Small Voting Trees

Michael A. Trick

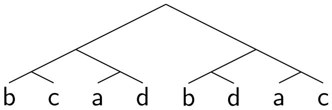
Tepper School of Business
Carnegie Mellon University
Pittsburgh, PA USA
trick@cmu.edu

COMSOC Amsterdam
December 2006

Outline

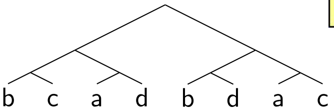
Implementable Rules

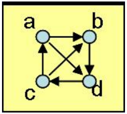
We are concerned with voting trees, with candidates at leaves.
Winner is determined by working up from leaves, using majority voting to choose between the two candidates.



Implementable Rules

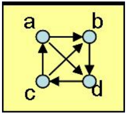
We are concerned with voting trees, with candidates at leaves.
Winner is determined by working up from leaves, using majority voting to choose between the two candidates.

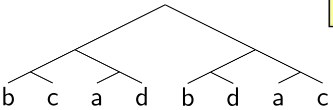


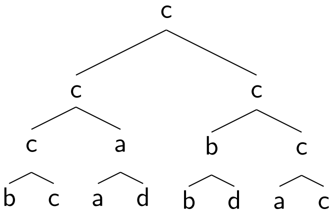


Implementable Rules

We are concerned with voting trees, with candidates at leaves.
Winner is determined by working up from leaves, using majority voting to choose between the two candidates.

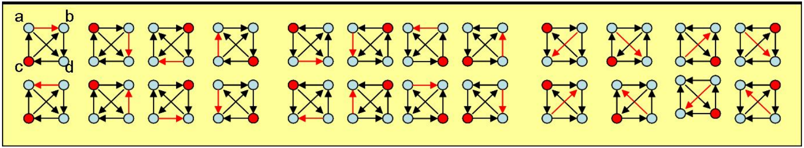






Implementable Rules

Given a set of tournaments $\mathcal{T}$, a voting tree defines a *rule* over $\mathcal{T}$.
Over all tournaments on 4 candidates with all candidates in top cycle, the previous tree gives the following rule:



(This is actually the Copeland rule with 2nd-order Copeland tiebreaking)

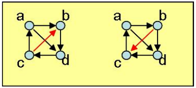
Question

What rules are implementable by voting trees?

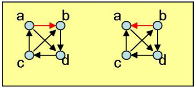
Implementable Rules

Assumption: all candidates appear in tree (rule is onto).
Clearly, rule must choose from top cycle of each tournament (including choosing Condorcet winner if it exists)

Sufficient?



all 16 pairs of winners are implementable.



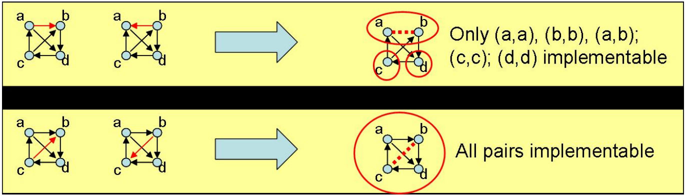
only $(a, a), (b, b), (c, c), (d, d)$ and (a, b) are implementable.

Pairwise Conjecture

Conjecture (Pairwise Conjecture)

A rule defined over all tournaments of n candidates is implementable if and only if it is implementable over all pairs of tournaments. (Srivasta and Trick, 1996)

Srivastava and Trick also give necessary and sufficient conditions for a rule to be implementable over a pair of tournaments.



Computational Procedure

- After a decade, little progress on conjecture (but no counterexamples either!)

