An Empirical Study of Seeding Manipulations and Their Prevention

Tyrel Russell and **Peter van Beek**
Cheriton School of Computer Science
University of Waterloo
Waterloo, Ontario, Canada N2L 3G1

Abstract

It is well known that cheating occurs in sports. In cup competitions, a common type of sports competition, one method of cheating is in manipulating the seeding to unfairly advantage a particular team. Previous empirical and theoretical studies of seeding manipulation have focused on competitions with unrestricted seeding. However, real cup competitions often place restrictions on seedings to ensure fairness, wide geographic interest, and so on. In this paper, we perform an extensive empirical study of seeding manipulation under comprehensive and realistic sets of restrictions. A generalized random model of competition problems is proposed. This model creates a realistic range of problem instances that are used to identify the sets of seeding restrictions that are hard to manipulate in practice. We end with a discussion of the implications of this work and recommendations for organizing competitions so as to prevent or reduce the opportunities for manipulating the seeding.

1 Introduction

It is well known that cheating occurs in sports. One method of cheating is where athletes or teams collude to manipulate the outcome of a sports competition by throwing games. Another method of cheating that arises in cup competitions is in manipulating the seeding or schedule—the initial matchups of teams—such that a particular team wins or has an increased chance of winning. Cup competitions are commonly used in both sports and elections as a social choice mechanism and having a balanced cup is a desirable property as no team gains an advantage by being placed higher in the tree.

The problem of seeding manipulation, or schedule control, has been well studied in the social choice literature and both a probabilistic model of the competition and a deterministic (win/loss) model have been considered [Bartholdi et al., 1992; Lang et al., 2007; Hazon et al., 2008; Vu et al., 2009; Williams, 2010]. For the probabilistic model, Hazon et al. [2008] show that the problem of determining the seeding which maximizes the probability of a team winning in a balanced cup is NP-Hard. Vu et al. [2009] provide an alternative proof of the above result and also consider additional variants where the probability of a team beating another team is restricted to a specific set of values such as the set $\{0, 0.5, 1\}$. For the deterministic model, Williams [2010] showed that certain sub-classes of the problem can be solved in polynomial time. The computational complexity of the general problem remains open for balanced cups, although it has been the subject of much investigation. The worst-case complexity results can be seen as a barrier to manipulation (cf. [Faliszewski et al., 2010]). However, Hazon et al. [2008] show that simple heuristics can be effective in approximately solving instances of the probabilistic model. Thus, seeding is open to manipulation by quite straightforward techniques.

However, previous studies of seeding manipulation have focused on competitions with unrestricted seeding. Real cup competitions often place restrictions on seedings to ensure fairness, wide geographic interest, and so on. As well, previous studies have not examined the incentive to manipulate. In this paper, we perform an extensive empirical study of seeding manipulation under comprehensive and realistic sets of restrictions. For our empirical study, we propose a generalized random model of cup competition problems. This model creates a more realistic range of problem instances than in previous work (where teams do not differ in their intrinsic quality or rank, and the seedings are unrestricted). The random model is used to identify the sets of seeding restrictions that are hard to manipulate in practice. We show that for both the probabilistic model and the deterministic (win/loss) model, there are sets of real-world restrictions where one can often effectively manipulate the seeding in practice. In the case of the deterministic model, we show that an optimal approach based on constraint programming can quickly solve almost all instances. In the case of the probabilistic model, we show that a previously proposed heuristic adapted to the sets of restrictions can effectively solve instances.

The goal of our work is to study which, if any, of these seeding restrictions are hard to manipulate in practice and which, if any, of these seeding restrictions reduce the incentives to manipulate the seeding by reducing the effectiveness of manipulation. We find that some sets of restrictions are an effective method for reducing the incentive for seeding manipulation, especially for weakly ranked teams. Based on our study, we provide recommendations for organizing competitions so as to prevent or reduce the opportunities for manipulating the seeding.

2 Background

One of the most common ways of organizing a competition among a set of teams is a cup. In a *cup*, the final winner is determined by a tree-like structure called a *competition tree*. Let m denote the number of teams in the cup, where the names of the teams are $t_1, t_2, \ldots, t_m$. The m leaves of the tree are labeled with the names of the teams, called the *seeding* of the competition. A seeding on m teams can be described by a permutation $(\pi_1, \pi_2, \ldots, \pi_m)$ of the set $\{1, 2, \ldots, m\}$, where the team t_{π_j} is the label for the j^{th} leaf node in the competition tree. Each internal node of the tree represents a match (or game) between its two child nodes. A *round* of a cup is defined to be all of the matches that occur at an equal height k, $1 \leq k \leq \log_2 m$, from the leaves of the competition tree. The winner of a match advances to the next round and the loser of the match is eliminated from the competition. The team that advances to the root of the tree is the winner of the competition. In this paper, perfectly balanced cups are examined and it is assumed that the number of teams is a power of 2. Balanced cups are the most widely used in practice as they are inherently the most fair.

