Maps of Tournaments: Distances, Experiments, and Data

Filip Nikolow[a], **Piotr Faliszewski**[a,*] **and Stanisław Szufa**[a,b]

[a]AGH University, Kraków, Poland
[b]CNRS, LAMSADE, Université Paris Dauphine – PSL, Paris, France
ORCID (Piotr Faliszewski): https://orcid.org/0000-0002-0332-4364, ORCID (Stanisław Szufa): https://orcid.org/0000-0001-6301-6227

Abstract. We form a "map of tournaments" by adapting the map framework from the world of elections. By a tournament we mean a complete directed graph where the nodes are the players and an edge points from a winner of a game to the loser (with no ties allowed). A map is a set of tournaments represented as points on a 2D plane, so that their Euclidean distances resemble the distances computed according to a given measure. We identify useful distance measures, discuss ways of generating random tournaments (and compare them to several real-life ones), and show how the maps are helpful in visualizing experimental results (also for knockout tournaments).

1 Introduction

In the map framework, introduced by Szufa et al. [40] and Boehmer et al. [6], we are given a set of (ordinal) elections together with so-called original distances between them, and we visualize them as points on a plane so that the distances between the points resemble the original ones. Such maps are helpful in presenting experimental results and in understanding relations between the elections (as well as the random models or the real-life data sources from which they come). For example, they were instrumental in better understanding the parameterization of the classic Mallows model [9]. Since the inception of the map idea, it was adapted to approval elections [41], stable roommates instances [10], and voting rules [23]. In this paper we adapt the framework to the case of *tournament graphs*.

Two most prominent types of tournaments studied in AI literature are the knockout and the round-robin ones. In the former, in each round the participants are arranged in pairs and play against each other, with the winners moving up to the next round; in the latter, every pair of participants plays against each other exactly once. In knockout tournaments, the winner is the last player standing, whereas for round-robin ones there is a whole set of different rules for victory (e.g., the Copeland rule selects the player(s) who win most matches). In either case, for research purposes it is convenient to represent the results of all possible games—even if not all are actually played—as a *tournament graph*, i.e., a complete directed graph, where for each two players there is an edge pointing from the winner to the loser (or, from who we expect to win to who we expect to lose, if perfect information is unavailable;[1] ties are forbidden). Such graphs allow us to compute tournament winners [13, 25] or possible winners [3, 47, 5], analyze players' margins of victory (MoV) [17, 20], check if it is possible to manipulate the results [42, 4], analyze various axiomatic properties of applied solution concepts [14, 16] and so on (see the surveys of Brandt et al. [15] and Suksompong [39] for references going beyond the just-mentioned examples). Indeed, both knockout and round-robin tournaments received significant attention within AI and computational social choice literature, but mostly in terms of theoretical studies, with experiments gaining traction only recently (in particular, we mention the works of Saile and Suksompong [37], Baumeister and Hogrebe [5], Brill et al. [17], and Führlich et al. [24]; works of Russell [35] and Brandt and Seedig [12] are rare examples of older experimental studies[2]). Our goal is to help researchers conduct experiments on tournaments in a rigorous, systematic way (we will often abbreviate *tournament graphs* as *tournaments*).

Components of a Map. To form a tournament map, we need three main components: A way to compute the distances between the tournaments, special tournaments to mark characteristic spots on the map, and a way to obtain the data.

Regarding the distance measure, an intuitive idea is to use the *graph edit distance (GED)*: Given two tournaments over the same number of players, its value is the number of edges that we need to reverse in one for it to become isomorphic to the other [38] (the two tournaments may include unrelated sets of players, such as NBA teams in one and bridge players in the other, so the distance needs to be invariant to their names; hence the use of isomorphism).

As computing its values is NP-hard [2], we also devised a polynomial-time computable distance, based on the Katz centrality measure: Given two tournaments, for each of them we compute Katz centralities of its vertices, put them in a vector (sorted in the nonincreasing order) and take the ℓ_1 distance of these vectors. While this distance and the maps it yields are not as appealing as the GED ones, they are sufficient and can be used for larger tournaments. (Similar type of a distance was previously used for approval elections [41]).

With a distance measure in hand, it is also important to identify characteristic tournaments. Two obvious choices are (a) the *ordered* tournament (T_{ord}), where the players are ranked from the strongest to the weakest, and each player defeats all the weaker ones, and the (b) *rock-paper-scissors (RPS)* tournament, where each player defeats half of the other ones (in fact, there are several nonisomorphic tournaments of this kind; one could also refer to these tournaments as *regular*). By mixing them, we obtain an (approximate) circumfer-

* Corresponding Author. Email: faliszew@agh.edu.pl

[1] It is also natural to consider weighted tournament graphs, where the edges are labeled with the probability of victory of the more likely winner, but we focus on the simpler, deterministic setting.

[2] There are experimental works beyond AI, e.g., in economics, but we mostly mean those connected to algorithmic investigations.

ence of the tournament space (at least in our maps).

Finally, to obtain tournaments we either generate synthetic ones (e.g., using the classic Condorcet noise model or generating nonisomorphic tournaments), or use real-life data. To this end, we collected data from NBA seasons and from a number of bridge tournaments. We find that the NBA tournaments are much closer to T_{ord} than the bridge ones, but both types of tournaments are quite specific in that they are far from covering the whole tournament space. In a number of our experiments, the NBA tournaments, as well as those nearby, gave the most varied results, which suggests that they are in a particularly diverse part of the tournament space.

We have also analyzed a dataset analogous to that of Brill et al. [17] and a dataset derived from the elections of Faliszewski et al. [22]. The former contains realistic data, whereas the latter one is quite specific and sometimes shows unexpected behavior (for example, elections generated uniformly at random, using the impartial culture model, tend to generate tournaments with fairly strong structure). Yet, many tournaments derived from elections land in a similar area of our maps as the NBA ones, so they can be useful, but should be analyzed with care.

Evaluation. We evaluate our maps in several ways. First, we compute the correlation between the GED and Katz distances. We find that, overall, the Katz distance is acceptable (if not ideal). Next, we show where tournaments generated according to a number of statistical cultures land on our maps and how it compares to real-life data. Finally, we show how our maps help in visualizing a number of experiments (in particular, regarding the number of winners that various solutions concepts for round-robin tournaments generate, and regarding possible manipulations of knockout tournaments). Based on such visualizations one can, for example, identify areas of the tournament space where various phenomena happen (as we do by pointing to the vicinity of the NBA tournaments) or predict how structure of the tournaments affects a particular quantity (e.g., in our experiments the distance from T_{ord} is a good estimator of the time needed to compute the tournament's Slater winners using an ILP solver).

