A Computational Analysis of the Tournament Equilibrium Set

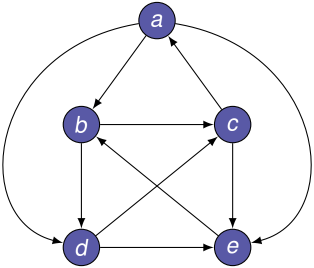
Felix Brandt Felix Fischer Paul Harrenstein Maximilian Mair

Ludwig-Maximilians-Universität, München

COMSOC, 4th September 2008

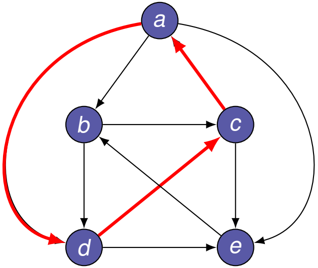
The Trouble with Tournaments

- Tournaments are complete and asymmetric graphs
- Multiple applications in: social choice theory, sports tournaments, game theory, psychometrics, biology, argumentation theory, webpage and journal ranking, etc.
- *However, how to select the winners of a tournament in the absence of transitivity?*



The Trouble with Tournaments

- Tournaments are complete and asymmetric graphs
- Multiple applications in: social choice theory, sports tournaments, game theory, psychometrics, biology, argumentation theory, webpage and journal ranking, etc.
- *However, how to select the winners of a tournament in the absence of transitivity?*

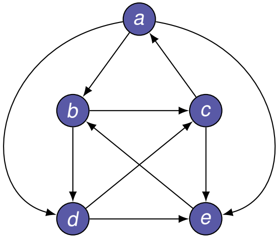


Overview

- Schwartz's *Tournament Equilibrium Set (TEQ)*
- How appealing is TEQ as a tournament solution?
- Schwartz's conjecture and monotonicity of TEQ
- Computational intractability of TEQ
- Heuristic and experiments
- Conclusion

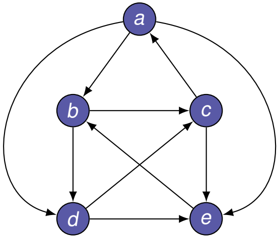
Tournaments

- A ***tournament*** $T = (A, \succ)$ consists of:
  - a finite set of ***alternatives*** A
  - a complete and asymmetric relation $\succ$ on A
  - $\overline{D}(a) = \{x \in A : x \succ a\}$, the set of dominators of a



- A ***tournament solution*** S associates each tournament $T = (A, \succ)$ with a subset $S(T)$ of A such that:
  - $S(T)$ non-empty if A is non-empty
  - $S(T)$ consists of the Condorcet winner only if there is one
- Examples: Copeland set, Top Cycle, Uncovered Set, Banks Set, Minimal Covering Set, Essential Set, ***Tournament Equilibrium Set (TEQ)***...

Tournaments



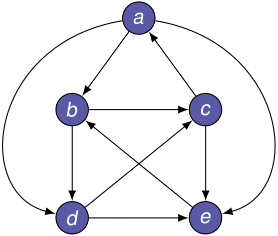
- A ***tournament*** $T = (A, \succ)$ consists of:
  - a finite set of ***alternatives*** A
  - a complete and asymmetric relation $\succ$ on A
  - $\overline{D}(a) = \{x \in A : x \succ a\}$, the set of dominators of a

- A ***tournament solution*** S associates each tournament $T = (A, \succ)$ with a subset $S(T)$ of A such that:
  - $S(T)$ non-empty if A is non-empty
  - $S(T)$ consists of the Condorcet winner only if there is one

- Examples: Copeland set, Top Cycle, Uncovered Set, Banks Set, Minimal Covering Set, Essential Set, ***Tournament Equilibrium Set (TEQ)***...

Tournaments

- A ***tournament*** $T = (A, \succ)$ consists of:
  - a finite set of ***alternatives*** A
  - a complete and asymmetric relation $\succ$ on A
  - $\overline{D}(a) = \{x \in A : x \succ a\}$, the set of dominators of a



- A ***tournament solution*** S associates each tournament $T = (A, \succ)$ with a subset $S(T)$ of A such that:
  - $S(T)$ non-empty if A is non-empty
  - $S(T)$ consists of the Condorcet winner only if there is one

- Examples: Copeland set, Top Cycle, Uncovered Set, Banks Set, Minimal Covering Set, Essential Set, ***Tournament Equilibrium Set (TEQ)***. . .

