Robust Value Maximization in Challenge the Champ Tournaments with Probabilistic Outcomes

Umang Bhaskar
Tata Institute of Fundamental Research
Mumbai, India
umang@tifr.res.in

Juhi Chaudhary
Indian Institute of Petroleum and Energy
Visakhapatnam, India
juhi.math@iipe.ac.in

Sushmita Gupta
The Institute of Mathematical Sciences, HBNI
Chennai, India
sushmitagupta@imsc.res.in

Pallavi Jain
Indian Institute of Technology Jodhpur
Jodhpur, India
pallavi@iitj.ac.in

Sanjay Seetharaman
The Institute of Mathematical Sciences, HBNI
Chennai, India
sanjays@imsc.res.in

ABSTRACT

Challenge the Champ is a simple tournament format, where an ordering of the players — called a seeding — is decided. The first player in this order is the initial champ, and faces the next player. The outcome of each match decides the current champion, who faces the next player in the order. Each player also has a popularity, and the value of each match is the popularity of the winner. Value maximization in tournaments has been previously studied when each match has a deterministic outcome. However, match outcomes are often probabilistic, rather than deterministic. We study robust value maximization in Challenge the Champ tournaments, when the winner of a match may be probabilistic. That is, we seek to maximize the total value that is obtained, irrespective of the outcome of probabilistic matches. We show that even in simple binary settings, for non-adaptive algorithms, the optimal robust value — which we term the VnaR, or the value not at risk — is hard to approximate. However, if we allow adaptive algorithms that determine the order of challengers based on the outcomes of previous matches, or restrict the matches with probabilistic outcomes, we can obtain good approximations to the optimal VnaR.

KEYWORDS

Tournament Value Maximization; Robust Value; Approximation Algorithms; Hardness of Approximation; Challenge the Champ Tournaments; GTEP

ACM Reference Format:
Umang Bhaskar, Juhi Chaudhary, Sushmita Gupta, Pallavi Jain, and Sanjay Seetharaman. 2026. Robust Value Maximization in Challenge the Champ Tournaments with Probabilistic Outcomes. In *Proc. of the 25th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2026), Paphos, Cyprus, May 25 – 29, 2026*, IFAAMAS, 9 pages. https://doi.org/10.65109/TEVB4201

Proc. of the 25th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2026), C. Amato, L. Dennis, V. Mascardi, J. Thangarajah (eds.), May 25 – 29, 2026, Paphos, Cyprus. © 2026 International Foundation for Autonomous Agents and Multiagent Systems (www.ifaamas.org). https://doi.org/10.65109/TEVB4201

1 INTRODUCTION

Tournaments are a fundamental mechanism for structuring competition. Beyond their well-established role in organizing play in sports and e-sports, where both competitive fairness and entertainment value matter, they also arise in other decision-making contexts, such as elections, where pairwise comparisons are meaningful [10, 21, 23, 31].

In tournaments, some matches naturally generate more audience interest than others — either because they feature a marquee player who is highly popular or because the matchup, such as between arch-rivals, is anticipated to be dramatic. Organizers of sporting events, as well as competitive platforms more broadly, often try to structure play to maximize such engagement value, and in turn the revenue. An especially simple yet versatile format is the Challenge the Champ (CtC) tournament: a fixed "champion seat" is defended against a sequence of challengers whose order — the *seeding* — is chosen by the organizer. The seeding strongly shapes the tournament's trajectory, and thus, by choosing the seeding, the organizer significantly influences the excitement it generates.

Formally, a CtC tournament, is a permutation of the players, called the seeding. The first player is the initial champion and is challenged by the second player. The winner of this match then plays the third player, and so on. With n players, a winner (i.e., the final champion) is determined in $n - 1$ matches. CtC is a specific instance of single-elimination tournaments, historically used in boxing, martial arts, and other combat sports where challengers rise to take on the current champion (or master).

A major goal of tournament organizers is to maximize the value — this could be the revenue, total viewership, tickets sold, etc. While there are many tournament value functions studied, typically each possible match has a value, and the value of the tournament is the sum of the values of the matches played. If outcomes were perfectly predictable, this design question would then be easy to state: choose a seeding that maximizes the total value of the tournament. In real-world scenarios, some pairwise results may be essentially certain, perhaps due to a large skill gap, while others are inherently uncertain and are modeled by probabilities. This probabilistic setting is a natural way to model real-life competitions, a fact that was noted in earlier papers [2, 9, 31]. Existing works on designing CtC tournaments, and more general ones, have either assumed full determinism [5] or aimed to optimize expectation-based metrics such as expected wins or revenue [32]. But expectation alone can be misleading: a seeding that is strong on average may fail to produce enough, or any, exciting games if a few upsets occur. This motivates the search for a *robust guarantee* in tournament design.

In this paper, we introduce a risk-averse alternative, called the Value Not At Risk (VnaR), defined as the minimum total value an algorithm can obtain, over all realizations of the uncertain matches. The VnaR measures the value the organizer can guarantee, *no matter how the uncertain games turn out*. This parallels risk-aware reasoning in finance, where measures such as Value At Risk (VaR) quantify the guaranteed part of a portfolio's return: instead of maximizing average value, the designer secures a safe baseline excitement level even under unfavorable outcomes.

Our contribution in this work is two-fold: Conceptually, we propose VnaR as a natural, risk-averse tournament design objective for the realistic setting where game outcomes are not fully predictable. Algorithmically, we present the first study of VnaR-maximization in CtC tournaments, and present a comprehensive set of results. For non-adaptive algorithms, we first establish the computational hardness of the problem even under severe restrictions, including additive and multiplicative approximation hardness and propose almost matching polynomial-time approximation algorithms. We then study adaptive algorithms, where decisions regarding future matches are deferred until earlier game outcomes are known, and show that good approximations can indeed be obtained. We note that adaptive tournament formats are used in real-world tournaments also, such as the FIDE Grand Swiss Tournament, where the contestants are paired to ensure that each player plays an opponent with a similar running score without playing the same opponent more than once. Match pairing for each round is done after the previous round has ended and depends on its results.

In the past, CtC tournaments, also known as *stepladder* tournaments in prior work, have been studied from multiple perspectives. These include satisfying axiomatic notions of fairness [1] and characterizing strength graphs under which a designated favorite player can win the tournament [30, 33]. When the strength graphs are probabilistic, Mattei et al. [22] studied how players could be bribed to lower their chances of beating the initial champion, aiming to maximize the champion's overall probability of winning while staying within a given budget. Building on this, Chaudhary et al. [13] explored the problem further from the perspective of parameterized complexity. In this paper, we study adaptivity in the presence of probabilistic events. It is worth noting that adaptivity in *knockout tournaments*, a form of single-elimination tournament, has been examined recently by Chaudhary et al. [12], though primarily from the perspective of coalition manipulation. In that setting, coalition players can strategically decide in each round which games to throw, depending on the outcomes of the previous matches. To the best of our knowledge, this is the first work on probabilistic CtC tournaments, where the goal is to maximize an objective function that quantifies the number of valuable games.

Related Work. Starting with the influential work of Vu et al. [31], a number of tournament formats have been widely adopted and systematically studied, including the *knockout* format [9, 16, 28, 30], the *round-robin* format [24, 27], the *double-elimination* format [3, 26], and the *Swiss system* [25, 29]. Each of these formats represents a distinct organizational principle, offering particular advantages and limitations in terms of ranking outcomes and elimination procedures. We note that across these various tournament models the goal is often to determine if there is a seeding that allows a particular player to win, or to study the effects of bribery, winner determination, etc. Quite naturally, one of the primary goals of tournaments is to maximize revenue, generated either through advertising, sale of tickets, sale of broadcast rights, and so on. This aspect of tournaments is studied in prior work on knockout tournaments [11, 15, 17] and is now being studied for CtC tournaments as well [5].

