Fair Seeding in Knockout Tournaments

THUC VU
and
YOAV SHOHAM
Stanford University

Most of the past work on the seeding of a knockout tournament has focused on maximizing the winning probability of the strongest player (so-called "predictive power"). In contrast, we focus on finding a fair seeding. We consider two alternative fairness criteria, adapted from the literature: envy-freeness and order preservation. For the first criterion we provide a solution for unconstrained tournaments, and provide an impossibility result for balanced tournaments. For the second criterion we have a similar result for unconstrained tournaments, but not for the balanced case. We provide instead a heuristic algorithm which we show through experiments to be efficient and effective. Surprisingly, the criterion becomes provably impossible to achieve when we add a weak condition guarding against the phenomenon of tournament dropout.

Categories and Subject Descriptors: I.2.11 [**Artificial Intelligence**]: Multiagent Systems

General Terms: Tournament Design

Additional Key Words and Phrases: Tournament Design, Optimization, Heuristic Algorithm

1. INTRODUCTION

Knockout tournaments are widely used in many different sporting events. In this simple and familiar tournament format, the players are placed at the leaf nodes of a binary tree (which is usually balanced). The players at sibling nodes will compete against each other in pairwise matches and the winners will move up the tree. The winner of the tournament is the player who reaches the root node of the tournament tree. The arrangement of the players at the leaf nodes is called the seeding of the tournament.

The seeding can significantly influence the result of a tournament. This has been demonstrated in several papers, but most of them focus on how to find a seeding that maximizes the winning probability of the strongest player (so-called "predictive power") (see, e.g., [Horen and Riezman 1985; Appleton 1995; Vu et al. 2009]). This usually means giving the strongest player the easiest schedule, while making it harder for the remaining players. An example of such seedings is given in Figure 1. In this example, players are numbered based on their strengths with player 1 being the strongest player. This seeding is rarely seen in practice however, because it seems to be very unfair to other players, especially the strong ones such as 2 or 3.

On the other hand, the most popular seeding used in major sport tournaments is the one that pairs up in the first stage the strongest player with the weakest player, the second strongest player with the second weakest player, so on and so forth. An example for 8 players is shown in Figure 2. One possible rationale is that it seems fair: confrontations between strong players are delayed until later rounds, and that helps to increase the chance that one of those strong players will win the

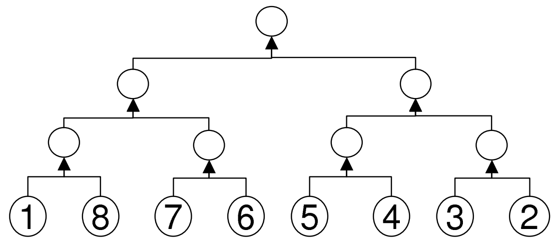

Fig. 1. A biased seeding that maximizes winning probability of player 1

tournament. This raises several questions: How do we capture this intuition about fairness? Is there any other seeding that is also fair? Can we always find such a fair seeding?

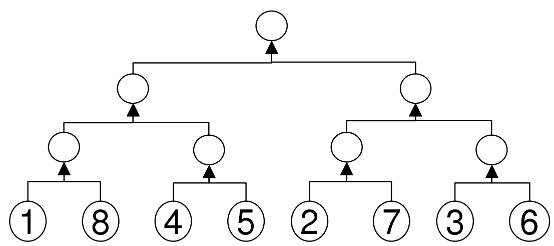

Fig. 2. A popular seeding with 8 players

To answer these questions, we consider two alternative fairness criteria adapted from the literature: envy-freeness and order preservation. We analyze separately each criterion across different tournament settings, providing both possibility and impossibility results. To explain our results we need to lay out the settings with all its variants, which we do in the next section. In Section 3 we summarize the existing results from the literature. We then analyze envy-freeness and order preservation in Section 4 and 5 respectively, and conclude in Section 6.

2. SETTINGS

We consider two different types of tournament structures and two different player models. Regarding the tournament structure, we address both balanced and unbalanced trees (as long as it is a binary tree with n leaf nodes).

For unbalanced tournaments, we want to draw attention to a special structure called "caterpillar" tree. In a tournament with a caterpillar structure, the players are arranged in a particular order. The first two players will compete against each other. The winner of the match will then advance forward and compete with the third player, so on and so forth. An example is shown in Figure 3. This structure is often addressed in the social choice literature as a voting tree structure with interesting properties (e.g., see [Fischer et al. 2009]).

