Protocols for Verifying Smooth Strategies in Bandits and Games

Miranda Christ
Columbia University
mchrist@cs.columbia.edu

Daniel Reichman
Worcester Polytechnic Institute
daniel.reichman@gmail.com

Jonathan Shafer
MIT
shaferjo@mit.edu

Abstract

We study protocols for verifying approximate optimality of strategies in multi-armed bandits and normal-form games. As the number of actions available to each player is often large, we seek protocols where the number of queries to the utility oracle is *sublinear* in the number of actions. We prove that such verification is possible for sufficiently *smooth* strategies that do not put too much probability mass on any specific action and provide protocols for verifying that a smooth policy for a multi-armed bandit is close to optimal. Our verification protocols require provably fewer arm queries than learning. Furthermore, we show how to use cryptographic tools to reduce the communication cost of our protocols. We complement our protocol by proving a nearly tight lower bound on the query complexity of verification in our settings. As an application, we use our bandit verification protocol to build a protocol for verifying approximate optimality of a strong smooth Nash equilibrium, with sublinear query complexity.

1 Introduction

An overabundance of available actions makes decision-making challenging in numerous settings, from consumer choice to logistical operations and public policy. To gain information about the (often stochastic) reward an action entails, an agent can simply perform that action and experience the outcome. However, in many situations it is infeasible for an agent with limited resources to try even a small fraction of the set of available actions. One method to mitigate such choice overload is to rely on an external source providing information about the utility of actions. For example, one may consult recommendations from other customers when deciding whether to purchase a product, or solicit advice from consulting firms when formulating a business strategy or public policy. This information however, may be unreliable, and it is not clear how to efficiently *verify* its accuracy. Namely, can one outsource information collection to an untrusted external party, and verify correctness by independently assessing the utility of just a few actions?

Verifying utility estimates arises in machine learning (ML) applications as well. For example, an end user may delegate ML training to a better-resourced but untrusted external party [Goldwasser et al., 2021, 2015]. In *reinforcement learning*, a training algorithm might return a set of estimates for the rewards of possible actions. Once again, a challenge for the user is to verify the fidelity of these estimates, which could be inaccurate for numerous reasons. For example, the external party could be malicious, careless and susceptible to errors, or the training algorithm may be faulty. To address this challenge, a line of research develops learning methods that are robust to adversarial data corruption [Charikar et al., 2017, Canetti and Karchmer, 2021]. The specific case of corrupted rewards (utilities) in multi-armed bandits has received attention as well [Jun et al., 2018, Zhang et al., 2022].

Similar verification issues emerge in multiagent settings studied in game theory, where each agent's utility may depend on the other agents' actions. An object of central interest in such settings is the

39th Conference on Neural Information Processing Systems (NeurIPS 2025).

(ε-approximate) *Nash equilibrium* (NE or ε-NE), which is a strategy profile[1] such that no agent can improve their own utility (by more than ε) by deviating unilaterally to another strategy. There is a host of questions related to verification and Nash equilibria; specifically, the current paper is motivated by scenarios such as:

- *An agent might want to verify that, given that the current strategies of the other agents are known, a proposed strategy is (approximately) the best possible.* For instance, suppose Alice is a corporation navigating a complex market, and she receives advice from a consulting firm that recommends a specific strategy $\tilde{\pi}$. Alice might subsequently like to verify that the proposed $\tilde{\pi}$ is indeed an (approximately) optimal strategy given current market conditions (i.e., given what other market participants are doing).
- *An agent might want to verify that, together with the known current strategy profile of the other agents, a proposed strategy forms an (approximate) Nash equilibrium.* For instance, in the previous example, assume the proposed strategy $\tilde{\pi}$ indeed has optimal expected utility for Alice in current market conditions. Nonetheless, adopting strategy $\tilde{\pi}$ might not be a good idea if, in the case where Alice follows strategy $\tilde{\pi}$, some other market participant, Bob, can unilaterally deviate to a strategy that benefits Bob but significantly damages Alice. Therefore, Alice might like to verify that, together with the current known strategy profile of the other market participants, the proposed strategy $\tilde{\pi}$ forms an (approximate) Nash equilibrium. That would be consistent with $\tilde{\pi}$ being a reasonable strategy for Alice to commit to.

We consider two popular models for problems of choosing between different actions: *multi-armed bandits* (MAB) Lattimore and Szepesvári [2020] in the single-agent setting, and *normal-form games* in the multi-agent setting. For both MAB and games, we consider the settings where n, the number of actions available to each player, is a large integer. The large number of actions, along with the prohibitive computational cost of computing approximate NE [Daskalakis et al., 2009, Rubinstein, 2017], motivates the reliance on advice from external parties. This leads to the following questions:

Question 1. *Given an MAB with n arms, can we* verify *whether a list of purported rewards is close to the true rewards and output a near-optimal strategy, using a number of bandit queries that is sublinear in n?*

Question 2. *Given a k-player, n-action normal-form game, can a prover convince a verifier that a strategy profile is an ε-NE, with the verifier making a number of payoff oracle queries that is sublinear in n?*

We address these questions within the framework of *interactive proof systems* [Goldwasser et al., 1989]. In this setting, a computationally bounded verifier interacts with an untrusted prover, and can verify the correctness of statements that are intractable to assess without interaction [Shamir, 1992]. A recent development in the study of interactive proofs is the application of this methodology to machine learning [Goldwasser et al., 2021]. We adopt this development in our setting of verification in MABs. MABs are a fundamental concept in machine learning and are used in numerous theoretical and applied settings. Therefore, we hope that our study could lead to further discoveries regarding the potential uses of interactive proofs in machine learning and operations research. We focus on the *offline* setting of MABs and do not consider regret minimization. Studying interactive proofs for the online setting is an interesting direction for future work.

Let us first give a simple example where the verification of an MAB can be significantly more efficient compared to finding an optimal or near optimal strategy. Consider the promise problem where an MAB is known to have a single arm whose reward a deterministic reward of 1, and all other $n-1$ arms have deterministic reward of 0. It is trivial to *verify* that a given arm communicated by a prover guarantees a reward of 1 using a single pull to that arm. On the other hand, *finding* such an arm can require $\Omega(n)$ pulls to the arms: if the single arm with reward one is chosen uniformly out of n arms, a linear number of queries to the arms of the bandits is needed to find it [Kearns et al., 2002, Mannor and Tsitsiklis, 2004b].

Similarly, for a given parameter γ it can be verified with probability at least $2/3$ that an MAB (with rewards in $[-1,1]$) has an arm with expected reward at least $\gamma-\varepsilon$ for some accuracy parameter ε, using $O(1/\varepsilon^2)$ queries to an arm provided by a prover. As before, finding a strategy with an expected reward of γ requires $\Omega(n)$ queries to the arms of the MAB.

[1] A *strategy* is a distribution over actions. A *strategy profile* is an assignment of a strategy to each player (i.e., agent) in the game.

