# Quantum Multi-Armed Bandits and Stochastic Linear Bandits Enjoy Logarithmic Regrets*

**Zongqi Wan[1,2], Zhijie Zhang[†3], Tongyang Li[4,5], Jialin Zhang[1,2], Xiaoming Sun[1,2]**

[1]Institute of Computing Technology, Chinese Academy of Sciences
[2]University of Chinese Academy of Sciences
[3]Center for Applied Mathematics of Fujian Province, School of Mathematics and Statistics, Fuzhou University
[4]Center on Frontiers of Computing Studies, Peking University
[5]School of Computer Science, Peking University
wanzongqi20s@ict.ac.cn, zzhang@fzu.edu.cn, tongyangli@pku.edu.cn, zhangjialin@ict.ac.cn, sunxiaoming@ict.ac.cn

## Abstract

Multi-arm bandit (MAB) and stochastic linear bandit (SLB) are important models in reinforcement learning, and it is well-known that classical algorithms for bandits with time horizon $T$ suffer $\Omega(\sqrt{T})$ regret. In this paper, we study MAB and SLB with quantum reward oracles and propose quantum algorithms for both models with $O(\mathrm{poly}(\log T))$ regrets, exponentially improving the dependence in terms of $T$. To the best of our knowledge, this is the first provable quantum speedup for regrets of bandit problems and in general exploitation in reinforcement learning. Compared to previous literature on quantum exploration algorithms for MAB and reinforcement learning, our quantum input model is simpler and only assumes quantum oracles for each individual arm.

## Introduction

Bandits are a fundamental model in reinforcement learning applied to problems where an agent has a fixed set of choices and the goal is to maximize its gain, while each choice's properties are only partially known at the time of allocation but may become better understood as iterations continue (Lattimore and Szepesvári 2020; Sutton and Barto 2018). Bandits exemplify the exploration-exploitation tradeoff where exploration aims to find the best choice and exploitation aims to obtain as many rewards as possible. Bandits have wide applications in machine learning, operations research, engineering, and many other areas (Chapelle, Manavoglu, and Rosales 2014; Lei, Tewari, and Murphy 2017; Silver et al. 2016; Villar, Bowden, and Wason 2015).

In this paper, we investigate two important bandit models: multi-armed bandits and stochastic linear bandits. In the multi-armed bandit (MAB) problem, there are $n$ arms where each arm $i \in [n] := \{1, 2, \dots, n\}$ is associated with an unknown reward distribution. We denote the expected reward of arm $i$ as $\mu(i) \in [0, 1]$. MAB has $T$ rounds. At round $t = 1, 2, \dots, T$, the learner chooses an arm $i_t$ and receives a reward $y_t$, a random variable drawn from the reward distribution of $i_t$. Denote the best arm with the largest expected reward to be $i^*$. The goal is to minimize the cumulative regret with respect to the best arm $i^*$ over $T$ rounds:

$$R(T) = \sum_{t=1}^{T} \left(\mu(i^*) - \mu(i_t)\right). \tag{1}$$

In the stochastic linear bandit (SLB) problem, the learner can play actions from a fixed action set $A \subseteq \mathbb{R}^d$. There is an unknown parameter $\theta^* \in \mathbb{R}^d$ which determines the mean reward of each action[1]. The expected reward of action $x$ is $\mu(x) = x^\top \theta^* \in [0, 1]$. It is often assumed that the action $x$ and the $\theta^*$ have bounded $L^2$-norm. That is, for some parameters $L, S > 0$,

$$\|x\|_2 \le L \text{ for all } x \in A, \text{ and } \|\theta^*\|_2 \le S. \tag{2}$$

Let $x^* = \mathrm{argmax}_{x\in A}\, x^\top \theta^*$ be the action with the largest expected reward. Same as MAB, SLB also has $T$ rounds. In round $t$, the learner chooses an action $x_t \in A$ and observes some realization of the reward $y_t$. The goal is again to minimize the cumulative regret

$$R(T) = \sum_{t=1}^{T} (x^* - x_t)^\top \theta^*. \tag{3}$$

Regarding the assumption on the reward distributions for both settings, a common assumption is *$\sigma$-sub-Gaussian*. Suppose $y$ is a random reward of some action $x$, then for any $\alpha \in \mathbb{R}$, the noise $\eta = y - \mu(x)$ has zero mean and satisfies $\mathbb{E}[\exp(\alpha X)] \le \exp\left(\alpha^2\sigma^2/2\right)$. In this paper, we consider the *bounded value* assumption and the *bounded variance* assumption. The bounded value assumption requires all rewards to be in a bounded interval, and without loss of generality, we assume that the reward is in $[0, 1]$. The bounded variance assumption requires the variances of all reward distributions to have a universal upper bound $\sigma^2$. Note that the bounded value assumption is more strict

[*] The missing proofs in this paper can be found in arXiv:2205.14988

[†] Corresponding author

[1] The "action" is the same as "arm" throughout the paper.

than the sub-Gaussian assumption, and the sub-Gaussian assumption is more strict than the bounded variance assumption. Nevertheless, even for the bounded value assumption, regrets of classical algorithms for MAB and SLB suffer from $\Omega(\sqrt{nT})$ (Auer et al. 2002) and $\Omega(d\sqrt{T})$ (Dani, Hayes, and Kakade 2008) lower bounds, respectively. Fundamentally different models are required to overcome this $\Omega(\sqrt{T})$ bound and quantum computation comes to our aid.

In our quantum bandit models, we assume that each time we pull an arm, instead of observing an immediate reward as our feedback, we get a chance to access a quantum unitary oracle $\mathcal{O}$ or its inverse $\mathcal{O}^\dagger$ once, where $\mathcal{O}$ encodes the reward distribution of this arm,

$$\mathcal{O}\colon |0\rangle \mapsto \sum_{\omega\in\Omega} \sqrt{P(\omega)}|\omega\rangle|y(\omega)\rangle \tag{4}$$

where $|\cdot\rangle$ is the so-called Dirac notation used to denote the quantum states, we will introduce it in the next section. $y\colon \Omega \to \mathbb{R}$ is a random variable whose domain is a finite sample space, and $P$ is a probability measure of $\Omega$. Such an oracle plays the role of reward in our quantum bandit models, and it is a natural generalization of classical reward models. If we perform a measurement on $\mathcal{O}|0\rangle$ with the standard basis immediately after we invoke $\mathcal{O}$, we will observe a realization of the random reward $y$, reducing to the classical bandit models. When we consider a learning environment simulated by a quantum algorithm, the simulation directly gives the quantum oracle. Such a situation arises in many reinforcement learning settings where the learning agents are in an artificial environment including games AI, autopilot, etc.; see for instance (Dunjko, Taylor, and Briegel 2015, 2016).