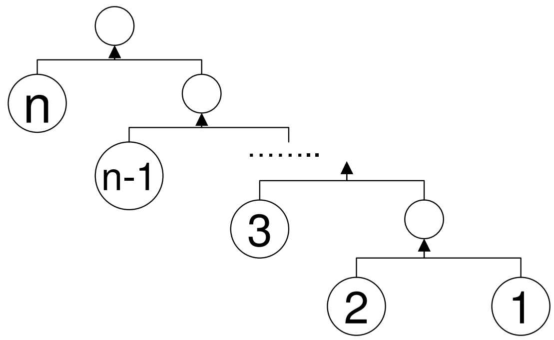

Fig. 3. A caterpillar tournament for n players

For balanced tournaments, S_n^* is a seeding formed by matching in the first stage the strongest player with the weakest player, the second strongest player with the second weakest player, and so on and so forth. For the successive stages, the sub-tree containing the strongest player is combined with the sub-tree with the weakest players possible, so on and so forth.

$$S_n^* = \Big[[1,n], ..., [i,(n-i+1)], [(\frac{n}{2}-i+1), (\frac{n}{2}+i)], ...\Big]$$

Here we encode the seeding by a permutation of the numbers between 1 and n. The k^{th} number is the ranking of the player that will be placed at the k^{th} leaf node of the binary tournament tree, from left to right. An example of S_n^* when $n = 8$ is $S_8^* = [1, 8, 4, 5, 2, 7, 3, 6]$ as shown in Figure 2.

Regarding player models, we assume that for any pairwise match, the probability of one player winning against the other is known. This probability can be obtained from past statistics or from some learning models. In the first model, we do not place any constraints on the probabilities besides the fundamental properties.

Definition 2.1 General Player Model. Given a set of n players, the winning probabilities between the players form a matrix P such that p_{ij} denotes the probability that player i will win against player j, $\forall(i \neq j) : 1 \leq i, j \leq n$, and P satisfies the following constraints:
1. $p_{ij} + p_{ji} = 1$
2. $0 \leq p_{ij}, p_{ji} \leq 1$

Note that there might be no transitivity between the winning probabilities, i.e., we can have $p_{ij} > 0.5$, $p_{jk} > 0.5$, and $p_{ki} > 0.5$.

The second player model is the monotonic model which is a special case of the general model. This model is popular and well known in the literature (see for example [Horen and Riezman 1985; Hwang 1982; Schwenk 2000; Vu et al. 2009]). The players are assumed to have unknown but fixed intrinsic strengths or abilities. They are numbered from 1 to n in descending order of their strengths and the winning probabilities between the players reflect these rankings.

Definition 2.2 Monotonic Player Model. Given a set of n players, the winning probabilities between the players form a matrix P such that p_{ij} denotes the proba-

bility that player i will win against player j, $\forall(i \neq j) : 1 \leq i, j \leq n$, and P satisfies the following constraints:
1. $p_{ij} + p_{ji} = 1$
2. $0 \leq p_{ij}, p_{ji} \leq 1$
3. $p_{ij} \leq p_{i(j+1)}$
4. $p_{ji} \geq p_{(j+1)i}$ (which is actually implied by (1) and (3))

We define a knockout tournament as the following:

Definition 2.3 General Knockout Tournament. Given a set N of players and a matrix P such that p_{ij} denotes the probability that player i will win against player j in a pairwise elimination match and $0 \leq p_{ij} = 1 - p_{ji} \leq 1$ ($\forall i, j \in N$), a knockout tournament $KT_N = (T, S)$ is defined by:
1. A tournament structure T which is a binary tree with $|N|$ leaf nodes
2. A seeding S which is a one-to-one mapping between the players in N and the leaf nodes of T

Note that we will write KT_N as KT when the context is clear.

Given the winning probability matrix P for a set of players N and a tournament $KT_N = (T, S)$, the probability of player k winning the tournament, denoted $q(k, KT_N)$, is calculated by the following recursive formula:
1. If $N = \{j\}$, then $q(k, KT_N) = \begin{cases} 1 & \text{if } k = j \\ 0 & \text{if } k \neq j \end{cases}$
2. If $|N| \geq 2$, let $KT_{N1} = (T_1, S_1)$ and $KT_{N2} = (T_2, S_2)$ be the two sub-tournaments of KT_N such that T_1 and T_2 are the two subtrees connected to the root node of T, and N_1 and N_2 are the set of players assigned to the leaf nodes of T_1 and T_2 by S_1 and S_2 respectively. If $k \in N_1$ then

$$q(k, KT_N) = \sum_{i \in N_2} q(k, KT_{N1}) \times q(i, KT_{N2}) \times p_{ki}$$

and symmetrically for $k \in N_2$.

Using the formula above, we can efficiently calculate the winning probabilities of the players in $O(|N|^2)$.

3. PAST RESULTS

Most of the related work focuses on balanced knockout tournaments with the monotonic player model. In [Schwenk 2000], it is suggested that good seedings should satisfy a property called "Sincerity Rewarded". The property requires that a better ranked player should never be given a tournament schedule more difficult than that of a lower ranked player.

Alternatively, in [Hwang 1982], a different property called "Monotonicity"[1] is used to evaluate seedings. This property is satisfied if and only if stronger players have higher chance of winning than weaker players.

We generalize "Sincerity Rewarded" and "Monotonicity" so that they can also be applied for the general player model. Unlike past work which uses probabilistic seedings [Schwenk 2000] or dynamic seedings [Hwang 1982] (i.e., re-seeding the

[1] Not to be confused with the monotonic player model

players after each round based on the results of the previous rounds), we focus on deterministic and static seedings instead.

