

Best Arm Identification with Quantum Oracles

Xuchuang Wang¹, Yu-Zhen Janice Chen¹, Matheus Guedes de Andrade¹, Jonathan Allcock²,
 Mohammad Hajiesmaili¹, John C.S. Lui³, and Don Towsley¹

¹College of Information and Computer Sciences, University of Massachusetts, Amherst, Massachusetts, USA

²Tencent Quantum Laboratory, Tencent, Hong Kong

³Department of Computer Science and Engineering, The Chinese University of Hong Kong, Hong Kong
 {xuchuangwang, yuzhenchen, mguedesdeand, hajiesmaili, towsley}@cs.umass.edu,
 jonallcock@tencent.com, cs Lui@cse.cuhk.edu.hk

Abstract

Best arm identification (BAI) is a key problem in stochastic multi-armed bandits, where K arms each has an associated reward distribution, and the objective is to minimize the number of queries needed to identify the best arm with high confidence. In this paper, we explore BAI using *quantum oracles*. For the case where each query probes only one arm ($m = 1$), we devise a quantum algorithm with a query complexity upper bound of $\tilde{O}(K\Delta^{-1}\log(1/\delta))$, where δ is the confidence parameter and Δ is the reward gap between best and second best arms. This improves on the classical bound by a factor of Δ^{-1} . For the general case where a single query can probe m arms ($1 \leq m \leq K$) simultaneously, we propose an algorithm with an upper bound of $\tilde{O}((K/\sqrt{m})\Delta^{-1}\log(1/\delta))$, improving by a factor of \sqrt{m} compared to the $m = 1$ case. We also provide query complexity lower bounds for both scenarios, which match the upper bounds up to logarithmic factors, and validate our theoretical results with Qiskit-based simulations.

1 Introduction

Best arm identification (BAI) is a fundamental problem in the bandits and online learning communities (Audibert, Bubeck, and Munos 2010; Bubeck, Munos, and Stoltz 2011; Mannor and Tsitsiklis 2004). Given $K \in \mathbb{N}^+$ arms, each arm k is associated with a reward distribution with unknown mean μ_k , and the goal of BAI is to identify the arm with the largest mean reward, with a confidence of $1 - \delta$, using as few queries as possible. The number of queries required is called the *query complexity*. Each query, in the classical setting, corresponds to the learner pulling (sampling) one arm and observing a reward drawn from the arm’s reward distribution. As the learner only cares about finding the best arm, the BAI problem is a pure exploration problem. BAI has many real world applications, such as, clinical trials (Robbins 1952), network routing (Barrachina-Muñoz and Belalta 2017), and crowdsourcing (Zhou, Chen, and Li 2014).

Recent progress in building quantum computers (Arute et al. 2019; Chow, Dial, and Gambetta 2021) and quantum networks (Wehner, Elkouss, and Hanson 2018; Azuma et al. 2022) has been encouraging, and wide applications of quantum systems are envisaged in the near future. In these quantum systems, BAI problems also emerge. For example, a

quantum network may contain multiple channels between source and destination nodes. Among these channels, one may want to determine the “best” one, where “best” may refer, for example, to the channel with the highest fidelity (Liu et al. 2024) or with the lowest noise (Li, Deng, and Zhou 2008). Another example is in distributed quantum computing (Cacciapuoti et al. 2019), where different quantum computers may have different performances when applied to the same problem, and one wants to identify the quantum computer that provides the best performance for a given task. Although one can still apply classical BAI algorithms to address these problems, we aim to show that the quantum information feedback from these quantum systems can be leveraged to improve the learning efficiency.

In this paper, we study the BAI problem in quantum systems, where the learner can query the arms using quantum queries. More specifically, we study two key advantages of quantum feedback in BAI: (1) quantum parallelism (Chuang and Yamamoto 1995), and (2) quantum entanglement (Einstein, Podolsky, and Rosen 1935). The quantum Monte Carlo estimator (parallelism, Lemma 1) provides a more efficient estimator for the learner to estimate arm rewards. Additionally, multi-qubit oracles with entangled quantum superposition inputs enable the learner to query multiple arms simultaneously (coherently) within a single query. We model the former advantage by *weak quantum oracles*, one for each arm, and the latter by a *constrained quantum oracle* which can query several arms coherently (both detailed in Section 2.2). When a constrained oracle can query all arms coherently, we call it a *strong quantum oracle*.

The development of effective algorithms for both oracles necessitates the use of quantum computing to manage quantum information feedback and leverage quantum parallelism and entanglement. However, obtaining a valid output from a quantum computing subroutine, such as amplitude amplification (Brassard et al. 2002), typically demands multiple consecutive queries on the same arm or a subset of arms for the constrained oracle. This characteristic renders the classical BAI algorithm design and analysis ineffective for BAI with quantum oracles. Consequently, it is imperative to contemplate new algorithm designs and analyses for BAI with quantum oracles. On the other hand, to investigate the fundamental limit of quantum BAI problems, we need to establish query complexity lower bounds. However,

Oracle	Lower Bound	Upper Bound
Classical (2)	$\Omega\left(\sum_k \frac{1}{\Delta_k^2} \log \frac{1}{\delta}\right)$ (Mannor and Tsitsiklis 2004)	$O\left(\sum_k \frac{1}{\Delta_k^2} \log \frac{1}{\delta}\right)$ (Karnin, Koren, and Somekh 2013)
Strong quantum (5)	$\Omega\left(\sqrt{\sum_k \frac{1}{\Delta_k^2}}\right)$ (Wang et al. 2021)	$\tilde{O}\left(\sqrt{\sum_k \frac{1}{\Delta_k^2} \log \left(\frac{1}{\delta}\right)}\right)$ (Wang et al. 2021)
Weak quantum (3)	$\Omega\left(\sum_k \frac{1}{\Delta_k} \log \left(\frac{1}{\delta}\right)\right)$	$\tilde{O}\left(\sum_k \frac{1}{\Delta_k} \log \left(\frac{1}{\delta}\right)\right)$
Constrained quantum (4)	$\Omega\left(\sum_{S \in \mathfrak{B}} \sqrt{\sum_{k \in S} \frac{1}{\Delta_k^2}}\right)$	$\tilde{O}\left(\sum_{S \in \mathfrak{B}} \sqrt{\sum_{k \in S} \frac{1}{\Delta_k^2} \log \left(\frac{1}{\delta}\right)}\right)$

Table 1: Comparison of query complexity bounds with classical and quantum BAI

given that quantum information (including parallelism and entanglement) offers more informative and inherently different query feedback than classical BAI, the classical proofs of BAI complexity lower bound are not applicable. Instead, one needs to adapt quantum computation and quantum information approaches to examine quantum BAI problems. Additionally, to empirically validate the performance of devised algorithms for BAI with quantum oracles, one has to utilize quantum circuits (Nielsen and Chuang 2002) and implement the necessary quantum computation subroutines using basic quantum logic gates.

