Contextual Dueling Bandits

Miroslav Dudík
Microsoft Research, New York, NY USA
MDUDIK@MICROSOFT.COM

Katja Hofmann
Microsoft Research, Cambridge, United Kingdom
KATJA.HOFMANN@MICROSOFT.COM

Robert E. Schapire
Microsoft Research, New York, NY USA
SCHAPIRE@MICROSOFT.COM

Aleksandrs Slivkins
Microsoft Research, New York, NY USA
SLIVKINS@MICROSOFT.COM

Masrour Zoghi
University of Amsterdam, Amsterdam, The Netherlands
M.ZOGHI@UVA.NL

Abstract
We consider the problem of learning to choose actions using contextual information when provided with limited feedback in the form of relative pairwise comparisons. We study this problem in the dueling-bandits framework of Yue et al. (2009), which we extend to incorporate context. Roughly, the learner’s goal is to find the best policy, or way of behaving, in some space of policies, although “best” is not always so clearly defined. Here, we propose a new and natural solution concept, rooted in game theory, called a von Neumann winner, a randomized policy that beats or ties every other policy. We show that this notion overcomes important limitations of existing solutions, particularly the Condorcet winner which has typically been used in the past, but which requires strong and often unrealistic assumptions. We then present three efficient algorithms for online learning in our setting, and for approximating a von Neumann winner from batch-like data. The first of these algorithms achieves particularly low regret, even when data is adversarial, although its time and space requirements are linear in the size of the policy space. The other two algorithms require time and space only logarithmic in the size of the policy space when provided access to an oracle for solving classification problems on the space.

Keywords: contextual dueling bandits, online learning, bandit algorithms, game theory.

1. Introduction
We study how to learn to act based on contextual information when provided only with partial, relative feedback. This problem naturally arises in fields such as information retrieval (IR) and recommender systems, where it has been shown that user feedback is considerably more reliable when interpreted as relative comparisons rather than absolute labels (Radlinski et al., 2008). For instance, in web search, for a particular query, the IR system may have several candidate rankings of documents that could be presented, with the best option being dependent upon the specific user. By presenting a mix or interleaving of two of the candidate rankings and observing the user’s response (Chapelle et al., 2012; Hofmann et al., 2013), it is possible for such a system to get feedback about
user preferences. However, this feedback is partial since it is only with respect to the two rankings that were chosen, and it is relative since it only tells which of the two rankings is preferred to the other.

The *dueling-bandits problem* of Yue et al. (2009) provides an approach to this setting. Abstractly, the learner is repeatedly faced with a set of possible actions, and may select two of these actions to face off in a duel whose stochastically determined winner is then revealed. Through such experimentation, the learner attempts to find the “best” of the actions. We focus on the *contextual dueling bandit* setting, where context can provide information that helps identify the best action. For instance, in the example above, the actions may be the candidate rankings to choose among, and the context may be additional information about the user or query that might help in choosing the best ranking. The learner’s goal now is to find a good *policy*, a rule for choosing actions based on context.

Similar to prior work on contextual (non-dueling) bandits (Auer et al., 2002; Langford and Zhang, 2007; Dudík et al., 2011), we propose a setting in which the learner has access to a space of policies $\Pi$, with the goal of performing as well as the “best” in the space. This space plays a role analogous to the hypothesis space in supervised learning. It will typically be extremely large or even infinite. We therefore explicitly aim for methods that will be applicable when this is the case.

Merely defining the precise goal of learning can be problematic in such a relative-feedback setting. When rewards are absolute, the best policy in $\Pi$ is clearly and easily defined as the one that achieves the highest expected reward since, by such an absolute measure, this policy beats every other policy. In a relative-feedback setting, since we have a means of obtaining pairwise comparisons between actions or policies, we might aim to choose the policy in $\Pi$ that (on average) beats every other policy in the class in such head-to-head competitions. Most previous work on dueling bandits (Yue et al., 2009; Yue and Joachims, 2011; Urvoy et al., 2013; Zoghi et al., 2014b) has in fact explicitly or implicitly assumed that such a Condorcet winner exists. But there are good reasons to doubt such a strong assumption, particularly when working with large and rich policy spaces. There are numerous examples, even in natural situations, where this assumption (and more generally, transitivity among policies) are known to fail (see (Gardner, 1970; Zoghi et al., 2014a), for instance). Indeed, the preferences of a population of users do not need to be transitive, even if each individual user has transitive preferences.

In this paper, we seek to improve the dueling bandits techniques in two respects. First, we seek to relax the modeling restrictions on which previous methods have depended so as to develop methods that are more generally applicable. Second, we seek to achieve a similar level of flexibility in the design of policies as for supervised learning algorithms.

**Contributions.** Our first contribution (in Section 3) is the introduction of a new solution concept, called the von Neumann winner, which is based on a game-theoretic interpretation. Like a Condorcet winner, when facing any other policy in a duel, a von Neumann winner has at least a 50% chance of winning; in this sense, a von Neumann winner is at least as good as every policy in the space. On the other hand, a von Neumann winner is always guaranteed to exist, without any extraneous assumptions. This guarantee is made possible by allowing policies to be selected in a randomized fashion, as is quite natural in such a learning setting.

With the goal of learning clarified, we turn to algorithms. As a warm-up, in Section 5, we give a fully online algorithm in which two copies of the Exp4.P multi-armed bandit algorithm (Beygelzimer et al., 2011) are run against one another (using a “sparring” approach previously suggested
by Ailon et al. (2014)). Although yielding good regret, this algorithm requires time and space linear in $|\Pi|$, which is impractical in most realistic settings where we would expect $\Pi$ to be enormous.

To address this difficulty, we propose an approach used previously in other works on contextual bandits (Langford and Zhang, 2007; Dudík et al., 2011). Specifically, we assume that we have access to a classification oracle for our policy class that can find the minimum-cost policy in $\Pi$ when given the cost of each action on each of a sequence of contexts. In fact, an ordinary cost-sensitive, multiclass classification learning algorithm can be used for this purpose, which suggests that, practically, this may be a reasonable and natural assumption.

We then consider techniques for constructing a von Neumann winner from empirical exploration data. (Although we focus on a batch-like setting, the resulting algorithm can be used online as well.) We analyze the statistical efficiency of this approach in Section 6. In Sections 7 and 8, we give two polynomial-time algorithms for computing an approximate von Neumann winner from data: one based on Kalai and Vempala (2003)’s follow-the-perturbed leader algorithm, and the other based on projected gradient ascent as studied by Zinkevich (Zinkevich, 2003). These techniques yield learning algorithms that approximate or perform as well as the von Neumann winner using data, time, and space that only depend logarithmically on the cardinality of the space $\Pi$, and therefore, are applicable even with huge policy spaces.

Other related work. Numerous algorithms have been proposed for the (non-contextual) dueling bandits problem: Interleaved Filter (Yue et al., 2009); Beat the Mean (BTM) (Yue and Joachims, 2011); Sensitivity Analysis of VAriables for Generic Exploration (SAVAGE) (Urvoy et al., 2013); Relative Confidence Sampling (Zoghi et al., 2014a); Relative Upper Confidence Bound (RUCB) (Zoghi et al., 2014b); Doubler, MultiSBM and Sparring (Ailon et al., 2014) and mergeRUCB (Zoghi et al., 2015). These methods impose various constraints on the problem at hand, ranging from the requirement that it arise from an underlying utility function (e.g. MultiSBM) to no constraint at all (e.g. SAVAGE); they mainly provided regret bounds that are logarithmic in the number of rounds, and at least linear in the number of actions. In principle, these methods could be applied to contextual dueling bandits by treating policies as actions. But this would lead to regret at least linear in the number of policies which is far worse than the logarithmic bounds obtained in this paper.