Retentiveness and the Tournament Equilibrium Set (TEQ)

Intuition: For S a solution concept:

- An alternative a is only "properly" dominated, if dominated by a "good" alternative
- No alternative selected by S should be "properly" dominated by an "outside" alternative not selected by S

Thomas Schwartz

Retentiveness and the Tournament Equilibrium Set (TEQ)

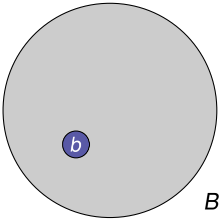
Intuition: For S a solution concept:

- An alternative a is only "properly" dominated, if dominated by ***an alternative selected by*** S ***from the dominators of*** a
- No alternative selected by S should be "properly" dominated by an "outside" alternative not selected by S

Thomas Schwartz

Retentiveness and the Tournament Equilibrium Set (TEQ)

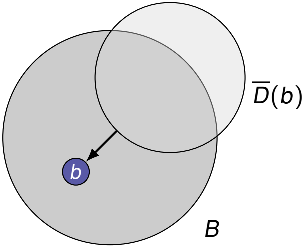
Definition: B is S-*retentive* if $B \neq \emptyset$ and $S(\overline{D}(b)) \subseteq B$ for all $b \in B$



Thomas Schwartz

Retentiveness and the Tournament Equilibrium Set (TEQ)

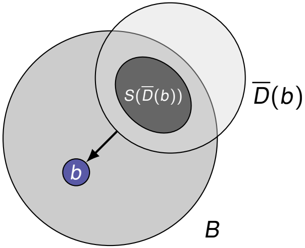
Definition: B is S-*retentive* if $B \neq \emptyset$ and $S(\overline{D}(b)) \subseteq B$ for all $b \in B$



Thomas Schwartz

Retentiveness and the Tournament Equilibrium Set (TEQ)

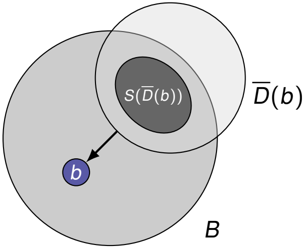
Definition: B is S-*retentive* if $B \neq \emptyset$ and $S(\overline{D}(b)) \subseteq B$ for all $b \in B$



Thomas Schwartz

Retentiveness and the Tournament Equilibrium Set (TEQ)

Definition: B is *S-retentive* if $B \neq \emptyset$ and $S(\overline{D}(b)) \subseteq B$ for all $b \in B$



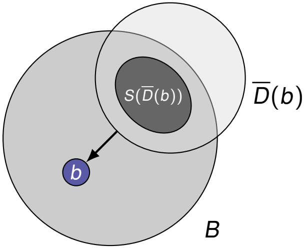
Thomas Schwartz

Definition $\mathring{S}$ returns the union of minimal S-retentive subsets

Definition TEQ is recursively defined by $TEQ(T) = \mathring{TEQ}(T)$

Retentiveness and the Tournament Equilibrium Set (TEQ)

Definition: B is *S-retentive* if $B \neq \emptyset$ and $S(\overline{D}(b)) \subseteq B$ for all $b \in B$

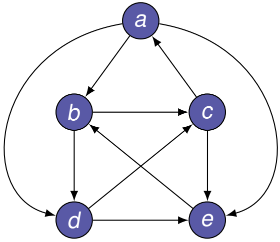


Thomas Schwartz

Alternative characterization

- TEQ-relation: $x \rightarrow y$ if and only if $x \in TEQ(\overline{D}(y))$
- TEQ is the top cycle of the TEQ-relation

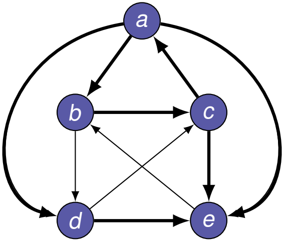
Retentiveness and the Tournament Equilibrium Set (TEQ)



Thomas Schwartz

x	$\overline{D}(x)$	$TEQ(\overline{D}(x))$
a	$\{c\}$	$\{c\}$
b	$\{a, e\}$	$\{a\}$
c	$\{b, d\}$	$\{b\}$
d	$\{a, b\}$	$\{a\}$
e	$\{a, c, d\}$	$\{a, c, d\}$

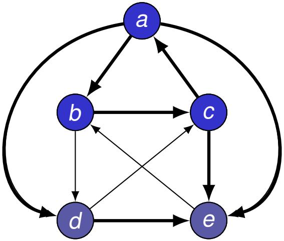
Retentiveness and the Tournament Equilibrium Set (TEQ)



Thomas Schwartz

x	$\overline{D}(x)$	$TEQ(\overline{D}(x))$
a	$\{c\}$	$\{c\}$
b	$\{a, e\}$	$\{a\}$
c	$\{b, d\}$	$\{b\}$
d	$\{a, b\}$	$\{a\}$
e	$\{a, c, d\}$	$\{a, c, d\}$

