IEICE TRANSACTIONS

on Fundamentals of Electronics, Communications and Computer Sciences

DOI:10.1587/transfun.2024EAP1078

Publicized:2025/01/31

A PUBLICATION OF THE ENGINEERING SCIENCES SOCIETY
The Institute of Electronics, Information and Communication Engineers
Kikai-Shinko-Kaikan Bldg., 5-8, Shibakoen 3 chome, Minato-ku, TOKYO, 105-0011 JAPAN

PAPER

Identification of the Strongest Die in Dueling Bandits

Shang LU[†], **Kohei HATANO**[†,††], **Shuji KIJIMA**[†††], ***and*** **Eiji TAKIMOTO**[†],

SUMMARY This work introduces the *dueling dice problem*, which is a variant of the multi-armed dueling bandit problem. A *die* is a set of m arms in this problem, and the goal is to find the best set of m arms from n arms ($m \leq n$) by an iteration of dueling dice. In a round, the learner arbitrarily chooses two dice $\alpha \subseteq [n]$ and $\beta \subseteq [n]$ and lets them duel, where she roles dice α and β, observes a pair of arms $i \in \alpha$ and $j \in \beta$, and receives a probabilistic result $X_{i,j} \in \{0, 1\}$. This paper investigates the sample complexity of an identification of the Condorcet winner die, and gives an upper bound $\mathrm{O}(nh^{-2}(\log\log h^{-1} + \log nm^2\gamma^{-1})m\log m)$ where h is a gap parameter and γ is an error parameter. Our problem is closely related to the dueling teams problem by Cohen et al. 2021. We assume a total order of the strength over arms similarly to Cohen et al. 2021, which ensures the existence of the Condorcet winner die, but we *do not* assume a total order of the strength over dice unlike Cohen et al. 2021.

key words: *Condorcet winner die, dueling bandit problem*

1. Introduction

In the realms of mathematics and computer science, the exploration of dice models has perennially captivated researchers [2], [3], [8], [12], [17]–[20]. Among the intriguing topics within this domain is the investigation into the transitivity of dice. For instance, if die A is favored over die B, and die B over die C, does it necessarily imply that die A prevails over die C? Surprisingly, the answer is negative. This phenomenon has spurred extensive research, particularly in the realm of applied probabilities, leading to the exploration of non-transitive dice relationships.

The concept of non-transitive relations among stochastic events found its roots in diverse fields, notably in the pioneering work of Black [2] in 1958 on the community voting problem, followed by Usiskin's [20] establishment of upper bounds for winning probabilities in certain non-transitive scenarios. Gardner [8] furthered this inquiry by associating non-transitivity with Efron's dice, thus initiating a broader investigation into non-transitive dice models. Subsequently, Savage [17] introduced a dice set defined by a regular partition. Later Schaefer and Schweig [19] and Schaefer [18] used the regular partition to establish the existence of corresponding regular partition dice sets for arbitrary tournaments. Buhler et al. [3] demonstrated that a magnitude relationship comprising repetitive dice could encompass all tournament graphs. Conrey et al. [6] employed statistical methodologies to generate numerous dice groups, estimating the proportion of non-transitive dice sets. Akin [1] provided an alternative method, showcasing the existence of regular partition dice sets obtainable in any tournament. Lu and Kijima [12], motivated by a design or decision of games with a *probabilistic* win-loss, investigated the conditions that a set of dice is non-transitive under some fairness conditions, as well as the conditions of the existence of the strongest die.

The *multi-armed bandit* problem is a popular topic in probability theory and machine learning, motivated by optimal decision-making in the exploration-exploitation tradeoff dilemma. In particular, the stochastic multi-armed bandit problem, featuring pairwise comparisons as actions, has been extensively explored under the framework of "dueling bandits" cf. [22]. Yue et al. [21] established a regret lower bound of $\Omega(n \log T)$ for the n-armed dueling bandit problem of T rounds. Komiyama et al. [11] further analyzed this lower bound, and determined the optimal constant factor for models assuming the existence of the best arm, which is often referred to as the *Condorcet winner arm*. They also proposed an algorithm with an upper bound of $\mathrm{O}(n \log T)$ to minimize regret when seeking the Condorcet winner arm.

Identification of the best arm is a central issue in the context. Haddenhorst et al. [9] addressed the best arm identification problem and delineated the relationship between the number of attempts and error as $\mathrm{O}(n \log nh^{-2}(\log\log h^{-1} + \log\gamma^{-1}))$ where h is a gap parameter and γ is an error parameter. Kalyanakrishnan and Stone, [10] initiated the study of best-m arm identification, which is based on observations from real-value. Mohajer et al. [15] provide an algorithm that returns the best-m arms with probability exceeding $1 - (\log n)^{-c_0}$ achieving the sample complexity $c_1\Delta_{m,m+1}^{-2}(n + m\log m)\max(\log\log n, \log m)$, where c_0 and c_1 are universal positive constants and $\Delta_{m,m+1}$ is the distinguishability between the m and the $(m + 1)$ best arms. Ren et al. [16] devised an algorithm for the best-m arms identification, resolving the problem with sample complexity $\mathrm{O}(nh^{-2}(\log\log h^{-1} + \log n\gamma^{-1}))$ by selecting arm pairs and evaluating experiences between them. Building on this, Cohen et al. [5] treated the set of arms as a team, identifying the optimal team through team pairs under the assumption of total order for both arms and teams. Leveraging this assumption, they equated the problem of finding the Condorcet winner team with that of finding the Condorcet winner arm, resulting in a sample complexity akin to best arm identification, namely $\mathrm{O}((n + m\log m)h^{-2}\max(\log\log n, \log m))$.

Related studies include the work of Chen et al [4], which addressed the problem of matching multiple candidates to multiple positions using the dueling bandits framework. In

[†] Department of Informatics, Kyushu University
[††] RIKEN AIP
[†††] Faculty of Data Science, Shiga University

this approach, a bipartite graph is employed, and in each round, two candidates are compared in a duel for a specific position, with the outcome revealing which candidate wins. The goal is to identify the optimal candidate-position matches with high probability after multiple rounds of sampling. Maiti et al. [13] worked within the dueling bandits framework. They established both upper and lower bounds on the sample complexity required to compute the Nash equilibrium for an unknown $2 \times n$ two-player zero-sum payoff matrix.

We here briefly review Cohen et al. [5] about the dueling team problem, which is closely related to this work. The dueling team problem is an online-learning problem where the learner observes noisy comparisons of disjoint pairs of k-sized teams from a universe of n players and finds the Condorcet winning team which wins against any other disjoint team with probability at least $1/2$. Cohen et al. [5] assume two major conditions of strong stochastic transitivity and consistency: roughly speaking, the strong stochastic transitivity provides a total order over teams regarding the probabilistic win-lose relationship and the consistency provides a total order over players regarding the contribution to any team. Then, they gave an upper bound $O((n + k \log k)\frac{\max(\log\log n, \log k)}{\Delta^2})$ of the number of duels required to identify the Condorcet winning team with high probability, where Δ is some parameter about a probability gap.