The method that is the most closely related to our work is Dueling Bandit Gradient Descent (DBGD) (Yue and Joachims, 2009). It is a policy gradient method that iteratively improves upon the current policy by conducting comparisons with nearby policies, assuming that the policy space comes equipped with a distance metric, and incrementally adapting the policy if a better alternative was encountered. It is similar to gradient optimization methods. As with all local optimization methods, DBGD imposes a convexity assumption on the dueling bandit problem for its performance guarantee: that is the dueling bandit problem needs to arise from the noisy observations of an underlying convex objective function. In this paper, we both relax the assumptions imposed by DBGD and improve upon the regret bound.

2. The contextual dueling bandits problem

In the dueling bandits problem (Yue et al., 2009), the learner has access to $K$ possible actions, $1, \ldots, K$, and attempts to determine the “best” action through repeated stochastic pairwise comparisons of actions, called duels. Thus, at each time-step, the learner chooses a pair of actions $(a, b)$ for a duel; the outcome of the duel is $+1$ if $a$ wins, and $-1$ if $b$ wins. The (unknown) expected value of this outcome is denoted $P(a, b)$, and is assumed to depend only on the selected pair $(a, b)$. In other
words, the probability that \( a \) beats \( b \) in a duel is \( (P(a, b) + 1)/2 \), and the two actions are exactly evenly matched if \( P(a, b) = 0 \). We say that \( a \) beats \( b \) to mean that the chance of \( a \) winning a duel with \( b \) is strictly greater than 1/2; similarly, \( a \) ties \( b \) if this probability is exactly 1/2.

The \( K \times K \) matrix \( P \) of all such expectations \( P(a, b) \) is called the preference matrix.\(^1\) This matrix is initially unknown to the learner, but can be discovered bit-by-bit through experimentation. We assume, of course, that all of the entries of \( P \) are in \([-1, +1]\), and furthermore, that \( P \) is skew-symmetric, meaning that \( P^\top = -P \) so that a duel \((b, a)\) is equivalent to (the negation of) a duel \((a, b)\). (This also implies \( P(a, a) = 0 \) for every action \( a \), as is natural.) Other than this, we strenuously avoid making any assumptions in the current work about the matrix \( P \). For instance, we do not make any assumptions regarding transitivity among the various actions.

In the contextual version of the dueling bandits problem, we suppose that the best way of acting may depend on some observable context. In other words, prior to choosing its actions, the learner is allowed to observe some value \( x \), the context, selected by Nature from some unspecified space \( X \). For instance, \( x \) might be a feature-vector description of a web-user. In this setting, the preference matrix is no longer static; rather, which actions are better than which others now varies and depends on the context, which therefore must be taken into account to fully optimize the choice of actions.

Formally, we assume that on every round \( t \) of the learning process, a context \( x_t \) and preference matrix \( P_t \) are chosen by Nature. The context \( x_t \) is revealed to the learner, but the preference matrix \( P_t \) remains hidden. Based on \( x_t \), the learner selects two actions \((a_t, b_t)\) for a duel, whose outcome has expectation determined by the current (hidden) preference matrix \( P_t \) in the usual way. Except where noted otherwise, in this paper, we always assume that each pair \((x_t, P_t)\) is chosen at random according to independent draws of some unknown joint distribution \( D \).

The goal is to determine which action to select as a function of the context. Such a mapping \( \pi \) from contexts \( x \) to actions \( a \) is called a policy. Typically, we are working with policies of a particular form, that is, from some policy space \( \Pi \). For instance, this space might represent the set of all decision trees. For simplicity, we assume that \( \Pi \) has finite cardinality. Nevertheless, although finite, we generally think of \( \Pi \) as an extremely large space, exponential in any reasonable measure of complexity. Several of our key results generalize immediately to the case that \( \Pi \) is in fact infinite.

3. **“Best” actions and the von Neumann winner**

In the (non-contextual) dueling bandit setting, the “best” action is not always well-defined because there is no measure of the absolute quality of actions. Existing work typically assumes the existence of the Condorcet winner (Urvoy et al., 2013; Zoghi et al., 2014b), that is, an action \( a^* \) that beats every other action \( a \neq a^* \). This is a very natural definition from a preference learning perspective, since \( a^* \) is indeed preferred to every other action. However, it has been shown that dueling bandit problems without Condorcet winners arise regularly in practice (Zoghi et al., 2014a).\(^2\)

Although there is no guarantee of a single action beating all others, the situation changes considerably if we simply allow actions to be selected in a randomized fashion. With this slight and natural relaxation, the problem of non-existence entirely vanishes. Thus, the idea is to find a probability vector \( w \) in \( \Delta_K \) (where \( \Delta_K \) is the simplex of vectors in \([0, 1]^K\) whose entries sum to 1) such that, for all actions \( b \):

\(^1\) In the literature, the preference matrix \( P \) often refers to the matrix of probabilities \( (P(a, b) + 1)/2 \). However, with this modification, \( P(a, b) \) becomes anti-symmetric around 0, which simplifies our arguments considerably.

\(^2\) See also Appendix A for more compelling evidence that this is indeed the case.
Thus, for every action $b$, if $a$ is selected randomly according to distribution $w$, then the chance of beating $b$ in a duel is at least $1/2$. A distribution $w$ with this property is said to be a von Neumann winner for the preference matrix $P$.

As the name reflects, this notion is intimately connected to a game-theoretic interpretation. Indeed, we can view preference matrix $P$ as describing a zero-sum matrix game. In such a game, the two players simultaneously choose distributions (or mixed strategies) $w$ and $u$ over rows and columns, respectively, yielding a gain to the row player of $w^\top Pu$. According to von Neumann’s celebrated minmax theorem, for any matrix $P$,

$$\max_{w \in \Delta_K} \min_{u \in \Delta_K} w^\top Pu = \min_{u \in \Delta_K} \max_{w \in \Delta_K} w^\top Pu,$$

the common value being the value of the game $P$. A maxmin strategy $w$ or a minmax strategy $u$ is one realizing the max or min on the left- or right-hand side of this equality, respectively. Finding these strategies is called solving the game.

In our case, we have assumed that the matrix $P$ is itself skew-symmetric, which means that the game it describes is a symmetric game. Such games are known to have value exactly equal to zero (see, for instance, (Owen, 1995, Theorem II.6.2)). Working through definitions, this means that $w$ is a maxmin strategy if and only if

$$\min_{u \in \Delta_K} w^\top Pu \geq 0.$$ 

But this is exactly equivalent to Eq. (1). Therefore, we have argued the following:

**Proposition 1** A probability vector $w$ is a von Neumann winner for preference matrix $P$ if and only if it is a maxmin strategy for the game $P$. Consequently, every preference matrix $P$ has a von Neumann winner.

**Extension to contextual dueling bandits.** Moving to the contextual dueling bandit setting, we can regard each policy $\pi$ as a kind of “meta-action,” and define a $|\Pi| \times |\Pi|$ preference matrix $M$ over these meta-actions. Thus, the rows and columns of $M$ are each indexed by policies in $\Pi$, and

$$M(\pi, \rho) = E_{(x, P) \sim D} [P(\pi(x), \rho(x))].$$

This quantity is the expected outcome when a “meta-duel” is held between the two policies $\pi$ and $\rho$, whose stochastic outcome is determined by randomly selecting $(x, P) \sim D$, and then holding an ordinary duel on $P$ between the actions $\pi(x)$ and $\rho(x)$. This huge matrix thus encodes the probability of any policy beating any other policy in a duel.

We can apply the definition of von Neumann winner to $M$, which will necessarily exist (by Proposition 1), and will be defined by a probability distribution $W$ over policies such that if $\pi$ is first chosen at random from $W$, then the probability that $\pi$ beats any other policy $\rho$ in a duel is at least $1/2$. That is, the randomized policy defined by $W$ beats or ties every policy in the space $\Pi$.