Retentiveness and the Tournament Equilibrium Set (TEQ)



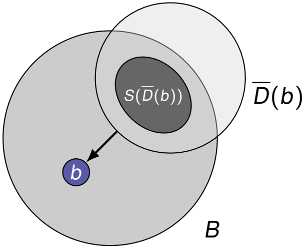
Thomas Schwartz

x	$\overline{D}(x)$	$TEQ(\overline{D}(x))$
a	$\{c\}$	$\{c\}$
b	$\{a, e\}$	$\{a\}$
c	$\{b, d\}$	$\{b\}$
d	$\{a, b\}$	$\{a\}$
e	$\{a, c, d\}$	$\{a, c, d\}$

$$TEQ(T) = \{a, b, c\}$$

Retentiveness and the Tournament Equilibrium Set (TEQ)

Definition: B is *S-retentive* if $B \neq \emptyset$ and $S(\overline{D}(b)) \subseteq B$ for all $b \in B$



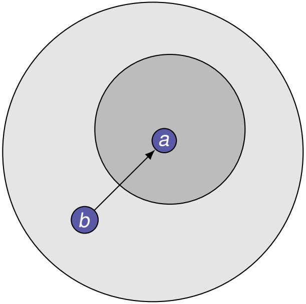
Thomas Schwartz

Definition *TEQ* satisfies CTC (Connected Top Cycle) if there is always a unique minimal TEQ-retentive subset

Schwartz's Conjecture: *TEQ satsifies CTC.*

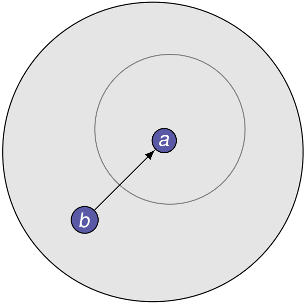
Desirable Properties of TEQ

- Monotonicity (MON)
- Strong Superset Property (SSP)
- Independence of non-winners (INW)



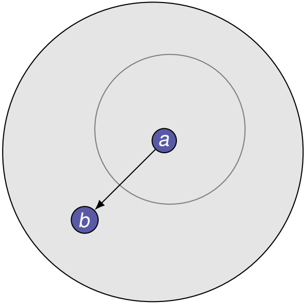
Desirable Properties of TEQ

- Monotonicity (MON)
- Strong Superset Property (SSP)
- Independence of non-winners (INW)



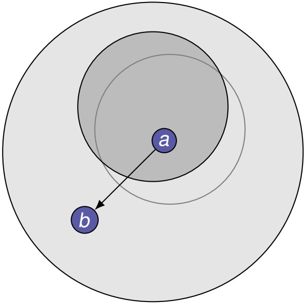
Desirable Properties of TEQ

- Monotonicity (MON)
- Strong Superset Property (SSP)
- Independence of non-winners (INW)



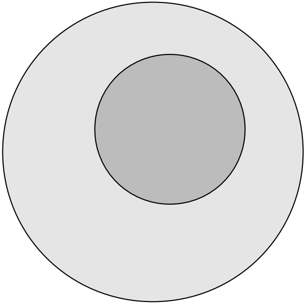
Desirable Properties of TEQ

- Monotonicity (MON)
- Strong Superset Property (SSP)
- Independence of non-winners (INW)



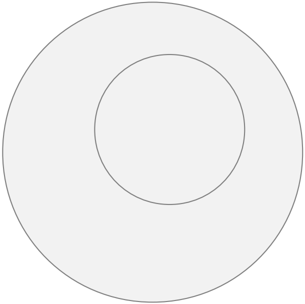
Desirable Properties of TEQ

- Monotonicity (MON)
- Strong Superset Property (SSP)
- Independence of non-winners (INW)



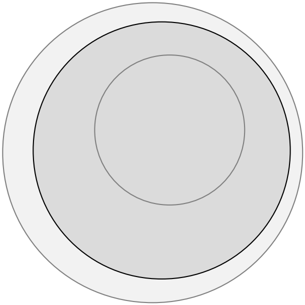
Desirable Properties of TEQ

- Monotonicity (MON)
- Strong Superset Property (SSP)
- Independence of non-winners (INW)



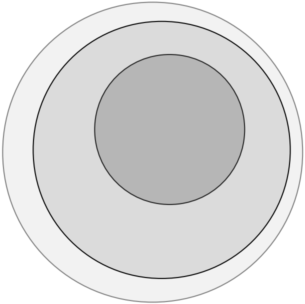
Desirable Properties of TEQ

- Monotonicity (MON)
- Strong Superset Property (SSP)
- Independence of non-winners (INW)