We summarize the key contributions of this paper as follows:

- For BAI with the weak quantum oracle, we derive a query complexity lower bound $\Omega\left(\sum_k (1/\Delta_k) \log(1/\delta)\right)$, showing that no quantum algorithm can achieve a smaller query complexity. Then, we propose an elimination-based quantum algorithm (Q-Elim) and derive its query complexity upper bound $\tilde{O}\left(\sum_k (1/\Delta_k) \log(1/\delta)\right)$, where the suboptimality gap $\Delta_k := \mu_1 - \mu_k$ is the difference in the mean rewards of the optimal arm and arm k , and $\tilde{O}(\cdot)$ hides poly-logarithmic factors. This implies that Q-Elim is near-optimal up to logarithmic factors for BAI with the weak quantum oracle (Section 3).
- For BAI with the m -constrained quantum oracle, we propose a partition-based quantum algorithm (Q-Part), and derive its query complexity upper bound $\tilde{O}\left(\sum_{S \in \mathfrak{B}} \sqrt{\sum_{k \in S} 1/\Delta_k^2} \log(1/\delta)\right)$, where \mathfrak{B} is a partition of the full arm set, i.e., a set of arm subsets, each subset S containing m arms. We also derive a query complexity lower bound of $\Omega\left(\sum_{S \in \mathfrak{B}} \sqrt{\sum_{k \in S} 1/\Delta_k^2}\right)$ for the partition algorithm class, which matches the upper bound of Q-Part up to logarithmic factors (Section 4).
- We implement our quantum algorithms using the IBM Qiskit (Qiskit contributors 2023). We first corroborate the superiority of our quantum algorithms over classical BAI algorithms. We then evaluate our algorithms under simulated quantum noise (Section 5).

Related Works Prior works on multi-armed bandits (MAB) typically focus on regret minimization and BAI. This paper focuses on the BAI setting (Even-Dar, Mannor, and Mansour 2002; Even-Dar et al. 2006; Mannor and Tsitsiklis 2004). The BAI setting can be divided into two categories: (1) BAI with fixed confidence—find the best arm with a confidence of at least $1 - \delta$ ($\delta \in (0, 1)$) using as few samples

as possible (Bubeck, Munos, and Stoltz 2011); and (2) BAI with fixed budget—given a fixed budget of Q queries, find the best arm with as high a probability as possible (Karnin, Koren, and Somekh 2013). In this paper, we focus on the former category which, for brevity, we will refer to simply as the BAI problem. BAI with the strong quantum oracle was first studied by Casalé et al. (2020); Wang et al. (2021), where they proposed a near-optimal quantum algorithm that enjoys a quadratic speedup in query complexity. We are the first to study the BAI problem with the weak and constrained quantum oracles. Besides BAI, another objective, regret minimization in bandit theory, has also been studied with quantum oracles, including Wan et al. (2023); Dai et al. (2023); Wu et al. (2023), etc. We defer a more detailed discussion of these works and other loosely related works to Appendix A.

In Table 1, we summarize the key results in this paper and compare them to prior works. Comparing the Δ_k -dependence of the complexities, we have

$$\underbrace{\sqrt{\sum_k \frac{1}{\Delta_k^2}}}_{\text{Strong oracle}} \leq \underbrace{\sum_{S \in \mathfrak{B}} \sqrt{\sum_{k \in S} \frac{1}{\Delta_k^2}}}_{m\text{-constrained oracle}} \leq \underbrace{\sum_k \frac{1}{\Delta_k}}_{\text{Weak oracle}} \leq \underbrace{\sum_k \frac{1}{\Delta_k^2}}_{\text{Classical oracle}} \quad (1)$$

All BAI problems with quantum oracles enjoy smaller query complexities than the classical one. The query complexity of the weak quantum oracle is worst among quantum oracles, which is due to the fact that the weak oracle cannot exploit quantum entanglement to probe multiple arms in parallel. The query complexity of the m -constrained quantum oracle lies between that of strong and weak oracles, and when $m = 1$ (resp., $m = K$) the complexity coincides with that of weak (resp., strong) oracles.

2 Model

2.1 Preliminaries

Best arm identification (BAI). Consider a multi-armed bandit (MAB) consisting of K arms, where each arm $k \in \mathcal{K} := \{1, 2, \dots, K\}$ is associated with a Bernoulli distribution $\mathcal{B}(\mu_k)$ with mean $\mu_k \in (0, 1)$.¹ An MAB instance is determined by the mean rewards of its arms, and we denote an instance \mathcal{I} with means μ_1, \dots, μ_K as $\mathcal{I} := \{\mu_1, \dots, \mu_K\}$.

¹More general distributions, such as sub-Gaussian or bounded distributions, have also been considered in the MAB literature (Auer and Ortner 2010; Lattimore and Szepesvári 2020).

For simplicity, we assume the K arms are labeled in descending order of their means: $\mu_1 > \mu_2 \geq \dots \geq \mu_K$, unknown to the learner, and denote the mean reward (suboptimality) gap as $\Delta_k := \mu_1 - \mu_k$ for suboptimal arms $k > 1$ and $\Delta_1 := \Delta_2$ for the optimal arm. We assume a unique optimal arm for the simplicity of the later presentation of the algorithms and analysis. One could extend the results to multiple optimal arms with techniques in bandits literature, e.g., find an ϵ -optimal arm (Even-Dar, Mannor, and Mansour 2002). Then, given confidence parameter $\delta \in (0, 1)$, the best arm identification (BAI) problem is to correctly output the best arm with a probability of at least $1 - \delta$ using as few queries as possible, noted as the query complexity Q .

Next, we present some basics notation from quantum computation and information (Nielsen and Chuang 2002).

Bra-ket notation. We make use of bra-ket notation to represent quantum states, where the “ket” $|x\rangle := (x_1, x_2, \dots, x_n)^T \in \mathbb{C}^n$ denotes a column vector of n complex numbers, while the “bra” $\langle x| := |x\rangle^\dagger = (x_1^*, x_2^*, \dots, x_n^*)$, a row vector, is the conjugate transpose of $|x\rangle$. For two quantum states $|x\rangle, |y\rangle \in \mathbb{C}^n$, their inner product is denoted as $\langle x|y\rangle := \sum_{i=1}^n x_i^* y_i \in \mathbb{C}$, and given another quantum state $|z\rangle \in \mathbb{C}^m$, the tensor product between $|x\rangle$ and $|z\rangle$ is denoted as $|x\rangle|z\rangle = |x\rangle \otimes |z\rangle := (x_1 z_1, x_1 z_2, \dots, x_n z_m) \in \mathbb{C}^n \otimes \mathbb{C}^m$.

Qubit. A “qubit” is a two-level quantum system $|\phi\rangle = (\alpha, \beta) \in \mathbb{C}^2$, often written as $|\phi\rangle = \alpha|0\rangle + \beta|1\rangle$, where $|0\rangle = (1, 0)^T$ and $|1\rangle = (0, 1)^T$ are two basis states, and $\alpha, \beta \in \mathbb{C}$ are complex numbers, called amplitudes, satisfying $|\alpha|^2 + |\beta|^2 = 1$. A measurement of the qubit in the $\{|0\rangle, |1\rangle\}$ basis will give a ‘0’ with probability $|\alpha|^2$ and a ‘1’ with probability $|\beta|^2$.