For the rest of the paper, we study how to compute (or approximate) such a von Neumann winner. Of course, because the space $\Pi$ and corresponding matrix $M$ are both gigantic, this will present significant computational challenges.
Other solution concepts. Given that the existence of the Condorcet winner is not guaranteed, a number of definitions have been proposed previously to remedy this issue (Schulze, 2011). Two such definitions are the Borda winner, the action that has the highest probability of winning a duel against a uniformly random action; and the Copeland winner, the action that wins the most pairwise comparisons (Urvoy et al., 2013). Both the Borda and the Copeland method fail the independence of clones criterion (Schulze, 2011), meaning that adding multiple identical copies of an action can change the Borda or Copeland winner. This criterion is particularly crucial in a dueling bandit setting, because a given policy class may contain many identical policies. In contrast, the von Neumann winner is one which performs at least as well as any individual policy, and is thus unaffected by the presence or absence of clones. See Appendix B for a more detailed discussion.

4. Learning scenarios

We consider two possible learning scenarios.

In the simpler of these, called explore-then-exploit, we suppose that the learner is allowed to explore for some number of rounds $m$ (where, as described above, on each round, the learner is presented with random context and permitted to run and observe the outcome of a duel between a pair of actions of its choosing). At the end of these $m$ rounds, the learner outputs a distribution $\hat{W}$ over policies in $\Pi$. The learner’s goal is to produce $\hat{W}$ which is an $\varepsilon$-approximate von Neumann winner, that is, for which

$$\min_{U \in \Delta_{|\Pi|}} \hat{W}^T MU \geq -\varepsilon$$

for some small $\varepsilon > 0$. In other words, for all $\pi \in \Pi$, $\hat{W}$ should beat $\pi$ with probability at least $1/2 - \varepsilon/2$. Naturally, $m$ should be “reasonable” as a function of $\varepsilon$. This setting is almost like learning from a passively selected batch of training examples, except that the learner has an active role in selecting which actions to play in each duel.

In the alternative full-explore-exploit setting, learning occurs in a fully online manner across $T$ rounds (in the manner described earlier), with performance measured using some notion of regret. In this paper, where we are working with policies and changing preference matrices, we propose to define regret to be

$$\max_{\pi \in \Pi} \frac{1}{2} \sum_{t=1}^{T} [P_t(\pi(x_t), a_t) + P_t(\pi(x_t), b_t)]. \quad (3)$$

If we can find an algorithm for which this regret is $o(T)$, then eventually the algorithm selects actions $(a_t, b_t)$ which cannot be beaten by any other policy $\pi \in \Pi$.

In the standard dueling-bandits setting with a static preference matrix, a seemingly different definition of regret was used by Yue et al. (2009) in terms of an assumed Condorcet winner. However, when specialized to their setting, and when provided with their same assumptions, their definition can be shown to be equivalent (up to constant factors) to Eq. (3).

5. Sparring Exp4.P

Our goal then is to find, approximate, or perform as well as a von Neumann winner, which, as we have seen, is a maxmin strategy for a particular game. Under this interpretation, it becomes...
especially natural to use ordinary no-regret learning algorithms as players of this game since it is known that such algorithms, when properly configured for this purpose, will converge to maximin or minmax strategies (Freund and Schapire, 1999). The idea is simply to run two independent copies of such an algorithm against one another. Such a “sparring” approach was previously proposed for dueling bandits by Ailon et al. (2014), though without details, and not in the contextual setting.

To be concrete, we consider using the multi-armed bandit algorithm Exp4.P (Beygelzimer et al., 2011) for this purpose in the full-explore-exploit setting. Exp4.P is well-suited since it is designed to work with partial information as in our bandit setting, and since it can handle the kind of adversarially generated data that arises unavoidably when playing a game. It also is designed to work with policies in a contextual setting like ours (or, more generally, to accept the advice of “experts”).

Specifically, the learning setting for Exp4.P is as follows (somewhat, but straightforwardly, modified for our present purposes). There are $K$ possible actions, $1, \ldots, K$, and a finite space $\Pi$ of policies. On each round $t = 1, \ldots, T$, an adversary chooses and reveals context $x_t$, and also chooses, but does not reveal rewards $r_t(1), \ldots, r_t(K) \in [-1, +1]$ for each of the $K$ actions. The learner then selects an action $a_t$, and receives the revealed reward $r_t(a_t)$. The learner’s total reward is thus $G_A = \sum_{t=1}^{T} r_t(a_t)$, while the reward of each policy $\pi$ is $G_\pi = \sum_{t=1}^{T} r_t(\pi(x_t))$. The learner’s goal is to receive reward close to that of the best policy. Beygelzimer et al. (2011) prove that, with probability at least $1 - \delta$, Exp4.P achieves reward

$$G_A \geq \min_{\pi \in \Pi} G_\pi - 12 \sqrt{KT \ln(||\Pi||/\delta)}$$

(subject to very benign conditions).

To use Exp4.P for contextual dueling bandits in the full-explore-exploit setting, we run two separate copies which are played against one another; let us call them row-Exp and column-Exp. We use the same actions, contexts, and policies for the two copies as for the given dueling bandits problem. On each round $t$, Nature chooses a context $x_t$ and preference matrix $P_t$. The context (but not the preference matrix) is revealed to row-Exp and column-Exp, which select actions $a_t$ and $b_t$, respectively. A duel is then held between these two actions; the outcome $r$ is passed as feedback to row-Exp (for its chosen action $a_t$), and its negation $-r$ is similarly passed to column-Exp.

Although, algorithmically, this is a complete description, to actually fulfill the requirements of the learning model, we also need to define rewards $r_t(a)$ for all of the actions that were not chosen. Furthermore, these rewards need to be defined before each copy of the algorithm chooses its action (or, more technically, in a manner that is conditionally independent of each copy’s choice). To this end, for every pair of actions $(a, b)$, we define a $\{-1, +1\}$-valued random variable $R_t(a, b)$ with expected value $P_t(a, b)$. Thus, $R_t(a, b)$ can be viewed as the outcome of a hypothetical duel between actions $a$ and $b$. These values are only used for the mathematical argument, and do not literally have to be computed. Only the pair $(a_t, b_t)$ is actually used in a duel.

For row-Exp, based on column-Exp’s chosen action $b_t$, we then define rewards $r^{(row)}_t(a) = R_t(a, b_t)$ for all $a$. And similarly, for column-Exp, based on row-Exp’s chosen action $a_t$, we define rewards $r^{(col)}_t(b) = -R_t(a_t, b)$ for all $b$. In particular, this means that row-Exp receives, for its chosen action $a_t$, the reward $R_t(a_t, b_t)$ (that is, the result of a duel between $a_t$ and $b_t$), while column-Exp receives the reward $-R_t(a_t, b_t)$ for its chosen action $b_t$.

**Theorem 2** Suppose sparring Exp4.P is used in the manner described above. Then with probability at least $1 - \delta$, its regret (as defined in Eq. (3)) is at most $O(\sqrt{KT \ln(||\Pi||/\delta)})$.

(All proofs appear in Appendix C.)
Note that this analysis holds also for adversarial data in which the pairs \((x_t, P_t)\) are selected by an adversary rather than at random. Also, we can adapt this algorithm for explore-then-exploit learning using standard techniques for online-to-batch conversion; given \(m\) exploration trials, such an algorithm will find an \(\varepsilon\)-approximate von Neumann winner where \(\varepsilon = O(\sqrt{K \ln(|\Pi|/\delta)/m})\).

Although yielding very good regret bounds and handling adversaries, this approach requires time and space proportional to \(|\Pi|\), and is therefore not practical for extremely large policy spaces.

6. Explore-first algorithms

We next begin a development that will lead to efficient methods for handling even extremely large policy spaces, under a particular assumption discussed below. We describe a general approach for exploration, for using the collected data to find a statistically sound solution, and for reducing the problem that must be solved to a more tractable form.

