Detecting Manipulation in Cup and Round Robin Sports Competitions

Peter van Beek

University of Waterloo

24th IEEE International Conference on Tools with Artificial Intelligence
November 7–9, 2012, Athens, Greece

Joint work with Tyrel Russell

Introduction

- Match rigging has been found in sports ranging from football to sumo wrestling and lawn bowling
- Previous work has focused on identifying the rigging of single matches (see, e.g., Duggan & Levitt, 2002; Hill, 2008; Maennig, 2005)
- However, cheating is known to extend to coalitions of teams rigging multiple matches to manipulate the placement of teams in a competition
- *Example:* 1971–72 Bundesliga scandal in German football
  - involved 52 players, nine teams, and the manipulation of 18 matches
  - aim was to attain the promotion and avoid the relegation of certain teams (see Maennig, 2005)

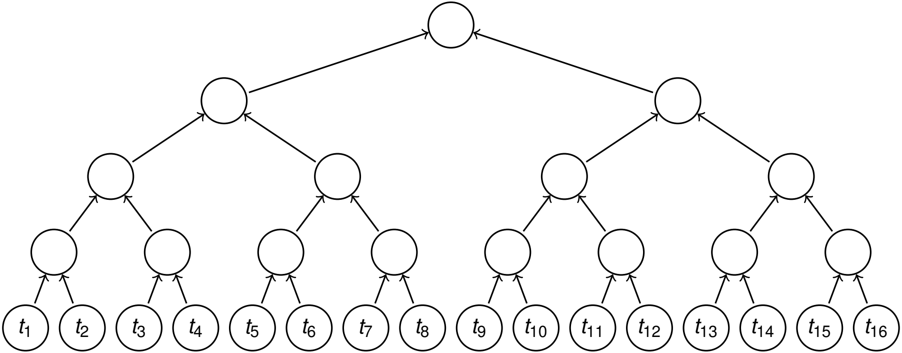
Introduction

- Our work: towards automated tools for detecting a coalition of teams manipulating the winner of a competition
- Central idea:
    - in a competition, some games are upsets (have unexpected results)
    - upsets may be genuine or manipulations
    - look for intentional behavior by recognizing a coalition's plan
    - prefer simpler plans, as each manipulation increases the risk of detection

Cup Competitions

Cup Competitions

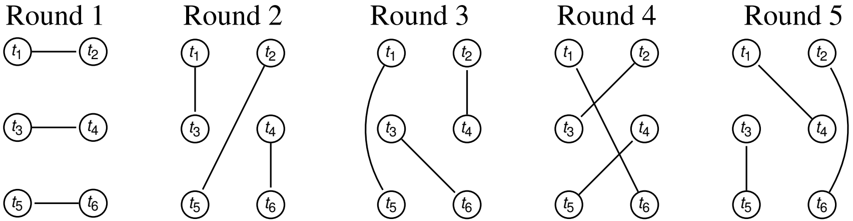
A *cup competition* is a competition where teams are paired in each round and the winner advances to the next round.



Round Robin Competitions

Round Robin Competitions

A *round robin competition* is a competition where each team plays every other team in the competition a specified number of times, usually once or twice.



Tournament Graphs

Tournament Graph

A *tournament graph* is a graph $G = (T, E)$ where T is the set of teams and E contains an edge from t_i to t_j if t_i defeats t_j in a fair game.

$$\begin{array}{c cccccccc}
 & \multicolumn{8}{c}{t_j} \\
 & 0 & 0 & 1 & 1 & 1 & 0 & 1 & 1 \\
 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \\
 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 \\
t_i & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 1 \\
 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 \\
 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 1 \\
 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\
 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0
\end{array}$$

Match Rigging

Upsets

An *upset* is an unexpected defeat; i.e., the team that won in the actual competition is not the team predicted to win by the tournament graph.

Manipulations

A *manipulation* is an upset, either executed or planned, that is intentional.