Interestingly, in [Horen and Riezman 1985] it is shown that for a balanced tournament of size 4 with monotonic player model, the seeding [1 4 2 3] always achieves "Monotonicity". Tournament of size 4 is a special case due to its extremely small size. For our paper, we focus on tournaments of sizes 8 or above.

4. FIRST FAIRNESS CRITERION: ENVY-FREENESS

In [Schwenk 2000], it is argued that a good seeding should not give any strong player a harder tournament schedule than a weaker player. One of the main reasons is that if strong players are given harder schedules, they will have incentives to underperform during the pre-season period or lie about their strengths in order to get lower rankings instead. We formalize this intuition to define envy-freeness as the first criterion for a seeding to be fair: no player envies the seeding position of another player weaker than him.

First we need to specify when a player is considered to be stronger than another. In the monotonic player model, the notion of a stronger player is clear. For any pair of players (i, j) such that $i < j$, player i is always stronger than player j. In the general model, that is no longer the case. Therefore we define a player to be stronger between the two if he dominates the second player in the winning probabilities against other players.

Definition 4.1 Dominance. Player i dominates another player j if and only if $\forall k : 1 \leq k \leq n$, it is the case that $p_{ik} \geq p_{jk}$ (which also implies $p_{ij} \geq 0.5 \geq p_{ji}$).

Notice that the definition of Dominance applies naturally to the monotonic model as well. When there is a tie between two players (i.e., all of their winning probabilities are equal), we need a tie-breaking rule. Our results hold for any monotonic tie-breaking rule. WLOG, we assume the tie is resolved based on lexicographic ordering: for any pair of players (i, j) such that $\forall k : 1 \leq k \leq n$, $p_{ik} = p_{jk}$ and $p_{ij} = p_{ji} = 0.5$, i is considered to dominate j if and only if $i < j$.

Given the definition of Dominance, we define Envy-freeness as the following.

Definition 4.2 Envy-free Seeding. Given a set N of players, a winning probability matrix P, and a knockout tournament $KT = (T, S)$, the seeding S is envy-free if and only if $\forall (i, j) \in N$ such that i dominates j, player i does not have a higher probability of winning the tournament by swapping his position in the seeding S with j's, i.e., $q(i, (T, S)) \geq q(i, (T, S' = S_{i \leftrightarrow j}))$.

Notice that in the general player model, if there is no pair of players such that one dominates the other, any seeding will trivially satisfy the requirement above. On the other hand, in the monotonic player model, an envy-free seeding will have to satisfy the requirement for all pairs of players.

4.1 Envy-freeness for Unconstrained Tournaments

When there is no constraint on the structure of the tournament, envy-freeness can be achieved by the following tournament structure and seeding method.

THEOREM 4.3. *Given a set N of players, a winning probability matrix P, envy-freeness can be achieved by the knockout tournament with the caterpillar structure* *and the seeding S such that for every pair of players (i, j), if i dominates j then j is placed below i on the tournament tree.*

The seeding above can be achieved easily by first constructing a directed acyclic graph in which each node represents a player and there is a directed edge from player i to player j ($\forall(i, j) \in N$) if and only if i dominates j. We then perform a topological sort of the graph and use this ordering to place players on the tournament tree.

Proof of Theorem4.3. We proof the theorem using induction on the number of players. *Base case:* When $|N| = 2$, there is only one possible binary tree with 2 leaf nodes.

Inductive step: Assume that the theorem holds for all N' such that $|N'| = n-1$, we need to show for all N such that $|N| = n$, the seeding we construct as above is envy-free. Let k^* be the player at the root node of the dominance graph of N, i.e., there is no player $k' \in N$ such that k' dominates k^*. Let $N' = N \setminus \{k^*\}$. Since N' has size $n-1$, based on the inductive hypothesis we can find an envy-free seeding S'. Let S be created by combining k^* and S' as in Figure 4. We need to show that S is also envy-free.

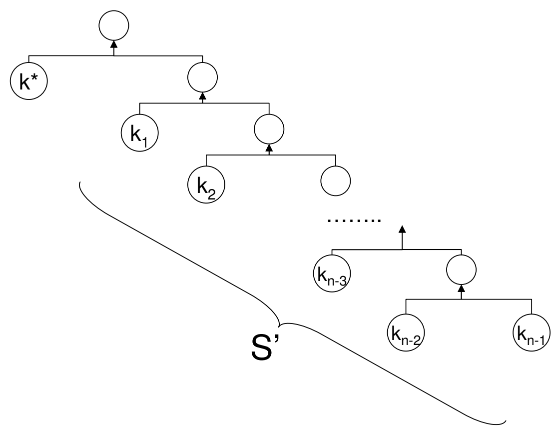

Fig. 4. The caterpillar seeding S created by combining k^* and S'

