Manipulating Stochastically Generated Single-Elimination Tournaments for Nearly All Players

Isabelle Stanton and Virginia Vassilevska Williams
{isabelle,virgi}@eecs.berkeley.edu

Computer Science Department
UC Berkeley

Abstract. We study the power of a tournament organizer in manipulating the outcome of a balanced single-elimination tournament by fixing the initial seeding. This problem is known as *agenda control for balanced voting trees*. It is not known whether there is a polynomial time algorithm that computes a seeding for which a given player can win the tournament, even if the match outcomes for all pairwise player match-ups are known in advance. We approach the problem by giving a sufficient condition under which the organizer can *always* efficiently find a tournament seeding for which the given player will win the tournament. We then use this result to show that for most match outcomes generated by a natural random model attributed to Condorcet, the tournament organizer can very efficiently make a large constant fraction of the players win, by manipulating the initial seeding.

Introduction

The study of election manipulation is an integral part of social choice theory. Results such as the Gibbard-Satterthwaite theorem [8, 13] show that all voting protocols that meet certain rationality criteria are manipulable. The seminal work of Bartholdi, Tovey and Trick [1, 2] proposes to judge the quality of voting systems using computational complexity: a protocol may be manipulable, but it may still be good if manipulation is computationally expensive. This idea is at the heart of computational social choice.

The particular type of election manipulation that we study in this paper is called *agenda control* and was introduced in [2]: there is an election organizer who has power over some part of the protocol, say the order in which candidates are considered. The organizer would like to exploit this power to fix the outcome of the election by making their favorite candidate win. [2] focused on plurality and Condorcet voting, agenda control by adding, deleting, or partitioning candidates or voters. We study the balanced binary cup voting rule, also called *balanced voting tree*, or a *balanced single-elimination* (SE) tournament: the number of candidates is a power of 2 and at each stage the remaining candidates are paired up and their votes are compared. The losers are eliminated and the winners move on to the next round until only one candidate remains. The power of the election organizer is to pick the pairing of the players in each round. We assume that the organizer knows all the votes in advance, i.e. for any two candidates, they know which candidate is preferred. In this case, picking the pairings for each round is equivalent to picking the initial tournament seeding.

Single-elimination is prevalent in sports tournaments such as Wimbledon or March Madness. In this setting, a tournament organizer may have some information, say from prior matches or betting experts, about the winner in any possible match. The organizer creates a *seeding* of the players through which they are distributed in the tournament bracket. The question is, can the tournament organizer abuse this power to determine the winner of the tournament?

There is significant prior work on this problem. Lang et al. [10] showed that if the tournament organizer only has probabilistic information about each match, then the agenda control problem is NP-hard. Vu et al [17, 18] showed that the problem is NP-hard even when the probabilities are in $\{0, 1, 1/2\}$ and that it is NP-hard to obtain a tournament bracket that approximates the maximum probability that a given player wins within any constant factor. Vassilevska Williams [16] showed that the agenda control problem is NP-hard even when the information is deterministic but some match-ups are disallowed. [16] also gave conditions under which the organizer can always make their favorite player win the tournament with advance knowledge of each match outcome. It is still an open problem whether the agenda control problem in this deterministic setting can be solved in polynomial time.

The binary cup is a complete binary voting tree. Other related work has studied more general voting trees [9, 7], and manipulation by the players themselves by throwing games to manipulate SE tournaments [12].

The match outcome information available to the tournament organizer can be represented as a weighted or unweighted tournament graph, a graph such that for every two nodes u, v exactly one of (u, v) or (v, u) is an edge. An edge (u, v) signifies that u beats v, and a weight p on an edge (u, v) means that u will beat v with probability p. With this representation, the agenda control problem becomes a computational problem on tournament graphs.

The tournament graph structure which comes from real world sports tournaments or from elections is not arbitrary. Although the graphs are not necessarily transitive, stronger players typically beat weaker ones. Some generative models have been proposed in order to study real-world tournaments. In this work, we study a standard model in social choice theory attributed to Condorcet (see, e.g., Young [19]). The model was more recently studied by Braverman and Mossel [3]. We refer to this model as the *Condorcet Random (CR)* model[1].

The CR model has an underlying total order of the players and the outcome of every match is probabilistic. There is some *global* probability $p < 1/2$ with which a weaker player beats a stronger player. This probability represents outside factors which do not depend on the players' abilities.

[1] A previous version of this paper referred to the model as the Braverman-Mossel model.

Vassilevska Williams [16] has shown that when $p \geq 16\sqrt{\ln(n)/n}$, with high probability, the model generates a tournament graph T such that there is always a poly-time computable seeding for which *any* given player is a single-elimination tournament winner, provided all match outcomes occur as T predicts.

This result was initially surprising as the CR model is often considered to be a good model of the real world. Recent work by Russell [11] confirmed the theoretical results of [16] by giving experimental evidence that in real world instances (from tennis, basketball and hockey tournaments) one can either quickly find a winning seeding for any player, or decide that it is not possible. Russell's work uses a variant of the generalized CR model that we will define later.