We focus mainly on the explore-then-exploit problem. Thus, on each of \(m\) rounds, a pair \((x_t, P_t)\) is selected at random, and the learner is permitted to choose and observe the outcome of a single duel \((a_i, b_i)\). Although \(x_i\) is observed, \(P_i\) is not. Nevertheless, it is possible for the learner to obtain a kind of noisy version of it. In particular, suppose the learner chooses the dueling pair \((a_i, b_i)\) uniformly at random, and let \(r_i\) be the outcome. We can then define a matrix \(\hat{P}_i\) where \(\hat{P}_i(a_i, b_i) = K^2 r_i\), and all other entries \(\hat{P}_i(a, b)\) are set to zero. It can be verified that the expected value of each entry \(\hat{P}_i(a, b)\) is exactly \(P_i(a, b)\); that is, \(E[\hat{P}_i P_i] = P_i\). Also, the entries are bounded so that \(|\hat{P}_i(a, b)| \leq L\) for all \((a, b)\), where \(L = K^2\). For generality, our development depends on only these two properties, and thus can be applied if another method is used to define \(\hat{P}_i\) (possibly with a different value of \(L\)).

**Statistical guarantees.** With these noisy versions of the empirical preference matrices, we can estimate the expected outcome in a “meta-duel” between two policies \(\pi\) and \(\rho\), that is, an entry of the matrix \(M\) defined in Eq. (2). In particular, let

\[
\hat{M}(\pi, \rho) = \frac{1}{m} \sum_{i=1}^{m} \hat{P}_i(\pi(x_i), \rho(x_i)).
\]

Then the expected value of this quantity is \(\hat{M}(\pi, \rho)\), the corresponding entry of \(M\). Moreover, using Hoeffding’s inequality and the union bound, we can straightforwardly show that, with probability at least \(1 - \delta\),

\[
\left| \hat{M}(\pi, \rho) - M(\pi, \rho) \right| \leq \varepsilon' \quad \text{for all } (\pi, \rho) \in \Pi \times \Pi,
\]

where \(\varepsilon' = O\left(L \sqrt{\ln(|\Pi|/\delta)/m}\right)\). Thus, although huge, the matrix \(\hat{M}\) is well-approximated by the matrix \(\hat{M}\) using only a moderate-sized sample.

Our goal is to find an approximate maxmin strategy for \(M\). However, this argument shows that it is sufficient to instead find an approximate maxmin strategy for \(\hat{M}\), which will be the approach taken by our algorithms:

**Lemma 3** Given the set-up above, suppose that Eq. (6) holds (as will be the case with probability at least \(1 - \delta\)), and suppose further that \(\hat{W}\) is a probability vector for which

\[
\min_{U \in \Delta_{|\Pi|}} \hat{W}^\top \hat{M} U \geq \max_{W \in \Delta_{|\Pi|}} \min_{U \in \Delta_{|\Pi|}} W^\top \hat{M} U - \varepsilon.
\]

Then \(\hat{W}\) is a \((2\varepsilon' + \varepsilon)\)-approximate von Neumann winner for \(M\).
We will later give efficient algorithms that allow us to compute approximate solutions \( \hat{w} \) with arbitrarily small \( \varepsilon \). Thus, if we choose \( \varepsilon = O(\varepsilon') \), then \( \hat{w} \) will be an \( O(\varepsilon') \)-approximate von Neumann winner. Although we have focused here on the explore-then-exploit setting, we comment that we can also use this same approach in a full-explore-exploit setting simply by doing all of the exploration on the first \( m \) trials, and using the computed approximate von Neumann winner on the remaining trials. Assuming \( T \) is known in advance, \( m \) can be chosen appropriately, leading to an algorithm with regret \( O\left(LT^{2/3}(\ln(|\Pi|/\delta))^{1/3}\right) \). This is clearly \( o(T) \), but weaker than the regret bound for sparring Exp4.P given in Section 5.

### A more compact version of the problem.

So our aim now is to find an approximate maxmin strategy for the matrix \( \hat{M} \). Although this matrix is gigantic in both dimensions, by leveraging how it was constructed from only a small number of empirical observations, we can re-express the problem in a far more compact form. To this end, let us define, for each policy \( \pi \in \Pi \), a *policy vector* \( \nu_{\pi} \in \mathbb{R}^{mK} \) that encodes the behavior of \( \pi \) on the exploration data. For readability, although a vector, we index entries of \( \nu_{\pi} \) by pairs \((i,a)\), and we define

\[
\nu_{\pi}(i,a) = \frac{1}{\sqrt{m}} \mathbb{1}\{\pi(x_i) = a\}.
\]

Thus, \( \nu_{\pi} \) is broken into \( m \) length-\( K \) blocks, with block \( i \) encoding in a natural way the action selected by \( \pi \) on \( x_i \). (The constant \( 1/\sqrt{m} \) is for normalization.)

We also define an \( mK \times mK \) block-diagonal matrix \( B \), where the \( m \) blocks along the diagonal are exactly the \( K \times K \) matrices \( \hat{P}_i \) described above. Formally, using the earlier indexing,

\[
B((i,a),(j,b)) = \mathbb{1}\{i = j\} \hat{P}_i(a,b).
\]

Working through these definitions, it can be verified that for any two policies \( \pi \) and \( \rho \), the quantity \( \nu_{\pi}^\top B \nu_{\rho} \) is exactly equal to \( \hat{M}(\pi,\rho) \) as defined in Eq. (5). This means that if \( W \) and \( U \) are probability vectors over \( \Pi \), then

\[
W^\top \hat{M} U = \left( \sum_\pi W(\pi) \nu_{\pi} \right)^\top B \left( \sum_\rho U(\rho) \nu_{\rho} \right).
\]

Therefore, the problem of finding a maxmin strategy for \( \hat{M} \) is equivalent to solving

\[
\max_{w \in C} \min_{u \in C} w^\top Bu \quad (7)
\]

where \( C \) is the convex hull of the set of all policy vectors \( \{\nu_{\pi} : \pi \in \Pi\} \). Furthermore, a solution \( w \in C \) is necessarily a convex combination of vectors \( \nu_{\pi} \), and therefore corresponds to a probability vector over policies.

The formulation given in Eq. (7) shows that \( B \) should itself be viewed as a game matrix, and that our goal is to approximately solve this game. This matrix has the advantage of being far smaller than \( \hat{M} \). However, unlike a conventional matrix game, the space from which the players’ vectors \( w \) and \( u \) are chosen is not the standard space of probability vectors, but rather the convex hull of an exponentially large set of vectors. What remains is the problem of finding a maxmin strategy solving Eq. (7); we give two algorithms in the sections that follow.
Classification oracle. Our algorithms assume that the policy space $\Pi$ is structured in a way that admits a certain computational operation that is quite natural in the realm of learning. Specifically, we assume the existence of a classification oracle. The input to this oracle is a sequence of cost vectors $c_1, \ldots, c_m$, each in $\mathbb{R}^K$, with the interpretation that $c_i(a)$ is the cost of choosing action $a$ on context $x_i$. The output of the oracle is the policy in $\Pi$ with minimum cost, that is,

$$\arg\min_{\pi \in \Pi} \sum_{i=1}^m c_i(\pi(x_i)).$$

(8)

Indeed, regarding the $x_i$’s as examples, the actions $a$ as labels or classes, and the policies $\pi$ as classifiers, we see that this oracle is in fact solving an empirical, cost-sensitive, multi-class classification problem. Thus, the assumption of such an oracle is an idealization based on the numerous cases in which effective classification algorithms already exist. In practice, we hope that the methods developed will be effective when using ordinary off-the-shelf classification algorithms as oracle.

Equivalently the classification oracle can be described in terms of vectors. Specifically, the cost vectors above can be identified with their concatenation, a single vector $c \in \mathbb{R}^{mK}$, divided naturally into $m$ blocks. Then the problem given in Eq. (8) is the same as

$$\arg\min_{w \in C} c \cdot w,$$

since the minimum, without loss of generality, will be a vector $v_\pi$ where $\pi$ minimizes Eq. (8). Therefore, in what follows, we use expressions of this latter form to indicate an invocation of our assumed classification oracle.

7. Sparring FPL

We give two algorithms for solving the optimization problem in Eq. (7). Although there exist many methods for solving such a game, the challenge here is the requirement that the solution be in the set $C$. As already seen in Section 5, regret minimization algorithms are a natural choice. However, most standard algorithms will not conform to this constraint.