Supplementary Material. For suplementary materials, visit: https://github.com/Project-PRAGMA/Map-of-Tournaments-ECAI-2025

2 Preliminaries

We use $[n]$ to denote the set $\{1, 2, ..., n\}$. By $\mathbb{R}_+$ we mean the set of nonnegative real numbers.

Graphs and Tournaments. A *directed graph* is a pair $G = (V, E)$, where V is the set of vertices and E is the set of edges (sometimes called arcs in the literature). Each edge is a pair of vertices, where we interpret (v, u) as pointing from v to u. By $\delta(v)$ we mean the outdegree of vertex v, i.e., the number of edges pointing out from it. All our graphs are directed.

A *tournament graph* (or a *tournament*, for short) is a graph where for each two vertices u and v there is exactly one edge between them (either from u to v or the other way round). We often refer to the vertices in a tournament as the *players* or *participants* and we interpret an edge (v, u) as meaning that player v wins against player u. We write $\mathcal{T}_n$ to denote the set of all tournaments over n players.

We say that graphs $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$ are isomorphic if there is a bijection $\pi: V_1 \to V_2$ such that for each two vertices $v, u \in V_1$, G_1 contains edge (v, u) if and only if G_2 contains edge $(\pi(v), \pi(u))$.

Distances. Given a set X, a function $d: X \times X \to \mathbb{R}_+$ is a *distance* if for all $x, y, z \in X$ it holds that: (a) $d(x, y) = 0$ if and only if $x = y$, (b) $d(x, y) = d(y, x)$, and (c) $d(x, y) \leq d(x, z) + d(z, y)$. If we relax the first condition to require that $d(x, x) = 0$ for all $x \in X$, then we get a pseudodistance. Abusing the terminology, we often refer to pseudodistances as distances.

Distances Between Vectors. Given two real-valued vectors, $x = (x_1, \ldots, x_n)$ and $y = (y_1, \ldots, y_n)$, their ℓ_1 distance is $\ell_1(x, y) = |x_1 - y_1| + \cdots + |x_n - y_n|$.

Distances Between Tournaments. Let d be a distance between tournaments (over the same numbers of players). We say that d is *isomorphism respecting* (*isomorphic*, for short) if it is invariant to reordering the players. So, if T_1 and T_2 are isomorphic and d is such a distance, then $d(T_1, T_2) = 0$.

Centrality Measures. A centrality measure is a function that given a vertex, associates it with its "importance" (vertex outdegree is among the simplest such measures). Let us fix numbers $\alpha \in [0, 1)$ and $\beta \geq 0$, and a graph $G = (V, E)$, with $V = \{v_1, \ldots, v_n\}$. Katz centralities of $v_1, \ldots, v_n$, denoted $\mathrm{katz}(v_1), \ldots, \mathrm{katz}(v_n)$, are numbers such that [26]:

$$\mathrm{katz}(v_i) = \textstyle\sum_{(v_i, v_j) \in E} \alpha \cdot \mathrm{katz}(v_j) + \beta.$$

We use $\alpha = 0.1$ and $\beta = 1$.[3] Katz centrality is related to the eigenvector [11] and PageRank centralities [34] as it implements the same general intuition: A vertex in a tournament is "important" (or, "strong" in the language of tournaments) if it connects to other "important" vertices (if it beats other "strong" players). For further discussions of eigenvector and PageRank centralities, see, e.g., their axiomatic characterizations due to Wąs and Skibski [43, 44]. Katz also has some connection to the outdegree centrality, as its value most strongly depends on the values of its neighbors.

3 Generating and Collecting Tournaments

Next, we describe the tournaments that we present in our maps. In most of our experiments, we either focus on tournaments with 7 or 12 players, but we also consider other sizes.

3.1 Characteristic Tournaments

Let n be the number of players. In the ordered tournament, T_{ord}, there is a strict ordering of the players from the strongest to the weakest and each player wins with all the weaker ones (i.e., T_{ord} is transitive; the number of players will always be clear from the context so we omit it in notation). Intuitively, T_{ord} is the "simplest" tournament, with a clear winner.

On the other end of the spectrum we have the RPS family of the "rock-paper-scissors" tournaments, where each player wins against half of the other ones (modulo parity of the number of players) and we typically expect a multiway tie. The name stems from a popular game, represented as a three-player tournament, where each player wins against one opponent and loses to the other one. The RPS family includes many nonisomorphic tournaments, of which we are particularly interested in T_{rps}, formed as follows: We let the players set be $V = \{v_0, \ldots, v_{n-1}\}$ and as i goes from 0 to $n - 1$, we add an edge from v_i to each of $v_{(i+1) \bmod n}, \ldots, v_{(i+\lfloor \frac{n}{2} \rfloor) \bmod n}$ (unless the result of this game was settled already, which happens for even numbers of players, once for each of the players $v_{\frac{n}{2}}, \ldots, v_{n-1}$).

[3] These are standard settings from the NetworkX Python library.

We also consider two families of tournaments that fall between the above two extremes. In the $\mathrm{ord}/\mathrm{RPS}$ one, for each $k \in [n-3]$ there is a tournament where k players form the T_{ord} subtournament and $n-k$ form the T_{rps} one, with all players from T_{ord} defeating all those from T_{rps} (note that we consider k only up to $n-3$, so the T_{rps} tournament has at least 3 players and, hence, is nondegenerate). Family $\mathrm{RPS}/\mathrm{ord}$ is defined analogously, but with the edges between the T_{ord} and T_{rps} subtournaments reversed.

3.2 Statistical Models of Tournaments

There are two main ways of generating random tournaments. We can either use "native" models that generate tournaments directly, or we can first generate ordinal elections and then derive their majority relations (the latter approach is taken, among other options, e.g., by Brill et al. [17] and Brandt and Seedig [12]). We focus on the following direct models, but we will mention election-related ones later (let $V = \{v_1, \dots, v_n\}$ be the set of players).