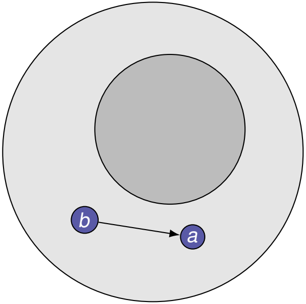
Desirable Properties of TEQ

- Monotonicity (MON)
- Strong Superset Property (SSP)
- Independence of non-winners (INW)



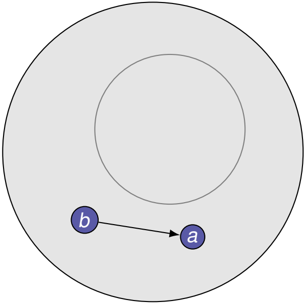
Desirable Properties of TEQ

- Monotonicity (MON)
- Strong Superset Property (SSP)
- Independence of non-winners (INW)



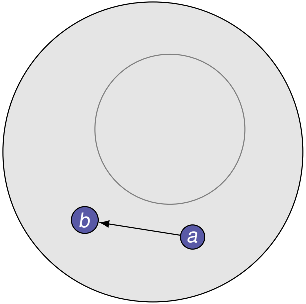
Desirable Properties of TEQ

- Monotonicity (MON)
- Strong Superset Property (SSP)
- Independence of non-winners (INW)



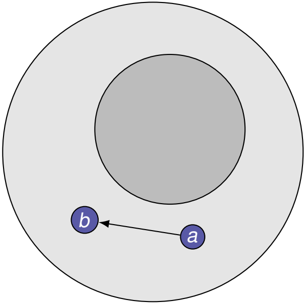
Desirable Properties of TEQ

- Monotonicity (MON)
- Strong Superset Property (SSP)
- Independence of non-winners (INW)



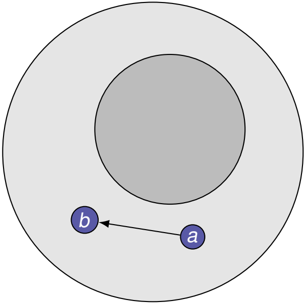
Desirable Properties of TEQ

- Monotonicity (MON)
- Strong Superset Property (SSP)
- Independence of non-winners (INW)



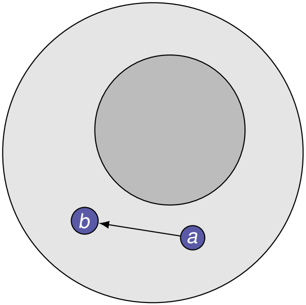
Desirable Properties of TEQ

- Monotonicity (MON)
- Strong Superset Property (SSP)
- Independence of non-winners (INW)



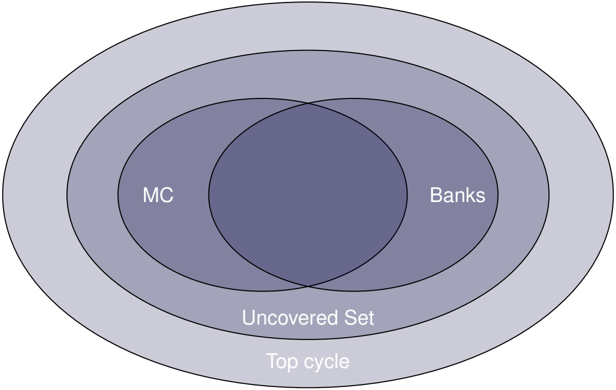
Desirable Properties of TEQ

- Monotonicity (MON)
- Strong Superset Property (SSP)
- Independence of non-winners (INW)

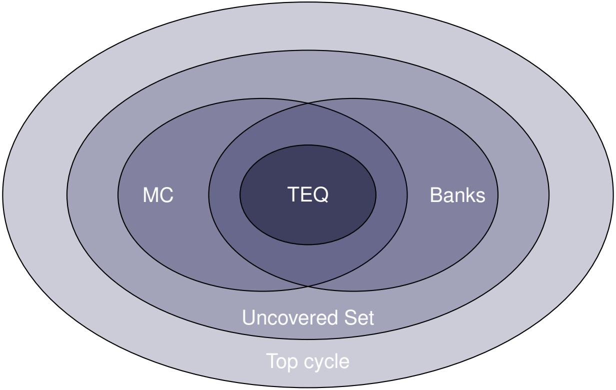


Theorem (Laffond et al., 1993): *TEQ satisfying CTC is equivalent to TEQ satisfying SSP, to TEQ satisfying INW, as well as to TEQ satisfying CTC.*