The result from [16], however, was meaningful only for large n. For instance, when $n = 512$, the noise parameter $p \geq 16\sqrt{\ln(n)/n}$ is close to $1/2$, and the result is not at all surprising since then all players are essentially indistinguishable. A natural question emerges: can we still make almost all players win with a much smaller noise value? A second question is, can we relax the CR model to allow a different error probability for each pair of players, and what manipulation is then possible? We address both questions.

Finally, the CR model has been previously considered in fault-tolerant and parallel computing. For instance, Feige et al. [6] consider comparison circuits that are incorrect with probability p and develop algorithms to sort this noisy data. In particular, one of their results uses tournaments for finding the maximum in parallel. In a sense, their algorithm provides a better mechanism for finding 'the' winner (the top player in the underlying total order) in the CR model, although this mechanism may not satisfy the other nice properties of SE tournaments.

Contributions. We continue the study begun in [16] on whether one can compute a winning SE tournament seeding for a *king* player when the match outcomes are known in advance. A king is a player K such that for any other player a, either K beats a, or K beats some other player who beats a. Kings are very strong players, yet the agenda control problem for SE tournaments is not known to be polynomial-time solvable even for kings. We show that in order for a winning seeding to exist for a king, it is sufficient for the king to be among the top third of the players when sorted by the number of potential matches they can win. Before our work only much stricter conditions were known, e.g. that it is sufficient if the king beats half of the players. Our more general result allows us to obtain better results for the Condorcet random model as well.

There are $\log n$ rounds in an SE tournament over n players, so a necessary condition for a player to be a winner is that it can beat at least $\log n$ players. We consider a generalization of the Condorcet random model in which the error probabilities $p(i,j)$ can vary but are all lower-bounded by a global parameter p. The expected outdegree of the weakest player i in such a tournament is $\sum_j p(i,j) \geq p(n-1)$, and it needs to be at least $\log n$ in order for i to win an SE tournament. Thus, we focus on the case where p is $\Omega(\log n/n)$, as this is a necessary condition for all players to be winners.

We consider tournaments generated with noise $p = \Omega(\log n/n)$. The ranking obtained by sorting the players in nondecreasing order of the number of matches they can win is known to be a constant factor approximation to the Slater ranking [14, 4], and is hence a good notion of ranking in itself. We show that for almost all tournaments generated by the CR model, one can efficiently compute a seeding so that essentially the top half of the players can be made SE winners. We also show that there is a trade-off between the amount of noise and the number of players that can be made winners: as the level of noise increases, the tournament can be fixed for a larger constant fraction and eventually for all of the players. While this result does not answer the question of whether it is computationally difficult to manipulate an SE tournament in general, it does show that for tournaments we might expect to see in practice, manipulation can be quite easy.

Condorcet Random Model – Formal Definition

The premise of the Condorcet random (CR) model is that there is an implicit ranking π of the players by intrinsic abilities so that $\pi(i) < \pi(j)$ means i has strictly better abilities than j. For ease of notation, we will assume that π is the identity permutation (if not, rename the players), so that $\pi(i)$ is i. When i and j play a match there may be outside influences so that even if $i < j$, j might beat i. The CR model allows that weaker players can beat stronger players, but only with probability $p < 1/2$. Here, p is a global parameter and if $i < j$, i beats j with probability $1 - p$. A random tournament graph generated in the CR model, a *CR tournament*, is defined as follows: for every i, j with $i < j$, add edge (i, j) independently with probability $1 - p$ and otherwise add (j, i). In other words, a CR tournament is initially a completely transitive tournament where each edge is independently reversed with probability p.

We generalize the CR model to the GCR model, in which j beats i with probability $p(j, i)$, where $p \leq p(j, i) \leq 1/2$ for all i, j with $i < j$, *i.e.* the error probabilities can differ but are all lower-bounded by a global p. A random tournament graph generated in the GCR model (*GCR tournament*) is defined as follows: for every i, j with $i < j$, add edge (i, j) independently with probability $1 - p(j, i)$ and otherwise add (j, i).

Unless noted otherwise, all graphs in the paper are tournament graphs over n vertices, where n is a power of 2, and all SE tournaments are balanced. In Table 1, we define the notation used in the rest of this paper. For the definitions, let $a \in V$ be any node, $X \subset V$ and $Y \subset V$ such that X and Y are disjoint. Given a player $\mathcal{A}$, A denotes $N^{out}(\mathcal{A})$ and B denotes $N^{in}(\mathcal{A})$.

The outcome of a round-robin tournament has a natural graph representation as a tournament graph. The nodes of a tournament graph represent the players, and a directed edge (a, b) represents a win of a over b.

We will use the concept of a *king* in a graph. Although the definition makes sense for any graph, it is particularly useful for tournaments, as the highest outdegree node is always a king. We also define a *superking*, as in [16].

