Can we achieve quantum speedup in regret for (classical) online learning problems?

Ongoing projects:

- Multi-armed bandits and stochastic linear bandits
- Stochastic convex bandits

Multi-armed Bandits

Classical multi-armed bandits: For arm x, with probability $P_x(\omega)$ we obtain output ω and reward $y^x(\omega)$. Example: $\Omega = \{\text{apple,orange,banana}\}$, $P = (0.2, 0.3, 0.5)$, $y = (1, 2, 3)$.

$$\text{Quantum:} \quad \mathcal{O}_x\colon |0\rangle \to \sum_{\omega \in \Omega_x} \sqrt{P_x(\omega)}|\omega\rangle|y^x(\omega)\rangle$$

Regret: $R(T) = \sum_{t=1}^{I} \left(\mu(i^*) - \mu(i_t)\right)$, where i^* is the arm with the largest expected reward and i_t is the arm we pulled in round t.

Stochastic linear bandit: $\mu(x) = x^\top \theta^*$ where $\theta^* \in \mathbb{R}^d$ determines the mean reward.

Multi-armed Bandits

Model	Reference	Setting	Assumption	Regret
MAB	[5, 6, 16]	Classical	sub-Gaussian	$\Theta(\sqrt{nT})$
MAB	Theorem 1	Quantum	bounded value	$O(n\log(T))$
MAB	Theorem 2	Quantum	bounded variance	$O(n\log^{5/2}(T)\log\log(T))$
SLB	[1, 11, 16, 20]	Classical	sub-Gaussian	$\widetilde{\Theta}\left(d\sqrt{T}\right)$
SLB	Theorem 3	Quantum	bounded value	$O(d^2\log^{5/2}(T))$
SLB	Theorem 4	Quantum	bounded variance	$O(d^2\log^4(T)\log\log(T))$

Table 1: Regret bounds on multi-armed bandits (MAB) and stochastic linear bandits (SLB).

Method: Quantum Monte Carlo (based on amplitude estimation)

- *If* $y \in [0,1]$, *there is a constant* $C_1 > 1$ *and a quantum algorithm* $\mathtt{QMC}_1(\mathcal{O}, \epsilon, \delta)$ *which returns an estimate* $\hat{y}$ *of* $\mathbb{E}[y]$ *such that* $\Pr[|\hat{y} - \mathbb{E}[y]| \geq \epsilon] \leq \delta$ *using at most* $\frac{C_1}{\epsilon}\log\frac{1}{\delta}$ *queries to* $\mathcal{O}$ *and* $\mathcal{O}^\dagger$.
- *If* y *has bounded variance, i.e., Var*$(y) \leq \sigma^2$, *then for* $\epsilon < 4\sigma$, *there is a constant* $C_2 > 1$ *and a quantum algorithm* $\mathtt{QMC}_2(\mathcal{O}, \epsilon, \delta)$ *which returns an estimate* $\hat{y}$ *of* $\mathbb{E}[y]$ *such that* $\Pr[|\hat{y} - \mathbb{E}[y]| \geq \epsilon] \leq \delta$ *using at most* $\frac{C_2\sigma}{\epsilon}\log_2^{3/2}(\frac{8\sigma}{\epsilon})\log_2(\log_2\frac{8\sigma}{\epsilon})\log\frac{1}{\delta}$ *queries to* $\mathcal{O}$ *and* $\mathcal{O}^\dagger$.

Roughly speaking, classical regret looks like $T \cdot \frac{1}{\sqrt{T}} = O(\sqrt{T})$ whereas quantum $T \cdot \frac{1}{T} = O(1)$.

[Wan, Zhang, L., Zhang, Sun, in submission] arXiv:2205.14988

Stochastic Convex Bandits

We can also give quantum algorithms for bandits in continuous domains.

Let $\mathcal{K} \subseteq \mathbb{R}^n$ be a convex body and $f \colon \mathcal{K} \to [0,1]$ be a convex function; $\mathcal{B}(0,1) \subseteq \mathcal{K} \subseteq \mathcal{B}(0,R)$.

An online learner and environment interact alternatively over T rounds. In round $t \in [T]$, the learner predicts $x_t \in \mathcal{K}$ and returns $f(x_t) + \xi_t$, where ξ_t is a sub-Gaussian random variable.

The learner aims to minimize the regret

$$\text{Regret}_T := \mathbb{E}\left[\sum_{t=1}^{T}(f(x_t) - f^*)\right], \quad \text{where} \quad f^* = \min_{x \in \mathcal{K}} f(x).$$

This is a difficult problem in classical online learning. Hazan and Li '16: $\tilde{O}(\exp(d^4)\sqrt{T})$; Bubeck et al. STOC'17: $\tilde{O}(d^{10.5}\sqrt{T})$; Lattimore and György COLT'21: $\tilde{O}(d^{4.5}\sqrt{T})$ (state-of-the-art).