(1) Our result.

This paper introduces the *dice dueling problem* (see Section 2 in detail), which is a variant of the multi-armed bandit problem. A *dice* is a set of m arms, and the goal is to select the best m [arms], which we call Condorcet winner die (see Section 2.2 for the definition), out of n arms ($m \leq n$) by dice duels. In each round, the learner arbitrarily chooses two dice $\alpha \subseteq \{1, \ldots, n\}$ and $\beta \subseteq \{1, \ldots, n\}$, and lets them duel: she roles the dice α and β, observes a pair of arms $i \in \alpha$ and $j \in \beta$, and receives a stochastic result $X_{i,j} \in \{0, 1\}$ according to the Bernoulli distribution with expectation $\mu_{i,j}$. The learner repeats rounds by choosing adequate pairs of dice and tries to find the best die. Of course, the learner knows in advance neither the Condorcet winner die in the possible $\binom{n}{m} = O(n^m)$ dice, nor the win-lose relations among the n arms, i.e., whether $\mu_{i,j} > 1/2$ or not. In this paper, a pair of dice α and β is allowed to duel even when $\alpha \cap \beta \neq \emptyset$ or $\alpha = \beta$. While the target of Cohen et al. [5] may be considered as a design of a professional sports team, our target could be associated with a deck-building game where the same items can face each other. We also remark that the learner in [5] observes only which team won while the learner of this paper observes up to the resulting casts, meaning that Cohen et al. [5] stands on a weaker assumption concerning the learner's observation.

According to the literature, we assume that $\mu_{i,j}$ is transitive, meaning that there is a total order over the n arms concerning the win-lose relationship, see e.g., [5]. This assumption allows the Condorcet winner die to exist: without loss of generality, we may assume $\mu_{i,j} > 1/2$ for any $i < j$, then, $\{1, \ldots, m\} \subseteq [n]$ is clearly the best die. We remark that the win-lose relationship among the dice are non-transitive in general, in contrast to Cohen et al. [5] assumes a total order over teams (corresponding to dice) and the strong stochastic transitivity among teams, which are indispensable assumptions in their algorithm and analysis.

This paper presents an algorithm for the identification of the best-m arms by dueling dice, and proves its sample complexity is $O(nh^{-2}(\log\log h^{-1} + \log nm^2\gamma^{-1})m \log m)$, that is a polynomial in n and m while the number of dice is $O(n^m)$. Our algorithm is essentially based on the simple Monte Carlo technique, and we prove the accuracy of our algorithm's output by using the Hoeffding inequality, similar to [5]. For an improvement of the sample complexity, we also employ a technique of the classical *selection algorithm* in a similar way as [16].

2. Preliminary

2.1 Stochastic n-armed dueling bandit problem

As a preliminary step, we introduce some terminology and notations about the stochastic n-armed dueling bandit problem. The stochastic n-armed *dueling* bandit problem involves $n \in \mathbb{N}$ arms. Let $[n] = \{1, 2, \ldots, n\}$ denote the set of arms. The learner chooses a pair of arms $i, j \in [n]$ and lets them duel, where i is allowed to duel with i in this paper. As a result of a duel, the learner receives a relative feedback $X_{ij} \in \{0, 1\}$ which follows the Bernoulli distribution with expectation μ_{ij}. We say i is *preferable* to j, denoted by $i \succ j$, if $\mu_{ij} > 1/2$. Notice that $\mu_{ij} = 1 - \mu_{ji}$ holds for any $i, j \in [K]$, and that $\mu_{ii} = 1/2$. Let $M = (\mu_{ij}) \in \mathbb{R}^{K \times K}$ denote the preference matrix over n.

In this paper, we assume that the preference relation is transitive, meaning that $\mu_{ij} > 1/2$ and $\mu_{jk} > 1/2$ imply $\mu_{ik} > 1/2$, thus $\succ$ is a total order on $[n]$. Furthermore, we assume that there exists $h > 0$ such that $|\mu_{ij} - \mu_{ji}| \geq h$ holds for any distinct i and j.

2.2 Dueling dice problem

Here, we define the *dueling dice problem* with the help of the terminology of n-armed dueling bandit problem described above. An *m-sided die* (or simply a *die*) is a subset of arms $[n]$ with m elements, in this paper. For convenience, let $\mathcal{D}$ denote the set of all dice, thus $|\mathcal{D}| = \binom{n}{m} = O(n^m)$. this paper. *Rolling a die* $\alpha \subseteq [n]$ is an action to choose an element of α uniformly at random.

In the dueling dice problem, the learner repeats rounds in each of which she arbitrarily chooses a pair of dice $\alpha, \beta \in \mathcal{D}$ and lets them duel; the learner rolls dice α and β, observes the casts $I \in \alpha$ and $J \in \beta$, and receives a relative feedback $X_{IJ} \in \{0, 1\}$ according to the Bernoulli distribution with expectation μ_{IJ}. We particularly remark that α and β can be overlapped, i.e., $\alpha \cap \beta \neq \emptyset$ is allowed for a dice duel. Then, it is not difficult to see that

$$\mathbb{P}(I \succ J) = \frac{1}{m^2} \sum_{i \in \alpha} \sum_{j \in \beta} \mu_{i,j} \tag{1}$$

holds. We say α is *preferable to* β, denoted by $\alpha \succ \beta$, if $\mathbb{P}(I \succ J) > \frac{1}{2}$. The learner does not know neither the value of $\mu_{i,j}$ nor whether $I \succ J$, in advance.

We say a die $\alpha \in \mathcal{D}$ is the *Condorcet winner* die if $\alpha \succ \beta$ holds for any $\beta \in \mathcal{D} \setminus \{\alpha\}$. Since we assume a total order over arms $[K]$, it is not difficult to see that $\{1, \ldots, m\}$ is the unique Condorcet winner die.

Definition 2.1 (dueling dice problem): Let $\gamma \in (0, 1)$. Given a set of n arms, identify the correct Condorcet winner die with probability at least $1 - \gamma$ where the gap parameter $h = \min_{ij \in \binom{[n]}{2}} |\mu_{ij} - \mu_{ji}|$ is unknown.

In this paper, we do not adopt the common assumptions typically used in Dueling Bandit settings, such as Strong Stochastic Transitivity (SST), as the intransitivity associated with such assumptions is widely observed in dice games [12]. For example, using the construction method in [15] to design three 5-sided dice, we can easily identify $A = \{2, 6, 7, 11, 14\}$, $B = \{1, 5, 8, 12, 15\}$,$C = \{3, 4, 9, 10, 13\}$, which satisfy the relationships $P(A > B) > 1/2$, $P(B > C) > 1/2$, $P(C > A) > 1/2$.