Computational Procedure

- After a decade, little progress on conjecture (but no counterexamples either!)
- Want a computational procedure to provide verification (or find counterexample)

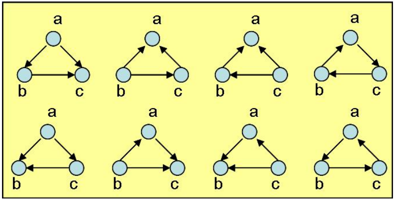
Computational Procedure

- After a decade, little progress on conjecture (but no counterexamples either!)
- Want a computational procedure to provide verification (or find counterexample)
- Algorithm to generate all small rules over small number of candidates (number of rules increases quickly with number of candidates)

Computational Procedure

- After a decade, little progress on conjecture (but no counterexamples either!)
- Want a computational procedure to provide verification (or find counterexample)
- Algorithm to generate all small rules over small number of candidates (number of rules increases quickly with number of candidates)
- Details in paper (essentially complete enumeration, using hash tables to quickly determine if a newly generated rule has already been generated)

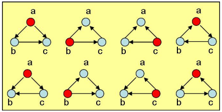
Rules on 3 Candidates

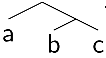


There are 8 tournaments on 3 candidates, so there are $3^8 = 6561$ rules over these tournaments. Of these, only 9 rules are Condorcet. The Pairwise conjecture requires each of these 9 to be implementable, and the computational procedure shows that to be the case.

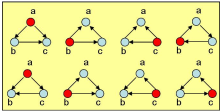
Tournaments on 3 Candidates

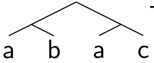
Always choose Condorcet candidate if it exists. Else:



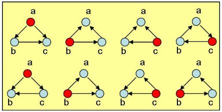


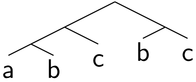
Type I: Choose *a*





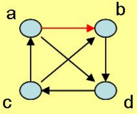
Type II: Choose candidate that beats *a*





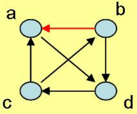
Type III: Choose candidate that loses to *a*

Structure of 4 Candidate Tournaments



Copeland: 2
a, c
Copeland: 1
b, d

Partner



Copeland: 2
b, c
Copeland: 1
a, d

Define
w_1: c
w_2: a(b)
l_1: b(a)
l_2: d

Implementable
w_1 : (c,c)
w_2 : (a,b)
l_2 : (d,d)

(w_2,l_1) : (a,a)
(l_1,w_2) : (b,b)

Number of Rules on 4 Candidates

- There are $2^6 = 64$ tournaments on 4 candidates, so $4^{64} = 3.4 * 10^{38}$ rules.

Number of Rules on 4 Candidates

- There are $2^6 = 64$ tournaments on 4 candidates, so $4^{64} = 3.4 * 10^{38}$ rules.
- This is reduced to $4^{24}3^8 = 1.8 * 10^{18}$ rules that choose from the top cycle (including Condorcet winners).

Number of Rules on 4 Candidates

- There are $2^6 = 64$ tournaments on 4 candidates, so $4^{64} = 3.4 * 10^{38}$ rules.
- This is reduced to $4^{24}3^8 = 1.8 * 10^{18}$ rules that choose from the top cycle (including Condorcet winners).
- Since the 24 tournaments with four candidates can be divided into pairs for which only 5 of the 16 possible pairs of winners is implementable, if the Pairwise Conjecture is true, there are $5^{12}3^8 = 1{,}601{,}806{,}640{,}625$ implementable rules.

Number of Rules on 4 Candidates

- There are $2^6 = 64$ tournaments on 4 candidates, so $4^{64} = 3.4 * 10^{38}$ rules.
- This is reduced to $4^{24}3^8 = 1.8 * 10^{18}$ rules that choose from the top cycle (including Condorcet winners).
- Since the 24 tournaments with four candidates can be divided into pairs for which only 5 of the 16 possible pairs of winners is implementable, if the Pairwise Conjecture is true, there are $5^{12}3^8 = 1{,}601{,}806{,}640{,}625$ implementable rules.
- If we ignore the tournaments with just 3 candidates in the top cycle (but still require Condorcet rules), that gives $5^{12} = 244{,}140{,}625$ which is at least conceivable for our computational approach to find.