Our first method is based on the Follow-the-Perturbed-Leader (FPL) algorithm of Kalai and Vempala (2003), which is designed for a standard online learning problem: Let $D$ and $L$ be subsets of $\mathbb{R}^m$. On each round $t = 1, \ldots, T$, the learner chooses a decision vector $d_t \in D$, and then receives a loss vector $\ell_t \in L$. The learner’s goal is to minimize its cumulative loss $\sum_{t=1}^T d_t \cdot \ell_t$ relative to the best possible loss using a fixed decision, that is, $\min_{d \in D} \sum_{t=1}^T d \cdot \ell_t$. FPL chooses $d_t$ as the best such vector based on a slightly perturbed version of the preceding losses; that is,

$$d_t = \arg\min_{d \in D} d \cdot \left( \sum_{\tau=1}^{t-1} \ell_\tau + p_t \right),$$

where $p_t \in \mathbb{R}^m$ is chosen uniformly at random from $[0, 1/\alpha]^m$. Kalai and Vempala (2003, Theorem 1.1) prove the following (slightly simplified) result: assume that $D$, $R$ and $A$ are such that for all $d \in D$ and $\ell \in L$ we have that $\|d\|_1 \leq D$, $|d \cdot \ell| \leq R$ and $\|\ell\|_1 \leq A$. Also, let $\alpha = \sqrt{2D/(RAT)}$. Then, for any sequence $\ell_1, \ldots, \ell_T \in L$,

$$\mathbb{E} \left[ \sum_{t=1}^T d_t \cdot \ell_t \right] \leq \min_{d \in D} \sum_{t=1}^T d \cdot \ell_t + 2\sqrt{2D}RAT.$$

(9)

where the expectation is over the random choice of perturbations. Kalai and Vempala prove this in the oblivious case that the adversary has fixed the $\ell_t$’s ahead of time. However, this restriction can
be relaxed to allow each $\ell_t$ to be selected adaptively in a possibly stochastic fashion that may depend on the entire preceding history through round $t - 1$, but not on the perturbation $p_t$ for the current round. Using a martingale argument and Azuma’s lemma (see also (Cesa-Bianchi and Lugosi, 2006, Lemma 4.1)), it can then be shown that, with probability at least $1 - \delta$,

$$
\sum_{t=1}^{T} d_t \cdot \ell_t \leq \min_{d \in D} \sum_{t=1}^{T} d \cdot \ell_t + 2\sqrt{2DRAT} + 2R\sqrt{2T \ln(1/\delta)}.
$$

(10)

We use this result to solve Eq. (7) by sparring two copies of FPL, called row-FPL and column-FPL, in the fashion of a repeated game. On every round $t$, row-FPL uses FPL to select a vector $w_t$, while column-FPL uses a different copy of FPL to select a vector $u_t$. We then define the resulting loss vectors to be $-Bu_t$ for row-FPL, and $w_t^\top B$ for column-FPL. Here is the complete algorithm:

- $t = 1, \ldots, T$:
  - Choose uniform random perturbations $p_t, q_t$ from $[0, 1/\alpha]^m$.
  - Let $w_t = \arg\min_{w \in C} w \cdot [-B(u_1 + \cdots + u_{t-1}) + p_t]$.
  - Let $u_t = \arg\min_{u \in C} [(w_1 + \cdots + w_{t-1})^\top B + q_t] \cdot u$.
- output $\bar{w} = \frac{1}{T} \sum_{t=1}^{T} w_t$

Note that the $\arg\min$ expression that appear in the algorithm can be solved using our assumed classification oracle. Also, the returned vector $\bar{w}$ is in $\mathcal{C}$, and in fact corresponds to a simple mixture of $T$ policies.

**Theorem 4** Let $\alpha = \sqrt{2/(L^2 T)}$. Then with probability at least $1 - \delta$, the vector $\bar{w}$ satisfies

$$
\min_{u \in \mathcal{C}} \bar{w}^\top Bu \geq \max_{w \in \mathcal{C}} \min_{u \in \mathcal{C}} w^\top Bu - 2\varepsilon
$$

where $\varepsilon = 2L\sqrt{2m/T} + 2L\sqrt{2 \ln(2/\delta)/T}$.

Thus, to find an $\varepsilon$-approximate solution, we can choose $T$ to be $O(L^2(m + \ln(1/\delta))/\varepsilon^2)$. This also gives a bound on the number of calls to classification oracle (which is called twice on each round). This algorithm is extremely simple and intuitive. But in the next section, we give an alternative, deterministic algorithm that gives a different trade-off between $m$ and $\varepsilon$.

8. **Projected gradient ascent**

We can also solve the game $B$ using online projected gradient ascent methods as studied by Zinkevich (2003). This algorithm maintains a vector $w_t \in \mathcal{C}$ corresponding to a strategy for the row player. On every round, a column strategy $u_t \in \mathcal{C}$ is chosen that is a “best response” to $w_t$. The strategy $w_t$ is updated by taking a small gradient step. The resulting vector $z_{t+1}$ is likely to be outside the set $\mathcal{C}$, and therefore is (approximately) projected back to $\mathcal{C}$, yielding $w_{t+1}$. So, the algorithm is like this:

- choose any $w_1 \in \mathcal{C}$
- for $t = 1, \ldots, T_{out}$:
  - $u_t = \arg\min_{u \in \mathcal{C}} w_t^\top Bu$
- \( z_{t+1} = w_t + \eta B u_t \)
- \( w_{t+1} = \text{ApproxProject}(z_{t+1}, w_t) \)

- output \( \mathbf{w} = \frac{1}{T_{\text{out}}} \sum_{t=1}^{T_{\text{out}}} w_t \)

Ideally, we would like for \( w_{t+1} \) to be the exact Euclidean projection of \( z_{t+1} \) onto \( C \), but instead need to settle for an approximation. For this purpose, the procedure \( \text{ApproxProject}(z, v_1) \), described below, computes an approximate projection of an arbitrary vector \( z \) onto \( C \). It takes as input a second vector \( v_1 \) that is already in \( C \), and which we can think of as an initial guess at the actual projection. The quality (as an approximation) of the returned vector \( v \) is allowed to depend on how close \( v_1 \) is to \( z \). Specifically, we require that, for all \( s \in C \),

\[
\|s - v\|^2 \leq \|s - z\|^2 + \alpha \cdot \|v_1 - z\|
\]

(11)

where \( \alpha \) is a constant that we specify later.

Using an analysis similar to Zinkevich (2003), but for fixed learning rate, and taking into account the errors introduced by imperfect projections, we can show the following:

**Lemma 5** For the algorithm described above with \( \eta = 2/(L\sqrt{T_{\text{out}}} \) , we have

\[
\frac{1}{T_{\text{out}}} \sum_{t=1}^{T_{\text{out}}} w_t^\top B u_t \geq \min_{w \in C} \frac{1}{T_{\text{out}}} \sum_{t=1}^{T_{\text{out}}} w^\top B u_t - \varepsilon
\]

where \( \varepsilon = 2L/\sqrt{T_{\text{out}}} + L\alpha/2. \)

We can prove that the returned vector \( \mathbf{w} \) is an \( \varepsilon \)-approximate maxmin solution using a technique similar to Freund and Schapire (1999). (Alternatively, we could use the average of the \( u_t \)'s which is an \( \varepsilon \)-approximate minmax solution by the same proof.)

**Theorem 6** The vector \( \mathbf{w} \) satisfies \( \min_{u \in C} \mathbf{w}^\top B u \geq \max_{w \in C} \min_{u \in C} \mathbf{w}^\top B u - \varepsilon \) where \( \varepsilon \) is as Lemma 5.

