Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence (IJCAI-21)
Survey Track

Tournaments in Computational Social Choice: Recent Developments

Warut Suksompong
School of Computing, National University of Singapore
warut@comp.nus.edu.sg

Abstract

Tournaments are commonly used to select winning alternatives in scenarios involving pairwise comparisons such as sports competitions and political elections. This survey discusses recent developments in two major lines of work—tournament solutions and single-elimination tournaments—with a focus on how computational social choice has brought new frameworks and perspectives into these decades-old studies.

1 Introduction

The theory of social choice is primarily concerned with choosing a socially desirable outcome from a given set of alternatives. In many practical scenarios, these decisions are made based on pairwise comparisons between alternatives, also known as *tournaments*. Problems pertaining to tournaments, and more generally to collective decision making, often involve the use of algorithms and have therefore attracted significant attention from computational social choice researchers over the past two decades [Brandt *et al.*, 2016b]. Much of the relevant research has been published in major artificial intelligence venues, including IJCAI.

A familiar example of tournaments arises in sports competitions, where players or teams compete against each other in head-to-head matches in order to determine the winner of the competition. Several sports competitions, including Grand Slam tennis, NCAA basketball, and FA Cup football, are run using *single-elimination tournaments*, also known as *knockout tournaments*. While sports fans love this tournament format for its "do-or-die" nature, the fact that not all pairs of players play each other in such a tournament raises several interesting questions. If the organizers want to help a certain player win the tournament, can they set up the tournament bracket to achieve that goal? What if the organizers can also bribe a limited number of players to intentionally lose?

Beyond sports competitions, tournaments serve to model a number of scenarios ranging from voting [Laslier, 1997] to webpage ranking [Brandt and Fischer, 2007] to biological interactions [Allesina and Levine, 2011]. In order to select the best alternatives according to pairwise comparisons, numerous methods—known as *tournament solutions*—have been proposed in the literature. Although most early work on tournament solutions was based on the axiomatic approach, with the recent advent of computational social choice, the solutions have also been intensively examined from the algorithmic and complexity-theoretic viewpoints. These fresh perspectives have in turn given rise to exciting new questions, methods, and frameworks for analysis.

Research in computational social choice on knockout tournaments and tournament solutions has been surveyed in extensive book chapters by Vassilevska Williams [2016] and Brandt *et al.* [2016a], respectively. The present survey expands and complements the two chapters from the Handbook of Computational Social Choice [Brandt *et al.*, 2016b] by focusing on work that appeared in the five years or so since the publication of these chapters, particularly in AI venues.

2 Single-Elimination Tournaments

In this section, we survey work on single-elimination (SE) tournaments. While there exist unbalanced SE tournaments (in extreme cases, a certain player proceeds directly to the final), the vast majority of real-life SE tournaments are balanced, and most of the research on this subject assumes balanced brackets. We will therefore restrict our attention to balanced SE tournaments.

2.1 Preliminaries

Let n denote the number of players in the tournament, where we assume for simplicity that $n = 2^r$ for some positive integer r. A (balanced) SE tournament is represented by a balanced binary tree with n leaves corresponding to the players. The assignment of the players to the leaves is called a *bracket*. The winner of the tournament is determined recursively: the winner of a leaf is the player at the leaf, and the winner of a subtree rooted at node u is the winner of the match between the winners of the subtrees rooted at the two children of u. An example of a SE tournament is shown in Figure 1.

In order to determine the outcome of each match, we assume that there is an underlying graph, called a *tournament graph*.[1] For any pair of players, the tournament graph indicates which player would win if they were to play each other. An important question in tournament fixing is whether, given

[1] In Section 3, a tournament graph will be referred to simply as a *tournament*, but in this section we use this terminology to distinguish it from a SE tournament.

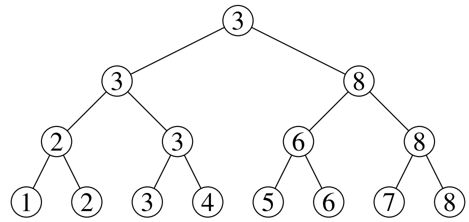

Figure 1: Example of a single-elimination tournament with 8 players

the knowledge of the tournament graph, the tournament organizers can choose a bracket to help their favorite player win.

Definition 2.1 (Tournament fixing problem)**.** Given a set of players, a tournament graph over the players, and a player of interest v, the *tournament fixing problem* (TFP) asks whether there exists a bracket such that v wins the tournament.

The complexity of TFP had been posed as an open question in several papers, before it was finally resolved by Aziz *et al.* [2018] via a complex reduction from a version of 3SAT.

Theorem 2.2 ([Aziz *et al.*, 2018])**.** TFP *is* NP*-complete.*

Kim and Vassilevska Williams [2015] strengthened this result by proving that the problem remains NP-complete even when the player of interest v is a "king" that beats $n/4$ other players in the tournament graph—a player v is said to be a *king* if v has distance at most 2 to any other player in the graph (i.e., for every other player v', either v beats v', or v beats another v'' who beats v'). Despite these hardness results, several interesting questions remain, such as identifying conditions under which a winning bracket for a player exists and can be found efficiently, or designing parameterized algorithms. We summarize recent results in some of these directions next.

2.2 Algorithms

Consider the problem #TFP of computing the number of brackets under which a particular player can win the tournament. Since an algorithm for #TFP can be used to solve TFP (by simply checking whether the answer is zero or not), #TFP is also NP-hard. Aziz *et al.* [2018] showed that even randomization cannot help: there is no fully polynomial-time randomized approximation scheme (FPRAS) unless NP is equal to RP, the complexity class consisting of problems that can be solved in randomized polynomial time. In spite of this, it is possible to solve #TFP faster than the brute-force solution.

