Quantum Machine Intelligence (2024) 6:58
https://doi.org/10.1007/s42484-024-00189-6

RESEARCH ARTICLE

Quantum contextual bandits and recommender systems for quantum data

Shrigyan Brahmachari[1,2] · **Josep Lumbreras**[2] · **Marco Tomamichel**[2,3]

Received: 13 March 2024 / Accepted: 31 July 2024 / Published online: 12 September 2024

Abstract
We study a recommender system for quantum data using the linear contextual bandit framework. In each round, a learner receives an observable (the context) and has to recommend from a finite set of unknown quantum states (the actions) which one to measure. The learner has the goal of maximizing the reward in each round, that is the outcome of the measurement on the unknown state. Using this model, we formulate the low energy quantum state recommendation problem where the context is a Hamiltonian and the goal is to recommend the state with the lowest energy. For this task, we study two families of contexts: the Ising model and a generalized cluster model. We observe that if we interpret the actions as different phases of the models, then the recommendation is done by classifying the correct phase of the given Hamiltonian, and the strategy can be interpreted as an online quantum phase classifier.

Keywords Quantum online learning · Reinforcement learning · Multi-armed stochastic bandits · Adaptive quantum strategies · Recommender systems

1 Introduction

Recommender systems are a class of online reinforcement learning algorithms that interact sequentially with an environment suggesting relevant items to a user. During the last decade, there has been an increasing interest in online recommendation techniques due to the importance of advertisement recommendation for e-commerce websites or the rise of movies and music streaming platforms (Gomez-Uribe and Hunt 2016; Dragone et al. 2019). Among different settings for recommender systems, in this work, we focus on the contextual bandit framework applied to the recommendation of quantum data. The contextual bandit problem is a variant of the multi-armed bandit problem where a learner at each round receives a context and given a set of actions (also called arms) has to decide the best action using the context information. After selecting an action, the learner will receive a reward, and for the next rounds, they will use the previous information of contexts and rewards in order to make their future choices. As in the classical multi-armed bandit problem, the learner has to find a balance between exploration and exploitation; exploration refers to trying different actions in order to eventually learn the ones with the highest reward, and exploitation refers to selecting the actions that apparently will give the highest reward immediately. For a comprehensive review of bandit algorithms, we refer to the book by Lattimore and Szepesvári (2020). Some real-life applications (Bouneffouf et al. 2020) of bandit include clinical trials (Durand et al. 2018), dynamic pricing (Cohen et al. 2020), advertisement recommendation (Tang et al. 2013), or online recommender systems (Li et al. 2010; McInerney et al. 2018). As an example, in Li et al. (2010), a news article recommender system was considered where the context is the user features, the actions are the articles to recommend, and the reward is modeled as a binary outcome indicating that the user clicks or not on the recommended article.

✉ Josep Lumbreras
josep.lumbreras@u.nus.edu

Shrigyan Brahmachari
sb818@duke.edu

1 Department of Electrical and Computer Engineering, Duke University Pratt School of Engineering, Box 90291, Durham, NC 27708, USA

2 Centre for Quantum Technologies, National University of Singapore, 3 Science Drive 2, Block S15, 117543, Singapore, Singapore

3 Department of Electrical and Computer Engineering, Faculty of Engineering, National University of Singapore, Singapore, Singapore

Quantum algorithms for the classical multi-armed bandit problem have been studied for the settings of best-arm identification (Casalé et al. 2020; Wang et al. 2021), exploration-exploitation with stochastic environments (Wan et al. 2022) (uncorrelated and linear correlated actions), and adversarial environments (Cho et al. 2022). Also, a quantum neural network approach was considered in Hu et al. (2019) for a simple best-arm identification problem. A quantum algorithm for a classical recommender system was considered in Kerenidis and Prakash (2016), claiming an exponential speedup over known classical algorithms, but later in Tang (2019), it was proven that the price of the speedup comes from the assumptions of the quantum state preparation part and argued that under related classical assumptions, a classical algorithm can also achieve the speedup. There are other more general reinforcement learning frameworks beyond bandits where actions affect the rewards in the long term such as Markov decision process. The quantum generalization of this framework has been considered in Barry et al. (2014); Ying et al. (2021), and although our model of study falls into their class, we can derive more concrete results since we study a specific setting.

We are interested in studying a recommender system for quantum data that is modeled by a set of unknown quantum processes—which is called the *environment*, and a set of tasks to perform using these quantum processes—which is called a *context set* (Fig. 1). A learner interacts sequentially with the environment receiving at each round a task from the context set and then choosing the best quantum process to perform this task. For example, we could model the environments as a set of noisy quantum computers, the context set as a set of different quantum algorithms, and then at each round, the learner is given a quantum algorithm to run, and their goal is to recommend the best quantum computer to do this task. We note that this model exemplifies the bandit exploration-exploitation trade-off since the learner has to try (explore) the different quantum computers in order to decide the best one but at the same time has to choose the best one (exploitation) to perform the task. This trade-off is interesting in a practical scenario because it captures settings where online decisions are important and or they have some associated cost that makes the learner always try to perform optimally. In our example, one could think that using a quantum computer costs money for the learner, so at each stage, they always want to select the ones that will output the best solutions.

In our work, we extend the setting considered in Lumbreras et al. (2022) where they studied the exploration-exploration trade-off of learning properties of quantum states. In our model, the environment is a set of unknown quantum states, the context set is a (finite or infinite) set of observables, and at each round, the learner receives an observable and has to perform a measurement on one of the unknown quantum states (the recommendation) aiming to maximize its outcome. We define this problem as the *quantum contextual bandit* (QCB), and we note that it falls into the class of linear contextual bandits (Abe et al. 2003; Auer 2003; Chu et al. 2011). The QCB is the basic framework where we formulate our recommender system for quantum data. We use as a figure of merit the regret, which is the cumulative sum of the difference between the expected outcome of the best and selected action at each round. We note that although our model is a subset of the problem of linear contextual bandit, it still suffers the same regret scaling in a worst-case scenario (see Sect. 3). The main goal of this work is to show how the bandit framework can be applied for recommender systems of quantum data rather than focus-

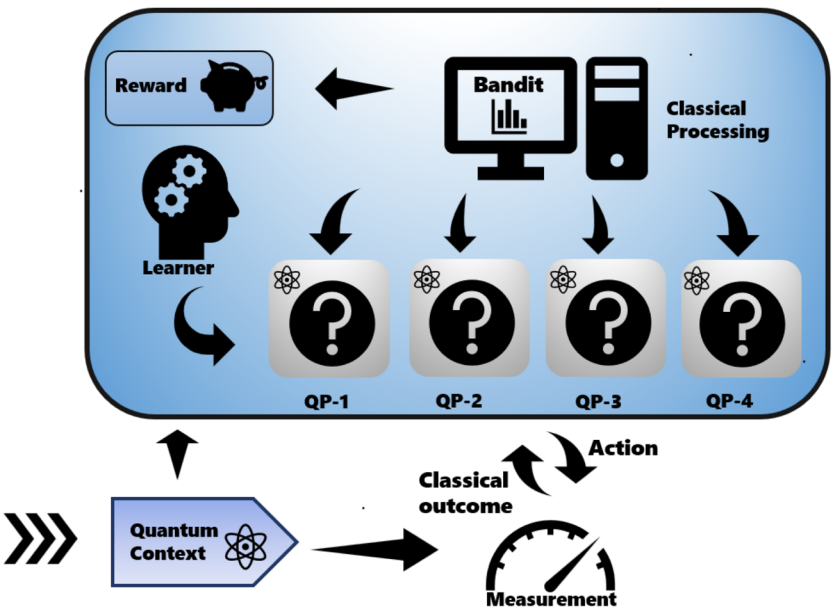

Fig. 1 Sketch of a recommender system for quantum data. The learner receives sequentially quantum contexts and feed them to the classical processing system. The context is also fed to the measurement system. The classical processing system uses the information about the context to pick one of the quantum processes (no information regarding these processes are known besides from measurements). The chosen quantum process is applied to the measurement system, and the measurement outcome is fed to the classical processing and is added to the cumulative reward

ing on finding particular settings where the regret has a better scaling. Finding a strategy that minimizes regret implies finding the mentioned balance between exploration-exploitation of the different actions. As a concrete recommendation task captured by the QCB model, we consider the *low energy quantum state recommendation problem*. In this problem, at each round, the learner receives a quantum Hamiltonian and has to recommend from the environment the state with the lowest energy. The ground state preparation problem is an important ingredient of NISQ algorithms (Bharti et al. 2022), and our model could be useful in order to implement an online recommendation algorithm that helps the learner choose the best ansatz for their energy minimization task when they have multiple problems to solve. One of the advantages of using bandit algorithms for this task is that they do not need to reconstruct the whole d-dimensional state, just the relevant part for the recommendation which depends on the structure of the context set. In order to do that, we combine a Gram-Schmidt procedure with classical linear bandit strategies. This allows our algorithm to store low-dimensional approximations of the unknown quantum states without prior knowledge of the context set. We remark that the goal of our problem is slightly different than the usual ground state finding problem since we focus on the recommendation part, where we just want to recommend a "good" enough state.