Quantum computing is an emerging technology, and there is a surging interest in understanding quantum versions of machine learning algorithms (see for instance the surveys (Arunachalam and de Wolf 2017; Biamonte et al. 2017; Dunjko and Briegel 2018; Schuld, Sinayskiy, and Petruccione 2015)). For bandit problems, Casalé et al. (Casalé et al. 2020) initiated the study of quantum algorithms for best-arm identification of MAB, and Wang et al. (Wang et al. 2021b) proved optimal results for best-arm identification of MAB with Bernoulli arms. As an extension, Wang et al. (Wang et al. 2021a) proposed quantum algorithms for finding an optimal policy for a Markov decision process with quantum speedup. These results focused on the exploration of reinforcement learning models, and in terms of the tradeoff between exploration and exploitation, the only work we are aware of is (Lumbreras, Haapasalo, and Tomamichel 2022), which proved that the regret of online learning of properties of quantum states has lower bounds $\Omega(\sqrt{T})$. (He et al. 2022) studied the quantum algorithm of adversarial convex bandit which has better dependence on the dimension of action space compared to classical algorithms. As far as we know, the quantum speedup of the regret of bandit models is yet to explore.

**Contributions.** We initiate the study of quantum algorithms for bandits, including MAB and SLB. Specifically, we formulate the quantum counterparts of bandit models where the learner can access a quantum oracle which encodes the reward distributions of different arms. For both models, we propose quantum algorithms which achieve $\mathrm{poly}(\log T)$ regret. Our results are summarized in Table 1. Note that our regret bounds have a worse dependence on $n$ and $d$ because pulling every arm once already incurs $\Omega(n)$ regret, and as a result, the $\Omega(n)$ factor is inevitable in our current algorithm.

Technically, we adapt classical UCB-type algorithms by employing Quantum Monte Carlo method (Montanaro 2015) (QMC) to explore the arms. QMC has a quadratic better sample complexity compared with classical methods, and this is the essence of our quantum speedup of the regret bound. However, different from the classical empirical mean estimator, QMC cannot provide any classical information feedback before it measures its quantum state. This means our quantum algorithm cannot observe feedback each round. Moreover, a quantum state collapses when it is measured and can not be reused again. This unique problem introduced by quantum subroutine makes the direct combination of traditional UCB-framework and QMC fail to achieve logarithmic regret. To address this problem, we propose a novel *adaptive staging technique*, where our algorithms adaptively divide consecutive time slots into stages with carefully selected stage lengths. Arm switching and measurement only happen at the end of a stage. For MAB, our algorithm QUCB doubles the length of the stage whenever we select an arm which has been selected before. For SLB, to exploit the linear dependency of arms, we introduce weighted least square estimators to estimate the parameter $\theta^*$, this estimator is different from the unweighted least square estimator used in traditional UCB-type algorithms. Moreover, we propose a novel stage length which is closely related to the variance-covariance matrices of these weighted least square estimators. We see these novel adaptive staging techniques as our main technical contribution. Along with these techniques, we can combine the traditional UCB framework with QMC to achieve $O(\mathrm{poly}(\log T))$ regret in both Quantum MAB and Quantum SLB settings.

Finally, we corroborate our theoretical findings with numerical experiments. The results are consistent with our theoretical results, visually proving the quantum speedup of bandit problems. We also consider the presence of quantum noise, and discuss the effects of different levels of quantum noise.

## Preliminaries of Quantum Computation

**Basics.** A *quantum state* can be seen as a $L^2$-normalized column vector $\vec{x} = (x_1, x_2, \ldots, x_m)$ in Hilbert space $\mathbb{C}^m$. Intuitively, $\vec{x}$ is a superposition of $m$ classical states, and $|x_i|^2$ is the probability for having the $i$-th state. In quantum computation, people use the Dirac *ket* notation $|x\rangle$ to denote the quantum state $\vec{x}$, and denote $\vec{x}^\dagger$ by the *bra* notation $\langle x|$. Given two quantum states $|x\rangle \in \mathbb{C}^m$ and $|y\rangle \in \mathbb{C}^n$, we denote their tensor product by $|x\rangle|y\rangle := (x_1y_1, x_1y_2, \ldots, x_my_{n-1}, x_my_n)$.

To observe classical information from a quantum state, one can perform a *quantum measurement* on this quantum state. Usually, a POVM (positive operator-valued measure)

| Model | Reference | Setting | Assumption | Regret |
|---|---|---|---|---|
| MAB | (Auer, Cesa-Bianchi, and Fischer 2002) | 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 | (Abbasi-Yadkori, Pál, and Szepesvári 2011) | 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).

is used, which is a set of positive semi-definite Hermitian matrices $\{E_i\}_{i\in\Lambda}$ satisfying $\sum_{i\in\Lambda} E_i = I$, $\Lambda$ here is the index set of the POVM. After applying the POVM on $|x\rangle$, outcome $i$ is observed with probability $\langle x|E_i|x\rangle$. The assumption $\sum_{i\in\Lambda} E_i = I$ guarantees that all probabilities add up to 1.

A quantum algorithm applies unitary operators to an input quantum state. In many cases, information of the input instance is encoded in a unitary operator $\mathcal{O}$. This unitary is called a *quantum oracle* which can be used multiple times by a quantum algorithm.