- Assumptions:
  - (i) some matches are labeled as upsets
  - (ii) tournament graph is known
- Could come from experts who know outcomes, relative strengths of teams, and historically how well teams have played against each other

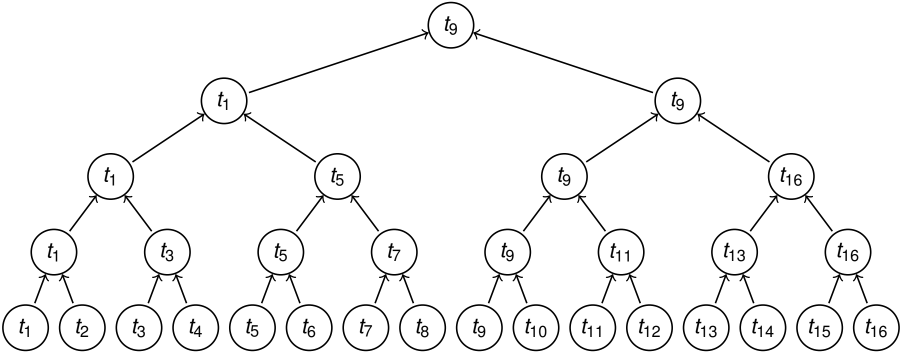
Coalitions Rigging Multiple Matches

Strategically Optimal Coalition

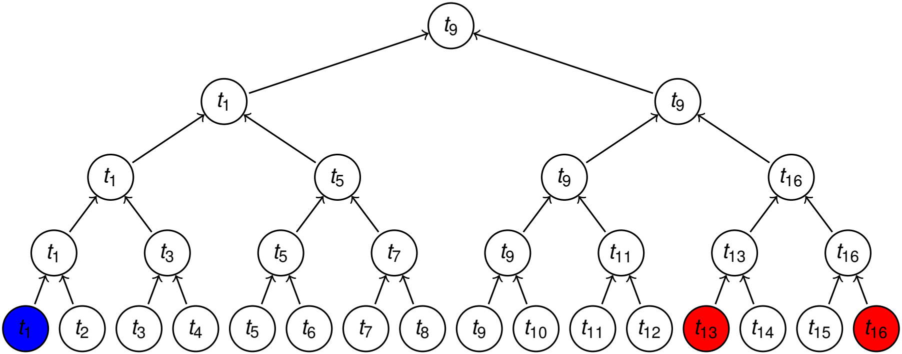
A coalition S is a *strategic coalition* for guaranteeing a team t_w wins if, for each round, the set of upsets by the coalition in that round contains all and only the manipulations that would have been executed in an optimal manipulation strategy for S in that round. A coalition S is a *strategically optimal coalition* if no proper subset of S is a strategic coalition.

- Strategy may need to change between rounds
- Simpler plans preferred
- Can relax optimality requirement: within k manipulations of optimal

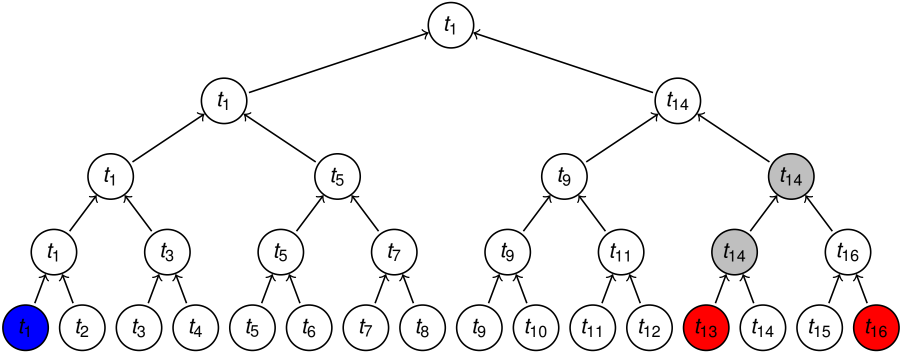
Detecting Strategically Optimal Coalitions in Cups