We also perform some numerical studies of the scaling of the regret for the cases where the context set is an Ising model and a generalized cluster model studied in Verresen et al. (2017). For these models, we propose unknown actions for the algorithm corresponding to ground states located at different phases, and then for each context received by the algorithm, we associate each action to a different phase and we reproduce a ground state phase diagram. We observe that the recommendation of the algorithm is done approximately by classifying the different phases of the studied models, and we are able to clearly distinguish them in the phase diagram.

The rest of the paper is organized as follows: in Sect. 2, we establish the mathematical model for the quantum contextual bandit and then define the notation used throughout the paper; in the next section, Sect. 3, we prove the lower bound on a performance metric (expected regret, which we define in Sect. 2) over all possible algorithms. In Sect. 4, we review the linear Upper Confidence Bound algorithm. In Sect. 5, we describe the low-energy recommendation system and adapt the LinUCB algorithm to this setting. We illustrate the efficiency of the algorithm through simulations of different context sets.

2 The model

First, we introduce some notation in order to define our model and present our results. We define $[T] = \{1, ..., T\}$ for $T \in \mathbb{N}$. Let $\mathcal{S}_d = \{\rho \in \mathbb{C}^{d\times d} : \rho \geq 0 \wedge \mathrm{Tr}(\rho) = 1\}$ denote the set of positive semi-definite operators with unit trace, i.e., *quantum states* that act on a d-dimensional Hilbert space $\mathbb{C}^d$. Moreover, *observables* are Hermitian operators acting on $\mathbb{C}^d$, collected in the set $\mathcal{O}_d = \{O \in \mathbb{C}^{d\times d} : O^\dagger = O\}$. We denote real d-dimensional column vectors as $\mathbf{v}$ and the inner product of two of them $\mathbf{u}, \mathbf{v} \in \mathbb{R}^d$ as $\mathbf{u}^\top\mathbf{v}$, where $\mathbf{u}^\top$ denotes the transpose of $\mathbf{u}$ that is a row vector. We use $\|\cdot\|_2$ in order to denote the 2-norm of a real vector. For a n-qubit system with Hilbert space dimension $d = 2^n$, we denote X_i, Y_i, and Z_i the x, y, z Pauli operators acting on the i-th qubit ($1 \leq i \leq n$). A Pauli observable can be expressed as the n-fold tensor product of the 2×2 Pauli matrices, i.e., it is an element of the set $\{I, X, Y, Z\}^{\otimes n}/I_{4^n\times 4^n}$. Note that there are $4^n - 1$ such observables, and each of them is orthogonal and therefore forms a basis, which we refer to as the *Pauli basis*.

The definition of our model takes some of the conventions used for the multi-armed quantum bandit (MAQB) problem (Lumbreras et al. 2022).

Definition 1 (Quantum contextual bandit) Let $d \in \mathbb{N}$. A d-dimensional *quantum contextual bandit* is given by a set of observables $\mathcal{C} = \{O_c\}_{c\in\Omega_\mathcal{C}} \subseteq \mathcal{O}_d$ that we call the *context set*, $(\Omega_\mathcal{C}, \Sigma_\mathcal{C})$ is a measurable space, and $\Sigma_\mathcal{C}$ is a σ-algebra of subsets of $\Omega_\mathcal{C}$. The bandit is in an *environment*, a finite set of quantum states $\gamma = \{\rho_1, \rho_2, \cdots, \rho_k\} \subset \mathcal{S}_d$, that it is unknown. The quantum contextual bandit problem is characterized by the tuple $(\mathcal{C}, \gamma)$.

Given the environment γ such that $|\gamma| = k$, we define the *action set* $\mathcal{A} = \{1, ..., k\}$ as the set of indices that label the quantum states $\rho_i \in \gamma$ in the environment. For every observable $O_c \in \mathcal{C}$, the spectral decomposition is given by

$$O_c = \sum_{i=1}^{d_c} \lambda_{c,i}\,\Pi_{c,i}, \tag{1}$$

where $\lambda_{c,i} \in \mathbb{R}$ denote the $d_c \leq d$ distinct eigenvalues of O_c and $\Pi_{c,i}$ are the orthogonal projectors on the respective eigenspaces. For each action $a \in \mathcal{A}$, we define the reward distribution with outcome $R \in \mathbb{R}$ as the conditional probability distribution associated with performing a measurement using O_c on ρ_a given by Born's rule

$$\begin{aligned} &Pr\,[R = r|A = a, O = O_c] \\ &\quad = P_{\rho_a}(r|a, c) = \begin{cases} \mathrm{Tr}(\rho_a\Pi_{c,i}) \text{ if } r = \lambda_{c,i}, \\ 0 \text{ else.} \end{cases} \end{aligned} \tag{2}$$

With the above definitions, we can explain the learning process. The learner interacts sequentially with the QCB over T rounds such that for every round $t \in [T]$:

1. The learner receives a context $O_{c_t} \in \mathcal{C}$ from some (possibly unknown) probability measure $P_{\mathcal{C}} : \Sigma \to [0, 1]$ over the set $\Omega_{\mathcal{C}}$.
2. Using the previous information of received contexts, actions played, and observed rewards, the learner chooses an action $A_t \in \mathcal{A}$.
3. The learner uses the context O_{c_t} and performs a measurement on the unknown quantum state ρ_{A_t} and receives a reward R_t sampled according to the probability distribution (2).

We use the index $c_t \in [m]$ to denote the observable O_{c_t} received at round $t \in [T]$. The strategy of the learner is given by a set of (conditional) probability distributions $\pi = \{\pi_t\}_{t\in\mathbb{N}}$ (policy) on the action index set $[k]$ of the form

$$\pi_t(a_t|a_1, r_1, c_1, ..., a_{t-1}, r_{t-1}, c_{t-1}, c_t), \tag{3}$$

defined for all valid combinations of actions, rewards, and contexts $(a_1, r_1, c_1, ..., a_{t-1}, r_{t-1}, c_{t-1})$ up to time $t - 1$. Then, if we run the policy π on the environment γ over $T \in \mathbb{N}$ rounds, we can define a joint probability distribution over the set of actions, rewards, and contexts as

$$P_{\gamma,\mathcal{C},\pi}(a_1, X_1, C_1, ..., a_T, X_T, C_T) = \int_{C_T}\int_{X_T}\cdots\int_{C_1}\int_{X_1}\prod_{t=1}^{T}\pi_t(a_t|a_1, r_1, c_1, ..., a_{t-1}, r_{t-1}, c_{t-1})\times \\ \times P_{\mathcal{C}}(dc_1)P_{\rho_{a_1}}(dr_1|a_1, c_1)\cdots P_{\mathcal{C}}(dc_T)P_{\rho_{a_T}}(dr_T|a_T, c_T). \tag{4}$$

Thus, the conditioned expected value of reward R_t is given by

$$\mathbb{E}_{\gamma,\mathcal{C},\pi}[R_t|A_t = a, O_{c_t} = O_c] = \mathrm{Tr}(\rho_a O_c), \tag{5}$$

where $\mathbb{E}_{\gamma,\mathcal{C},\pi}$ denotes the expectation value over the probability distribution Eq.4. The goal of the learner is to maximize its expected cumulative reward $\sum_{t=1}^{T}\mathbb{E}_{\gamma,\mathcal{C},\pi}[R_t]$ or equivalently minimizing the *cumulative expected regret*

$$\mathrm{Regret}_T^{\gamma,\mathcal{C},\pi} = \sum_{t=1}^{T}\mathbb{E}_{\gamma,\mathcal{C},\pi}\left[\max_{\rho_i\in\gamma}\mathrm{Tr}(\rho_i O_{c_t}) - R_t\right]. \tag{6}$$

For a given action $a \in \mathcal{A}$ and context $O_c \in \mathcal{C}$, the *suboptimality gap* is defined as

$$\Delta_{a,O_c} = \max_{i\in\mathcal{A}}\mathrm{Tr}(\rho_i O_c) - \mathrm{Tr}(\rho_a O_c). \tag{7}$$

Note that the learner could try to learn the distribution of contexts $P_{\mathcal{C}}$; however, this will not make a difference in minimizing the regret. The strategy of the learner has to be able to learn the relevant part of the unknown states $\{\rho_a\}_{a=1}^{k}$ that depend on the context set and at the same time balance the tradeoff between exploration and exploitation. We note that it is straightforward to generalize the above setting to continuous sets of contexts $\mathcal{C}$. In order to do that, we need a well-defined probability distribution $P_{\mathcal{C}}(O)dO$ over the context set $\mathcal{C}$.