Definition 1. *A* king *in $G = (V, E)$ is a node $\mathcal{A}$ such that for every other $x \in V$ either $(\mathcal{A}, x) \in E$ or there exists $y \in V$ such that $(\mathcal{A}, y), (y, x) \in E$.*

Notation	
$N^{out}(a) = \{v \mid (a,v) \in E\}$	$N^{out}_X(a) = N^{out}(a) \cap X$
$N^{in}(a) = \{v \mid (v,a) \in E\}$	$N^{in}_X(a) = N^{in}(a) \cap X$
$out(a) = \lvert N^{out}(a)\rvert$	$out_X(a) = \lvert N^{out}_X(a)\rvert$
$in(a) = \lvert N^{in}(a)\rvert$	$in_X(a) = \lvert N^{in}_X(a)\rvert$
$\mathcal{H}^{in}(a) = \{v \mid v \in N^{in}(a),\ out(v) > out(a)\}$	
$\mathcal{H}^{out}(a) = \{v \mid v \in N^{out}(a), out(v) > out(a)\}$	
$\mathcal{H}(a) = \mathcal{H}^{in}(a) \cup \mathcal{H}^{out}(a)$	
$E(X,Y) = \{(u,v) \mid (u,v) \in E, u \in X, v \in Y\}$	

Table 1. A summary of the notation used in this paper.

Definition 2. *A* superking *in $G = (V,E)$ is a node $\mathcal{A}$ such that for every other $x \in V$ either $(\mathcal{A}, x) \in E$ or there exist $\log n$ nodes $y_1, \ldots, y_{\log n} \in V$ such that $\forall i,\ (\mathcal{A}, y_i), (y_i, x) \in E$.*

Kings that are also SE winners

Being a king in the tournament graph is not a sufficient condition for a player to also be able to win an SE tournament. For instance, a player may be a king by beating only 1 player who, in turn, beats all the other players. This king beats less than $\log n$ players, so it cannot win an SE tournament. [16] considered the question of how strong a king player needs to be in order for there to always exist a winning SE tournament seeding for which they win the SE tournament.

Theorem 1. *[16] Let $G = (V,E)$ be a tournament graph and let $\mathcal{A} \in V$ be a king. One can efficiently construct a winning single-elimination tournament seeding for $\mathcal{A}$ if either $\mathcal{H}^{in}(\mathcal{A}) = \emptyset$, or $out(\mathcal{A}) \geq n/2$.*

We generalize the above result by giving a condition which completely subsumes the one in Theorem 1.

Theorem 2 (Kings with High Outdegree). *Let G be a tournament graph on n nodes and $\mathcal{A}$ be a king. If $out(\mathcal{A}) \geq |\mathcal{H}^{in}(\mathcal{A})| + 1$, then one can efficiently compute a winning single-elimination seeding for $\mathcal{A}$.*

To see that the above theorem implies Theorem 1, note that if $out(\mathcal{A}) \geq n/2$, then $|\mathcal{H}^{in}(a)| \leq n/2 - 1 \leq out(\mathcal{A}) - 1$. Also, if $\mathcal{H}^{in}(\mathcal{A}) = \emptyset$ and $n \geq 2$, then $out(\mathcal{A}) \geq 1 \geq 1 + |\mathcal{H}^{in}(\mathcal{A})|$.

Theorem 2 is more general than Theorem 1. In Figure 1 we have an example of a tournament where node $\mathcal{A}$ satisfies the requirements of Theorem 2, but not those of Theorem 1. Here, $|\mathcal{H}^{in}(\mathcal{A})| = \frac{n}{4}$ and $|N^{out}(\mathcal{A})| = \frac{n}{4} + 1$. The purpose of node a is just to guarantee that $\mathcal{A}$ is a king. The example requires that each node in $N^{in}(\mathcal{A}) \setminus \mathcal{H}^{in}(\mathcal{A})$ has lower outdegree than $\mathcal{A}$; it suffices to use an outdegree-balanced[2] tournament for this set.

[2] An outdegree-balanced tournament is a tournament in which every vertex has outdegree equal to half the graph; such a tournament can easily be constructed inductively.

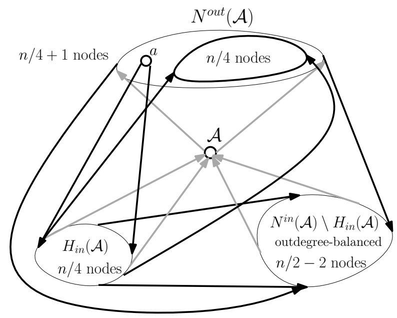

Fig. 1. An example for which Theorem 1 does not apply, but Theorem 2 does apply.

The intuition behind the proof of Theorem 2 is partially inspired by our recent results in [15]. There we show that a large fraction of highly ranked nodes can be tournament winners, provided a matching exists from the lower ranked to the higher ranked players. In this paper, we are working with a king node, and are able to weaken the matching requirement. Instead, we carefully construct matchings that maintain that $\mathcal{A}$ is a king over the graph, while eliminating the elements of $\mathcal{H}^{in}(\mathcal{A})$ until we reduce the problem to the case of Theorem 1.