Observe that in the first example, we have a promise problem in the sense that we know that there exists a single arm with reward 1 and that this is the arm with the highest reward. In the second example, we verify a *lower bound* on the value of an optimal strategy, but we do not verify an upper bound on the best possible value, [2] hence we do not verify that a given strategy is nearly optimal. This is not a coincidence: Even in the interactive proof setting, we show that $\Omega(n/\varepsilon^2)$ queries are necessary. Therefore to achieve verification with a sublinear number of queries to the arms of the bandits, we need to make additional assumptions on either the MAB, or the set of strategies considered. Here we show that verification with a sublinear number of queries to the arms of the bandit is possible for *smooth strategies* discussed in the next section.

In interactive proofs, the number of bits communicated between the prover and the verifier may become a bottleneck. Therefore, it is desirable to have *communication efficient* protocols where the number of bits communicated is sublinear in n as well. We consider communication-efficient protocols in the setting where a prover wishes to convince the verifier of the value of the optimal policy, but does not need to send the optimal policy itself. We show that such communication-efficient protocols can be achieved for sufficiently smooth MABs by using cryptographic primitives such as vector commitments and SNARKs. While our basic protocol consists of just a single message sent from the prover to the verifier, our protocol with a sublinear number of bits of communication uses multiple rounds of interaction.

2 Verifying smooth strategies: model and results

2.1 Smooth strategies: motivation and background

A smooth strategy is a distribution over actions that is not too concentrated on any one action. More precisely, a strategy is σ-smooth if each action is selected with probability at most σ. Clearly $\sigma \in [1/n, 1]$, with smaller values of σ corresponding to smoother (i.e., closer to uniform) distributions. To ensure that our protocols have sublinear query complexity, it is needed that $\sigma n = o(n)$. This is a natural assumption to ensure that the distribution is "well spread" in the sense that the support size of the distribution tends to infinity with n.

As a motivation, in many applications each arm in an MAB represents some resource that a system can utilize (e.g., a server in a data center, a driver in a ride-hailing service). Often, as the system grows, the number of available arms increases, as does the total number of arm-pulls, but the capacity of each individual arm remains bounded (e.g., as a ride-hailing service grows, the total number of rides and drivers increase, but the number of rides each individual driver can handle remains the same). If the total number of arm-pulls is, say, linear in the number of arms, then the MAB policy must be σ-smooth with $\sigma = O(1/n)$.

In an MAB, a strategy is an ε-optimal σ-smooth strategy if it is σ-smooth and there is no σ-smooth strategy with a utility that is greater by more than ε (Definition 5.3). In a normal-form game, a strategy is an ε-approximate strong σ-smooth NE if every player's strategy is σ-smooth, and no player can unilaterally deviate to a σ-smooth strategy that improves their expected payoff by more than ε.

We stress that the optimal *smooth* strategy can have significantly worse expected utility compared to the best *unconstrained* policy. For example, suppose $\sigma = 1/k$ for some integer $k > 1$ and that a single arm has utility 1 whereas all others have utility 0. Here the optimal unconstrained policy has utility 1, while the optimal σ-smooth policy has utility $1/k$. This "cost of smoothness" can be quantified as $u_1 - \frac{1}{k}\sum_{i=1}^{k} u_i$, where $u_1 \geq u_2 \geq u_3 \ldots$ are the utilities of the arms sorted in decreasing order. This holds simply because the optimal σ-smooth policy assigns equal weight to the top k arms, while the optimal unconstrained strategy has utility u_1.

The key property of smooth strategies of not putting too much mass on any particular action can be used to model unpredictable players whose strategies cannot be captured by a "small" set of actions that characterizes their actions. Such strategies can prove beneficial when predicting the move of a player can result in significant negative utility (loss) for such a player. Perhaps surprisingly, our results show that properties of games with unpredictable players allow for efficient (sublinear) verification. Daskalakis et al. [2024] study smooth strategies in games. They provide additional

[2] Informally, in the language of interactive proofs, the aforementioned verification procedure has the completeness property but not soundness.

motivations for studying smooth strategies, and note that a strong σ-smooth NE (where $\varepsilon = 0$) is guaranteed to exist in every normal-form game.

As noted, the study of smooth strategies in the context of normal-form games was recently proposed by Daskalakis et al. [2024]. One motivation for their study of smooth strategies comes from the area of *smoothed analysis* [Spielman and Teng, 2009] which studies the influence of *random noise* on the time complexity of algorithms. There are at least two ways of studying the smoothing influence of noise on complexity issues in games and MABs. One is to assume that the instances (e.g., the payoff matrix of the normal form game) are subjected to random noise. The other, which we focus on here, is to consider *smooth strategies* for players: namely distributions over actions that do not assign too much mass to any one action. For example, for MABs we seek strategies that maximize the expected reward, subject to the constraint that the strategy is a smooth distribution. Daskalakis et al. [2024] observe that distributions that are perturbed with random noise often have such smoothness properties. Additionally, those authors provide several further motivations for studying smooth strategies in games. They also discuss how smooth strategies can be used to generate behavior that is more similar in style to humans. Furthermore, as elaborated by Daskalakis et al. [2024] smoothens is related to several equilibrium notions in game theory.

2.2 Our setting

Verification in multi-armed bandits. In this setting, both a prover and a verifier have access to an n-armed bandit. This access is given via an oracle: one can query the oracle by specifying an arm, and in return receive a reward drawn from that arm's utility distribution. The prover and verifier communicate interactively, and at the end of the interaction the verifier either rejects or outputs a policy. If both the prover and the verifier follow the protocol, the verifier should output an approximately optimal smooth policy with probability at least $2/3$; this property is called *completeness*. The protocol should also satisfy *soundness*, in that even if the prover behaves arbitrarily, the verifier should output a non-optimal or non-smooth policy with probability at most $1/3$. Finally, the protocol should be efficient in that both the prover and verifier run in polynomial time, and the verifier makes $o(n)$ queries.[3]

Verification in normal-form games. We give a high-level overview of our results due to space constraints: details can be found in the supplementary material as well as the arXiv version Christ et al. [2025]. In this setting, both the prover and the verifier have access to a k-player, n-action normal-form game. This access is given via a game oracle: one can query the oracle by specifying a strategy profile,[4] and in return receive a vector of payoffs (one per player) corresponding to a tuple of actions drawn from the specified strategy profile. The prover and the verifier are both given an explicit description of a proposed strategy profile π, as well as an optimality parameter ε, a smoothness parameter σ, and a slackness parameter η. The prover and verifier communicate interactively, and at the end of the interaction the verifier either accepts or rejects. Analogously to the MAB setting, this protocol must satisfy completeness and soundness. Completeness requires that if both parties follow the protocol and π is indeed an ε-approximate strong σ-smooth NE, the verifier accepts with probability at least $2/3$. Soundness requires that for any (possibly malicious and computationally unbounded) prover, if π is not an $(\varepsilon + \eta)$-approximate strong σ-smooth NE, the verifier rejects with probability at least $2/3$. We note that adapting the MAB verification for verification in games requires certain technicalities such as introducing an additional *slackness parameter* η. Details can be found in the Games Section in the Supplementary Material.