Nonisomorphic Sampler. We use the package `nauty` of McKay and Piperno [32] to generate nonisomorphic tournaments. For the case of 7 players, we generate all such tournaments and for other cases only a subset of them; `nauty` does not guarantee uniform distribution in any clear sense, so we generate a large number of such tournaments and sample from them (for improved diversity).

Uniform Model. In the uniform model, for each pair of players we choose the result of their game (i.e., the orientation of the edge between them) independently, uniformly at random.

Condorcet Noise Model. We are given probability p as a parameter. We start with the ordered tournament and for each pair of players we reverse the result of their game with probability $1-p$.

Strength Models. Each player v_i has some strength $w(i) \in \mathbb{R}_+$. To generate a tournament, for each two players v_i and v_j, the former wins with probability $\frac{w(i)}{w(i)+w(j)}$. E.g., we consider exponential strength functions, $w^{\alpha}_{\exp}(i) = \alpha^i$, where $\alpha \geq 1$ is a parameter.

Remark 3.1. *Models similar to the strength one appear in the literature in various forms and shapes. In particular, Ryvkin and Ortmann [36] suggest how players' abilities should translate to victory probabilities (see also the work of Aldous [1]). Saile and Suksompong [37] consider a "gap model" based on a similar premise.*

3.3 Collected Real-Life Data

We have collected data from the NBA League (top US basketball league) and from Polish Sports Bridge Association (which manages the bridge card game leagues in Poland).

NBA. We have formed a tournament for each of the 20 regular NBA seasons (i.e., not counting playoffs) between the 2000/2001 and 2019/2020 ones. The first four of these seasons include 29 teams and the later ones include 30 teams. During each season, each NBA team plays every other one between two and four times (depending on the conference and the division they are in). Given two teams, we convert the results of their games to an edge in the tournament by choosing as the winner the team that, altogether, scored more points against the other one. Details regarding ties are in the full version of the paper.

Bridge. Polish Sports Bridge Association stores an archive of the results from the Polish leagues (https://www.pzbs.pl/liga-archiwum); we collected the results of the round-robin parts of the top, north, and south leagues for seasons from 2017/2018 to 2022/2023. Results of the matches were directly converted to tournament graphs using the Victory Points scored in head-to-head contest between each two teams (the team with higher score wins). Altogether, this dataset consists of 17 tournaments and each tournament has 16 players.

3.4 Dataset Composition

Finally, we describe four datasets that we consider. Each of them includes the T_{ord} and T_{rps} tournaments, as well as $n-3$ tournaments from the $\mathrm{ord}/\mathrm{RPS}$ family and $n-3$ tournaments from the $\mathrm{RPS}/\mathrm{ord}$ one (where n is the number of players in the tournaments from a given dataset) in addition to the tournaments described below.

Size-7 Dataset. This dataset consists of all 456 nonisomorphic tournaments over 7 players (this was the largest number of players for which it was reasonable to generate all the tournaments; for slightly more players the number of such tournaments is too large to conveniently present them on a map, and for significantly more players even computing all nonisomorphic graphs is challenging).

Size-12 Dataset. This dataset regards 12 players and is our main object of study. We formed this dataset as follows:

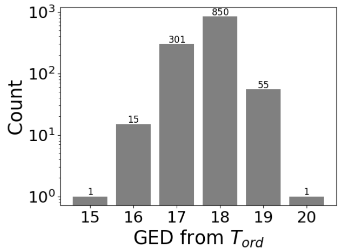
1. We generated 100 nonisomorphic tournaments.[4]
2. We generated 20 tournaments using the uniform model.
3. For each $p \in \{0.1, 0.2, 0.3, 0.4\}$, we generated 5 tournaments using the Condorcet noise model.
4. For each function among $w^{e}_{\exp}(i) = e^i$, $w^{2}_{\exp}(i) = 2^i$, $w_{\mathrm{lin}}(i) = i$, $w_{\log}(i) = \log(i)$, $w_{\mathrm{root}}(i) = \sqrt{i}$, we generated 10 tournaments using the strength model with this strength function. We also generated 10 tournaments by choosing $\alpha \in [1, 2]$ uniformly at random (separately for each tournament) and using the $w^{\alpha}_{\exp}(i) = \alpha^i$.
5. For each of the NBA and bridge tournaments we, formed four copies of this tournament and restricted each of them to randomly (and independently) selected 12 players. Hence, altogether, we have 80 NBA tournaments and 68 bridge tournaments.

MoV Dataset. This dataset consists of tournaments generated in the same way as in the work of Brill et al. [17], except that we considered 12 players (to maintain similarity to the above dataset) and generated 10 tournaments for each statistical model instead of 100. We also included 100 nonisomorphic tournaments from `nauty`, as in the Size-12 dataset.

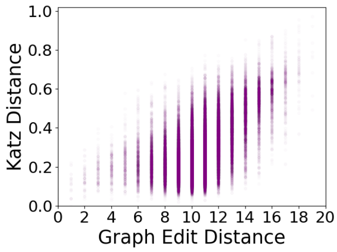
Election Dataset. We generated tournaments based on the election data of Faliszewski et al. [22]. In their dataset, each election has 8 candidates and 96 voters, where each voter ranks the candidates from the most to the least desirable one. Given such an election, we form a tournament where candidates are players and a candidate a wins against candidate b if he or she is preferred by a majority of the voters (i.e., we use their majority relations; we resolve ties uniformly at random).

Remark 3.2. *In particular, the election dataset includes elections generated using the impartial culture model, where each vote (i.e., each ranking of the candidates) is sampled uniformly at random.*

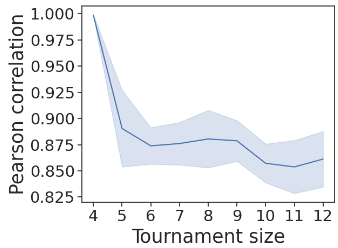
[4] We used `nauty` with the res/mod option set to $0/200000$, generated 800 000 nonisomorphic graphs, and sampled 100.



(a) Histogram of GED distances between T_{ord} and RPS ones.



(b) Comparison of the GED and Katz distances on the Size-12 dataset.



(c) PCC between GED and Katz distances, depending on the number of players.

Figure 1: Basic insights regarding tournaments. Plot (a) shows a histogram of GED distances between T_{ord} and all RPS tournaments with 11 players, plot (b) compares the GED and Katz distances in the Size-12 dataset (each point is a pair of tournaments, its x coordinate is their GED distance and its y coordinate is the Katz distance), and plot (c) shows PCC of the GED and Katz distances between T_{ord} and 100 tournaments generated using the uniform model, depending on the number of players (for each number of players we repeated the experiment 10 times; the shaded area shows standard deviation).