**Quantum reward oracle.** We generalize MAB and SLB to their quantum counterparts, where we can exploit the power of quantum algorithms. Our quantum bandit problems basically follow the framework of classical bandit problems. There are also $T$ rounds. In every round, the learner must select an action, and the regret is defined as (1) and (3). For stochastic linear bandits, we also admit the bounded norm assumption (2). The main difference is that, in our quantum version, the immediate sample reward is replaced with a chance to access an unitary oracle $\mathcal{O}_x$ or its inverse encoding the reward distribution $P_x$ of the selected arm $x$. Formally, let $\Omega_x$ be the sample space of the distribution $P_x$. We assume that there is a finite[2] sample space $\Omega$ such that $\Omega_x \subseteq \Omega$ for all $x \in A$ (or for all $i \in [n]$ in the MAB setting). $\mathcal{O}_x$ is defined as follows ($\mathcal{O}_i$ is defined similarly by replacing $x$ with $i$):

$$\mathcal{O}_x \colon |0\rangle \mapsto \sum_{\omega\in\Omega_x} \sqrt{P_x(\omega)}|\omega\rangle|y^x(\omega)\rangle \tag{5}$$

where $y^x \colon \Omega_x \to \mathbb{R}$ is the random reward associated with arm $x$. We say $\mathcal{O}_x$ *encodes* probability measure $P_x$ and random variable $y^x$.

During a quantum bandit task, the learner maintains a quantum circuit. At round $t = 1, 2, \ldots, T$, it chooses an arm $x_t$ by invoking either of the unitary oracles $\mathcal{O}_{x_t}$ or $\mathcal{O}^\dagger_{x_t}$ at most once. After this, the immediate expected regret $\mu(x^*) - \mu(x)$ is added to the cumulative regret. The learner can choose whether to perform a quantum measurement at any round. During two successive rounds, it can place arbitrary unitaries in the circuit. We call the bandit problem equipped with the quantum reward oracle defined above the *Quantum Multi-armed Bandits* (QMAB) and *Quantum Stochastic Linear Bandits* (QSLB).

*Remark* 1. In previous papers (Casalé et al. 2020; Wang et al. 2021b) investigating the quantum best arm identification problem, they consider the MAB model with Bernoulli rewards using a stronger coherent query model allowing superposition between different arms. That is,

$$O\colon |i\rangle_I|0\rangle_B \mapsto |i\rangle_I(\sqrt{p_i}|1\rangle_B + \sqrt{1-p_i}|0\rangle_B) \tag{6}$$

where the state of the quantum register $I$ corresponds to arm $i$, and $p_i$ represents the mean rewards of arm $i$. This is a stronger oracle assumption compared with our assumption (5). Because our model only has $O_i$ for each arm separately and cannot entangle different arms. In another word, if we are given the oracle in (6), then we can construct oracles $\mathcal{O}_x$ in (5) by calling $\mathcal{O}$ with the register $I$ fixed at $x$. We adopt (5) because in regret minimization problem, one should emphasize the exploration-exploitation tradeoff, and we must bind the action of the learner and the feedback the learner gets in a single round. The coherent model is not suitable for our setting since it explores all arms together, while (5) makes more direct and fair comparisons to classical bandit models.

**Quantum Monte Carlo method.** To achieve quantum speedup for QMAB and QSLB, we use the Quantum Monte Carlo method (Montanaro 2015) stated below to estimate the mean rewards of actions.

**Lemma 1** (Quantum Monte Carlo method (Montanaro 2015))**.** *Assume that $y : \Omega \to \mathbb{R}$ is a random variable with bounded variance, $\Omega$ is equipped with a probability measure $P$, and the quantum unitary oracle $\mathcal{O}$ encodes $P$ and $y$.*

- *If $y \in [0, 1]$, there is a constant $C_1 > 1$ and a quantum algorithm 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 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*

[2] This assumption requires the reward noise distribution to have finite support, which is not stated explicitly in classical setups. However, when a classical bandit algorithm is running on a physical realization of the classical computer, any real number is represented in finite bits as a floating-point number. In this sense, no classical bandit algorithm can really sample from a distribution with infinite support. Therefore, this assumption is no more stringent in nature than that of the classical bandits.

*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* $\mathcal{O}^\dagger$.

Note that Lemma 1 demonstrates a quadratic quantum speedup in $\epsilon$ for estimating $\mathbb{E}[y]$ because classical methods such as the Chernoff bound take $O(1/\epsilon^2\log(1/\delta))$ samples to estimate $\mathbb{E}[y]$ within $\epsilon$ with probability at least $1-\delta$. This is a key observation utilized by our quantum algorithms.

## Quantum Multi-Armed Bandits

In this section, we present an algorithm called QUCB (Algorithm 1) for QMAB with $O(n\log T)$ regret. QUCB adopts the canonical upper confidence bound (UCB) framework, combined with Quantum Monte Carlo method. During its execution, it maintains three quantities for each arm $i$: an estimate $\hat{\mu}(i)$ of $\mu(i)$, a confidence radius $r_i$ such that $\mu(i)\in[\hat{\mu}(i)-r_i,\hat{\mu}(i)+r_i]$ with high probability. Besides, it maintains $N_i$, the stage length when the algorithm decides to pull arm $i$.

Classical UCB algorithms for MAB also maintain a confidence interval during the MAB game, the length of this confidence interval decreases as the number of times the corresponding arm is selected. To be exact, if an arm is selected for $N$ rounds, then the length of the confidence interval of its reward is $O\left(\frac{1}{\sqrt{N}}\right)$. Since we can obtain a quadratic speedup by using QMC to estimate the mean reward of an arm, the length of the confidence interval is expected to be improved to $O\left(\frac{1}{N}\right)$, then it will be enough to derive a $O(n\log T)$ regret bound. But the introduction of QMC leads to another problem, that is, before we measure the quantum state, we cannot observe any feedback. Moreover, if we measure the quantum state, then the state collapses. To solve this unique problem of quantum bandits, QUCB adaptively divides whole $T$ rounds into several stages and it only updates the confidence interval and switches its arm at the beginning of each stage. Specifically, in each stage $s$, it first chooses arm $i_s$ which has the largest $\mu(i_s)+r_{i_s}$, i.e., the right endpoint of the arm's confidence interval. Then, $r_{i_s}$ is reduced by half and $N_{i_s}$ is doubled, and the algorithm plays arm $i_s$ for next $N_{i_s}$ rounds. During this stage, QMC is invoked with $N_{i_s}$ queries to $\mathcal{O}_{i_s}$ to update a new estimation $\hat{\mu}(i_s)$ which has better accuracy. After having done all the above, the algorithm enters into the next stage. The algorithm terminates after it plays $T$ rounds. We show in Theorem 1 and Corollary 1 that Algorithm 1 achieves an $O(n\log T)$ expected cumulative regret.

