ARTIFICIAL REPLAY: A META-ALGORITHM FOR HARNESSING HISTORICAL DATA IN BANDITS

Anonymous authors
Paper under double-blind review

ABSTRACT

While standard bandit algorithms sometimes incur high regret, their performance can be greatly improved by "warm starting" with historical data. Unfortunately, how best to incorporate historical data is unclear: naively initializing reward estimates using all historical samples can suffer from *spurious data* and *imbalanced data coverage*, leading to computational and storage issues—particularly in continuous action spaces. We address these two challenges by proposing ARTIFICIAL REPLAY, a meta-algorithm for incorporating historical data into any arbitrary base bandit algorithm. ARTIFICIAL REPLAY uses only a subset of the historical data *as needed* to reduce computation and storage. We show that for a broad class of base algorithms that satisfy *independence of irrelevant data* (IIData), a novel property that we introduce, our method achieves equal regret as a full warm-start approach while potentially using only a fraction of the historical data. We complement these theoretical results with a case study of K-armed and continuous combinatorial bandit algorithms, including on a green security domain using real poaching data, to show the practical benefits of ARTIFICIAL REPLAY in achieving optimal regret alongside low computational and storage costs.

1 INTRODUCTION

Multi-armed bandits and their variants are robust models for many real-world problems. Resulting algorithms have been applied to wireless networks (Zuo & Joe-Wong, 2021), COVID testing regulations (Bastani et al., 2021), and conservation efforts to protect wildlife from poaching (Xu et al., 2021). Typical bandit algorithms assume no prior knowledge of the expected rewards of each action, simply taking actions online to address the exploration–exploitation trade-off. However, many real-world applications offer access to *historical data*. For example, in the wildlife conservation setting, we may have access to years of historical patrol records that should be incorporated to learn poaching risk before deploying any bandit algorithm; we incur significant losses by ignoring such prior data.

There is no consensus on how to optimally incorporate this historical data into online learning algorithms. The naive approach uses the full historical dataset to initialize reward estimates (Shivaswamy & Joachims, 2012), possibly incurring unnecessary and onerous computation and storage costs. These costs are particularly salient in continuous action settings with adaptive discretization, where the number of discretized regions is a direct function of the number of historical samples. If excessive data was collected on poor-performing actions, this *spurious data* with *imbalanced data coverage* would lead us to unnecessarily process and store an extremely large number of fine discretizations in low-performing areas of the action space, even when a significantly coarser discretization would be sufficient to inform us that region is not worth exploring. These two key challenges highlight that *the value of information of the historical dataset may not be a direct function of its size*. Real-world decision makers echo this sentiment: Martin et al. (2017) note that for conservation decisions, more information does not always translate into better actions; time is the resource which matters most.

A natural question one can ask is: *Is there an efficient way (in terms of space, computational, and sample complexity) to use historical data to achieve regret-optimal performance?* To this end, we propose ARTIFICIAL REPLAY, a meta-algorithm that modifies *any base bandit algorithm* to harness historical data, and show that our approach produces a regret-optimal policy while using the min-

imum number of historical samples. ARTIFICIAL REPLAY reduces computation and storage costs by only using historical data on an *as needed* basis. The key intuition is if we could *choose* which samples to include in the historical dataset, a natural approach would be to use a regret-optimal bandit algorithm to guide the sampling. ARTIFICIAL REPLAY builds on this intuition by using historical data as a replay buffer to artificially simulate online actions. Every time the base algorithm picks an action, we first check the historical data for any unused samples from the chosen action. If an unused sample exists, update the reward estimates and continue *without* advancing to the next timestep. Otherwise, sample from the environment, update the estimates using the observion, and continue to the next timestep. While this idea is easiest to understand in the context of the standard K-armed bandit problem, we discuss later how this framework naturally extends to other structure and information models, including continuous action spaces and semi-bandit feedback.

Although ARTIFICIAL REPLAY seems to be a natural heuristic to minimize use of historical data, it is not clear how to analyze its regret—specifically how much it loses compared to "full warm-start" (i.e., where the base algorithm is initialized with the full dataset). Surprisingly, however, we prove that under a widely applicable condition, the regret of ARTIFICIAL REPLAY (as a random variable) is distributionally identical to that of a full warm-start approach, while also guaranteeing significantly better time and storage complexity. Specifically, we show a *sample-path coupling*[1] between our ARTIFICIAL REPLAY approach and the full warm start approach with the same base algorithm, as long as the base algorithm satisfies a novel *independence of irrelevant data* (IIData) assumption.

Finally, we show the practical benefits of our method by instantiating ARTIFICIAL REPLAY for several broad classes of bandits and evaluating on real-world data. To highlight the breadth of algorithms that satisfy the IIData property, we provide examples of regret-optimal IIData polices for K-armed and continuous combinatorial bandits. We use these examples to prove that ARTIFICIAL REPLAY can lead to arbitrary better storage and computational complexity requirements. We close with a case study of combinatorial bandit algorithms for continuous resource allocation in the context of green security domains, using a novel adaptive discretization technique. Across the experiments, we observe concrete gains in storage and runtime using real-world poaching data from the ARTIFICIAL REPLAY framework over a range of base algorithms.

1.1 RELATED WORK

Multi-armed bandit problems have a long history in the online learning literature. We highlight the most closely related works below; for more extensive references please see our detailed discussion in Appendix B and Bubeck et al. (2012); Slivkins (2019); Lattimore & Szepesvári (2020).

Multi-Armed Bandit Algorithms. The design and analysis of bandit algorithms have been considered under a wide range of models. These algorithms were first studied in the K-armed bandit model in Lai & Robbins (1985), where the decision maker has access to a finite set of K possible actions at each timestep. However, numerous follow-up works have considered similar approaches when designing algorithms in continuous action spaces (Kleinberg et al., 2019) and with combinatorial constraints (Chen et al., 2013; Xu et al., 2021; Zuo & Joe-Wong, 2021). Our work provides a framework to modify existing algorithms to harness historical data. Moreover, we also propose a novel algorithm to incorporate adaptive discretization for combinatorial multi-armed bandits for continuous resource allocation, extending the discrete model from Zuo & Joe-Wong (2021).