[L., Zhang, in submission]

Stochastic Convex Bandits

Quantumly, we assume the stochastic evaluation oracle as follows:

$$O_f|x, y\rangle = |x\rangle \int_{\xi \in \mathbb{R}} \sqrt{p_x(\xi)} |f(x) + y + \xi\rangle \, \mathrm{d}\xi \quad \forall x \in \mathbb{R}^n, y \in \mathbb{R}.$$

Main Theorem. There is a quantum algorithm for which $\mathrm{Regret}_T = \tilde{O}(n^5 \log(T) \log(TR))$.

The algorithm is based on an algorithm for optimizing approximately convex functions by Belloni et al. in COLT'15, which applies simulated annealing with hit-and-run walks in $\mathcal{K}$.

- Quantum walk: Quantum speedup of the hit-and-run walk in $\mathcal{K}$.
- Quantum Monte Carlo method

[L., Zhang, in submission]

Deep Thinking

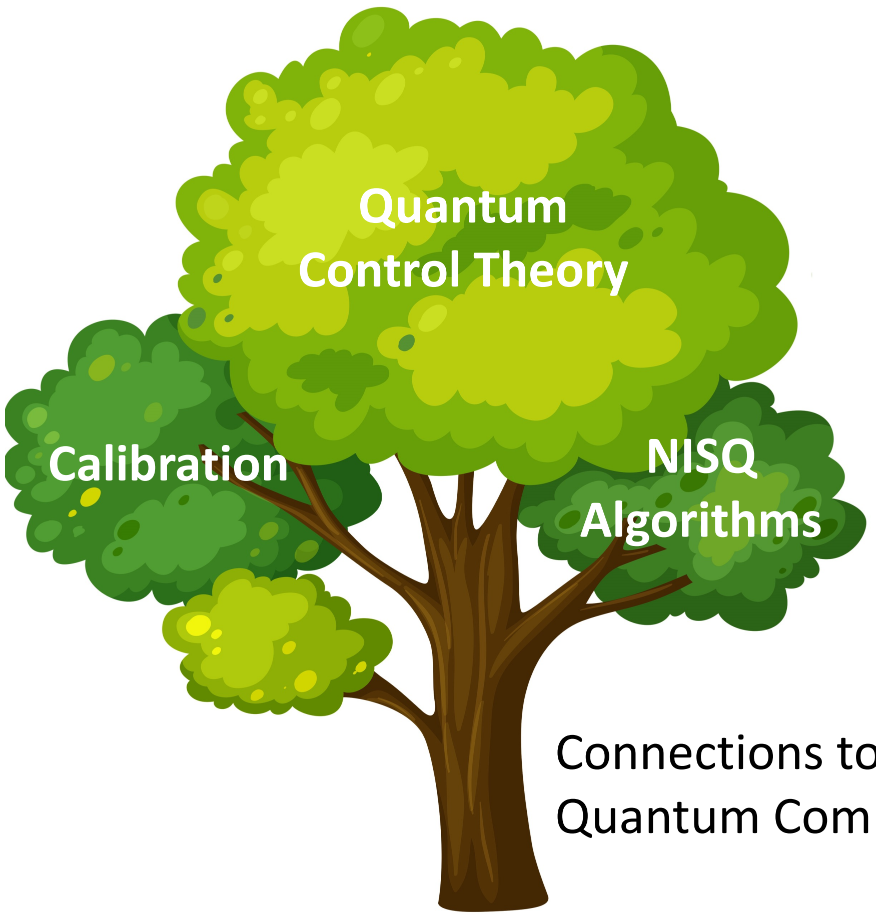
this work

Online learning $\rightleftarrows$ **Quantum Computing**

Multi-armed bandits

Stochastic linear bandits

Deeper Thinking



Connections to
Quantum Computer Systems

this work

Online learning ⇄ **Quantum Computing**

Multi-armed bandits
Stochastic linear bandits

Brainstorm: Quantum Control Theory

Applications to control theory has become a leading research trend in online learning.

See for instance the talk "Rethinking Control" at IAS by Elad Hazan (on YouTube).

Need to first figure out: How to formulate the problems in quantum computer systems as a control theory problem.

After that, tools from online learning may play a natural role (similar to this work).

Online learning $\rightleftarrows$ **Quantum Computing**

Quantum speedup of
control theory problems?

Brainstorm: Calibration

Our work suggests that changing states is not an issue for shadow tomography (learning measurement outcomes). In other words, post-processing by online learning promises robustness.

Does this help with calibration in real quantum computer systems?

Brainstorm: NISQ Algorithms

In the case that online mirror descent and multiplicative weight updates are not very available on current quantum computers, how about the NISQ counterparts?

In the quantum MAB paper, we demonstrated numerical experiments for its quantum speedup with IBM Qiskit, depolarizing noise with two-qubit error 1% and one-qubit error 0.3%.

Summary

We give online algorithms for learning changing quantum states by measurements. Our algorithms apply to various settings, including path length, family of dynamical models, k-shift, etc.

The other direction, quantum speedup for online learning, is also an interesting direction. Provable guarantee has been for various stochastic bandit problems in discrete and continuous domains.

We believe that this work may have further connections to quantum computer systems research, such as quantum control, calibration, NISQ algorithms, etc.

Thank you! Questions and comments: tongyangli@pku.edu.cn