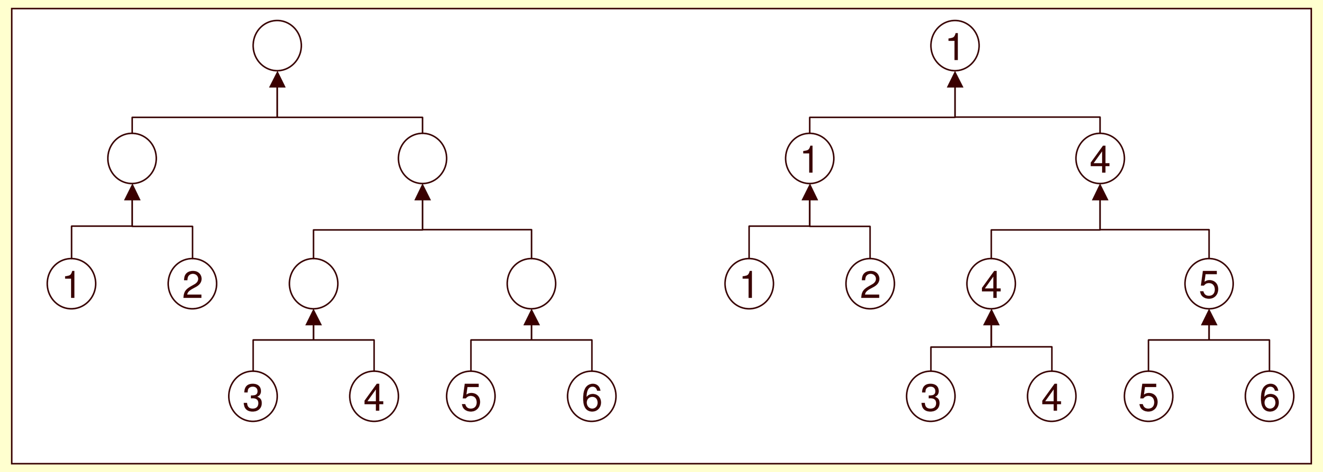
On the Seeding Problems for Knockout Tournaments

Thuc Vu, Alon Altman, Yoav Shoham
Computer Science Department, Stanford University

Knockout Tournament and the Design Space

Knockout is one of the most popular tournament formats

- Players placed at leaf-nodes of a binary tree
- Winner of pairwise matches moving up the tree



Very rich design space with many dimensions:

- Objective functions:
  - Predictive power vs. Fairness vs. Interestingness etc…
- Structures of the tournament
  - Unconstrained vs. Balanced vs. Limited matches
- Models of the players/ Information available
  - Unconstrained vs. Monotonic vs. Deterministic etc…
- Sizes of the problem
  - Exact small cases vs. Unbounded cases
- Type of results
  - Theoretical vs. Experimental

General Model and Problem

Input:

- A set N of players
- A vector V of the values of each player
- Matrix P of winning probabilities
  - $P_{i,j}$ – probability i win against j
  - $0 \le P_{i,j}=1- P_{j,i} \le 1$
  - No transitivity required

A knockout tournament K is defined by:

- Tournament structure T – a binary tree
- Seeding S – a mapping from N to leaf nodes of T

Objective functions:

- MaxP: Maximizing the winning probability of a target player
- MaxE: Maximizing the expected value of the winner
- MaxI: Maximizing the interestingness of a tournament
  - Defined as the expected sum of the interestingness of each possible match, which in turn is dependent on the players' values and the winning probabilities between them. The interestingness of a match is proportional to the competitiveness between the players and their values.

Goal: Given the input, find (T,S) that achieves the target objective

Theoretical Results

The setting:

- Objective functions: MaxP for a target player *k*
- Tournament structure: Balanced knockout tournament
- Model of the players: General winning probabilities

Theorem:

Given a set of players N, and a winning probability matrix P, it is NP-hard to find the optimal balanced knockout tournament KT that maximizes the winning probability of the target player *k*.

Proof Idea: Show the decision version NP-complete by using a reduction from the Vertex Cover problem. We construct a tournament *KT* with a special player *o* such that *o* wins *KT* if and only if there exists a vertex cover of size at most k.

Extension of the result: The theorem above holds even when the winning probabilities can be 0, 1, or 0.5 only.

Experimental Setup

The setting:

- Objective functions: MaxP, MaxE, and MaxI
- Tournament structure: Balanced knockout tournament
- Model of the players: Monotonic
- Size of the tournament: Up to 128 players
  - Previous work: Only for tournament of size 4 and 8

Monotonic model: The winning probability matrix P satisfies

- $P_{i,j}+P_{j,i}=1$
- $P_{i,j} \ge P_{j,i}$ for all (i,j): i≤j
- $P_{i,j} \le P_{j,j+1}$ for all (i,j)

Types of solution:

- Ordinal: The seeding is based on the ordering of the players only
- Cardinal: The seeding is based on the actual winning probabilities

Heuristic Algorithm for Cardinal Solutions:

The algorithm is based on a Hill-Climbing approach. Given an initial seeding, the algorithm keep trying to improve the objective value by swapping random pairs of sub-tournament trees.