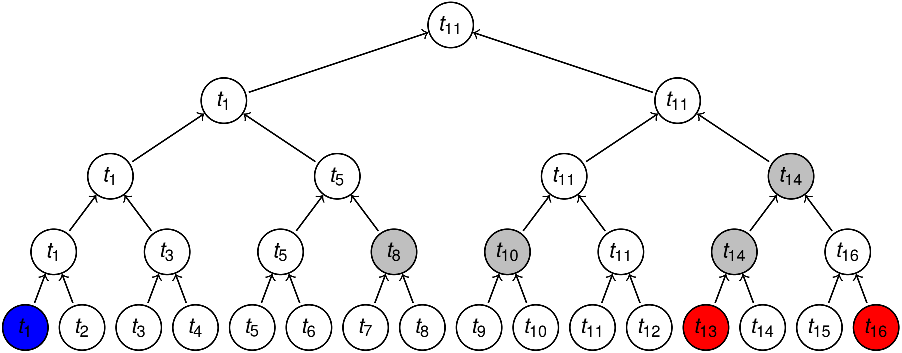
Example 1: Cup Competition



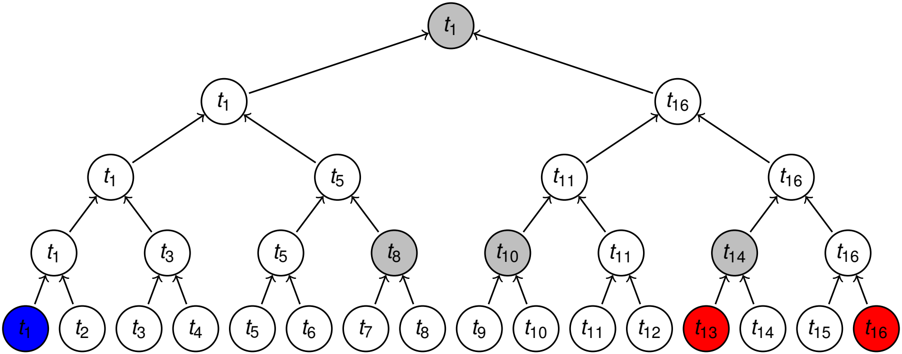
Example 1: Cup Competition



Example 2: Cup Competition



Example 2: Cup Competition

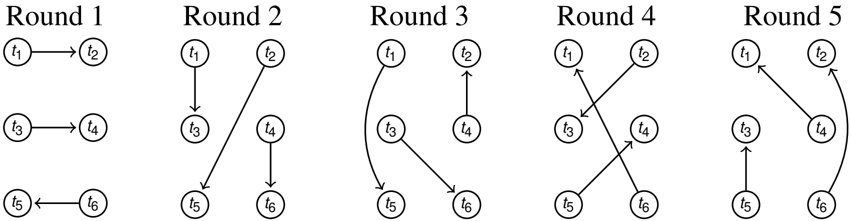


Detecting Strategically Optimal Coalitions in Cups

Algorithm for detecting strategically optimal coalitions in cups

- Posthoc analysis of tournament results
- Uses dynamic programming to construct strategically optimal coalitions
    1. start at leaves (seeding) of cup competition
    2. merge optimal coalitions for two sub-trees
    3. prune based on not establishing desired team and non-optimality (i.e., uses too many manipulations)

Detecting Strategically Optimal Coalitions in Round Robins



Round 1

Round 2

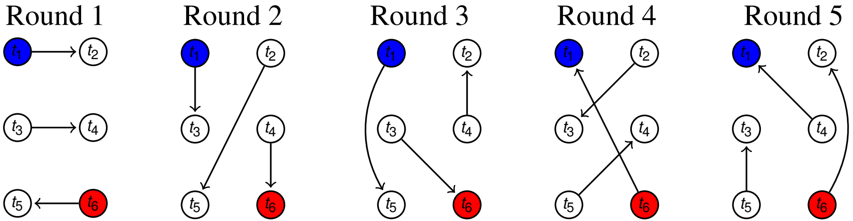
Round 3

Round 4

Round 5

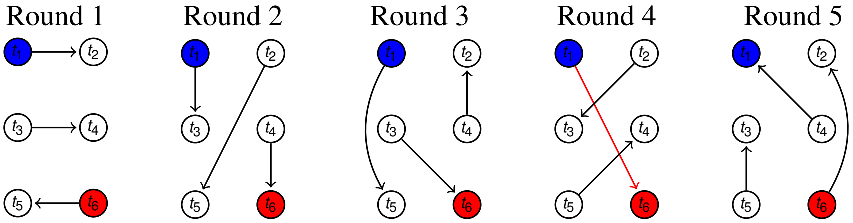
Example 1: Round Robin

- Two types of manipulations: coalition members losing to the desired winner and losing amongst themselves
- *Simple manipulation strategies:* only use first type