A seeding can be generated by a random draw but most often a person (or committee), called the *scheduler*, determines the seeding. As such, seeding is open to manipulation. A *seeding manipulation strategy* is any deliberate placement of teams in the seeding in an effort to cause or increase the likelihood that a desired team, t_w, wins the competition.

As in previous sports and voting literature, we assume that there is available a probabilistic model of the competition in the form of an $m \times m$ matrix P, where entry $P_{i,j}$ denotes the probability that team t_i wins a match against team t_j. The probabilities could be based on past matches or on expert opinions. If the probabilities are restricted to be 0 and 1, the model is deterministic and can be represented as a directed graph where the nodes are the teams and there is an edge from t_i to t_j if t_i wins a match against t_j with certainty. Such a graph is often referred to as a *tournament* graph.

3 Restrictions on Seedings

Previous studies have considered unrestricted seeding, where any placement of teams is permitted. However, unrestricted seedings can be undesirable in practice and restrictions are often added to ensure that the seeding conforms to the desires of the organizers such as, for example, that top teams do not face each other immediately.

We have identified four different restrictions that can be placed on the seeding process. These restrictions broadly cover known restrictions that occur in practice.

Team ranking restriction. Let $rank(t)$ be a function that gives the ranking of a team t, where smaller values mean higher rank or quality. A *team ranking restriction* requires that the ranking function is known and can be objectively determined. For example, the ranking function could be determined by the teams' win-loss records over a season.

Team arrangement restriction. A primary purpose of a seeding is to ensure that two highly ranked teams do not meet each other early in the competition. For this purpose, a commonly used arrangement of the teams is as follows. For simplicity of notation, assume that $rank(1) = 1$, $rank(2) = 2$, and so on. For $m = 2$ teams, the seeding is $(1, 2)$; i.e., the top ranked team plays the team ranked second. Suppose $(\pi_1, \pi_2, \ldots, \pi_{2^{r-1}})$ is the seeding for 2^{r-1} teams. Then the seeding for $m = 2^r$ teams is $(\pi_1, m+1-\pi_1, \pi_2, m+1-\pi_2, \ldots, \pi_{2^{r-1}} m+1-\pi_{2^{r-1}})$. For example, the seedings for four and eight teams are $(1, 4, 2, 3)$ and $(1, 8, 4, 5, 2, 7, 3, 6)$, respectively. A *team arrangement restriction* requires that the method for arranging teams is known and can be objectively determined. In our experiments we used the above arrangement.

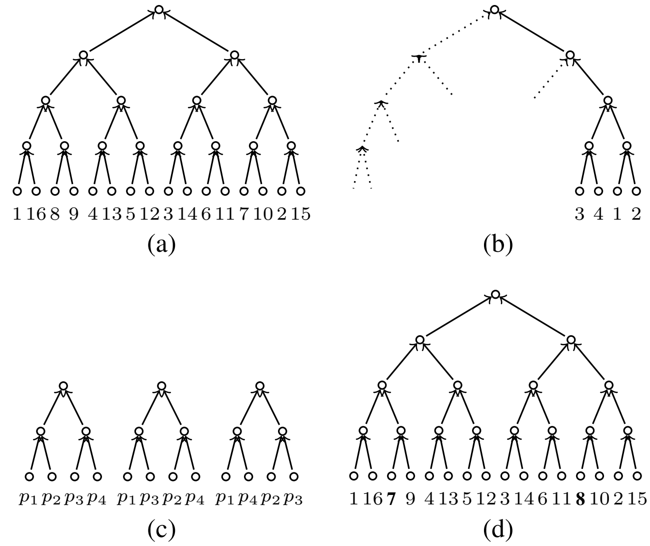

Figure 1: **(a)** The arrangement of teams by rank used by the NCAA Division I Basketball Championship. The numbers below the leaf nodes represent the rank of the team at that leaf. **(b)** A seeding can be manipulated so that all of the top teams play each other in the first two rounds. **(c)** The possible ways that the winners of the four pools of the NCAA could be arranged. **(d)** Changing the rank of a team from eight to seven can completely change their opponents.