Theorem 2.3 ([Kim and Vassilevska Williams, 2015])**.** #TFP *can be solved in time* $O(2^n\text{poly}(n))$.

The approach of Kim and Vassilevska Williams [2015] is recursive: to compute the number of winning brackets for a player v, it considers all possible ways of partitioning the set of players S into two subsets T and $S \setminus T$ of equal size such that $v \in T$, iterate over all players $w \in S \setminus T$ beaten by v, and compute the number of winning brackets of v in T and w in $S \setminus T$. At the core of their algorithm is a subroutine based on the ideas of Björklund *et al.* [2007] on fast subset convolution. This subroutine ensures that given functions f, g taking subsets of size $k/2$ of some ground set, a function h taking subsets of size k defined by

$$h(S) = \sum_{\substack{T \subseteq S \\ |T|=k/2}} f(T) \cdot g(S \setminus T),$$

and oracle access to f and g, computing $h(S)$ for all S of size k takes time $O(k \cdot 2^n)$. The resulting algorithm for #TFP also takes $O(2^n\text{poly}(n))$ space. Aziz *et al.* [2018] gave an algorithm that offers a range of time-space trade-offs: when optimizing for time, their algorithm takes time $O(2.83^n)$ and space $O(1.75^n)$, while if space is limited, the algorithm requires time $4^{n+o(n)}$ but space only poly(n).

As polynomial-time algorithms for TFP are not to be expected in general, a natural direction is to identify tractable cases. Aziz *et al.* [2018] showed that the problem is efficiently solvable when the players form a constant number of types, where the outcome of a match between two players depends only on the players' types. Another useful parameter of the problem is the size of a smallest *feedback arc set*, i.e., a set of edges whose removal leaves the tournament acyclic. Aziz et al. gave a dynamic programming algorithm running in time $n^{O(k)}$, where k denotes this size. Ramanujan and Szeider [2017] improved upon this result by showing that TFP is in fact fixed-parameter tractable (FPT) with respect to k; in particular, they presented an algorithm running in time $2^{O(k^2 \log k)}n^{O(1)}$. Their algorithm relies on translating TFP into an algebraic system of equations and feeding it into an integer linear programming (ILP) solver. This running time was later improved by Gupta *et al.* [2018a].

Theorem 2.4 ([Gupta *et al.*, 2018a])**.** *TFP can be solved in time* $2^{O(k \log k)}n^{O(1)}$*, where* k *is the size of the smallest feedback arc set.*

Besides a faster running time, the algorithm of Gupta *et al.* [2018a] has the advantage that it is combinatorial in nature and does not rely on any "black box" like an ILP solver. At a high level, it first guesses a "template tree" and then fills up the paths and subtrees that are not already determined by the guess in a greedy manner. Later on, Gupta *et al.* [2019] provided the first "polynomial kernelization" for TFP parameterized by k: given any instance of TFP, their kernelization returns an equivalent SAT encoding whose size (i.e., the number of clauses and variables) is polynomial in k and independent of the size of the original tournament. Aronshtam *et al.* [2017] also studied related problems through the lens of parameterized complexity. Nevertheless, it remains interesting in future work to determine whether TFP is FPT with respect to other parameters, for example the size of a minimum *feedback vertex set*, i.e., a set of vertices whose removal renders the tournament acyclic.

2.3 Structural Results

Since some players may not be able to win a SE tournament regardless of the bracket, a line of work has provided conditions under which a player is a SE winner for at least one bracket. An example of this is the "superking" condition: a king v is said to be a *superking* if for every player v' beating v, there exist at least $\log_2 n$ players v'' such that v beats v'' and v'' beats v. Another condition is that of a *king of high outdegree*, i.e., a king with outdegree d who loses to fewer than d players with outdegree larger than d. Vassilevska Williams [2010] and Stanton and Vassilevska Williams [2011] showed that a superking and a king of high outdegree can always win a SE tournament, respectively. Both of these results were later generalized in one fell swoop by Kim *et al.* [2017].

Theorem 2.5 ([Kim *et al.*, 2017])**.** *Let v be a king, and denote by A and B the set of remaining players who lose to v and beat v, respectively. Suppose that B is a disjoint union of three (possibly empty) sets H, I, J such that*

1. $|H| < |A|$;
2. *Every player in I loses to at least $\log_2 n$ players in A;*
3. *Every player in J beats at most $|A|$ players.*

Then, v is a SE winner. Moreover, there exists a polynomial-time algorithm that computes a winning bracket for v.

Intuitively, each of the three sets H, I, J are weak in some respect. The set H has small size, each player in I can be reached from v via a non-negligible number of paths of length two, and players in J have relatively low outdegrees. Note that the superking result corresponds to taking $H = J = \emptyset$, while the "king of high outdegree" result corresponds to taking $I = \emptyset$. The proof of this theorem proceeds by induction. At each stage, the goal is to construct a matching of the players such that for the subsequent round of the tournament (with half of the players remaining), v is still a king and the three conditions remain fulfilled. The matching is constructed by first considering a maximum matching from A to H, then a maximum matching from the remaining players in A to $I \cup J$, and finally performing arbitrary matchings within each of the sets A, H, and $I \cup J$.