**Theorem 1.** *Let* $C_1$ *be the constant specified in Lemma 1. Under the bounded value assumption, with probability at least* $1-n\delta\log_2\left(\frac{T}{nC_1\log\frac{1}{\delta}}\right)$*, the cumulative regret of QUCB*$_1(\delta)$ *satisfies* $R(T)\le 8(n-1)C_1\log\frac{1}{\delta}$.

*Proof.* At the end of each stage (including the initialization stage described in line 1-5 of Algorithm 1), by Lemma 1, QMC have enough queries to output an estimation $\hat{\mu}(i)$ such that

$$|\hat{\mu}(i)-\mu(i)|\le r_i \qquad (7)$$

holds with probability at least $1-\delta$ for any $i\in[n]$.

```
Algorithm 1: QUCB$_1(\delta)$
   Parameters: fail probability $\delta$
 1: for $i = 1 \to n$ do
 2:     $r_i \leftarrow 1$ and $N_i \leftarrow \frac{C_1}{r_i}\log\frac{1}{\delta}$
 3:     play arm $i$ for consecutive $N_i$ rounds
 4:     run QMC$_1(\mathcal{O}_i, r_i, \delta)$ to get an estimation $\hat{\mu}(i)$ for
        $\mu(i)$
 5: end for
 6: for each stage $s = 1, 2, \ldots$ (terminate when we have
    used $T$ queries to all $\mathcal{O}_i$) do
 7:     Let $i_s \leftarrow \mathrm{argmax}_i\ \hat{\mu}(i) + r_i$ (if argmax has multiple
        choices, pick an arbitrary one)
 8:     update $r_{i_s} \leftarrow r_{i_s}/2$ and $N_{i_s} \leftarrow \frac{C_1}{r_{i_s}}\log\frac{1}{\delta}$
 9:     Play $i_s$ for the next $N_{i_s}$ rounds, update $\hat{\mu}(i_s)$ by run-
        ning QMC$_1(\mathcal{O}_{i_s}, r_{i_s}, \delta)$
10: end for
```

For each arm $i$, let $\mathcal{S}_i$ be the set of stages when arm $i$ is played, and denote $|\mathcal{S}_i|=K_i$. Initial stages are not included in $\mathcal{S}_i$. According to Algorithm 1, each time we find arm $i$ in the argmax in Line 7 in some stages, $r_i$ is reduced by half, and $N_i$ is doubled subsequently. Then we play arm $i$ for consecutive $N_i$ rounds. This means that the number of rounds of each stage in $\mathcal{S}_i$ are $2C_1\log\frac{1}{\delta}$, $4C_1\log\frac{1}{\delta}$, …, $2^{K_i}C_1\log\frac{1}{\delta}$. In total, arm $i$ has been played for $\left(2^{K_i+1}-1\right)C_1\log\frac{1}{\delta}$ rounds. Because the total number of rounds is at most $T$, we have

$$nC_1\log\frac{1}{\delta}+\sum_{i=1}^{n}\left(2^{K_i+1}-1\right)C_1\log\frac{1}{\delta}\le T. \qquad (8)$$

where the first term of (8) is the number of rounds in the initialization stage. Because $2^x$ is a convex function in $x\in[0,+\infty)$, by Jensen's inequality we have $\sum_{i=1}^{n}2^{K_i+1}\ge n\cdot 2^{1/n\sum_{i=1}^{n}(K_i+1)}$. Plugging this into (8), we have

$$\sum_{i=1}^{n}K_i\le n\log_2\left(\frac{T}{nC_1\log\frac{1}{\delta}}\right)-n.$$

Since QMC is called for $n+\sum_{i=1}^{n}K_i$ times, by the union bound, with probability at least $1-n\delta\log_2\left(\frac{T}{nC_1\log\frac{1}{\delta}}\right)$, the output estimate of every invocation of QMC satisfies (7). We refer to the event as the *good* event and assume that it holds below.

Recall that $i^*$ is the optimal arm and $i_s$ is the arm chosen by the algorithm during stage $s$. By the argmax in Line 7 of Algorithm 1, $\hat{\mu}(i_s)+r_{i_s}\ge\hat{\mu}(i^*)+r_{i^*}$. Under the good event, $\mu(i_s)+r_{i_s}\ge\hat{\mu}(i_s)$ and $\hat{\mu}(i^*)+r_{i^*}\ge\mu(i^*)$. Therefore, $\mu(i_s)+2r_{i_s}\ge\hat{\mu}(i_s)+r_{i_s}\ge\hat{\mu}(i^*)+r_{i^*}\ge\mu(i^*)$, and it follows that $\Delta_{i_s}:=\mu(i^*)-\mu(i_s)\le 2r_{i_s}$.

For each arm $i$, we denote by $R(T;i)$ the contribution of arm $i$ to the cumulative regret over $T$ rounds. By our notation above, arm $i$ is pulled in $K_i$ stages and the initialization stage. In initialization stages it is pulled for $C_1\log\frac{1}{\delta}$ times. In each stage of $\mathcal{S}_i$ it is pulled for $N_i=2C_1\log\frac{1}{\delta}$, $4C_1\log\frac{1}{\delta}$, …, $2^{K_i}C_1\log\frac{1}{\delta}$ times respectively, and the reward gap $\Delta_i\le 2r_i$ in the last stage is $2\cdot\frac{1}{2^{K_i-1}}=2^{2-K_i}$.