2.3 Our contributions

We give a high-level overview of our results. Formal statements and proofs can be found in Section 5 and in the full version Christ et al. [2025]. We construct interactive protocols for MABs and normal-form games, where both the verifier and prover have polynomial running time in all parameters. We also present corresponding lower bounds, showing that verification using our protocols is strictly more efficient than learning in terms of query complexity, and that our protocols have near-optimal dependence on key parameters.

[3] As usual, it is possible to efficiently amplify the success probability from $2/3$ to $1 - \delta$ for δ arbitrarily small, both in our protocols for bandits and in the protocols for games.

[4] As in Footnote 1.

1. **Efficient verification for MABs.** We construct a protocol for verifying ε-optimal σ-smooth strategies where the verifier makes $\tilde{O}\big(\sigma n/\varepsilon^2\big)$ queries to the MAB oracle. The protocol consists of a single $\tilde{O}(n)$-bit message sent from the prover to the verifier. A formal statement is given in Theorem 5.5.

2. **Lower bounds for MABs.** We prove a matching lower bound of $\Omega(\sigma n/\varepsilon^2)$ on the number of queries needed by the verifier in *any* such MAB verification protocol. We also prove that learning approximately-optimal smooth strategies requires $\Omega(n)$ queries, which (together with our verification protocol) implies that verification can be more efficient than learning. A Formal statement appears in the Multi-armed bandits Section in the Supplementary Material.

3. **Lower communication using cryptography.** We show how to obtain an interactive proof for the value of the optimal smooth policy of a bandit, where the asymptotic number of arms pulls is the same as in our original protocol. The prover now sends at most $O\big(\lambda \cdot (n\sigma \log^3(1/\varepsilon)/\varepsilon\big)$ bits while affecting the probability of correctness by a negligible additive error: A constant λ suffices to ensure an additive error of at most 10^{-6}. Therefore, assuming $n\sigma/\varepsilon = o(n)$, we have a protocol with sublinear communication. A formal statement is given in Theorem 5.6.

4. **Efficient verification of smooth NE in games.** For normal-form games with k players and n actions, we construct an interactive protocol for verifying that a given strategy profile is an approximate smooth NE, with slackness $\eta > 0$. That is, the verifier accepts if the input strategy profile is an ε-approximate σ-smooth NE, and it rejects if the input strategy profile is not an $(\varepsilon + \eta)$-approximate σ-smooth NE. The verifier uses $\tilde{O}\big(k\sigma n/\eta^2\big)$ queries to the game oracle. The full statements and proofs appear in the Games Section in the Supplementary Material.

   In contrast, for constant ε, Theorem 4 in Rubinstein [2017] states a lower bound of $2^{\Omega(k)}$ queries to the game oracle for computing an ε-NE without the help of a prover, and this lower bound extends also to computing σ-smooth ε-NE for ε constant and $\sigma = \Theta(1/n)$ (see Remark 12 in Daskalakis et al., 2024).[5] Thus, our verification protocol offers substantial savings in terms of k as well. We also provide a linear matching lower bound based on similar ideas of the lower bound for verifying MABs.

2.4 Proof ideas

Our bandit verification protocol relies on the following observation. Consider a lying prover trying to convince a verifier that a given σ-smooth bandit strategy π is approximately optimal, although there exists another σ-smooth strategy π^* whose expected reward exceeds that of π by more than ε. Consider a protocol that requires the prover to provide a good estimate of the expected utility of each arm. Because π^* is smooth, it follows that in order to conceal the existence of π^*, the prover must lie about the utilities of *many* arms. Thus, it suffices for the verifier to independently estimate the utilities of a few randomly chosen arms, and reject if any of the prover's purported utilities are too far from the verifier's estimates. Importantly, whereas the prover must lie about *many* arms, it is enough for the verifier to catch *just one* lie. In particular, if the prover lies about an β fraction of utilities, the verifier only needs to query roughly $1/\beta$ arms in order to detect the lie with constant probability. In more detail, assume that the verifier is promised that, if the prover is lying enough to require rejection (the lie distorts the utility of the optimal policy by at least ε), then the prover is in particular lying by more than α on at least a β fraction of the arms for some specific known values $\alpha, \beta > 0$. This knowledge implies that $\beta n \sigma \alpha \geq \varepsilon$ and also $\alpha \geq \varepsilon$. Based on the promise, the verifier can detect the lie by pulling $O(\frac{1}{\beta} \cdot \frac{1}{\alpha^2})$ arms which by the above inequalities is at most $O(n\sigma/\varepsilon^2)$. While α, β are not known, the verifier can use a simple partitioning scheme to guess them, which adds a logarithmic term to the query complexity of the verifier.

Our lower bound of $\Omega(\sigma n/\varepsilon^2)$ on the number of queries needed for MAB verification relies on a reduction to the coin problem where one needs to decide with a few queries whether a given coin has bias $1/2 - \varepsilon$ or bias $1/2 + \varepsilon$. Our proof uses ideas from Even-Dar et al. [2002].

[5] Remark 12 in Daskalakis et al., 2024 refers to the case where ε and σ are both constant. However, as discussed in Footnote 8, this corresponds to $\sigma = \Theta(1/n)$ in our notation for smoothness.

Using succinct non-interactive arguments of knowledge (SNARKs) and vector commitments, we significantly reduce the communication between the prover and the verifier in our bandit verification protocol (the Preliminaries Section gives more details about these cryptographic tools). A vector commitment allows one to commit to a vector and reveal individual components whose consistency with the commitment can be proven. The commitment and opening proofs require space independent of the length of the vector. A SNARK allows a prover to succinctly prove that a given instance belongs to a polynomial-time computable relation.

A central observation in our NE verification protocol is that we can use the bandit verification protocol to ensure that no player has a profitable smooth deviation from a proposed strategy profile π. That is, for each $i \in [n]$, if all players except player i behave according to the profile π, then player i is choosing between n actions, each of which has some fixed reward distribution.[6] This is exactly an n-armed bandit. So player i has no profitable smooth deviation if and only if its current strategy is an optimal smooth strategy for an appropriately defined bandit. Our NE verification protocol essentially performs this bandit verification k times, once for each player. We also show that a linear dependence on k is necessary.