Quantum query model. In the quantum query model, one has access to a black-box unitary operator (i.e., oracle) which implements a given transformation. The objective is to study the *query complexity*, i.e., the number of calls Q to the oracle needed to solve a given task; all other possible costs, e.g., gate complexity, are ignored. This is a commonly used model for studying quantum algorithms (Childs 2017, §20) and can be used, for instance, to obtain algorithmic running time lower bounds (Klauck, Špalek, and De Wolf 2007). In this paper, we study the query complexity of best arm identification with fixed confidence under weak and constrained quantum oracles.

2.2 Quantum Oracles

Before introducing the quantum oracles, we first recall the classical oracle for the BAI problem. That is, when querying an arm k , one obtains a reward drawn from a Bernoulli distribution $\mathcal{B}(\mu_k)$ with *unknown* mean μ_k , i.e.,

$$X_k \sim \mathcal{B}(\mu_k). \quad (2)$$

We refer to (2) as the *classical oracle*.

In the quantum setting, the Bernoulli distributions can be mapped to oracles $O_{\text{weak}}^{(k)}$ (one for each k) that act as follows,

$$O_{\text{weak}}^{(k)} : |0\rangle_R \mapsto \sqrt{1 - \mu_k} |0\rangle_R + \sqrt{\mu_k} |1\rangle_R, \quad (3)$$

where the register $|\cdot\rangle_R$ represents a single-qubit “bandit reward” register with basis states $|0\rangle$ and $|1\rangle$. The output qubit encodes the Bernoulli reward, meaning that if one measures the output in the basis $\{|0\rangle, |1\rangle\}$, the probability of observing $|1\rangle$ is μ_k , while the probability of observing $|0\rangle$ is $1 - \mu_k$. We refer to (3) as the *weak quantum oracle*.

Note that directly measuring the output qubits reduces the weak oracle to a Bernoulli distribution. However, aside from direct measurement, the output qubits enable efficient quantum parallelism through quantum computing algorithms, which we elaborate in Section 3.

To harness the entanglement properties of quantum information in real-world quantum systems, we consider a more general quantum oracle that allows simultaneous querying of multiple arms. In addition to the reward register $|\cdot\rangle_R$, we introduce an “arm index” register $|\cdot\rangle_I$, which has K orthogonal basis states $\{|k\rangle_I\}_{k=1}^K$, each corresponding to an arm. A quantum state in the $|\cdot\rangle_I$ register can be expressed as $\sum_{k=1}^K a_k |k\rangle_I$, where $a_k \in \mathbb{C}$ are the amplitudes of the arms, and normalization requires that $\sum_{k=1}^K |a_k|^2 = 1$.

With the assistance of the arm index register, we define a *constrained quantum oracle* that outputs states entangling the arm index and reward registers. Assuming the oracle can access $m \in \{1, 2, \dots, K\}$ arms simultaneously, for any subset of arms $\mathcal{S} \subseteq \mathcal{K}$ with $|\mathcal{S}| = m$ and $\sum_{k \in \mathcal{S}} |a_k|^2 = 1$, the oracle is defined as follows:

$$O_{\text{cons}}^{(\mathcal{S})} : \sum_{k \in \mathcal{S}} a_k |k\rangle_I |0\rangle_R \mapsto \sum_{k \in \mathcal{S}} a_k |k\rangle_I \left(\sqrt{1 - \mu_k} |0\rangle_R + \sqrt{\mu_k} |1\rangle_R \right). \quad (4)$$

when $m = 1$, the oracle reduces to the weak quantum oracle in (3), and when $m = K$, it becomes the strong quantum oracles as follows,

$$O_{\text{stro}} : \sum_{k=1}^K a_k |k\rangle_I |0\rangle_R \mapsto \sum_{k=1}^K a_k |k\rangle_I \left(\sqrt{1 - \mu_k} |0\rangle_R + \sqrt{\mu_k} |1\rangle_R \right). \quad (5)$$

The constrained quantum oracle in (4) is more powerful than the weak oracle in (3) because it can access multiple arms coherently in a single query, whereas the weak oracle only allows access to one arm at a time. In Section 4, we present a BAI algorithm using the m -constrained oracle, which outperforms the weak oracle when $m > 1$.

In practice, coherently querying a large number of channels may be technologically challenging, which motivates the general $m \leq K$ case. This limitation reflects a technology constraint where more options exist than can be accessed simultaneously. Such technological constraints may also affect, for example, access to quantum states stored in memory. In this case, a weak oracle would support individual calls to memory, while an m -constrained oracle functions like a dynamically loadable quantum random access memory (QRAM, see Appendix B), capable of querying multiple entries at once.

3 BAI with Weak Quantum Oracle

In this section, we address the BAI problem using a weak quantum oracle as described in (3). Querying this oracle for arm k yields the state $\sqrt{1 - \mu_k} |0\rangle + \sqrt{\mu_k} |1\rangle$. To estimate μ_k efficiently, we use the following lemma:

Lemma 1 (Performance of QuEst, adapted from Montanaro (2015); Grinko et al. (2021)). *For a weak quantum oracle $O_{\text{weak}}^{(k)}$ in (3), there exists a constant $C_1 > 1$ and a quantum estimation algorithm $\text{QuEst}(O_{\text{weak}}^{(k)}, \epsilon, \delta)$ that estimates μ_k with precision ϵ and confidence δ (i.e., $\mathbb{P}(|\hat{\mu}_k - \mu_k| \geq \epsilon) \leq \delta$), using at most $\frac{C_1}{\epsilon} \log \frac{1}{\delta}$ queries.*

This quantum estimator QuEst achieves a quadratic speedup over the classical estimators that require $O((1/\epsilon^2) \log(1/\delta))$ queries. Unfortunately, QuEst lacks flexibility: it does not generate any information before the entire procedure has completed, unlike classical estimators that improve estimates incrementally during the samples arriving and allows for sample reuse.

To address this issue, we first use QuEst to develop a batch-based elimination algorithm for BAI with the weak quantum oracle in Section 3.1. We then establish an upper bound on the query complexity of this algorithm in Section 3.2. Finally, in Section 3.3, we present a lower bound for any BAI algorithm using a weak quantum oracle, highlighting the fundamental limits of the task.

3.1 Algorithm Design

Algorithm 1 presents a quantum elimination algorithm (Q-Elim) for BAI. The core idea of the elimination process is to maintain a candidate arm set C (initially set to the full arm set \mathcal{K}), gradually identify and remove suboptimal arms from C as learning progresses, and terminate when C contains only one arm, which is then declared as the optimal arm.

Although several classical elimination algorithms, such as successive elimination (Even-Dar et al. 2006), have been proposed for BAI using classical oracles, these cannot be directly adapted by simply replacing classical estimators with the quantum estimator from Lemma 1 due to the rigidity of the quantum estimator (one cannot acquire any information from QuEst before the entire procedure completed).