We will need a technical lemma from prior work relating the indegree and outdegree of two nodes in order to prove Theorem 2. By definition, if a node $\mathcal{A}$ is a king then for every other node b, $N^{out}(\mathcal{A}) \cap N^{in}(b) \neq \emptyset$. The following lemma is useful for showing a node is a king.

Lemma 1 ([16]). *Let a be a given node, $A = N^{out}(a), B = N^{in}(a)$, $b \in B$. Then $out(a) - out(b) = in_A(b) - out_B(b)$. In particular, $out(a) \geq out(b)$ if and only if $out_B(b) \leq in_A(b)$.*

Proof of Theorem 2: We will design the matching for each consecutive round r of the tournament. In the induced graph before the r^{th} round, let $\mathcal{H}_r$ be the subset of $\mathcal{H}^{in}(\mathcal{A})$ that is still live, A_r be the current outneighborhood of $\mathcal{A}$ and B_r be the current inneighborhood of $\mathcal{A}$. We will keep the invariant that if $B_r \setminus \mathcal{H}_r \neq \emptyset$, we have $|A_r| \geq |\mathcal{H}_r| + 1$, $\mathcal{A}$ is a king and the subset of nodes from the inneighborhood of $\mathcal{A}$ that have larger outdegree than $\mathcal{A}$ is contained in $\mathcal{H}_r$.

We now assume that the invariant is true for round $r-1$. We will show how to construct round r. If $\mathcal{H}_r = \emptyset$ we are done by reducing the problem to Theorem 1, so assume that $|\mathcal{H}_r| \geq 1$. We begin by taking a maximal matching M_r from A_r to $\mathcal{H}_r$. Since $|A_r| \geq |\mathcal{H}_r| + 1$, $A_r \setminus M_r \neq \emptyset$ i.e. M_r cannot match all of A_r. Now, let M'_r be a maximal matching from $A_r \setminus M_r$ to $B_r \setminus \mathcal{H}_r$.

If $A_r \setminus (M'_r \cup M_r) \neq \emptyset$, there is some node a' leftover to match $\mathcal{A}$ to. Otherwise, pick any $a' \in M'_r \cap A_r$. Remove the edge matched to a' from M'_r and match a' with $\mathcal{A}$. To complete the matching, create maximal matchings within $\bar{A}_r = A_r \setminus (M'_r \cup M_r) \setminus \{a'\}$, $\bar{B}_r = B_r \setminus \mathcal{H}_r \setminus M'_r$ and $\mathcal{H}_r \setminus M_r$. Either 0 or 2 of $|\bar{A}_r|, |\bar{B}_r|, |\mathcal{H}_r \setminus M_r|$ can be odd and so there are at most 2 unmatched nodes that can be matched against each other. Let M be the union of these matchings.

We will now show that the invariants still hold. Notice that $\mathcal{A}$ is still a king on the sources of the created matching M. Now, consider any node b from $B_r \setminus \mathcal{H}_r$ which is a source in M. We have two choices. The first is that b survived by beating another node of B_r so it lost at least one outneighbor from B_r. Since

M'_r was maximal, b may have lost at most one of its inneighbors (a'). Hence,

$$out_{B_{r+1}}(b) + 1 \leq (out_{B_r}(b) - 1 + 1) \leq in_{A_r}(b) - 1 \leq in_{A_{r+1}}(b).$$

By Lemma 1 this means that $out(b) \leq out(\mathcal{A})$. The second choice is if b survived by beating a leftover node $\bar{a}$ from A_r. This can only happen if $A_r \setminus (M'_r \cup M_r) \neq \emptyset$. Thus, $\bar{a}$ was in $A_r \setminus (M'_r \cup M_r)$. However, since M'_r was maximal, $\bar{a}$ must lose to b, and so all inneighbors of b from A_r move on to the next round, and $out(b) \leq out(\mathcal{A})$. Thus $\mathcal{A}$ has outdegree at least as high as all nodes in $B_{r+1} \setminus \mathcal{H}_{r+1}$.

Now we consider A_{r+1} vs $\mathcal{H}_{r+1}$. We have

$$|A_{r+1}| \geq \lfloor (|A_r| + |M'_r| + |M_r| - 1)/2 \rfloor, \text{ and}$$

$$|\mathcal{H}_{r+1}| \leq \lceil (|\mathcal{H}_r| - |M_r|)/2 \rceil = \lfloor (|\mathcal{H}_r| + 1 - |M_r|)/2 \rfloor.$$

Since $|\mathcal{H}_r| \geq 1$ we must have $|M_r| \geq 1$. If either $|M_r| \geq 2$, $|A_r| \geq |\mathcal{H}_r| + 2$, or $|M'_r| \geq 1$ then it must be that $|A_{r+1}| \geq \lfloor (|\mathcal{H}_r| + 2)/2 \rfloor \geq |\mathcal{H}_{r+1}| + 1$. Also, if $|\mathcal{H}_r|$ is even then $|A_{r+1}| \geq |\mathcal{H}_r|/2 = 1 + \lfloor (|\mathcal{H}_r| - 1)/2 \rfloor \geq |\mathcal{H}_{r+1}| + 1$, and the invariant is satisfied for round $r + 1$.