Number of Rules on 4 Candidates

- There are $2^6 = 64$ tournaments on 4 candidates, so $4^{64} = 3.4 * 10^{38}$ rules.
- This is reduced to $4^{24}3^8 = 1.8 * 10^{18}$ rules that choose from the top cycle (including Condorcet winners).
- Since the 24 tournaments with four candidates can be divided into pairs for which only 5 of the 16 possible pairs of winners is implementable, if the Pairwise Conjecture is true, there are $5^{12}3^8 = 1{,}601{,}806{,}640{,}625$ implementable rules.
- If we ignore the tournaments with just 3 candidates in the top cycle (but still require Condorcet rules), that gives $5^{12} = 244{,}140{,}625$ which is at least conceivable for our computational approach to find.
- Of these 4096 choose among the Copeland winners, and 1 chooses only Copeland losers (interesting to find these).

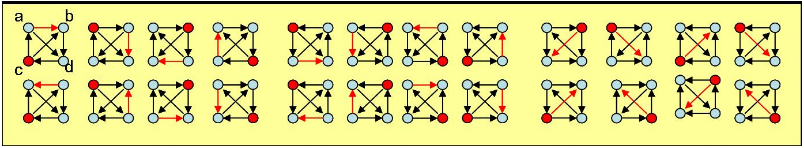
Results so far

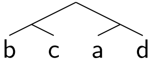
We have found 19,650,758 rules so far, 3102 of the Copeland winner (out of 4096), and the Copeland Loser Rule (up to 26 leaves in the tree).

Size	Number	Copeland
4	15	3
5	102	0
6	424	0
7	1104	0
8	2377	19
9	5486	4
10	11232	18
11	21768	36
12	40420	36
13	70600	96
14	116670	60
15	187560	96

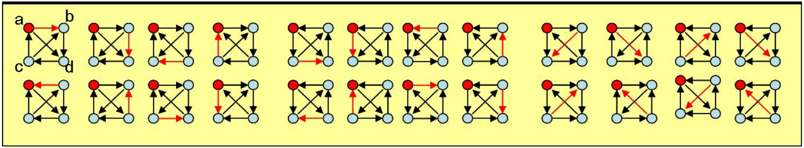
Size	Number	Copeland
16	294510	240
17	439102	192
18	633986	138
19	895648	292
20	1231551	368
21	1655920	148
22	2188704	240
23	2829882	318
24	3595685	276
25	4464020	296
26	5428012	224

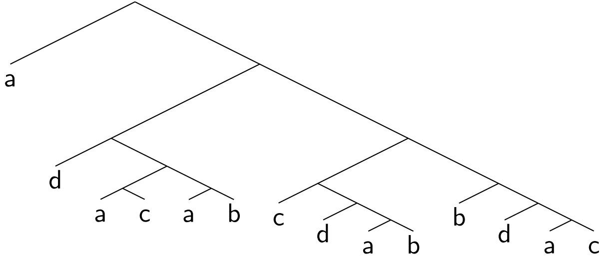
Smallest Rule, Choice from Copeland Winners





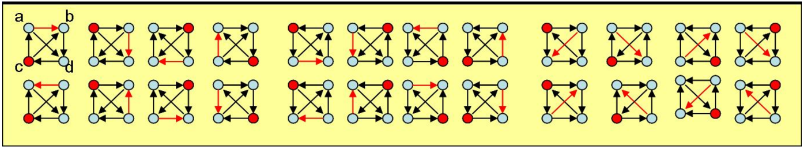
4 Node, Top Cycle, Lexicographic Tiebreak