In addition to kings, another notion that has featured in structural results is that of 3-*kings*, i.e., players who can reach any other player via a path of length at most three. Perhaps surprisingly, 3-kings are significantly weaker than kings with respect to SE tournaments: even though a king that beats at least $n/2$ players is always a SE winner,[2] a 3-king may not be able to win the tournament even if it beats $n - 3$ players [Kim and Vassilevska Williams, 2015, Figure 2]. Kim and Vassilevska Williams [2015] and Kim *et al.* [2017] provided different sets of conditions under which a 3-king can win a SE tournament. However, these conditions are rather restrictive compared to the conditions for kings in Theorem 2.5, which leaves the question of whether a 3-king is guaranteed to be a SE winner under more general conditions.

2.4 Bribery

As Gupta *et al.* [2018b] noted, sometimes one wants to win a SE tournament "by any means necessary". Besides fixing the tournament bracket, a common way to achieve this goal is to bribe certain players to intentionally lose a match. This setting was first studied by Kim and Vassilevska Williams [2015], who defined the problem of Bribery-TFP (BTFP), where in addition to choosing the bracket, the organizers can bribe up to b players to lose a match that they would otherwise win.[3] If $b = 0$, then BTFP is equivalent to TFP which is NP-hard (Theorem 2.2), whereas if $b = \log_2 n$, then the tournament can easily be rigged by bribing all players that our favorite player faces. By using a reduction from TFP, Kim and Vassilevska Williams showed that the problem is computationally intractable in between.[4]

Theorem 2.6 ([Kim and Vassilevska Williams, 2015])**.** *For any constant $\varepsilon > 0$,* BTFP *is* NP*-hard when $b \leq (1 - \varepsilon) \log_2 n$.*

Gupta *et al.* [2019] presented "obfuscation operations" which can take in one bribery solution and output another solution in polynomial time. Their operations are relevant when the bribery in the given solution is too conspicuous or cannot be realized (e.g., when some players refuse to be bribed). In addition, they gave an exact algorithm for BTFP running in time $2^{O(k^2 \log k)} n^{O(1)}$, where k denotes the size of a smallest feedback arc set—this implies that like the version without bribery, BTFP is FPT with respect to k. Nevertheless, the running time of this algorithm is worse than that of the corresponding algorithm for TFP (Theorem 2.4), where the dependence on k in the exponent is linear instead of quadratic.

Konicki and Vassilevska Williams [2019] studied bribery in SE tournaments using models involving probabilities; we discuss their contributions in the next subsection.

2.5 Probabilistic Approaches

In real-life tournaments, players have varying strengths, and therefore not all tournament graphs are equally likely to occur. A model for generating tournament graphs that has been studied in several papers is the *Condorcet random model*. In this model, there is a linear ordering of players from strongest to weakest, and a probability $p < 1/2$. For each pair of players, the stronger player beats the weaker player with probability $1 - p$ (so the weaker player wins with probability p), independently of other pairs. If $p \in o(\log n/n)$, then the weakest player is expected to win only $o(\log n)$ matches—this is insufficient to be a SE winner since such a winner must beat $\log_2 n$ players. On the other hand, Vassilevska Williams [2010] showed that when $p \in \Omega(\sqrt{\log n/n})$, it is likely that every player can win a SE tournament under some bracket, leaving a gap of roughly $\Theta(\sqrt{n})$. This gap was closed by Kim *et al.* [2017], who established that $\Omega(\log n/n)$ is the threshold where the transition occurs. Manurangsi and Suksompong [2021] extended this result to the *generalized random model*: for each pair of players i and j, player i beats player j with probability $p_{i,j}$ independently of other pairs, where $p_{i,j} + p_{j,i} = 1$ for all $i \neq j$.

Theorem 2.7 ([Manurangsi and Suksompong, 2021])**.** *Assume that the tournament is generated according to the generalized random model with $p_{i,j} \in \Omega(\log n/n)$ for all i, j. With high probability, for each player, there exists a bracket under which the player wins the SE tournament.*

[2] This follows by taking $H = B$ and $I = J = \emptyset$ in Theorem 2.5.

[3] Bribery has also been studied in other settings, perhaps most notably in voting [Faliszewski and Rothe, 2016].

[4] The setting where the bracket is given in advance but bribery is allowed has also been studied [Russell and Walsh, 2009; Mattei *et al.*, 2015].

The Condorcet random model was also studied by Konicki and Vassilevska Williams [2019] in the context of bribery. These authors showed that for any p, it is sufficient to bribe the top $O(\log n)$ players in the linear ordering to make any player win the SE tournament with high probability. Their proof relies on Theorem 2.5 in order to establish the existence of winning brackets.

So far, we have assumed that the tournament graph is deterministic, i.e., if a pair of players were to play against each other, it is already known with certainty who would win. In particular, even though the Condorcet random model provides a randomized way to generate a tournament graph, the resulting graph is deterministic. A more general version of TFP encodes the uncertainty into the tournament graph itself [Vu *et al.*, 2009]. Specifically, for each pair of players i and j, the probability that i beats j is given by $q_{i,j}$. This generalization of TFP is sometimes referred to as the *probabilistic tournament fixing problem* (PTFP).

Chatterjee *et al.* [2016] investigated the robustness of SE brackets in PTFP. They considered perturbing each entry of the probability matrix by at most ε, and asked how much the probability of a certain player winning the tournament can drop through such perturbations. Intriguingly, they showed that the robustness can vary significantly across brackets.