Pooling restriction. Let $m = 2^r$, for some r. Pooling partitions the m teams into 2^{r-k} *pools* each of size 2^k, for some k. The teams in a pool play each other before playing other teams; i.e., the winner of a pool goes on to play other teams outside of the pool. A *pooling restriction* requires that the method for partitioning the teams into pools is known and can be objectively determined. In our experiments, we used pools of size 16 and followed the NCAA method where each team is assigned a rank from 1 to 16, and one team of each rank belongs to a pool.

Pool arrangement restriction. A *pool arrangement restriction* requires that the method for arranging the order of the pools—i.e., the order that the winners of the pools play each other—is known and can be objectively determined. Let $P_1, \ldots, P_{2^{r-k}}$ be the pools. In our experiments we simply ordered the pools in ascending order so that the winner of pool P_1 played the winner of pool P_2, and so on.

In each case, the absence of a restriction means that this aspect of the seeding (team ranking, team arrangement, pooling, pooling arrangement) is under the control of the scheduler and is thus open to manipulation. In the case of team ranking, in our experiments we restricted the manipulation of a team's rank to be within one of its true rank.

Example 1. *The NCAA Division I Basketball Championship, commonly called March Madness, is held annually in March and April [NCAA, 2009]. The championship uses a cup structure where 64 teams are seeded in a balanced cup tree. The teams are separated into pools of sixteen teams. Each team is assigned a rank from 1 to 16 and only one team of each rank may belong to a pool. The cup has four pools of teams but the makeup of the pools is made by the scheduler. Figure 1a shows how teams are arranged within a pool. If this arrangement was not fixed, it would be possible to guarantee that three of the top four teams in the pool would be eliminated after the second round (see Figure 1b). March Madness championships do not have a pool arrangement restriction. As such, the scheduler is able to arrange the final two rounds as they see fit. The possible arrangements of the four pools of the NCAA Championship are shown in Figure 1c. There is no restriction on team ranking—it is decided by the scheduler—and the scheduler can use this to change the seeding [Coleman et al., 2010]. The imprecise nature of the rank generation allows for weaker teams to be boosted in rank and top teams to be diminished. While it is unlikely that large changes would be made to the rank, it is possible to modify the rank of teams by only a difference of one and still gain a large advantage (see Figure 1d).*

Given the different restrictions, it is possible to construct families of restrictions and classify competitions based on the restrictions present. As a naming convention, a family of restrictions is considered as a four tuple where each tuple value can be either 0 or 1 depending on the presence of the restriction. An example tuple for the family without any restrictions would be 0000. The tuple placements represent, from left to right, the pooling restriction, the team arrangement restriction, the team ranking restriction and the pool arrangement restriction. Families that are equivalent regardless of a restriction being present are merged and the corresponding tuple value is replaced with a don't care value or X. The complete set of families along with the restriction relationship are shown in Figure 2. An arc from one family to another represents that a restriction was added to the source family to produce the new family. The *solution set* is the set of all possible seedings which conform to a given family of restrictions. Two families are equivalent if the solution set is identical. For more details and proofs, see [Russell, 2010, Chapter 4.4].

Each restriction of the seeding implies that the choice is fixed without input from the scheduler. While it is possible to fully restrict a schedule, the result may be undesirable. For example, if geographic constraints are used to ensure unmanipulable pooling then top teams may be paired in the early rounds since they happen to be geographically close. We next describe each family of restrictions and, if possible, a real world example which uses these restrictions is described.

The most restricted family is 1111, as all of the restrictions are enforced. Competitions with these restrictions arise in the playoffs of many North American sports, including the

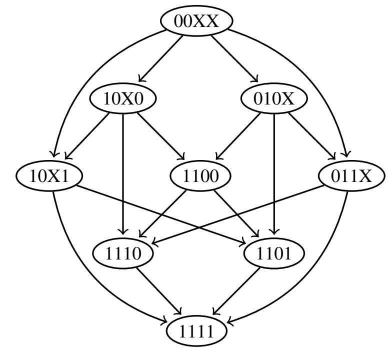

Figure 2: Families of seeding restrictions. Each label is a four tuple where 1 means fixed, 0 means unfixed and X is a don't care value. Don't care values are used to denote when the models are solution equivalent. The four tuple values represent are, from left to right, pooling, team arrangement, team ranking and pool arrangement.

NFL, NBA, and the NHL (where the pools are restricted by geography, for example). The next most restricted family is 1101 where only the team ranking restriction is not enforced. An example of this would be professional tennis where the ranking of players do not necessarily correspond to the World Ranking of the same players. The 1100 family of restrictions requires that the team arrangement and pooling of teams is fixed but the ranking of teams can be modified and the arrangement of pools is unfixed. Since professional tennis competitions only have a single pool of teams, this family of restrictions could be applied to tennis as well.