Consequently, given a pair of candidates a and b, the probability that a would defeat b if we first generated an election using the impartial culture model and then generated a tournament from this election is 0.5*. So, thus generated tournaments seem to be analogous to those from the uniform model, but the results of the games are not independent (for example, if we consider three candidates, a, b, and c, and three voters, then the probability that in an election drawn using the impartial culture model a majority of the voters ranks a over b—under the condition that a majority of the voters also ranks a over c—is* $^{68}/_{108} \approx 0.629$ *and not* 0.5*).*

We omit from the election dataset tournaments that are equal to T_{ord} (for readers familiar with elections literature, this includes, e.g., tournaments generated from single-peaked elections). For better comparison with the other datasets, we included 500 nonisomorphic tournaments from `nauty`.

4 Distances Between Tournaments

Below we describe two distances that we use to generate our maps. The first one is precise, but hard to compute for larger graphs, whereas the latter is fast to compute, but less precise.

4.1 Graph Edit Distance

Given tournaments $T_1, T_2 \in \mathcal{T}_n$, their *graph edit distance (GED)*, denoted $d_{\mathrm{GED}}(T_1, T_2)$, is the number of edges that we need to reverse in T_1 to obtain a tournament isomorphic to T_2 [38]. It is well-known that d_{GED} is a pseudodistance and, by definition, it is isomorphic. On the negative side, deciding if the GED distance between two tournaments is at most a given value is NP-hard [2], but it can be computed using an ILP formulation of Lerouge et al. [30]. GED is natural, intuitive, and appealing. Below we provide some of its features relevant to our analysis.

First, we observe that GED is closely related to the *pairwise* distance of Szufa et al. [40]. For our discussion, the only difference is that the latter is defined on weighted tournaments, where each edge has a weight between 0 and 1 (in the language of Szufa et al. [40], pairwise is defined on *weighted majority relations* of ordinal elections); GED is equal to pairwise if we assume all edges to have weight 1. Consequently, the largest GED distance between two tournaments is bounded from above by the largest pairwise distance between two weighted tournaments, which for tournaments with n players is $\frac{1}{4}(n^2 - n)$ [8]. By the next proposition, we see that the largest GED distance is comparable to this value.

Proposition 4.1. *If n is an odd integer representing the number of players, then $d_{\mathrm{GED}}(T_{\mathrm{ord}}, T_{\mathrm{rps}}) = {}^1/_8(n^2 - 1)$.*

However, the above is certainly not the largest possible GED value as even in the RPS family there are graphs that are further away from T_{ord} than T_{rps}. Indeed, in Figure 1a we see a histogram of distances between the RPS tournaments and T_{ord}, for 11 players (this was the largest size for which we were able to generate all RPS tournaments). In this case, $d_{\mathrm{GED}}(T_{\mathrm{ord}}, T_{\mathrm{rps}}) = 15$ and all the other RPS tournaments are further away from T_{ord}.

It turns out that for sufficiently large n, the farthest distance of a tournament from T_{ord} is of the form $\frac{1}{4}(n^2 + n) \pm O(n^{\frac{3}{2}})$; see sequence A003141 in the encyclopedia of integer sequences [33], as well as the book of Erdös and Spencer [21, p. 42]. Thus, for large enough n, the largest pairwise distance is not too far off from the largest GED one. Nonetheless, if needed, we prefer to normalize GED distances by the distance between T_{ord} and T_{rps} because the largest possible distance is at most twice as large and value $d_{\mathrm{GED}}(T_{\mathrm{ord}}, T_{\mathrm{rps}})$ has clear interpretation. In our data we did not observe too many tournaments that would be much further away.

4.2 Katz Distance

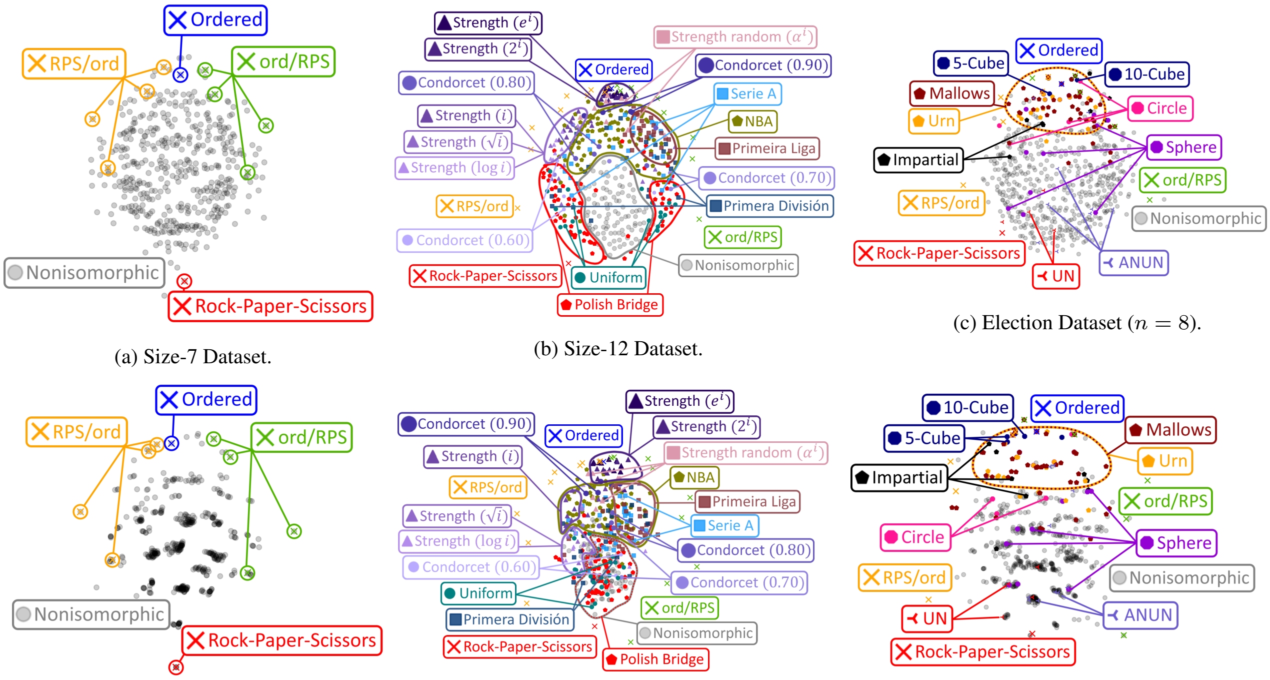
Our second distance is based on the idea that two tournaments are similar if Katz centralities of their players are similar. For a tournament T, we write $\mathrm{katz}^{\downarrow}(T)$ to denote the vector of Katz centralities of the players in T, sorted in the nonincreasing order.