Vu et al. [31] define the *probabilistic tournament fixing* question, where the input consists of a set of players, of which one is distinguished and the probability value p_{ij} associated with a game involving players i and j. The paper presented a recursive formula to calculate the winning probability of a player, and for the case of a balanced tournament, showed NP-hardness of deciding whether the distinguished player wins with at least some probability, even under additional restrictions. Subsequently, Aziz et al. [2] and Blazej et al. [9] have furthered the study of probabilistic tournament fixing in the knockout format.

Our Setting. We study tournament value maximization when the outcome of matches can be uncertain. We focus on CtC tournaments when the value of each match is either 1 or 0, determined by the popularity of the winner (this is termed *player-popularity-based* tournament values). Our input thus consists of a popularity (1 or 0) for each player, as well as a strength graph where additionally each directed edge, say (p_1, p_2), has a weight in $[0, 1]$, indicating the probability of the outcome determined by the edge direction. Thus, if edge (i, j) has weight $p_{ij} \in [0, 1]$, then in a match between these players, player i beats player j with probability p_{ij}, while j beats i with probability $1 - p_{ij}$. If the weight on an edge is 1, such a match (and the corresponding edge) is called *deterministic*, while a weight strictly less than 1 is an *uncertain* edge and the resulting outcome is said to be probabilistic. We say a player i *beats* j (or j is beaten by i) to mean that i beats j deterministically, i.e., $p_{ij} = 1$.

Define $\mathcal{P}$ as the set of popular players, $\mathcal{U}$ as the set of unpopular players that are beaten by at least one popular player, and $\mathcal{W}$ be the set of remaining unpopular players. Let $n_p = |\mathcal{P}|$ and $n_u = |\mathcal{U}|$. Prior work by Bhaskar et al. [5] shows that in CtC tournaments, in the absence of uncertainty (i.e., when every match is deterministic), the optimal seeding — the sequence of matches that maximizes the value — can be obtained in polynomial time. In fact, if n_p is the number of players with value 1 (termed *popular players*) and n_u is the number of players with value 0 (termed *unpopular players*) that are beaten by at least one popular player, then the optimal tournament value is exactly $n_p + n_u - 1$.

Value not at Risk. In this work, where some matches are uncertain, we focus on maximizing the total value obtained (equivalently, the number of matches won by popular players) in the *worst case with probability 1*. Our objective is thus to maximize the total value for a risk-averse tournament organizer. Formally stated, for a seeding σ, the *value not at risk* (the VnaR(σ)) is the total value obtained with probability 1. That is, for any possible outcomes of uncertain matches, the total value obtained is at least the VnaR. Our primary objective is to obtain a seeding with maximum possible VnaR. Note that this may require uncertain matches to be played. See Figure 1 for an illustration. The following proposition shows that, as in the deterministic setting, the unpopular players in $\mathcal{W}$ do not contribute to the optimal value.[1]

Proposition 1.1 (♣). *The optimal* VnaR *obtainable by any algorithm is at most* $n_p + n_u - 1$.

If VnaR* denotes the optimal VnaR of a given instance, we say an algorithm is an f-additive approximation if the value obtained for any outcome is at least VnaR* $- f$. Similarly, an algorithm is an f-multiplicative approximation if the value obtained for any outcome is at least $f \cdot$ VnaR*.

Our Contribution. We first note that a non-adaptive seeding with VnaR $n_p + \lceil \frac{n_u}{n_p} \rceil - 1$ is easily obtained as follows. Let a_p be a popular player that beats the maximum number of unpopular players in $\mathcal{U}$, say k. Clearly, $k \geq \lceil n_u/n_p \rceil$. In the seeding, we first put a_p, followed by the k unpopular players beaten by a_p. This is followed by the remaining popular players, and finally, all the other players. Under this seeding, the first $n_p + k - 1$ matches are all won by popular players, yielding a value of at least $n_p + \frac{n_u}{n_p} - 1$. From Proposition 1.1, if $n_u \leq c \cdot n_p$, a multiplicative $\max\{1/n_p, 1/(c+1)\}$-approximate non-adaptive seeding is thus readily obtainable.

We start with studying non-adaptive algorithms that output a *fixed seeding* without seeing any probabilistic outcomes. Even in this simple setting of binary values, we show strong negative results. In fact, our negative results hold when the optimal VnaR is $n_p + n_u - 1$, the maximum possible. These results imply that computing a fixed seeding that obtains a large value with probability 1 is hard, even when the optimal VnaR is actually the largest possible value, $n_p + n_u - 1$.

Our first hardness result is the following.

Theorem 1.2. *Unless* P=NP*, there is no non-adaptive polynomial-time algorithm that obtains a seeding that additively approximates the optimal* VnaR *with a factor of* $(n_p)^{1-\varepsilon}/(1+\alpha)$ *for any fixed* $0 < \varepsilon \leq 1$ *and* $\alpha > 0$.

Under the Exponential-time Hypothesis (ETH) [19], we have a stronger statement of hardness if $n_p^{\varepsilon} > (f(n_p)/2)\log^2 n_p$.

Theorem 1.3. *Assuming the* ETH*, there is no non-adaptive polynomial-time algorithm that obtains a seeding that additively approximates the optimal* VnaR *with a factor* $\frac{n_p}{f(n_p)\log^2 n_p}$*, for any polynomial-time computable and non-decreasing function* $f \in \omega(1)$.

Furthermore, we show that unless P=NP, no non-adaptive polynomial-time algorithm can multiplicatively approximate the optimal VnaR by a factor better than $1/\sqrt{n_p}$.

Theorem 1.4. *Unless* P=NP*, there is no non-adaptive polynomial-time algorithm that obtains a seeding that multiplicatively approximates the optimal* VnaR *with a factor* $\frac{1}{n_p^{1-\varepsilon}}$ *for any fixed* $0.5 < \varepsilon \leq 1$.

Let OPT denote the optimal VnaR of an instance. We note that Theorem 1.2 rules out a polynomial-time algorithm that obtains a seeding of VnaR at least OPT $- (n_p)^{1-a}/(1+\alpha)$, where $0 < a \leq 1$, while Theorem 1.4 rules out one with VnaR at least OPT$/(n_p)^{1-b}$, where $0.5 < b \leq 1$. Thus, when n_p is sufficiently large, the latter gives a stronger inapproximability guarantee than the former.

[1] Statements marked ♣ are proved in the full version of the paper [6].

Given the negative results for non-adaptive algorithms, we then study adaptive algorithms that can obtain, by deferring the decision about who is introduced next, a large value with probability 1. Thus such an algorithm constructs a seeding adaptively, depending on the outcome of previous matches. For adaptive algorithms, we are able to show an additive approximation to the optimal VnaR that depends on the *chromatic number* of a certain graph.[2] In particular, let $G_p = (V_p, E_p)$ be the subgraph of the strength graph such that V_p is the set of popular players and E_p is the set of uncertain edges incident on V_p.

Theorem 1.5. *Given a proper coloring of* G_p *with* k *colors, there is a polynomial-time adaptive algorithm that obtains* VnaR *at least* $(n_p + n_u - 1) - (k - 1)$.

We note that, in general the chromatic number is hard to approximate [20] but if the number of uncertain edges is small (say q), one can indeed compute the chromatic number of G_p in time $O^*(4^q)$, where the O^* notation hides polynomial factors [8].