The 1110 family of restrictions requires that the pooling, team arrangement and ranking are all immutable but the pool arrangement is not restricted. The $011X$ family of restrictions allows the pools to be unfixed but the team arrangement and ranking are fixed. The $10X1$ family of restrictions requires that pooling and the arrangement of the pools are fixed while leaving the arrangement of the teams within the pools susceptible to manipulation. The $10X0$ family of restrictions requires a specific pooling but offers no other restriction on the seeding. It is not known if there are any competitions which use either the 1110, $011X$, $10X1$ or $10X0$ families of restrictions when generating seedings. The $010X$ family of restrictions requires a specific arrangement of teams but requires no other restrictions. This model is used in scheduling the NCAA Division I Basketball Championship [NCAA, 2009] and World Cup soccer. The most unconstrained family of restrictions is the $00XX$ family of restrictions. This is the model used in previous work (e.g., [Lang et al., 2007; Hazon et al., 2008; Vu et al., 2009; Williams, 2010]). This type of completely unconstrained seeding is often used by local and amateur sports.

4 Seeding Manipulation Strategies

In this section, we present the techniques we used in our experiments for finding seeding manipulations.

Deterministic model. The computational complexity of finding a seeding manipulation strategy for the deterministic

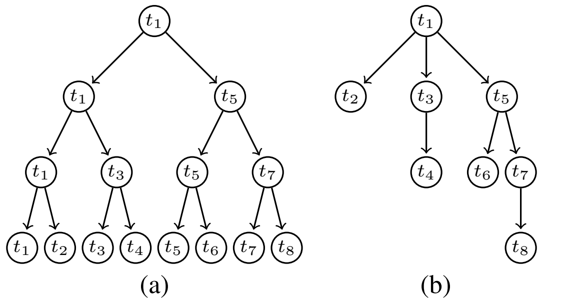

Figure 3: **(a)** A cup competition denoting the winner in each node. **(b)** The corresponding binomial spanning tree which represents the result of the competition.

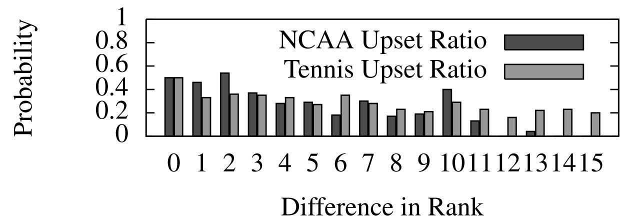

Figure 4: The probability that a team whose rank is i positions lower than another team would upset that team, as estimated from all games in the 25 tournaments from 1985–2009 for the NCAA and the grand slam tennis events from 2000–2010.

model is an open question, and a polynomial algorithm has yet to be discovered. The alternative to giving up on these problems is to apply the machinery used to solve NP-Hard problems and look at the practical complexity of these problems. This constitutes neither a proof of efficiency or of hardness but rather shows, as is done in graph isomorphism [Sorlin and Solnon, 2004], that these problems can be solved even if their complexity is indeterminate.

The basic problem is determining if a given team t_w is the winner under any seeding of the tournament given the tournament graph. Lang et al. [Lang et al., 2007] showed that the least constrained problem is solved by finding a binomial spanning tree within the tournament graph structure. The winner of the cup represents the root of the tree. The children are the teams winner defeated in each round from left to right. This notion is then applied recursively to each child. If there exists an assignment of the teams so that the edges in the binomial spanning tree are also in the tournament graph then the team at the root is a winner.

Example 2. *Figure 3a shows a cup competition with the expected winner of each match according to the tournament graph labeled in each node. The winner of the cup is t_1 and t_1 becomes the root of the binomial tree. Since t_1 defeats t_2, t_3 and t_5 in each successive round, t_2, t_3 and t_5 are made the children of t_1. t_2 does not win a single game so they are a leaf node. t_3 defeats t_4 and t_5 defeats t_6 and t_7. Finally, t_4 and t_6 are leaf nodes and t_7 defeats t_8, which is a leaf node. The completely binomial tree is shown in Figure 3b.*

To tackle these problems, we adapted a constraint programming framework for solving sub-graph isomorphism [Zampelli et al., 2010]. The constraint approach for the simple model is described first and then additional constraints are added to solve the other models (for more details on the constraint programming approach, see [Russell, 2010, Chapter 4.4]). There are m variables labeled $v_1, \dots v_m$, with the domain of each variable $D(v_i) = [1 \dots m]$. Since each variable is a node in a spanning tree of the tournament graph, an AllDifferent constraint is added to ensure that no node is used more than once, $alldifferent(v_1, \dots, v_m)$. Edge constraints are added to the variables of the graph to ensure that if there is an assignment of v_i and v_j to the variables in the spanning tree which share an edge then $(v_i, v_j) \in E$. Lastly, the constraint that $v_1 = t_w$ ensures that t_w wins the tournament, if possible.