Note that the index of the stage in $\mathcal{S}_i$ does not influence the gap $\Delta_{i_s}$. Therefore, we can use $2^{2-K_i}$ to bound the gap of $\mu(i^*)$ and $\mu(i_s)$. For those arm $i$ which are only pulled in the initialization stage, we bound their reward gap to 1. Thus, we have

$$\begin{aligned} R(T; i) &\leq \max \left\{ \sum_{k=0}^{K_i} 2^k C_1 \log \frac{1}{\delta} \cdot 2^{2-K_i}, C_1 \log \frac{1}{\delta} \right\} \\ &\leq 8 C_1 \log \frac{1}{\delta}. \end{aligned}$$

The cumulative regret is the summation of $R(T; i)$ for $i \neq i^*$; we have

$$R(T) = \sum_{i \neq i^*} R(T; i) \leq 8(n-1) C_1 \log \frac{1}{\delta}.$$

It completes the proof since we have good event with probability at least $1 - n\delta \log_2 \left( \frac{T}{n C_1 \log \frac{1}{\delta}} \right)$. □

**Corollary 1.** *Set $\delta = \frac{1}{T}$, QUCB$_1(\delta)$ satisfies*

$$\mathbb{E}[R(T)] \leq (8(n-1)C_1 + 1) \log_2 T = O(n \log T).$$

For the bounded variance assumption, we can slightly modify Algorithm 1 to obtain a new algorithm called QUCB$_2(\delta)$ and bound its regret to poly-logarithmic order with similar proofs.

**Theorem 2.** *Let $C_2$ be the constant in Lemma 1. Under the bounded variance assumption, with probability at least $1 - n\delta \log_2 \left( \frac{T}{n C_2 \log \frac{1}{\delta}} \right)$, the cumulative regret of QUCB$_2(\delta)$ satisfies $R(T) \leq O\left( n\sigma \log^{3/2}(T) \log\log(T) \log \frac{1}{\delta} \right)$. Moreover, setting $\delta = 1/T$,*

$$\mathbb{E}[R(T)] = O\left( n\sigma \log^{5/2}(T) \log\log(T) \right).$$

## Quantum Stochastic Linear Bandits

In this section, we present the algorithm QLinUCB as well as its analysis for quantum stochastic linear bandits, showing a $\mathrm{poly}(\log T)$ regret bound of QLinUCB. For QSLB setting, we combine LinUCB with QMC to exploit the power of quantum oracle. Recall the obstacles introduced by the quantum subroutine we encountered when we design the algorithm for QMAB. In the QSLB setting, we face the same problem and we again use the staging technique to solve the problem. However, the doubling stage length in QUCB does not work here, and it can only lead to a regret bound related to the size of the action set. We aim to obtain a regret bound which have a dependence on the dimension of the action set rather than its size. In fact, similar to classical SLB, we allow the action set to be infinite. Thus, we must consider the linear dependence of different actions, and use different stage lengths from QUCB to fit this situation.

As a quantum variant of the classical algorithm LinUCB, QLinUCB also adopts the canonical upper confidence bound (UCB) framework. It runs in several stages. In stage $s$, it first constructs a confidence region $\mathcal{C}_{s-1}$ for the true parameter $\theta^*$, and then picks the best action $x_s \in A$ over $\mathcal{C}_{s-1}$ as shown in the line 4 of Algorithm 2. After $x_s$ is determined, it sets an carefully selected accuracy value $\epsilon_s$ for stage $s$ and plays action $x_s$ for the next $\frac{C_1}{\epsilon_s} \log \frac{m}{\delta}$ rounds, where $m := d \log(\frac{L^2 T^2}{d\lambda} + 1)$ is an upper bound for the number of total stages, see Lemma 2. When playing action $x_s$ during this stage, the algorithm implements a quantum circuit for QMC$(\mathcal{O}_{x_s}, \epsilon_s, \frac{\delta}{m})$ and gets an estimate $y_s$ of $x_s^\top \theta^*$ with accuracy $\epsilon_s$ and error probability less than $\delta/m$. After that, it updates the estimate $\hat{\theta}_s$ of $\theta^*$ using a weighted least square estimator. That is,

$$\hat{\theta}_s = \operatorname*{argmin}_{\theta \in \Theta} \sum_{k=1}^{s} \frac{1}{\epsilon_k^2} \| x_k^\top \theta - y_k \|_2^2 + \lambda \|\theta\|_2^2. \tag{9}$$

where $\lambda$ is a regularizing parameter. We give estimates $y_k$ different weights according to their accuracy in this least square estimator. Note that classical LinUCB use an unweighted least square estimator to estimate the parameter $\theta^*$, and this is another modification we makes other than the staging technique and the introduction of QMC. The estimator (9) has simple closed-form solution as follows. Let

$$V_s = \lambda I + \sum_{k=1}^{s} \frac{1}{\epsilon_k^2} x_k x_k^\top = \lambda I + X_s^\top W_s X_s$$

Then

$$\hat{\theta}_s := V_s^{-1} X_s^\top W_s Y_s,$$

where $X_s, Y_s, W_s$ are defined in line $9, 10, 11$ of Algorithm 2. Besides, with the definition of $V_s$, QLinUCB actually sets $\epsilon_s = \|x_s\|_{V_{s-1}^{-1}}$ where $V_{s-1}$ is calculated in stage $s-1$. Our choice of $\epsilon_s$ and the $\frac{1}{\epsilon_k^2}$ weight of the least square estimator in (9) are the key components of the quantum speedup of QLinUCB.

Although the stage length of QLinUCB is not doubling like what happens in QUCB, we find the quantity $\det(V_s)$ get doubled each stage, that is, $\det(V_{s+1}) = 2\det(V_s)$. Since $\det(V_s)$ is closely related to the total number of rounds of $s$ stages, thus $\det(V_s)$ can not grow too large. Using these facts, we can still upper bound the number of total stages to $O(d \log T)$, which is shown in Lemma 2.

**Lemma 2.** *Algorithm 2 has at most $m = d \log(\frac{L^2 T^2}{d\lambda} + 1)$ stages, where $\lambda$ is the regularizing parameter in (9).*

Then we show in the following lemma that the confidence regions we construct in each stage contain the true parameter $\theta^*$ with high probability.

**Lemma 3.** *With probability at least $1 - \delta$, for all $s \geq 0$, $\theta^* \in \mathcal{C}_s := \{\theta \in \mathbb{R}^d : \|\theta - \hat{\theta}_s\|_{V_s} \leq \lambda^{1/2} S + \sqrt{ds}\}$.*

Together with Lemma 2 and Lemma 3, following the standard optimism in the face of uncertainty proof we can bound the cumulative regret of each stage to $O\left(d\sqrt{\log T}\right)$, leading to our regret bound.