A significant challenge in designing our quantum algorithm is determining when to perform quantum estimation QuEst and arm elimination. We address this by proposing a batch-based exploration and elimination scheme, where $j \in \{1, 2, \dots\}$ denotes the batch number. In each batch, we query all remaining arms in the candidate arm set C a number of times depending on the batch number j (Line 2), conduct $\text{QuEst}(O_{\text{weak}}^{(k)}, 2^{-j}, \frac{\delta}{2^j|C|})$ to estimate the mean rewards of arms in C based on the queries from this batch (Line 3), and eliminate newly identified suboptimal arms (Line 5) at the end of the batch. As j increases, we progressively increase both the number of queries (Line 6) and the estimation accuracy of QuEst (Lines 2 and 3).

Algorithm 1 Q-Elim: Quantum elimination algorithm for BAI with weak quantum oracle

Input: fixed confidence parameter δ and number of arms K
Initialize: empirical mean $\hat{\mu}_k \leftarrow 0$, candidate arm set $C \leftarrow \mathcal{K}$, batch number $j \leftarrow 1$

- 1: **while** $|C| > 1$ **do**
- 2: Query each arm $k \in C$ for $C_1 2^j \log(2^j |C| / \delta)$ times
- 3: Run $\text{QuEst}(O_{\text{weak}}^{(k)}, 2^{-j}, \frac{\delta}{2^j |C|})$ for each arm k in C and update these arms' estimates $\hat{\mu}_k$
- 4: $\hat{\mu}_{\max} \leftarrow \max_{k \in C} \hat{\mu}_k$
- 5: $C \leftarrow C \setminus \{k \in C : \hat{\mu}_k + 2 \cdot 2^{-j} \leq \hat{\mu}_{\max}\}$ ► Arm elimination
- 6: $j \leftarrow j + 1$

Output: the single remaining arm in C .

3.2 Query Complexity Upper Bound for Elimination Algorithm

Theorem 1 (Query complexity upper bound of Algorithm 1). *Given confidence parameter $\delta \in (0, 1)$, the query complexity of Q-Elim is upper bounded as follows,*

$$Q \leq \sum_{k \in \mathcal{K}} \log_2 \left(\frac{4}{\Delta_k} \right) \frac{16C_1}{\Delta_k} \log \frac{K}{\delta},$$

where \log is the natural logarithm, and \log_2 is the logarithmic function base 2.

Comparison with the query complexity lower bound in Theorem 2 shows that our upper bound in Theorem 1 is *tight up to logarithmic factors*. Compared to the classical oracle sample complexity upper bound of $O(\sum_{k \in \mathcal{K}} (1/\Delta_k)^2 \log(1/\delta))$ (Karnin, Koren, and Somekh 2013), the query complexity upper bound in Theorem 1 has a quadratic improvement in the dependence on $1/\Delta_k$ for each individual arm. In contrast, the strong quantum oracle sample complexity upper bound $\tilde{O}(\sqrt{\sum_k 1/\Delta_k^2} \log(1/\delta))$ (Wang et al. 2021) achieves an overall quadratic speedup. That is, as the first inequality of (1) shows, the coefficient of the query complexity lower bound of the weak quantum oracle is larger than that of the strong oracle, and is, in the worst case, \sqrt{K} times larger.

3.3 Lower Bounds for BAI with weak quantum oracle

Lastly, we present a query complexity lower bound for BAI with a weak quantum oracle. This lower bound describes the fundamental limits of the BAI task with a weak quantum oracle and is independent of the specific algorithm used.

Theorem 2 (Query complexity lower bound for best arm identification). *Given a quantum multi-armed bandits instance $\mathcal{I}_0 = \{\mu_1, \dots, \mu_K\}$ where $\mu_k \in (0, 1/2)$ for all k and $\mu_1 > \mu_2 \geq \mu_k$ for any $k \neq 1$, any algorithm that identifies the optimal arm with a given confidence $1 - \delta$, $\delta \in (0, 1)$ requires Q queries to the weak quantum oracle, where*

$$Q \geq \sum_{k \in \mathcal{K}} \frac{1}{4\Delta_k} \log \frac{1}{4\delta}.$$

Thus, to identify the best arm with confidence $1 - \delta$, it is *necessary* to pull *each* arm k at least $1/(4\Delta_k) \log 1/(4\delta)$ times. The proof of this lower bound consists of two steps: (1) apply the quantum hypothesis testing techniques to prove a lower bound for the task of two arm identification, and (2) extend the lower bound of the two-arm case to multiple arms via adapting the lower bound proof of the classical best arm identification. The detailed proof is presented in Appendix F.

First, Theorem 2 demonstrates that the query complexity of $\mathcal{Q}\text{-Elim}$, as established in Theorem 1, is near-optimal (up to some logarithm factors). Compared to the classical oracle's sample lower bound $\Omega\left(\sum_{k \in \mathcal{K}} \frac{1}{\Delta_k^2} \log \frac{1}{\delta}\right)$ (Mannor and Tsitsiklis 2004), our lower bound shows a linear dependence on $1/\Delta_k$ rather than quadratic. When compared to the strong quantum oracle's sample complexity lower bound $\Omega\left(\sqrt{\sum_k \frac{1}{\Delta_k^2}} (1 - \sqrt{\delta(1 - \delta)})\right)$ (Wang et al. 2021, Theorem 5), the weak oracle's query complexity lower bound has a larger coefficient, which can be up to \sqrt{K} times greater in the worst case. However, our lower bound improves on the dependence on δ , as $\log(1/\delta)$ is significantly larger than $1 - \sqrt{\delta(1 - \delta)}$ when δ is small.

4 BAI with m -Constrained Quantum Oracle

In this section, we present a partition algorithm for BAI with the m -constrained quantum oracle. We first present some key subroutines in Section 4.1 on quantum computing, and then present our algorithm in Section 4.2, followed by the algorithm's query complexity upper bound in Section 4.3, as well as a lower bound for any partition algorithms in Section 4.4.

4.1 Key Quantum Subroutines

Variable-Time Algorithm Construction The variable-time algorithm of Ambainis (2010); Wang et al. (2021) can be used to transform an m -constrained quantum oracle with a reward register $|\cdot\rangle_R$ into an oracle (VTA) that outputs a state with a flag register $|\cdot\rangle_F$ which distinguishes arms with large mean rewards from other arms. For an m -constrained oracle $\mathcal{O}_{\text{cons}}^{(S)}$ and a subset \mathcal{S} , VTA takes an interval $I = [a, b]$ with $0 < a < b < 1$ and a parameter $\alpha \in (0, 1)$ as inputs. It divides \mathcal{S} into three subsets: $\mathcal{S}_{\text{right}} := \{k \in \mathcal{S} : \mu_k \geq b - \frac{b-a}{8}\}$ (high rewards); $\mathcal{S}_{\text{left}} := \{k \in \mathcal{S} : \mu_k < b - \frac{b-a}{2}\}$ (low rewards); $\mathcal{S}_{\text{middle}} := \mathcal{S} \setminus (\mathcal{S}_{\text{right}} \cup \mathcal{S}_{\text{left}})$ (intermediate rewards). The output state is:

$$\text{VTA}(\mathcal{O}_{\text{cons}}^{(S)}, \mathcal{S}, I = [a, b], \alpha) : \frac{1}{\sqrt{m}} \sum_{k \in \mathcal{S}} |k\rangle_I |1\rangle_F \rightarrow \frac{1}{\sqrt{m}} \left(\sum_{k \in \mathcal{S}_{\text{right}}} |k\rangle_I |1\rangle_F + \sum_{k \in \mathcal{S}_{\text{left}}} |k\rangle_I |0\rangle_F + \sum_{k \in \mathcal{S}_{\text{middle}}} |k\rangle_I |\phi_k\rangle_F \right), \quad (6)$$

where $|\cdot\rangle_F$ indicates the subsets $\mathcal{S}_{\text{right}}$ and $\mathcal{S}_{\text{left}}$ with $|1\rangle_F$ and $|0\rangle_F$ respectively. Arms in $\mathcal{S}_{\text{middle}}$ are represented by $|\phi_k\rangle_F$, with specific states depending on α and the MAB instance. The probability of observing $|1\rangle_F$ is $p_{\text{good}} :=$