This basic constraint-based approach can be extended with additional constraints to reflect the additional restrictions. The most useful is a global constraint proposed by Larrosa and Valiente [2002], called nRF+, that enforces slightly stronger consistency on the children of nodes within the tree. This ensures that for each value of a parent node there are enough distinct values in the domains of the child nodes to satisfy each child node. For each additional restriction, the basic edge constraint is modified to allow only those edges that are edges in the graph and match the appropriate criteria. For pooling, this ensures that the lower edges on the graph match only teams from the same pool and, for pool arrangement, that the upper edges of the graph meet the correct arrangement. For team arrangement, the teams must meet the criteria for the arrangement. Modifying ranking associates a new rank variable with each edge in the binomial tree and restricts the rank of a given team based on the team arrangement.

Representing the seeding as binomial tree over a simple list representation, removes 2^n rotational symmetries, where n is the height of the cup tree. This representation reflects who plays whom at each level and allows for any possible placement in the cup tree as long as the restrictions are respected.

Probabilistic model. The computational complexity of finding a seeding manipulation strategy for the probabilistic model is known to be NP-Hard for balanced cups [Hazon et al., 2008; Vu et al., 2009]. Hazon et al. [2008] propose a number of heuristics for solving this problem and show that for small problems the heuristics are close to optimal. Their results showed that a heuristic called BestWin was a good heuristic for solving these problems. However, Hazon et al. [2008] assume there is no restriction on the seeding.

The BestWin heuristic greedily assigns the teams that t_w, the team that the scheduler wishes to unfairly advantage, is most likely to defeat in each round. In other words, trying to generate likely winners of subtrees that t_w will defeat. This procedure is then applied recursively to every other teams unless that team is a leaf and would be expected to win no games. We modified the BestWin heuristic to account for the restrictions applied to the model. When greedily assigning an opponent during the procedure, only those teams which do not violate the restrictions are compared and selected. When ranking is unfixed, each team attempts to obtain the lowest rank possible as this increases the likelihood that they will face weaker teams.

5 Experimental Results

While there is some evidence to suggest there is bias in seeding real world cup competitions [Coleman et al., 2010], there is no definitive data set to test the efficiency of the constraint programming approach described above. Therefore to test the constraint programming approach under the various restrictions, a set of realistic test benchmarks are generated. The first step is to generate a realistic tournament graph.

To generate real world probability matrices and tournament graphs, real world data was mined to determine *upset ratios*: probability distributions of how often better teams lost to worse teams given their difference in ranks. The NCAA Division I Basketball competition brackets from 1985 to 2009 and grand slam tennis events from 2000 to 2010 were mined for this information. Upset ratios were constructed for each sport from the data to give the distribution of upsets given the difference in rank. In the case of tennis, ranks were pooled into groups of eight to remove some of the fine grained granularity of ranking. This is also the case in the NCAA data as there are multiple teams of each rank in each competition. These ratios can be seen in Figure 4.

From the upset ratios, Gaussian distributions for each difference in rank were created with the means taken from the upset ratios and the variances from the errors in the ratios. By sampling the Gaussians, realistic probability matrices were generated for problems of size 16, 32, 64, 128, 256. Tournament graphs are generated in a similar model by sampling the distribution for each game to determine whether the lower ranked team defeated the higher ranked team. The largest cup balanced competitions found in practice are those used by professional tennis where there are 128 participants. For each probability matrix, every team is tested to determine whether the probability of the team winning could be increased from expected and, for each tournament graph, every team is tested to determine whether there exists a seeding which would ensure their victory. All of the families of restrictions were tested except the 1110 family, since it is a strictly easier variant of the 0000 family, and the 1111 family, which can be solved in linear time using simple checks. For most families of restrictions the random instances generated were quickly solved (within a few seconds). Only in a single family of restrictions, the $010X$ family, under the deterministic model were there instances which timed out and there were relatively few of those.