Incorporating Historical Data. Several papers have started to investigate how to incorporate historical data into bandit algorithms, starting with Shivaswamy & Joachims (2012) who consider a K-armed bandit model where each arm has a dataset of historical pulls. The authors develop a "warm start" UCB algorithm to initialize the confidence bound of each arm based on the full historical data—prior to learning. Bouneffouf et al. (2019) extended similar techniques to models with pre-clustered arms. These techniques were later extended to Bayesian and frequentist linear contextual bandits, where the linear feature vector is initialized based on standard regression over the historical data (Oetomo et al., 2021; Wang et al., 2017). Our work provides a contrasting approach to harnessing historical data in algorithm design: our meta-algorithm can be applied to *any* standard bandit framework and uses the historical data only *as needed*, leading to improved computation and storage gains.

[1] That is, we construct the regret process for both algorithms simultaneously on a joint probability space such that each individual process has the correct marginal, but both processes are equal over each sample path.

2 PRELIMINARIES

We now define the general bandit model and specify the finite-armed and online combinatorial allocation settings that we study in our experiments. See Appendix C for details.

2.1 GENERAL STOCHASTIC BANDIT MODEL

We consider a stochastic bandit problem with a fixed action set $\mathcal{A}$. Let $\Re : \mathcal{A} \to \Delta([0,1])$ be a collection of independent and *unknown* reward distributions over $\mathcal{A}$. Our goal is to pick an action $a \in \mathcal{A}$ to maximize $\mathbb{E}[\Re(a)]$, the expected reward, which we denote $\mu(a)$. The optimal reward is:

$$\text{OPT} = \max_{a \in \mathcal{A}} \mu(a) \ . \tag{1}$$

For now, we do not impose any additional structure on $\mathcal{A}$, which could potentially be discrete, continuous, or encode combinatorial constraints.

Online Structure. Since the mean reward function $\mu(a)$ is initially unknown, we consider settings where the algorithm interacts with the environment sequentially over T timesteps. At timestep $t \in [T]$, the decision maker picks an action $A_t \in \mathcal{A}$ according to their policy π. The environment then reveals a reward R_t sampled from the distribution $\Re(A_t)$. The optimal reward OPT would be achieved using a policy with full knowledge of the true distribution. We thus define *regret* as:

$$\text{REGRET}(T, \pi) = T \cdot \text{OPT} - \textstyle\sum_{t=1}^{T} \mu(A_t) \ . \tag{2}$$

Historical Data. We assume that the algorithm designer has access to a historical dataset $\mathcal{H}^{hist} = \{a_j^{\mathcal{H}}, R_j^{\mathcal{H}}\}_{j \in [H]}$ containing H historical points with actions $\{a_j^{\mathcal{H}}\}_{j \in [H]}$ and rewards $R_j^{\mathcal{H}}$ sampled according to $\Re(a_j^{\mathcal{H}})$. We do not make any assumptions on how the historical actions $a_j^{\mathcal{H}}$ are chosen. Our goal is to efficiently incorporate this additional data to improve the performance of an algorithm.

2.2 FINITE, CONTINUOUS, AND COMBINATORIAL ACTION SPACES

Finite-Armed Bandit. The finite-armed bandit model can be viewed in this framework by considering K discrete actions $\mathcal{A} = [K] = \{1, \ldots, K\}$.

Combinatorial Multi-Armed Bandit for Continuous Resource Allocation (CMAB-CRA). A central planner has access to a metric space $\mathcal{S}$ of resources with metric $d_{\mathcal{S}}$. They are tasked with splitting a total amount of B divisible budget across N different resources within $\mathcal{S}$. An action consists of choosing N resources, i.e., N points in $\mathcal{S}$, and allocating the budget among that chosen subset. The feasible space of allocations is $\mathcal{B} = [0,1]$ and the feasible action space is:

$$\mathcal{A} = \left\{ (\vec{\mathbf{p}}, \vec{\beta}) \in \mathcal{S}^N \times \mathcal{B}^N \;\middle|\; \textstyle\sum_{i=1}^{N} \beta^{(i)} \leq B, \quad d_{\mathcal{S}}(\mathbf{p}^{(i)}, \mathbf{p}^{(j)}) \geq \epsilon \quad \forall i \neq j \right\} . \tag{3}$$

The chosen action must satisfy the budgetary constraint (i.e., $\sum_i \beta^{(i)} \leq B$), and the chosen resources must be distinct (aka ϵ-away from each other according to $d_{\mathcal{S}}$ for some $\epsilon > 0$). We additionally assume that $\Re$ decomposes independently over the (resource, allocation) pairs, in that $\mu(a) = \sum_{i=1}^{N} \mu(\mathbf{p}^{(i)}, \beta^{(i)})$. Lastly, we assume the algorithm observes semi-bandit feedback of the form $(\mathbf{p}_t^{(i)}, \beta_t^{(i)}, R_t^{(i)})_{i \in [N]}$ for each resource and allocation pair sampled according to $\Re(\mathbf{p}_t^{(i)}, \beta_t^{(i)})$.

Extension to Green Security. The CMAB-CRA model can be used to specify green security domains from Xu et al. (2021) by letting the space $\mathcal{S}$ represent a protected area and letting $\mathcal{B}$ represent the discrete set of patrol resources to allocate, such as number of ranger hours per week, with the total budget B being 40 hours. This formulation generalizes to a more realistic continuous space model of the landscape, instead of the artificial fixed discretization that was considered in prior work consisting of 1×1 sq. km regions of the park. In Section 5, we show that enabling patrol planning at a continuous level can help park rangers more precisely identify poaching hotspots.