3 Lower bound

In this section, we derive a lower bound for the cumulative expected regret by finding, for any strategy, a QCB that is hard to learn. Our regret lower bound proof for the QCB model relies on a reduction to a classical multi-armed stochastic bandit given in Theorem 5.1 in Auer et al. (2003). Now, we briefly review the multi-armed stochastic bandit problem.

The *multi-armed stochastic bandit* problem is defined by a discrete set of probability distributions $\nu = (P_a : a \in [k])$ that is called the environment, and μ_i is the mean of the probability distribution P_i for $i \in [k]$. The learner interacts sequentially with the bandit selecting at each round $t \in [T]$ an action $a \in [k]$ and sampling a reward R_t distributed accordingly to P_a. The expected cumulative regret is defined as

$$\mathrm{Regret}_T^{\nu,\pi} = \sum_{t=1}^{T}\max_{a\in[k]}\mu_a - \mathbb{E}_{\nu,\pi}[R_t], \tag{8}$$

where π and $\mathbb{E}_{\nu,\pi}$ are both defined analogously from the definitions of the previous section accordingly to this model. It is important to remark that in this setting, the actions are independent, meaning that when the learner samples from one action, then it cannot use this information to learn about other actions.

Using the above model, we describe the multi-armed stochastic bandit studied in Theorem 5.1 in Auer et al. (2003). The bandit is constructed defining an environment $\nu = (P_a : a \in [k])$ for $k \geq 2$ such that P_a are Bernoulli distributions for all $a \in [k]$ with outcomes $\{l_1, l_2\}$. Then, we set the distributions as follows: we choose an index $i \in [k]$ uniformly at random and assign $P_i(R = l_1) = \frac{1+\Delta}{2}$ for some $\Delta > 0$ and $P_a(R = l_1) = \frac{1}{2}$ for $a \neq i$. Thus, there is a unique best action corresponding to $a = i$. Then, choosing $\Delta = \epsilon\sqrt{\frac{k}{n}}$ for some small positive constant ϵ, for $n \geq k$, the expected regret for any strategy will scale as

$$\mathrm{Regret}_T^{\nu,\pi} = \Omega(\sqrt{kT}). \tag{9}$$

Theorem 2 *Consider a quantum contextual bandit with underlying dimension $d = 2^n$ and $n \in \mathbb{N}$, context size $c \geq 1$ and $k \geq 2$ actions. Then, for any strategy π, there exists a*

context set $\mathcal{C}$, $|\mathcal{C}| = c$, a probability distribution over the context set $\mathcal{C}$ $P_{\mathcal{C}}$ and an environment $\gamma \in \mathcal{S}_d$ such that for the QCB defined by $(\mathcal{C}, \gamma)$, the expected cumulative regret will scale as

$$\text{Regret}_T^{\gamma,\mathcal{C},\pi} = \Omega\left(\sqrt{kT} \cdot \min\left\{d, \sqrt{c}\right\}\right), \tag{10}$$

for $T \geq k \min\{c, d^2\}$.

Proof We use a similar technique to Abe et al. (2003); Chu et al. (2011) in order to analyze the regret by dividing the problem into subsets of independent rounds. We start dividing the T rounds in $c' = \min\{c, d^2 - 1\}$ groups of $T' = \lfloor \frac{T}{c'} \rfloor$ elements. We say that time step t belongs to group s if $\lfloor \frac{t}{T'} \rfloor = s$. We construct a context set $\mathcal{C}$ by picking a set of c' distinct Pauli observables (which is possible since the maximum number of independent Pauli observables is $d^2 - 1 \geq c'$), so $\mathcal{C} = \{\sigma_i\}_{i=1}^{c'}$. Recall that a Pauli observable is a n-fold tensor product of the 2×2 Pauli matrices; thus, the reward will be a binary outcome $r_t \in \{-1, 1\}$. Then, the context distribution works as follows: at each group s of rounds, the learner will receive a different context $\sigma_s \in \mathcal{C}$, so at group s, the learner only receives σ_s.

We want to build an environment such that for each group of rounds $s \in [c']$, all probability distributions are uniform except one that is slightly perturbed. We associate each Pauli observable σ_i to one unique action $a \in [k]$, and we do this association uniformly at random (each action can be associated with more than 1 Pauli observable). Then, each action $a \in [k]$ will have $\{\sigma_{a,1}, ..., \sigma_{a,n_a}\}$ associated Paulis observables, and we can construct the following environment $\gamma = \{\rho_a\}_{a=1}^k$ where

$$\rho_a = \frac{I}{d} + \sum_{j=1}^{n_a} \frac{\Delta}{d} \sigma_{a,j}, \tag{11}$$

$n_a \in \{0, 1, ..., d^2 - 1\}$, $\sum_{a=1}^k n_a = c'$ and Δ is some positive constant. For every group $s \in [c']$, the learner will receive a fixed context $\sigma_s \in \mathcal{C}$, and there will be a unique action a' with $P_{\rho_{a'}}(1|A_t = a', s) = \frac{1}{2} + \frac{\Delta}{2}$ (probability of obtaining $+1$), and the rest $a \neq a'$ will have $P_{\rho_a}(1|a, s) = \frac{1}{2}$ (uniform distributions). Thus, using that the contexts are independent ($\text{Tr}(\sigma_i \sigma_j)$ for $i \neq j$), we can apply (9) independently to every group s and we obtain a regret lower bound $\Omega(\sqrt{T'k}) = \Omega(\sqrt{\frac{Tk}{c'}})$. Note that in order to apply (9), we need $T' \geq k$ or equivalently $T \geq c'k$. Thus, summing all the c' groups, we obtain the total regret scales as

$$\text{Regret}_T^{\gamma,\mathcal{C},\pi} = \Omega\left(c'\sqrt{\frac{Tk}{c'}}\right) = \Omega\left(\sqrt{kT} \cdot \min\left\{d, \sqrt{c}\right\}\right). \tag{12}$$

□

4 Algorithm

In this section, we review the linear model of multi-armed stochastic bandits and one of the main classical strategies that can be used to minimize regret in this model and also in the QCB model.

4.1 Linear disjoint single context bandits and QCB

The classical setting that matches our problem is commonly referred to as linear contextual bandits (Chu et al. 2011), although it has received other names depending on the specific setting such as linear disjoint model (Li et al. 2010) or associative bandits (Auer 2003). The setting that we are interested in uses discrete action sets, and optimal algorithms are based on upper confidence bounds (UCB). While these algorithms use the "principle of optimism in the face of uncertainty," there are other approaches like a Thompson sampling (Agrawal and Goyal 2013) algorithm, but they are not optimal for discrete action sets. We use the contextual linear disjoint bandit model from Li et al. (2010) where each action $a \in [k]$ has an associated unknown parameter $\theta_a \in \mathbb{R}^d$ and at each round t the learner receives a context vector $\mathbf{c}_{t,a} \in \mathbb{R}^d$ for each action. Then, after selecting an action $a \in [k]$, the sampled reward is

$$R_t = \boldsymbol{\theta}_a^\top \mathbf{c}_{t,a} + \eta_t, \tag{13}$$

where η_t is some bounded subgaussian noise such that $\mathbb{E}[R_t|A_t = a] = \boldsymbol{\theta}_a \cdot \mathbf{c}_{t,a}$. Recall that a random variable X is called σ-subgaussian if for al $\mu \in \mathbb{R}$ we have $\mathbb{E}[\mu X] \leq \exp\left(\sigma^2 \mu^2/2\right)$.

In order to map the above setting to the d-dimensional QCB model $(\gamma, \mathcal{C})$, it suffices to consider a vector parametrization (similarly done for the MAQB (Lumbreras et al. 2022)). We choose a set $\{\sigma_i\}_{i=1}^{d^2}$ of independent Hermitian matrices and parametrize any $\rho_a \in \gamma$ and $O_l \in \mathcal{C}$ as

$$\rho_a = \sum_{i=1}^{d^2} \theta_{a,i} \sigma_i, \quad O_l = \sum_{i=1}^{d^2} c_{l,i} \sigma_i, \tag{14}$$

where $\theta_{a,i} = \text{Tr}(\rho_a \sigma_i)$ and $c_{l,i} = \text{Tr}(O_c \sigma_i)$ and we define the vectors $\boldsymbol{\theta}_a = (\theta_{a,i})_{i=1}^{d^2} \in \mathbb{R}^{d^2}$ and $\mathbf{c}_l = (c_{a,i})_{i=1}^{d^2} \in \mathbb{R}^{d^2}$. Then, we note that for the QCB model, the rewards will be given by Eq. 13 with the restriction that since we only receive one observable at each round, then the context vector is constant among all actions. Thus, in our model, the rewards have the following expression:

$$R_t = \boldsymbol{\theta}_a^\top \mathbf{c}_t + \eta_t. \tag{15}$$

We denote this classical model as *linear disjoint single context bandits*. In order to make clear when the classical real vectors parametrize an action $\rho_a \in \gamma$ or context $O_l \in \mathcal{C}$, in Eq. 14, we will use the notation $\boldsymbol{\theta}_{\rho_a}$ and $\mathbf{c}_{O_l}$ respect to the standard Pauli basis.