Theorem 2.8 ([Chatterjee *et al.*, 2016])**.** *There exist deterministic tournament graphs such that for one winning bracket of a player, the winning probability can drop by $\Theta(\varepsilon n)$ through ε-perturbations, whereas for another winning bracket of the same player, the drop is only $\Theta(\varepsilon \log n)$.*

Konicki and Vassilevska Williams [2019] also considered probabilistic tournament graphs. They showed that even when the probability matrix is "monotonic", i.e., $q_{i,j} \geq q_{i,j-1}$ for all $i \leq j-2$, the probabilistic version of BTFP is NP-hard. An interesting open question is whether the probabilistic version of TFP is also computationally hard for monotonic probability matrices.

3 Tournament Solutions

In this section, we address work on tournament solutions. Although research on this topic in the last few years has arguably been less coherent than on single-elimination tournaments, some new frameworks and perspectives have emerged, which have opened up intriguing directions for future work.

3.1 Preliminaries

A *tournament* is defined by a set of *alternatives* and a *dominance relation* between the alternatives: for every pair of alternatives, one dominates the other. The tournament can also be viewed as a directed graph $T = (V, E)$, with the vertices in V corresponding to the alternatives and the edges in E to the dominance relation. The *Copeland score* of an alternative x is the number of alternatives that x dominates, i.e., the outdegree of x in T. For alternatives x and y, we write $x \succ y$ to denote that x dominates y, and for sets of alternatives X and Y, we write $X \succ Y$ to mean that $x \succ y$ for all $x \in X$ and $y \in Y$. An example of a tournament is shown in Figure 2.

A *tournament solution* is a function that maps each tournament to a nonempty subset of its alternatives, usually called the set of *winners* or the *choice set*. Some common tournament solutions are listed below.

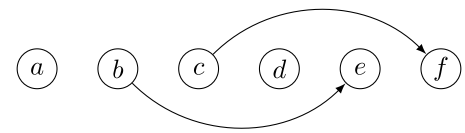

Figure 2: A tournament T with 6 alternatives. All omitted edges are assumed to point from right to left. The Copeland score of e is 3.

- The *Copeland set (CO)* is the set of alternatives with the largest Copeland score.
- The *top cycle* (TC) is the (unique) smallest nonempty set W of alternatives such that $W \succ V(T) \setminus W$.
- The *uncovered set* (UC) is the set of alternatives that are not "covered" by any other alternative, where an alternative x is said to *cover* another alternative y if $x \succ y$ and for any alternative z such that $y \succ z$, we also have $x \succ z$. Equivalently, the uncovered set coincides with the set of kings from Section 2.
- The *Banks set* (BA) is the set of alternatives that appear as the maximal element of some maximal transitive subtournament.

The containment relations $CO(T) \subseteq UC(T) \subseteq TC(T)$ and $BA(T) \subseteq UC(T)$ hold for any tournament T [Laslier, 1997]. For the tournament T in Figure 2, we have $CO(T) = \{f\}$, $TC(T) = \{b, c, d, e, f\}$, and $UC(T) = BA(T) = \{c, d, e, f\}$.

3.2 Query Complexity

While much of the work on tournament solutions in computational social choice has focused on the computational complexity, it is also interesting to examine these solutions from a *query complexity* perspective. This study was conducted by Dey [2017]. In the query model, instead of the tournament being given as an input, an algorithm has to make (deterministic) queries in order to discover it. With each query, the algorithm can find out the orientation of a desired edge. The goal of the algorithm is to compute the set of winners for each tournament solution using the minimum number of queries in the worst case. Unlike the computational complexity, which can be exponential, the query complexity is always at most $O(n^2)$, since the entire tournament is described by $O(n^2)$ edges. Unfortunately, Dey showed that for all of the tournament solutions above, making $\Theta(n^2)$ queries is inevitable.

Theorem 3.1 ([Dey, 2017])**.** *Any deterministic algorithm that computes the Copeland set, the top cycle, the uncovered set, or the Banks set of a tournament must make $\Omega(n^2)$ queries in the worst case.*

To demonstrate the idea of Dey's proof, consider the top cycle. Assume for simplicity that $n = 4s + 2$ for some positive integer s. We partition the set of alternatives into two sets $A = \{a_1, \ldots, a_{2s+1}\}$ and $B = \{b_1, \ldots, b_{2s+1}\}$. An alternative a_i dominates the alternatives $a_{i+1}, a_{i+2}, \ldots, a_{i+s}$, where the indices are taken modulo $2s+1$; an analogous dominance

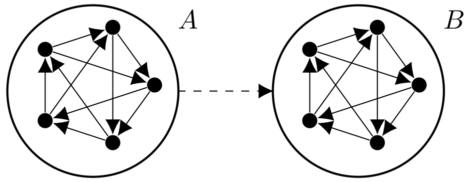

Figure 3: An illustration of the proof of Theorem 3.1 for TC

relation holds within the set B. (See Figure 3 for an illustration of the case $s = 2$.) For any algorithm, we will construct an adversary that forces the algorithm to make $\Omega(n^2)$ queries as follows. When the algorithm makes a query, if the query is within the set A or the set B, the adversary answers according to the specified relations. Else, the query is between alternatives $a \in A$ and $b \in B$—in this case, the adversary answers that a dominates b. We claim that the algorithm must query all $(2s+1)^2 \in \Theta(n^2)$ edges between A and B. Indeed, suppose that some such edge is left unqueried. If the algorithm answers that at least one vertex of B is included in TC, the adversary orients all edges from A towards B, thereby making the top cycle a subset of A. Otherwise, the algorithm answers that the top cycle is a subset of A. In that case, the adversary orients all unqueried edges between A and B from B to A. As a result, every alternative in B can reach all other alternatives, meaning that the top cycle contains B. The proofs for CO, UC, and BA use similar ideas.