Since S' is envy-free and k^* is not dominated by any player in N', we just need to show that k^* will not have a higher winning probability by swapping its seeding position with any player $k_i \in N'$. Using the same notation as in Section 2, we denote S as $[k^*, k_1, k_2, ..., k_{i-1}, k_i, k_{i+1}, ..., k_{n-1}]$. Let S_i be the seeding created after k^* is swapped with k_i: $S_i = [k_i, k_1, k_2, ..., k_{i-1}, k^*, k_{i+1}, ..., k_{n-1}]$. Let $p(k_j)$, $\forall j : i < j \leq n-1$, be the probability that player k_j will get to play against k^* (in other words, the probability that k_j will win the sub-tournament among $\{k_{i+1}, ..., k_{n-1}\}$). The probability that k^* will win the tournament with the seeding S_i, $q(k^*, S_i)$, is the same as with the seeding $S'_i = [k_1, k_2, ..., k_{i-1}, k_i, k^*, k_{i+1}, ..., k_{n-1}]$ because:

$$q(k^*, S_i) = \sum_{j:i<j\leq n-1} [p(k_j) \times p_{k^*k_j}] \times \prod_{j:1\leq j\leq i} p_{k^*k_j} = q(k^*, S'_i)$$

Let's consider the winning probability of k^* in S'_{i-1}, which is created by swapping k^* and k_i (i.e., $S'_{i-1} = [k_1, k_2, ..., k_{i-1}, k^*, k_i, k_{i+1}, ..., k_{n-1}]$).

$$\begin{aligned}
q(k^*, S'_{i-1}) &= \sum_{l:i<j\leq n-1} [p(k_j)p_{k_jk_i}p_{k^*k_j} + p(k_j)p_{k_ik_j}p_{k^*k_i}] \times \prod_{j:1\leq j\leq i-1} p_{k^*k_j} \\
&= \sum_{l:i<j\leq n-1} p(k_j)[p_{k_jk_i}p_{k^*k_j} + p_{k_ik_j}p_{k^*k_i}] \times \prod_{j:1\leq j\leq i-1} p_{k^*k_j} \\
&\geq \sum_{l:i<j\leq n-1} p(k_j) \times min(p_{k^*k_j}, p_{k^*k_i}) \times \prod_{j:1\leq j\leq i-1} p_{k^*k_j} \\
&\geq \sum_{l:i<j\leq n-1} p(k_j) \times p_{k^*k_j} \times p_{k^*k_i} \times \prod_{j:1\leq j\leq i-1} p_{k^*k_j} \\
&\geq \sum_{l:i<j\leq n-1} p(k_j) \times p_{k^*k_j} \times \prod_{j:1\leq j\leq i} p_{k^*k_j} \\
&\geq q(k^*, S'_i)
\end{aligned}$$

The first inequality holds because $p_{k_jk_i} + p_{k_ik_j} = 1$. The second inequality holds because $0 \leq p_{k^*k_j}, p_{k^*k_i} \leq 1$.

This means when we swap the position of k^* with the position of the player right in front of k^* in the seeding, the winning probability of k^* does not decrease. Thus if we keep swapping the position of k^* with k_j in S'_j sequentially for $j = (i-1) \rightarrow 1$, we have the following sequence of inequalities:

$$q(k^*, S_i) = q(k^*, S'_i) \leq q(k^*, S'_{i-1}) \leq ... \leq q(k^*, S'_2) \leq q(k^*, S'_1) \leq q(k^*, S)$$

This shows that k^* will not have a higher winning probability by swapping its seeding position with player k_i. Thus S is envy-free. □

4.2 Envy-freeness for Balanced Tournaments

When the tournament structure has to be a balanced binary tree, it is no longer possible to always achieve envy-freeness even when the winning probabilities between players are monotonic. Here we assume n, the number of players, is a power of 2.

THEOREM 4.4. *For any $n = 2^k \geq 8$, there exists a set of n players with a monotonic winning probability matrix P such that it is not possible to find an envy-free seeding S for the balanced knockout tournament between these n players.*

PROOF. We first prove the result for the case when $n = 8$, and then for general n.

The case of $n = 8$: We construct a tournament between 8 players with the winning probability matrix as shown in Table I (note that we only show the top half of the matrix). Since there are only 315 unique seedings for a tournament of 8 players, we can exhaustively check all of them by using a computer program. Given this winning probability matrix, for each of those 315 seedings, there is always a pair of players whose probabilities of winning the tournament violate the criterion.

The case of $n > 8$: To construct a tournament between n players, we add $n - 8$ dummy players to the set of 8 players above. The dummy players lose to the original 8 players with probability 1. Note that player 7 and 8 can be regarded as

	2	3	4	5	6	7	8
1	0.5	0.5	0.5	$0.5+2\epsilon$	$0.5+2\epsilon$	1	1
2	-	0.5	0.5	0.5	$0.5+\epsilon$	1	1
3	-	-	0.5	0.5	$0.5+\epsilon$	1	1
4	-	-	-	0.5	$0.5+\epsilon$	1	1
5	-	-	-	-	0.5	1	1
6	-	-	-	-	-	1	1
7	-	-	-	-	-	-	0.5

Table I. The winning probabilities between 8 players

the dummy players to the top 6 players since player 7 and 8 lose with probability 1 to these players. Let M denote the set of the top 6 players from 1 to 6. Notice that the participations of the dummy players, 7, and 8 do not change the winning probabilities of the players in M. Essentially the tournament between n players is reduced to a tournament between the players in M as if we take out all of the dummy players and player 7 and 8.