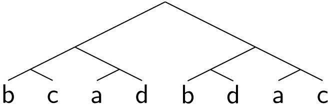




4 Node, Copeland winner, 2nd Order Copeland tiebreak

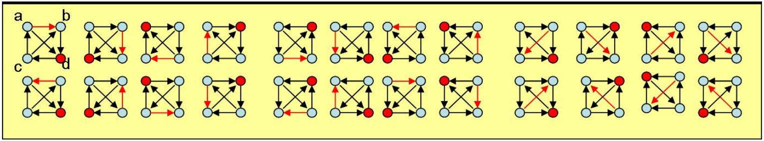
Always choose w_1

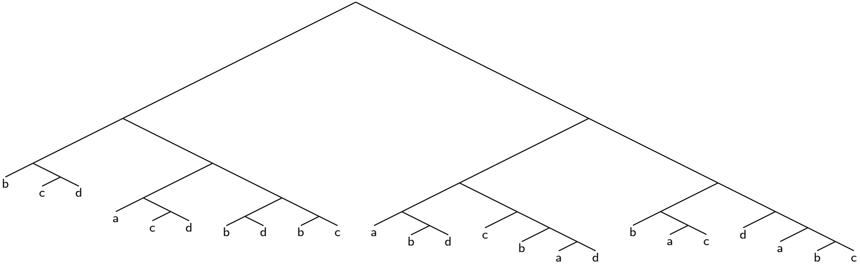




Copeland Loser in Top Cycle, 2nd Order Copeland tiebreak

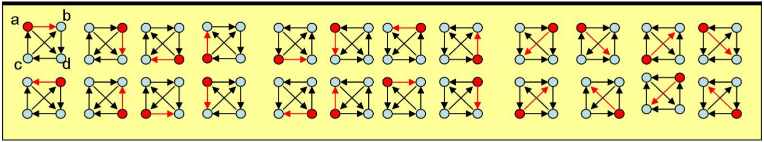
Always choose l_2

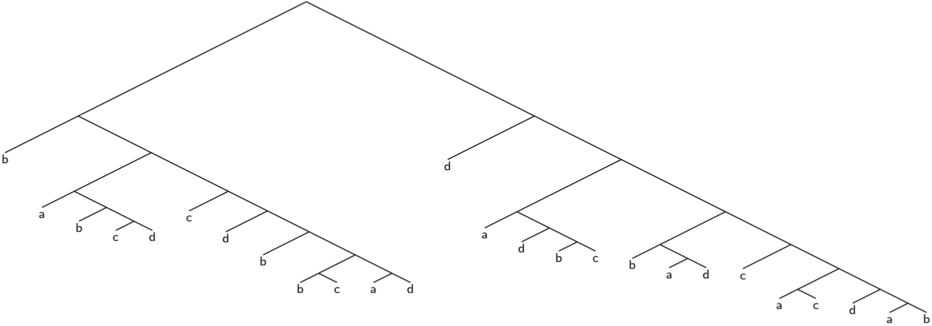




Copeland Winner, Reverse Copeland tiebreak

Always choose w_2. **Not found yet**





(Chooses correctly 15/24 times)

Conclusions and Future Directions

- Rules get very complicated very quickly (not surprising literature has few rules). 90% of rules on 4 candidates have at least 27 leaves!

Conclusions and Future Directions

- Rules get very complicated very quickly (not surprising literature has few rules). 90% of rules on 4 candidates have at least 27 leaves!
- Computational procedure reasonable approach for 4 candidates

Conclusions and Future Directions

- Rules get very complicated very quickly (not surprising literature has few rules). 90% of rules on 4 candidates have at least 27 leaves!
- Computational procedure reasonable approach for 4 candidates

- Improving procedure: use of symmetry

Conclusions and Future Directions

- Rules get very complicated very quickly (not surprising literature has few rules). 90% of rules on 4 candidates have at least 27 leaves!
- Computational procedure reasonable approach for 4 candidates

- Improving procedure: use of symmetry
- Characterization of rules with small trees

Conclusions and Future Directions

- Rules get very complicated very quickly (not surprising literature has few rules). 90% of rules on 4 candidates have at least 27 leaves!
- Computational procedure reasonable approach for 4 candidates

- Improving procedure: use of symmetry
- Characterization of rules with small trees
- Bounding tree size

Conclusions and Future Directions

- Rules get very complicated very quickly (not surprising literature has few rules). 90% of rules on 4 candidates have at least 27 leaves!
- Computational procedure reasonable approach for 4 candidates

- Improving procedure: use of symmetry
- Characterization of rules with small trees
- Bounding tree size
- Would appreciate someone proving or disproving Pairwise Conjecture!