Despite these negative results, Dey [2017] showed that if the top cycle is small, then it is possible to compute all of these tournament solutions with fewer queries. Specifically, if the size of TC of a given tournament T is at most t, then there exists an algorithm for computing each of the above tournament solutions using $O\left(nt + \frac{n \log n}{\log(1-1/t)}\right)$ queries. Identifying other parameters that make the problem tractable is an interesting direction for future research.

3.3 Margin of Victory

Given that the purpose of tournament solutions is to distinguish the stronger alternatives from the weaker ones, it is perhaps surprising that most common tournament solutions tend to select all alternatives in large random tournaments. Indeed, Fey [2008] and Scott and Fey [2012] showed that the top cycle, the uncovered set, and the Banks set are unlikely to exclude any alternative from a tournament drawn according to the *uniform random model*, wherein each edge is oriented in either direction with probability $1/2$ independently of other edges.[5] Moreover, choosing a large set of alternatives is sometimes unavoidable if certain properties of the tournament solutions are desired. For example, Brandt *et al.* [2018] proved that any tournament solution satisfying a normatively appealing property called "stability" (including the top cycle) must select at least half of the alternatives on average.

In order to differentiate among the winning alternatives of any given tournament, Brill *et al.* [2020] proposed a generic framework for refining tournament solutions based on a concept called *margin of victory* (MoV).[6] The MoV of a winning alternative captures how close it is to dropping out of the winner set, where distance is measured in terms of the number of edges that need to be reversed. This notion can also be viewed in terms of bribery, specifically as the amount of bribe that must be paid to other players or the referees in order to obtain the desired outcome.[7] Brill et al. investigated the complexity of computing the MoV and established bounds on its value for several tournament solutions.

Theorem 3.2 ([Brill *et al.*, 2020]). *Computing the* MoV *for winners with respect to CO, TC, and UC can be done in polynomial time, whereas the corresponding problem for BA is* NP*-hard. For all of these tournament solutions, the* MoV *can be as high as $\lfloor n/2 \rfloor$, but no higher.*

For all four tournament solutions, the bound $\lfloor n/2 \rfloor$ is attained by an alternative that dominates the remaining $n-1$ alternatives, when these $n-1$ alternatives form a subtournament with the structure described for each of the sets A and B following Theorem 3.1 (see Figure 3).

In a follow-up paper, the same authors examined the MoV notion from the structural and experimental perspectives [Brill *et al.*, 2021]. On the structural side, they defined the axioms of *cover-consistency* and *(strong) degree-consistency*. For a tournament solution S, we say that MoV_S is cover-consistent if for any alternatives x and y such that x covers y, we have $\text{MoV}_S(x) \geq \text{MoV}_S(y)$. We say that MoV_S is strong degree-consistent if the same inequality holds for any x, y such that the Copeland score of x is at least that of y.

Theorem 3.3 ([Brill *et al.*, 2021]). *For each tournament solution $S \in \{CO, TC, UC, BA\}$, MoV_S satisfies cover-consistency. On the other hand, among these tournament solutions, only MoV_{TC} satisfies strong degree-consistency.*[8]

Theorem 3.3 implies that the margin of victory is typically aligned with the covering relation, an important indicator of strength of alternatives. Moreover, for most of the tournament solutions, the MoV provides information on the tournament under consideration beyond simply the outdegrees. Besides these structural contributions, Brill *et al.* [2021] also performed experiments demonstrating how the discriminative power of the MoV varies across tournament solutions.

3.4 Relationship to Single-Elimination Winners

The set of single-elimination winners considered in Section 2 can also be viewed as a tournament solution. In this light, it is natural to explore its relation to traditional tournament solutions. For example, can a player chosen by a certain tournament solution always win a SE tournament? How large is

[5]The uniform random model is a special case of the Condorcet random model where $p = 1/2$. Saile and Suksompong [2020] extended some of these results to the generalized random model (cf. Theorem 2.7).

[6]Similar concepts of the same name have been applied in other domains including voting [Xia, 2012], sports modeling [Kovalchik, 2020], and political districting [Stoica *et al.*, 2020].

[7]Brill *et al.* [2020] also defined the MoV for non-winning alternatives, and considered a more general setting where reversing different edges may have unequal costs. Related problems have been studied by Faliszewski *et al.* [2009], Russell and Walsh [2009], Aziz *et al.* [2015], and Yang and Guo [2017].

[8]MoV_{CO} satisfies a weaker version of degree-consistency.

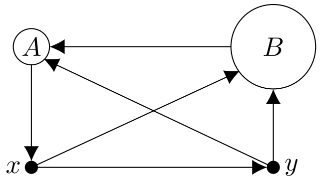

Figure 4: Example of a tournament in which the uncovered set and the set of single-elimination winners are largely disjoint.

the overlap between the set of SE winners and the set of winners according to other tournament solutions? The answers to some of these questions were given by Kim *et al.* [2017].

Theorem 3.4 ([Kim *et al.*, 2017])**.** *Any alternative in CO is a SE winner. On the other hand, for any constant $c \in (0, 1)$, there exists a tournament such that the proportion of alternatives in UC that are SE winners is less than c, and the proportion of SE winners that are in UC is also less than c.*

The proof of the CO claim is succinct: any alternative in the Copeland set is a king (i.e., contained in UC) and dominates at least $n/2$ other alternatives, so we may directly apply Theorem 2.5 with $I = J = \emptyset$. For UC, consider the tournament T in Figure 4, where A and B are subtournaments. Since all alternatives in B are covered by y while the remaining alternatives are uncovered, we have $UC(T) = A \cup \{x, y\}$. If B is exponentially larger than A, the players in A cannot win enough matches to win a SE tournament. On the other hand, one can check that all players in B are SE winners provided that they have roughly equal strength. This means that the vast majority of SE winners are in B.