$\frac{1}{m} \left(|\mathcal{S}_{\text{right}}| + \sum_{k \in \mathcal{S}_{\text{middle}}} |\beta_k|^2 \right)$, where β_k depends on $|\phi_k\rangle_F$. The algorithm's pseudocode is in Appendix C.1.

Amplitude amplification (Amplify) and amplitude estimation (Estimate) The Amplify and Estimate are two fundamental quantum computing algorithms (Brassard et al. 2002). Amplify enhances the amplitude of a target basis state, while Estimate estimates the amplitude of that state. Since these algorithms are well-established, we omit their pseudocode and direct interested readers to Brassard et al. (2002) for details. In this work, we apply both algorithms with the VTA oracle in (6), using $|1\rangle_F$ as the target state. The performance of Amplify and Estimate in this context is discussed in Lemma 5 in Appendix C.1.

Good Ratio (GoodRatio) Subroutine The good ratio subroutine is based on the variable-time algorithm (VTA) and amplitude estimation (Estimate). It takes an m -constrained oracle $\mathcal{O}_{\text{cons}}^{(S)}$ for a subset of arms \mathcal{S} , an interval $I = [a, b]$, and a confidence parameter δ as inputs, and outputs an estimate of the ratio of "good arms" in \mathcal{S} , where the "good arms" are the arms with mean reward greater than a in the interval I . The subroutine is detailed in Algorithm 4 in Appendix C.2. Lemma 2 provides the subroutine's performance guarantees.

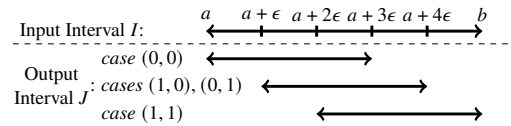
Lemma 2 (Performance of GoodRatio). *Given an interval $I = [a, b]$ and a confidence parameter $0 < \delta < 1$, there exists a $\text{GoodRatio}(\mathcal{O}_{\text{cons}}^{(S)}, \mathcal{S}, I = [a, b], \delta)$ subroutine which uses $O(G)$ queries to output an estimate \hat{p}_{good} of the "good arm" ratio p_{good} such that*

$$0.9 \left(p_{\text{good}} - \frac{0.1}{m} \right) < \hat{p}_{\text{good}} < 1.1 \left(p_{\text{good}} + \frac{0.1}{m} \right)$$

with probability at least $1 - \delta$, where the parameter $G := \sqrt{\frac{1}{(b-a)^2} + \frac{1}{|\mathcal{S}_{\text{right}}|} \sum_{k \in \mathcal{S}_{\text{left}} \cup \mathcal{S}_{\text{middle}}} \frac{1}{(b-\mu_k)^2}} \text{polylog} \left(\frac{m}{\delta(b-a)} \right)$.

Lemma 2 guarantees that GoodRatio provides a good estimate of the ratio of good arms in the subset \mathcal{S} with high probability and with in a reasonable number of queries.

Partition Shrink (PartShrink) Subroutine The partition shrink subroutine takes as input the m -constrained oracles $\mathcal{O}_{\text{cons}}^{(S)}$ for each subset \mathcal{S} in the partition set \mathfrak{B} , the partition set \mathfrak{B} itself, an interval I , and parameters $h \in \{1, 2\}$ and $\delta \in (0, 1)$. The parameter $h = 1$ (resp. $h = 2$) directs the algorithm to shrink the input interval I so that the best arm μ_1 (resp. the second best arm μ_2) lies inside the output interval J . Utilizing a technique from quantum ground state preparation (Lin and Tong 2020), PartShrink divides the input interval $I = [a, b]$ into five sub-intervals of equal length and outputs a new interval J consisting of three consecutive sub-intervals, as illustrated below:



Which case the output interval J above corresponds to depends on the input parameters and the mean reward

Algorithm 2 Q-Part: Partition Algorithm for BAI with coherent query constrained m

Input: full arm set \mathcal{K} , confidence parameter δ , constraint m
Initialize: $\delta \leftarrow \delta/2$, $I_1, I_2 \leftarrow [0, 1]$, $\delta' \leftarrow \delta$
1: Partition the full arm sets to $\lceil K/m \rceil$ subsets, each with m arms, together denoted as a set \mathfrak{B}
 \triangleright Stage (i): identify best arm subset
2: **while** $\min I_1 - \max I_2 < 2|I_1|$ or $|\mathfrak{B}| > 1$ **do**
3: $I_1 \leftarrow \text{PartShrink}\left((O_{\text{cons}}^{(S)})_{\forall S \in \mathfrak{B}}, \mathfrak{B}, I_1, 1, \delta'\right)$
4: $I_2 \leftarrow \text{PartShrink}\left((O_{\text{cons}}^{(S)})_{\forall S \in \mathfrak{B}}, \mathfrak{B}, I_2, 2, \delta'\right)$
5: **for** $S \in \mathfrak{B}$ **do**
6: **if** $\text{GoodRatio}\left(O_{\text{cons}}^{(S)}, I_1, \delta'\right) = 0$ **then**
 \triangleright If no good arm inside subset S
7: $\mathfrak{B} \leftarrow \mathfrak{B} \setminus S$ \triangleright Subset elimination
8: $\delta' \leftarrow \delta'/2$ \triangleright Halve confidence parameter
9: $\ell_1 \leftarrow \min I_1$, $\ell_2 \leftarrow \max I_2$
10: $\mathcal{S} \leftarrow \mathfrak{B}$ \triangleright Only remaining subset in \mathfrak{B}
 \triangleright Stage (ii): identify best arm
11: Construct variable-time quantum algorithm $\mathcal{A} \leftarrow \text{VTA}(O_{\text{cons}}^{(S)}, \mathcal{S}, I = [\ell_2, \ell_1], 0.01\delta)$
12: $k \leftarrow \text{Amplify}(\mathcal{A}, \delta')$
Output: arm k

of the arms in the interval I . We refer the detail to the PartShrink subroutine in Algorithm 5 in Appendix C.3. The performance guarantees are provided in Lemma 3.