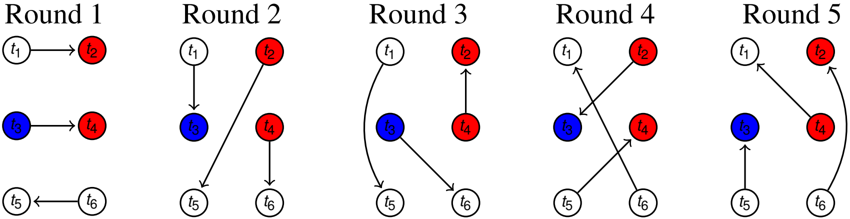
Example 1: Round Robin

- Two types of manipulations: coalition members losing to the desired winner and losing amongst themselves
- *Simple manipulation strategies:* only use first type



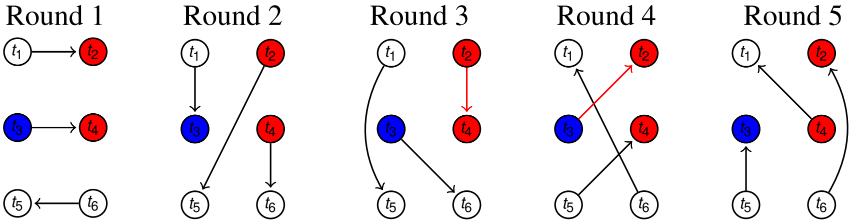
Example 2: Round Robin

- Two types of manipulations: coalition members losing to the desired winner and losing amongst themselves
- *Complex manipulation strategies:* use both types of manipulations



Example 2: Round Robin

- Two types of manipulations: coalition members losing to the desired winner and losing amongst themselves
- *Complex manipulation strategies:* use both types of manipulations



Detecting Strategically Optimal Coalitions in Round Robins

Algorithm for detecting strategically optimal coalitions in round robins

1. construct constraint satisfaction problem
   - state constraints on strategic coalitions that achieve the goal of establishing a team t_w as winner
   - find all such possible coalitions
2. construct minimal cost feasible flow problem
   - prune coalitions that do not achieve the goal in a minimal number of manipulations in each round

Experimental Evaluation

1. Cup competitions: randomly generated instances based on NCAA Division I Basketball Championship:
    - 64 teams ranked and seeded using pools of 16 teams
    - best-plays-worst paradigm
    - tournament graphs and upsets generated from a distribution that was estimated from 25 past championships (1985–2009)
2. Cup competitions: 40 Grand Slam Tennis events (2001–2010)
    - 128 players
    - upset recorded if winner was eight positions or more lower in rank
    - no surprises found
3. Round robins: randomly generated instances
    - instances from 4 to 40 teams
    - with and without coalitions
    - simple and complex manipulation strategies

Experiment 1: Cup Competitions

- NCAA random instances
- Accurately detects manipulation when it occurs
- False positives occur but can be ordered heuristically efficiently

Size	Accuracy	Top 1	Top 10	Top 20
16	76.7	77.7%	100.0%	100.0%
32	81.2	67.8%	100.0%	100.0%
64	85.4	61.3%	99.4%	99.9%
128	89.4	49.1%	94.5%	98.4%
256	93.5	31.7%	78.1%	87.0%

Experiment 3: Round Robins

- Random instances
- Again, accurately detects manipulation when it occurs

	Accuracy		Ave. number	
Size	Simple	Complex	Simple	Complex
6	88.5	98.2	1.6	1.2
12	97.5	100.0	1.6	1.0
18	99.5	98.0	2.0	1.1
24	99.5	100.0	2.2	1.5
30	99.0	100.0	2.1	1.9
36	99.5	100.0	1.6	2.7
40	99.5	100.0	1.5	2.2

Conclusion

- Formalized the notion of strategic behavior of a coalition of teams desiring to manipulate a competition
- Algorithms for detecting such coalitions from upsets
    - for both cup and round robin competitions
    - used constraint programming, dynamic programming, and network flows to detect coalitional cheating
- Experimental evaluation on real and randomly generated instances
    - accurately and quickly identify cheating coalition, if present
    - often no (or a few easily dismissed) false positives, if no cheating present
- The practical benefit of our approach
    - useful tool for posthoc analysis
    - a starting point for further investigation

Future Work

1. Probabilistic model of the competition
    - how can teams manipulate a competition within a probabilistic model
    - tools for recognizing such manipulation
2. Incorporate cost-benefit analysis
    - distinguish likely from unlikely coalitions based on potential reward