While any alternative with the maximum Copeland score can win a SE tournament, there can be SE winners with low Copeland scores. An extreme example is a player who can beat only $\log n$ other players and wins a SE tournament by beating precisely these players. An interesting question arises from a randomized point of view: is it true that for every tournament graph, the winner according to a bracket chosen uniformly at random must have a high Copeland score? Hulett [2019] answered this question in the negative.

Theorem 3.5 ([Hulett, 2019])**.** *There exists a tournament graph such that the SE winner according to a uniformly random bracket has Copeland score $n \cdot 2^{-\Theta(\sqrt{\log n})}$.*

For comparison, choosing a uniformly random alternative from any tournament already yields a much higher Copeland score of $(n - 1)/2$, and the quantity $n \cdot 2^{-\Theta(\sqrt{\log n})}$ is still lower than, say, $\Theta(n/\log n)$. Hulett's proof is based on constructing a probability distribution over a class of tournament graphs for which weak alternatives win a SE tournament under most brackets. Her result shows that as an indicator of alternative strength, the ability to win a SE tournament does not necessarily align with the Copeland score.

3.5 Randomized Tournament Solutions

For this final subsection, we turn our attention to variants of tournament solutions where instead of returning a subset of the alternatives, our functions return a probability distribution over them. We refer to such functions as *randomized tournament solutions*. A desirable property in this context is *Condorcet-consistency*, which means that an alternative that dominates all other alternatives should receive probability 1. Indeed, it would be strange if a player wins all of her matches and yet leaves the competition empty-handed. Another important consideration is strategyproofness—no group of competitors should be able to significantly improve the probability that one of them wins the tournament, by fixing the outcomes of their matches. If a group of size k cannot increase their combined probability by more than α, we say that the function is *k-Strongly-Non-Manipulable-α* (k-SNM-α).

When $k = 2$, no Condorcet-consistent randomized tournament solution can be 2-SNM-α for any $\alpha < 1/3$. To see this, consider a tournament with three players a, b, c such that a beats b, b beats c, and c beats a. Regardless of the randomized tournament solution, some pair of players necessarily receive a combined probability of at most $2/3$. However, this pair of players can reverse their match outcome and increase their probability to 1 due to Condorcet-consistency. Schneider *et al.* [2017] showed that the bound $1/3$ can be attained via a simple rule central to the present survey.[9]

Theorem 3.6 ([Schneider *et al.*, 2017])**.** *A uniformly random SE bracket is* 2-SNM-$1/3$.

The proof of Theorem 3.6 uses a clever coupling argument that ties a bracket where a pair of players can potentially gain from manipulation with two other brackets where no manipulation potential for this pair exists. Schneider et al. also showed that several other formats are either 2-SNM-$1/2$ or worse, making the high resistance to manipulation of random SE tournaments all the more striking.

For $k \geq 2$, Schneider *et al.* [2017] showed that no rule is k-SNM-α for $\alpha < \frac{k-1}{2k-1}$, and conjectured that this is tight. The conjecture was refuted by Schvartzman *et al.* [2020], who proved that no rule is k-SNM-$1/2$ for large enough k, and established the existence of a k-SNM-$2/3$ rule for all k.

4 Conclusion

The study of tournaments has given rise to a rich and fascinating literature, and several exciting directions remain for future research. In particular, it would be interesting to investigate other tournament formats in greater depth, for instance double-elimination [Stanton and Vassilevska Williams, 2013; Aziz *et al.*, 2018], round-robin, stepladder [Yang and Dimitrov, 2021], Swiss-system, multi-stage tournaments, as well as those involving promotion and relegation features. Other important avenues include performing empirical studies on real-world tournaments, for example using data from sports competitions [Russell and van Beek, 2011; Mattei and Walsh, 2016; Eidelstein *et al.*, 2019], and examining the effects of the tournament structure on fairness [Ryvkin and Ortmann, 2008; Suksompong, 2016; Arlegi and Dimitrov, 2020].

Acknowledgments

The author acknowledges support from an NUS Start-up Grant and thanks the reviewers for their suggestions.

[9] Ding and Weinberg [2021] generalized this result to tournament graphs in which the outcomes are not necessarily deterministic.

References

[Allesina and Levine, 2011] Stefano Allesina and Jonathan M. Levine. A competitive network theory of species diversity. *Proceedings of the National Academy of Sciences*, 108(14):5638–5642, 2011.

[Arlegi and Dimitrov, 2020] Ritxar Arlegi and Dinko Dimitrov. Fair elimination-type competitions. *European Journal of Operational Research*, 287(2):528–535, 2020.

[Aronshtam *et al.*, 2017] Lior Aronshtam, Havazelet Cohen, and Tammar Shrot. Tennis manipulation: Can we help Serena Williams win another tournament? *Annals of Mathematics and Artificial Intelligence*, 80(2):153–169, 2017.

[Aziz *et al.*, 2015] Haris Aziz, Markus Brill, Felix Fischer, Paul Harrenstein, Jérôme Lang, and Hans Georg Seedig. Possible and necessary winners of partial tournaments. *Journal of Artificial Intelligence Research*, 54:493–534, 2015.

[Aziz *et al.*, 2018] Haris Aziz, Serge Gaspers, Simon Mackenzie, Nicholas Mattei, Paul Stursberg, and Toby Walsh. Fixing balanced knockout and double elimination tournaments. *Artificial Intelligence*, 262:1–14, 2018.