3. Best Die Identification Algorithm

In this section, we propose an algorithm for the dueling dice problem, and provide an upper bound of the sample complexity. Algorithm 1 shows the main body of our algorithm for best die identification (BDI), which calls Algorithm 2 at lines 4, 14, 15 and Algorithm 3 at line 16 as subroutines. Algorithm 4 is called at lines 5 and 9 in Algorithm 3 as a subroutine. In Algorithm 1, an element $N_{i,j}$ of the matrix N denotes the number of comparisons of pair arms i and j where $N_{i,j} = N_{j,i}$ and element $K_{i,j}$ of the matrix K denotes the number of events that i is preferred to j. An element $\hat{\mu}_{i,j}$ of the matrix $\hat{\mu}$ is given by $\hat{\mu}_{i,j} = \frac{K_{i,j}}{N_{i,j}}$ where we set $0/0 = 1/2$. Then we establish the following theorem for BDI.

Roughly speaking, Algorithm 1 chooses a candidate of the strongest die $\alpha_t \subseteq [n]$ for $t = 1, 2, \ldots$, and updates it by replacing with some members of $\beta_t \subset [n] \setminus \alpha_t$. For the updates, Algorithm 1 estimates whether $\mu_{ij} > 1/2$ for $i, j \in [n]$ by dueling dice at lines 4, 14, 15, where Algorithm 2 is called as a subroutine to determine with high probability whether $\mu_{ij} > 1/2$ for any pair $(i, j) \in \alpha \times \beta$ for the input pair of dice α and β. Then, Algorithm 1 updates the candidate from α_t to $\alpha_{t+1} \subseteq \alpha_t \cup \beta_t$ at line 16, where Algorithm 3 is called as a subroutine to find best m-arms in $\alpha \cup \beta$ according to (an estimated) μ_{ij} for $i, j \in \alpha_t \cup \beta_t$.

Theorem 3.1: Given a set of arms $[n]$, the number of faces of a dice m $(1 \le m \le n)$, and a parameter γ $(0 < \gamma < 1)$ as an input, BDI terminates after $\mathrm{O}(nh^{-2}(\log\log h^{-1} + \log nm^2\gamma^{-1})m \log m)$ rounds of dice duels in expectation, and returns an optimal Condorcet winner die with probability at least $1 - \gamma$.

Algorithm 1 Best Die Identification (BDI)

```
Input: $n$ arms, $m \in \mathbb{N}, \gamma \in (0, 1)$.
Output: $\alpha \subseteq [n]$.
1: Set $R_0 \leftarrow [n], \gamma' \leftarrow \frac{\gamma}{60n}$.
2: Set $N \leftarrow (0) \in \mathbb{Z}_{\ge 0}^{n \times n}, K \leftarrow (0) \in \mathbb{Z}_{\ge 0}^{n \times n}, \hat{\mu} \leftarrow (1/2) \in [0, 1]^{n \times n}$.
3: Arbitrarily choose $\alpha_1 \subseteq R_0$ such that $|\alpha_1| = m$.
4: DD($\alpha_1, \alpha_1, \frac{\gamma'}{m^2}, N, K$).
5: $R_1 \leftarrow R_0 \setminus \alpha_1$.
6: Set $t \leftarrow 1$.
7: while $R_t \neq \emptyset$ do
8:     if $|R_t| \ge m$ then
9:         Arbitrarily choose $\beta_t \subseteq R_t$ such that $|\beta_t| = m$.
10:    else
11:        Arbitrarily choose $R' \subseteq \alpha_t$ such that $|R'| = m - |R_t|$,
12:        $\beta_t \leftarrow R_t \cup R'$.
13:    end if
14:    DD($\beta_t, \beta_t, \frac{\gamma'}{m^2}, N, K$).
15:    DD($\alpha_t, \beta_t, \frac{\gamma'}{m^2}, N, K$).
16:    $\alpha_{t+1} \leftarrow$ DMM($\alpha_t, \beta_t, \hat{\mu}, m$).
17:    $R_{t+1} \leftarrow R_t \setminus \beta_t$.
18:    $t \leftarrow t + 1$.
19: end while
20: return $\alpha_t$.
```

Algorithm 2 Dice Dueling Algorithm DD($\alpha, \beta, \gamma, N, K$)

```
Set $s \leftarrow 1, h_s \leftarrow 2^{-s-1}, \gamma_s \leftarrow \frac{6\gamma}{\pi^2 s^2}$.
while $\exists (i', j') \in \alpha \times \beta$ such that $i' \neq j'$ and $\left|\frac{K_{i',j'}}{N_{i',j'}} - \frac{K_{j',i'}}{N_{j',i'}}\right| < h_s$ do
    $Q_s \leftarrow \frac{25}{h_s^2} \log \frac{2}{\gamma_s}$.
    while $\exists (i'', j'') \in \alpha \times \beta$ such that $i'' \neq j''$ and $N_{i'',j''} < Q_s$ do
        Roll $(\alpha, \beta)$, observe $(i, j)$, and receive $X_{i,j} \in \{0, 1\}$.
        if $i \neq j$ then
            $N_{i,j} \leftarrow N_{i,j} + 1, K_{i,j} \leftarrow K_{i,j} + X_{i,j}$.
        end if
    end while
    $s \leftarrow s + 1$.
end while
```

3.1 Analysis of Algorithm 2

To prove Theorem 3.1, this section analyzes Algorithm 2 and establishes Lemma 3.2 below. Algorithm 2 is a subroutine of Algorithm 1 designed to determine with high probability whether $\mu_{ij} > 1/2$ for any elements within an input pair of dice α and β, updating the total numbers N_{ij} of duels between $i, j \in [n]$ and their results K_{ij}.

Algorithm 2 repeats dice duels so as to estimate $\min_{(i,j) \in \alpha \times \beta} |\mu_{ij} - \mu_{ji}|$. On condition that the estimated $\min_{(i,j) \in \alpha \times \beta} |\mu_{ij} - \mu_{ji}|$ is at least $h_s = 2^{-s-1}$, we prove that $\mu_{ij} > 1/2$ is correctly determined with high probability at least $1 - \gamma_s$ where $\gamma_s = \frac{6\gamma}{\pi^2 s^2}$ if every pair $i, j \in \alpha \times \beta$ is compared at least $Q_s = \frac{25}{h_s^2} \log \frac{2}{\gamma_s}$ times in total.