Let's consider all possible tournament trees T_M for these 6 players. We first prove the following property of T_M: For any seeding to be envy-free, in the *reduced* tournament tree T_M of the players in M, there exists no pair of players (i, j) such that the difference between their depths is greater than 1 as in Figure 5.

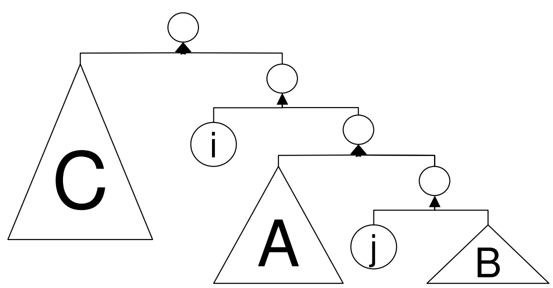

Fig. 5. A reduced tournament structure that does not satisfy envy-freeness

This is true since we can always swap j with the dummy player who plays against i in the original tournament and improve the winning probability of j. To show this, let's first calculate the winning probability of j before the swap:

$$Q_1 = \sum_{l \in B}[q(l,B) \times p_{jl}] \times \sum_{k \in A}[q(k,A) \times p_{jk}] \times p_{ji} \times \sum_{h \in C}[q(h,C) \times p_{jh}]$$

Recall that $q(k, A)$ is the probability that player k will win the sub-tournament A.

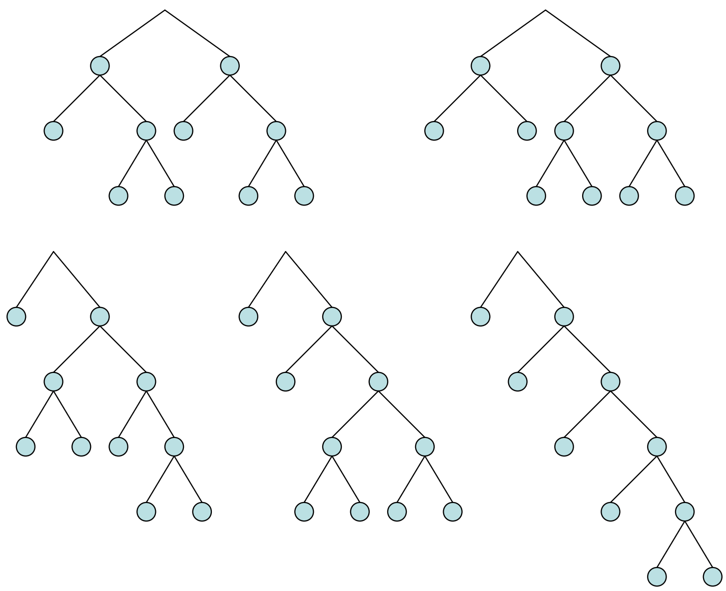

Fig. 6. Possible tournament trees for 6 players

After the swap, the winning probability of j becomes:

$$\begin{aligned}
Q_2 &= \sum_{l\in B}\sum_{k\in A} q(l,B)\times q(k,A)\times[p_{jl}p_{lk}+p_{jk}p_{kl}]\times p_{ji}\times\sum_{h\in C}[q(h,C)\times p_{jh}]\\
&\geq \sum_{l\in B,k\in A} q(l,B)\times q(k,A)\times \min(p_{jl},p_{jk})\times p_{ji}\times\sum_{h\in C}[q(h,C)\times p_{jh}]\\
&> \sum_{l\in B,k\in A} q(l,B)\times q(k,A)\times (p_{jl}\times p_{jk})\times p_{ji}\times\sum_{h\in C}[q(h,C)\times p_{jh}]\\
&= \sum_{l\in B} q(l,B)\times p_{jl}\times\sum_{k\in A} q(k,A)\times p_{jk}\times p_{ji}\times\sum_{h\in C}[q(h,C)\times p_{jh}] = Q_1
\end{aligned}$$

The first inequality holds because $p_{lk}+p_{kl}=1$. The second inequality holds because for j, l, and k in the top 6 players, p_{jl} and p_{jk} is less than 1. Note that we do not assume any relationship between i and j here since our focus is on j being envious of the dummy player who got matched up with i in the original tournament.

There are only 6 possible knockout tournament trees for 6 players as shown in Figure 6. However, only the two top trees satisfy the condition that there is no pair of players with a difference in their depths greater than 1. For each of these trees, notice that for the two players who are placed at a smaller depth than the rest, they must have played against dummy players in the original tournament before they can advance to the current round. Thus for both choices, they are analogous to a seeding of the 8 players we have considered in the first part of the proof (recall that player 7 and 8 play the same role as the dummy players). Therefore there does not exist any envy-free seeding for these two possible reduced trees either. This implies there is no envy-free seeding for the original tournament tree. □

Note that the same result does not apply for unconstrained tournaments because in the proof above, we make use of the fact that to advance to the next round, any player must play against some player (which can be a dummy player). In the case of unconstrained tournament, a player can be advance directly to the final round.