Observing that $k \leq n_p$, and combined with the deterministic seeding that obtains VnaR $n_p + \lceil n_u/n_p \rceil - 1$, we obtain the following result, demonstrating the large gap between adaptive and non-adaptive algorithms for this problem.

Theorem 1.6 (♣). *There is an adaptive algorithm that obtains* VnaR *at least* $\left(n_p + n_u - 1\right)/2$.

1.1 Notation and Preliminaries

Let $n = |N|$, where N denotes the set of players in the input. The *strength graph* $G = (N, E)$ has a directed edge between every pair of players, and a weight in $[0, 1]$ for each edge. We assume the match outcomes are independent.

Recall that a seeding (usually denoted σ) is a permutation of the players, that determines the order of challengers in a CtC tournament. The player $\sigma(1)$ is the initial champion, and $\sigma(2)$ is the first challenger. The winner of this match is challenged by player $\sigma(3)$, and so on. Hence, there are exactly $n - 1$ matches. The *value of a match* is determined by the popularity of the winner of the match, and the *value of a seeding* is the sum of the values of the matches played. Note that certain matches may have uncertain outcomes, and hence the value of a seeding — as well as the sequence of matches played —may be probabilistic.

Sub-seedings. A seeding σ_0 is said to be a *sub-seeding* of σ if σ_0 is a contiguous sub-permutation of σ. It is helpful to describe a sub-seeding using intervals on players. That is, for two players $a, b \in N$, we use $\sigma[a, b]$ to denote the sub-seeding that includes all players between a and b in the permutation (including a and b). The sub-seedings $\sigma[a, b)$, $\sigma(a, b]$, $\sigma(a, b)$, exclude b, a, and both a and b respectively. For a player $a \in N$, we use $\sigma[*, a)$ to denote the set of all players introduced until a, excluding a. The sub-seedings $\sigma[a, *]$, $\sigma(a, *]$, and $\sigma[*, a]$ are defined analogously.

Arborescences and Backbones. A *caterpillar arborescence* (arborescence, in short) is a directed graph where removing all vertices with out-degree zero yields a directed path. A seeding σ of the players induces several spanning arborescences among the players,

[2] The *chromatic number* of a graph is the minimum number of colors required to obtain a proper coloring of the graph, i.e., an assignment of colors to the vertices of such that no two adjacent vertices receive the same color.

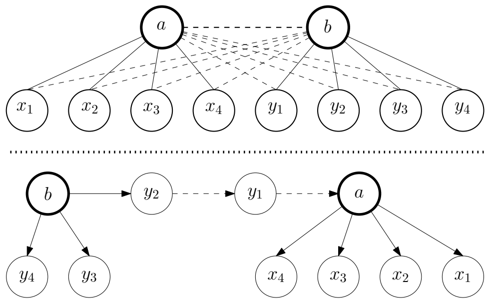

Figure 1: Illustration of a strength graph, and a spanning arborescence. The instance contains two popular players, a and b, and eight unpopular players $x_1, \ldots, x_4$ and $y_1, \ldots, y_4$. Solid(dashed) edges are deterministic(uncertain). Edges between unpopular players are uncertain and omitted for clarity. The backbone of the arborescence is formed by the vertices $\{b, y_2, y_1, a\}$. The VnaR is 7, and is witnessed by the seeding $\langle a, x_1, x_2, x_3, x_4, y_1, y_2, b, y_3, y_4 \rangle$.

depending on the outcomes of the probabilistic games. Note that if the seeding results only in deterministic games, then the spanning arborescence is unique. It is important to note that *an arborescence implies a seeding as well as a sequence of match outcomes*, some of which may be probabilistic. This observation is used critically in our hardness proof.

For an arborescence, the *backbone* refers to the path obtained after the removal of the leaves. Note that the players in the backbone are those that win at least one game in the tournament. Any player who loses a game upon introduction has out-degree zero in the arborescence and is called the *child* of the player it is beaten by.

Convention about backbone. We note that an edge $u \to v$ on the backbone indicates the following: player v was the champion when player u was introduced, and u subsequently defeated v. In all figures, we position player u to the left of player v.

Arborescence with the Minimum Value. Note that for a given seeding, there can be many resulting arborescences as a result of the sequence of probabilistic outcomes. The one that yields the lowest value is of special interest to us. In fact, when we refer to the VnaR score of a seeding, we are actually referring to the value obtained by the least value arborescence. Thus, VnaR of the tournament τ is the maximum VnaR score among all possible seedings.

Mathematically, let σ be a seeding and let $\mathcal{A}(\sigma)$ denote the set of all arborescences that can result from σ under different realizations of the probabilistic games. We use score(A) to denote the value of an arborescence A; that is, the sum of the values of the matches played. Moreover, the score of a sub-seeding σ_0 of σ in the arborescence/outcome A_σ is the sum of the popularity values of the matches played by players in σ_0 in the outcome A_σ. The VnaR score of σ is defined as $\text{VnaR}(\sigma) = \min_{A \in \mathcal{A}(\sigma)} \text{score}(A)$. The VnaR of the tournament is then defined as $\text{VnaR}(\tau) = \max_{\sigma \in \Sigma} \text{VnaR}(\sigma)$, where Σ is the set of all possible seedings.

The following result about the *minimum-value arborescence* is a critical tool in our analysis.

Lemma 1.7 (♣). *For any given seeding σ, a minimum-value arborescence can be computed in time $O(n^2)$.*

2 THE HARDNESS OF MAXIMIZING VNAR

In the Hamiltonian Path problem, we are given a directed graph and the goal is to decide if there is a path that visits each vertex exactly once (such a path is called a *Hamiltonian path*). This problem has been shown to be NP-hard even in simple digraphs [4], where any two vertices have at most one edge between them.[3]. We prove the hardness of maximizing VnaR via a reduction from Hamiltonian Path.

Let $D = (V, \hat{E})$ be an instance of Hamiltonian Path where D is a simple directed graph on $\hat{n}$ vertices. We construct an instance $\mathcal{T}$ with strength graph $G = (V_p \cup V_u, E)$ as follows.

- $V_p = \{v_p : v \in V\}$ and all players in V_p have popularity 1. They form the set of popular players. Note that the number of popular players $n_p = \hat{n}$. For the sake of simplicity, we sometimes refer to popular players and the popular vertices interchangeably;
- $V_u = \{v_u : v \in V\}$ and all players in V_u have popularity 0. They form the set of unpopular players. The player v_u is called the *unpopular copy* of v_p, and v_p is called the *popular copy* of v_u;
- For each $v \in V$, add $v_p \to v_u$ to E (that is, v_p beats v_u);
- For each pair $\{a, b\} \subseteq V$, add $a_u \to b_p$ to E;
- For each pair $\{a, b\} \subseteq V$,
  (1) if $a \to b \in \hat{E}$, we add $a_p \to b_p$ to E,
  (2) else (there is no edge between a and b in $\hat{E}$, and) we add an uncertain edge between a_p and b_p to E;
- For any pair $\{a, b\} \subseteq V_p \cup V_u$ that has not been considered before, we add an uncertain edge between a and b to E.

The following lemma establishes the equivalence between Hamiltonian Path and maximizing VnaR.

Lemma 2.1 (♣). *D has a Hamiltonian path if and only if there is a seeding in $\mathcal{T}$ with* $\text{VnaR} \geq 2n_p - 1$.

In the following sections, we show that the above reduction establishes the hardness of finding even approximately good solutions. Our hardness results rely on the following result by Björklund, Husfeldt, and Khanna [7, Theorem 4], on finding long paths.