Definition 4.1. *Let T_1 and T_2 be two tournaments over the same number of players. We define their Katz distance as: $d_{\mathrm{katz}}(T_1, T_2) = \ell_1(\mathrm{katz}^{\downarrow}(T_1), \mathrm{katz}^{\downarrow}(T_2))$.*

Remark 4.1. *Sorting the vectors of Katz centralities ensures that the Katz distance is isomorphic.*

Since Katz centrality can be computed in polynomial time, and we only compute it $2n$ times to obtain the Katz distance, computing d_{katz} can also be done in polynomial time. Katz distance is a pseudodistance as two different tournaments can have the same sorted Katz vectors without being isomorphic.

We can define analogous distances using various other centrality measures and, indeed, we tried nine different ones (see Table 1 for their list and see the full version of the paper for their definitions; most of them are also covered in the overview of Koschützki et al. [27]). It turned out that Katz is superior, both because the maps it



(a) Size-7 Dataset.

(b) Size-12 Dataset.

(c) Election Dataset ($n = 8$).

(d) Size-7 Dataset.

(e) Size-12 Dataset.

(f) Election Dataset ($n = 8$).

Figure 2: Maps of different datasets computed using GED (upper row) and Katz distance (lower row).

generates most resemble those produced using GED and because it is most strongly correlated with GED, as measured using Pearson and Spearman correlation coefficients (PCC and SCC, computed for the Size-12 dataset; see Table 1). We illustrate the correlation between the Katz distance and GED in Figure 1b (analogous figures for the other centrality-based distances are in the full version of the paper).

Additionally, in Figure 1c we show how the PCC between the Katz distance and GED changes as we vary the number of candidates, while looking at the distance between T_{ord} and tournaments generated from the uniform model. While we can only consider up to 12 players (due to intractability of GED), we see that this value seems to stabilize. This is a good sign as distance from T_{ord} is one of the more important features of both the synthetic and real-life tournaments that we consider (this will become visible in the maps presented in the following two sections).

5 Map of Tournaments

For each of our four datasets and both distances (GED and Katz), we have prepared its map as follows: For each pair of tournaments we computed the distance between them (we call these distances *original*) and, then, for each tournament we found a point on a 2D plane so that the Euclidean distances between the points would resemble the original ones (we used the MDS algorithm [28]). The resulting maps are in Figure 2 (see the full paper for a map with larger tournaments). We comment on the maps' features below.

Table 1: PCC and SCC between centrality-based distances and GED on the Size-12 dataset.

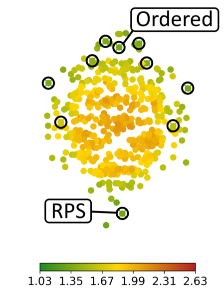
Centrality measure	PCC	SCC
Katz Centrality	0.573	0.513
Degree Centrality	0.558	0.493
Closeness Centrality	0.440	0.388
Eigenvector Centrality	0.429	0.403
Laplacian Centrality	0.422	0.378
Harmonic Centrality	0.397	0.365
PageRank	0.375	0.367
Betweenness Centrality	0.243	0.283
Load Centrality	0.243	0.282

GED Versus Katz. While the maps obtained using GED are different from those obtained using the Katz distance, both seem to provide the same intuitions. For example, in Figures 2b and 2e we see that the bridge tournaments form two separate clusters in the GED map, but only a single cluster in the Katz map. However, in both types of maps these tournaments have similar properties: They are far from T_{ord}, located in the similar part of the tournament space (given that Katz fails to distinguish the tournaments provided by `nauty`).

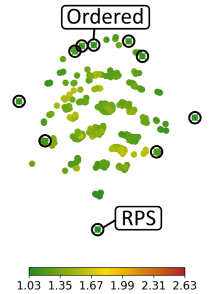
Quality of the Embedding. For a pair of tournaments T_1 and T_2 on a given map, we write $d(T_1, T_2)$ to denote their original distance (either GED or Katz) and $d_{\text{Euc}}(T_1, T_2)$ to denote the Euclidean distance between their points (in both cases, normalized by the respective distance between T_{ord} and T_{rps}). Then, we define the distortion of T_1 and T_2 as $\max(d(T_1,T_2), d_{\text{Euc}}(T_1,T_2)) / \min(d(T_1,T_2), d_{\text{Euc}}(T_1,T_2))$ (the notion was first used to evaluate maps by Boehmer et al. [7]).

To evaluate the quality of an embedding, for each tournament we compute its average distortion (i.e., the average over distortions between this tournament and all the other ones). In Figure 3 we present the maps where each point is colored according to its distortion. We see that the distortions are similar for both the Size-7 and Size-12 datasets and, generally, the Katz maps are presented more accurately. A possible explanation is that GED assumes only a very limited set of values (e.g., for the Size-12 dataset, all GED distances are in the set $\{0, 1, \ldots, 20\}$), which makes it difficult to present them in 2D space (Boehmer et al. [7] achieve much better distortion values for maps of ordinal elections, but the maps of stable roommates produced by Boehmer et al. [10] are similar in this respect to ours[5]).

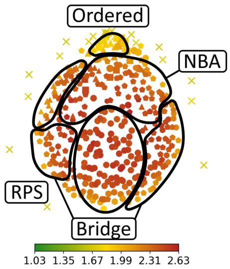
[5] Formally, Boehmer et al. [7] look at slightly different notion than the dis-



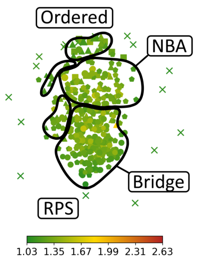
(a) GED, Size-7 dataset.



(b) Katz, Size-7 dataset.