**Theorem 3.** *Under the bounded value assumption, with probability at least $1 - \delta$, the regret of QLinUCB$_1(\delta)$ satisfies*

$$R(T) = O\left( d^2 \log^{3/2} \left( \frac{L^2 T^2}{d\lambda} + 1 \right) \log \frac{d \log(\frac{L^2 T^2}{d\lambda} + 1)}{\delta} \right).$$

Algorithm 2: QLinUCB$_1(\delta)$

```
Parameters: fail probability $\delta$
Initialize $V_0 \leftarrow \lambda I_d$, $\hat{\theta}_0 \leftarrow \mathbf{0} \in \mathbb{R}^d$ and $m \leftarrow d\log(\frac{L^2T^2}{d\lambda}+1)$.
for each stage $s = 1, 2, \dots$ (terminate when we have used $T$ queries to all $\mathcal{O}_i$) do
    $\mathcal{C}_{s-1} \leftarrow \{\theta \in \mathbb{R}^d : \|\theta - \hat{\theta}_{s-1}\|_{V_{s-1}} \leq \lambda^{1/2}S + \sqrt{d(s-1)}\}$.
    $(x_s, \tilde{\theta}_s) \leftarrow \mathrm{argmax}_{(x,\theta)\in A\times\mathcal{C}_{s-1}} x^\top\theta$.
    $\epsilon_s \leftarrow \|x_s\|_{V_{s-1}^{-1}}$.
    for the next $\frac{C_1}{\epsilon_s}\log\frac{m}{\delta}$ rounds do
        Play action $x_s$ and run QMC$_1(\mathcal{O}_{x_s}, \epsilon_s, \delta/m)$, getting $y_s$ as an estimation of $x_s^\top\theta^*$.
    end for
    $X_s \leftarrow (x_1, x_2, \dots, x_s)^\top \in \mathbb{R}^{s\times d}$.
    $Y_s \leftarrow (y_1, y_2, \dots, y_s)^\top \in \mathbb{R}^s$.
    $W_s \leftarrow \mathrm{diag}\left(\frac{1}{\epsilon_1^2}, \frac{1}{\epsilon_2^2}, \dots, \frac{1}{\epsilon_s^2}\right)$
    Update $V_s \leftarrow V_{s-1} + \frac{1}{\epsilon_s^2}x_s x_s^\top$ and $\hat{\theta}_s \leftarrow V_s^{-1}X_s^\top W_s Y_s$.
end for
```

*Moreover, the expected regret of QLinUCB$_2(\frac{m}{T})$ satisfies*

$$\mathbb{E}[R(T)] = O\left(d^2\log^{5/2}\left(\frac{L^2T^2}{d\lambda}+1\right)\right).$$

For the bounded variance assumption, we have a similar result with an additional overhead of $O\left(\log^{3/2}(T)\log\log(T)\right)$ in the regret bound.

**Theorem 4.** *Under the bounded variance assumption, with probability at least $1-\delta$, the regret of QLinUCB$_2(\delta)$ satisfies*

$$R(T) = O\left(\sigma d^2\log^3(\sigma LT)\log\log(\sigma LT)\log\frac{m}{\delta}\right).$$

*Moreover, setting $\delta = \frac{m}{T}$, we have*

$$\mathbb{E}[R(T)] = O\left(\sigma d^2\log^4(\sigma LT)\log\log(\sigma LT)\right).$$

## Numerical Experiments

We conduct experiments to demonstrate the performance of our two quantum variants of bandit algorithms. For simplicity, we use the Bernoulli rewards in both bandit settings. When considering the Bernoulli noise, we can use the Quantum Amplitude Estimation algorithm in (Brassard et al. 2002) as our mean estimator. In this section, we first perform simulations without the quantum noise, where we can run algorithms for a huge amount of rounds to show the advantage of QUCB and QLinUCB on regret. Then, we consider the presence of quantum noise and study the influence of quantum noise to regret. Specifically, we consider a widely used quantum noise model called depolarizing noise. For all experiments, we repeat for 100 times and calculate the average regret and standard deviation. Our experiments are executed on a computer equipped with Xeon E5-2620 CPU and 64GB memory.

### Experiments without Quantum Noise

**QMAB setting.** For the QMAB setting, we run UCB and QUCB on a 2-arm bandit for $T = 10^6$ rounds. Classical UCB algorithm has an instance-dependent $O(\log T)$ bound. Let $SA$ be the set of all sub-optimal arms, then its cumulative regret $R(T)$ satisfies $\mathbb{E}[R(T)] \leq O\left(\sum_{i\in SA}\frac{1}{\Delta_i}\log T\right)$ (Lattimore and Szepesvári 2020), where $\Delta_i$ is the reward gap of arm $i$. That is to say if the reward gap is independent with time horizon $T$, then UCB also has an $O(\log T)$ regret. Thus, to compare UCB and QUCB, we set the reward gap of our experimental instance relatively small. Overall, we set the mean reward of the optimal arm to be $0.5$, then we let the reward gaps of the sub-optimal arm be $0.01$. The result are shown in Figure 1 (a). From the result, we can see that QUCB has much lower expected regret and variance than UCB when the reward gap is small. Furthermore, we find that even if we set the parameter $\delta$ much greater than $1/T$ which is the parameter used in Corollary 1, the regret still maintains low variance, which means the error probability is not as high as our theoretical bound.