3 ARTIFICIAL REPLAY FOR HARNESSING HISTORICAL DATA

We propose ARTIFICIAL REPLAY, a meta-algorithm that can be integrated with any base algorithm to incorporate historical data. We later prove that for any base algorithm satisfying *independence*

Algorithm 1 ARTIFICIAL REPLAY

Require: Historical dataset $\mathcal{H}^{hist} = \{(a_j^{\mathcal{H}}, R_j^{\mathcal{H}})\}_{j \in [H]}$, base algorithm Π, time horizon T

Initialize set of used historical data points $\mathcal{H}_1^{on} = \emptyset$
for $t = \{1, \ldots, T\}$ **do**
  Initialize `flag` to be `True`
  while `flag` is `True` **do**
    Pick action $\tilde{A}_t \sim \Pi(\mathcal{H}_t^{on} \cup \mathcal{H}_t)$
    if $\tilde{A}_t$ is not contained in $\mathcal{H}^{hist} \setminus \mathcal{H}_t^{on}$ **then**
      Update `flag` to be `False` ▷ Finish a full timestep
      Set online action $A_t = \tilde{A}_t$
      Execute action A_t and observe reward $R_t \sim \Re(A_t)$ ▷ Take online sample
      Update $\mathcal{H}_{t+1} = \mathcal{H}_t \cup \{(A_t, R_t)\}$ and $\mathcal{H}_{t+1}^{on} = \mathcal{H}_t^{on}$
    else
      Add to $\mathcal{H}_t^{on}$ one sample for $\tilde{A}_t$ from historical dataset $\mathcal{H}^{hist}$

of irrelevant data (IIData), a novel property we introduce, ARTIFICIAL REPLAY has identical regret to an approach which uses the full historical data upfront—showing that our approach reduces computation and storage costs without trading off performance.

Algorithm Formulation. Any algorithm for online stochastic bandits can be thought of as a function mapping arbitrary ordered histories (i.e., collections of observed (a, R) pairs) to a distribution over actions in $\mathcal{A}$. More specifically, let $\Pi : \mathcal{D} \to \Delta(\mathcal{A})$ be an arbitrary algorithm where $\mathcal{D}$ denotes the collection of possible histories (i.e., $\mathcal{D} = \cup_{i \geq 0}(\mathcal{A} \times \mathbb{R}_+)$). The policy obtained by a base algorithm Π *without incorporating historical data* simply takes the action sampled according to the policy $\pi_t^{\text{IGNORANT}(\Pi)} = \Pi(\mathcal{H}_t)$ where $\mathcal{H}_t$ is the data observed by timestep t. In comparison, consider an algorithm $\pi_t^{\text{FULL START}(\Pi)}$ which follows the same policy but uses the *full historical data upfront*, so takes the action sampled according to $\Pi(\mathcal{H}^{hist} \cup \mathcal{H}_t)$.

3.1 ARTIFICIAL REPLAY

The ARTIFICIAL REPLAY meta-algorithm incorporates the historical data $\mathcal{H}^{hist}$ into an arbitrary base algorithm Π, resulting in a policy we denote by $\pi^{\text{ARTIFICIAL REPLAY}(\Pi)}$. See Algorithm 1 for the pseudocode. We let $\mathcal{H}_t^{on}$ be the set of historical datapoints used by the start of time t. Initially, we have that $\mathcal{H}_1^{on} = \emptyset$. For an arbitrary timestep t, the ARTIFICIAL REPLAY approach works as follows:

Let $\tilde{A}_t \sim \Pi(\mathcal{H}_t^{on} \cup \mathcal{H}_t)$ be the *proposed action* at the start of time t. Since we are focused on *simulating* the algorithm with historical data, we break into cases whether or not the current set of unused historical datapoints (i.e., $\mathcal{H}^{hist} \setminus \mathcal{H}_t^{on}$) contains any additional information about $\tilde{A}_t$.

- **No historical data available**: If $\tilde{A}_t$ is not contained in $\mathcal{H}^{hist} \setminus \mathcal{H}_t^{on}$, then the *selected action* is $A_t = \tilde{A}_t$, and we advance to timestep $t + 1$. We additionally set $\mathcal{H}_{t+1}^{on} = \mathcal{H}_t^{on}$.
- **Historical data available**: If $\tilde{A}_t$ is contained in $\mathcal{H}^{hist} \setminus \mathcal{H}_t^{on}$, add that data point to $\mathcal{H}_t^{on}$ and repeat by picking another *proposed action*. We remain at time t.

Strikingly, our framework imposes minimal computational overhead on top of existing algorithms, simply requiring a data structure to verify whether $\tilde{A} \in \mathcal{H}^{hist} \setminus \mathcal{H}_t^{on}$, which can be implemented efficiently with hashing techniques. It is clear that the runtime and storage complexity of ARTIFICIAL REPLAY is no worse than FULL START. Additionally, this approach extends naturally to the following different models of the action set and feedback structure:

Continuous Spaces. The ARTIFICIAL REPLAY framework can be applied in continuous action spaces with discretization-based algorithms. For example, suppose that Π wants to select an action $a \in \mathcal{A}$, but the historical data has a sample from $a + \epsilon$ (i.e., a slightly perturbed point). Discretization-based algorithms avoid these precision issues since they map the continuous space to a series of regions which together cover the action set, and run algorithms or subroutines over the discretization. Checking for historical data simply checks for data within the bounds of the chosen discretized action.

Semi-Bandit Feedback. ARTIFICIAL REPLAY also naturally extends to combinatorial action sets with semi-bandit feedback where actions are *decomposable*, that is, they can be written as $a = (a_1, \ldots, a_N)$ with independent rewards. Suppose that Π wants to select an action $a = (a_1, a_2, \ldots, a_N)$ but the historical data has a sample from $(a'_1, a_2, \ldots a'_N)$. Even if the combinatorial action a does not appear in its entirety in the historical data, as long as there exists some subcomponent $a_i^{\mathcal{H}}$ (sometimes referred to as "subarm" in combinatorial bandits) in the historical data (e.g., a_2) , we can add that subcomponent $a_i^{\mathcal{H}}$ to $\mathcal{H}_t^{on}$ to update the base algorithm and sample another action.

3.2 INDEPENDENCE OF IRRELEVANT DATA AND REGRET COUPLING

It is not immediately clear how to analyze the regret of ARTIFICIAL REPLAY. To enable regret analysis, we introduce a new property for bandit algorithms, *independence of irrelevant data*, which essentially requires that when an algorithm is about to take an action, providing additional data about other actions (i.e., those not selected by the algorithm) will not influence the algorithm's decision.

Definition 3.1 (Independence of irrelevant data)**.** *A deterministic base algorithm Π satisfies the* **independence of irrelevant data** *(IIData) property if whenever $A = \Pi(\mathcal{H})$ then*

$$\Pi(\mathcal{H}) = \Pi(\mathcal{H} \cup \mathcal{H}') \tag{4}$$

for any $\mathcal{H}'$ containing data from any actions a' other than A (that is, $a' \neq A$).