**Computing approximate projections.** It remains to describe the approximate-projection procedure \( \text{ApproxProject}(z, v_1) \). Given an arbitrary vector \( z \) and another vector \( v_1 \in C \), the goal of the algorithm, as in Eq. (11), can be restated as that of finding a vector \( v \in C \) for which

\[
\min_{s \in C} F(s, v) \geq -\alpha \cdot \|v_1 - z\|
\]

(12)

where we define \( F(s, v) = \|s - z\|^2 - \|s - v\|^2 = 2s \cdot (v - z) + \|z\|^2 - \|v\|^2 \). Note that \( F \) is linear in \( s \) (for each \( v \)), and concave in \( v \) (for each \( s \)). To ensure that Eq. (12) holds, we give an algorithm that aims to maximize the left-hand side of this inequality. (As a side note, the maximizing vector turns out to be exactly the projection of \( z \) onto \( C \), although we do not require that fact for our algorithm and analysis.)

To this end, we use an algorithm that resembles repeated play of a game in which the payoff is defined by \( F \). The \( s \) player uses best response on each round, while the \( v \) player again uses a variant of online gradient ascent applied to the function \( F(\cdot, s_t) \). The algorithm takes a parameter \( \nu \in (0, 1] \), and uses \( v_1 \in C \), which was provided as an argument to \( \text{ApproxProject}(z, v_1) \), as an initial vector. Here is the algorithm:
for $t = 1, \ldots, T_{in}$:
- $s_t = \arg\min_{s \in C} s \cdot (v_t - z)$
- $v_{t+1} = (1 - \nu)v_t + \nu s_t$

output $\nabla = \frac{1}{T_{in}} \sum_t v_t$

Note that $v_t$ is in $C$ for every $t$ (by convexity of $C$), and therefore $\nabla$ is as well.

Let $v^*$ be the projection of $z$ onto $C$. We can prove the following for this algorithm using $\|v^* - v_t\|^2$ as a potential function.

**Lemma 7** For the above algorithm with $\nu = \|z - v_1\|/\sqrt{T_{in}}$ and $\delta = \frac{8}{\sqrt{T_{in}}} \|z - v_1\|$, we have

$$\frac{1}{T_{in}} \sum_{t=1}^{T_{in}} F(s_t, v_t) \geq \frac{1}{T_{in}} \sum_{t=1}^{T_{in}} F(s_t, v^*) - \delta.$$ 

Next, we show that $\nabla$ satisfies the specification given in Eq. (12).

**Theorem 8** For the algorithm above, $\min_{s \in C} F(s, \nabla) \geq -\delta$ with $\delta$ set as in Lemma 7. Thus, Eq. (12) holds for $\nabla$ if we set $\alpha = 8/\sqrt{T_{in}}$.

Finally, combining with Lemma 5 and Theorem 6, this shows that the overall solution $\hat{w}$ will be an $\varepsilon$-approximate maxmin solution where $\varepsilon = 2L/\sqrt{T_{out}} + 4L/\sqrt{T_{in}}$. Thus, we can obtain any desired value of $\varepsilon$ by setting $T_{in} = T_{out} = \lceil 36L^2/\varepsilon^2 \rceil$. The resulting number of calls to the classification oracle will be $T_{out} + T_{in}T_{out} = O(L^4/\varepsilon^4)$. As earlier noted, compared to sparring FPL, this bound gives a different trade-off between $m$ and $\varepsilon$. For the case that $\varepsilon = O(\varepsilon')$, and with $\varepsilon'$ as in Section 6, this algorithm gives a better bound by a factor of $O(\ln |\Pi|)$.

**Acknowledgments**

We would like to thank Alekh Agarwal for insightful comments and discussions.

**References**

Nir Ailon, Zohar Karnin, and Thorsten Joachims. Reducing dueling bandits to cardinal bandits.
In *Proceedings of the International Conference on Machine Learning (ICML)*, pages 856–864, 2014.

Peter Auer, Nicolò Cesa-Bianchi, Yoav Freund, and Robert E. Schapire. The nonstochastic multiarmed bandit problem. *SIAM J. Computing*, 32(1):48–77, 2002.

Alina Beygelzimer, John Langford, Lihong Li, Lev Reyzin, and Robert Schapire. Contextual bandit algorithms with supervised learning guarantees. *J. Mach. Learn. Res.*, 15:19–26, 2011.

Nicolò Cesa-Bianchi and Gábor Lugosi. *Prediction, Learning, and Games*. Cambridge University Press, 2006.

Olivier Chapelle, Thorsten Joachims, Filip Radlinski, and Yisong Yue. Large-scale validation and analysis of interleaved search evaluation. *ACM Transactions on Information Systems (TOIS)*, 30 (1):6:1–6:41, 2012.
Miroslav Dudík, Daniel Hsu, Satyen Kale, Nikos Karampatziakis, John Langford, Lev Reyzin, and Tong Zhang. Efficient optimal learning for contextual bandits. In Uncertainty in Artificial Intelligence, pages 169–178, 2011.

Yoav Freund and Robert E. Schapire. Adaptive game playing using multiplicative weights. Games and Economic Behavior, 29:79–103, 1999.

Martin Gardner. Mathematical games: The paradox of the nontransitive dice and the elusive principle of indifference. Scientific American, 223:110–114, 1970.

Fan Guo, Lei Li, and Christos Faloutsos. Tailoring click models to user goals. In WSCD, pages 88–92, 2009a.

Fan Guo, Chao Liu, and Yi-Min Wang. Efficient multiple-click models in web search. In Proceedings of the International Conference on Web Search and Data Mining (WSDM), pages 124–131, 2009b.

Katja Hofmann, Shimon Whiteson, and Maarten de Rijke. A probabilistic method for inferring preferences from clicks. In Proceedings of the International Conference on Information and Knowledge Management (CIKM), pages 249–258, 2011.

Katja Hofmann, Shimon Whiteson, and Maarten de Rijke. Fidelity, soundness, and efficiency of interleaved comparison methods. ACM Transactions on Information Systems (TOIS), 31(4), 2013.

Adam Kalai and Santosh Vempala. Efficient algorithms for the online decision problem. In Conference on Learning Theory (COLT), pages 26–40, 2003.

John Langford and Tong Zhang. The epoch-greedy algorithm for contextual multi-armed bandits. In NIPS, pages 817–824, 2007.

Guillermo Owen. Game Theory. Emerald Group Publishing Limited, 3rd edition, 1995.

Filip Radlinski, Madhu Kurup, and Thorsten Joachims. How does clickthrough data reflect retrieval quality? In Proceedings of the International Conference on Information and Knowledge Management (CIKM), pages 43–52, 2008.

Markus Schulze. A new monotonic, clone-independent, reversal symmetric, and Condorcet-consistent single-winner election method. Social Choice and Welfare, 36(2):267–303, 2011.

Tanguy Urvoys Fabrice Clerot, Raphael Féraud, and Sami Naamane. Generic exploration and k-armed voting bandits. In Proceedings of the International Conference on Machine Learning (ICML), pages 91–99, 2013.

Yisong Yue and Thorsten Joachims. Interactively optimizing information retrieval systems as a dueling bandits problem. In Proceedings of the International Conference on Machine Learning (ICML), pages 1201–1208, 2009.

Yisong Yue and Thorsten Joachims. Beat the mean bandit. In Proceedings of the International Conference on Machine Learning (ICML), pages 241–248, 2011.
Yisong Yue, J. Broder, R. Kleinberg, and T. Joachims. The k-armed dueling bandits problem. In Conference on Learning Theory (COLT), 2009.

Martin Zinkevich. Online convex programming and generalized infinitesimal gradient ascent. In Proceedings of the International Conference on Machine Learning (ICML), pages 928–936, 2003.

Masrour Zoghi, Shimon Whiteson, Maarten de Rijke, and Rémi Munos. Relative confidence sampling for efficient on-line ranker evaluation. In Proceedings of the International Conference on Web Search and Data Mining (WSDM), pages 73–82, 2014a.

Masrour Zoghi, Shimon Whiteson, Rémi Munos, and Maarten de Rijke. Relative upper confidence bound for the k-armed dueling bandits problem. In Proceedings of the International Conference on Machine Learning (ICML), pages 10–18, 2014b.