4.2 Linear upper confidence bound algorithm

Now, we discuss the main strategy for the linear disjoint single context model (15) that is the LinUCB (linear upper confidence bound) algorithm (Auer 2003; Chu et al. 2011; Varsha et al. 2008; Rusmevichientong and Tsitsiklis 2010; Abbasi-Yadkori et al. 2011). We describe the procedure of LinUCB for selecting an action, and we leave for the next section a complete description of the algorithm for the QCB setting.

At each time step t, given the previous rewards $R_1, ..., R_{t-1} \in \mathbb{R}$, selected actions $a_1, ..., a_{t-1} \in [k]$, and observed contexts $\mathbf{c}_1, ..., \mathbf{c}_t \in \mathbb{R}^d$, the LinUCB algorithm builds the *regularized least squares estimator* for each unknown parameter $\boldsymbol{\theta}_a$ that have the following expression:

$$\tilde{\boldsymbol{\theta}}_{t,a} = V_{t,a}^{-1} \sum_{s=1}^{t-1} R_s \mathbf{c}_s \mathbb{I}\{a_t = a\}, \tag{16}$$

where $V_{t,a} = I + \sum_{s=1}^{t-1} \mathbf{c}_s \mathbf{c}_s^\top \mathbb{I}\{a_t = a\}$. Then, LinUCB selects the following action according to

$$a_{t+1} = \underset{a \in [k]}{\operatorname{argmax}} \tilde{\boldsymbol{\theta}}_{t,a}^\top \mathbf{c}_t + \alpha \sqrt{\mathbf{c}_t^\top V_t^{-1} \mathbf{c}_t}, \tag{17}$$

where $\alpha > 0$ is a constant that controls the width of the confidence region on the direction of $\mathbf{c}_{t,a}$. The idea behind this selection is to use an overestimate of the unknown expected value using an upper confidence bound. This is the principle behind UCB (Lai 1987) which is the main algorithm that gives rise to this class of optimistic strategies. The value of the constant α is chosen depending on the structure of the action set. In the next section, we will discuss the appropriate choice of α for our setting. We note that this strategy can also be applied in an adversarial approach where the context is chosen by an adversary instead of sampled from some probability distribution.

The above procedure is shown to be sufficient for practical applications (Li et al. 2010), but the algorithms achieving the optimal regret bound are SupLinRel (Auer 2003) and BaseLinUCB (Chu et al. 2011). They use a phase elimination technique that consists of each round playing only with actions that are highly rewarding, but still, the main subroutine for selecting the actions is LinUCB. This technique is not the most practical for applications, but it was introduced in order to derive rigorous regret upper bounds. For these strategies, if we apply it to a d-dimensional QCB bandit $(\gamma, \mathcal{C})$, they achieve the almost optimal regret bound of

$$\text{Regret}_T^{\gamma,\mathcal{C},\pi} = O\left(d\sqrt{kT \ln^3(T^2 \log(T))}\right). \tag{18}$$

The above bound comes from Chu et al. (2011), and it is adapted to our setting using the vector parametrization (14). Our model uses a different unknown parameter $\boldsymbol{\theta}_a \in \mathbb{R}^d$ for each action $a \in [k]$. This model can be easily adapted to settings where they assume only one unknown parameter shared by all actions if we enlarge the vector space and define $\boldsymbol{\theta} = (\theta_1, ..., \theta_k) \in \mathbb{R}^{dk}$ as the unknown parameter. Their regret analysis works under the normalization assumptions $\|\boldsymbol{\theta}\|_2 \leq 1$, $\|\mathbf{c}_t\|_2 \leq 1$ and the choice of $\alpha = \sqrt{\frac{1}{2}\ln(2T^2k)}$. We note that it matches our lower bound (10) except for the logarithmic terms. While we deal with arbitrarily large qubit systems, because of the choice of the families of Hamiltonians, the dimension the algorithm works in (which we define as d_{eff}) is much smaller than the dimension of the entire Hilbert of space of these multi-qubit systems. This is discussed in detail in the next section. This dimension is sufficiently less, and the previous argument regarding the upper and lower bounds holds. It is important to note that this effective dimension is less than the entire Hilbert space in these chosen families, but not in general sets of Hamiltonians.

5 Low energy quantum state recommender system

In this section, we describe how the QCB framework can be adapted for a recommender system for low-energy quantum states. We consider a setting where the learner is given optimization problems in an online fashion and is able to encode these problems into Hamiltonians and also has access to a set of unknown preparations of (mixed) quantum states that they want to use in order to solve these optimization problems. The task is broken into several rounds; at every round, they receive an optimization problem and are required to choose the state that they will use for that problem. As a recommendation rule, we use the state with the lowest energy with respect to the Hamiltonian where the optimization problem is encoded. We denote this problem as the *low energy quantum state recommendation problem*. We note that our model focuses on the recommendation following the mentioned rule. After selecting the state, the learner will use it for the optimization problem (for example, the initial ansatz state of a variational quantum eigensolver), but that is a separate task. When a learner chooses an action, they must perform an energy measurement using the given Hamiltonian on the state

corresponding to the chosen action. Then, the measurement outcomes are used to model rewards, and their objective is to maximize the expected cumulative reward, i.e., the expectation on the sum of the measurement outcomes over all the rounds played. Any Hamiltonian can be written as a linear combination of Pauli observables, and we use this property to perform measurements. Now, by measuring each of these Pauli observables (since these measurements are conceivable, Peruzzo et al. (2014)), and taking the appropriately weighted sum of the measurement outcomes, we can simulate such a measurement. The QCB framework naturally lends itself to this model, where the Hamiltonians are the contexts, and the set of states that can be prepared reliably serve as the actions.

In this paper, we study some important families of Hamiltonians—specifically, the Ising and a generalized cluster model from Verresen et al. (2017), which are linear combinations of Pauli observables with nearest-neighbor interactions, and for n qubits can be written as

$$H_{\text{ising}}(h) = \sum_{i=1}^{n} (Z_i Z_{i+1} + h X_i), \tag{19}$$

$$H_{\text{cluster}}(j_1, j_2) = \sum_{i=1}^{n} (Z_i - j_i X_i X_{i+1} - j_2 X_{i-1} Z_i X_{i+1}), \tag{20}$$

where $h, j_1, j_2 \in \mathbb{R}$. In the Ising model, h corresponds to the external magnetic field. Specifically, we consider QCB with the following context sets

$$\begin{aligned} \mathcal{C}_{\text{Ising}} &= \left\{ H_{\text{ising}}(h) : h \in \mathbb{R} \right\}, \\ \mathcal{C}_{\text{cluster}} &= \left\{ H_{\text{cluster}}(j_1, j_2) : j_1, j_2 \in \mathbb{R} \right\}. \end{aligned} \tag{21}$$

Important families of Hamiltonians like the models discussed above show translation-invariance and are spanned by Pauli observables showing nearest-neighbor interactions, and as a result, span a low-dimensional subspace. We illustrate the scheme described above through the example of the Ising Model contexts. The Pauli observables that need to be measured are $\{X_i\}_{i\in[n]}$ and $\{Z_i Z_{i+1}\}_{i\in[n]}$. These observables have 2 possible measurement outcomes, -1 and 1, and by the reward distribution of a Pauli observable M given by Born's rule (2) on a quantum state ρ, the reward can be modeled as

$$R_{M,\rho} = 2\text{Bern}\left(\frac{\text{Tr}(M\rho) + 1}{2}\right) - 1, \tag{22}$$

where $\text{Bern}(x) \in \{0, 1\}$ is a random variable with Bernoulli distribution with mean $x \in [0, 1]$. By performing such a measurement for all the Pauli observables and adding the rewards, the reward for $\mathcal{C}_{\text{Ising}}$ is

$$R_{\text{Ising}} = -h \sum_{M \in X_i, i\in[n]} R_{M,\rho} - \sum_{M' \in Z_i Z_{i+1}, i\in[n]} R_{M',\rho}, \tag{23}$$

where we took the negative of the sum of the measurements because we are interested in a recommender system for the lowest energy state. A similar formulation applies to the QCB with generalized cluster Hamiltonian contexts.