Proposition 2.2 ([7]). *Unless* P=NP, *there is no deterministic, polynomial-time algorithm that finds a path of length $(\hat{n}-1)/\hat{n}^{1-\varepsilon}$ for any fixed $0 < \varepsilon \leq 1$ even in simple[4] directed graphs on $\hat{n}$ vertices where a Hamiltonian path is guaranteed to exist.*

2.1 Additive Hardness of Approximation

Let $D = (V, \hat{E})$ be a simple directed graph on $\hat{n}$ vertices where a Hamiltonian path is guaranteed to exist. We construct an instance $\mathcal{T}$ with strength graph $G = (V_p \cup V_u, E)$ on $n = \hat{n} + \hat{n}$ players following the same reduction that establishes Lemma 2.1. Note that $n_p = n_u = \hat{n}$. Observe that in the reduced instance $\mathcal{T}$, it is straightforward to find a seeding with VnaR $n_p - 1$ by first introducing the popular players in an arbitrary order, followed by the unpopular players in an arbitrary order. Towards establishing the hardness of finding an approximately good seeding, we study the interesting case where the computed seeding has a score strictly

[3]We note that Theorem 6.1.3 in [4] shows that the closely related Hamiltonian Cycle is hard, and a simple modification (removal of certain edges) gives us the desired result.
[4]Though not explicitly stated so, one can observe that the given reductions involve only simple directed graphs.

greater than $n_p - 1$. Suppose that a polynomial-time algorithm computes a seeding σ with VNAR $n_p - 1 + \gamma$ for some $\gamma > 0$.

The following lemmas on the structure of a minimum-value arborescence A_σ (obtained by applying Lemma 1.7) allow us to establish the inapproximability.

Lemma 2.3 (♣). *There are at least $\gamma - 1$ popular players on A_σ that have their unpopular copies as a child.*

Lemma 2.4 (♣). *There are at most $n_p - \gamma$ unpopular players on the backbone of A_σ.*

Lemma 2.5 (♣). *Suppose that a_p, b_p are two popular players such that in A_σ, a_p is introduced after b_p, and both a_u is a child of a_p and b_u is a child of b_p. Suppose that only popular players are introduced in $A_\sigma(a_u, b_u)$, and let X denote that set. Then, there is a deterministic path P from a to b in D with intermediate vertices corresponding to those from X alone, that is, $P \subseteq X \cup \{a, b\}$. Moreover, such a path can be computed in polynomial time.*

Now, we are ready to prove our result. Applying the pigeonhole principle on Lemmas 2.3 and 2.4, we obtain a sequence S of $(\gamma - 1)/(n_p - \gamma)$ popular players such that there is no unpopular player on the backbone between any two players in S. After repeated application of Lemma 2.5 (on adjacent players in S), we obtain a deterministic path of length $(\gamma - 1)/(n_p - \gamma) - 1$ in the graph D. Setting $\gamma = (1+n_p\delta)/(1+\delta)$ where $\delta = 1+(n_p-1)/n_p^{1-\varepsilon}$, we obtain a deterministic path of length $(n_p-1)/n_p^{1-\varepsilon}$ in D, that is NP-hard to find. So unless P=NP, there is no polynomial-time algorithm that can find a seeding with score at least $n_p-1+\gamma = (2n_p-1)-(n_p-\gamma)$.

Recall that D is assumed to contain a Hamiltonian path. By Lemma 2.1, we know that the reduced instance has VNAR $2n_p - 1$. Thus, we rule out a polynomial-time additive approximation algorithm of factor at most $n_p - \gamma$. Moreover, we show the following which completes the proof of Theorem 1.2.

Lemma 2.6 (♣). *For any $0 < \varepsilon < 1$ and $\alpha > 0$ and sufficiently large $n_p \in \mathbb{N}$, we have $n_p^{1-\varepsilon}/(1+\alpha) \leq n_p - \gamma$.*

Björklund, Husfeldt, and Khanna [7] also showed that assuming the ETH [19] (a stronger conjecture than P ≠ NP), no polynomial-time algorithm can find a path of length $f(n_p)\log^2 n_p$ for any polynomial-time computable and non-decreasing function $f \in \omega(1)$ even in simple directed graphs on n_p vertices where a Hamiltonian path is guaranteed to exist. Fix any such function f. Observe that $f/2 \in \omega(1)$. Setting $\gamma = (1 + n_p\delta)/(1 + \delta)$ where $\delta = 1 + (f(n_p)/2)\log^2 n_p$, we obtain a deterministic path of length $(f(n_p)/2)\log^2 n_p$ in D, that is hard to find. Following the same arguments as before, we rule out approximation factors that are at most $n_p - \gamma = (n_p-1)/(2+(f(n_p)/2)\log^2 n_p)$. In particular, we rule out factor $n_p/f(n_p)\log^2 n_p \leq (n_p-1)/(2\times(f(n_p)/2)\log^2 n_p) \leq n_p-\gamma$. Observe that when $n_p^{\varepsilon} > (f(n_p)/2)\log^2 n_p$ the approximation factor ruled out by assuming ETH is higher than the approximation factor ruled out by assuming P ≠ NP. Thus, we get a stronger inapproximability result depending on the choice of f. Overall, we have proved Theorem 1.3.

2.2 Multiplicative Hardness of Approximation

In this subsection, we establish a different hardness of approximation, via a slightly different reduction compared to the one in Section 2.1. Let $D = (V, \hat{E})$ be a simple directed graph on $\hat{n}$ vertices where a Hamiltonian path is guaranteed to exist. Let $d = \hat{n}^2$. We construct an instance $\mathcal{T}$ with strength graph $G = (V_p \cup V_u, E)$ on $n = \hat{n} + \hat{n}d$ players as follows.

- $V_p = \{v_p : v \in V\}$ and all players in V_p have popularity 1. They form the set of popular players; note that $n_p = \hat{n}$);
- $V_u = \{v_{u,i} : v \in V, i \in [d]\}$ and all players in V_u have popularity 0. They form the set of unpopular players. For each $v \in V$, we call the players in $\{v_{u,i} : i \in [d]\}$ as the unpopular copies of v;
- For each $v \in V$ and $i \in [d]$, add $v_p \to v_{u,i}$ to E (that is, v_p beats its unpopular copies);
- For each pair $\{a, b\} \subseteq V$,
  (1) if $a \to b \in \hat{E}$, we add $a_p \to b_p$ to E;
  (2) else (there is no edge between a and b in $\hat{E}$, and) we add an uncertain edge between a_p and b_p to E.
- For any pair $\{a, b\} \subseteq V_p \cup V_u$ that has not been considered before, we add an uncertain edge between a and b to E.

Suppose that a polynomial-time algorithm computes a seeding σ with VNAR at least $\ell d + n_p - 1$ for some $\ell = (n_p)^{\varepsilon}$ where $\varepsilon > 0.5$. Note that the VNAR of a seeding can be computed in polynomial time, Lemma 1.7. Let A_σ denote the spanning arborescence of σ that corresponds to the (sequence of) following probabilistic events:

- whenever there is a probabilistic game between an unpopular and a popular player, the unpopular player wins.

By construction, we have the following equivalent description of the probabilistic events that generate the arborescence A_σ: a game between a popular player v_p and an unpopular player w_u in A_σ is won by v_p implies that $v = w$ (that is, the latter is a copy of the former), else it is won by w_u. We begin with the following property.

Observation 2.7. *Any popular player in the backbone of A_σ who has an unpopular child must be the latter's copy.*

Due to the definition of σ, it follows that it has a prefix whose VNAR is exactly $\ell d + n_p - 1$. Our argument will focus on this prefix. Formally, we proceed as follows.