Lemma 3 (Performance of PartShrink). *Given $h \in \{1, 2\}$, an interval $I = [a, b]$, and a confidence parameter $0 < \delta < 1$, supposing $\mu_h \in I$ and $|I| \geq \Delta_2/8$, there exists a $\text{PartShrink}\left((O_{\text{cons}}^{(S)})_{\forall S \in \mathfrak{B}}, \mathfrak{B}, I, h, \delta\right)$ subroutine which*

1. *outputs an interval J with $|J| = 3|I|/5$ such that $\mu_h \in J$ with a probability of at least $1 - \delta$, and*
2. *uses $O\left(\sum_{S \in \mathfrak{B}} \sqrt{\sum_{k \in S} \frac{1}{\Delta_k^2}} \text{polylog}\left(\frac{K}{m\delta\Delta_2}\right)\right)$ queries.*

Lemma 3 guarantees that PartShrink outputs an interval J containing the mean reward μ_h with high probability and in a reasonable number of queries. The proofs of Lemmas 2 and 3 are presented in Appendix E.1.

Next, we present the partition algorithm for BAI with the m -constrained quantum oracle that builds on the GoodRatio and PartShrink subroutines.

4.2 Algorithm Design

This section presents the partition algorithm (Q-Part in Algorithm 2) using the m -constrained quantum oracle. The algorithm partitions K arms into K/m subsets² and queries arms within each subset to find the optimal one.

Initially, Q-Part partitions the K arms into K/m subsets $\mathcal{S}_1, \dots, \mathcal{S}_{K/m}$ (Line 1), each containing m arms, and denote $\mathfrak{B} := \{\mathcal{S}_1, \dots, \mathcal{S}_{K/m}\}$. The algorithm has two main

²If K/m is not an integer, add n dummy arms (where $n < m$) to make $m \mid (K + n)$.

stages: (i) identifying the subset containing the optimal arm (Lines 2-8) and (ii) finding the best arm within that subset (Lines 9-12).

To find the optimal arm's subset, Q-Part uses an elimination process. It starts with all subsets in \mathfrak{B} and progressively removes those without the best arm until one remains. The algorithm maintains two intervals, I_1 and I_2 , both initialized to $[0, 1]$. Within the while loop (Line 2), PartShrink is applied to shrink the intervals I_1 and I_2 (Lines 3-4). Then, GoodRatio checks each remaining subset to see if it contains an arm with a mean reward in I_1 . If not, the subset is eliminated (Line 7). The loop ends when only one subset remains, and I_1 and I_2 are separated by a gap of at least $2|I_1|$ (i.e., $\min I_1 - \max I_2 \geq 2|I_1|$).

Upon completion of Stage (i), Q-Part identifies the subset containing the best arm, with I_1 containing the mean reward μ_1 of the best arm and I_2 containing the mean reward μ_2 of the second-best arm. The endpoints $\ell_1 = \min I_1$ and $\ell_2 = \max I_2$ separate the best arm from the rest (Line 9). To find the optimal arm, Q-Part uses a variable-time algorithm (VTA) in (6) with the remaining subset \mathcal{S} and interval $[\ell_2, \ell_1]$ as inputs (Line 11), which produces the expected output $\frac{1}{\sqrt{m}}(|k^*\rangle_I |1\rangle_F + \sum_{k \in \mathcal{S} \setminus \{k^*\}} |k\rangle_I |0\rangle_F)$. Amplify then determines the index of the optimal arm k^* (Line 12), which guarantees to output the best arm in the set \mathcal{S} with a probability of at least $1 - \delta'$.

4.3 Query Complexity Upper Bound for Partition Algorithm

We derive a query complexity upper bound for Q-Part (Algorithm 2), and its detail proof is deferred to Appendix E.2.

Theorem 3 (Query complexity upper bound for Q-Part of the m -constrained quantum oracle). *With confidence parameter $\delta \in (0, 1)$ and an arm partition \mathfrak{B} , the query complexity of Algorithm 2 is $O\left(\sum_{S \in \mathfrak{B}} \sqrt{\sum_{k \in S} \frac{1}{\Delta_k^2}} \text{polylog}\left(\frac{K}{\delta\Delta_2}\right)\right)$, where $\Delta_k = \mu_1 - \mu_k$ is the reward gap of arm k , and Δ_2 is the minimal reward gap.*

As Δ_2 is the smallest reward gap, Theorem 3 simplifies the upper bound to $\tilde{O}((K/\sqrt{m})\Delta_2^{-1})$. Thus, a smaller m (better coherence) reduces the query complexity. When $m = 1$ (weak quantum oracle), Q-Part's complexity is $\tilde{O}(\sum_{k \in \mathcal{K}} \Delta_k^{-1} \log \frac{1}{\delta})$, which matches Q-Elim's bound for a weak oracle (Theorem 2). However, Q-Elim's bound $O(\log \frac{1}{\delta} \log \frac{K}{\delta})$ is better than Q-Part's polylogarithmic factor $O(\text{polylog} \frac{K}{\delta\Delta_2})$ (at most $\log^3 \frac{K}{\delta\Delta_2}$) because Q-Elim's parameters are optimized for weak oracles. When $m = K$ (strong quantum oracle), Q-Part reduces to the algorithm by Wang et al. (2021), as no further partitioning is needed ($\mathfrak{B} = \{\mathcal{K}\}$). For $1 < m < K$, Q-Part's complexity lies between that of Wang et al. (2021)'s strong oracle and Q-Elim's weak oracle (see (1)).

4.4 Query Complexity Lower Bounds for the m -Constrained Quantum Oracle

In Section 4.4, we establish lower bounds to demonstrate the tightness and optimality of the Q-Part algorithm. The key

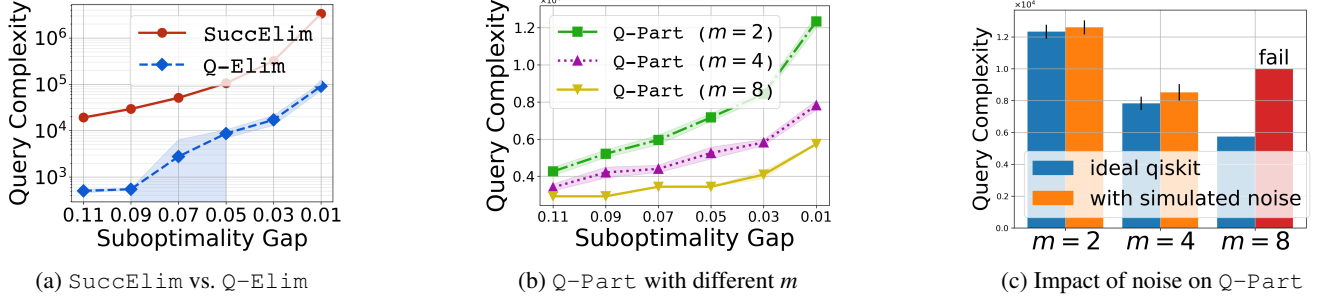


Figure 1: Performance evaluation of Q-Elim and Q-Part. The results for Q-Part are based on taking the constant multiplicative prefactor from Lemma 5 to be 1. In reality this constant may be larger than 1 and thus the results for Q-Part and Q-Elim are not directly comparable. Figure 1c is conducted for gap = 0.01 in a simulated 127-qubit quantum computer.

challenge is proving the lower bound with the outer summation over all subsets in \mathfrak{B} (i.e., $\sum_{S \in \mathfrak{B}}$). This summation indicates that queries on each subset are “orthogonal”, meaning information gained from one subset does not overlap with others. To address this, we define a class of partition algorithms for m -constrained oracles, which ensures that queries on one subset of arms cannot be used to infer information about arms in any of the other subsets. We then derive a lower bound for any partition algorithm, as stated in Theorem 4, with a proof provided in Appendix G.2.