IIData is a natural robustness property for an algorithm to satisfy, highlighting that the algorithm evaluates actions independently when making decisions. IIData is conceptually analogous to the independence of irrelevant alternatives (IIA) axiom in computational social choice as a desiderata used to evaluate voting rules (Arrow, 1951). In Theorem 3.2 we show that for any base algorithm satisfying IIData, the regret of $\pi^{\text{FULL START}(\Pi)}$ and $\pi^{\text{ARTIFICIAL REPLAY}(\Pi)}$ will be equal.

Theorem 3.2. *Suppose that algorithm Π satisfies the independence of irrelevant data property. Then for any problem instance, horizon T, and historical dataset $\mathcal{H}^{hist}$ we have the following:*

$$\pi_t^{\text{ARTIFICIAL REPLAY}(\Pi)} \overset{d}{=} \pi_t^{\text{FULL START}(\Pi)}$$

$$\text{REGRET}(T, \pi^{\text{ARTIFICIAL REPLAY}(\Pi)}) \overset{d}{=} \text{REGRET}(T, \pi^{\text{FULL START}(\Pi)}) \ .$$

This theorem shows that ARTIFICIAL REPLAY allows us to achieve identical regret guarantees as FULL START while simultaneously using data more efficiently. However, we note that IIData itself is a weak property: naive deterministic algorithms which always output the same action (hence does not adapt to the data at all) satisfies IIData, but clearly are not regret-optimal. The critical question is whether there exist regret-optimal algorithms that also satisfy IIData. In the subsequent section, we show three example regret-optimal algorithms which satisfy this property, even in the complex CMAB-CRA setting. In the existing bandit literature, there has been a narrow focus on only finding regret-optimal algorithms. We propose that IIData is another desirable property that implies ease and robustness for optimally and efficiently incorporating historical data.

4 IIDATA ALGORITHMS

In this section, we provide regret-optimal IIData algorithms for two settings: the K-armed and CMAB-CRA models. For more details and proofs, see Appendices D and F.

4.1 K-ARMED BANDITS

The first algorithm we propose, named Monotone UCB (denoted as MONUCB), is derived from the UCB1 algorithm introduced in Auer et al. (2002). At every timestep t, the algorithm tracks the following: *(i)* $\overline{\mu}_t(a)$ for the estimated mean reward of action $a \in [K]$, *(ii)* $n_t(a)$ for the number of times the action a has been selected by the algorithm prior to timestep t, and *(iii)* $\text{UCB}_t(a)$ for an upper confidence bound estimate for the reward of action a. At every timestep t, the algorithm picks the action A_t which maximizes $\text{UCB}_t(a)$ (breaking ties deterministically). After observing R_t, we increment $n_{t+1}(A_t) = n_t(A_t) + 1$, update $\overline{\mu}_{t+1}(A_t)$, and set:

$$\text{UCB}_{t+1}(A_t) = \min\left\{\text{UCB}_t(A_t),\ \ \overline{\mu}_{t+1}(A_t) + \sqrt{\tfrac{2\log(T)}{n_{t+1}(A_t)}}\right\} . \tag{5}$$

Algorithm 2 Monotone UCB (MONUCB)

1: Initialize $n_1(a) = 0$, $\overline{\mu}_1(a) = 1$, and $\text{UCB}_1(a) = 1$ for each $a \in [K]$
2: **for** $t = \{1, \ldots, T\}$ **do**
3: Let $A_t = \arg\max_{a \in [K]} \text{UCB}_t(a)$
4: Receive reward R_t sampled from $\Re(A_t)$
5: Update $n_{t+1}(A_t) = n_t(A_t) + 1$, $n_{t+1}(a) = n_t(a)$ for $a \neq A_t$
6: Update $\overline{\mu}_{t+1}(A_t) = (n_t(A_t)\overline{\mu}_t(A_t) + R_t)/n_{t+1}(A_t)$, $\overline{\mu}_{t+1}(a) = \overline{\mu}_t(a)$ for $a \neq A_t$
7: Update $\text{UCB}_{t+1}(a) = \text{UCB}_t(a)$ for $a \neq A_t$ and
$\text{UCB}_{t+1}(A_t) = \min\{\text{UCB}_t(A_t), \overline{\mu}_{t+1}(A_t) + \sqrt{2\log(T)/n_{t+1}(A_t)}\}$

We note that the only modification of Monotone UCB from UCB1 is the additional step forcing the UCB estimates to be monotone decreasing over t. It is clear that this modification has no affect on the regret guarantees. Under the "good event" analysis, if $\text{UCB}_t(a) \geq \mu(a)$ with high probability, then the condition still holds at time $t+1$, even after observing a new data point. In the following theorem, we show that MONUCB satisfies IIData and is regret-optimal, acheiving the same instance-dependent regret bound as the standard UCB1 algorithm.

Theorem 4.1. *Monotone UCB satisfies the IIData property and has:*

$$\text{REGRET}(T, \pi^{\text{IGNORANT(MONUCB)}}) = O(\sum\nolimits_a \log(T)/\Delta(a)) \textit{ where } \Delta(a) = \max_{a'} \mu(a') - \mu(a). \quad (6)$$

This guarantee allows us to use Theorem 3.2 to establish that ARTIFICIAL REPLAY and FULL START have identical regret with MONUCB as a base algorithm. In the next theorem, we show that ARTIFICIAL REPLAY is robust to *spurious data*, where the historical data has excess samples $a_j^{\mathcal{H}}$ coming from poor performing actions. Spurious data imposes computational and storage challenges, since the FULL START approach will pre-process the full historical dataset regardless of the observed rewards or the inherent value of the historical data. In contrast, ARTIFICIAL REPLAY will only use the amount of data useful for learning.

Theorem 4.2. *For every $H \in \mathbb{N}$ there exists a historical dataset $\mathcal{H}^{hist}$ with $|\mathcal{H}^{hist}| = H$ where the runtime of $\pi^{\text{FULL START(MONUCB)}} = \Omega(H + T)$ whereas the runtime of $\pi^{\text{ARTIFICIAL REPLAY(MONUCB)}} = O(T + \log(T))$.*

Lastly, to complement the computational and storage improvements of ARTIFICIAL REPLAY applied to MONUCB, we can also show an improvement of regret. This analysis crucially uses the regret coupling, since FULL START(MONUCB) is much easier to reason about than its ARTIFICIAL REPLAY counterpart.