Let σ_0 be the minimal-prefix sub-seeding of σ, such that the partial arborescence A_{σ_0} obtained from A_σ by only considering the players in σ_0 has score exactly $\ell d + n_p - 1$. In what follows, we will establish a set of structural properties about the sub-seeding σ_0 and the corresponding arborescence A_{σ_0}. These properties will enable us to obtain a sufficiently long path in D which establishes the inapproximability of VNAR. For the ease of exposition and to aid modularity, we break the analysis into paragraphs.

Decomposing the seeding into blocks. We will begin by decomposing the seeding into blocks punctuated by popular players that satisfy certain specific properties. The underlying idea is that these blocks allow us to identify deterministic paths, which are either present in the backbone or must be inferred from the uncertain edges. In certain scenarios, there is a large score increase if certain probabilistic outcomes are different. This score increase, in turn, allows us to identify a deterministic path such that the score increase is proportional to the length of that path in D.

We begin our formal discussion by identifying a scenario where a deterministic path in D can be easily identified. Let v_u^1 be the

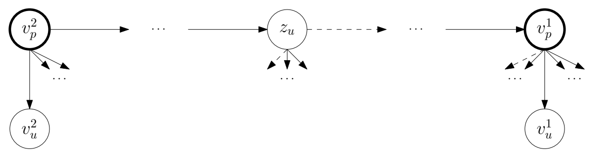

Figure 2: Illustration of a tagged block due to partners (v_p^1, v_p^2). Here, the bold circles represent the popular backbone players, and the normal circles are unpopular players. The solid(dashed) arrows represent the deterministic (uncertain) edges between players.

last-introduced (in A_{σ_0}) unpopular child of v_p^1 and v_u^2 be the first-introduced (in A_{σ_0}) unpopular child of v_p^2.

Definition 2.8. We define a *clean pair* to be a pair of popular players (v_p^1, v_p^2) if they satisfy the following properties in A_{σ_0}:

(i) both are on the backbone of A_{σ_0} with v_p^2 introduced after v_p^1;
(ii) each of them has at least one unpopular child;
(iii) there is no other popular player on the backbone between them who has an unpopular child; and
(iv) either the backbone between them contains only popular players, or there is a deterministic path from v_p^2 to v_p^1 in D such that the intermediate vertices correspond to the popular players introduced between v_u^1 and v_u^2 in σ_0.

Moreover, the sub-seeding $\sigma_0[v_p^1, v_u^2]$ is called a *clean block.*

The following lemma identifies a scenario when a deterministic path is easy to detect: the existence of a clean block.

Lemma 2.9 (♣). *Let $(v_p^{i,1}, v_p^{i,2})$, denote a clean block. Let X denote the set of popular players introduced in $\sigma_0(v_u^{i,1}, v_u^{i,2})$. Then, there exists a deterministic path P from $v^{i,2}$ to $v^{i,1}$ in D such that the intermediate vertices correspond to those from X alone, that is, $P \subseteq X \cup \{v_p^{i,1}, v_p^{i,2}\}$. Moreover, such a path can be computed in polynomial time.*

We will show that A_{σ_0} consists of several consecutive clean blocks which allow us to identify a sufficiently long path. Towards that, we analyze the possibility that between two popular players in the backbone, whose respective children are their unpopular copies, there exists an unpopular player in the backbone.

Definition 2.10. We define a *tagged pair* as a pair of popular players (v_p^1, v_p^2), illustrated in Figure 2, if they satisfy the following properties in A_{σ_0}:

(i) both are on the backbone of A_{σ_0} with v_p^2 introduced after v_p^1;
(ii) each of them has at least one unpopular child;
(iii) there is no other popular player on the backbone between them who has an unpopular child; and
(iv) the backbone between them contains at least one unpopular player, and there is no path in the graph D from v_p^2 to v_p^1 where the intermediate path vertices correspond to the popular players introduced between v_u^1 and v_u^2 in σ_0.

Moreover, the members of a tagged pair are called the *(tagged) partners*, and the sub-seeding $\sigma_0[v_p^1, v_u^2]$ is called a *tagged block.*

Let $\{(v_p^{i,1}, v_p^{i,2})\}_{i\geq 1}$ denote the set of tagged pairs in σ_0, indexed in the order in which they are introduced in σ_0. Note that condition (iii) ensures that these pairs cannot interleave. Additionally, we will call the members of a tagged pair to be (tagged) partners.

Consider a tagged pair $(v_p^{i,1}, v_p^{i,2})$: from now on, we call it the i^{th} pair. As in the case of clean blocks, we will use v_u^1 to denote the last-introduced (in A_{σ_0}) unpopular child of v_p^1 and v_u^2 to denote the first-introduced (in A_{σ_0}) unpopular child of v_p^2.

Let x_p^i denote either $v_p^{i,1}$ or the last popular player that is introduced between $v_u^{i,1}$ and $v_u^{i,2}$ such that there is a deterministic path in the graph D from the vertex x^i to $v^{i,1}$ in which the intermediate vertices of the path correspond to the popular players introduced between the tagged partners in σ_0. (Note that this deterministic path also exists in the reduced instance within the subgraph induced on the popular players.)

We identify a special player w^i depending on who is the champion right after the introduction of x_p^i. In the following definition, we consider the champion, denoted by c, right after x_p^i is introduced. Thus, c is either x_p^i or whoever is the champion right before the introduction of x_p^i. We define w^i as follows.

- If x_p^i is *not* deterministically beaten by the previous champion, we set $w^i = x_p^i$.
- Else, $w^i = c$.

Note that, irrespective of what value it takes, w^i is a popular player. We first prove the following result.

Lemma 2.11 (♣). *Consider any player y introduced in $\sigma_0[x_p^i, v_u^{i,2})$ such that $y \neq w^i$. Then player y either deterministically loses to w^i or has an uncertain edge to w^i.*

We prove the following result, which allows us to analyze an alternate event in which the score of a tagged block may increase.

Lemma 2.12 (♣). *There is a probabilistic event in which w^i beats everyone from its introduction till $v_u^{i,2}$ and then loses to $v_u^{i,2}$.*

Winning the block. We refer to the condition of w^i beating everyone till the introduction of $v_u^{i,2}$ and then losing to $v_u^{i,2}$, as w^i *winning the block.*

Since w^i is a popular player, w^i winning the block ensures that, in the resulting arborescence, every game after the introduction of w^i up to the introduction of $v^{i,2}$ gives value. Thus, the score up to here is at least as much as the score obtained by A_{σ_0}. But, there is zero score after the introduction of $v_u^{i,2}$, since the unpopular player $v_u^{i,2}$ wins all the remaining games.

In what follows, we will analyze the score increase due to w^i winning the block. We will argue that if the score increases, due to w^i winning the block, is upper-bounded, then the score of the seeding up to the start of that block is lower bounded, Lemma 2.13. This allows us to identify a prefix of the seeding in which all tagged blocks have large score increases.

Let s^i denote the change in score from the original arborescence A_{σ_0} to the one when w^i is winning the block. If $s^i \leq \ell d/2 + n_p - 1$, then we call i a *low-scoring* block.