Since the monotonic player model is a special case of the general model, we have the following corollary.

Corollary 4.5. *For any given $n = 2^k \geq 8$, it is not always possible in the general player model to find an envy-free seeding for a balanced knockout tournament of size n.*

The envy-freeness requirement is too strong to achieve for all cases of winning probabilities when the tournament has to be balanced. In the next section we discuss another criterion which seems to require a weaker condition for a seeding to be fair with the hope that it can be easier to satisfied.

5. SECOND FAIRNESS CRITERION: ORDER PRESERVATION

Instead of focusing on the seeding positions of the players, we place a requirement on their probabilities of winning the tournament instead: stronger players must have higher probabilities of winning the tournament than weaker players. This requirement seems to be less restrictive than the envy-freeness criterion, yet it still encourages the players to perform accordingly to their strengths.

Definition 5.1 Order Preserving Seeding. Given a set N of players, a winning probability matrix P, and a knockout tournament $KT = (T, S)$, the seeding S is order preserving if and only if $\forall(i, j) \in N$ such that i dominates j, player i does not have a lower probability of winning the tournament than player j, i.e., $q(i, (T, S)) \geq q(j, (T, S))$.

The criterion is adapted and generalized from the notion of "Monotonicity" first introduced in [Hwang 1982]. As with the first criterion, for any pair of players such that no player dominates the other, we do not place any requirement on the relative ordering of their winning probabilities.

5.1 Order Preservation for Unconstrained Tournaments

When there is no constraint on the structure of the tournament, it is easy to see that the same tournament structure and seeding in Theorem 4.3 will achieve order preservation.

Theorem 5.2. *Given a set N of players, a winning probability matrix P, order preservation can be achieved by the knockout tournament with the caterpillar structure and the seeding S such that for every pair of players (i, j), if i dominates j then j is placed below i on the tournament tree.*

Proof. Recall from Theorem 4.3 that the seeding S created as above is envy-free. Using proof by contradiction, we assume S is not order preserving, and show that it will not be envy-free either.

If S is not order preserving, there must exist two player i, and j such that i dominates j, and the winning probability of i is smaller than j. Since i dominates

j, i must be placed above j on the tournament tree. Let A be the set of players placed above i on the tournament tree, B between i and j, and C below j. The winning probabilities of j given the seeding S can be calculated as the following:

$$q(j, S) = \prod_{k \in A} p_{jk} \times p_{ji} \times \prod_{k \in B} p_{jk} \times \prod_{k \in C} q(k, C) p_{jk}$$

Here we use $q(k, C)$ to denote the probability that player $k \in C$ will win the sub-tournament between the players in C and advance to the match against player j.

□

Notice that the method above does not take into consideration the actual values of the winning probabilities between players. The criterion can be achieved as long as the relative ordering of the players are known. It would be very useful if the same result applied for the case of balanced tournament. Unfortunately, as we show in the next section, no fixed seeding is always order preserving, even when the winning probabilities are monotonic.

5.2 Order Preservation for Balanced Tournaments

If the seeding S_n^* as in Figure 2 could be shown to always satisfy order preservation, this would justify its popularity in practice. However, as we show in the following theorem, there are cases in which this seeding violates the criterion.

THEOREM 5.3. *For any* fixed *seeding S of a balanced tournament of size $n = 2^k \geq 8$, there exists a monotonic winning probability matrix P such that S is not order preserving.*

PROOF. We first show by induction that if there exists a fixed seeding S that always satisfies order preservation, then S has to be S_n^* (as defined in Section 2).

Base case: When $n = 2$, $S_2^* = [1\ 2]$ is the only seeding.

Inductive case: Assume that the claim holds for $n = m$, we need to show that it also holds for $n = 2m$. Let's consider the following winning probabilities between the players:

—$\forall i, j : 1 \leq i, j < 2m$, $p_{ij} = 0.5$
—$\forall i : 1 \leq i < 2m$, $p_{i(2m)} = 1$

The only order preserving seedings in this case are the ones that pair up player 1 and $2m$ in the first round. Otherwise whoever gets to play against player $2m$ will have a winning probability of $\frac{2}{m}$, whereas player 1 only wins with probability $\frac{1}{m}$. Consequently, for each player k from 2 to m, we show that k has to be matched up with player $2m - k + 1$ by using the following winning probabilities:

—$\forall i, j : 1 \leq i, j \leq 2m - k$, $p_{ij} = 0.5$
—$\forall i, j : 1 \leq i \leq 2m - k, 2m - k + 1 \leq j \leq 2m$, $p_{ij} = 1$
—$\forall i, j : 2m - k + 1 \leq i, j \leq 2m$, $p_{ij} = 0.5$

Now let's consider the case in which the top m players win against the bottom m players with probability 1. We have shown above that player k has to be paired up with player $2m - k + 1$. Therefore, all the top m players will advance to the next round with probability 1. Based on the inductive hypothesis, they must be seeded by S^*_m. This proves that the seeding for $2m$ players must be S^*_{2m}.