Tables 1 and 3 show the percentage of teams that could be manipulated for a given rank and that even relatively small restrictions can drastically reduce the number of teams which could be made winners via seeding manipulation. As expected, it is more likely that the scheduler could generate a seeding manipulation strategy for strong teams. The results show that, at least on these instances, seedings can be easily generated to manipulate the results of the tournament. Therefore, the cup competitions are open for potential abuse by the scheduler of the competition from a practical point of view.

Table 1: *Deterministic (win/loss) model (NCAA).* The effect of seeding restrictions on the percentage of teams that could be made the winner via manipulation of the seeding, for various tournament sizes and rank of team i = 1, 4, 8, 12, 16. Each entry is the average of 100 instances.

Model	Size	1	4	8	12	16
Unrestricted (00XX)	16	97	95	95	82	32
	32	100	100	100	100	87
	64	100	100	100	100	100
	128	100	100	100	100	100
	256	100	100	100	100	100
	Size	1	4	8	12	16
Pooled (10X0)	16	97	95	95	82	32
	32	97	96	93	85	37
	64	99	99	98	88	40
	128	99	97	96	85	44
	256	98	97	96	87	39
	Size	1	4	8	12	16
NCAA (010X)	16	88	68	20	11	0
	32	100	100	93	90	6
	64	100	100	100	100	16
	128	100	100	100	100	32−33†
	256	100	100	100	100	0−51†
	Size	1	4	8	12	16
Tennis (1101)	16	88	68	20	11	0
	32	83	65	20	7	0
	64	85	64	17	7	0
	128	86	61	16	6	0
	256	84	58	17	6	0

† Percentage is given as a range as not all instances could be solved to optimally due to resource restrictions.

Tables 2 and 4 show the difference between the expected probability of a team of rank i winning and the average probability of a team of the same rank winning under heuristic manipulation. A larger difference implies there is greater incentive for manipulation. The results show that for unconstrained problems there is a significant incentive for cheating, especially if they desire top ranked teams to win. However, as restrictions are added this incentive is reduced. This is particularly true for those models with a team arrangement restriction. For the 1101 family of restrictions, there is virtually no incentive for changing the seeding to improve the chances of weak teams and little incentive for even the strongest teams.

6 Discussion and Conclusion

Cup competitions are prevalent in sports. Given our experimental results, it is possible to make recommendations about the method for organizing such a competition in order to prevent or reduce the possibility that a cup competition can be manipulated in favor of a particular team by manipulating the seeding. The basic idea is that certain combinations of restrictions make it either impossible or more difficult to manipulate the seeding.

From the deterministic results in Tables 1 and 3, the opportunity for a scheduler to manipulate the competition increases

Table 2: *Probabilistic model (NCAA).* The effect of seeding restrictions on the increase in the expected probability ($\times$ 100) of winning the competition due to seeding manipulation over a fair seeding, for various tournament sizes and rank of team i = 1, 4, 8, 12, 16. Each entry is the average of 100 instances.

Model	Size	1	4	8	12	16
	16	11.9	9.2	3.7	1.5	0.2
	32	11.7	8.4	2.5	0.7	0.0
Unrestricted	64	9.5	5.8	1.6	0.3	0.0
(00XX)	128	7.4	3.9	0.8	0.1	0.0
	256	5.9	2.5	0.4	0.0	0.0
	Size	1	4	8	12	16
	16	12.0	9.1	3.7	1.5	0.2
	32	7.4	4.8	1.2	0.4	0.0
Pooled	64	4.2	2.4	0.4	0.1	0.0
(10X0)	128	2.4	1.1	0.1	0.0	0.0
	256	1.3	0.5	0.1	0.0	0.0
	Size	1	4	8	12	16
	16	1.3	3.2	0.4	0.2	0.0
	32	3.5	1.7	0.4	0.0	0.0
NCAA	64	2.1	1.0	0.2	0.0	0.0
(010X)	128	1.6	0.6	0.1	0.0	0.0
	256	1.1	0.4	0.0	0.0	0.0
	Size	1	4	8	12	16
	16	1.7	2.9	0.4	0.2	0.0
	32	1.1	1.4	0.1	0.0	0.0
Tennis	64	0.6	0.6	0.0	0.0	0.0
(1101)	128	0.3	0.3	0.0	0.0	0.0
	256	0.2	0.1	0.0	0.0	0.0

Table 3: *Deterministic (win/loss) model (Tennis).* The effect of seeding restrictions on the percentage of teams that could be made the winner via manipulation of the seeding, for various tournament sizes and rank of team i = 1, 4, 8, 12, 16. Each entry is the average of 100 instances.