The following result, Lemma 2.13, allows us to bound the value of the score *inside* a tagged block. We use the property of w^i winning the block to give an estimate of the score that can be obtained within a tagged block. At the heart of this argument is the fact

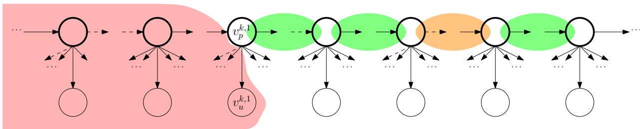

Figure 3: A figure illustrating the key ideas used thus far in the proof of Theorem 1.4. Here we represent A_σ using the same conventions as Figure 2. The green and orange ovals denote clean blocks and high-scoring tagged blocks, respectively. The first low-scoring tagged block starts with $v_p^{k,1}$. The red area is the removed portion because of a low-scoring tagged block.

that in A_{σ_0} only the popular players on the backbone between the tagged partners contribute to the score by playing each other. But since there are unpopular players in the backbone, the score obtained from within a tagged block can actually be higher, such as when w^i wins the block. This insight allows us to bound the score of the sub-seeding up to the introduction of the first partner of the first low-scoring (tagged) block, Lemma 2.14.

Lemma 2.13 (♣). *The score obtained by the sub-seeding $\sigma_0[v_p^{i,1}, *]$ when w^i wins the block is at most $s^i + n_p - 1$.*

The next proof is crucial to our proof of detecting sufficiently long paths in D. We do so by showing that there is a prefix with large score and no low scoring tagged blocks. More specifically, we show that there is *prefix* of σ_0, namely $\sigma_0[*, v_p^{k,1})$, where k is the first low scoring block, that must have a large score.

Lemma 2.14 (♣). *Suppose that $\sigma_0[v_p^{k,1}, v_u^{k,2}]$ denotes the first low-scoring block in A_{σ_0}. Then, the score obtained in A_{σ_0} due to the sub-seeding $\sigma_0[*, v_p^{k,1})$ is at least $\ell d/2 - n_p$.*

Now we are ready to obtain a prefix σ_1 of σ_0 (and thus, of σ as well) that does not contain any low-scoring blocks. If σ_0 does not contain any low-scoring blocks, then we set $\sigma_1 = \sigma_0$. Otherwise, we set $\sigma_1 = \sigma_0[*, v_p^{k,1})$, where k denotes the first low scoring tagged block. From now on, we use A_{σ_1} to denote the partial arborescence obtained from A_{σ_0} by only considering the players in σ_1. Applying Lemma 2.14, we have the following.

Corollary 2.15. *There is a prefix σ_1 of σ_0, such that the score obtained by the arborescence A_{σ_1} is at least $\ell d/2 - n_p$. Moreover, if there is a tagged block i in σ_1, then $s^i > \ell d/2 + n_p - 1$.*

Consequently, what remains to be shown is how the existence of a prefix with a large score allows us to obtain a path of a certain length. Figure 3 illustrates a snapshot of the arborescence at this point in the proof.

Computing a Long Path. Using Lemma 2.9, we know that in clean blocks, there are deterministic paths connecting the two popular partners of the block with intermediate vertices from the block itself. By stitching together these paths, we can find a longer path in the graph D. However, it is not clear how to deal with tagged blocks – pairs of popular players who have their unpopular copies as children but do not have a deterministic path.

In the following lemma, we address this matter based on the fact that the score of A_{σ_1} is at least $\ell d/2 - n_p$. The critical idea in the proof is to pin-point a sub-seeding of σ_1 which does not have a tagged block, and the backbone does not have any unpopular players and has a large number of popular players who beat each of their unpopular copies. Such a sub-seeding implies a directed path, as shown earlier in Lemma 2.5.

Lemma 2.16 (♣). *For large enough n_p, there exists an $\varepsilon' > 0$ such that in polynomial time we can find in D a path of length at least $n_p^{\varepsilon'}$.*

Recall that the input digraph D is assumed to contain a Hamiltonian path, and it is NP-hard to find a path of length $(n_p - 1)/n_p^{1-\varepsilon''} \leq n_p^{\varepsilon'}$ for an appropriately chosen ε'' given ε' (Proposition 2.2). Combining Lemma 2.16 with the fact that the VNaR of σ is at least $\ell d + n_p - 1$ where $\ell = n_p^{\varepsilon}$ with $0.5 < \varepsilon \leq 1$, we have proved our hardness of approximation, Theorem 1.4.

3 ALGORITHMS FOR THE VNAR

We first show a non-adaptive algorithm for a special case where each uncertain edge in the strength graph is either between two popular players, or two unpopular players. Thus, any edge between a popular and an unpopular player is deterministic. We establish a lower bound on VNaR in terms of the chromatic number of the graph $G_p = (V_p, E_p)$, which is a subgraph of the strength graph where V_p is the set of popular players and E_p is the set of uncertain edges incident on V_p.

Theorem 3.1. *Suppose that for each uncertain edge in the strength graph, the end points are either both popular or both unpopular. There is a seeding with VNaR at least $(n_p + n_u - 1) - (k - 1)$, where k is the chromatic number of G_p.*

Proof. As before, we say a player p *beats* p' if p beats p' deterministically. We describe our algorithm for the required seeding.

(1) Let $C_1, \ldots, C_k$ be a proper k-coloring of G_p. Note that there is no edge between two vertices in any set/color class C_i. In particular, there is no uncertain edge between any two vertices in C_i.

(2) For each $i \in [k]$, let P_i be a Hamiltonian path in the strength graph restricted to the vertices of C_i; note that such a path exists (and can be found in polynomial time) because the considered graph is a complete directed graph. Let f_i and ℓ_i denote the first and last vertices of P_i, respectively. We will process the color classes in the increasing order of their label.

(3) We construct the seeding by introducing the player ℓ_1 first, and then sequentially introducing each unpopular player beaten by ℓ_1 (in an arbitrary order), one at a time. After this, we introduce the next popular player along the Hamiltonian path, and then each unpopular player beaten by it (who has not played before). Continuing in this manner we place all the vertices in C_1, and all the unpopular players beaten by at least one player in P_1. Observe that the champion after all players in C_1 have played is f_1.

(4) We consider the relationship between ℓ_2 and f_1. If ℓ_2 beats f_1, we proceed by introducing ℓ_2, then each unpopular player beaten by it, and so on following Step 3. Otherwise, either f_1 beats ℓ_2 or there is an uncertain edge between f_1 and ℓ_2. Let X be the set of unpopular players that are beaten by ℓ_2. One of the following two scenarios must occur:

(a) There exists an unpopular player $v \in X$ that beats f_1. Then, we first introduce v in the seeding, followed by ℓ_2, and then the remaining vertices in X (if any); and then move along the path P_2 following Step 3.

(b) All players in X are beaten by f_1. Then , we first introduce all vertices in X, followed by ℓ_2. If ℓ_2 wins, we move along the path P_2 following Step 3; else, we repeat the same arguments while comparing f_1 with the next player along the path P_2, and continue the above process on the remaining vertices in this color class, and then on the remaining color classes.

Next, we complete the proof of the theorem by bounding the value of the constructed seeding.

Lemma 3.5.1 (♣). *The constructed seeding has* VNAR *at least* $(n_p + n_u - 1) - (k - 1)$. □

Recall that we supposed that we have an instance in which for each uncertain edge, the end points are either both popular or both unpopular. Let VNAR* denote the optimal VNAR score in the given instance. Note that the proof of Theorem 3.1 yields the following result since each step can be executed in polynomial time.