On the other hand, assume that $|M_r| = 1, |M'_r| = 0, |A_r| = |\mathcal{H}_r| + 1$ and $|\mathcal{H}_r|$ is odd. This necessarily implies that $|B_r \setminus \mathcal{H}_r| \leq 1$. Since $|A_r| = |\mathcal{H}_r| + 1$ is even, $|B_r|$ must be odd and so $|B_r \setminus \mathcal{H}_r|$ must be even. $|B_r \setminus \mathcal{H}_r|$ can only be 0. This means $|\mathcal{H}_r| = n_r/2 - 1$ (where n_r is the current number of nodes). We can conclude that $\mathcal{A}$ is a king with outdegree at least half the graph and the tournament can be efficiently fixed so that $\mathcal{A}$ wins by Theorem 1. □

Theorem 2 implies the following corollaries.

Corollary 1 *Let $\mathcal{A}$ be a king in a tournament graph. If $|\mathcal{H}^{in}(\mathcal{A})| \leq (n-3)/4$, then one can efficiently compute a winning SE tournament seeding for $\mathcal{A}$.*

Corollary 2 *Let $\mathcal{A}$ be a king in a tournament graph. If $|\mathcal{H}(\mathcal{A})| \leq n/3 - 1$, then one can efficiently compute a winning SE tournament seeding for $\mathcal{A}$.*

The proof of Corollary 1 follows by the fact that if $|\mathcal{H}^{in}(\mathcal{A})| = k$, then $out(\mathcal{A}) \geq (n-k)/3$. Corollary 2 simply states that any player in the top third of the bracket who is a king is also a tournament winner.

Proof of Corollary 2: Let $K = |\mathcal{H}(\mathcal{A})|$. Then the outdegree of $\mathcal{A}$ is at least $(n - K - 1)/2$. Let $h = |\mathcal{H}^{in}(\mathcal{A})|$. By Theorem 2, a sufficient condition for $\mathcal{A}$ to be able to win an SE tournament is that $out(\mathcal{A}) \geq h + 1$. Hence it is sufficient that $n - K - 1 \geq 2h + 2$, or that $2h + K \leq n - 3$. Since $2h + K \leq 3K$, it is sufficient that $3K \leq n - 3$, and since $K \leq (n-3)/3$ we have our result. □

Condorcet Random Model

We can now apply our results to graphs generated by the CR Model. From prior work we know that if $p \geq C\sqrt{\ln n/n}$ for $C > 4$, then with probability at least $1 - 1/\text{poly}(n)$, any node in a tournament graph generated by the CR model can win an SE tournament. However, since p must be less than $1/2$, this result only applies for $n \geq 512$. Moreover, even for $n = 8192$ the relevant value of p is $> 13\%$ which is a very high noise rate. We consider how many players can be efficiently made winners when p is a slower growing function of n. We show that even when $p \geq C \ln n/n$ for a large enough constant C, a constant fraction of the top players in a CR tournament can be efficiently made winners.

Theorem 3 (CR Model Winners for Lower p). *For any given constant $C > 16$, there exists a constant n_C so that for all $n > n_C$ the following holds. Let $p \geq C \ln n/n$, and G be a tournament graph generated by the CR model with error p. With probability at least $1 - 3/n^{C/8-2}$, any node v with $v \leq n/2 - 5C\sqrt{n \ln n}$ can win an SE tournament.*

This result applies for $n \geq 256$ and also reduces the amount of noise needed. For example, if $C = 17$ then when $n = 8192$, it is only necessary that $p < 2\%$, as opposed to $> 13\%$. This is a significant improvement. The proof of Theorem 3 uses Theorem 2 and Chernoff-Hoeffding bounds.

Theorem 4 (Chernoff-Hoeffding). *Let $X_1, \ldots, X_n$ be random variables with $X = \sum_i X_i$, $E[X] = \mu$. Then for $0 \leq D < \mu$, $Pr[X \geq \mu + D] \leq exp(-D^2/(4\mu))$ and $Pr[X < \mu - D] \leq exp(-D^2/(2\mu))$.*

Proof of Theorem 3: Let C be given. Consider player j. The expectation of the number n_j of outneighbors of j in G is

$$E[n_j] = (1-p)(n-j) + (j-1)p = n(1-p) - p - j(1-2p).$$

This is exactly where we use the CR model. Our result is not directly applicable to the GCR model because this is only a lower bound on the expectation of n_j in that model. We will show that with high probability, all n_j are concentrated around their expectations and that all players $j \leq n/2$ are kings.