In the rest of this section, we illustrate a modified LinUCB algorithm for the QCB setting. Then, we implement this recommender system, where the contexts are Hamiltonians belonging to the Ising and generalized cluster models (21), and demonstrate our numerical analysis of the performance of the algorithm by studying the expected regret. We also demonstrate that depending on the action set, the algorithm is able to approximately identify the phases of the context Hamiltonians.

5.1 Gram-Schmidt method

Similarly to the task of shadow tomography (Aaronson 2018) and classical shadows (Huang et al. 2020), we do not need to reconstruct the full quantum states since the algorithm has only to predict the trace between the contexts and the unknown quantum states. Thus, the LinUCB algorithm has only to store the relevant part of the estimators for this computation. As the measurement statistics depend only on the coefficient corresponding to the Pauli observables spanning the observables in the context set, only those Pauli observables in the expansion of the estimators are relevant. This means that our algorithm can operate in a space with a smaller dimension than the entire spaces spanned by n qubits, which has a dimension that is exponential in the number of qubits. As a side note, we would like to remark that one could hope to apply classical shadows to our problem since both tasks seem very similar. However, this is not the case since we need an online method that performs the best recommendation at each time step t.

In order to exploit this property to improve the space complexity of the LinUCB algorithm, we use the Gram-Schmidt procedure in the following way. At any round, a basis for the vector parameterizations (as shown in Eq. 14) of all the previously received contexts is stored. If the incoming vector parameterization of the context is not spanned by this basis, the component of the vector orthogonal to the space spanned by this set is found by a Gram-Schmidt orthonormalization-like process, and this component is added to the set after normalization. Therefore, at any round, there will be a list of orthonormal vectors that span the subspace of all the vector parameterizations of the contexts received so far, and the size of the list will be equal to the dimension of the subspace, which we call *effective dimension*, i.e.,

$$d_{\text{eff},t} = \dim(\{O_{c_t} \in \mathcal{C}\} : t \in [T]). \tag{24}$$

From now on, we will omit the subscript for the time step t and simply denote the effective dimension as d_{eff}. Instead of

feeding the context vectors directly, for any incoming context vector, we construct d_{eff}-dimensional vectors, whose i^{th} term is the inner product of the context vector and the i^{th} basis vector. In case the incoming vector is not spanned by the basis, we first update the list by a Gram-Schmidt procedure (which will result in an addition of another orthonormal vector to the list and an increase in d_{eff} by 1) and then construct a d_{eff}-dimensional vector as described before. This vector is fed to the LinUCB algorithm. The Gram-Schmidt procedure is stated in Algorithm 1, and the modified LinUCB algorithm is stated explicitly in Algorithm 2. The efficiency of this method is well illustrated in the case where all the contexts are local Hamiltonians. As an example, we discuss the case of generalized cluster Hamiltonians. Note that the space complexity of the standard QCB framework is $O(kd^2)$, where k is the number of actions, and d is the dimension of the vector parameterizations of the contexts. In the standard LinUCB technique, the context vectors $\mathbf{c_t}, t \in [T]$ would be 4^n-dimensional, where n is the number of qubits the Hamiltonian acts on, in which case the space complexity of the algorithm is $O(k4^{2n})$. In our studies, the contexts are Ising Hamiltonians and a generalized cluster Hamiltonian (21) with $d_{\text{eff}} \leq 2$ and $d_{\text{eff}} \leq 3$, respectively. Since the vectors fed into the modified LinUCB are d_{eff}-dimensional, the space complexity is $O(kd_{\text{eff}}^2)$, i.e., $O(4k)$ and $O(9k)$, respectively.

5.2 Phase classifier

In order to implement the numerical simulations, we need to choose the environments for the QCB with context sets $\mathcal{C}_{\text{ising}}$ and $\mathcal{C}_{\text{cluster}}$. Elements of both context sets are parameterized by tunable parameters. We study the performance of the recommender system by choosing a context probability distribution that is uniform (over a chosen finite interval on the real line) on these parameters. Then, we chose the actions as ground states of Hamiltonians that corresponded to the limiting cases (in terms of the parameters) of these models. In order to study the performance of our strategy apart from the expected regret (6), we want to observe how the actions are chosen. For every action, we maintained a set, which contained all the Hamiltonians for which that action was chosen. We observed that almost all the elements in each of these sets belonged to the same phase of the Hamiltonian models.

In order to study the performance of the algorithm in this respect, we define the *classifier regret* as

$$\text{ClassifierRegret}_T^{\gamma,\mathcal{C},\pi} = \sum_{t=0}^{T-1} \mathbb{I}\left[a_t \neq a_{\text{optimal,t}}\right], \tag{25}$$

where $a_{\text{optimal,t}} = \text{argmax}_{a\in[k]} \text{Tr}\,(O_t \rho_a)$, and $O_t \in \mathcal{C}$ is the context observable received in t^{th} round. Note that the above classifier regret is not guaranteed to be sublinear like expected regret is Eq. 18 for the LinUCB strategy. This can be understood intuitively: consider a scenario where the bandit picks an action with a small sub-optimality gap (7); then, the linear regret will increase by a very small amount, the classifier regret will increase by one unit, as all misclassifications have equal contribution to regret. These, however, are theoretic worst-case scenarios, and this classifier regret is useful to study the performance of the algorithm in practice in our settings.

5.3 Numerical simulations

Before we move into the specific cases, we note the importance of the choice of α in Algorithm 2. While the theoretical analysis of the LinUCB algorithm depends on the choice of α, in practice, one can tune this value to observe a better performance. We primarily use the α described in Lattimore and Szepesvári (2020) (Chapter 19) given by

$$\alpha_t = m + \sqrt{2\log\left(\frac{1}{\delta}\right) + d\log\left(1 + \frac{tL^2}{d}\right)}. \tag{26}$$

Algorithm 1 Gram-Schmidt Algorithm (Gram($\mathbf{c}$, V_a, $\mathbf{b}_a$, CBasis)).

Input $[\mathbf{c}, \{V_a\}_{a\in\mathcal{A}}, \{\mathbf{b}_a\}_{a\in\mathcal{A}}, \text{CBasis}]$
for $\mathbf{v}$ in CBasis **do**
  $\mathbf{c} \leftarrow \mathbf{c} - (\mathbf{v}^\top \mathbf{c})\mathbf{v}$
  $\mathbf{v}_{\text{ct}} = \mathbf{v}_{\text{ct}} \oplus (\mathbf{v}^\top \mathbf{c})$
end for
if $\mathbf{c}! = \mathbf{0}$ **then**
  $\mathbf{v}_{\text{ct}} = \mathbf{v}_{\text{ct}} \oplus \|\mathbf{c}\|_2$
  Add $\mathbf{c}/\|\mathbf{c}\|_2$ to CBasis
  for $a = 1, 2, \ldots, K$ **do**
    Set $V_a = V_a \oplus I_1$, $\mathbf{b}_a = \mathbf{b}_a \oplus \mathbf{0}_1$
  end for
end if
Return $\left[\mathbf{c}', \{V_a\}_{a\in\mathcal{A}}, \{\mathbf{b}_a\}_{a\in\mathcal{A}}, \text{CBasis}\right]$

Algorithm 2 LinUCB with Gram-Schmidt.

Input $\alpha \in \mathbb{R}$
Set CBasis = [ ]
Set $V_a = \mathbf{1}$, $\mathbf{b}_a = \mathbf{0}$, $\forall a \in \mathcal{A}$
for $t = 1, 2, \ldots$ **do**
  $\left[\mathbf{c'}_{O_t}, \{V_a\}_{a\in\mathcal{A}}, \{\mathbf{b}_a\}_{a\in\mathcal{A}}, , \text{CBasis}\right]$
  $\leftarrow$ Gram($\mathbf{c}_{O_t}$, $\{V_a\}_{a\in\mathcal{A}}$, $\{\mathbf{b}_a\}_{a\in\mathcal{A}}$, , CBasis)
  for $a \in \mathcal{A}$ **do**
    $\tilde{\boldsymbol{\theta}}_{\rho_a} \leftarrow V_a^{-1}\mathbf{b}_a$
    $p_{t,a} \leftarrow \tilde{\boldsymbol{\theta}}_{\rho_a}\mathbf{c'}_{O_t} + \alpha\sqrt{\mathbf{c}'^{\top}_{O_t} V_a^{-1} \mathbf{c'}_{O_t}}$
  end for
  Choose action $\mathbf{a_t} = \text{argmax}_{a\in\mathcal{A}}\, p_{t,a}$;
  Measure state ρ_{a_t} with O_{c_t} and observe reward R_{O_t}
  Set $V_{a_t} \leftarrow V_{a_t} + \mathbf{c'}_{O_t}\mathbf{c'}^{\top}_{O_t}$
  Set $\mathbf{b}_{a_t} \leftarrow \mathbf{b}_{a_t} + R_{O_t}\mathbf{c'}_{O_t}$
end for

Here, L and m are upper bounds on the 2-norm of the action vectors and unknown parameter, respectively, d is the dimension, and δ is once more a probability of failure.