Observation 3.2. *Given a proper coloring of* G_p *with* $\hat{k}$ *colors, a seeding with* VNAR *at least* $(n_p + n_u - 1) - (\hat{k} - 1) \geq \text{VNAR}^* - (\hat{k} - 1)$ *can be computed in polynomial time.*

Let VC_{prob} denote the size of a minimum vertex cover in G_p. It is known that a vertex cover in G_p of size at most 2VC_{prob} can be computed in polynomial time. Let S denote such a vertex cover. From S, we obtain a $(|S| + 1)$-proper coloring of G_p as follows: $|S|$ many colors for each vertex in S and 1 color for the vertices in $V_p \setminus S$. Applying Observation 3.2, we have the following.

THEOREM 3.3. *Suppose that for each uncertain edge in the strength graph, the end points are either both popular or both unpopular. A seeding with* VNAR *at least* $n_p + n_u - 1 - (2\text{VC}_{\text{prob}} - 1) \geq \text{VNAR}^* - (2\text{VC}_{\text{prob}} - 1)$ *can be computed in polynomial time.*

Beyond Polynomial Time. More generally, consider any graph class C for which there is an algorithm, be it exact-exponential, FPT or XP etc, to compute the chromatic decomposition with time complexity T_C. Then, we can use that algorithm in conjunction with our construction to create a seeding with a guaranteed score.

THEOREM 3.4. *Suppose that a proper coloring of* G_p *that uses k colors can be computed in time* $T(G_p)$. *Then, there exists an algorithm that computes a seeding with* VNAR *at least* $(n_p+n_u-1)-(k-1)$ *in time* $T(G_p)+n^{O(1)}$, *where n is the number of players in the instance.*

Specifically, for classes for which an optimal chromatic decomposition can be found efficiently, we have a polynomial time algorithm that can compute a seeding of value at least $(n_p+n_u-1)-(\chi(G_p)-1)$. Such classes include perfect graphs and bipartite graphs, where the problem is polynomial-time solvable; bounded tree-width graphs, where it is FPT w.r.t. $k + tw$ [14]; and P_5-free graphs, where it is XP w.r.t. k [18].

From now on, we consider the problem without any restrictions on the end points of uncertain edges. We note that the instance in Figure 1 exhibits an example where the guarantees of Theorem 1.5 do not hold. Observe that in that instance we have $n_p = 2$, $n_u = 8$, $\chi(G_p) = 2$, and although the bound given by Theorem 1.5 is 8, the VNAR is only 7, witnessed by the seeding $\langle a, x_1, x_2, x_3, x_4, y_1, y_2, b, y_3, y_4 \rangle$.

3.1 Adaptive Algorithm

Recall that an algorithm is *adaptive* if it constructs a seeding adaptively, depending on the outcomes of earlier matches.

THEOREM 1.5. *Given a proper coloring of* G_p *with k colors, there is a polynomial-time adaptive algorithm that obtains* VNAR *at least* $(n_p + n_u - 1) - (k - 1)$.

PROOF. The first three steps of our algorithm are the same as that in the proof of Theorem 3.1, and thus we describe the remaining steps. We consider the relationship between ℓ_2 and f_1.

(4) If ℓ_2 beats f_1, we proceed by introducing ℓ_2, then each unpopular player beaten by ℓ_2, and so on following Step 3. Otherwise, either f_1 beats ℓ_2 or there is an uncertain edge between f_1 and ℓ_2. Let X be the set of unpopular players that are beaten by ℓ_2. Note that there is no player in X that loses to f_1. One of the following two scenarios must occur:

(a) There exists an unpopular player v in X that beats f_1. In this case, we first introduce v in the seeding, followed by ℓ_2, and then the remaining vertices in X (if any), and then move along the path P_2 following Step 3.

(b) All players in X have an uncertain edge with f_1. We introduce the players in X (in an arbitrary order), and stop if any such player beats f_1. If one of the players beat f_1, then we introduce ℓ_2, and then the remaining vertices in X (if any), and then move along the path P_2 following Step 3. Else all players in X have been introduced and f_1 is the champion, and we introduce ℓ_2. If ℓ_2 wins, we move along the path P_2 following Step 3; else, we repeat the same arguments while comparing f_1 with the next player along the path P_2, and continue the above process on the remaining vertices in this color class, and then on the remaining color classes.

The algorithm obtains VNAR at least $(|n_p| + |n_u| - 1) - (k - 1)$, via a proof identical to Lemma 3.5.1 in Theorem 3.1. □

4 CONCLUSION

In this work, we introduced and studied the VNAR objective for CtC tournaments. VNAR provides a robust guarantee, ensuring a minimum achievable value regardless of the outcomes of uncertain games. Our results include polynomial-time algorithms and nearly matching hardness of approximation results.

Coping with this hardness is a direction for future work. Potential approaches include exponential-time algorithms, exploration of parameters for fixed-parameter tractable algorithms, and algorithms for specific input restrictions. Beyond VNAR, related objective functions in tournaments with probabilistic outcomes include: maximizing expected value and maximizing the likelihood of achieving a certain value. Finally, the scoring function too can be generalized to allow player and round dependencies.

ACKNOWLEDGMENTS

UB and JC acknowledge support from the Department of Atomic Energy, Government of India, under project no. RTI4014. Part of this work was done while JC was a visiting fellow in TIFR.

REFERENCES

[1] Ritxar Arlegi and Dinko Dimitrov. 2020. Fair elimination-type competitions. *Eur. J. Oper. Res.* 287, 2 (2020), 528–535. https://doi.org/10.1016/J.EJOR.2020.03.025

[2] Haris Aziz, Serge Gaspers, Simon Mackenzie, Nicholas Mattei, Paul Stursberg, and Toby Walsh. 2014. Fixing a Balanced Knockout Tournament. In *Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence, 2014*, Carla E. Brodley and Peter Stone (Eds.). AAAI Press, 552–558. https://doi.org/10.1609/AAAI.V28I1.8805

[3] Haris Aziz, Serge Gaspers, Simon Mackenzie, Nicholas Mattei, Paul Stursberg, and Toby Walsh. 2018. Fixing balanced knockout and double elimination tournaments. *Artif. Intell.* 262 (2018), 1–14. https://doi.org/10.1016/J.ARTINT.2018.05.002

[4] Jørgen Bang-Jensen and Gregory Z. Gutin. 2009. *Digraphs - Theory, Algorithms and Applications, Second Edition.* Springer.

[5] Umang Bhaskar, Juhi Chaudhary, and Palash Dey. 2025. Maximizing Value in Challenge the Champ Tournaments. In *Proceedings of the 24th International Conference on Autonomous Agents and Multiagent Systems, AAMAS 2025*, Sanmay Das, Ann Nowé, and Yevgeniy Vorobeychik (Eds.). International Foundation for Autonomous Agents and Multiagent Systems / ACM, 307–315. https://dl.acm.org/doi/10.5555/3709347.3743544

[6] Umang Bhaskar, Juhi Chaudhary, Sushmita Gupta, Pallavi Jain, and Sanjay Seetharaman. 2026. Robust Value Maximization in Challenge the Champ Tournaments with Probabilistic Outcomes. arXiv:2602.14966 [cs.GT] https://arxiv.org/abs/2602.14966

[7] Andreas Björklund, Thore Husfeldt, and Sanjeev Khanna. 2004. Approximating Longest Directed Paths and Cycles. In *Automata, Languages and Programming: 31st International Colloquium, ICALP 2004 (Lecture Notes in Computer Science, Vol. 3142)*, Josep Díaz, Juhani Karhumäki, Arto Lepistö, and Donald Sannella (Eds.). Springer, 222–233. https://doi.org/10.1007/978-3-540-27836-8_21

[8] Andreas Björklund, Thore Husfeldt, and Mikko Koivisto. 2009. Set Partitioning via Inclusion-Exclusion. *SIAM J. Comput.* 39, 2 (2009), 546–563. https://doi.org/10.1137/070683933