Showing that each n_j is concentrated around its expectation is a standard application of the Chernoff bounds and a union bound. Therefore, for $C > 16$ and $n > 2$, we have $2/n^{C^2/4} < 1/n^C$. Hence, with probability at least $1 - 1/n^{C-1}$ for every j, $|E[n_j] - n_j| \leq C\sqrt{n \ln n}$.

We assume n is large enough so that $n >> \sqrt{n \ln n}$ and that $p \leq 1/4$ so that $1 \geq (1-2p) \geq 1/2$. Now fix $j \leq n/2$. By the concentration result, this implies

$$n_j \geq 3n/4 - 1 - j - C\sqrt{n \ln n} \geq n/4 - 1 - C\sqrt{n \ln n} \geq \varepsilon n,$$

where $\varepsilon = 1/8$ works. The probability that j is a king is quite high: the probability that some node z has no inneighbor from $N^{out}(j)$ is at most

$$n(1-p)^{n_j} \leq n(1 - C \ln n/n)^{(n/(C \ln n)) \cdot C\varepsilon \ln n} \leq 1/n^{\varepsilon C - 1}.$$

By a union bound, the probability that some node j is not a king is at most $1/n^{\varepsilon C-2}$. Therefore, we can conclude that the probability that all the n_j are concentrated around their expectations and all nodes $j \leq n/2$ are kings is at least $1 - (1/n^{C-1} + 1/n^{\varepsilon C-2})$.

We now need to upper bound $|\mathcal{H}^{in}(j)|$. We are interested in how many nodes with $i < j + 2C\sqrt{n \ln n}/(1-2p)$ appear in $N^{in}(j)$: if we have an upper bound on them, we can apply Theorem 2 to get a bound on j. First, consider how small $n_j - n_i$ can be for any i:

$$n_j - n_i \geq (i-j)(1-2p) - 2C\sqrt{n \ln n}.$$

So for $i \geq j + 2C\sqrt{n \ln n}/(1-2p)$, $n_j \geq n_i$ with high probability. The expected number of nodes $i < j$ that appear in $N^{in}(j)$ is $(1-p)(j-1)$. By the Chernoff bound, the probability that at least $(1-p)(j-1) + C\sqrt{j \ln n}$ of the $j-1$ nodes less than j are in $N^{in}(j)$ is $\leq exp(-C^2 j \ln n/4j) = n^{-C^2/4}$. Therefore, with probability at least $1 - 1/n^{C^2/4}$, the number of such i is at most $(1-p)(j-1) + C\sqrt{j \ln n}$. By a union bound, this holds for all j with probability at least $1 - 1/n^{C^2/4-1}$. Now, we can say with high probability that $|\mathcal{H}^{in}(j)|$ is at most

$$(1-p)(j-1) + C\sqrt{j \ln n} + \frac{2C\sqrt{n \ln n}}{1-2p} \leq (1-p)(j-1) + 5C\sqrt{n \ln n}.$$

By Theorem 2, for there to be a winning seeding for j, it is sufficient that $\mathcal{H}^{in}(j) < n_j$ or that

$$(1-p)(j-1) + 5C\sqrt{n \ln n} < n(1-p) - p - j(1-2p) - C\sqrt{n \ln n}.$$

Rearranging the above equation, it is sufficient if

$$j < n/2 + \frac{pn}{(2(2-3p))} + \frac{(1-2p)}{(2-3p)} - 24C\sqrt{n \ln n}/5,$$

and so for all $j \leq n/2 - 5C\sqrt{n \ln n}$, there is a winning seeding for j with probability at least

$$1 - (2/n^{C-1} + 1/n^{\varepsilon C-2}) \geq 1 - 3/n^{C/8-2}.$$

□

Improving the result for the GCR model through perfect matchings.

Next, we show that there is a trade-off between the constant in front of $\log n/n$ and the fraction of nodes that can win an SE tournament. The proofs are based on the following result of Erdős and Rényi [5]. Let $B(n,p)$ denote a random bipartite graph on n nodes in each partition such that every edge between the two partitions appears with probability p.

Theorem 5 (Erdős and Rényi [5]). *Let c_n be any function of n, then consider $G = B(n,p)$ for $p = (\ln n + c_n)/n$. The probability that G contains a perfect matching is at least $1 - 2/e^{c_n}$.*

For the particular case $c_n = \Theta(\ln n)$, G contains a perfect matching with probability at least $1 - 1/\text{poly}(n)$.

Lemma 2. *Let $C \geq 64$ be a constant. Let $n \geq 16$ and G be a GCR tournament for $p = C \ln n/n$. With probability at least $1 - 2/n^{C/32-1}$, G is such that one can efficiently construct a winning SE tournament seeding for the node ranked 1.*

Proof. We will call the top ranked node s. We will show that with high probability s has outdegree at least $n/4$ and that every node in $N^{in}(s)$ has at least $\log n$ inneighbors in $N^{out}(s)$. This makes s a superking, and by [16], s can win an SE tournament.