3 Related work

There is a substantial body of work studying algorithms for MABs for finding a distribution over actions (or a single action) that maximizes expected utility [Lattimore and Szepesvári, 2020, Garivier and Kaufmann, 2016]. Several works demonstrate a lower bound of $\Omega(n)$ (where n is the number of arms) on the number of arm-pulls that are needed to find a strategy that maximizes utility for the player [Karnin et al., 2013, Chen and Li, 2015, Chen et al., 2017, Mannor and Tsitsiklis, 2004a, Even-Dar et al., 2002, Assadi and Wang, 2022]. Specifically, Even-Dar et al. [2002] provides an algorithm for identifying the best arm up to an additive error of ε with probability of success at least $1-\delta$. They achieve query complexity of $O(n\log(1/\delta)/\varepsilon^2)$ improving the naive algorithm whose query complexity is $O(n\log(n/\delta)/\varepsilon^2)$. Furthermore, they complement their result by providing a matching lower bound on the query complexity of any algorithm achieving such error guarantees in the PAC framework based on a reduction to the coin problem. A lower bound of $\Omega(n\log(1/\delta)/\varepsilon^2)$ was obtained using different methods in Mannor and Tsitsiklis [2004a]. They also show a lower bound of $\Omega((\frac{1}{\varepsilon^2})(n+\log(1/\delta)))$ for the case where the vector of utilities of the arms is known (but it is not known which arm has which utility) and the goal is to find the arm with the largest expected utility. There is also extensive work on regret minimization strategies in MABs [Lai and Robbins, 1985], a setting that we do not touch on in this paper.

A rich line of work examines the complexity of finding Nash equilibria (NE). The problem of finding an approximate NE in a normal-form game with at least 3 players is known to be complete for the complexity class PPAD [Daskalakis et al., 2009]. Even for the seemingly simpler case of 2 players, computing an exact NE is PPAD-complete [Chen and Deng, 2006], and computing an approximate NE is hard under the Exponential Time Hypothesis for PPAD [Rubinstein, 2017]. While finding a NE is hard, verifying that a given strategy profile is an (approximate) NE can be done in polynomial time.[7] Finally, the study of the query complexity of approximate NE in 2-player normal-form games has received significant attention [Babichenko, 2019, Fearnley et al., 2015, Göös and Rubinstein, 2023]. It is known [Göös and Rubinstein, 2023] that the trivial upper bound of n^2 is nearly tight: In certain games $\Omega(n^{2-o(1)})$ queries may be needed to find an approximate NE.

The intractability of finding NE or ε-NE in arbitrary normal-form games motivated Daskalakis et al. [2024] to introduce the notion of σ-smooth NE where each player places a probability mass of at most $1/(n\sigma)$ for each of the n actions and σ is a smoothness parameter in $[1/n, 1]$ (their parametrization of smoothness differs from ours: recall that we call a strategy smooth if for every action the mass on the action is at most σ). They consider two equilibrium notions related to smoothness: in a *strong* σ-smooth NE, each player is executing a smooth strategy and cannot improve their utility by deviating unilaterally to a smooth strategy. In a *weak* σ-smooth NE, again no player can improve their utility by deviating unilaterally to a smooth strategy; however, the strategies of players in a weak σ-smooth NE need not be σ-smooth. Analogous notions are defined by Daskalakis et al. [2024]

[6] The reward distribution for each action is fixed, because the reward distribution is a function of the strategies of the remaining players, which are fixed according to π.

[7] This is true for any problem in FNP, of which PPAD is a subset.

for ε-approximate (smooth) NE where players cannot improve their utilities by more than ε. For strong ε-approximate σ-smooth equilibrium, Daskalakis et al. [2024] prove there exists an algorithm for finding such an equilibrium in time $n^{O\left(k^4 \log(k/\varepsilon)/\varepsilon^2\right)}$, where k is the number of players. For weak ε-approximate σ-smooth NE, they offer an algorithm with runtime complexity independent of n (the runtime depends only on the number of players k, the smoothness parameter σ, and the approximation parameter ε).

Several works study interactive proofs for machine learning [Goldwasser et al., 2021, Mutreja and Shafer, 2023, Gur et al., 2024, Caro et al., 2024b,a]. Their main focus is on verification in supervised learning and similar settings. For instance, verifying that a proposed hypothesis satisfies the agnostic PAC requirement with respect to a (fixed and known) hypothesis class and a (fixed but unknown) population distribution, using less access (samples or queries) to the population distribution than is necessary for agnostic PAC learning. The questions of verifying MAB strategies and NE in normal-form games with few queries to the bandit or game oracle are not studied there.

4 Preliminaries

Let $\mathbb{N} = \{1, 2, 3, \dots\}$. For $n \in \mathbb{N}$, we denote by $[n]$ the set $\{1, 2, \dots, n\}$ and assume throughout that the set of actions of players both in MABs and games is $[n]$. For a set Ω, we write $\Delta(\Omega)$ to denote the set of all probability measures defined on the measurable space $(\Omega, \mathcal{F})$, where $\mathcal{F}$ is some fixed σ-algebra that is implicitly understood. We often identify a distribution $p \in \Delta([n])$ with the vector $p = (p_1, \dots, p_n)$ such that $p_i = p(i) = \mathbb{P}_{x \sim p}[x = i]$. Finally, we define the notion of a smooth distribution:

Definition 4.1. *Let $n \in \mathbb{N}$ and $\sigma \in [1/n, 1]$. A probability distribution $p \in \Delta([n])$ is called σ-smooth if for every $i \in [n]$, $p_i \leq \sigma$.*

The degree of smoothness of a distribution is governed by the parameter σ. For $\sigma = 1$, smoothness is vacuous as every probability distribution is 1-smooth. On the other extreme, when $\sigma = 1/n$ the distribution is the smoothest possible: the uniform distribution.

5 Bandits

5.1 Definitions

Definition 5.1 (Bandit)**.** *Let $n \in \mathbb{N}$. An n-arm bandit is a vector of n distributions $q = (q_1, \dots, q_n) \in (\Delta([0,1]))^n$.*

A bandit defines a bandit oracle such that, given a query $i \in [n]$ (corresponding to "pulling the i-th arm of the bandit"), the oracle returns a utility $x \sim q_i$ sampled independently of all previous oracle queries and responses.

The expected utilities vector of q is a vector $u = \mathrm{utility}(q) \in [0,1]^n$ such that $u_i = \mathbb{E}_{x \sim q_i}[x]$ for all $i \in [n]$. A strategy for an n-arm bandit is a distribution $\pi = (\pi_1, \dots, \pi_n) \in \Delta([n])$. The expected utility of π with respect to u is $\mathbb{E}_{i \sim \pi, x \sim q_i}[x] = \sum_{i=1}^{n} \pi_i u_i = \pi \cdot u$.