Since we have shown that the fixed seeding could only possibly be S^*_n, we just need to find a winning probability matrix such that S^*_n violates the criterion. Consider the following winning probabilities:

—$\forall i, j : 1 \leq i < 6 < j \leq n$, $p_{ij} = 1$
—$\forall i : 1 \leq i \leq n$, $p_{7i} = p_{8i} = 1$
—$\forall i, j : 1 \leq i, j \leq 5$, $p_{ij} = 0.5$
—$p_{36} = p_{46} = p_{56} = 0.5$, and $p_{16} = p_{26} = 1$
—All other winning probabilities are 0.5

Let's consider the last 3 rounds of the tournament. Only the top 8 players make to these rounds and they are seeded by $S^*_8 = [1, 8, 4, 5, 2, 7, 3, 6]$. Intuitively, since player 3 has only 50% chance of winning against player 6, player 2 has more chance of getting into the final match than player 1. If we carry out the calculations, we have $q(1, S^*_n) = 0.25 < q(2, S^*_n) = 0.375$, which violates order preservation. □

Since the monotonic player model is a special case of the general model, we have the following corollary.

Corollary 5.4. *For any given $n = 2^k \geq 8$, there is no* fixed *seeding S that is always order preserving for any balanced tournaments of size n in the general player model.*

The results above do not preclude the existence of a fair seeding for any *given* set of players and winning probabilities. While we do not yet have a proof for such existence, we propose a heuristic algorithm to find an order preserving seeding and show through experiments that the algorithm is indeed efficient and effective. The pseudocode is provided in Algorithm 1.

The recursive algorithm attempts to find a fair seeding for each half of the initial seeding. It then checks to see if there are any pairs of players that violate the order preserving condition. If there are, it will pick the pair with the maximum difference between the players and swap the seeding positions of those players. It will then repeat the whole process.

To test the algorithm, we generate 100k test cases for each of the values of n between 8 and 256. For each test case, we randomly generate the winning probability matrix for n players and then use the algorithm to find a seeding that satisfies order preservation. We use S^*_n (defined in Section 2) as the initial seeding input.

It is not trivial to generate the winning probabilities since we need to make sure the matrix satisfies the monotonicity condition while ensuring certain randomness. The process is composed of the following steps:

(1) Generate the winning probabilities of player 1 by sampling $(n - 1)$ values uniformly from [0.5,1] and sort them in ascending order.
(2) Assign $p_{ii} = 0.5$ $\forall i$.
(3) Generate p_{ij} ($\forall (i, j) : i < j$) by sampling uniformly between $p_{i(j-1)}$ and $p_{(i-1)j}$, and then assign $p_{ji} = 1 - p_{ij}$.

Algorithm 1 Find-fair-seed (S: Initial seeding)

```
if S satisfies fairness then
  Return S;
end if
done ← false;
while !done do
  done ← true;
  Find-fair-seed (the first half of S);
  Find-fair-seed (the second half of S);
  N ← the set of players in S;
  wp ← winning probabilities of the players in N;
  max_dif ← 0;
  for i, j ∈ N : j − i > max_dif do
    if wp(i) < wp(j) then
      done ← false;
      max_dif ← j − i;
      i* ← i;
      j* ← j;
    end if
  end for
  if !done then
    Swap the seeding positions of i*, j* in S;
  end if
end while
Return S;
```

We show in Figure 7 a graph of the average and maximum number of swaps are made before a fair seeding is found. The graph shows that both quantities grow linearly with n. And even in the case of $n = 256$ players, the maximum number of swaps needed is still very low. This shows that the algorithm indeed works well in practice.

Even though a seeding that satisfies order preservation can be found efficiently by the algorithm above, unfortunately the criterion becomes provably impossible to achieve when a weak condition is added. We discuss this additional condition and how it interacts with the order preservation criterion in the next section.

5.3 Robustness against Dropout

The experimental results of the heuristic algorithm in the previous section suggest that perhaps it is easy and always possible to find an order preserving seeding. However, this optimism is quite fragile. When we add a reasonable and innocuous requirement of robustness against dropout, the criterion becomes provably impossible to achieve in certain cases.

A dropout occurs when a player forfeits his participation in a tournament due to exogenous factors such as injuries or illness, or in order to manipulate the outcome of the tournament. This phenomenon can significantly influence the winning probabilities of the remaining players. We define robustness against dropout as the