Lemma 3.2: Given α, β and $\gamma \in (0, 1)$, Algorithm 2 terminates after $\mathrm{O}(h^{-2}(\log\log h^{-1} + \log \gamma^{-1})m^2 \log m)$ rounds of dice dueling in expectation, and $\mathbb{1}\{\hat{\mu}_{i,j} > 1/2\} = \mathbb{1}\{\mu_{i,j} > 1/2\}$ holds for any $i \in \alpha$, $j \in \beta$ satisfying $i \neq j$ with proba-

Algorithm 3 Dice Median of Median Algorithm DMM(α,β,μ,m)

```
$R = \alpha \cup \beta$.
Set $S_{up} \leftarrow \emptyset$, $S_{down} \leftarrow \emptyset$, $median \leftarrow \emptyset$.
Split $R$ into $L = \lceil \frac{|R|}{5} \rceil$ sets ($S_i, i \in [L]$).
for $i = 1$ to $L$ do
    $S_i \leftarrow$ SelectionSort($S_i, \mu$).
    $median \leftarrow median \cup S_i[3]$.
end for
if $|median| \le 5$ then
    $median \leftarrow$ SelectionSort($median, \mu$).
    $a \leftarrow median[\frac{|median|}{2} + 1]$.
else
    $median \leftarrow$ DMM($median, \emptyset, \mu, \frac{|median|}{2} + 1$).
    $a \leftarrow median[\frac{|median|}{2} + 1]$
end if
for each $j \in R \setminus \{a\}$ do
    if $\mu_{j,a} > 1/2$ then
        $S_{up} \leftarrow j$
    else
        $S_{down} \leftarrow j$
    end if
end for
if $|S_{up}| > m$ then
    return DMM($S_{up}, \emptyset, \mu, m$).
else if $|S_{up}| = m$ then
    return $S_{up}$.
else if $|S_{up}| = m - 1$ then
    return $S_{up} \cup \{a\}$.
else
    return $S_{up} \cup \{a\} \cup$ DMM($S_{down}, \emptyset, \mu, m - |S_{up}| - 1$).
end if
```

Algorithm 4 SelectionSort(S, μ)

```
for $i \leftarrow 1$ to $|S|$ do
    $l \leftarrow i$.
    $a \leftarrow S[i]$.
    for $j \leftarrow i + 1$ to $|S|$ do
        if $\mu_{i,j} < \frac{1}{2}$ then
            $l \leftarrow j$.
            $a \leftarrow S[j]$.
        end if
    end for
    $S[l] \leftarrow S[i]$.
    $S[i] \leftarrow a$.
end for
return $S$.
```

bility at least $1 - \gamma$.

Proof : First, we prove the correctness of the algorithm, and then we discuss its sample complexity.

(1) Correctness:

Let $h^* = \min\{|\mu_{i,j} - \mu_{j,i}| \mid (i,j) \in \alpha \times \beta, \text{ and } i \ne j\}$. Suppose for $(i,j) \in \alpha \times \beta$ that $|\mu_{i,j} - \mu_{j,i}| = h^*$ holds.

To begin with, we claim for s satisfying $h_s \ge 2h^*$ that Algorithm 2 (DD for short) terminates with probability at most γ_s, i.e., the algorithm proceeds the while loop unless $h_s < 2h^*$ with high probability. By Lemma 3.3, appearing below,

$$\mathbb{P}\left(|\mu_{i,j} - \hat{\mu}_{i,j}| \le \frac{h_s}{5}\right) \ge 1 - \gamma_s \tag{2}$$

holds for any s. When $|\mu_{i,j} - \hat{\mu}_{i,j}| \le h_s/5$ holds, we see

$$\begin{aligned} |\hat{\mu}_{i,j} - \hat{\mu}_{j,i}| &= \left|(\hat{\mu}_{i,j} - \mu_{i,j}) + (\mu_{j,i} - \hat{\mu}_{j,i}) + (\mu_{i,j} - \mu_{j,i})\right| \\ &\le 2\left|\hat{\mu}_{i,j} - \mu_{i,j}\right| + \left|\mu_{i,j} - \mu_{j,i}\right| \\ &\le \tfrac{2}{5}h_s + h^* \\ &\le \tfrac{9}{10}h_s \end{aligned} \tag{3}$$

holds where the last inequality follows from the assumption that $h^* < h_s/2$. (3) contradicts to the condition of a termination, that is $|\hat{\mu}_{i,j} - \hat{\mu}_{j,i}| \ge h_s$, meaning that the algorithm goes to the next iteration with probability at least $1 - \gamma_s$ by (2).

Next, we claim for s satisfying $h_s < 2h^*$ that $\mathbb{P}(\hat{\mu}_{i,j} < \hat{\mu}_{j,i}) < \gamma_s$ holds if $\mu_{i,j} > \mu_{j,i}$, i.e., the probability of a misjudgement is small. Notice that $\mu_{i,j} - \mu_{j,i} > h^*$ and $\mu_{i,j} + \mu_{j,i} = 1$ imply $\mu_{i,j} > \frac{1}{2} + \frac{h^*}{2}$. Then, we have

$$\begin{aligned} \mathbb{P}\left(|\mu_{i,j} - \hat{\mu}_{i,j}| > \tfrac{h_s}{5}\right) &\ge \mathbb{P}\left(\mu_{i,j} - \hat{\mu}_{i,j} > \tfrac{h_s}{5}\right) \\ &= \mathbb{P}\left(\hat{\mu}_{i,j} < \mu_{i,j} - \tfrac{h_s}{5}\right) \\ &\ge \mathbb{P}\left(\hat{\mu}_{i,j} < \tfrac{1}{2} + \tfrac{h^*}{2} - \tfrac{h_s}{5}\right) \\ &\ge \mathbb{P}\left(\hat{\mu}_{i,j} < \tfrac{1}{2}\right) \end{aligned} \tag{4}$$

where the last inequality follows $h_s < 2h^*$. By (2) and (4), we obtain $\mathbb{P}\left(\hat{\mu}_{i,j} < \frac{1}{2}\right) \le \gamma_s$, that is nothing but the claim of this paragraph since $\hat{\mu}_{i,j} + \hat{\mu}_{j,i} = 1$. It could be obvious that $\mathbb{P}(\hat{\mu}_{i,j} > \hat{\mu}_{j,i}) < \gamma_s$ holds for s satisfying $h_s < 2h^*$ if $\mu_{i,j} < \mu_{j,i}$, similarly.

Lastly, let $s^* = \max\{s \mid h_s \ge 2h^*\}$ then

$$\begin{aligned} &\mathbb{P}(\mathbb{1}\{\mu_{i,j} > 1/2 \ne \mathbb{1}\{\hat{\mu}_{i,j} > 1/2\}\}) \\ &< \sum_{s=1}^{s^*} \mathbb{P}(\text{DD terminates}) + \sum_{s=s^*+1}^{\infty} \mathbb{P}(\text{DD misjudges}) \\ &< \sum_{s=1}^{\infty} \gamma_s = \sum_{s=1}^{\infty} \frac{6\gamma}{\pi^2 s^2} \le \gamma \end{aligned}$$

hold where the second last equality follows the sum of the reciprocals of the positive square integers. We obtain the correctness of the algorithm.