Inclusions



Inclusions



Computational Intractability of TEQ

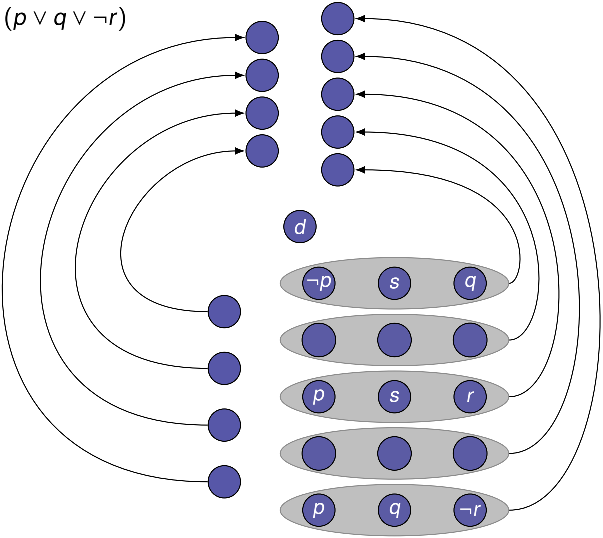
Theorem *Deciding whether an alternative is in TEQ is NP-hard.*

Proof: Reduction from 3-SAT, also observing that

- the construction also works for membership in the Banks set
- TEQ is included in the Banks set. □