Definition 5.2 (Smooth bandit strategy)**.** *Let $n \in \mathbb{N}$ and $\sigma \in [1/n, 1]$. A strategy $\pi \in \Delta([n])$ for an n-arm bandit is σ-smooth if $\pi_i \leq \sigma$ for all $i \in [n]$.*

Definition 5.3 (Optimal smooth bandit strategy)**.** *Let $n \in \mathbb{N}$, $\varepsilon \geq 0$, $\sigma \in [1/n, 1]$, let $u \in [0,1]^n$ be the expected utilities vector of an n-arm bandit, and let $\pi \in \Delta([n])$ be a strategy. We say that π is ε-competitive with respect to σ-smooth policies for u, if for every σ-smooth strategy $\pi' \in \Delta([n])$,*

$$\pi' \cdot u - \pi \cdot u \leq \varepsilon.$$

If in addition π is σ-smooth, then we say that π is an ε-optimal σ-smooth strategy for u.[8]

Definition 5.4 (Verification of optimality for smooth bandit strategies)**.** *An interactive proof system for verification of ε-optimal σ-smooth policies for n-arm bandits is a pair of algorithms (V, P) such*

[8] These are special cases of definitions in Daskalakis et al. [2024]. The first definition corresponds to a weak ε-approximate σ'-smooth Nash equilibrium for u, and the second definition corresponds to a strong ε-approximate σ'-smooth Nash equilibrium for u, where $\sigma' = 1/(n\sigma)$.

that for all $n \in \mathbb{N}$, and for every n-arm bandit q with expected utilities vector $u = \text{utility}(q) \in [0,1]^n$ and bandit oracle $\mathcal{O}_q$, and for all $\sigma \in [1/n, 1]$ and $\varepsilon \geq 0$, the following two conditions hold:

- **Completeness**. *Let the random variable*
$$\pi_V = \left[V^{\mathcal{O}_q}(n, \varepsilon, \sigma), P^{\mathcal{O}_q}(n, \varepsilon, \sigma)\right] \in \Delta([n]) \cup \{\mathsf{reject}\}$$
denote the output of V after interacting with P, when each of them receives (n, ε, σ) as input and has oracle access to $\mathcal{O}_q$. Then
$$\mathbb{P}[(\pi_V \neq \mathsf{reject}) \ \wedge \ (\forall\ \sigma\text{-smooth } \pi' \in \Delta([n]) : \ \pi' \cdot u - \pi_V \cdot u \leq \varepsilon)] \geq \frac{2}{3}.$$
- **Soundness**. *For any (possibly malicious and computationally unbounded) prover P' (which in particular may depend on n, ε, σ and q), the verifier's output $\pi_V = \left[V^{\mathcal{O}_q}(n, \varepsilon, \sigma), P'\right] \in \Delta([n]) \cup \{\mathsf{reject}\}$ satisfies*
$$\mathbb{P}[(\pi_V = \mathsf{reject}) \ \vee \ (\forall\ \sigma\text{-smooth } \pi' \in \Delta([n]) : \ \pi' \cdot u - \pi_V \cdot u \leq \varepsilon)] \geq \frac{2}{3}.$$

In both conditions, the probability is over the randomness of $\mathcal{O}_q$ and V, as well as P or P'.

Theorem 5.5 (Verification for bandits)**.** *Let $n \in \mathbb{N}$, let $\sigma \in [1/n, 1]$, let $\varepsilon \geq 0$. There exists an interactive proof system (V, P) for verification of ε-optimal σ-smooth policies for n-arm bandits such that:*

- *The protocol consists of a single message of $O(n \log(1/\varepsilon))$ bits sent from P to V.*
- *P performs $m_P = O\big(n \log(n/\varepsilon)/\varepsilon^2\big)$ nonadaptive queries to the bandit oracle and runs in time $\mathsf{poly}(n, 1/\varepsilon)$.*
- *V performs*
$$m_V = O\left(\frac{n\sigma}{\varepsilon^2} \cdot \log\left(\frac{n\sigma}{\varepsilon}\right) \log\left(\frac{1}{\varepsilon}\right)\right)$$
nonadaptive queries to the bandit oracle, and runs in time $\mathsf{poly}(n, 1/\varepsilon)$.

In particular, if $\sigma = \Theta(1/\sqrt{n})$ then $m_V = \tilde{O}(\sqrt{n})$, and if $\sigma = \Theta(1/n)$ then m_V is independent of n.

The pseudocode of the verification protocol and the proof of Theorem 5.5 appear in the Multi-armed bandits Section in the Supplementary Material.

5.2 Vector commitments

Cryptography preliminaries Let $\lambda \in \mathbb{N}$ denote the security parameter. We write p.p.t. to mean probabilistic polynomial time. We let $\mathsf{negl}(\lambda)$ denote a function that is $O(1/\lambda^c)$ for all $c > 0$.

A *vector commitment* [Catalano and Fiore, 2013] is a tuple of p.p.t. algorithms:

- $\mathsf{KeyGen}(1^\lambda, n) \to \mathsf{pp}$: takes as input the security parameter and size n of the vectors to be committed, and outputs public parameters pp.
- $\mathsf{Commit}_{\mathsf{pp}}(v) \to c_v, \mathsf{aux}$: takes as input a length-n vector v, and outputs a commitment c_v and auxiliary information aux. aux often contains the entire committed vector v.
- $\mathsf{Open}_{\mathsf{pp}}(v_i, i, \mathsf{aux}) \to \mathsf{pf}$: takes as input a value v_i, an index i, and auxiliary information aux. It outputs a proof pf that v_i is the i^{th} component of v corresponding to aux.
- $\mathsf{Verify}_{\mathsf{pp}}(c_v, v_i, i, \mathsf{pf}) \to \{\mathsf{accept}, \mathsf{reject}\}$: takes as input a commitment c_v, a value v_i, an index i, and a proof pf. It accepts if and only if c_v commits to a vector whose i^{th} component is v_i.

Vector commitments must satisfy *correctness* and *position binding*. Correctness requires that with overwhelming probability, any honestly generated public parameters and honestly committed vectors yield valid opening proofs for all of their components. Position binding requires that it is infeasible for any non-uniform p.p.t. adversary to produce a commitment and two valid proofs for *different* openings of that commitment. For precise statements, please see the Cryptography Section in the Supplementary Material.

5.3 Succinct non-interactive arguments of knowledge

In our protocols we use succinct non-interactive arguments of knowledge (SNARKs). More information regarding SNARKs can be found in Groth [2016].

A succinct non-interactive argument of knowledge for a relation generator $\mathcal{R}$ is a tuple of p.p.t. algorithms:

- $\mathsf{Setup}(1^\lambda, R) \to \mathsf{pp}, \tau$: takes as input the security parameter and a relation $R \in \mathcal{R}$, and outputs public parameters pp and a simulation trapdoor τ.
- $\mathsf{Prove}(R, \mathsf{pp}, \phi, w) \to \mathsf{pf}$: takes as input a relation R, public parameters pp, and a statement-witness pair $(\phi, w) \in R$. It outputs a proof pf of this pair's membership in the relation.
- $\mathsf{Verify}(R, \mathsf{pp}, \phi, \mathsf{pf}) \to \{\mathsf{accept}, \mathsf{reject}\}$: takes as input a relation R, public parameters pp, a statement ϕ, and a proof pf. It should accept if and only if ϕ has a witness w such that $(\phi, w) \in R$.

We consider SNARKs that satisfy *perfect completeness* and *computational knowledge soundness*.