(2) Sample complexity:

Let Z be a random variable denoting the total number of dueling dice α and β in DD, then, our goal is to prove $\mathbb{E}[Z] = \mathrm{O}(m^2 \log(m) h^{-2}(\log\log h^{-1} + \log\gamma^{-1}))$. Notice that the termination condition of DD is both $N_{i,j} \ge Q_s$ and $\left|\frac{K_{i,j}}{N_{i,j}} - \frac{K_{j,i}}{N_{j,i}}\right| < h_s$ holds for any $(i,j) \in \alpha\times\beta$. By the correctness proof, we know that the condition of $\left|\frac{K_{i,j}}{N_{i,j}} - \frac{K_{j,i}}{N_{j,i}}\right| < h_s$ holds with high probability for $s > s^*$. Then, we are mainly concerned with the condition that $N_{i,j} \ge Q_s$ holds for all

$(i, j) \in \alpha \times \beta$.

Let $Y(z) = (Y_1(z), \dots, Y_{m^2}(z))$ for $z = 0, 1, \dots$ be a Markovian process defined as follows; Let $Y(0) = \mathbf{0}$ and $Y(z+1)$ is stochastically determined from $Y(z)$ where choose $L \in [m^2]$ uniformly at random and set

$$Y_l(z+1) = \begin{cases} Y_l(z) + 1 & (l = L) \\ Y_l(z) & \text{(otherwise).} \end{cases}$$

Let

$$Z'(x) = \min\{z \mid \forall l \in [m^2], Y_l(z) \geq x\}$$

for $x \in \mathbb{Z}_{>0}$. Then, it is not difficult to see that

$$\mathbb{E}[Z] \leq \mathbb{E}[Z'(Q_S)] \tag{5}$$

holds where S is a random variable denoting the value of s when DD terminates. Thus, we are concerned with an upper bound of $\mathbb{E}[Z'(Q_S)]$. We remark for a fixed $x \in \mathbb{Z}_{>0}$ that

$$\mathbb{E}[Z'(x)] \leq \sum_{x'=1}^{x} cm^2 \log m = cxm^2 \log m \tag{6}$$

holds with some constant c, which follows the x times repetition of the coupon collector (cf. [14]).

It is not difficult to observe that

$$\mathbb{E}[Z'(Q_S)] = \sum_{s=1}^{\infty} \mathbb{E}[Z'(Q_s) \mid S = s]\mathbb{P}(S = s) \tag{7}$$

holds. For convenience, let

$$\phi(s) = \mathbb{E}[Z'(Q_s) \mid S = s] = \mathbb{E}[Z'(Q_s)]$$

then,

$$\begin{aligned}(7) &= \sum_{s=1}^{\infty} \phi(s)\mathbb{P}(S = s) \\ &= \sum_{s=1}^{s^*} \phi(s)\mathbb{P}(S = s) + \sum_{s=s^*+1}^{\infty} \phi(s)\mathbb{P}(S = s) \\ &\leq \phi(s^*) + \sum_{s=s^*+1}^{\infty} \phi(s)\mathbb{P}(S = s) \qquad (8)\end{aligned}$$

holds. For any *fixed s*, we see that

$$\begin{aligned}\phi(s) &= \mathbb{E}[Z'(Q_s) \mid S = s] \\ &\leq cQ_s m^2 \log m \qquad \text{(by (6))} \\ &= c(25h_s^{-2} \log \tfrac{2}{\gamma_s})m^2 \log m \\ &\quad (\text{since } Q_s = 25h_s^{-2} \log \tfrac{2}{\gamma_s}, \text{ cf. Alg. 2}) \\ &= c'4^s \log(\tfrac{\pi^2 s^2}{3\gamma})m^2 \log m \\ &\quad (\text{since } h_s = 2^{-s-1} \text{ and } \gamma_s = \tfrac{6\gamma}{\pi^2 s^2}, \text{ cf. Alg. 2}) \\ &= c'4^s(2 \log s + \log \gamma^{-1} + d)m^2 \log m \qquad (9)\end{aligned}$$

holds with some constants c' and d (precisely $c' = 100c$ and $d = \log(\frac{\pi^2}{3})$). Notice that $2h^* \leq h_{s^*}$ and $h \leq h^*$ imply that $s^* < -\log_2 2h$. Thus, for the first term of (8),

$$\begin{aligned}\phi(s^*) &\leq c'4^{-\log_2 2h+1}(2 \log(-\log_2 2h) + \log \gamma^{-1} + d)m^2 \log m \\ &= c'4h^{-2}(2 \log \log_2(2h)^{-1} + \log \gamma^{-1} + d)m^2 \log m \\ &= \mathrm{O}(h^{-2}(\log \log h^{-1} + \log \gamma^{-1})m^2 \log m) \qquad (10)\end{aligned}$$

holds.

Concerning the second term of (8), we remark for $s \geq s^* + 1$ that

$$\begin{aligned}\mathbb{P}(S = s) &\leq (1 - \gamma_{s^*})\gamma_{s^*}^{s-s^*-1} \qquad \text{(by (2))} \\ &\leq \gamma_{s^*}^{s-s^*-1} \\ &\leq \left(\frac{6\gamma}{\pi^2}\right)^{s-s^*-1} \\ &\quad (\text{since } \gamma_{s^*} = \tfrac{6\gamma}{\pi^2 s^{*2}}, \text{ cf. Alg. 2. }) \\ &\leq 0.2^{s-s^*-1} \qquad (11)\end{aligned}$$

holds, where the last inequality follows the assumption that $\gamma < 1/4$. Then, the second term of (8) is bounded by

$$\begin{aligned}&\sum_{s=s^*+1}^{\infty} \phi(s)\mathbb{P}(S = s) \\ &\leq \sum_{s=s^*+1}^{\infty} c'4^s(2 \log s + \log \gamma^{-1} + d)m^2 \log(m)0.2^{s-s^*-1} \\ &\text{(by (9) and (11))} \\ &= c'4^{s^*+1}m^2 \log m \sum_{s=s^*+1}^{\infty} (2 \log s + \log \gamma^{-1} + d)0.8^{s-s^*-1}. \qquad (12)\end{aligned}$$