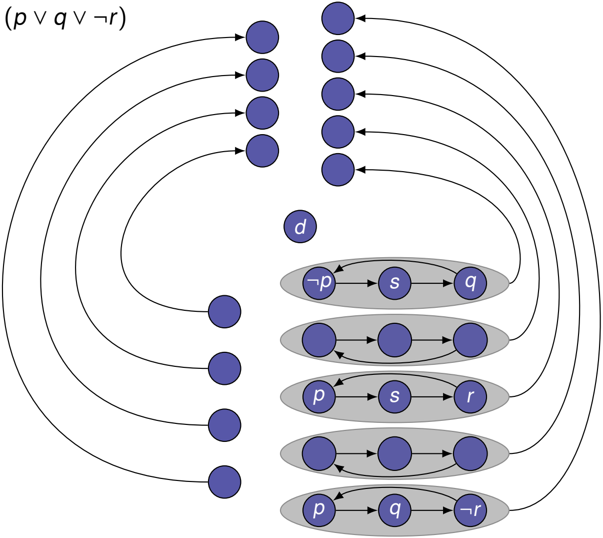
Computational Intractability of TEQ

$(\neg p \vee s \vee q) \wedge (p \vee s \vee r) \wedge (p \vee q \vee \neg r)$



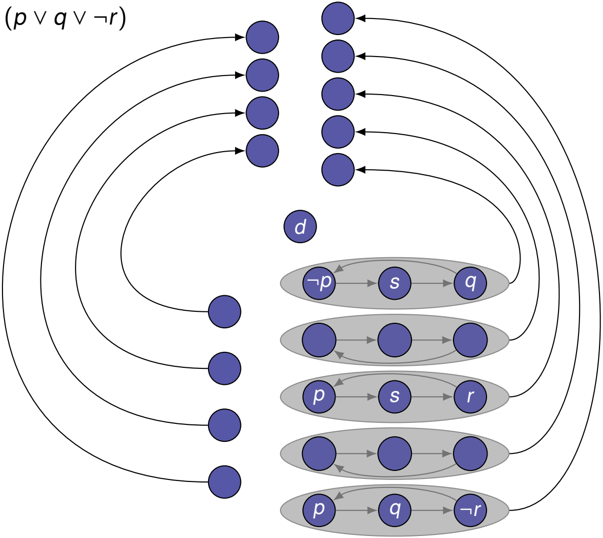
Computational Intractability of TEQ

$(\neg p \vee s \vee q) \wedge (p \vee s \vee r) \wedge (p \vee q \vee \neg r)$



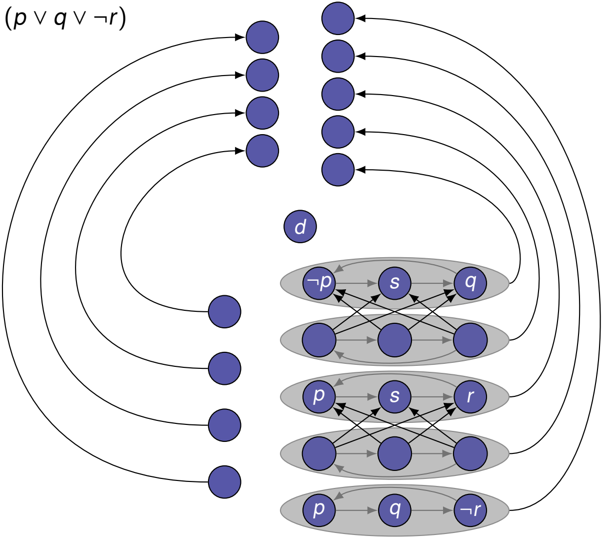
Computational Intractability of TEQ

$(\neg p \vee s \vee q) \wedge (p \vee s \vee r) \wedge (p \vee q \vee \neg r)$



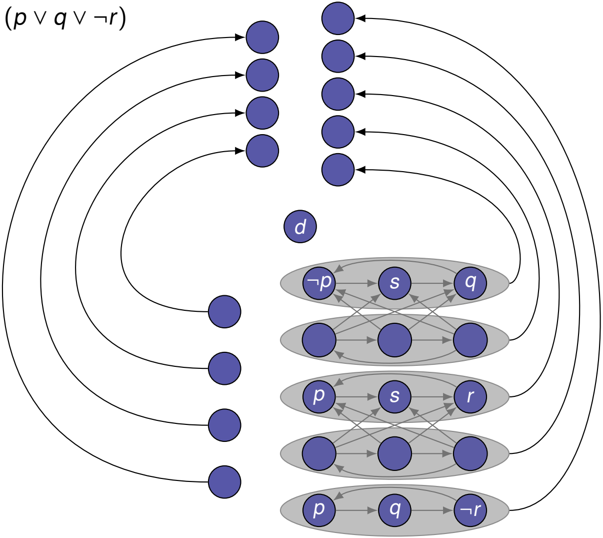
Computational Intractability of TEQ

$(\neg p \vee s \vee q) \wedge (p \vee s \vee r) \wedge (p \vee q \vee \neg r)$



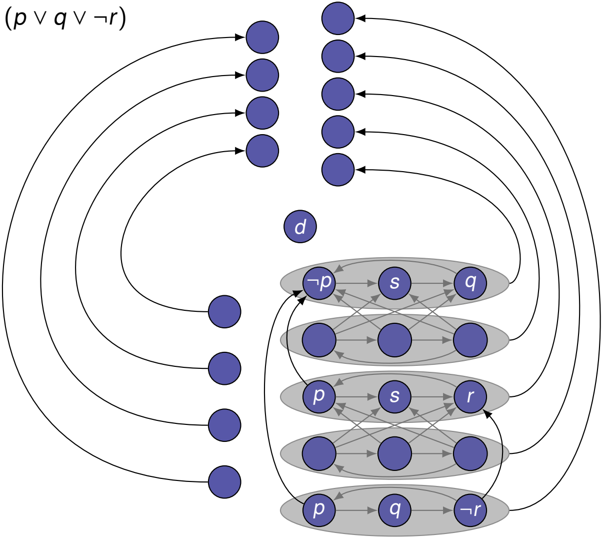
Computational Intractability of TEQ

$(\neg p \vee s \vee q) \wedge (p \vee s \vee r) \wedge (p \vee q \vee \neg r)$



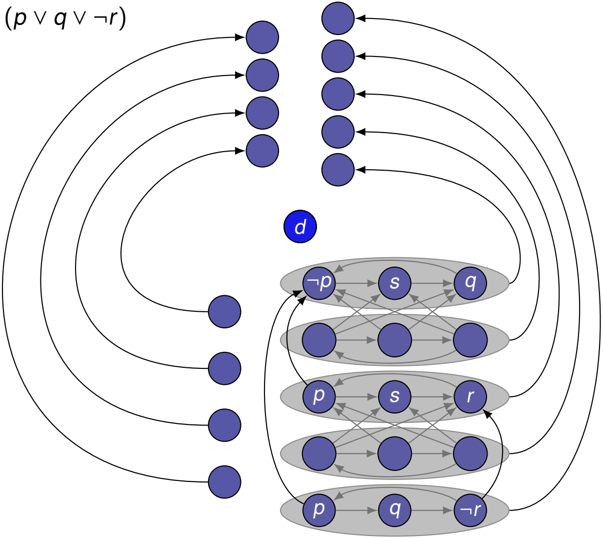
Computational Intractability of TEQ

$(\neg p \vee s \vee q) \wedge (p \vee s \vee r) \wedge (p \vee q \vee \neg r)$



Computational Intractability of TEQ

$(\neg p \vee s \vee q) \wedge (p \vee s \vee r) \wedge (p \vee q \vee \neg r)$



Computational Intractability of TEQ

Theorem *Deciding whether an alternative is in TEQ is NP-hard.*

Proof: Reduction from 3-SAT, also observing that

- the construction also works for membership in the Banks set
- TEQ is included in the Banks set. □