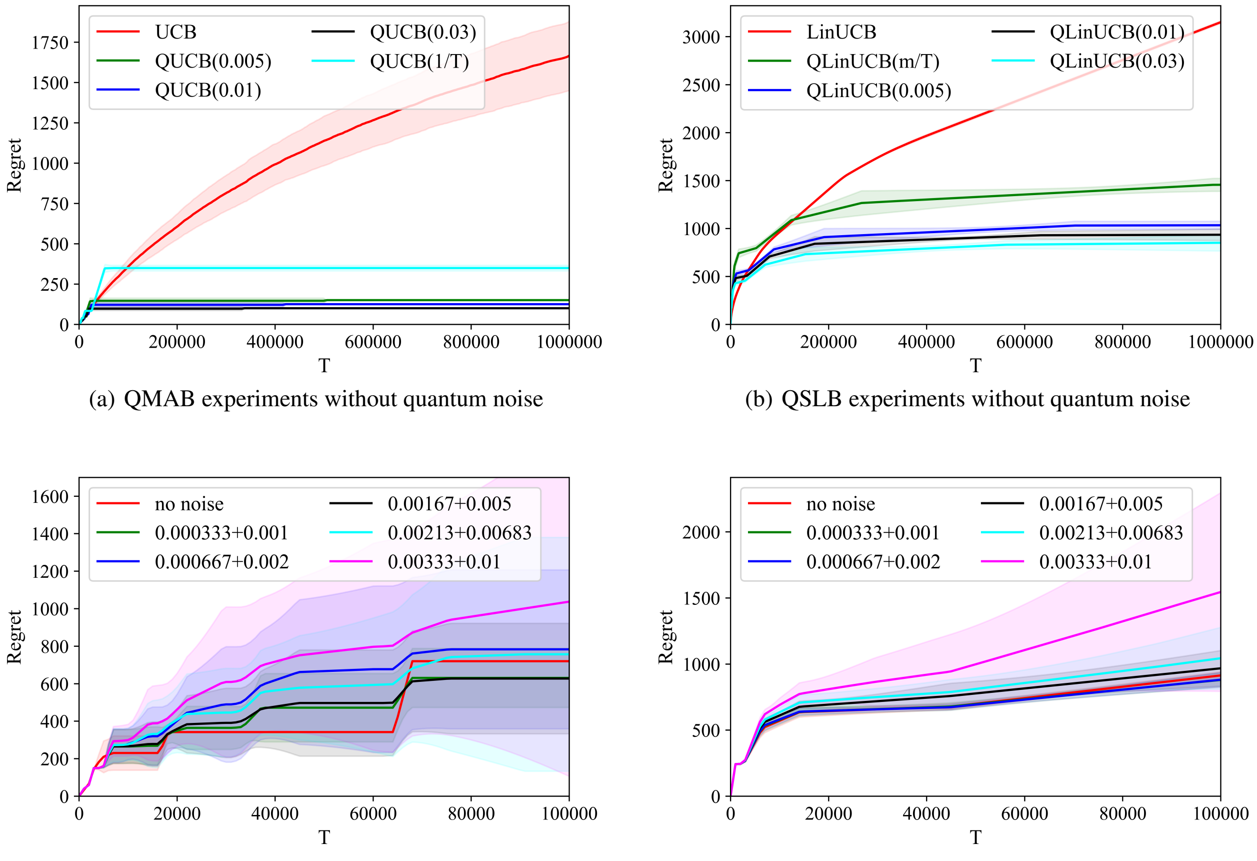
**QSLB setting.** For the QSLB setting, we study an instance in $\mathbb{R}^2$. We take time horizon $T = 10^6$. We use the finite action set and spread 50 actions equally spaced on the positive part of the unit circle. We set parameter $\theta^* = (\cos(0.35\pi), \sin(0.35\pi))$. We compare our algorithm with the well-known LinUCB. The simulation result is shown in Figure 1 (b), $\lambda$ is set to 1 throughout the numerical experiments. It can be observed that QLinUCB has lower regret than LinUCB, and they both have small variances.

### Experiments with Depolarizing Noise

To study the effects of quantum noise on our two algorithms, we conduct simulations with depolarizing noise channels using the Python open-source quantum computing tools kit "Qiskit".

**Noise model.** We choose $\{\mathrm{U1}, \mathrm{U2}, \mathrm{U3}, \mathrm{CNOT}\}$ as our basis gates set, and we consider the depolarizing noise model. This model is considered in quantum computers with many qubits and gates, including Sycamore (Arute et al. 2019). The depolarizing noise channel $E$ is defined as an operator acting on the density operator: $E(\rho) = (1-\mathrm{err})\rho + \mathrm{err}\cdot\mathrm{tr}(\rho)\frac{I_m}{2^m}$, where err is the error rate and $m$ is the number of qubits of this channel. For error rate, we try $\{0.001, 0.002, 0.005, 0.01\}$ for two-qubit channel, and set the single-qubit error rate as the $\frac{1}{3}$ of the two-qubit error rate. Besides, we also try single-qubit error of $0.00213$ and a two-qubit error of $0.00683$, which are the error rate of Sycamore.

**Results.** For the QMAB setting, we choose the 2-arm instance with rewards $\{0.4, 0.5\}$. For the QSLB setting, we use the same instance as the noise-free experiment. Since simulating quantum noise channels on a classical computer is time-consuming, we set the time horizon of the bandit instances to be $10^5$. Note that, since the time horizon in our instance is relatively small, the quantum variants of UCB cannot outperform the classical UCB in this case even though the quantum variants have advantages in asymptotic order.



(a) QMAB experiments without quantum noise

(b) QSLB experiments without quantum noise

(c) QMAB experiments with quantum noise

(d) QSLB experiments with quantum noise

Figure 1: Results of the numerical experiments. In (c) and (d), the number $0.00333 + 0.01$ in the legend means that we set the error rate of a single-qubit channel to be $0.00333$ and the error rate of a two-qubit channel to be $0.01$.

This is because QMC introduces larger constant factors. In this section, we only focus on the performance of the quantum algorithms with different error rates. We plot the results in Figure 1 (c) and (d). From the figures, the expected regret of QUCB is not affected much by depolarizing noise when the two-qubit depolarizing error rate is at most $0.00683$. Even if the error rates are $0.00333$ and $0.01$, the regret of QUCB is still much better than pulling arms randomly (which incurs an expected regret of $5000$ at the end). However, the presence of depolarizing noise does increase the variance of the regret. As for QLinUCB, when the two-qubit channel error rate is no more than $0.00683$, the regret keeps almost unchanged and small variance all the time. When the two-qubits channel error rate is $0.01$, QLinUCB suffers from higher expected regret and variance. Overall, at the level of noise that can be achieved by today's quantum computers, the regrets of our algorithms are only relatively little affected when the time horizon is in the order of $10^5$.