(c) GED, Size-12 dataset.



(d) Katz, Size-12 dataset.

Figure 3: Distortion of the the Size-7 and Size-12 maps.

Location of Synthetic Tournaments. We see that the smaller the p parameter in the Condorcet noise model or the faster-growing the strength function in the strength models, the closer are the generated tournaments to T_{ord} in the Size-12 maps (Figures 2b and 2e), which we find quite intuitive.

We note that tournaments generated using the strength models are closer to the RPS/ord tournaments than to the ord/RPS ones. We explain this effect on the example of the linear strength function, $w(i) = i$: We have 12 players, $v_1, \ldots, v_{12}$, with strengths $1, \ldots, 12$, respectively. The probability that in the generated tournament v_{12} beats v_{11} is $^{12}/_{23} \approx 0.52$. Similarly, the probabilities that v_{12} beats v_{10}, and that v_{11} beats v_{10}, are, respectively, $^{12}/_{22} \approx 0.545$ and $^{11}/_{21} \approx 0.52$. All these values are quite close to 0.5. However, the probability that v_2 beats v_1 is already $^2/_3 = 0.66$, which is notably higher. In other words, the subtournament between the strongest players is more likely to resemble T_{rps}, whereas the subtournament between the weakest ones is more likely to resemble T_{ord}. Since the strongest players are also more likely to defeat the weakest ones, altogether we are more likely to get a structure that resembles some tournament from the RPS/ord family than from the ord/RPS one.

Coverage of the Real-Life Tournaments. In Figures 2b and 2e, we see that in the areas where NBA and bridge tournaments appear, we also see synthetically generated tournaments. Further, aside from the close vicinity of T_{ord}, synthetic tournaments do not appear in other areas. This suggests that, at least as far as NBA and bridge goes, our synthetic models are realistic. The MoV dataset (depicted in the full version of the paper) is similar in this respect and, so, the data of Brill et al. [17] covers reasonable parts of the space of tournaments. On the other hand, most of the tournaments from the Elections dataset are in the vicinity of T_{ord} (see Figure 2c and Figure 2f). Thus, one should be careful with using elections to generate tournament data if the purpose is to study realistic tournaments (and not elections themselves) as one might miss relevant areas of the tournament space. On the other hand, our election-based tournaments are in the same part of the map as the NBA tournaments and, later on, we will see that this area is particularly interesting (in the sense that tournaments from this area lead to diverse results in experiments).

Remark 5.1. *Let us compare the locations of tournaments generated using the (election-based) impartial culture model and the (tournament-native) uniform model (recall Remark 3.2). The former ones appear in the top halves of the maps, often quite close to T_{ord}, whereas the latter ones are placed in the bottom halves, much closer to T_{rps}. This is surprising as, intuitively, one would expect these two models to be very similar. It turns out that the correlations between players, introduced in the impartial culture model, are strong.*

NBA Versus Bridge. It is striking how the NBA tournaments are much closer to T_{ord} than the bridge ones. Intuitively, one would expect that in sports there would be stronger and weaker teams, and some sort of approximate hierarchy would be visible in the tournaments. This indeed is so in the NBA data, but not in bridge. One reason is that unweighted tournaments, as studied in this paper, are somewhat insufficient to fully capture the complexity of the bridge league. There, teams not only win or lose, but also collect points. The final outcome depends on the total number of collected points. The best teams can play very risky if there is a chance to collect a large bonus but, in consequence, occasionally they lose games that they could have won had they played more conservatively (but, in the end, it is worth it for them). Since such tournaments happen in real-life, we believe that one should take them into account in experiments.

Computation Time. Computing the Size-12 map for GED took several days on a server with 2 Intel(R) Xeon(R) Gold 6338 CPU @ 2.00GHz (32 cores with 2 threads in each CPU), whereas for Katz it took 89 seconds. Times reported in the following section were obtained on the same machine, but using only a single thread.

6 Examples of Experiments Using Maps

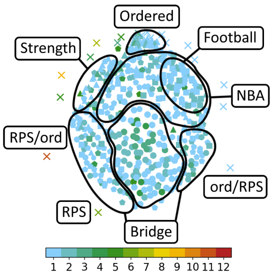
In this section, we use our maps to visualize the results of several experiments regarding winner determination in round-robin and knockout tournaments. All the experiments use the Size-12 dataset.

Round-Robin Tournaments. We consider the following three rules for selecting winners in round-robin tournaments (popular in the literature and based on different principles):

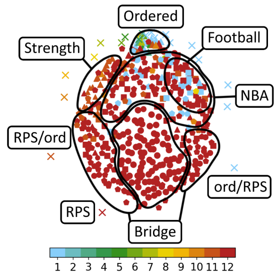
1. Under the Copeland rule, a score of a given player is equal to the number of players that he or she wins against (i.e., his or her outdegree). The player with the highest score wins. If there is more than one player with the highest score, they all tie as winners.
2. Top cycle of a tournament is the smallest nonempty set of players such that each of its members wins against each nonmember. The top cycle rule declares as winners all the members of the top cycle.
3. Under the Slater rule, we find a ranking of players that minimizes the number of pairs of players a, b such that a is ahead of b in the ranking, but loses to b in the tournament, and declare as the winner the top player in this ranking. As there may be several rankings that fulfill this condition, the rule can have several tied winners.

In Figures 4a, 4b, and 4c we present maps where each point (tournament) is colored according to the number of Copeland, top cycle, and Slater winners, respectively. We see that out of the 368 tournaments in the dataset, 224 (60%) have a unique winner under Copeland and 236 (64%) have a unique winner under Slater (for each of these rules there are cases where it has a unique winner but the other one does not) and no part of the map seems to stand out.

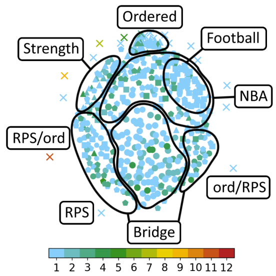
tortion that we consider, but distortion values can be deduced from it.



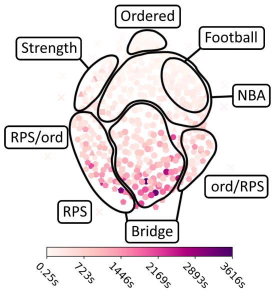
(a) Copeland winners count.



(b) Top cycle winners count.



(c) Slater winners count.