Theorem 4.3. *Let H_a be the number of datapoints in $\mathcal{H}^{hist}$ for each action $a \in [K]$. Then the regret of Monotone UCB with historical dataset $\mathcal{H}^{hist}$ is:*

$$\text{REGRET}(T, \pi^{\text{ARTIFICIAL REPLAY(MONUCB)}}) \leq O\Big(\sum_{a \in [K]: \Delta_a \neq 0} \max\big\{0, \tfrac{\log(T)}{\Delta(a)} - H_a \Delta(a)\big\}\Big).$$

Theorem 4.2 together with Theorem 4.3 helps highlight the advantage of using ARTIFICIAL REPLAY over FULL START in terms of improving computational complexity while maintaining an equally improved regret guarantee.

We close with an example of a K-armed bandit algorithm which does not satisfy the IIData assumption. Thompson Sampling (Russo et al., 2018), which samples arms according to the posterior probability that they are optimal, does not satisfy IIData. Data from other actions other than the one chosen will adjust the posterior distribution, and hence will adjust the selection probabilities as well. While we do not obtain a regret coupling, in Fig. 8 (appendix) we show that there are still empirical gains for using ARTIFICIAL REPLAY over FULL START across a variety of base algorithms.

4.2 CMAB-CRA

Incorporating historical data optimally and efficiently is difficult in continuous action settings. Two natural approaches are to *(i)* discretize the action space $\mathcal{A}$ based on the data using nearest neighbor

estimates, or *(ii)* learn a regression of the mean reward using available data. Consider a setting where excessive data is collected from poor-performing actions. Discretization-based algorithms will unnecessarily process and store a large number of discretizations in low-performing regions of the space. Regression-based methods will use compute resources to learn an accurate predictor of the mean reward in irrelevant regions. The key issues are that the computational and storage cost grows with the size of the historical dataset, and the estimation and discretization is done independent of the quality of the reward.

To contrast this approach, we present two discretization-based algorithms that satisfy IIData with strong performance guarantees. In particular, we detail fixed and adaptive discretization algorithms that only use the historical dataset to update estimates of the reward. Due to space, we describe the algorithm only at a high level, but see Appendix D for more details.

Our algorithms are Upper Confidence Bound (UCB) style as the selection rule maximizes Eq. (1) over the combinatorial action set (Eq. (3)) through a discretization of $\mathcal{S}$. For each allocation $\beta \in \mathcal{B}$, the algorithm maintains a collection of regions $\mathcal{P}_t^\beta$ of $\mathcal{S}$ which covers $\mathcal{S}$. For the fixed discretization variant, $\mathcal{P}_t^\beta$ is fixed at the start of learning, and in the adaptive discretization version it is refined over the course of learning based on observed data. At every timestep t and region $\mathcal{R} \in \mathcal{P}_t^\beta$, the algorithm tracks the following: *(i)* $\overline{\mu}_t(\mathcal{R}, \beta)$ for the estimated mean reward of region $\mathcal{R}$ at allocation β, *(ii)* $n_t(\mathcal{R}, \beta)$ for the number of times $\mathcal{R}$ has been selected at allocation β prior to timestep t, and *(iii)* $\text{UCB}_t(\mathcal{R}, \beta)$ for an upper confidence bound estimate. At a high level, our algorithm performs three steps in each iteration t:

1. **Action selection**: Greedily select at most N regions in $\mathcal{P}_t^\beta$ to maximize $\text{UCB}_t(\mathcal{R}, \beta)$ subject to the budget constraints (see Eq. (10) in the appendix). Note that we must additionally ensure that each region is selected at only a *single* allocation value.
2. **Update parameters**: For each of the selected regions, increment $n_t(\mathcal{R}, \beta)$ by one, update $\overline{\mu}_t(\mathcal{R}, \beta)$ based on observed data, and set $\text{UCB}_{t+1}(\mathcal{R}, \beta) = \min\{\text{UCB}_t(\mathcal{R}, \beta), \overline{\mu}_t(\mathcal{R}, \beta) + b(n_t(\mathcal{R}, \beta))\}$ for some appropriate bonus term $b(\cdot)$. This enforces monotonicity in the UCB estimates similar to MONUCB and is required for the IIData property.
3. **Re-partition**: This step differentiates the *adaptive discretization* algorithm from fixed discretization, which maintains the same partition across all timesteps. We split a region when the confidence in its estimate (i.e., $b(n_t(\mathcal{R}, \beta))$) is smaller than the diameter of the region. This condition may seem independent of the *quality* of a region, but since it is incorporated into a learning algorithm, the number of samples in a region is correlated with its reward. In Fig. 4 (appendix) we highlight how the adaptive discretization algorithm hones in on regions with large reward without knowing the reward function before learning.

These algorithms modify existing approaches applied to CMAB-CRA in the bandit and reinforcement learning literature, which have been shown to be regret-optimal (Xu et al., 2021; Sinclair et al., 2021). We additionally note that these approaches are IIData.

Theorem 4.4. *The fixed and adaptive discretization algorithms when using a greedy solution to solve Eq. (1) have property IIData.*

Requiring the algorithm to use a greedy approximation solution introduces limitations. However, under additional assumptions on the mean reward function $\mu(\mathbf{p}, \beta)$, the greedy solution is provably optimal. For example, optimality of the greedy approximation holds when $\mu(\mathbf{p}, \beta)$ is piecewise linear and monotone, or more broadly when $\mu(a)$ is submodular.

5 EXPERIMENTS

We show the benefits of ARTIFICIAL REPLAY by showing that our meta-algorithm achieves identical performance to FULL START while offering significant practical advantages in reducing runtime and storage. We consider two classes of bandit domains: K-armed and CMAB-CRA. As part of our evaluation on combinatorial bandits, we introduce a new model for green security games with continuous actions by adaptively discretizing the landscape of a large protected area in East Africa.

All of the code to reproduce the experiments is available at `[link redacted]`. Results are averaged over 60 iterations with random seeds, with standard error plotted; experiment details and additional results are available in Appendix E.