Let

$$g(s) = (2 \log s + \log \gamma^{-1} + d)0.8^{s-s^*-1}$$

for $s = s^* + 1, s^* + 2, \dots$, then

$$\begin{aligned}g(s) &\leq \frac{\log s}{\log s^*}(2 \log s^* + \log \gamma^{-1} + d)0.8^{s-s^*-1} \\ &\leq \left(\frac{0.9}{0.8}\right)^{s-s^*} (2 \log s^* + \log \gamma^{-1} + d)0.8^{s-s^*-1} \\ &= 1.25(2 \log s^* + \log \gamma^{-1} + d)\frac{0.9^{s-s^*}}{0.8}\end{aligned}$$

for $s \geq s^* + 1$ when $s^* \geq 5$. Then,

$$\begin{aligned}(12) &= c'4^{s^*+1}m^2 \log m \sum_{s=s^*+1}^{\infty} g(s) \\ &\leq 1.25c'4^{s^*+1}m^2 \log m(2 \log s^* + \log \gamma^{-1} + d) \sum_{s=s^*+1}^{\infty} 0.9^{s-s^*} \\ &\leq 1.25c'4^{s^*+1}m^2 \log m(2 \log s^* + \log \gamma^{-1} + d)\frac{1}{1 - 0.9} \qquad (13)\end{aligned}$$

holds. Since $s^* < -\log_2 2h$,

$$
\begin{aligned}
(13) &\leq 12.5c'4^{-\log_2 2h+1}m^2 \log m(2\log(-\log_2 2h)+\log\gamma^{-1}+d)\\
&= 12.5c'(4h^{-2})m^2(\log m)(2\log\log_2(2h)^{-1}+\log\gamma^{-1}+d)\\
&= \mathrm{O}(h^{-2}(\log\log h^{-1}+\log\gamma^{-1})m^2\log m) \qquad (14)
\end{aligned}
$$

holds. By (5) and (8) with (10) and (14), we obtain the claim.

Lemma 3.3: For lines 4 to 9 in Algorithm 2, if $N_{i,j} \geq Q_s$ holds for $i \in \alpha$, $j \in \beta$ satisfying $i \neq j$ and $s \in \mathbb{N}$ then $\mathbb{P}\left(|\mu_{i,j} - \hat{\mu}_{i,j}| > \frac{h_s}{5}\right) \leq \gamma_s$ holds.

Proof : Suppose $N_{i,j} \geq Q_s$. Let $X_{i,j}(1), \ldots, X_{i,j}(Q_s) \in \{0,1\}^{Q_s}$ denote the history of duels between dice i and j. Let $\hat{\mu}_{i,j} = \frac{\sum_{k=1}^{Q_s} X_{i,j}(k)}{Q_s}$, then

$$\mathbb{P}\left(|\hat{\mu}_{i,j} - \mu_{i,j}| > \frac{h_s}{5}\right) \leq 2e^{-Q_s(h_s/5)^2} = \gamma_s$$

holds by the Hoeffding's inequality (see Appendix). We obtain the claim.

3.2 Analysis of Algorithm 3

Algorithm 3 (DMM for short) selects the best m arms from $\alpha \cup \beta$. To reduce the sample complexity concerning γ' given at line 1 in Algorithm 1, we employ the selection algorithm in a similar way as [16]. Since an input $\hat{\mu}$ of Algorithm 3 may not be transitive, we employ the selection sort μ at lines 5 and 9 calling a subroutine Algorithm 4 which definitely? terminates even for a nontransitive μ.

Lemma 3.4: Given α, β, μ and m, Algorithm 3 outputs the best m arms from $\alpha \cup \beta$, and it requires at most $60m$ comparisons.

Proof : The proof consists of two parts: the proof of the correctness and the proof of the sample complexity.

(1) Correctness:

Suppose that $\mathbb{1}\{\hat{\mu}_{i,j} > 1/2\} = \mathbb{1}\{\mu_{i,j} > 1/2\}$ for any $i, j \in \alpha \cup \beta$, which holds with high probability, as we prove later. In the case, we claim that DMM outputs the best m arms. DMM consists of three steps:

Step 1. Lines between 4 and 14 find the median of median as a "pivot."

Step 2. Lines between 15 and 21 partitions the set $\alpha \cup \beta$ into subsets S_{up} and S_{down}. Then each of S_{up} and S_{down} has a size at most $\frac{7}{10}|\alpha \cup \beta|$. Clearly, $\mu_{x,a} > 1/2$ for any $x \in S_{\mathrm{up}}$, and $\mu_{y,a} < 1/2$ for any $y \in S_{\mathrm{down}}$.

Step 3. If the cardinality of S_{up} exceeds m, we recursively invoke DMM on S_{up} to identify the strongest m arms in lines 22 and 23. In the case where the cardinality of S_{up} is either m or $m-1$, we directly obtain the strongest m arms in lines between 24 and 27. If $|S_{\mathrm{up}} \cup \{a\}| < m$ then we extract the remaining arms from S_{down}, ensuring these arms constitute the strongest in S_{down}.

Thus, according to the standard argument of selection algorithm [7], DMM outputs the best m arms if $\mathbb{1}\{\hat{\mu}_{i,j} > 1/2\} = \mathbb{1}\{\mu_{i,j} > 1/2\}$. Notice that DMM terminates normally even if $\mathbb{1}\{\hat{\mu}_{i,j} > 1/2\} \neq \mathbb{1}\{\mu_{i,j} > 1/2\}$ since the selection sort Algorithm 4 called at lines 5 and 9 runs deterministically in $\binom{n}{2}$ comparisons depending on the input.

(2) Sample complexity:

Let $T(k)$ denote the number of comparisons in DMM(α,β,μ,m) for $|\alpha \cup \beta| = k$. At line 5, the number of comparisons is at most $\binom{5}{2} = 10$. Since $L \leq k/5$, the total number of comparisons between lines 4 and 7 is at most $10 \times k/5 = 2k$. Line 9 requires at most 10 comparisons, while line 12 requires at most $T(\frac{k}{5})$ comparisons since $|median| = \frac{k}{5}$, then the total number of comparisons in Step 1 requires $2k + T(\frac{k}{5})$ for large k. The number of comparisons in Step 2 is $k-1$.

For lines between 22 and 30 we claim that $\max\{|S_{up}|, |S_{down}|\}$ is at most $\frac{7k}{10}$. Since the size of *median* is $\frac{k}{5}$, $\frac{k}{10}$ arms of *median* is weaker than a. Thus we can exclude at least $\frac{3k}{10}$ arms weaker than a even in the worst case. Similarly, at least $\frac{3k}{10}$ arms are stronger than a, which implies that $\max\{|S_{up}|, |S_{down}|\}$ is at most $\frac{7k}{10}$. Thus the number of comparisons in Step 3 is at most $T(\frac{7k}{10})$, either at line 23 or at line 29. Then we obtained the following recurrence relation,

$$T(k) < 2k + T\left(\frac{k}{5}\right) + k - 1 + T\left(\frac{7k}{10}\right). \qquad (15)$$

Let $T(k) = 30k$. Then,

$$
\begin{aligned}
T(k) &< 2k + T\left(\frac{k}{5}\right) + k - 1 + T\left(\frac{7k}{10}\right) \qquad (16)\\
&< 3k + 6k + 21k\\
&\leq 30k
\end{aligned}
$$

holds by (15), and we obtain $T(2m) \leq 60m$.