[9] Václav Blazej, Sushmita Gupta, M. S. Ramanujan, and Peter Strulo. 2024. On Controlling Knockout Tournaments Without Perfect Information. In *19th International Symposium on Parameterized and Exact Computation, IPEC 2024 (LIPIcs, Vol. 321)*, Édouard Bonnet and Pawel Rzazewski (Eds.). Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 7:1–7:15. https://doi.org/10.4230/LIPICS.IPEC.2024.7

[10] J.M. Buchanan, R.D. Tollison, and G. Tullock. 1980. *Toward a Theory of the Rent-seeking Society.* Number no. 4 in Texas A & M University economics series. Texas A & M University. https://books.google.co.in/books?id=DC-8AAAAIAAJ

[11] Juhi Chaudhary, Hendrik Molter, and Meirav Zehavi. 2024. How to Make Knockout Tournaments More Popular?. In *Thirty-Eighth AAAI Conference on Artificial Intelligence, AAAI 2024*, Michael J. Wooldridge, Jennifer G. Dy, and Sriraam Natarajan (Eds.). AAAI Press, 9582–9589. https://doi.org/10.1609/AAAI.V38I9.28814

[12] Juhi Chaudhary, Hendrik Molter, and Meirav Zehavi. 2025. Adaptive Manipulation for Coalitions in Knockout Tournaments. In *AAAI-25, Sponsored by the Association for the Advancement of Artificial Intelligence*, Toby Walsh, Julie Shah, and Zico Kolter (Eds.). AAAI Press, 13700–13708. https://doi.org/10.1609/AAAI.V39I13.33497

[13] Juhi Chaudhary, Hendrik Molter, and Meirav Zehavi. 2025. Parameterized Analysis of Bribery in Challenge the Champ Tournaments. *J. Artif. Intell. Res.* 83 (2025). https://doi.org/10.1613/JAIR.1.17239

[14] Marek Cygan, Fedor V. Fomin, Lukasz Kowalik, Daniel Lokshtanov, Dániel Marx, Marcin Pilipczuk, Michal Pilipczuk, and Saket Saurabh. 2015. *Parameterized Algorithms.* Springer. https://doi.org/10.1007/978-3-319-21275-3

[15] Dmitry Dagaev and Alex Suzdaltsev. 2018. Competitive intensity and quality maximizing seedings in knock-out tournaments. *J. Comb. Optim.* 35, 1 (2018), 170–188. https://doi.org/10.1007/S10878-017-0164-7

[16] Klim Efremenko, Hendrik Molter, and Meirav Zehavi. 2025. Tournament Robustness via Redundancy. In *Proceedings of the 26th ACM Conference on Economics and Computation, EC 2025*, Itai Ashlagi and Aaron Roth (Eds.). ACM, 66–85. https://doi.org/10.1145/3736252.3742494

[17] Sushmita Gupta, Ramanujan Sridharan, and Peter Strulo. 2024. An Exercise in Tournament Design: When Some Matches Must Be Scheduled. In *Thirty-Eighth AAAI Conference on Artificial Intelligence, AAAI 2024*, Michael J. Wooldridge, Jennifer G. Dy, and Sriraam Natarajan (Eds.). AAAI Press, 9749–9756. https://doi.org/10.1609/AAAI.V38I9.28833

[18] Chính T. Hoàng, Marcin Kaminski, Vadim V. Lozin, Joe Sawada, and Xiao Shu. 2010. Deciding k-Colorability of P_5-Free Graphs in Polynomial Time. *Algorithmica* 57, 1 (2010), 74–81. https://doi.org/10.1007/S00453-008-9197-8

[19] Russell Impagliazzo and Ramamohan Paturi. 2001. On the Complexity of k-SAT. *J. Comput. Syst. Sci.* 62, 2 (2001), 367–375. https://doi.org/10.1006/JCSS.2000.1727

[20] Sanjeev Khanna, Nathan Linial, and Shmuel Safra. 2000. On the Hardness of Approximating the Chromatic Number. *Comb.* 20, 3 (2000), 393–415. https://doi.org/10.1007/s004930070013

[21] Jean-François Laslier. 1997. *Tournament solutions and majority voting.* Vol. 7. Springer.

[22] Nicholas Mattei, Judy Goldsmith, Andrew Klapper, and Martin Mundhenk. 2015. On the complexity of bribery and manipulation in tournaments with uncertain information. *J. Appl. Log.* 13, 4 (2015), 557–581. https://doi.org/10.1016/J.JAL.2015.03.004

[23] Sherwin Rosen. 1986. Prizes and Incentives in Elimination Tournaments. *The American Economic Review* (1986), 701–715.

[24] Tyrel Russell and Toby Walsh. 2009. Manipulating Tournaments in Cup and Round Robin Competitions. In *Algorithmic Decision Theory, First International Conference, ADT 2009. Proceedings (Lecture Notes in Computer Science, Vol. 5783)*, Francesca Rossi and Alexis Tsoukiàs (Eds.). Springer, 26–37. https://doi.org/10.1007/978-3-642-04428-1_3

[25] Pascal Sauer, Ágnes Cseh, and Pascal Lenzner. 2024. Improving ranking quality and fairness in Swiss-system chess tournaments. *Journal of Quantitative Analysis in Sports* 20, 2 (2024), 127–146. https://doi.org/doi:10.1515/jqas-2022-0090

[26] Isabelle Stanton and Virginia Vassilevska Williams. 2013. The structure, efficacy, and manipulation of double-elimination tournaments. *Journal of Quantitative Analysis in Sports* 9, 4 (2013), 319–335. https://doi.org/doi:10.1515/jqas-2012-0055

[27] Warut Suksompong. 2016. Scheduling asynchronous round-robin tournaments. *Oper. Res. Lett.* 44, 1 (2016), 96–100. https://doi.org/10.1016/J.ORL.2015.12.008

[28] Warut Suksompong. 2021. Tournaments in Computational Social Choice: Recent Developments. In *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI 2021*, Zhi-Hua Zhou (Ed.). ijcai.org, 4611–4618. https://doi.org/10.24963/IJCAI.2021/626

[29] Balázs R. Sziklai, Péter Biró, and László Csató. 2022. The efficacy of tournament designs. *Comput. Oper. Res.* 144 (2022), 105821. https://doi.org/10.1016/J.COR.2022.105821

[30] Virginia Vassilevska Williams. 2016. Knockout Tournaments. In *Handbook of Computational Social Choice*, Felix Brandt, Vincent Conitzer, Ulle Endriss, Jérôme Lang, and Ariel D. Procaccia (Eds.). Cambridge University Press, 453–474. https://doi.org/10.1017/CBO9781107446984.020

[31] Thuc Vu, Alon Altman, and Yoav Shoham. 2009. On the complexity of schedule control problems for knockout tournaments. In *8th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2009)*, Carles Sierra, Cristiano Castelfranchi, Keith S. Decker, and Jaime Simão Sichman (Eds.). IFAAMAS, 225–232. https://dl.acm.org/citation.cfm?id=1558044

[32] Thuc Vu and Yoav Shoham. 2011. Fair Seeding in Knockout Tournaments. *ACM Trans. Intell. Syst. Technol.* 3, 1 (2011), 9:1–9:17. https://doi.org/10.1145/2036264.2036273

[33] Yongjie Yang and Dinko Dimitrov. 2021. Weak transitivity and agenda control for extended stepladder tournaments. *Economic Theory Bulletin* 9, 1 (2021), 27–37. https://doi.org/10.1007/s40505-020-00190-9