Finally, while we study the performance of our algorithm in our simulations with estimates of expected regret and expected classifier regret, it is important to note that in an experimental setup, the learner will only be able to measure the cumulative reward at every round. However, since these are simulations, we are able to study the regret as well, as they are standard metrics to gauge the performance of the algorithms. In the next subsection, we discuss our simulations of the QCB bandit $(\gamma, \mathcal{C}_{\text{cluster}})$ model, and later, the QCB bandit $(\gamma, \mathcal{C}_{\text{Ising}})$ is discussed in the Appendix 7.

5.3.1 Generalized cluster model

We study the performance of the recommender system for the QCB bandit $(\gamma, \mathcal{C}_{\text{cluster}})$, where the generalized cluster Hamiltonians (Verresen et al. 2017) act on 10 qubits and 100 qubits, respectively. We observe that the performance of the algorithm is not affected by the number of qubits, as the effective dimension of the context set remains unchanged, i.e., $d_{\text{eff}} = 3$. We study the expected regret and classifier regret for these two cases and illustrate the system's performance in finding the phases of the generalized cluster Hamiltonians. This model was also studied in Caro et al. (2022), where they designed a quantum convolutional neural network to classify quantum states across phase transitions. We chose 5 actions corresponding to approximate ground states of Hamiltonians that are the limiting cases of the generalized cluster model, i.e., generalized cluster Hamiltonians with parameters j_1, j_2 in Eq. 20, $j_1, j_2 \rightarrow \{0, 0\}, \{0, \infty\}, \{\infty, 0\}, \{0, -\infty\}$and $\{0, -\infty\}$. Note that these methods of approximating ground states are only for simulation purposes.

Initially, a steep growth in regret is observed, followed by a sudden slower pace. On looking closely, in the plot Fig. 3, we find that the regret indeed continues to grow, albeit at a slower pace. This was explained by observing that the sub-optimality gap of the second-best action is quite small in comparison to the sub-optimality gaps of the rest of the actions. Initially, the

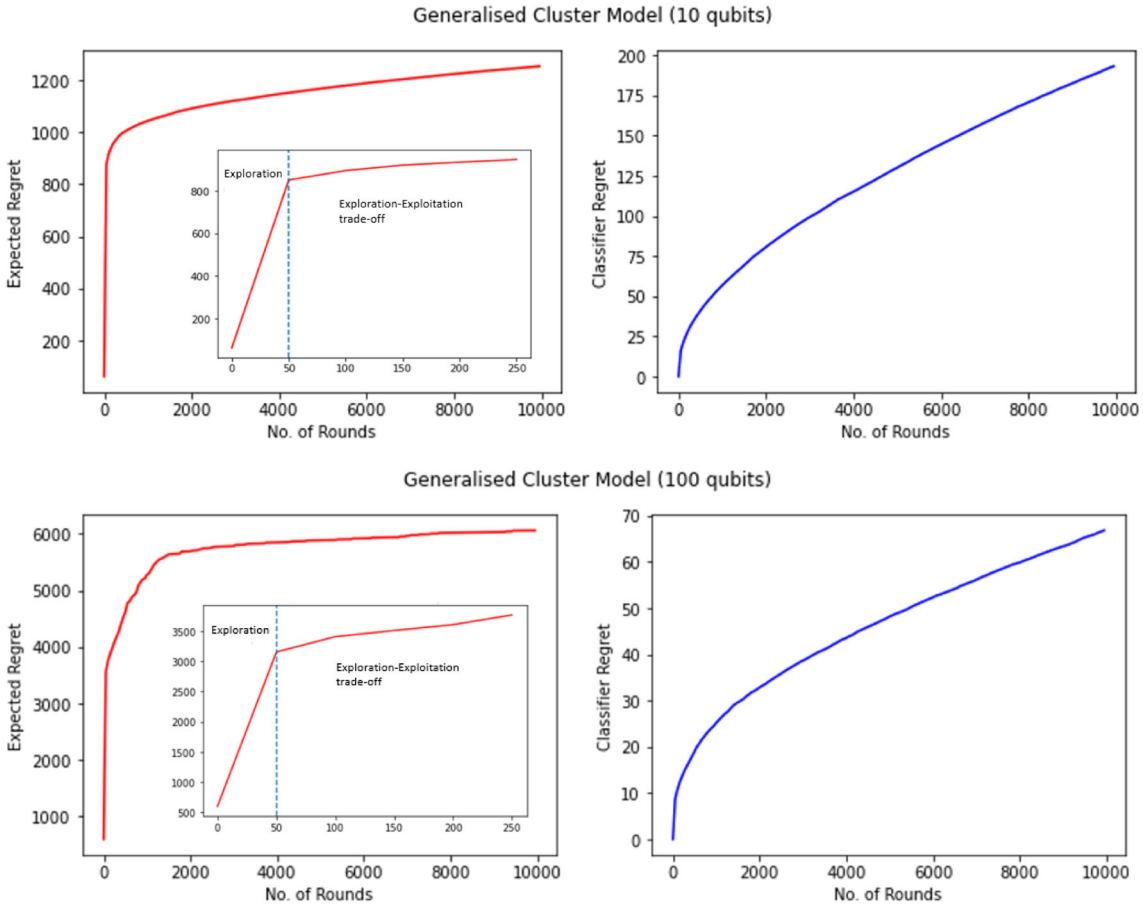

Fig. 2 Plots for regret and classifier regret for QCB bandit $(\gamma, \mathcal{C})$, where the Hamiltonians in $\mathcal{C}$ are a specific form of generalized cluster models acting on 10 and 100 qubits, respectively. The performance is not very different since $d_{\text{eff}} = 3$ (24) for both cases. The action set is chosen to be approx. ground states of some generalized cluster Hamiltonians

LinUCB algorithm does not have enough information about the unknown parameters and has to play all actions resulting in an exploration phase. However, at some point, the bandit recognizes the "bad" actions and plays either the best action or the action with a small sub-optimality gap most of the time—this is when the bandit has begun to balance exploration and exploitation. This is illustrated by observing the growth of the regret before and after the first 50 rounds in the insets of Fig. 2. At the beginning of this subsection, we mentioned that the recommendation system picks the same action for context Hamiltonians belonging to the same phase. We illustrate this in Fig. 3. In the scatter plot, when a context generalized cluster Hamiltonian is received, a dot is plotted with the x-axis and y-axis coordinates corresponding to its parameters j_1, j_2, respectively. Depending on the action picked by the algorithm, we associate a color with the dot. The resultant plot is similar (but not exact) to the phase diagram of the generalized cluster Hamiltonian. We stress that the algorithm that we are using for the below plots is the same as the one for Fig. 2, and we are just plotting a visual representation in terms of the phases. We do not expect that this method can compete with other phase classifiers if the algorithms were designed for such tasks as the one in Huang et al. (2021).

6 Outlook

This work describes the first steps for recommending quantum data by implementing the bandit framework in a rigorous fashion for practical scenarios. We provide a recommender system based on the theory of linear contextual bandits and show that the upper and lower bounds on the performance meet asymptotically. We also demonstrate its efficiency in practice through simulations. Later, we show how such a system could also be used to recognize phases of Hamiltonians.

We restricted our attention to a model where the expected rewards follow a linear function in terms of the context and the unknown states. While the low energy quantum state recommendation problem uses the outcome of the measurement as a reward, one could think of other recommendation tasks with more complicated reward functions. Non-linear rewards have been studied in the bandit literature and receive the name of structured bandits (Lattimore and Munos 2014; Combes et al. 2017; Russo and Roy 2013). This model could be a natural extension of the QCB for other recommendation tasks where the rewards are not in one-to-one correspondence with measurement outcomes. Going back to the general model, the environment is modeled by a set of unknown quantum processes, which in the QCB model, we assumed to be a set of stationary unknown quantum states. In a more general scenario, we can consider environments that change with time due to some Hamiltonian evolution or noise interaction with an external environment. In the bandit literature, non-stationary environments were first considered in Gittins (1979); Gittins et al. (2011), where each action was associated with a Markov chain or the restless bandit model (Whittle 1988) where the Markov chain associated to each action evolves with time. More recently, in Luo et al. (2018), they studied a contextual bandit model with non-stationary environments. We expect that recommender systems for quantum data can also be extended to similar settings.