Model	Size	1	4	8	12	16
	16	100	99	99	93	54
	32	100	100	100	100	95
Unres.	64	100	100	100	100	100
(00XX)	128	100	100	100	100	100
	256	100	100	100	100	100
	Size	1	4	8	12	16
	16	100	99	99	93	54
	32	99	99	99	91	55
Pooled	64	99	99	98	90	60
(10X0)	128	99	99	98	89	60
	256	99	99	98	90	58
	Size	1	4	8	12	16
	16	89	87	55	23	11
	32	100	100	100	98	72
NCAA	64	100	100	100	99	94
(010X)	128	100	100	100	99−100†	81−100†
	256	100	100	100	99−100†	97−100†
	Size	1	4	8	12	16
	16	89	87	55	23	11
	32	89	80	53	26	9
Tennis	64	90	75	47	20	8
(1101)	128	90	78	46	20	8
	256	91	75	45	20	8

† Percentage is given as a range as not all instances could be solved to optimally due to resource restrictions.

as fewer restrictions are applied. Therefore, if a scheduler is assumed to have perfect or near perfect information about the teams in the competition, the only method for increasing the difficulty of manipulation is to increase the number of restrictions on the seeding. NHL Hockey, NBA Basketball and NFL Football, for example, do not use a scheduler as the seeding is fully determined by the restrictions.

However, it is often the case that the scheduler will not have perfect knowledge about the outcome of each game and must rely on probabilistic data to construct a manipulated seeding. Tables 2 and 4 show that there is variability in the incentive for manipulation by the scheduler between various families of restrictions when using a heuristic manipulation scheme. Unlike the deterministic case, the argument made for restrictions in the probabilistic case is that certain restrictions reduce the incentive for cheating. Pooling restriction by itself seems to have little effect on the ability of the scheduler to generate a seeding which improves the probability of t_w winning. In contrast, for the heuristic tested, team arrangement has a significant effect even when the rank is allowed to vary by one from the actual ranking of the team.

Given these results, it seems prudent for the organizers of the competitions to enforce team arrangement as this can reduce the incentive for manipulating the seeding. Pooling restrictions can be helpful when used in conjunction with team arrangement but provides little protection when used without additional restrictions. As in the deterministic cases, further restrictions do decrease the probability that a team can manipulate the competition but this is sometimes undesirable. For example, fixed pooling in a competition like the NCAA Division I Championship may cause two top teams to be paired before the lucrative Final Four portion of the competition. A seeding that produces this result would be undesirable even though it could reduce the incentive to manipulate.

As fully restricting the seeding is sometimes undesirable or difficult, it is also necessary to detect cheating when it occurs. Random drawing could be used but suffers from two issues: the seeding may contain matchups that are undesirable and draw rigging is a common issue because the randomization process can be corrupted. In tennis, for example, only the top 32 players are drawn in public to ensure certain properties to the seeding but the other 96 players are drawn by a computer program, which could be designed to implement a manipulated seeding as described in this work. Coleman et al. [2010] examined the bias of selection in the NCAA Division I Basketball championship and concluded that the bias was due to in part the discrepancy between the perceived ranking and actual ranking. There are many different ranking measures that could be used to construct a likely ranking of teams and compared against the results (e.g., see [Cassady et al., 2005] and references therein). One important thing to note here is that the quality of the detection is highly dependent on the qual-

Table 4: *Probabilistic model (Tennis).* The effect of seeding restrictions on the increase in the expected probability (× 100) of winning the competition due to seeding manipulation over a fair seeding, for various tournament sizes and rank of team i = 1, 4, 8, 12, 16. Each entry is the average of 100 instances.

Model	Size	1	4	8	12	16
	16	4.1	3.7	1.7	1.2	0.1
	32	3.8	2.7	0.9	0.4	0.1
Unrestricted	64	3.1	1.7	0.4	0.1	0.1
(00XX)	128	2.6	1.1	0.2	0.1	0.0
	256	2.1	0.6	0.1	0.0	0.0
	Size	1	4	8	12	16
	16	4.1	3.7	1.7	1.2	0.1
	32	2.6	1.8	0.7	0.3	0.0
Pooled	64	1.7	0.8	0.2	0.1	0.0
(10X0)	128	1.0	0.3	0.1	0.0	0.0
	256	0.6	0.1	0.1	0.0	0.0
	Size	1	4	8	12	16
	16	1.5	0.8	0.3	0.2	0.1
	32	1.3	0.9	0.3	0.1	0.0
NCAA	64	1.3	0.6	0.1	0.0	0.0
(010X)	128	1.1	0.4	0.1	0.0	0.0
	256	0.8	0.2	0.1	0.0	0.0
	Size	1	4	8	12	16
	16	1.5	0.8	0.3	0.2	0.1
	32	0.9	0.4	0.1	0.1	0.0
Tennis	64	0.6	0.2	0.0	0.0	0.0
(1101)	128	0.3	0.1	0.0	0.0	0.0
	256	0.2	0.0	0.0	0.0	0.0