Perfect completeness requires that for all $\lambda \in \mathbb{N}$, $R \in \mathcal{R}$, and $(\phi, w) \in R$:

$$\Pr_{(\mathsf{pp},\tau)\leftarrow\mathsf{Setup}(1^\lambda,R)}[\mathsf{pf} \leftarrow \mathsf{Prove}(R, \mathsf{pp}, \phi, w) \ : \ \mathsf{accept} \leftarrow \mathsf{Verify}(R, \mathsf{pp}, \phi, \mathsf{pf})]$$

Computational knowledge soundness requires that there exists a non-uniform p.p.t. extractor that can extract a witness whenever an adversary can compute an accepting proof. That is, for all non-uniform adversaries $\mathcal{A}$, there exists a non-uniform p.p.t. extractor $\mathcal{X}_\mathcal{A}$ such that

$$\Pr\left[\begin{array}{l|l} (\phi, w) \notin R \text{ and} & (R, z) \leftarrow \mathcal{R}(1^\lambda) \\ \mathsf{Verify}(R, \mathsf{pp}, \phi, \mathsf{pf}) \to \mathsf{accept} & (\phi, \tau) \leftarrow \mathsf{Setup}(1^\lambda) \\ & ((\phi, w), \mathsf{pf}) \leftarrow (\mathcal{A}||\mathcal{X}_\mathcal{A})(R, z, \mathsf{pp}) \end{array}\right] \leq \mathsf{negl}(\lambda),$$

where $(\mathcal{A}||\mathcal{X}_\mathcal{A})$ denotes that the extractor has access to the adversary's internal state and randomness.

5.3.1 A low-communication protocol variant

Here we outline how to use vector commitments and SNARKs in a new protocol with lower communication cost by having the prover send a commitment to the vector $\tilde{u}$ of purported rewards, rather than sending $\tilde{u}$ in full. It uses a SNARK to prove that the optimal smooth policy with respect to $\tilde{u}$ has a claimed value. The verifier then proceeds exactly as in the smooth bandit verification protocol; but instead of examining $\tilde{u}$ directly at each index, it asks the prover for the value and opening proof. Below, we sketch the ideas behind the proof of correctness. The implementation of the protocol as well as the bandit verification protocol on which the low communication variant builds on, can be found in the Bandit Section in the Supplementary Material.

Theorem 5.6. *Let $\lambda \in \mathbb{N}$ be the security parameter, let $n \in \mathbb{N}$, let $\varepsilon \in (0, 1)$, let q be an n-armed bandit, and let u denote be the vector of expected utilities of q. There exists a protocol with the following properties. The protocol consists of a trusted setup phase, in which shared parameters are generated by a trusted entity; and an interactive phase between a prover and a verifier. Assuming the security of the underlying SNARK Π and vector commitment VC, our protocol satisfies:*

- ***Completeness:*** *If the prover behaves honestly, the verifier outputs a value t that is within ε of the value of the optimal σ-smooth policy with probability at least $\frac{2}{3} - \mathsf{negl}(\lambda)$.*
- ***Soundness:*** *Even if the p.p.t. prover behaves arbitrarily, the probability that the verifier outputs a value t that is* not *within ε of the optimal value is at most $\frac{1}{3} + \mathsf{negl}(\lambda)$.*

The efficiency of the protocol is as follows:

- *If Π has constant-sized proofs, and VC has constant-sized commitments and opening proofs, the protocol consists of $O\big(\lambda \cdot (n\sigma \log^3(1/\varepsilon)/\varepsilon\big)$ bits sent between P and V.*
- *P performs $m_P = O\big(n \log(n/\varepsilon)/\varepsilon^2\big)$ nonadaptive queries to the bandit oracle and runs in time $\mathsf{poly}(n, 1/\varepsilon)$.*

- *V performs*

$$m_V = O\left(\frac{n\sigma}{\varepsilon^2} \cdot \log\left(\frac{n\sigma}{\varepsilon}\right) \log\left(\frac{1}{\varepsilon}\right)\right)$$

nonadaptive queries to the bandit oracle, and runs in time $\mathsf{poly}(n, 1/\varepsilon)$.

Proof. **Communication** The prover sends a commitment and SNARK proof, each of which consist of $O(\lambda)$ bits. In the interactive phase, the verifier sends $O(a_b \log(1/\epsilon)) = O(n\sigma \log^2(1/\epsilon)/\epsilon)$ indices, each of which can be written in $\log(1/\epsilon)$ bits. The prover sends $O(a_b \log(1/\epsilon))$ openings and opening proofs, requiring $O((n\sigma \log^3(1/\epsilon)/\epsilon)$ bits in total. The number of queries made by the prover and verifier to the bandit oracle is identical to in bandit verification protocol.

We prove the soundness of the protocol. Completeness is proved in the Supplementary Material.

Soundness Towards a contradiction, consider a p.p.t. adversary $\mathcal{A}$ that acts as the prover and with probability at least $1/3 + 1/\mathsf{poly}(\lambda)$ causes the verifier to output t that is ϵ-far from the true value of the optimal σ-smooth policy for some bandit q. Recall that computational knowledge soundness of Π implies that there exists a p.p.t. extractor $\mathcal{X}_{\mathcal{A}}$ that, with overwhelming probability, computes $(\pi, \tilde{u})$ such that $(c_\pi, c_v, t; \pi, \tilde{u}) \in \mathcal{R}_{\mathsf{VC.pp},n}$. That is, c_v is a commitment to $\tilde{u}$, π is indeed an optimal σ-smooth policy for $\tilde{u}$, and the value of π is indeed t. Now, position binding of the vector commitment implies that with overwhelming probability all openings of $\tilde{u}$ that the prover sends to the verifier are either rejected, or indeed match the corresponding component of $\tilde{u}$.

Soundness of the protocol now follows exactly from the analysis of the bandit verification protocol. □

We remark that there exist SNARKs with constant $(O(\lambda))$-sized proofs for arithmetic circuit satisfiability, which have knowledge soundness in the generic group model [Groth, 2016]. There also exist vector commitments with constant-sized commitments and opening proofs; for example, the CDH-based scheme of Catalano and Fiore [2013].

6 Conclusion

We have studied protocols to verify near optimality of protocols and strategies for MABs and games. A natural direction is to study interactive proofs for near optimality of policies in Markov Decision Processes Sutton and Barto [2018] with horizon larger than 1. It appears that additional ideas are needed in this setting to deal with the network structure associated with the transitions of the MDP which is absent in the MAB setting. Additionally, empirical validation of the protocols here is of interest.

Many games include multiple sequential actions of players. Such games are often represented in *extensive-form*. Devising protocols to verify properties of strategies in extensive-form is an interesting future research direction.

Acknowledgments and Disclosure of Funding

MC was partially supported by a Google CyberNYC grant, an Amazon Research Award, and NSF grants CCF-2312242, CCF-2107187, and CCF-2212233. JS was partially supported by NSF CNS-2154149, an Amazon Research Award, and by Vinod Vaikuntanathan's Simons Investigator Award.