(d) Slater running time.

Figure 4: The numbers of winners under the Copeland rule (top left), top cycle (top right), and Slater (bottom left), as well as the average running time to compute Slater winners (bottom right).

The top cycle consists of a unique member only for 46 of our tournaments (12.5%) and includes all the players in 268 of them (72%). While it was well-known that top cycles often are large, we found the extent to which this happens in the Size-12 dataset surprising. We note that the NBA tournaments, as well as those in their close vicinity, have top cycles with the most varied cardinalities.

In Figure 4d we show a map where each tournament's color gives the time needed to compute the Slater rule on this input using an ILP solver (computing Slater winners is NP-hard [2, 19, 18]): For each tournament we averaged the computation time over 15 attempts (average standard deviation was 5% of the running time). We see that the running time increases together with the distance from T_{ord}, without a clear dependence on how a particular tournament was generated, except that very structured tournaments (such as T_{rps} and members of the RPS/ord and ord/RPS families) require very little time (one could also point out that the most demanding instances belong to the family of "nonisomorphic" tournaments generated using `nauty`, but these tournaments are also farthest from T_{ord}).

Knockout Tournaments. Given a tournament graph T, by a knockout tournament we refer to a rooted, complete binary tree, where each leaf denotes a player from T and, then, each inner node is labeled with the player from one of the child nodes that, according to T, wins their game against the player from the other child node. The player in the root node is the winner. (The matching of the players to the leaves of the tree is called the *bracket*). The complexity of choosing a bracket so that a particular candidate wins was studied in depth in the literature [29, 42, 45, 4]; see also the overviews of Williams [46] and Suksompong [39] for further references (also regarding the necessary and sufficient conditions for a player to be a possible winner) and the work on bribery and manipulation in tournaments [31].

Along the lines of the above-mentioned literature, for each of the tournaments from the Size-12 dataset we perform the following experiment: We generate 1000 random brackets, and we analyze who

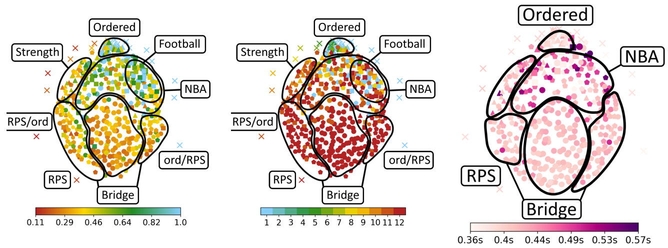


Figure 5: The winning probability of a player who wins most frequently in a knockout tournament based on the given one, under a random bracket (left); The number of possible knockout tournament winners (center), and an average time needed to check, using an ILP model, which players have brackets under which they win (right).

is the winner. In Figure 5 (left), we present the winning probability of a player who wins most frequently in a given tournament. Next, in Figure 5 (center) we present the number of players that may win a knockout tournament based on a given one (computed using an ILP model), and in Figure 5 (right) the time needed to find these possible winners (averaged over five attempts). In the majority of tournaments (i.e., 55% of them), for each player there is bracket under which he or she wins. Given these results, it would be interesting to see where real-life knockout tournaments (such as the tennis ones) appear on the map. However, to place them there, we would need the results of all possible matches, and not just those actually played.

While the maps in Figure 5 present different data than those in the preceding section, their colorings show similar trends. In particular, the depicted values either seem to be correlated with the distance from T_{ord} (left) or are most varied in the NBA area (center and right).

7 Conclusions and Future Work

We have adapted the "map of elections" framework to the case of tournament graphs. While doing so, we have found that direct methods for sampling tournaments seem to cover the space of those realistic tournaments that we analyzed, whereas using majority relations of ordinal elections (generated using typical statistical models) gives tournaments in quite a specific part of the space. We obtained two datasets of tournaments, one based on the NBA basketball league and one based on the Polish bridge leagues. Results of our experiments were either correlated with the distance from the ordered tournament T_{ord} or were most varied on the tournaments in the vicinity of the NBA ones (where also the election-based tournaments are located).

As far as future work goes, the most pressing issue is to find a distance that would be fast to compute, would capture relations between tournaments well, and could be applied to tournaments of different sizes. Further, it would be interesting to find statistical models of tournaments that appear in those parts of the map where our GED maps only show the tournaments from `nauty`.

8 Acknowledgements

This project has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (grant agreement No 101002854), from the French government under the management of Agence Nationale de la Recherche as part of the France 2030 program, reference ANR-23-IACL-0008.

References

[1] D. Aldous. Elo ratings and the sports model: A neglected topic in applied probability? *Statistical Science*, 32(4):616–629, 2017.

[2] N. Alon. Ranking tournaments. *SIAM Journal on Discrete Mathematics*, 20(1–2):137–142, 2006.

[3] H. Aziz, M. Brill, F. Fischer, P. Harrenstein, J. Lang, and H. Seedig. Possible and necessary winners of partial tournaments. *Journal of Artificial Intelligence Research*, 54:493–534, 2015.

[4] H. Aziz, S. Gaspers, S. Mackenzie, N. Mattei, P. Stursberg, and T. Walsh. Fixing balanced knockout and double elimination tournaments. *Artificial Intelligence*, 262:1–14, 2018.

[5] D. Baumeister and T. Hogrebe. Complexity of scheduling and predicting round-robin tournaments. In *Proceedings of AAMAS-21*, pages 178–186, 2021.

[6] N. Boehmer, R. Bredereck, P. Faliszewski, R. Niedermeier, and S. Szufa. Putting a compass on the map of elections. In *Proceedings of IJCAI-2021*, pages 59–65, 2021.

[7] N. Boehmer, R. Bredereck, E. Elkind, P. Faliszewski, and S. Szufa. Expected frequency matrices of elections: Computation, geometry, and preference learning. In *Proceedings of NeurIPS-2022*, 2022.

[8] N. Boehmer, P. Faliszewski, R. Niedermeier, S. Szufa, and T. Was. Understanding distance measures among elections. In *Proceedings of IJCAI-2022*, pages 102–108, 2022.

[9] N. Boehmer, P. Faliszewski, and S. Kraiczy. Properties of the Mallows model depending on the number of alternatives: A warning for an experimentalist. In *Proceedings of ICML-2023*, 2023.