3.3 Proof of Theorem 3.1

Proof (Proof of Theorem 3.1): The proof consists of two parts: the proof of the correctness and the proof of the sample complexity. For convenience, we use BDI for Algorithm 1.

(1) Correctness:

We claim that BDI outputs the correct Condorcet winner die, i.e., the strongest m arms, with a probability of at least $1-\gamma$.

Firstly, we prove for any $t \geq 1$ that

$$\mathbb{P}(\alpha_{t+1} \text{ is not the best } m \text{ of } [n] - R_{t+1}) \leq 60mt\gamma', \quad (17)$$

i.e., α_{t+1} is the best m arms out of $[n] - R_{t+1}$ with probability at least $1 - 60mt\gamma'$. We prove it by an induction. For the base case, we prove α_2 is the best m arms out of $[n] - R_2 = \alpha_1 \cup \beta_1$. By DD$(\alpha_1, \alpha_1, \gamma'/m^2, N, K)$ at line 4, we obtain the empirical estimation of the strength relationships among arms within

α_1. By Lemma 3.2, we know that $\mathbb{1}\{\hat{\mu}_{i,j} > 1/2\} \neq \mathbb{1}\{\mu_{i,j} > 1/2\}$ holds for any $i \in \alpha_1$, $j \in \alpha_1$ satisfying $i \neq j$ with probability at most γ'/m^2. By union bound, we have

$$\begin{aligned}
&\mathbb{P}(\{\mathbb{1}\{\hat{\mu}_{i,j} > 1/2\} \neq \mathbb{1}\{\mu_{i,j} > 1/2\} \mid i \in \alpha_1, j \in \alpha_1\})\\
&\leq \sum_{i \in \alpha_1, j \in \alpha_1} \mathbb{P}(\mathbb{1}\{\hat{\mu}_{i,j} > 1/2\} \neq \mathbb{1}\{\mu_{i,j} > 1/2\})\\
&\leq \gamma'.
\end{aligned}$$

In a similar way, we have $\mathbb{P}(\{\mathbb{1}\{\hat{\mu}_{i,j} > 1/2\} \neq \mathbb{1}\{\mu_{i,j} > 1/2\} \mid i \in \beta_1, j \in \beta_1\}) \leq \gamma'$ for arms in β_1 at line 14, and $\mathbb{P}(\{\mathbb{1}\{\hat{\mu}_{i,j} > 1/2\} \neq \mathbb{1}\{\mu_{i,j} > 1/2\} \mid i \in \alpha_1, j \in \beta_1\}) \leq \gamma'$ for arms between α_1 and β_1 at line 15. We select the best m arms from $\alpha_1 \cup \beta_1$ at line 16, where DMM$(\alpha_1, \beta_1, \hat{\mu}, m)$ requires at most $60m$ comparisons by Lemma 3.4. Thus,

$$\mathbb{P}(\alpha_2 \text{ is not the best } m \text{ of } \alpha_1 \cup \beta_1) \leq 60m\gamma'$$

by a union bound. We obtain (17) for $t = 1$.

Inductively assuming (17) for $t \geq 1$, we prove it for $t + 1$. Similarly to the base case, the algorithm selects m arms from $[n] - R_t = \alpha_t \cup \beta_t$ such that

$$\mathbb{P}(\alpha_{t+2} \text{ is not the best } m \text{ of } \alpha_{t+1} \cup \beta_{t+1}) \leq 60m\gamma'$$

holds. Then,

$$\begin{aligned}
&\mathbb{P}(\alpha_{t+2} \text{ is not the best } m \text{ of } [n] - R_{t+2})\\
&= \mathbb{P}(\{\alpha_{t+1} \text{ is not the best } m \text{ of } [n] - R_{t+1}\} \cup\\
&\{\alpha_{t+2} \text{ is not the best } m \text{ of } \alpha_{t+1} \cup \beta_{t+1}\})\\
&\leq 60mt\gamma' + 60m\gamma'\\
&= 60m(t+1)\gamma'
\end{aligned}$$

by a union bound. We obtain (17).

Notice that the size of R_t decreases by m in each round since we select β of m arms and let it duel with α. Thus BDI terminates in $\lfloor \frac{n}{m} \rfloor$ rounds. Let $t = \lfloor \frac{n}{m} \rfloor$, then

$$\begin{aligned}
\mathbb{P}(\alpha_{t+1} \text{ is not the best } m \text{ of } [n]) &\leq 60mt\gamma'\\
&\leq 60m\lfloor \tfrac{n}{m} \rfloor \gamma'\\
&\leq 60n\gamma'\\
&\leq \gamma
\end{aligned}$$

where we used $\gamma' = \gamma/60n$. This implies BDI outputs the Condorcet Winner die with probability at least $1 - \gamma$.

(2) Sample complexity:

Let Z denote the total number of dice dueling in BDI. Notice that all dice dueling occurs only within DD. Let Y denote the number of dice dueling in DD and let r denote the number of times the DD is used. Then $Z = Yr$. Notice that $r = 2 \times \lfloor \frac{n}{m} \rfloor + 1$ since the number of iterations in the while loop is $\lfloor \frac{n}{m} \rfloor$ in BDI. By Lemma 3.2, we have $\mathbb{E}[Y] \leq 13.5c'(4h^{-2})m^2(\log m)(2 \log \log_2(2h)^{-1} + \log(\gamma'/m^2)^{-1} + d)$, then

$$\begin{aligned}
\mathbb{E}[Z] =& r\mathbb{E}[Y]\\
\leq& (2\lfloor \tfrac{n}{m} \rfloor + 1)13.5c'(4h^{-2})m^2(\log m)(2 \log \log_2(2h)^{-1}\\
&+ \log(\gamma'/m^2)^{-1} + d)\\
=& (2\lfloor \tfrac{n}{m} \rfloor + 1)13.5c'(4h^{-2})m^2(\log m)(2 \log \log_2(2h)^{-1}\\
&+ \log 60nm^2\gamma^{-1} + d)\\
=& \mathrm{O}(nh^{-2}(\log \log h^{-1} + \log nm^2\gamma^{-1})m \log m).
\end{aligned}$$

where we used $\gamma' = \gamma/60n$.