[Björklund *et al.*, 2007] Andreas Björklund, Thore Husfeldt, Petteri Kaski, and Mikko Koivisto. Fourier meets Möbius: Fast subset convolution. In *Proceedings of the 39th Annual ACM Symposium on Theory of Computing (STOC)*, pages 67–74, 2007.

[Brandt and Fischer, 2007] Felix Brandt and Felix A. Fischer. PageRank as a weak tournament solution. In *Proceedings of the 3rd International Workshop on Internet and Network Economics (WINE)*, pages 300–305, 2007.

[Brandt *et al.*, 2016a] Felix Brandt, Markus Brill, and Paul Harrenstein. Tournament solutions. In Felix Brandt, Vincent Conitzer, Ulle Endriss, Jérôme Lang, and Ariel D. Procaccia, editors, *Handbook of Computational Social Choice*, chapter 3, pages 57–84. Cambridge University Press, 2016.

[Brandt *et al.*, 2016b] Felix Brandt, Vincent Conitzer, Ulle Endriss, Jérôme Lang, and Ariel D. Procaccia, editors. *Handbook of Computational Social Choice*. Cambridge University Press, 2016.

[Brandt *et al.*, 2018] Felix Brandt, Markus Brill, Hans Georg Seedig, and Warut Suksompong. On the structure of stable tournament solutions. *Economic Theory*, 65(2):483–507, 2018.

[Brill *et al.*, 2020] Markus Brill, Ulrike Schmidt-Kraepelin, and Warut Suksompong. Refining tournament solutions via margin of victory. In *Proceedings of the 34th AAAI Conference on Artificial Intelligence (AAAI)*, pages 1862–1869, 2020.

[Brill *et al.*, 2021] Markus Brill, Ulrike Schmidt-Kraepelin, and Warut Suksompong. Margin of victory in tournaments: Structural and experimental results. In *Proceedings of the 35th AAAI Conference on Artificial Intelligence (AAAI)*, 2021.

[Chatterjee *et al.*, 2016] Krishnendu Chatterjee, Rasmus Ibsen-Jensen, and Josef Tkadlec. Robust draws in balanced knockout tournaments. In *Proceedings of the 25th International Joint Conference on Artificial Intelligence (IJCAI)*, pages 172–179, 2016.

[Dey, 2017] Palash Dey. Query complexity of tournament solutions. In *Proceedings of the 31st AAAI Conference on Artificial Intelligence (AAAI)*, pages 2992–2998, 2017.

[Ding and Weinberg, 2021] Kimberly Ding and S. Matthew Weinberg. Approximately strategyproof tournament rules in the probabilistic setting. In *Proceedings of the 12th Innovations in Theoretical Computer Science Conference (ITCS)*, pages 14:1–14:20, 2021.

[Eidelstein *et al.*, 2019] Noa Eidelstein, Lior Aronshtam, Eitan Eidelstein, and Tammar Shrot. Controlling subtournaments: Easy or hard problem? *Annals of Mathematics and Artificial Intelligence*, 86(4):257–268, 2019.

[Faliszewski and Rothe, 2016] Piotr Faliszewski and Jörg Rothe. Control and bribery in voting. In Felix Brandt, Vincent Conitzer, Ulle Endriss, Jérôme Lang, and Ariel D. Procaccia, editors, *Handbook of Computational Social Choice*, pages 146–168. Cambridge University Press, 2016.

[Faliszewski *et al.*, 2009] Piotr Faliszewski, Edith Hemaspaandra, Lane A. Hemaspaandra, and Jörg Rothe. Llull and Copeland voting computationally resist bribery and constructive control. *Journal of Artificial Intelligence Research*, 35:275–341, 2009.

[Fey, 2008] Mark Fey. Choosing from a large tournament. *Social Choice and Welfare*, 31(2):301–309, 2008.

[Gupta *et al.*, 2018a] Sushmita Gupta, Sanjukta Roy, Saket Saurabh, and Meirav Zehavi. When rigging a tournament, let greediness blind you. In *Proceedings of the 27th International Joint Conference on Artificial Intelligence (IJCAI)*, pages 275–281, 2018.

[Gupta *et al.*, 2018b] Sushmita Gupta, Sanjukta Roy, Saket Saurabh, and Meirav Zehavi. Winning a tournament by any means necessary. In *Proceedings of the 27th International Joint Conference on Artificial Intelligence (IJCAI)*, pages 282–288, 2018.

[Gupta *et al.*, 2019] Sushmita Gupta, Saket Saurabh, Ramanujan Sridharan, and Meirav Zehavi. On succinct encodings for the tournament fixing problem. In *Proceedings of the 28th International Joint Conference on Artificial Intelligence (IJCAI)*, pages 322–328, 2019.

[Hulett, 2019] Reyna Hulett. Single-elimination brackets fail to approximate Copeland winner. In *Proceedings of the 22nd International Conference on Approximation Algorithms for Combinatorial Optimization Problems (APPROX)*, pages 13:1–13:20, 2019.

[Kim and Vassilevska Williams, 2015] Michael P. Kim and Virginia Vassilevska Williams. Fixing tournaments for kings, chokers, and more. In *Proceedings of the 24th International Joint Conference on Artificial Intelligence (IJCAI)*, pages 561–567, 2015.

[Kim *et al.*, 2017] Michael P. Kim, Warut Suksompong, and Virginia Vassilevska Williams. Who can win a single-elimination tournament? *SIAM Journal on Discrete Mathematics*, 31(3):1751–1764, 2017.