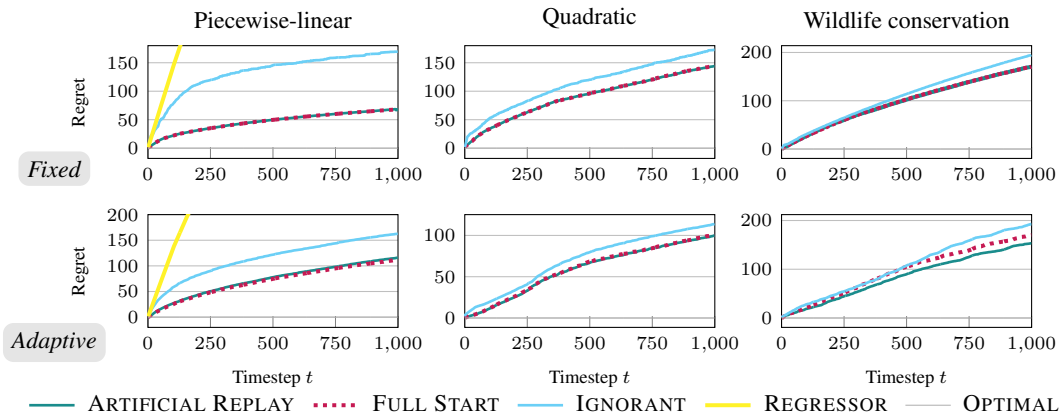

Figure 1: *(CMAB-CRA)* Cumulative regret (y-axis; lower is better) across time $t \in [T]$. ARTIFICIAL REPLAY performs equally as well as FULL START across all domain settings, including both fixed discretization (top row) and adaptive discretization (bottom). The REGRESSOR performs quite poorly.

Domains. We conduct experiments on the two bandit models described in Section 2.2: finite K-armed bandits and CMAB-CRA, using both fixed and adaptive discretization. For the continuous combinatorial setting, we provide two stylized domains: a piecewise-linear and a quadratic reward function. To emphasize the practical benefit of ARTIFICIAL REPLAY, we evaluate on a real-world resource allocation setting for biodiversity conservation. We study real ranger patrol data from a wildlife park in East Africa[2], shared as part of a collaboration with a national conservation non-profit. We use historical patrol observations to build the history $\mathcal{H}^{hist}$; we analyze these historical observations in detail in Appendix E to show that this dataset exhibits both spurious data and imbalanced coverage as discussed in Section 4.

Baselines. We compare ARTIFICIAL REPLAY against IGNORANT and FULL START approaches for each setting. In the K-armed model, we use MONUCB as the base algorithm. In CMAB-CRA we use fixed and adaptive discretization as well as REGRESSOR, a neural network learner that is a regression-based approach analogue to FULL START. REGRESSOR is initially trained on the entire historical dataset, then iteratively retrained after 128 new samples are collected. We also compute for each setting the performance of an OPTIMAL action based on the true rewards and a RANDOM baseline that acts randomly while satisfying the budget constraint.

Results. The results in Fig. 1 empirically validate our theoretical result from Theorem 3.2: the performance of ARTIFICIAL REPLAY is identical to that of FULL START, and reduces regret considerably compared to the naive IGNORANT approach. We evaluate the regret (compared to OPTIMAL) of each approach across time $t \in [T]$. Concretely, we consider the three domains of piecewise-linear reward, quadratic reward, and green security with continuous space $\mathcal{S} = [0, 1]^2$, $N = 5$ possible action components, a budget $B = 2$, and 3 levels of effort. We include $H = 300$ historical data points. See Fig. 9 (appendix) for regret and analysis of historical data use on the K-armed bandit.

Not only does ARTIFICIAL REPLAY achieve equal performance, but its computational benefits over FULL START are clear even on practical problem sizes. As we increase historical data from $H = \{10; 100; 1{,}000; 10{,}000\}$ in Fig. 2, the proportion of irrelevant data increases. Our method achieves equal performance, overcoming the previously unresolved challenge of *spurious data*, while FULL START suffers from arbitrarily worse storage complexity (Theorem 4.2). With 10,000 historical samples and a time horizon of 1,000, we see that 58.2% of historical samples are irrelevant to producing the most effective policy.

When faced with *imbalanced data coverage*, the benefits of ARTIFICIAL REPLAY become clear—most notably in the continuous action setting with adaptive discretization. In Fig. 3, as we increase

[2]Wildlife park details redacted for anonymous submission.

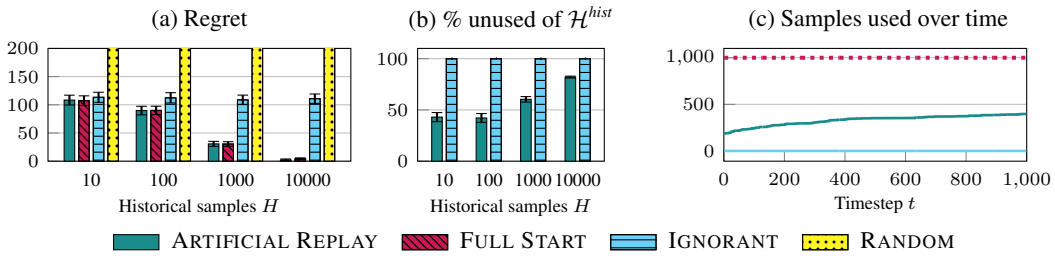

Figure 2: *(K-Armed)* Increasing the number of historical samples H leads FULL START to use unnecessary data, particularly as H gets very large. ARTIFICIAL REPLAY achieves equal performance in terms of regret (plot a) while using less than half the historical data (plot b). In plot c we see that with $H = 1{,}000$ historical samples, ARTIFICIAL REPLAY uses (on average) 117 historical samples before taking its first online action. The number of historical samples used increases at a decreasing rate, using only 396 of 1,000 total samples by the horizon T. Results are shown on the K-armed bandit setting with $K = 10$ and horizon $T = 1{,}000$.