Computational Intractability of TEQ

Theorem *Deciding whether an alternative is in TEQ is NP-hard.*

Proof: Reduction from 3-SAT, also observing that

- the construction also works for membership in the Banks set
- TEQ is included in the Banks set. □

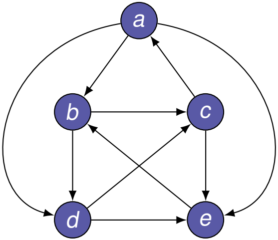
Remarks:

- Computing TEQ is also intractable.
- Compare Woeginger's reduction from graph three-colorability for NP-completeness of membership in the Banks set
- NP-hardness result does not depend on Schwartz's conjecture
- Reduction shows the NP-hardness of any tournament solution between Banks and TEQ

A Heuristic for Computing TEQ

- Recursive definition of TEQ suggests an exponential naive algorithm
- Naive algorithm can be improved upon by *assuming TEQ satisfies CTC*
- Idea: *Start with the alternatives with minimal dominator sets (Copeland winners) and calculate the TEQ-relation backwards until you end up in* ***the*** *TEQ top cycle.*

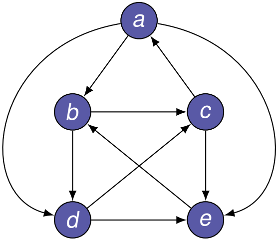
procedure $\mathtt{TEQ}(X)$
$R \leftarrow \emptyset$
$B \leftarrow C \leftarrow$ Copeland set of X
loop
  $R \leftarrow R \cup \{(b, a) : a \in C \text{ and } b \in \mathtt{TEQ}(\overline{D}(a))\}$
  $D \leftarrow \bigcup_{a \in C} \mathtt{TEQ}(\overline{D}(a))$
  if $D \subseteq B$ **then return** $TC_B(R)$ **end if**
  $C \leftarrow D$
  $B \leftarrow B \cup C$
end loop



A Heuristic for Computing TEQ

- Recursive definition of TEQ suggests an exponential naive algorithm
- Naive algorithm can be improved upon by *assuming TEQ satisfies CTC*
- Idea: *Start with the alternatives with minimal dominator sets (Copeland winners) and calculate the TEQ-relation backwards until you end up in* ***the*** *TEQ top cycle.*

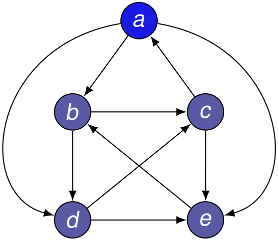
x	$\overline{D}(x)$	$TEQ(\overline{D}(x))$
a	$\{c\}$	$\{c\}$
b	$\{a, e\}$	$\{a\}$
c	$\{b, d\}$	$\{b\}$
d	$\{a, b\}$	$\{a\}$
e	$\{a, c, d\}$	$\{a, c, d\}$



A Heuristic for Computing TEQ

- Recursive definition of TEQ suggests an exponential naive algorithm
- Naive algorithm can be improved upon by *assuming TEQ satisfies CTC*
- Idea: *Start with the alternatives with minimal dominator sets (Copeland winners) and calculate the TEQ-relation backwards until you end up in* ***the*** *TEQ top cycle.*

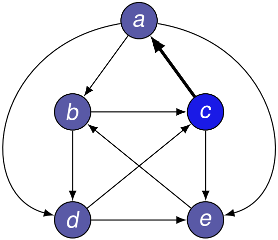
x	$\overline{D}(x)$	$TEQ(\overline{D}(x))$
a	$\{c\}$	$\{c\}$
b	$\{a,e\}$	$\{a\}$
c	$\{b,d\}$	$\{b\}$
d	$\{a,b\}$	$\{a\}$
e	$\{a,c,d\}$	$\{a,c,d\}$



A Heuristic for Computing TEQ

- Recursive definition of TEQ suggests an exponential naive algorithm
- Naive algorithm can be improved upon by *assuming TEQ satisfies CTC*
- Idea: *Start with the alternatives with minimal dominator sets (Copeland winners) and calculate the TEQ-relation backwards until you end up in* ***the*** *TEQ top cycle.*

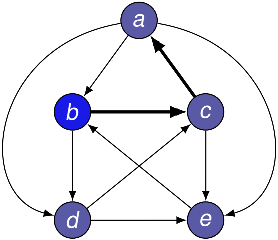
x	$\overline{D}(x)$	$TEQ(\overline{D}(x))$
a	$\{c\}$	$\{c\}$
b	$\{a, e\}$	$\{a\}$
c	$\{b, d\}$	$\{b\}$
d	$\{a, b\}$	$\{a\}$
e	$\{a, c, d\}$	$\{a, c, d\}$