The probability that s beats any node j is $> 1/2$, the expected outdegree of s is $> (n-1)/2$. By a Chernoff bound, the probability that s has outdegree $< n/4$ is at most $exp(-(n-1)/16) << 1/n^{C/32-1}$. Given that the outdegree of s is at least $n/4$, the expected number of inneighbors in $N^{out}(s)$ of any particular node y in $N^{in}(s)$ is at least $(n/4) \cdot (C \ln n/n) = (C/4) \ln n$.

We can show that each node in $N^{in}(s)$ has at least $\log n$ inneighbors from $N^{out}(s)$ by using a Chernoff bound and union bound. By a Chernoff bound, the probability that y has less than $(C/8) \ln n$ inneighbors from $N^{out}(s)$ is at most $exp(-(C/32) \ln n) = 1/n^{C/32}$. By a union bound, the probability that some $y \in N^{in}(s)$ has less than $(C/8) \ln n$ inneighbors from $N^{out}(s)$ is at most $1/n^{C/32-1}$. Therefore, s is a superking is with probability at least $1 - 2/n^{C/32-1}$ where $n \geq 16$, $n/4 \geq \log n, C > 64$, and $(C/8) \ln n \geq \log n$. □

Lemma 2 concerned itself only with the player who is ranked highest in intrinsic ability. The next theorem shows that as we increase the noise factor, we can fix the tournament for an increasingly large set of players. As the noise level increases, we can argue recursively that there exists a matching from $\frac{n}{2} + 1 \ldots n$ to $1 \ldots \frac{n}{2}$, and from $\frac{3n}{4} + 1 \ldots n$ to $\frac{n}{2} + 1 \ldots \frac{3n}{4}$ and so forth. These matchings form each successive round of the tournament, eliminating all the stronger players.

Theorem 6. *Let $n \geq 16$, $i \geq 0$ be a constant and $p \geq 64 \cdot 2^i \ln n/n \in [0, 1]$. With probability at least $1 - 1/poly(n)$, one can efficiently construct a winning SE seeding for any of the top $1 + n(1 - 1/2^i)$ players in a GCR tournament.*

Proof. Let G be a GCR tournament for $p = C2^i \ln n/n$, $C \geq 64$. Let S be the set of all $n/2^{i-1}$ players j with $j > n(1 - 1/2^{i-1})$. Let s be a node with $1 + n(1 - 1/2^{i-1}) \leq s \leq 1 + n(1 - 1/2^i)$. The probability that s wins an SE tournament on the subtournament of G induced by S is high: there is a set X of at least $n/2^i - 1$ nodes that are after s. By Lemma 2, s wins an SE tournament on $X \cup \{s\}$ with high probability $1 - \frac{2}{(n/2^i)^{C/32-1}}$.

In addition, by Theorem 5, with probability at least $1 - \frac{2}{(n/2^i)^{C-1}}$, there is a perfect matching from $X \cup \{s\}$ to $S \setminus (X \cup \{s\})$. For every $1 \leq k \leq i-1$, consider

$$A_k = \{x \mid 1 + n(1 - 1/2^k) \leq x\}, \text{ and}$$

$$B_k = \{x \mid 1 + n(1 - 1/2^{k-1}) \leq x \leq n(1 - 1/2^k)\}.$$

Then $A_{k-1} = A_k \cup B_k$, $A_k \cap B_k = \emptyset$, and $|A_k| = |B_k| = n/2^k$. Hence $p \geq C \ln |A_k|/|A_k|$ for all $k \leq i-1$. By Theorem 5, the probability that there is no perfect matching from A_k to B_k for a particular k is at most $2/(n/2^k)^{C2^{i-k}-1}$. This value is maximized for $k = i$, and it is $2/(n/2^i)^{C-1}$. Thus by a union bound, with probability at least $1 - 2i/(n/2^i)^{C-1} = 1 - 1/\text{poly}(n)$, there is a perfect matching from A_k to B_k, for every k.

Thus, with probability at least $1 - 1/\text{poly}(n)$, s wins an SE tournament in G with high probability, and the full bracket seeding can be constructed by taking the unions of the perfect matchings from A_k to B_k and the bracket from S. □

For the CR model we can strengthen the bound from Theorem 3 by combining the arguments from Theorems 3 and 6.

Theorem 7. *There exists a constant n_0 such that for all $n > n_0$ the following holds. Let $i \geq 0$ be a constant, and $p = 64 \cdot 2^i \ln n/n \in [0,1]$. With probability at least $1 - 1/poly(n)$, one can efficiently construct a winning seeding for any of the top $n(1 - 1/2^{i+1}) - (80/2^{i/2})\sqrt{n \ln n}$ players in a CR tournament.*

As an example, for $p = 256 \ln n/n$, Theorem 7 says that any of the top $7n/8 - 40\sqrt{n \ln n}$ players are winners while Theorem 6 only gives $3n/4 + 1$ for this setting of p in the GCR model.