Evaluation Method:

Since the number of distinct seedings grows extremely fast for tournaments of more than 8 players (638×10^6 seedings for 16 players), we propose an effective and efficient upper bound for each of the objective functions to evaluate the optimality of different solutions.

Experimental Results

Results for |N|=8:

Special ordinal and cardinal solutions:

- S^8_1: 1 8 7 6 5 4 3 2
- S^8_2 : 1 8 4 5 2 7 3 6
- $HC(S^8_1)$& $HC(S^8_2)$: heuristic function applied on S^8_1& S^8_2 respectively

Observations:

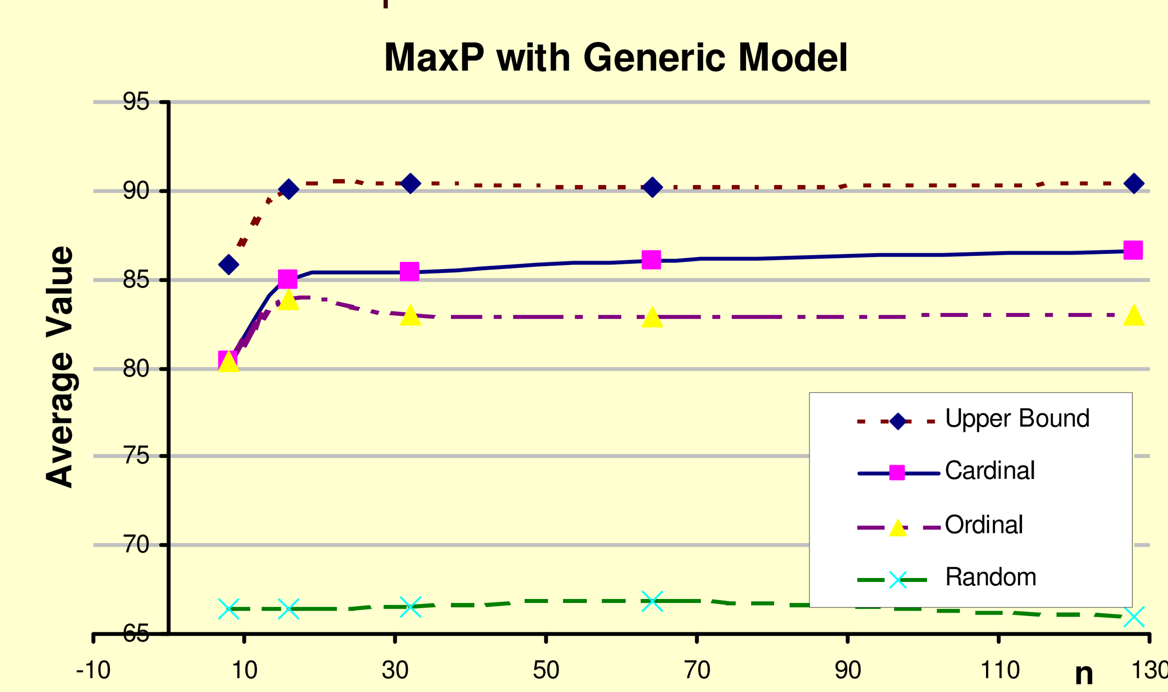
- No ordinal solution performs well across all objectives
- For MaxE and MaxI, no ordinal solution achieves optimality with high frequency
- Cardinal solutions almost always achieve optimality

	MaxP		MaxE		MaxI	
	Freq.	Val.	Freq.	Val.	Freq.	Val.
Ordinal						
Average	NA	71.88	NA	94.54	NA	92.93
Worst	NA	57.82	NA	87.36	NA	82.88
17452836	0	77.54	2.03	97.94	11.95	99.1
S^8_2	0	84.12	6.87	98.68	**22.6**	**99.5**
18562734	0	90.47	11.29	99.22	12.69	99.04
18572634	0	93.6	9.05	99.25	0.37	97.98
S^8_1	**99.78**	**100**	4.06	99.11	0.3	96.2
18672435	0.89	99.14	9.7	99.45	0.05	96.84
18672534	0.22	98.74	**40.07**	**99.63**	1.43	97.26
Cardinal						
$HC(S^8_1)$	**100**	**100**	**93.97**	**99.99**	**91.6**	**99.97**
$HC(S^8_2)$	**100**	**100**	**94.99**	**99.99**	**95.79**	**99.99**

The average percentage compared to the optimal values and the percentage of tournaments with optimality over 100M tournaments

Results for |N|>8:

As |N| grows exponentially, the objective values of our cardinal solutions remain close to the values of the upper bound. This implies that our upper bound is relatively tight, and our cardinal solution is close to being optimal. Using both of them allows us to have a good approximation of the optimal values.



The upper bound and the objective values of different solutions for MaxP