[Konicki and Vassilevska Williams, 2019] Christina Konicki and Virginia Vassilevska Williams. Bribery in balanced knockout tournaments. In *Proceedings of the 18th International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, pages 2066–2068, 2019.

[Kovalchik, 2020] Stephanie Kovalchik. Extension of the Elo rating system to margin of victory. *International Journal of Forecasting*, 36(4):1329–1341, 2020.

[Laslier, 1997] Jean-François Laslier. *Tournament Solutions and Majority Voting*. Springer-Verlag, 1997.

[Manurangsi and Suksompong, 2021] Pasin Manurangsi and Warut Suksompong. Generalized kings and single-elimination winners in random tournaments. In *Proceedings of the 30th International Joint Conference on Artificial Intelligence (IJCAI)*, 2021.

[Mattei and Walsh, 2016] Nicholas Mattei and Toby Walsh. Empirical evaluation of real world tournaments. *CoRR*, abs/1608.01039, 2016.

[Mattei *et al.*, 2015] Nicholas Mattei, Judy Goldsmith, Andrew Klapper, and Martin Mundhenk. On the complexity of bribery and manipulation in tournaments with uncertain information. *Journal of Applied Logic*, 13(4):557–581, 2015.

[Ramanujan and Szeider, 2017] M. S. Ramanujan and Stefan Szeider. Rigging nearly acyclic tournaments is fixed-parameter tractable. In *Proceedings of the 31st AAAI Conference on Artificial Intelligence (AAAI)*, pages 3929–3935, 2017.

[Russell and van Beek, 2011] Tyrel Russell and Peter van Beek. An empirical study of seeding manipulations and their prevention. In *Proceedings of the 22nd International Joint Conference on Artificial Intelligence (IJCAI)*, pages 350–356, 2011.

[Russell and Walsh, 2009] Tyrel Russell and Toby Walsh. Manipulating tournaments in cup and round robin competitions. In *Proceedings of the 1st International Conference on Algorithmic Decision Theory (ADT)*, pages 26–37, 2009.

[Ryvkin and Ortmann, 2008] Dmitry Ryvkin and Andreas Ortmann. The predictive power of three prominent tournament formats. *Management Science*, 54(3):492–504, 2008.

[Saile and Suksompong, 2020] Christian Saile and Warut Suksompong. Robust bounds on choosing from large tournaments. *Social Choice and Welfare*, 54(1):87–110, 2020.

[Schneider *et al.*, 2017] Jon Schneider, Ariel Schvartzman, and S. Matthew Weinberg. Condorcet-consistent and approximately strategyproof tournament rules. In *Proceedings of the 8th Innovations in Theoretical Computer Science Conference (ITCS)*, pages 35:1–35:20, 2017.

[Schvartzman *et al.*, 2020] Ariel Schvartzman, S. Matthew Weinberg, Eitan Zlatin, and Albert Zuo. Approximately strategyproof tournament rules: On large manipulating sets and cover-consistence. In *Proceedings of the 11th Innovations in Theoretical Computer Science Conference (ITCS)*, pages 3:1–3:25, 2020.

[Scott and Fey, 2012] Alex Scott and Mark Fey. The minimal covering set in large tournaments. *Social Choice and Welfare*, 38(1):1–9, 2012.

[Stanton and Vassilevska Williams, 2011] Isabelle Stanton and Virginia Vassilevska Williams. Rigging tournament brackets for weaker players. In *Proceedings of the 22nd International Joint Conference on Artificial Intelligence (IJCAI)*, pages 357–364, 2011.

[Stanton and Vassilevska Williams, 2013] Isabelle Stanton and Virginia Vassilevska Williams. The structure, efficacy, and manipulation of double-elimination tournaments. *Journal of Quantitative Analysis in Sports*, 9(4):319–335, 2013.

[Stoica *et al.*, 2020] Ana-Andreea Stoica, Abhijnan Chakraborty, Palash Dey, and Krishna Gummadi. Minimizing margin of victory for fair political and educational districting. In *Proceedings of the 19th International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, pages 1305–1313, 2020.

[Suksompong, 2016] Warut Suksompong. Scheduling asynchronous round-robin tournaments. *Operations Research Letters*, 44(1):96–100, 2016.

[Vassilevska Williams, 2010] Virginia Vassilevska Williams. Fixing a tournament. In *Proceedings of the 24th AAAI Conference on Artificial Intelligence (AAAI)*, pages 895–900, 2010.

[Vassilevska Williams, 2016] Virginia Vassilevska Williams. Knockout tournaments. In Felix Brandt, Vincent Conitzer, Ulle Endriss, Jérôme Lang, and Ariel D. Procaccia, editors, *Handbook of Computational Social Choice*, chapter 19, pages 453–474. Cambridge University Press, 2016.

[Vu *et al.*, 2009] Thuc Vu, Alon Altman, and Yoav Shoham. On the complexity of schedule control problems for knockout tournaments. In *Proceedings of the 8th International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, pages 225–232, 2009.

[Xia, 2012] Lirong Xia. Computing the margin of victory for various voting rules. In *Proceedings of the 13th ACM Conference on Economics and Computation (EC)*, pages 982–999, 2012.

[Yang and Dimitrov, 2021] Yongjie Yang and Dinko Dimitrov. Weak transitivity and agenda control for extended stepladder tournaments. *Economic Theory Bulletin*, 9(1):27–37, 2021.

[Yang and Guo, 2017] Yongjie Yang and Jiong Guo. Possible winner problems on partial tournaments: a parameterized study. *Journal of Combinatorial Optimization*, 33(3):882–896, 2017.