Definition 1 (Partition algorithm class). *A partition algorithm for BAI with the m -constrained oracle is one that partitions the full arm set into several subsets at initialization, each with at most m arms, and always follows this fixed partition when querying arms during algorithm execution.*

Theorem 4 (Query complexity lower bound for m -constrained oracle). *To identify the best arm with a probability of at least $1 - \delta$ with the m -constrained oracle with parameter m , any partition algorithm needs to spend at least the following number of queries, $\Omega(\sum_{S \in \mathfrak{B}} \sqrt{\sum_{k \in S} 1/\Delta_k^2})$, where \mathfrak{B} is the partition of arms.*

Note that Q-Part (Algorithm 2) belongs to this partition algorithm class, and if the arm partition \mathfrak{B} in the lower bound is the same as the one chosen in Q-Part, then this lower bound matches the upper bound for Q-Part in Theorem 3 up to some logarithmic factors. This implies that the bounds in both Theorems 3 and 4 are tight, and Q-Part is near-optimal within the partition algorithm class.

5 Qiskit-based Simulation

We compare the quantum algorithms Q-Elim (for the weak quantum oracle) and Q-Part (for the m -constrained quantum oracle) with the classical successive elimination SuccElim (Even-Dar et al. 2006).

We set $\delta = 0.1$ and $K = 8$ arms with mean rewards ranging from $0.99 - (k - 1) \times \Delta$ (where $k \in \{1, \dots, K\}$) and vary Δ from 0.11 to 0.01 in steps of 0.02 to analyze its effect on query complexity. Details of the Qiskit implementation are in Appendix H, and the code is provided in

the supplementary material. We implement Q-Elim with $m = 1$ and Q-Part with $m = 2, 4$, and 8. For $m = 8$, the m -constrained oracle is equivalent to the strong quantum oracle. The default confidence parameter for SuccElim is $c = 4$ (Even-Dar et al. 2006). Results, averaged over 50 trials, are shown in Figure 1.

Figure 1a (with y-axis in log-scale) shows that Q-Elim outperforms SuccElim, demonstrating the benefits of quantum information with the weak oracle. As Δ decreases, SuccElim’s query complexity increases faster than Q-Elim’s, validating the quantum improvement of dependence on Δ from Δ^{-2} to Δ^{-1} (see Appendix H for curve-fitting). Figure 1b compares Q-Part’s performance for $m = 2, 4, 8$ as Δ varies. Increasing m improves performance, confirming the advantage of quantum parallelism predicted by the $\tilde{O}((K/\sqrt{m})\Delta^{-1})$ bound from Theorem 3.

We also assess the impact of noise using Qiskit’s simulation of IBM’s 127-qubit device. Figure 1c shows that Q-Elim’s performance decreases by 2.38% and 8.09% for $m = 2$ and $m = 4$, respectively. For $m=8$, Q-Part fails due to high noise, as the increased qubit and gate requirements exceed practical limits, impairing the algorithm’s functionality. This highlights the importance to study the restrictive m -constrained quantum oracles under a noisy environment.

6 Conclusion

In this paper, we explore the best arm identification (BAI) problem using weak and m -constrained quantum oracles. We introduce the m -constrained oracle, which generalizes both the weak oracle ($m = 1$) and the strong oracle ($m = K$). Our quantum algorithms, Q-Elim for the weak oracle and Q-Part for the constrained oracle, offer significant improvements over classical methods. Specifically, Q-Elim achieves a quadratic speedup at the arm level due to quantum parallelism, while Q-Part provides a quadratic speedup at the subset level due to quantum entanglement. We establish query complexity lower bounds for both quantum BAI problems that align with our upper bounds, indicating near-optimal performance. Our experiments using Qiskit confirm these theoretical results.

Acknowledgments

The work of Mohammad Hajiesmaili is supported by NSF CNS-2325956, CAREER-2045641, CPS-2136199, CNS-2102963, and CNS-2106299. The work of John C.S. Lui is supported in part by SRFS2122-4S02. The work of Don Towsley is supported in part by the NSF grant CNS-1955744, NSF-ERC Center for Quantum Networks grant EEC-1941583, and DOE Grant AK000000001829.