ity of the assumed ranking and, therefore, is only as good as the ranking model used. With pooling, pool arrangement and team pairing, it is more difficult to analyze this type of manipulation. Since a fair seeding of a unfixed restriction should be random, a deviation from random on a set of seedings would have to be shown in order to conclude that cheating was occurring. For example, if a particular team or group of teams was always favorably pooled away from their strongest opponents and this occurred over a number of different seedings then there would be cause for further investigation.

In conclusion, seeding manipulation is a known issue in sports competitions [Coleman et al., 2010] and previous work focuses on the complexity of manipulation [Lang et al., 2007; Hazon et al., 2008; Vu et al., 2009] and on the existence of effective techniques for manipulation [Hazon et al., 2008]. In contrast, this work looks at examining whether seeding manipulation is possible under realistic restrictions and determining the incentive for manipulation under the same set of restrictions. Against a scheduler with perfect knowledge, the only effective method for reducing the possibility of manipulation is to introduce additional restrictions. However, in the probabilistic case, experimental data suggests that organizers combating seeding manipulation should restrict the seeding with a specific team arrangement and, if possible, other restrictions. This is shown to be an effective method for reducing the incentive for seeding manipulation, especially for weakly ranked teams. As a completely restricted seeding is undesirable in some cases, detection methods such as the one described by Coleman et al. [2010] should be used in conjunction with the enforced restrictions. The combined use of restrictions and detection provides the organizers of competitions with tools to combat seeding manipulation by reducing the incentive for manipulation.

References

[Bartholdi et al., 1992] Bartholdi, J.; Tovey, C.; and Trick, M. 1992. How hard it is to control an election? *Math. and Computer Modeling* 16:27–40.

[Cassady et al., 2005] Cassady, C. R.; Maillart, L. M.; and Salman, S. 2005. Ranking sports teams: A customizable quadratic assignment approach. *Interfaces* 35:497–510.

[Coleman et al., 2010] Coleman, B. J.; DuMond, J. M.; and Lynch, A. K. 2010. Evidence of bias in NCAA tournament selection and seeding. *Managerial and Decision Economics*. Online March 2010.

[Faliszewski et al., 2010] Faliszewski, P.; Hemaspaandra, E.; Hemaspaandra, L. A. 2010. Using complexity to protect elections. *CACM* 53(11):74–82.

[Hazon et al., 2008] Hazon, N.; Dunne, P. E.; Kraus, S.; and Wooldridge, M. 2008. How to rig elections and competitions. In *Proc. of COMSOC*, 301–312.

[Lang et al., 2007] Lang, J.; Pini, M. S.; Rossi, F.; Venable, K. B.; and Walsh, T. 2007. Winner determination in sequential majority voting. In *Proc. of IJCAI*, 1372–1377.

[Larrosa and Valiente, 2002] Larrosa, J., and Valiente, G. 2002. Constraint satisfaction algorithms for graph pattern matching. *Math. Structures in Comp. Sci.* 12:403–422.

[NCAA, 2009] NCAA. 2009. NCAA division I men's basketball championship: Principles and procedures for establishing the bracket. http://www.ncaa.com/graphics/champpage/Bracket_Prin-Proc_2009-10_07.07.09.pdf.

[Russell, 2010] Russell, T. 2010. A Computational Study of Problems in Sports. PhD Dissertation. University of Waterloo.

[Sorlin and Solnon, 2004] Sorlin, S., and Solnon, C. 2004. A global constraint for graph isomorphism problems. In *Proc. of CP-AI-OR*, 287–302.

[Vu et al., 2009] Vu, T.; Altman, A.; and Shoham, Y. 2009. On the complexity of schedule control problems for knockout tournaments. In *Proc. of AAMAS*, 225–232.

[Walsh, 2010] Walsh, T. 2010. An empirical study of the manipulability of single transferable voting. In *Proc. of ECAI*, 257–262.

[Williams, 2010] Williams, V. V. 2010. Fixing a tournament. In *Proc. of AAAI*, 895–900.

[Zampelli et al., 2010] Zampelli, S.; Deville, Y.; and Solnon, C. 2010. Solving subgraph isomorphism problems with constraint programming. *Constraints* 15:327–353.