A Heuristic for Computing TEQ

- Recursive definition of TEQ suggests an exponential naive algorithm
- Naive algorithm can be improved upon by *assuming TEQ satisfies CTC*
- Idea: *Start with the alternatives with minimal dominator sets (Copeland winners) and calculate the TEQ-relation backwards until you end up in* ***the*** *TEQ top cycle.*

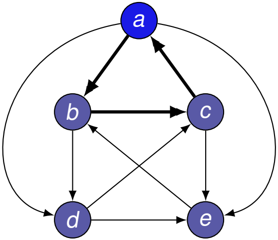
x	$\overline{D}(x)$	$TEQ(\overline{D}(x))$
a	$\{c\}$	$\{c\}$
b	$\{a, e\}$	$\{a\}$
c	$\{b, d\}$	$\{b\}$
d	$\{a, b\}$	$\{a\}$
e	$\{a, c, d\}$	$\{a, c, d\}$



A Heuristic for Computing TEQ

- Recursive definition of TEQ suggests an exponential naive algorithm
- Naive algorithm can be improved upon by *assuming TEQ satisfies CTC*
- Idea: *Start with the alternatives with minimal dominator sets (Copeland winners) and calculate the TEQ-relation backwards until you end up in* ***the*** *TEQ top cycle.*

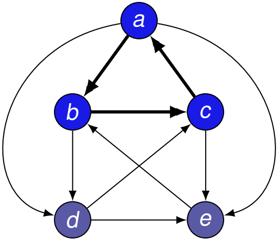
x	$\overline{D}(x)$	$TEQ(\overline{D}(x))$
a	$\{c\}$	$\{c\}$
b	$\{a, e\}$	$\{a\}$
c	$\{b, d\}$	$\{b\}$
d	$\{a, b\}$	$\{a\}$
e	$\{a, c, d\}$	$\{a, c, d\}$



A Heuristic for Computing TEQ

- Recursive definition of TEQ suggests an exponential naive algorithm
- Naive algorithm can be improved upon by *assuming TEQ satisfies CTC*
- Idea: *Start with the alternatives with minimal dominator sets (Copeland winners) and calculate the TEQ-relation backwards until you end up in* ***the*** *TEQ top cycle.*

x	$\overline{D}(x)$	$TEQ(\overline{D}(x))$
a	$\{c\}$	$\{c\}$
b	$\{a, e\}$	$\{a\}$
c	$\{b, d\}$	$\{b\}$
d	$\{a, b\}$	$\{a\}$
e	$\{a, c, d\}$	$\{a, c, d\}$



Experimental Results: Evaluation of the Heuristic

|A|	Floyd-Warshall	Kosaraju	Algorithm 1
50	0.48 s	0.59 s	0.09 s
100	53.33 s	65.73 s	9.57 s
150	1 166 s	1 429 s	210 s

Uniform random tournaments ($p = 0.5$)

|A|	Floyd-Warshall	Kosaraju	Algorithm 1
50	13.87 s	16.56 s	0.01 s
100	18 416 s	21 382 s	8.46 s
150	—	—	1273 s

Structured random tournaments ($p = 0.8$)

- Two versions of naive algorithm depending on transitive closure subalgorithm
- Floyd-Warshall slightly outperforms Kosaraju despite worse asymptotic complexity
  - Hidden constants are amplified as consequence of TEQ's recursive definition
- Our heuristic outperforms naive algorithm by factor five on uniform tournaments
  - Dramatically faster on structured tournaments than naive algorithm

Searching for Counterexamples to Schwartz's Conjecture

$\|A\|$	no. of non-isomorphic tournaments on A
1	1
2	1
3	2
4	4
5	12
6	56
7	456
8	6 880
9	191 536
10	9 733 056
11	903 753 248
12	154 108 311 168
n	$\approx \frac{2^{\binom{n}{2}}}{n!}$

Result: *Exhaustive search of all tournaments up to 10 alternatives revealed no counterexample to Schwartz's conjecture. (Testing for 11 alternatives using a list of non-isomorphic tournaments (42GB) provided by Brendan McKay in progress)*

Conclusion

- Attractiveness of TEQ dependent on Schwartz's conjecture
- Deciding TEQ membership is NP-hard
- Heuristic significantly improves on naive algorithm
- So far no counterexample for Schwartz's conjecture found by:
  - random sampling among millions of tournaments
  - exhaustive search in tournaments up to 11 alternatives