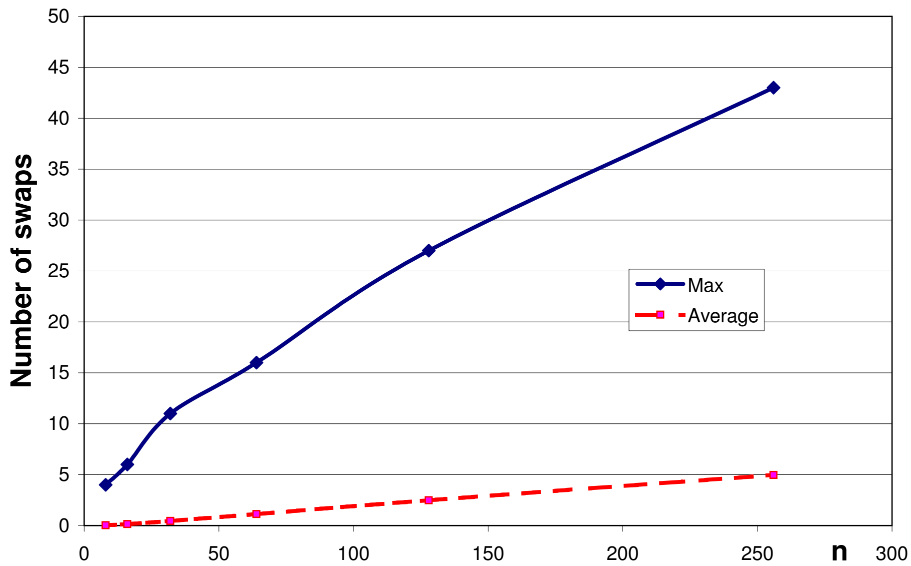

Fig. 7. The average and maximum number of swaps over 100k tournaments for each n

following: there does not exist a pair of players (i, j) such that i dominates j, and also j gains while i loses in winning probability after some player drops out. We allow the re-seeding of the remaining players after a dropout and if the tournament has to be balanced, a dummy player will be added.

Definition 5.5 Robustness against Dropout. Given a set N of players, a winning probability matrix P, and a knockout tournament $KT = (T, S)$. The seeding S is robust against dropout if for any dropout of player $k \in N$, there exists a seeding S' for the remaining $|N| - 1$ players such that $\forall (i, j) \in N$, if i dominates j and $q(i, (T, S')) - q(i, (T, S)) \leq 0$ then $q(j, (T, S')) - q(j, (T, S)) \leq 0$.

It is easy to show that given a winning probability matrix, the robustness requirement is achievable for both balanced and unbalanced tournaments (e.g., the tournament shown in Figure 3). Indeed the condition is quite weak. There are other stronger conditions that can be used. For example, that no weak player gains in winning probability more than the players stronger than him (this in fact implies the requirement we proposed). Nevertheless, when we add this weak requirement to the fairness criterion, surprisingly, it is no longer always possible to find a seeding that can satisfy both (even when the winning probabilities are monotonic).

THEOREM 5.6. *For any $n = 2^k \geq 8$, there exists a set of n players with a monotonic winning probability matrix P such that it is not possible to find a seeding S such that S satisfies simultaneously the order preservation requirement and the condition for robustness against dropout.*

PROOF. For a given n, we show how to construct the winning probability matrix P. There will be 3 top players and $n-3$ dummy players. The winning probabilities between the top 3 players are: $p_{1,2} = 0.5$, $p_{1,3} = 0.6$, and $p_{2,3} = 0.5$. The dummy players will lose against the top 3 players with probability 1. Thus the tournament is reduced to a tournament between these 3 players. There are only three possible seedings: [1 *vs.* 2] *vs.* 3, [1 *vs.* 3] *vs.* 2, or [2 *vs.* 3] *vs.* 1. The first two seedings are not order preserving (e.g., for the first seeding S_1, $q(1, S_1) = 0.3 < 0.45 = q(3, S_1)$). The last seeding does not satisfy the necessary condition of robustness: when player 3 drops out, the winning probability of player 2 increases from 0.25 to 0.5, whereas the winning probability of player 1 decreases from 0.55 to 0.5. Note that this proof works for both balanced and unbalanced tournament. □

6. CONCLUSION

We have introduced two alternative criteria for fairness: envy-freeness and order preservation. We showed that when there is no constraint on the structure of the tournament, fairness can be achieved easily for both criteria. For balanced tournaments, we provided two impossibility results: one for any seeding under the first criterion, and the other for any fixed seeding under the second criterion. When the seeding can vary, depending on the actual values of the winning probabilities, we proposed a heuristic algorithm to find a seeding that satisfies order preservation. We showed that the algorithm is efficient and effective through experiments.

However, our hope of being able to find a fair seeding in all cases was dimmed by that fact that when we included a requirement of robustness, the criterion became provably impossible to satisfy for all cases. This suggests one should try to find the conditions of the winning probabilities between players that will guarantee the existence of a fair seeding. We leave this for future work.

REFERENCES

Appleton, D. R. 1995. May the best man win? *The Statistician 44,* 4, 529–538.

Fischer, F., Procaccia, A. D., and Samorodnitsky, A. 2009. A new perspective on implementation by voting trees. In *EC '09.* ACM, New York, NY, USA, 31–40.

Horen, J. and Riezman, R. 1985. Comparing draws for single elimination tournaments. *Operations Research 33,* 2 (mar), 249–262.

Hwang, F. K. 1982. New concepts in seeding knockout tournaments. *The American Mathematical Monthly 89,* 4 (apr), 235–239.

Schwenk, A. J. 2000. What is the correct way to seed a knockout tournament? *The American Mathematical Monthly 107,* 2 (feb), 140–150.

Vu, T., Altman, A., and Shoham, Y. 2009. On the complexity of schedule control problems for knockout tournaments. In *AAMAS '09.* Richland, SC, 225–232.