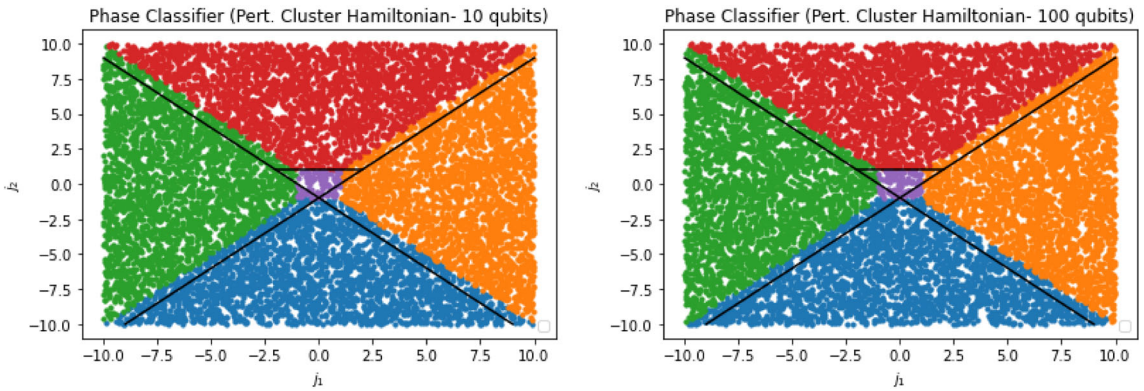

Fig. 3 These plots illustrate how the recommender system identifies the phases of the generalized cluster Hamiltonian. The x and y-axis represent the coupling coefficients of the generalized cluster Hamiltonian received as context. Like the Ising model simulations, we associate a color to each action. For any context, $H_{\text{cluster}}(j_1, j_2)$ corresponding to any of the T rounds, one of these actions is picked by the algorithm. We plot the corresponding colored dot (blue for the ground state of $H_{\text{cluster}}(-\infty, 0)$, orange for $H_{\text{cluster}}(0, \infty)$, red for $H_{\text{cluster}}(\infty, 0)$, green for $H_{\text{cluster}}(0, -\infty)$, and purple for $H_{\text{cluster}}(0, 0)$) at the appropriate coordinates for rounds that follow after the bandit has "learned" the actions, i.e., the growth in regret has slowed down. The lines in black represent the actual phase diagram

Appendix

In this appendix, we discuss the simulations for the QCB bandit $(\gamma, \mathcal{C}_{\text{ising}})$ setting, in a similar fashion as described in Section 5.3. We study the performance of the recommender system for the QCB bandit $(\gamma, \mathcal{C}_{\text{ising}})$, where the Ising Hamiltonians act on 10 qubits and 100 qubits, respectively. We observe that the performance of the algorithm is not affected by the number of qubits, as the effective dimension of the context set remains unchanged. We study the expected regret and classifier regret for these two cases and illustrate the system's performance in finding the phases of the Ising model. The action set corresponds to the ground state of the 3 limiting cases of the Ising model, i.e., Ising Hamiltonians for parameter h in Eq. 19, $h = 0$, $h \to -\infty$ and $h \to \infty$.

Once more, like the generalized cluster model simulations, we observe a distinct "exploration" stage, followed by an "exploration-exploitation trade-off" stage which we again plot separately. This is illustrated by observing the growth of the regret before and after the first 50 rounds shown in the insets of Fig. 4.

In the beginning of this subsection, we mentioned that the recommendation system picks the same action for context Hamiltonians belonging to the same phase. We illustrate this in Fig. 5. In the scatter plot, when a context Ising Hamiltonian is received, a dot is plotted with the x-axis coordinate corresponding to its parameter. Depending on the action picked by the algorithm, we associate a color to the dot. The resultant plot is very similar to that of the phase diagram of an Ising model. The Ising model is known to have phase transitions at $h = -1, 1$, resulting in 3 phases, $(-\infty, -1]$, $[-1, 1]$ and $[1, \infty)$, and we observe that different actions were played in each of these ranges.

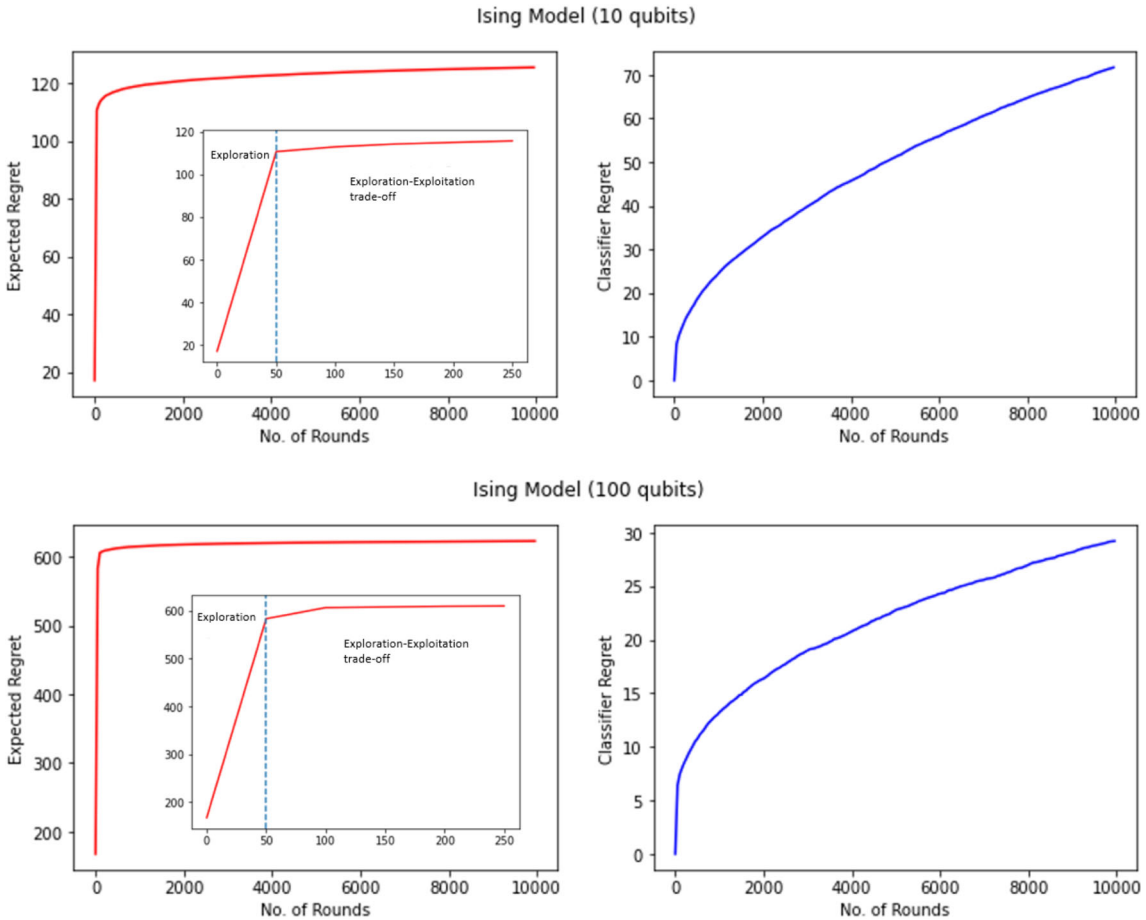

Fig. 4 Plots for regret and classifier regret for QCB bandit $(\gamma, \mathcal{C})$, where the Hamiltonians in $\mathcal{C}$ are Ising Hamiltonians acting on 10 and 100 qubits, respectively. The performance is not very different since $d_{\text{eff}} = 2$ (24) for both cases. The action set is chosen to be approx. ground states of some Ising Hamiltonians

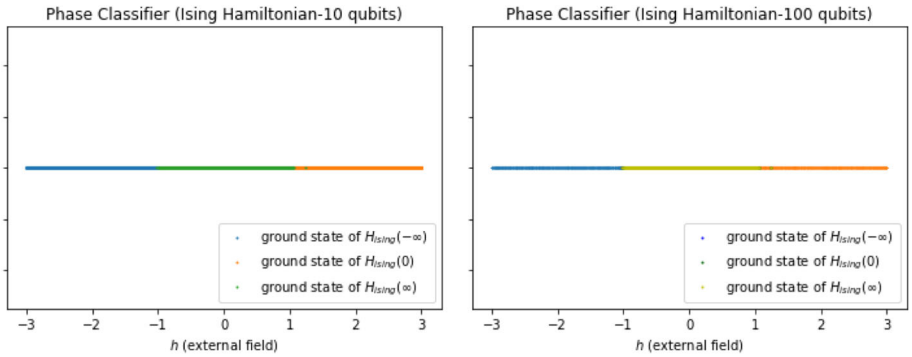

Fig. 5 These plots illustrate how the recommender system identifies the phases of the Ising Hamiltonian. The x-axis represents the external field coefficient of the Ising Hamiltonian received as context. The blue, green, or yellow mark indicates that the algorithm plays the 1st,2nd or 3rd action. We plot the corresponding colored dot at the appropriate coordinates, for rounds that follow after the bandit has "learned" the actions, i.e., the growth in regret has slowed down

Acknowledgements This research is supported by the National Research Foundation, Singapore and A*STAR under its CQT Bridging Grant and the Quantum Engineering Programme grant NRF2021-QEP2-02-P05.