Masrour Zoghi, Shimon Whiteson, and Maarten de Rijke. MergeRUCB: A method for large-scale online ranker evaluation. In Proceedings of the International Conference on Web Search and Data Mining (WSDM), pages 17–26, 2015.

Appendix A. Failure of the Condorcet winner to exist

Here, we investigate the reliability of the Condorcet assumption by replicating the experiment in (Zoghi et al., 2014a, §6.1) in a slightly different setting: as in their setting, we consider a family of K-armed dueling bandit problems arising from the ranker evaluation problem in IR, where the comparisons are carried out using Probabilistic Interleave (Hofmann et al., 2011) and using click models (Guo et al., 2009a,b) to simulate user behavior. The rankers are sampled from the set of 136 rankers provided with the MSLR dataset3. However, unlike the experiments in Zoghi et al. (2014a), we use an informational click model, rather than a perfect one Hofmann et al. (2013): the former simulates the behavior of a user who is seeking general information about a broad topic, while the former represents an idealized user, who meticulously examines every document in the retrieved list. As such, we claim that the setting we consider here is a more realistic one.

The plot in Figure 1 shows the probability with which the encountered dueling bandit problems contain Condorcet winners. As this figure demonstrates, in this setting, the occurrence of the Condorcet winner drops rapidly as the number of rankers grows.

This shows that even in this simple non-contextual example the assumption that there exists a Condorcet winner is too unreliable to be practical. Needless to say that in the contextual dueling bandit problem, where one is dealing with a potentially very large and diverse set of policies, the likelihood of one policy dominating every single other policy is rather far-fetched.

Appendix B. Comparison between the Copeland and von Neumann winners

A Copeland winner is defined to be any arm that beats the largest number of other arms. It is a generalization of the Condorcet winner in the sense that if the Condorcet winner exists, it will be a Copeland winner. However, we claim that the von Neumann winner is a more natural generalization
than the Copeland winner for the following two reasons: first, in the absence of a Condorcet winner, Copeland winners, both individually and as a collective, can lose to an arm that is not a Copeland winner, whereas the von Neumann winner beats or ties with every single arm; secondly, the set of Copeland winners can be altered by the introduction of “clones,” i.e. arms whose corresponding rows of the preference matrix are identical to each other.

To demonstrate this lack of stability of the Copeland winners, consider any $K + 3$-armed example with $K > 4$, where arms $a_1, a_2, a_3$ beat all other arms and the three of them are in a cycle, with $a_1$ beating $a_2$, $a_2$ beating $a_3$ and $a_3$ beating $a_1$ all with probability $1$. It is easy to see that these three arms are the only Copeland winners with Copeland score equal to $K + 1$ and also form the support of the von Neumann distribution: indeed, the von Neumann distribution is simply the uniform distribution on these three arms. Now, let us consider a slight modification of this problem, where we add one more arm, called $a_0$, that is nothing but a duplicate of arm $a_1$ (hence $p_{01} = 0.5$ and $p_{0j} = p_{1j}$ for all $j > 1$). In the following we explain what happens to the set of Copeland winners after this modification: in the presence of ties, there are three sensible definitions that one could use for the Copeland score; these definitions and the corresponding scores for the top four arms in our modified example can be found in Table 1. As the quantities in this table show, regardless of the definition of Copeland score used, the set of Copeland winners for our new $K + 4$-armed dueling bandit problem does not contain all of $a_0, \ldots, a_3$. Indeed, under no definition can arm $a_2$ be considered a Copeland winner.

On the other hand, arms $a_0, a_1, a_2, a_3$ still form the support of the von Neumann distribution of this modified dueling bandit problem: if we assign weights $w_0, w_1, \ldots, w_K$ to these four arms such
Table 1: Copeland scores of the top arms in the duplicated example

| Def’n of Copld($a_i$) | $\sum_j 1_{p_{ij} > 0.5}$ | $\sum_j 1_{p_{ij} \geq 0.5}$ | $\sum_j 1_{p_{ij} > 0.5} - \sum_j 1_{p_{ij} < 0.5}$ |
|-----------------------|---------------------------|--------------------------|---------------------------------|
| $a_0, a_1$            | $K + 1$                   | $K + 3$                  | $K$                             |
| $a_2$                 | $K + 1$                   | $K + 2$                  | $K - 1$                         |
| $a_3$                 | $K + 2$                   | $K + 3$                  | $K + 1$                         |

that

$$w_0 + w_1 = w_2 = w_3 = \frac{1}{3} \quad \text{and} \quad w_4 = \cdots = w_K = 0,$$

then given any arm $a_j$ and sampling $a_i$ according to the probability distribution given by the weights $w_i$, then $a_i$ would beat $a_j$ on average if $j > 3$ and they would tie if $j \leq 3$.

We consider the lack of stability under cloning illustrated by this example to be a major drawback of the Copeland score as a measure of quality.

Appendix C. Proofs

Proof [Theorem 2] Let us first take the point of view of Exp4.P. Plugging in to Eq. (4), we have that, with probability at least $1 - \delta/4$, for all $\pi \in \Pi$,

$$\sum_{t=1}^T R_t(a_t, b_t) \geq \sum_{t=1}^T R_t(\pi(x_t), b_t) - O(\sqrt{KT \ln(|\Pi|/\delta)}).$$

Further, using Azuma’s lemma and union bound, we can show that, with probability at least $1 - \delta/4$, for every $\pi \in \Pi$

$$\sum_{t=1}^T R_t(\pi(x_t), b_t) \geq \sum_{t=1}^T P_t(\pi(x_t), b_t) - O(\sqrt{KT \ln(|\Pi|/\delta)}).$$

Similarly, from column-Exp’s perspective, with high probability, for all $\pi \in \Pi$,

$$-\sum_{t=1}^T R_t(a_t, b_t) \geq -\sum_{t=1}^T R_t(a_t, \pi(x_t)) - O(\sqrt{KT \ln(|\Pi|/\delta)})$$

and, by Azuma’s lemma and the skew-symmetry of $P_t$,

$$-\sum_{t=1}^T R_t(a_t, \pi(x_t)) \geq \sum_{t=1}^T P_t(\pi(x_t), a_t) - O(\sqrt{KT \ln(|\Pi|/\delta)}).$$

Combining and rearranging now yields the theorem. \hfill \blacksquare
Proof [Lemma 3] Eq. (6) implies that $W^T\hat{MU}$ is within $\epsilon'$ of $W^T MU$, for all probability vectors $W$ and $U$. Therefore,

$$\min_{U \in \Delta_{[N]}} \hat{W}^T MU \geq \min_{U \in \Delta_{[N]}} \hat{W}^T \hat{MU} - \epsilon'$$

$$\geq \max_{W \in \Delta_{[N]}} \min_{U \in \Delta_{[N]}} W^T \hat{MU} - \epsilon' - \epsilon$$

$$\geq \max_{W \in \Delta_{[N]}} \min_{U \in \Delta_{[N]}} W^T MU - 2\epsilon' - \epsilon$$

$$= -(2\epsilon' + \epsilon).$$

Proof [Theorem 4] Note that in our case, we can choose $D = \sqrt{m}$, $A = L\sqrt{m}$ and $R = L$.

Let $\bar{u} = \frac{1}{T} \sum_{t=1}^{T} u_t$. Then we have the following chain of inequalities holding with probability at least $1 - \delta$:

$$\min_{u \in C} \max_{w \in C} w^T Bu - \epsilon \leq \max_{w \in C} w^T B\bar{u} - \epsilon$$

$$\leq \frac{1}{T} \sum_{t=1}^{T} w_t^T Bu_t$$

(13)

$$\leq \min_{u \in C} w^T Bu + \epsilon$$

(14)

Here, Eqs. (13) and (14) follow directly from Eq. (10) applied, respectively, to row-FPL and column-FPL. Noting that

$$\max_{w \in C} \min_{u \in C} w^T Bu = \min_{u \in C} \max_{w \in C} w^T Bu,$$

the theorem now follows.