## Conclusion

In this paper, we proved that quantum versions of multi-arm bandits and stochastic linear bandits both enjoy $O(\mathrm{poly}(\log T))$ regrets. To the best of our knowledge, this is the first provable quantum speedup for regrets of bandit problems and in general exploitation in reinforcement learning. Compared to previous literature on quantum exploration algorithms for MAB and reinforcement learning, our quantum input model is simpler and only assumes quantum oracles for each individual arm. We also corroborate our results with numerical experiments.

Our work raises several natural questions for future investigation. First, it is natural to seek quantum speedups for regrets of other bandit problems. Second, additional research is needed to achieve speedup in $n$ and $d$ for the regrets of MAB and SLB, respectively; this may require a reasonable new model. Third, it is worth understanding whether algorithms with $T$-independent regret exist, or we can prove a matching quantum lower bound.

## Acknowledgements

We thank Ronald de Wolf for helpful discussions. This work was supported in part by the Strategic Priority Research Program of Chinese Academy of Sciences under Grant No. XDA27000000, the National Natural Science Foundation of China Grants No. 61832003, 62272441, 61872334. Tongyang Li was supported by a startup fund from Peking University, and the Advanced Institute of Information Technology, Peking University.

## References

Abbasi-Yadkori, Y.; Pál, D.; and Szepesvári, C. 2011. Improved algorithms for linear stochastic bandits. *Advances in Neural Information Processing Systems*, 24.

Arunachalam, S.; and de Wolf, R. 2017. Guest column: a survey of quantum learning theory. *ACM SIGACT News*, 48(2): 41–67. arXiv:1701.06806.

Arute, F.; Arya, K.; Babbush, R.; Bacon, D.; Bardin, J. C.; Barends, R.; Biswas, R.; Boixo, S.; Brandao, F. G.; Buell, D. A.; et al. 2019. Quantum supremacy using a programmable superconducting processor. *Nature*, 574(7779): 505–510. arXiv:1910.11333.

Auer, P.; Cesa-Bianchi, N.; and Fischer, P. 2002. Finite-time analysis of the multiarmed bandit problem. *Machine learning*, 47(2): 235–256.

Auer, P.; Cesa-Bianchi, N.; Freund, Y.; and Schapire, R. E. 2002. The nonstochastic multiarmed bandit problem. *SIAM Journal on Computing*, 32(1): 48–77.

Biamonte, J.; Wittek, P.; Pancotti, N.; Rebentrost, P.; Wiebe, N.; and Lloyd, S. 2017. Quantum machine learning. *Nature*, 549(7671): 195. arXiv:1611.09347.

Brassard, G.; Høyer, P.; Mosca, M.; and Tapp, A. 2002. Quantum amplitude amplification and estimation. *Contemporary Mathematics*, 305: 53–74. arXiv:quant-ph/0005055.

Casalé, B.; Di Molfetta, G.; Kadri, H.; and Ralaivola, L. 2020. Quantum bandits. *Quantum Machine Intelligence*, 2(1): 1–7. arXiv:2002.06395.

Chapelle, O.; Manavoglu, E.; and Rosales, R. 2014. Simple and scalable response prediction for display advertising. *ACM Transactions on Intelligent Systems and Technology)*, 5(4): 1–34.

Dani, V.; Hayes, T. P.; and Kakade, S. M. 2008. Stochastic Linear Optimization under Bandit Feedback. In *21st Annual Conference on Learning Theory*, 355–366.

Dunjko, V.; and Briegel, H. J. 2018. Machine learning & artificial intelligence in the quantum domain: a review of recent progress. *Reports on Progress in Physics*, 81(7): 074001. arXiv:1709.02779.

Dunjko, V.; Taylor, J. M.; and Briegel, H. J. 2015. Framework for learning agents in quantum environments. arXiv:1507.08482.

Dunjko, V.; Taylor, J. M.; and Briegel, H. J. 2016. Quantum-enhanced machine learning. *Physical Review Letters*, 117(13): 130501. arXiv:1610.08251.

He, J.; Yang, F.; Zhang, J.; and Li, L. 2022. Quantum algorithm for online convex optimization. *Quantum Science and Technology*, 7(2): 025022.

Lattimore, T.; and Szepesvári, C. 2020. *Bandit algorithms*. Cambridge University Press.

Lei, H.; Tewari, A.; and Murphy, S. A. 2017. An actor-critic contextual bandit algorithm for personalized mobile health interventions. arXiv:1706.09090.

Lumbreras, J.; Haapasalo, E.; and Tomamichel, M. 2022. Multi-armed quantum bandits: Exploration versus exploitation when learning properties of quantum states. *Quantum*, 6: 749.

Montanaro, A. 2015. Quantum speedup of Monte Carlo methods. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 471(2181): 20150301. arXiv:1504.06987.

Schuld, M.; Sinayskiy, I.; and Petruccione, F. 2015. An introduction to quantum machine learning. *Contemporary Physics*, 56(2): 172–185.

Silver, D.; Huang, A.; Maddison, C. J.; Guez, A.; Sifre, L.; Van Den Driessche, G.; Schrittwieser, J.; Antonoglou, I.; Panneershelvam, V.; Lanctot, M.; et al. 2016. Mastering the game of Go with deep neural networks and tree search. *Nature*, 529(7587): 484–489.

Sutton, R. S.; and Barto, A. G. 2018. *Reinforcement learning: An introduction*. MIT press.

Villar, S. S.; Bowden, J.; and Wason, J. 2015. Multi-armed bandit models for the optimal design of clinical trials: benefits and challenges. *Statistical Science: a Review Journal of the Institute of Mathematical Statistics*, 30(2): 199. arXiv:1507.08025.

Wang, D.; Sundaram, A.; Kothari, R.; Kapoor, A.; and Roetteler, M. 2021a. Quantum algorithms for reinforcement learning with a generative model. In *International Conference on Machine Learning*, 10916–10926. PMLR. arXiv:2112.08451.

Wang, D.; You, X.; Li, T.; and Childs, A. M. 2021b. Quantum Exploration Algorithms for Multi-Armed Bandits. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, 10102–10110. arXiv:2007.07049.