Code Availability The code used for the simulations in this manuscript has been posted in this Github repository for public use. The authors may be contacted for any clarifications.

References

Aaronson S (2018) Shadow tomography of quantum states. In: Proceedings of the 50th annual ACM SIGACT symposium on theory of computing, pp 325–338

Abbasi-Yadkori , Pál D, Szepesvári Cs (2011) Improved algorithms for linear stochastic bandits. In: Advances in neural information processing systems, Curran Associates, Inc., 24

Abe N, Biermann AW, Long PM (2003) Reinforcement learning with immediate rewards, linear hypotheses. Algorithmica 37(4):263–293

Agrawal S, Goyal N (2013) Thompson sampling for contextual bandits with linear payoffs. In: International conference on machine learning, PMLR, pp 127–135

Auer P (2003) Using confidence bounds for exploitation-exploration trade-offs. Journal of Machine Learning Research, 3:397–422. ISSN 1532-4435

Auer P, Cesa-Bianchi N, Freund Y, Schapire RE (2003) The nonstochastic multiarmed bandit problem. SIAM J. Comput, 32 (1): 48–77. ISSN 0097-5397 https://doi.org/10.1137/S0097539701398375

Barry J, Barry DT, Aaronson S (2014) Quantum partially observable Markov decision processes. Phys Rev A 90:032311. https://doi.org/10.1103/PhysRevA.90.032311

Bharti K, Cervera-Lierta A, Kyaw TH, Haug T, Alperin-Lea S, Anand A, Degroote M, Heimonen H, Kottmann JS, Menke T, Mok W, Sim S, Kwek L, Aspuru-Guzik A (2022) Noisy intermediate-scale quantum algorithms. Rev Mod Phys 94:015004. https://doi.org/10.1103/RevModPhys.94.015004

Bouneffouf D, Rish I, Aggarwal C, (2020) Survey on applications of multi-armed and contextual bandits. In: 2020 IEEE Congress on Evolutionary Computation (CEC), IEEE pp 1–8

Caro MC, Huang H, Cerezo M, Sharma K, Sornborger A, Cincio L, Coles PJ (2022) Generalization in quantum machine learning from few training data. Nat Commun 13(1):4919

Casalé B, Di Molfetta G, Kadri H, Ralaivola L (2020) Quantum bandits. Quantum Mach Intell 2. https://doi.org/10.1007/s42484-020-00024-8

Cho B, Xiao Y, Hui P, Dong D (2022) Quantum bandit with amplitude amplification exploration in an adversarial environment. arXiv:2208.07144

Chu W, Li L, Reyzin L, Schapire RE (2011) Contextual bandits with linear payoff functions. In: AISTATS, pp 208–214

Cohen M, Lobel I, Paes Leme R (2020) Feature-based dynamic pricing. Manage Sci 66(11):4921–4943. https://doi.org/10.1287/mnsc.2019.3485

Combes R, Magureanu S, Proutiere A (2017) Minimal exploration in structured stochastic bandits. In: Proceedings of the 31st International conference on neural information processing systems, pp 1761–1769

Dragone P, Mehrotra R, Lalmas M (2019) Deriving user-and content-specific rewards for contextual bandits. In: The world wide web conference pp 2680–2686

Durand A, Achilleos, C Iacovides D, Strati K, Mitsis GD, Pineau J (2018) Contextual bandits for adapting treatment in a mouse model of de novo carcinogenesis. In: Machine learning for healthcare conference, PMLR pp 67–82

Gittins JC (1979) Bandit processes and dynamic allocation indices. J Roy Stat Soc: Ser B (Methodol) 41(2):148–164

Gittins J, Glazebrook K, Weber R (2011) Multi-armed bandit allocation indices. John Wiley & Sons

Gomez-Uribe CA, Hunt N (2016) The netflix recommender system: algorithms, business value, and innovation. ACM Trans Manage Inf Syst 6(4) ISSN 2158-656X. https://doi.org/10.1145/2843948

Huang HY, Kueng R, Torlai G, Albert VV, Preskill J (2021) Provably efficient machine learning for quantum many-body problems. Science, 377. https://api.semanticscholar.org/CorpusID:262748752

Huang H, Kueng R, Preskill J (2020) Predicting many properties of a quantum system from very few measurements. Nat Phys 16(10):1050–1057

Hu W, Wei, Hu J (2019) Training a quantum neural network to solve the contextual multi-armed bandit problem. Nat Sci 11(01):17

Kerenidis I, Prakash A (2016) Quantum recommendation systems. arXiv:1603.08675

Lai TL (1987) Adaptive treatment allocation and the multi-armed bandit problem. Ann Stat 15(3):1091–1114. https://doi.org/10.1214/aos/1176350495

Lattimore T, Szepesvári C (2020) Bandit algorithms. Cambridge University Press

Lattimore T, Munos R (2014) Bounded regret for finite-armed structured bandits. Advances in Neural Information Processing Systems, 27

Li L, Chu W, Langford J, Schapire RE (2010) A contextual-bandit approach to personalized news article recommendation. In: Proceedings of the 19th international conference on world wide web, ACM, pp 661–670 ISBN 9781605587998. https://doi.org/10.1145/1772690.1772758

Lumbreras J, Haapasalo E, Tomamichel M (2022) Multi-armed quantum bandits: exploration versus exploitation when learning properties of quantum states. Quantum, 6:749. ISSN 2521-327X https://doi.org/10.22331/q-2022-06-29-749

Luo H, Wei C, Chen-Yu, Agarwal A, Langford J (2018) Efficient contextual bandits in non-stationary worlds. In: Conference on learning theory, PMLR, pp 1739–1776

McInerney J, Lacker B, Hansen S, Higley K, Bouchard H, Gruson A, Mehrotra R (2018) Explore, exploit, and explain: personalizing explainable recommendations with bandits. In: Proceedings of the 12th ACM conference on recommender systems pp 31–39

Peruzzo A, McClean J, Jarrod Shadbolt P, Yung M, Zhou X, Love PJ, Aspuru-Guzik A, O'brien JL, (2014) A variational eigenvalue solver on a photonic quantum processor. Nat Commun 5(1):1–7

Rusmevichientong P, Tsitsiklis JN (2010) Linearly parameterized bandits. Math Oper Res 35(2):395–411. https://doi.org/10.1287/moor.1100.0446

Russo D, Van Roy B (2013). Eluder dimension and the sample complexity of optimistic exploration. Advances in Neural Information Processing Systems, 26

Tang E (2019) A quantum-inspired classical algorithm for recommendation systems. In: Proceedings of the 51st annual ACM SIGACT symposium on theory of computing, pp 217–228

Tang L, Rosales R, Singh A, Agarwal D (2013) Automatic ad format selection via contextual bandits. In: Proceedings of the 22nd ACM international conference on information and knowledge management, Association for Computing Machinery, 1587–1594. ISBN 9781450322638 https://doi.org/10.1145/2505515.2514700

Varsha D, Hayes T, Kakade S (2008) Stochastic linear optimization under bandit feedback. In: Proceedings of the 21st conference on learning theory, pp 355–366

Verresen R, Moessner R, Pollmann F (2017) One-dimensional symmetry protected topological phases and their transitions. Phys Rev B 96(16):165124

Wang D, You X, Li T, Childs A (2021) Quantum exploration algorithms for multi-armed bandits. Proc AAAI Conf Artif Intell 35:10102–10110

Wan Z, Zhang Z, Li T, Zhang J, Sun X (2022) Quantum multi-armed bandits and stochastic linear bandits enjoy logarithmic regrets. arXiv:2205.14988

Whittle P (1988) Restless bandits: activity allocation in a changing world. Journal of Applied Probability, 25(A):287–298

Ying M, Feng Y, Ying S (2021) Optimal policies for quantum Markov decision processes. Int J Autom Comput 18(3):410–421. https://doi.org/10.1007/s11633-021-1278-z