[10] N. Boehmer, K. Heeger, and S. Szufa. A map of diverse synthetic stable matching instances. *Journal of Artificial Intelligence Research*, 79: 1113–1166, 2024.

[11] P. Bonacich. Factoring and weighting approaches to status scores and clique identification. *Journal of Mathematical Sociology*, 2(1):113–120, 1972.

[12] F. Brandt and H. Seedig. On the discriminative power of tournament solutions. In *Proceedings of the International Conference of the German Operations Research Society*, pages 53–58, 2014.

[13] F. Brandt, F. Fischer, and P. Harrenstein. The computational complexity of choice sets. *Mathematical Logic Quarterly*, 55(4):444–459, 2009.

[14] F. Brandt, M. Brill, F. Fischer, and P. Harrenstein. Minimal retentive sets in tournaments. *Social Choice and Welfare*, 42(3):551–574, 2014.

[15] F. Brandt, M. Brill, and P. Harrenstein. Tournament solutions. In F. Brandt, V. Conitzer, U. Endriss, J. Lang, and A. D. Procaccia, editors, *Handbook of Computational Social Choice*, pages 57–84. Cambridge University Press, 2016.

[16] F. Brandt, M. Brill, and P. Harrenstein. Extending tournament solutions. *Social Choice and Welfare*, 51(2):193–222, 2018.

[17] M. Brill, U. Schmidt-Kraepelin, and W. Suksompong. Margin of victory for tournament solutions. *Artificial Intelligence*, 302:103600, 2022.

[18] P. Charbit, S. Thomassé, and A. Yeo. The minimum feedback arc set problem is np-hard for tournaments. *Combinatorics, Probability and Computing*, 16(1):1–4, 2007.

[19] V. Conitzer. Computing Slater rankings using similarities among candidates. In *Proceedings of the 21st National Conference on Artificial Intelligence*, pages 613–619. AAAI Press, July 2006.

[20] M. Döring and J. Peters. Margin of victory for weighted tournament solutions. In *Proceedings of AAMAS-23*, pages 1716–1724, 2023.

[21] P. Erdös and J. Spencer. *Probabilistic Methods in Combinatorics*. Academic Press, 1974.

[22] P. Faliszewski, A. Kaczmarczyk, K. Sornat, S. Szufa, and T. Was. Diversity, agreement, and polarization in elections. In *Proceedings of IJCAI-2023*, pages 2684–2692, 2023.

[23] P. Faliszewski, M. Lackner, K. Sornat, and S. Szufa. An experimental comparison of multiwinner voting rules on approval elections. In *Proceedings of IJCAI-2023*, pages 2675–2683, 2023.

[24] P. Führlich, Á. Cseh, and P. Lenzner. Improving ranking quality and fairness in swiss-system chess tournaments. In *Proceedings of EC-2022*, pages 1101–1102, 2022.

[25] O. Hudry. A survey on the complexity of tournament solutions. *Mathematical Social Sciences*, 57(3):292–303, 2009.

[26] L. Katz. A new status index derived from sociometric analysis. *Psychometrika*, 18:39–43, 1953.

[27] D. Koschützki, K. Lehmann, L. Peeters, S. Richter, D. Tenfelde-Podehl, and O. Zlotowski. Centrality indices. In U. Brandes and T. Erlebach, editors, *Network Analysis: Methodological Foundations [outcome of a Dagstuhl seminar, 13-16 April 2004]*, pages 16–61. Springer, 2004.

[28] J. Kruskal. Multidimensional scaling by optimizing goodness of fit to a nonmetric hypothesis. *Psychometrika*, 29(1):1–27, 1964.

[29] J. Lang, M. S. Pini, F. Rossi, K. Venable, and T. Walsh. Winner determination in sequential majority voting. In *Proceedings of IJCAI-2007*, pages 1372–1377, 2007.

[30] J. Lerouge, Z. Abu-Aisheh, R. Raveaux, P. Héroux, and S. Adam. Exact graph edit distance computation using a binary linear program. In *Structural, Syntactic, and Statistical Pattern Recognition*, pages 485–495, Cham, 2016. Springer International Publishing.

[31] N. Mattei, J. Goldsmith, A. Klapper, and M. Mundhenk. On the complexity of bribery and manipulation in tournaments with uncertain information. *Journal of Applied Logic*, 13(4):557–581, 2015.

[32] B. McKay and A. Piperno. Practical graph isomorphism, II. *Journal of Symbolic Computation*, 60:94–112, 2014.

[33] OEIS Foundation Inc. The on-line encyclopedia of integer sequences, 2020. URL http://oeis.org/A008302.

[34] L. Page, S. Brin, R. Motwani, and T. Winograd. The PageRank citation ranking: Bringing order to the web. Technical Report TR-423, Stanford InfoLab, 1999.

[35] T. Russell. *A Computational Study of Problems in Sports*. PhD thesis, University of Waterloo, Ontario, Canada, 2010.

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

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

[38] A. Sanfeliu and K. Fu. A distance measure between attributed relational graphs for pattern recognition. *IEEE Transactions on Systems, Man, and Cybernetics*, 13(3):353–362, 1983.

[39] W. Suksompong. Tournaments in computational social choice: Recent developments. In *Proceedings of IJCAI-2021*, pages 4611–4618, 2021.

[40] S. Szufa, P. Faliszewski, P. Skowron, A. Slinko, and N. Talmon. Drawing a map of elections in the space of statistical cultures. In *Proceedings of AAMAS-20*, pages 1341–1349, 2020.

[41] S. Szufa, P. Faliszewski, L. Janeczko, M. Lackner, A. Slinko, K. Sornat, and N. Talmon. How to sample approval elections? In *Proceedings of IJCAI-2022*, pages 496–502, 2022.

[42] T. Vu, A. Altman, and Y. Shoham. On the complexity of schedule control problems for knockout tournaments. In *Proceedings of the 8th International Conference on Autonomous Agents and Multiagent Systems*, pages 225–232, 2009.

[43] T. Was and O. Skibski. An axiomatization of the eigenvector and Katz centralities. In *Proceedings of AAAI-2018*, pages 1258–1265, 2018.

[44] T. Was and O. Skibski. Axiomatic characterization of pagerank. *Artificial Intelligence*, 318:103900, 2023.

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

[46] V. V. Williams. Knockout tournaments. In *Handbook of Computational Social Choice*, chapter 19, pages 453–474. Cambridge University Press, 2016.

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