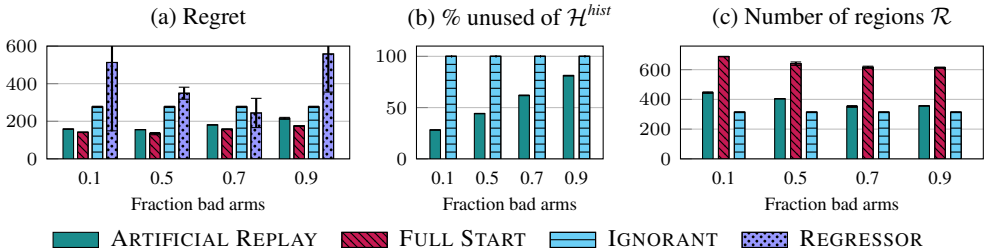

Figure 3: *(CMAB-CRA)* Holding $H = 10{,}000$ constant, we increase the fraction of historical data samples on bad arms (bottom 20% of rewards). The plots show (a) regret, (b) % of unused historical data and (c) number of discretized regions in partition $\mathcal{P}$. ARTIFICIAL REPLAY enables significantly improved runtime and reduced storage while matching the performance of FULL START. Results on the CMAB-CRA setting with adaptive discretization on the quadratic domain.

the number of historical samples on bad regions (bottom 20th percentile of reward), the additional data require finer discretization, leading to arbitrarily worse storage and computational complexity for FULL START with equal regret. In Fig. 3(c), we see that with 10% of data on bad arms, ARTIFICIAL REPLAY requires only 446 regions $\mathcal{R}$ compared to 688 used by FULL START; as we get more spurious data and that fraction increases to 90%, then ARTIFICIAL REPLAY requires only 356 regions while FULL START still stores 614 regions.

6 CONCLUSION

We present ARTIFICIAL REPLAY, a meta-algorithm that modifies *any base bandit algorithm* to efficiently harness historical data. We show that under a widely applicable IIData condition, the regret of ARTIFICIAL REPLAY (as a random variable) is distributionally identical to that of a full warm-start approach, while also guaranteeing significantly better time and storage complexity. We additionally give examples of regret-optimal IIData algorithms in the K-armed and CMAB-CRA settings. Our experimental results highlight the advantage of using ARTIFICIAL REPLAY over FULL START via a variety of base algorithms, applied to K-armed and continuous combinatorial bandit models. Directions for future work include *(i)* find IIData algorithms in other bandit domains such as linear contextual bandits, *(ii)* incorporate the ARTIFICIAL REPLAY approach into reinforcement learning, and *(iii)* provide theoretical bounds showing that ARTIFICIAL REPLAY has *optimal* data usage when incorporating historical data.

REFERENCES

Kenneth J Arrow. *Social choice and individual values*. John Wiley & Sons, 1951.

Peter Auer, Nicolo Cesa-Bianchi, and Paul Fischer. Finite-time analysis of the multi-armed bandit problem. *Machine learning*, 47(2):235–256, 2002.

Hamsa Bastani, Kimon Drakopoulos, Vishal Gupta, Ioannis Vlachogiannis, Christos Hadjicristodoulou, Pagona Lagiou, Gkikas Magiorkinis, Dimitrios Paraskevis, and Sotirios Tsiodras. Efficient and targeted COVID-19 border testing via reinforcement learning. *Nature*, 599(7883): 108–113, 2021.

Djallel Bouneffouf, Srinivasan Parthasarathy, Horst Samulowitz, and Martin Wistuba. Optimal exploitation of clustering and history information in multi-armed bandit. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, (IJCAI-19)*, pp. 2016–2022, 2019.

Sébastien Bubeck, Nicolo Cesa-Bianchi, et al. Regret analysis of stochastic and nonstochastic multi-armed bandit problems. *Foundations and Trends in Machine Learning*, 5(1):1–122, 2012.

Stefano Canessa, Gurutzeta Guillera-Arroita, José J Lahoz-Monfort, Darren M Southwell, Doug P Armstrong, Iadine Chadès, Robert C Lacy, and Sarah J Converse. When do we need more data? a primer on calculating the value of information for applied ecologists. *Methods in Ecology and Evolution*, 6(10):1219–1228, 2015.

Wei Chen, Yajun Wang, and Yang Yuan. Combinatorial multi-armed bandit: General framework and applications. In *Proceedings of the Thirtieth International Conference on Machine Learning (ICML-13)*, volume 28, pp. 151–159. PMLR, 2013.

Wei Chu, Lihong Li, Lev Reyzin, and Robert Schapire. Contextual bandits with linear payoff functions. In *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics (AISTATS-11)*, pp. 208–214, 2011.

Adam N Elmachtoub, Ryan McNellis, Sechan Oh, and Marek Petrik. A practical method for solving contextual bandit problems using decision trees. In *Proceedings of the Thirty-Third Conference on Uncertainty in Artificial Intelligence (UAI-17)*, 2017.

Fei Fang, Peter Stone, and Milind Tambe. When security games go green: Designing defender strategies to prevent poaching and illegal fishing. In *Proceedings of the Twenty-fourth International Joint Conference on Artificial Intelligence (IJCAI-15)*, 2015.

Shahrzad Gholami, Amulya Yadav, Long Tran-Thanh, Bistra Dilkina, and Milind Tambe. Don't put all your strategies in one basket: Playing green security games with imperfect prior knowledge. In *Proceedings of the 18th International Conference on Autonomous Agents and Multiagent Systems (AAMAS-19)*, pp. 395–403, 2019.

Debarun Kar, Benjamin Ford, Shahrzad Gholami, Fei Fang, Andrew Plumptre, Milind Tambe, Margaret Driciru, Fred Wanyama, Aggrey Rwetsiba, Mustapha Nsubaga, et al. Cloudy with a chance of poaching: Adversary behavior modeling and forecasting with real-world poaching data. In *Proceedings of the 16th International Conference on Autonomous Agents and Multiagent Systems (AAMAS-17)*, 2017.

Robert Kleinberg and Nicole Immorlica. Recharging bandits. In *2018 IEEE 59th Annual Symposium on Foundations of Computer Science (FOCS-18)*, pp. 309–319. IEEE, 2018.

Robert Kleinberg, Alexandru Niculescu-Mizil, and Yogeshwer Sharma. Regret bounds for sleeping experts and bandits. *Machine Learning*, 80(2):245–272, 2010.