References

Sepehr Assadi and Chen Wang. Single-pass streaming lower bounds for multi-armed bandits exploration with instance-sensitive sample complexity. In Sanmi Koyejo, S. Mohamed, A. Agarwal, Danielle Belgrave, K. Cho, and A. Oh, editors, *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*, 2022. URL http://papers.nips.cc/paper_files/paper/2022/hash/d5e9cf50dc182447a916bc56d4d42942-Abstract-Conference.html.

Yakov Babichenko. Informational bounds on equilibria (a survey). *SIGecom Exch.*, 17(2):25–45, 2019. doi:10.1145/3381329.3381333. URL https://doi.org/10.1145/3381329.3381333.

Ran Canetti and Ari Karchmer. Covert learning: How to learn with an untrusted intermediary. In Kobbi Nissim and Brent Waters, editors, *Theory of Cryptography - 19th International Conference, TCC 2021, Raleigh, NC, USA, November 8-11, 2021, Proceedings, Part III*, volume 13044 of *Lecture Notes in Computer Science*, pages 1–31. Springer, 2021. doi:10.1007/978-3-030-90456-2_1. URL https://doi.org/10.1007/978-3-030-90456-2_1.

Matthias C. Caro, Jens Eisert, Marcel Hinsche, Marios Ioannou, Alexander Nietner, and Ryan Sweke. Interactive proofs for verifying (quantum) learning and testing. *CoRR*, abs/2410.23969, 2024a. doi:10.48550/ARXIV.2410.23969. URL https://doi.org/10.48550/arXiv.2410.23969.

Matthias C. Caro, Marcel Hinsche, Marios Ioannou, Alexander Nietner, and Ryan Sweke. Classical verification of quantum learning. In Venkatesan Guruswami, editor, *15th Innovations in Theoretical Computer Science Conference, ITCS 2024, January 30 to February 2, 2024, Berkeley, CA, USA*, volume 287 of *LIPIcs*, pages 24:1–24:23. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2024b. doi:10.4230/LIPICS.ITCS.2024.24. URL https://doi.org/10.4230/LIPIcs.ITCS.2024.24.

Dario Catalano and Dario Fiore. Vector commitments and their applications. In Kaoru Kurosawa and Goichiro Hanaoka, editors, *Public-Key Cryptography - PKC 2013 - 16th International Conference on Practice and Theory in Public-Key Cryptography, Nara, Japan, February 26 - March 1, 2013. Proceedings*, volume 7778 of *Lecture Notes in Computer Science*, pages 55–72. Springer, 2013. doi:10.1007/978-3-642-36362-7_5. URL https://doi.org/10.1007/978-3-642-36362-7_5.

Moses Charikar, Jacob Steinhardt, and Gregory Valiant. Learning from untrusted data. In Hamed Hatami, Pierre McKenzie, and Valerie King, editors, *Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing, STOC 2017, Montreal, QC, Canada, June 19-23, 2017*, pages 47–60. ACM, 2017. doi:10.1145/3055399.3055491. URL https://doi.org/10.1145/3055399.3055491.

Lijie Chen and Jian Li. On the optimal sample complexity for best arm identification. *CoRR*, abs/1511.03774, 2015. URL http://arxiv.org/abs/1511.03774.

Lijie Chen, Jian Li, and Mingda Qiao. Towards instance optimal bounds for best arm identification. In Satyen Kale and Ohad Shamir, editors, *Proceedings of the 30th Conference on Learning Theory, COLT 2017, Amsterdam, The Netherlands, 7-10 July 2017*, volume 65 of *Proceedings of Machine Learning Research*, pages 535–592. PMLR, 2017. URL http://proceedings.mlr.press/v65/chen17b.html.

Xi Chen and Xiaotie Deng. Settling the complexity of two-player Nash equilibrium. In *47th Annual IEEE Symposium on Foundations of Computer Science (FOCS 2006), 21-24 October 2006, Berkeley, California, USA, Proceedings*, pages 261–272. IEEE Computer Society, 2006. doi:10.1109/FOCS.2006.69. URL https://doi.org/10.1109/FOCS.2006.69.

Miranda Christ, Daniel Reichman, and Jonathan Shafer. Protocols for verifying smooth strategies in bandits and games. *CoRR*, abs/2507.10567, 2025. doi:10.48550/ARXIV.2507.10567. URL https://doi.org/10.48550/arXiv.2507.10567.

Constantinos Daskalakis, Paul W. Goldberg, and Christos H. Papadimitriou. The complexity of computing a Nash equilibrium. *Commun. ACM*, 52(2):89–97, 2009. doi:10.1145/1461928.1461951. URL https://doi.org/10.1145/1461928.1461951.

Constantinos Daskalakis, Noah Golowich, Nika Haghtalab, and Abhishek Shetty. Smooth Nash equilibria: Algorithms and complexity. In Venkatesan Guruswami, editor, *15th Innovations in Theoretical Computer Science Conference, ITCS 2024, January 30 to February 2, 2024, Berkeley, CA, USA*, volume 287 of *LIPIcs*, pages 37:1–37:22. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2024. doi:10.4230/LIPICS.ITCS.2024.37. URL https://doi.org/10.4230/LIPIcs.ITCS.2024.37.

Eyal Even-Dar, Shie Mannor, and Yishay Mansour. PAC bounds for multi-armed bandit and markov decision processes. In Jyrki Kivinen and Robert H. Sloan, editors, *Computational Learning Theory, 15th Annual Conference on Computational Learning Theory, COLT 2002, Sydney, Australia, July 8-10, 2002, Proceedings*, volume 2375 of *Lecture Notes in Computer Science*, pages 255–270. Springer, 2002. doi:10.1007/3-540-45435-7_18. URL https://doi.org/10.1007/3-540-45435-7_18.

John Fearnley, Martin Gairing, Paul W. Goldberg, and Rahul Savani. Learning equilibria of games via payoff queries. *J. Mach. Learn. Res.*, 16:1305–1344, 2015. doi:10.5555/2789272.2886792. URL https://dl.acm.org/doi/10.5555/2789272.2886792.

Aurélien Garivier and Emilie Kaufmann. Optimal best arm identification with fixed confidence. In Vitaly Feldman, Alexander Rakhlin, and Ohad Shamir, editors, *Proceedings of the 29th Conference on Learning Theory, COLT 2016, New York, USA, June 23-26, 2016*, volume 49 of *JMLR Workshop and Conference Proceedings*, pages 998–1027. JMLR.org, 2016. URL http://proceedings.mlr.press/v49/garivier16a.html.

Shafi Goldwasser, Silvio Micali, and Charles Rackoff. The knowledge complexity of interactive proof systems. *SIAM J. Comput.*, 18(1):186–208, 1989. doi:10.1137/0218012. URL https://doi.org/10.1137/0218012.

Shafi Goldwasser, Yael Tauman Kalai, and Guy N. Rothblum. Delegating computation: Interactive proofs for muggles. *J. ACM*, 62(4):27:1–27:64, 2015. doi:10.1145/2699436. URL https://doi.org/10.1145/2699436.