4. Concluding Remarks

This paper studied the sample complexity bounds for finding the strongest die in the dueling dice problem, which is a variant of the dueling bandits problem. We proposed the Best Die Identification algorithm which is the first algorithm to find the best-m arms with dice dueling setting and without an assumption of a total order over dice. Then, we gave the expected sample complexity $\mathrm{O}(nh^{-2}(\log \log h^{-1} + \log nm^2\gamma^{-1})m \log m)$.

An information-theoretical lower bound of the problem is a future work; in particular, it is important to clarify if the $m \log m$ term of the upper bound is tight. Cohen et al.[5] showed $\Omega(n)$ duels are necessary to identify the best m arms under the SST assumption. But, any lower bound is not known without the SST assumption. A regret analysis of the dice dueling problem is also interesting.

Acknowledgments

The authors are grateful to the anonymous reviewers for their invaluable comments. This work is partly supported by JSPS KAKENHI Grant Numbers JP23K24905, JP23H03348, JP21H03396 and JP23K21645, MEXT KAKENHI Grant Number JP20H05967, and JST ERATO Grant Number JPMJER2301, Japan.

References

[1] E. Akin. Generalized intransitive dice: Mimicking an arbitrary tournament. *Journal of Dynamics and Games*, 2021, 8(1):1–20.

[2] D. Black. *The Theory of Committees and Elections*. Cambridge Univ. Press, 1958.

[3] J. Buhler, R. Graham, and A. Hales. Maximally non-transitive dice. *Amer Math*, 125(2018):389–399.

[4] W. Chen, Y. Du, L. Huang, and H. Zhao. Combinatorial pure exploration for dueling bandits. *In Proceedings of the International Conference on Machine Learning (ICML)*, pages 1531–1541, 2020.

[5] L. Cohen, U. Schmidt-Kraepelin, and Y. Mansour. Dueling bandits with team comparisons. *Advances in Neural Information Processing Systems*, 34:20633–20644, 2021.

[6] B. Conrey, J. Gabbard, K. Grant, A. Liu, and K. Morrison. Intransitive dice. *Math. Mag.*, 89(2016):133–143.

[7] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein. *Introduction to Algorithms, Fourth Edition*. The MIT Press, Cambridge, Massachusetts, 2022.

[8] M. Gardner. The paradox of the nontransitive dice and the elusive principal of indifference. *Sci. Amer.*, 223(1970):110–114.

[9] B. Haddenhorst, V. Bengs, J. Brandt, and E. Hullermeier. Testification of condorcet winners in dueling bandits. *In Proceedings of Conference on Uncertainty in Artificial Intelligence*, (UAI 2021):1195–1205.

[10] S. Kalyanakrishnan and P. Stone. Eficient selection of multiple bandit arms: Theory and practice. *Proceedings of the 27th International Conference on Machine Learning*, (ICML 2010):511–518.

[11] J. Komiyama, J. Honda, H. Kashima, and H. Nakagawa. Regret lower bound and optimal algorithm in dueling bandit problem. *In proc. COLT*, 40:1141–1154, 2015.

[12] S. Lu and S. Kijima. Is there a strongest die in a set of dice with the same mean pips? *Proceedings of the AAAI Conference on Artificial Intelligence*, 36(5):5133–5140, 2022.

[13] A. Maiti, K. Jamieson, and L. Ratliff. Instance-dependent sample complexity bounds for zero-sum matrix games. *International Conference on Artificial Intelligence and Statistics*, pages 9429–9469, 2023.

[14] M. Mitzenmacher and E. Upfal. *Probability and Computing: Randomization and Probabilistic Techniques in Algorithms and Data Analysis,2nd ed.* Cambridge Univ. Press, 2017.

[15] S. Mohajer, C. Suh, and A. Elmahdy. Active learning for top-k rank aggregation from noisy comparisons. *In Proceedings of the 34th International Conference on Machine Learning*, 70:2488–2497, 2017.

[16] W. Ren, J. Liu, and N. B. Shroff. The sample complexity of best-k items selection from pairwise comparisons. *In International Conference on Machine Learning*, 2020:8051–8072, 2020.

[17] R.P. Savage. The paradox of non-transitive dice. *Amer. Math. Monthly*, 101:429–436, 1994.

[18] A. Schaefer. Balanced non-transitive dice ii: Tournaments, 2017.

[19] A. Schaefer and J. Schweig. Balanced nontransitive dice. *The College Mathematics Journal*, 48(1):429–436, 2017.

[20] Z. Usiskin. Max-min probabilities in the voting paradox. *Ann. Math. Statist.*, 35:857–862, 1964.

[21] Y. Yue, J. Broder, R. Kleinberg, and T. Joachims. The k-armed dueling bandits problem. *Journal of Computer and System Sciences*, 78(5):1538–1556, 2012.

[22] Y. Yue and T. Joachims. Interactively optimizing information retrieval systems as a dueling bandits problem. *In Proceedings of International Conference on Machine Learning*, 26th:1201–1208, 2009.

Appendix A: Concentration inequality

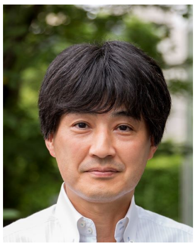
Lemma Appendix A.1 (Hoeffding inequality[14]): Suppose $X_1, X_2, \dots$ to be i.i.d. random variables $X_n \sim \text{Bernoulli}(\mu)$ for $\mu \in [0,1]$. For $t \in \mathbb{N}$ let $\hat{\mu}^t$ be the corresponding empirical distribution after the t observations $X_1, X_2, \dots, X_t$, i.e., $\hat{\mu}^t = \frac{1}{t}\sum_{s=1}^{t} \mathbb{1}_{\{X_s=1\}}$. Then, we have for any $\epsilon > 0$ and $t \in \mathbb{N}$ the estimate

$$\mathbb{P}\left(|\hat{\mu}^t - \mu| > \epsilon\right) \le 2e^{-t\epsilon^2}.$$

Shang LU received B.E. from Central China Normal University in 2016 and M.Sc. in IGSES, Department of AIT from Kyushu University in 2021. He is currently a Ph.D Student at Kyushu University. His research interest include game theory, linear programming and online learning.

Kohei HATANO received Ph.D. from Tokyo Institute of Technology in 2005. Currently, he is a professor at Department of Informatics in Kyushu University. He is also the leader of the Computational Learning Theory team at RIKEN AIP. His research interests include machine learning, computational learning theory, online learning and their applications.

Shuji KIJIMA received his Ph.D. degree in mathematical informatics from the University of Tokyo in 2007. After working at Kyoto University and Kyushu University, he is currently a professor at Shiga University since 2022. His research interests include random structures and algorithms.



Eiji TAKIMOTO received Ph.D. from Tohoku University in 1991. Currently, he is a professor at Department of Informatics in Kyushu University. His research interests include computational learning theory, online learning, and computational complexity.