Robert Kleinberg, Aleksandrs Slivkins, and Eli Upfal. Bandits and experts in metric spaces. *Journal of the ACM (JACM)*, 66(4):1–77, 2019.

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.

Tara G Martin, Abbey E Camaclang, Hugh P Possingham, Lynn A Maguire, and Iadine Chadès. Timing of protection of critical habitat matters. *Conservation Letters*, 10(3):308–316, 2017.

Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Alex Graves, Ioannis Antonoglou, Daan Wierstra, and Martin Riedmiller. Playing Atari with deep reinforcement learning. In *Proceedings of the Twenty-Seventh Conference on Neural Information Processing Systems (NeurIPS-13)*, 2013.

Thanh H Nguyen, Arunesh Sinha, Shahrzad Gholami, Andrew Plumptre, Lucas Joppa, Milind Tambe, Margaret Driciru, Fred Wanyama, Aggrey Rwetsiba, and Rob Critchlow. CAPTURE: A new predictive anti-poaching tool for wildlife protection. In *Proceedings of the 2016 International Conference on Autonomous Agents & Multiagent Systems (AAMAS-16)*, 2016.

Bastian Oetomo, R Malinga Perera, Renata Borovica-Gajic, and Benjamin IP Rubinstein. Cutting to the chase with warm-start contextual bandits. In *2021 IEEE International Conference on Data Mining (ICDM-21)*, pp. 459–468. IEEE, 2021.

Andrew J Plumptre, Richard A Fuller, Aggrey Rwetsiba, Fredrick Wanyama, Deo Kujirakwinja, Margaret Driciru, Grace Nangendo, James EM Watson, and Hugh P Possingham. Efficiently targeting resources to deter illegal activities in protected areas. *Journal of Applied Ecology*, 51 (3):714–725, 2014.

Larry D Pyeatt and Adele E Howe. Decision tree function approximation in reinforcement learning. In *Proceedings of the Third International Symposium on Adaptive Systems: Evolutionary Computation and Probabilistic Graphical Models*, volume 2, pp. 70–77, 2001.

Yundi Qian, Chao Zhang, Bhaskar Krishnamachari, and Milind Tambe. Restless poachers: Handling exploration-exploitation tradeoffs in security domains. In *Proceedings of the 2016 International Conference on Autonomous Agents & Multiagent Systems (AAMAS-16)*, pp. 123–131, 2016.

Daniel Russo and Benjamin Van Roy. Learning to optimize via information-directed sampling. *Operations Research*, 66(1):230–252, 2018.

Daniel J Russo, Benjamin Van Roy, Abbas Kazerouni, Ian Osband, Zheng Wen, et al. A tutorial on Thompson sampling. *Foundations and Trends in Machine Learning*, 11(1):1–96, 2018.

Tom Schaul, John Quan, Ioannis Antonoglou, and David Silver. Prioritized experience replay. In *Proceedings of the Fourth International Conference on Learning Representations (ICLR-16)*, 2017.

Pannagadatta Shivaswamy and Thorsten Joachims. Multi-armed bandit problems with history. In *Proceedings of the Fifteenth International Conference on Artificial Intelligence and Statistics (AISTATS-12)*, pp. 1046–1054. PMLR, 2012.

Sean R Sinclair, Siddhartha Banerjee, and Christina Lee Yu. Adaptive discretization in online reinforcement learning. *arXiv preprint arXiv:2110.15843*, 2021.

Aleksandrs Slivkins. Contextual bandits with similarity information. In *Proceedings of the Twenty-Fourth Annual Conference On Learning Theory (COLT-11)*, pp. 679–702, 2011.

Aleksandrs Slivkins. Introduction to multi-armed bandits. *Foundations and Trends in Machine Learning*, 12(1-2):1–286, 2019. ISSN 1935-8237.

Adith Swaminathan and Thorsten Joachims. Batch learning from logged bandit feedback through counterfactual risk minimization. *Journal of Machine Learning Research*, 16(52):1731–1755, 2015.

William TB Uther and Manuela M Veloso. Tree based discretization for continuous state space reinforcement learning. In *Proceedings of the Fifteenth National Conference on Artificial Intelligence (AAAI-98)*, 1998.

Lu Wang, Chengyu Wang, Keqiang Wang, and Xiaofeng He. BiUCB: A contextual bandit algorithm for cold-start and diversified recommendation. In *2017 IEEE International Conference on Big Knowledge (ICBK-17)*, pp. 248–253, 2017.

Lily Xu, Shahrzad Gholami, Sara Mc Carthy, Bistra Dilkina, Andrew Plumptre, Milind Tambe, Rohit Singh, Mustapha Nsubuga, Joshua Mabonga, Margaret Driciru, et al. Stay ahead of poachers: Illegal wildlife poaching prediction and patrol planning under uncertainty with field test evaluations. In *Proceedings of the IEEE 36th International Conference on Data Engineering (ICDE-20)*, 2020.

Lily Xu, Elizabeth Bondi, Fei Fang, Andrew Perrault, Kai Wang, and Milind Tambe. Dual-mandate patrols: Multi-armed bandits for green security. In *Proceedings of the 35th AAAI Conference on Artificial Intelligence (AAAI-21)*, 2021.

Rong Yang, Benjamin J Ford, Milind Tambe, and Andrew Lemieux. Adaptive resource allocation for wildlife protection against illegal poachers. In *Proceedings of the 13th International Conference on Autonomous Agents and Multiagent Systems (AAMAS-14)*, pp. 453–460, 2014.

Chicheng Zhang, Alekh Agarwal, Hal Daumé Iii, John Langford, and Sahand Negahban. Warm-starting contextual bandits: Robustly combining supervised and bandit feedback. In *Proceedings of the 36th International Conference on Machine Learning (ICML-19)*, volume 97, pp. 7335–7344, 2019.

Jinhang Zuo and Carlee Joe-Wong. Combinatorial multi-armed bandits for resource allocation. In *55th Annual Conference on Information Sciences and Systems (CISS-21)*, pp. 1–4. IEEE, 2021.

Jinhang Zuo, Xiaoxi Zhang, and Carlee Joe-Wong. Observe before play: Multi-armed bandit with pre-observations. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI-20)*, volume 34, pp. 7023–7030, 2020.