References

- Ambainis, A. 2000. Quantum lower bounds by quantum arguments. In *Proceedings of the thirty-second annual ACM symposium on Theory of computing*, 636–643.
- Ambainis, A. 2010. Variable time amplitude amplification and a faster quantum algorithm for solving systems of linear equations. *arXiv preprint arXiv:1010.4458*.
- 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.
- Audibert, J.-Y.; Bubeck, S.; and Munos, R. 2010. Best arm identification in multi-armed bandits. In *COLT*, 41–53. Cite-seer.
- Auer, P.; and Ortner, R. 2010. UCB revisited: Improved regret bounds for the stochastic multi-armed bandit problem. *Periodica Mathematica Hungarica*, 61(1-2): 55–65.
- Azuma, K.; Economou, S. E.; Elkouss, D.; Hilaire, P.; Jiang, L.; Lo, H.-K.; and Tzitrin, I. 2022. Quantum repeaters: From quantum networks to the quantum internet. *arXiv preprint arXiv:2212.10820*.
- Barrachina-Muñoz, S.; and Bellalta, B. 2017. Learning optimal routing for the uplink in LPWANs using similarity-enhanced e-greedy. In *2017 IEEE 28th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC)*, 1–5. IEEE.
- Brahmachari, S.; Lumbreras, J.; and Tomamichel, M. 2023. Quantum contextual bandits and recommender systems for quantum data. *arXiv preprint arXiv:2301.13524*.
- Brassard, G.; Hoyer, P.; Mosca, M.; and Tapp, A. 2002. Quantum amplitude amplification and estimation. *Contemporary Mathematics*, 305: 53–74.
- Bubeck, S.; Munos, R.; and Stoltz, G. 2011. Pure exploration in finitely-armed and continuous-armed bandits. *Theoretical Computer Science*, 412(19): 1832–1852.
- Buchholz, S.; Kübler, J. M.; and Schölkopf, B. 2023. Multi armed bandits and quantum channel oracles. *arXiv preprint arXiv:2301.08544*.
- Cacciapuoti, A. S.; Caleffi, M.; Tafuri, F.; Cataliotti, F. S.; Gherardini, S.; and Bianchi, G. 2019. Quantum internet: Networking challenges in distributed quantum computing. *IEEE Network*, 34(1): 137–143.
- Casalé, B.; Di Molfetta, G.; Kadri, H.; and Ralaivola, L. 2020. Quantum bandits. *Quantum Machine Intelligence*, 2(1): 1–7.
- Childs, A. M. 2017. Lecture notes on quantum algorithms. *Lecture notes at University of Maryland*.
- Childs, A. M.; Kothari, R.; and Somma, R. D. 2017. Quantum algorithm for systems of linear equations with exponentially improved dependence on precision. *SIAM Journal on Computing*, 46(6): 1920–1950.
- Cho, B.; Xiao, Y.; Hui, P.; and Dong, D. 2022. Quantum bandit with amplitude amplification exploration in an adversarial environment. *arXiv preprint arXiv:2208.07144*.
- Chow, J.; Dial, O.; and Gambetta, J. 2021. IBM Quantum breaks the 100-qubit processor barrier. *IBM Research Blog*.
- Chuang, I. L.; and Yamamoto, Y. 1995. Simple quantum computer. *Physical Review A*, 52(5): 3489.
- Dai, Z.; Lau, G. K. R.; Verma, A.; Shu, Y.; Low, B. K. H.; and Jaillet, P. 2023. Quantum Bayesian Optimization. In *Thirty-seventh Conference on Neural Information Processing Systems*.
- Einstein, A.; Podolsky, B.; and Rosen, N. 1935. Can quantum-mechanical description of physical reality be considered complete? *Physical review*, 47(10): 777.
- Even-Dar, E.; Mannor, S.; and Mansour, Y. 2002. PAC bounds for multi-armed bandit and Markov decision processes. In *COLT*, volume 2, 255–270. Springer.
- Even-Dar, E.; Mannor, S.; Mansour, Y.; and Mahadevan, S. 2006. Action Elimination and Stopping Conditions for the Multi-Armed Bandit and Reinforcement Learning Problems. *Journal of machine learning research*, 7(6).
- Ganguly, B.; Wu, Y.; Wang, D.; and Aggarwal, V. 2023. Quantum Computing Provides Exponential Regret Improvement in Episodic Reinforcement Learning. *arXiv preprint arXiv:2302.08617*.
- Giovannetti, V.; Lloyd, S.; and Maccone, L. 2008. Quantum random access memory. *Physical review letters*, 100(16): 160501.
- Grinko, D.; Gacon, J.; Zoufal, C.; and Woerner, S. 2021. Iterative quantum amplitude estimation. *npj Quantum Information*, 7(1): 52.
- Grover, L. K. 1996. A fast quantum mechanical algorithm for database search. In *Proceedings of the Twenty-eighth Annual ACM Symposium on Theory of Computing*, 212–219.
- Grover, L. K.; and Radhakrishnan, J. 2005. Is partial quantum search of a database any easier? In *Proceedings of the seventeenth annual ACM symposium on Parallelism in algorithms and architectures*, 186–194.
- Hamoudi, Y. 2021. Quantum Sub-Gaussian Mean Estimator. In *29th Annual European Symposium on Algorithms (ESA 2021)*. Schloss Dagstuhl-Leibniz-Zentrum für Informatik.
- Holevo, A. S. 2003. *Statistical structure of quantum theory*, volume 67. Berlin: Springer Science & Business Media.
- Kargin, V. 2005. ON THE CHERNOFF BOUND FOR EFFICIENCY OF QUANTUM HYPOTHESIS TESTING. *The Annals of Statistics*, 33(2): 959–976.
- Karnin, Z.; Koren, T.; and Somekh, O. 2013. Almost optimal exploration in multi-armed bandits. In *International Conference on Machine Learning*, 1238–1246. PMLR.

- Kaye, P.; Laflamme, R.; and Mosca, M. 2006. *An introduction to quantum computing*. Oxford: OUP Oxford.
- Klauck, H.; Špalek, R.; and De Wolf, R. 2007. Quantum and classical strong direct product theorems and optimal time-space tradeoffs. *SIAM Journal on Computing*, 36(5): 1472–1493.
- Lattimore, T.; and Szepesvári, C. 2020. *Bandit algorithms*. Cambridge: Cambridge University Press.
- Li, T.; and Zhang, R. 2022. Quantum Speedups of Optimizing Approximately Convex Functions with Applications to Logarithmic Regret Stochastic Convex Bandits. *Advances in Neural Information Processing Systems*, 35: 3152–3164.
- Li, X.-H.; Deng, F.-G.; and Zhou, H.-Y. 2008. Efficient quantum key distribution over a collective noise channel. *Physical Review A*, 78(2): 022321.
- Lin, L.; and Tong, Y. 2020. Near-optimal ground state preparation. *Quantum*, 4: 372.
- Liu, M.; Li, Z.; Wang, X.; and Lui, J. C. 2024. LinkSelFiE: Link Selection and Fidelity Estimation in Quantum Networks. In *IEEE INFOCOM 2024-IEEE Conference on Computer Communications*, preprint. IEEE.
- Lumbreras, J.; Haapasalo, E.; and Tomamichel, M. 2022. Multi-armed quantum bandits: Exploration versus exploitation when learning properties of quantum states. *Quantum*, 6: 749.
- Mannor, S.; and Tsitsiklis, J. N. 2004. The sample complexity of exploration in the multi-armed bandit problem. *Journal of Machine Learning Research*, 5(Jun): 623–648.
- Montanaro, A. 2015. Quantum speedup of Monte Carlo methods. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 471(2181): 20150301.
- Naruse, M.; Chauvet, N.; Uchida, A.; Drezet, A.; Bachelier, G.; Huant, S.; and Hori, H. 2019. Decision making photonics: solving bandit problems using photons. *IEEE Journal of Selected Topics in Quantum Electronics*, 26(1): 1–10.
- Nielsen, M. A.; and Chuang, I. 2002. Quantum computation and quantum information.
- Ohno, H. 2023. Quantum greedy algorithms for multi-armed bandits. *Quantum Information Processing*, 22(2): 101.
- Qiskit contributors. 2023. Qiskit: An Open-source Framework for Quantum Computing.
- Robbins, H. 1952. Some aspects of the sequential design of experiments.
- Wan, Z.; Zhang, Z.; Li, T.; Zhang, J.; and Sun, X. 2023. Quantum Multi-Armed Bandits and Stochastic Linear Bandits Enjoy Logarithmic Regrets. In *Proceedings of the AAAI Conference on Artificial Intelligence*.
- Wang, D.; You, X.; Li, T.; and Childs, A. M. 2021. Quantum exploration algorithms for multi-armed bandits. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, 10102–10110.
- Wehner, S.; Elkouss, D.; and Hanson, R. 2018. Quantum internet: A vision for the road ahead. *Science*, 362(6412): eaam9288.
- Wu, Y.; Guan, C.; Aggarwal, V.; and Wang, D. 2023. Quantum Heavy-tailed Bandits. *arXiv preprint arXiv:2301.09680*.
- Zhong, H.; Hu, J.; Xue, Y.; Li, T.; and Wang, L. 2023. Provably Efficient Exploration in Quantum Reinforcement Learning with Logarithmic Worst-Case Regret. *arXiv preprint arXiv:2302.10796*.
- Zhou, Y.; Chen, X.; and Li, J. 2014. Optimal PAC multiple arm identification with applications to crowdsourcing. In *International Conference on Machine Learning*, 217–225. PMLR.