Proof. As in Theorem 6, for every $1 \leq k \leq i$, consider

$$A_k = \{x \mid 1 + n(1 - 1/2^k) \leq x\}, \text{ and}$$

$$B_k = \{x \mid 1 + n(1 - 1/2^{k-1}) \leq x \leq n(1 - 1/2^k)\}.$$

Then $A_{k-1} = A_k \cup B_k$, $A_k \cap B_k = \emptyset$, and $|A_k| = |B_k| = n/2^k$. By the argument from Theorem 6, w.h.p. there is a perfect matching from A_k to B_k, for all k.

Consider A_i. By Theorem 3, with probability $1 - 1/\text{poly}(n)$, we can efficiently fix the tournament for any of the first $n/2^{i+1} - 80\sqrt{(n/2^i)\ln(n/2^i)}$ nodes in A_i. Combining the construction with the perfect matchings between A_k and B_k, we can efficiently construct a winning tournament seeding for any of the top

$$n - \frac{n}{2^i} + \frac{n}{2^{i+1}} - 80\sqrt{\frac{n}{2^i}\ln(\frac{n}{2^i})} \geq n(1 - \frac{1}{2^{(i+1)}}) - \frac{80}{2^{i/2}}\sqrt{n \ln n} \text{ nodes.}$$

□

Conclusions

In this paper, we have shown a tight bound (up to a constant factor) on the noise needed to fix an SE tournament for a large fraction of players when the match outcomes are generated by the CR model. As this model is believed to be a good model for real-world tournaments, this result shows that many tournaments in practice can be easily manipulated. In some sense, this sidesteps the question of whether it is NP-hard to fix a tournament in general by showing that it is easy on examples that we care about.

Acknowledgments. The authors are grateful for the detailed comments from the anonymous reviewers of WSCAI. The first author was supported by the NDSEG and NSF Graduate Fellowships and NSF Grant CCF-0830797. The second author was supported by the NSF under Grant #0963904 and under Grant #0937060 to the CRA for the CIFellows Project. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the NSF or the CRA.

References

1. J. Bartholdi, C. Tovey, and M. Trick. The computational difficulty of manipulating an election. *Social Choice Welfare*, 6(3):227–241, 1989.
2. J. Bartholdi, C. Tovey, and M. Trick. How hard is it to control an election. *Mathematical and Computer Modeling*, pages 27–40, 1992.
3. M. Braverman and E. Mossel. Noisy sorting without resampling. In *SODA*, pages 268–276, 2008.
4. D. Coppersmith, L. Fleischer, and A. Rudra. Ordering by weighted number of wins gives a good ranking for weighted tournaments. In *SODA*, pages 776–782, 2006.
5. P. Erdős and A. Rényi. On random matrices. *Publications of the Mathematical Institute Hungarian Academy of Science*, 8:455–561, 1964.
6. U. Feige, D. Peleg, P. Laghavan, and E. Upfal. Computing with unreliable information. In *STOC*, pages 128–137, 1990.
7. F. Fischer, A. D. Procaccia, and A. Samorodnitsky. On voting caterpillars:approximating maximum degree in a tournament by binary trees. In *COMSOC*, 2008.
8. A. Gibbard. Manipulation of voting schemes: a general result. *Econometrica*, 41, 1973.
9. N. Hazon, P.E. Dunne, S. Kraus, and M. Wooldridge. How to rig elections and competitions. In *COMSOC*, 2008.
10. J. Lang, M. S. Pini, F. Rossi, K. B. Venable, and T. Walsh. Winner determination in sequential majority voting. In *IJCAI*, 2007.
11. T. Russell. A computational study of problems in sports. *University of Waterloo PhD Disseration*, 2010.
12. T. Russell and T. Walsh. Manipulating tournaments in cup and round robin competitions. In *Algorithmic Decision Theory*, 2009.
13. M. A. Satterthwaite. Strategy-proofness and arrow's conditions: Existence and correspondence theorems for voting procedures and social welfare functions. *Journal of Economic Theory*, 10, 1975.
14. P. Slater. Inconsistencies in a schedule of paired comparisons. *Biometrika*, 48(3/4):303–312, 1961.
15. I. Stanton and V. Vassilevska Williams. Rigging tournament brackets for weaker players. *IJCAI*, 2011.
16. V. Vassilevska Williams. Fixing a tournament. In *AAAI*, pages 895–900, 2010.
17. T. Vu, A. Altman, and Y. Shoham. On the complexity of schedule control problems for knockout tournaments. In *AAMAS*, 2009.
18. T. Vu, N. Hazon, A. Altman, S. Kraus, Y. Shoham, and M. Wooldridge. On the complexity of schedule control problems for knock-out tournaments. *JAIR*, 2010.
19. H. P. Young. Condorcets theory of voting. *The American Political Science Review*, 82(4):1231–1244, 1988.