Proof [Lemma 5] For all $w \in C$, we have

$$\|w - w_{t+1}\|^2 - \|w - w_t\|^2 \leq \|w - z_{t+1}\|^2 - \|w - w_t\|^2 + \alpha \eta L$$

(15)

$$= -2\eta (w - w_t)^T Bu_t + \eta^2 \|Bu_t\|^2 + \alpha \eta L$$

(16)

Here, Eq. (15) uses Eq. (11), applied to our case where we have $z_{t+1} - w_t = \eta Bu_t$. Eq. (16) follows from straightforward algebra. Since $\|w - w_1\| \leq 2$, summing over $t = 1, \ldots, T_{out}$ yields, for all $w \in C$,

$$-4 \leq \|w - w_{T_{out}+1}\|^2 - \|w - w_1\|^2$$

$$\leq -2\eta \sum_{t=1}^{T_{out}} w^T Bu_t + 2\eta \sum_{t=1}^{T_{out}} w_t^T Bu_t + \eta^2 L^2 T_{out} + \alpha \eta L T_{out}.$$

Re-arranging completes the lemma.
**Proof [Theorem 6]** Let $\mathbf{u} = \frac{1}{T_{out}} \sum_{t=1}^{T_{out}} \mathbf{u}_t$. Then

$$
\max_{\mathbf{w} \in C} \min_{\mathbf{u} \in C} \mathbf{w}^\top \mathbf{B} \mathbf{u} \geq \min_{\mathbf{u} \in C} \mathbf{w}^\top \mathbf{B} \mathbf{u}
$$

$$
= \min_{\mathbf{u} \in C} \frac{1}{T_{out}} \sum_{t=1}^{T_{out}} \mathbf{w}_t^\top \mathbf{B} \mathbf{u}
$$

$$
\geq \frac{1}{T_{out}} \sum_{t=1}^{T_{out}} \min_{\mathbf{u} \in C} \mathbf{w}_t^\top \mathbf{B} \mathbf{u}
$$

$$
= \frac{1}{T_{out}} \sum_{t=1}^{T_{out}} \mathbf{w}_t^\top \mathbf{B} \mathbf{u}_t
$$

$$
\geq \max_{\mathbf{w} \in C} \frac{1}{T_{out}} \sum_{t=1}^{T_{out}} \mathbf{w}^\top \mathbf{B} \mathbf{u}_t - \varepsilon
$$

$$
= \max_{\mathbf{w} \in C} \mathbf{w}^\top \mathbf{B} \mathbf{u} - \varepsilon
$$

$$
\geq \min_{\mathbf{u} \in C} \max_{\mathbf{w} \in C} \mathbf{w}^\top \mathbf{B} \mathbf{u} - \varepsilon.
$$

**Proof [Lemma 7]** We have

$$
\|\mathbf{v}^* - \mathbf{v}_{t+1}\|^2 - \|\mathbf{v}^* - \mathbf{v}_t\|^2
$$

$$
= \|\mathbf{v}^* - \mathbf{v}_t + \nu (\mathbf{v}_t - \mathbf{s}_t)\|^2 - \|\mathbf{v}^* - \mathbf{v}_t\|^2
$$

$$
= 2\nu (\mathbf{v}^* - \mathbf{v}_t) \cdot (\mathbf{v}_t - \mathbf{s}_t) + \nu^2 \|\mathbf{v}_t - \mathbf{s}_t\|^2
$$

$$
\leq 2\nu (\mathbf{v}^* - \mathbf{v}_t) \cdot (\mathbf{v}_t - \mathbf{s}_t) + 4\nu^2
$$

$$
\leq \nu (F(\mathbf{s}_t, \mathbf{v}_t) - F(\mathbf{s}_t, \mathbf{v}^*)) + 4\nu^2.
$$

Eq. (17) uses $\|\mathbf{v}_t - \mathbf{s}_t\| \leq \|\mathbf{v}_t\| + \|\mathbf{s}_t\| \leq 2$. To see Eq. (18), note that

$$
2(\mathbf{v}^* - \mathbf{v}_t) \cdot (\mathbf{v}_t - \mathbf{s}_t)
$$

$$
= 2\mathbf{v}^* \cdot \mathbf{v}_t - 2\|\mathbf{v}_t\|^2 - 2\mathbf{v}^* \cdot (\mathbf{v}_t - \mathbf{s}_t)
$$

$$
\leq \|\mathbf{v}^*\|^2 + \|\mathbf{v}_t\|^2 - 2\|\mathbf{v}_t\|^2 - 2\mathbf{v}^* \cdot (\mathbf{v}_t - \mathbf{s}_t)
$$

$$
= [2\mathbf{s}_t \cdot (\mathbf{v}_t - \mathbf{z}) + \|\mathbf{z}\|^2 - \|\mathbf{v}_t\|^2]
$$

$$
- [2\mathbf{s}_t \cdot (\mathbf{v}^* - \mathbf{z}) + \|\mathbf{z}\|^2 - \|\mathbf{v}^*\|^2]
$$

$$
= F(\mathbf{s}_t, \mathbf{v}_t) - F(\mathbf{s}_t, \mathbf{v}^*).
$$

The inequality here uses the fact that, for any two vectors $\mathbf{u}$ and $\mathbf{w}$, we have $2\mathbf{u} \cdot \mathbf{w} \leq \|\mathbf{u}\|^2 + \|\mathbf{w}\|^2$.

Also,

$$
\|\mathbf{v}^* - \mathbf{v}_1\| \leq \|\mathbf{v}^* - \mathbf{z}\| + \|\mathbf{z} - \mathbf{v}_1\|
$$

$$
= \min_{\mathbf{v} \in C} \|\mathbf{v} - \mathbf{z}\| + \|\mathbf{z} - \mathbf{v}_1\|
$$

$$
\leq \|\mathbf{v}_1 - \mathbf{z}\| + \|\mathbf{z} - \mathbf{v}_1\|.
$$
Summing Eq. (18) for $t = 1, \ldots, T_{in}$ and combining with Eq. (19) gives

$$-4\|z - v_1\|^2 \leq \|v^* - v_{T_{in} + 1}\|^2 - \|v^* - v_1\|^2 \leq \nu \sum_{t=1}^{T_{in}} F(s_t, v_t) - \nu \sum_{t=1}^{T_{in}} F(s_t, v^*) + 4
\nu^2 T_{in}.$$  

Re-arranging and applying our choice of $\nu$ completes the lemma.

**Proof [Theorem 8]** Let

$$s = \frac{1}{T_{in}} \sum_{t=1}^{T_{in}} s_t.$$ 

Then

$$\min_{s \in \mathcal{C}} F(s, \nabla) \geq \min_{s \in \mathcal{C}} \frac{1}{T_{in}} \sum_{t=1}^{T_{in}} F(s, v_t) \geq \frac{1}{T_{in}} \sum_{t=1}^{T_{in}} \min_{s \in \mathcal{C}} F(s, v_t) = \frac{1}{T_{in}} \sum_{t=1}^{T_{in}} F(s_t, v_t) \geq \frac{1}{T_{in}} \sum_{t=1}^{T_{in}} F(s_t, v^*) - \delta = F(s, v^*) - \delta \geq \min_{s \in \mathcal{C}} F(s, v^*) - \delta \geq \|z - v^*\|^2 - \delta \geq -\delta.$$  

Eq. (20) uses Jensen’s inequality and the fact that $F(s, \cdot)$ is concave for each $s$. Eq. (21) follows from our choice of $s_t$ (which minimizes $F(\cdot, v_t)$). Eq. (22) uses linearity of $F(\cdot, v)$ for each $v$. And Eq. (23) uses $F(s, v^*) \geq \|z - v^*\|^2$ for all $s \in \mathcal{C}$, which follows from simple Euclidean geometry and the Pythagorean theorem.