Shafi Goldwasser, Guy N. Rothblum, Jonathan Shafer, and Amir Yehudayoff. Interactive proofs for verifying machine learning. In James R. Lee, editor, *12th Innovations in Theoretical Computer Science Conference, ITCS 2021, January 6-8, 2021, Virtual Conference*, volume 185 of *LIPIcs*, pages 41:1–41:19. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2021. doi:10.4230/LIPICS.ITCS.2021.41. URL https://doi.org/10.4230/LIPIcs.ITCS.2021.41.

Mika Göös and Aviad Rubinstein. Near-optimal communication lower bounds for approximate Nash equilibria. *SIAM J. Comput.*, 52(6):S18–316, 2023. doi:10.1137/19M1242069. URL https://doi.org/10.1137/19m1242069.

Jens Groth. On the size of pairing-based non-interactive arguments. In Marc Fischlin and Jean-Sébastien Coron, editors, *Advances in Cryptology - EUROCRYPT 2016 - 35th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Vienna, Austria, May 8-12, 2016, Proceedings, Part II*, volume 9666 of *Lecture Notes in Computer Science*, pages 305–326. Springer, 2016. doi:10.1007/978-3-662-49896-5_11. URL https://doi.org/10.1007/978-3-662-49896-5_11.

Tom Gur, Mohammad Mahdi Jahanara, Mohammad Mahdi Khodabandeh, Ninad Rajgopal, Bahar Salamatian, and Igor Shinkar. On the power of interactive proofs for learning. In Bojan Mohar, Igor Shinkar, and Ryan O'Donnell, editors, *Proceedings of the 56th Annual ACM Symposium on Theory of Computing, STOC 2024, Vancouver, BC, Canada, June 24-28, 2024*, pages 1063–1070. ACM, 2024. doi:10.1145/3618260.3649784. URL https://doi.org/10.1145/3618260.3649784.

Kwang-Sung Jun, Lihong Li, Yuzhe Ma, and Xiaojin (Jerry) Zhu. Adversarial attacks on stochastic bandits. In Samy Bengio, Hanna M. Wallach, Hugo Larochelle, Kristen Grauman, Nicolò Cesa-Bianchi, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*, pages 3644–3653, 2018. URL https://proceedings.neurips.cc/paper/2018/hash/85f007f8c50dd25f5a45fca73cad64bd-Abstract.html.

Zohar Shay Karnin, Tomer Koren, and Oren Somekh. Almost optimal exploration in multi-armed bandits. In *Proceedings of the 30th International Conference on Machine Learning, ICML 2013, Atlanta, GA, USA, 16-21 June 2013*, volume 28 of *JMLR Workshop and Conference Proceedings*, pages 1238–1246. JMLR.org, 2013. URL http://proceedings.mlr.press/v28/karnin13.html.

Michael J. Kearns, Yishay Mansour, and Andrew Y. Ng. A sparse sampling algorithm for near-optimal planning in large markov decision processes. *Mach. Learn.*, 49(2-3):193–208, 2002. doi:10.1023/A:1017932429737. URL https://doi.org/10.1023/A:1017932429737.

Tze Leung Lai and Herbert Robbins. Asymptotically efficient adaptive allocation rules. *Advances in applied mathematics*, 6(1):4–22, 1985.

Tor Lattimore and Csaba Szepesvári. *Bandit algorithms*. Cambridge University Press, 2020.

Shie Mannor and John N. Tsitsiklis. The sample complexity of exploration in the multi-armed bandit problem. *J. Mach. Learn. Res.*, 5:623–648, 2004a. URL https://jmlr.org/papers/volume5/mannor04b/mannor04b.pdf.

Shie Mannor and John N Tsitsiklis. The sample complexity of exploration in the multi-armed bandit problem. *Journal of Machine Learning Research*, 5(Jun):623–648, 2004b.

Saachi Mutreja and Jonathan Shafer. PAC verification of statistical algorithms. In Gergely Neu and Lorenzo Rosasco, editors, *The Thirty Sixth Annual Conference on Learning Theory, COLT 2023, 12-15 July 2023, Bangalore, India*, volume 195 of *Proceedings of Machine Learning Research*, pages 5021–5043. PMLR, 2023. URL https://proceedings.mlr.press/v195/mutreja23a.html.

Aviad Rubinstein. Settling the complexity of computing approximate two-player Nash equilibria. *SIGecom Exch.*, 15(2):45–49, 2017. doi:10.1145/3055589.3055596. URL https://doi.org/10.1145/3055589.3055596.

Adi Shamir. IP = PSPACE. *J. ACM*, 39(4):869–877, 1992. doi:10.1145/146585.146609. URL https://doi.org/10.1145/146585.146609.

Daniel A. Spielman and Shang-Hua Teng. Smoothed analysis: an attempt to explain the behavior of algorithms in practice. *Commun. ACM*, 52(10):76–84, 2009. doi:10.1145/1562764.1562785. URL https://doi.org/10.1145/1562764.1562785.

Richard S. Sutton and Andrew G. Barto. *Reinforcement learning - an introduction, 2nd Edition*. MIT Press, 2018. URL http://www.incompleteideas.net/book/the-book-2nd.html.

Xuezhou Zhang, Yiding Chen, Xiaojin Zhu, and Wen Sun. Corruption-robust offline reinforcement learning. In Gustau Camps-Valls, Francisco J. R. Ruiz, and Isabel Valera, editors, *International Conference on Artificial Intelligence and Statistics, AISTATS 2022, 28-30 March 2022, Virtual Event*, volume 151 of *Proceedings of Machine Learning Research*, pages 5757–5773. PMLR, 2022. URL https://proceedings.mlr.press/v151/zhang22c.html.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The paper establishes the contributions and scope stated in the abstract

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We clearly elaborate on the assumptions we make in the paper

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

Justification: We provide full proofs of all claims, either in the paper or the Supplementary Material.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [NA]

   Justification: This paper does not have experiments

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [NA]

Justification: No code or data in the paper (other than pseudocode visible to everyone reading the paper)

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [NA]

   Justification: No experiments in the paper

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [NA]

   Justification: No experiments in the paper

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).

- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [NA]

   Justification: No experiments in the paper

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: We reviewed the NeurIPS code of ethics and made sure there are no violations

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: We discuss possible benefits of verification of multi-armed bandits. The discussion is very theoretical and high-level

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.

- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: No data or models in our paper

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: We do not have code, data or models in our purely theoretical paper.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.

- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [NA]

    Justification: We do not have new assets in the paper

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: No crowdsourcing in our paper

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: No research including human subjects was done in our paper

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

Answer: [Yes]

Justification: Content in the paper is original and was created by the authors without resorting to LLMs (other than minor language edits)

Guidelines:

- The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy (`https://neurips.cc/Conferences/2025/LLM`) for what should or should not be described.