On the Fixed-Parameter Tractability of Composition-Consistent Tournament Solutions

Hans Georg Seedig

COMSOC 2010
August 14, 2010

Joint work with Felix Brandt and Markus Brill

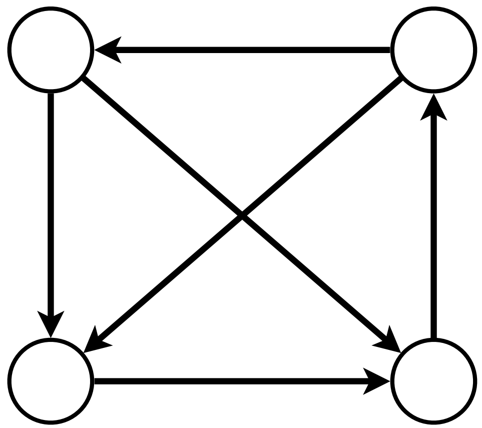
PREFERENCE AGGREGATION IN MULTIAGENT SYSTEMS

Overview

- Tournaments
  - Components and Decomposition
- Tournament Solutions
  - Composition Consistency
- Parametrized Complexity
  - Fixed-parameter tractability
- Algorithm
- Experiments

Tournaments

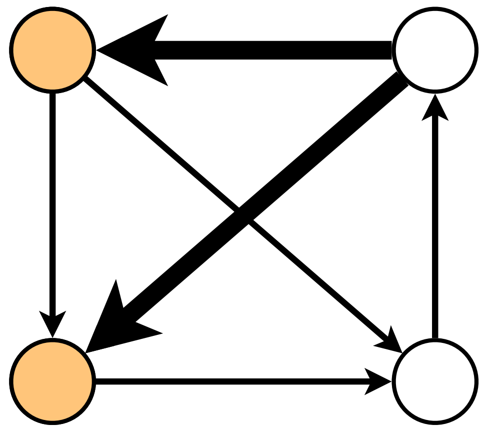
- T=(A, >) is a tournament
  - A is a finite set of candidates or alternatives
  - > is an asymmetric and complete binary relation on the alternatives
    - a > b means 'a dominates b' or 'a is preferred over b'
    - pairwise majority outcome of an election
    - > may be cyclic



- Corresponds to complete oriented graph

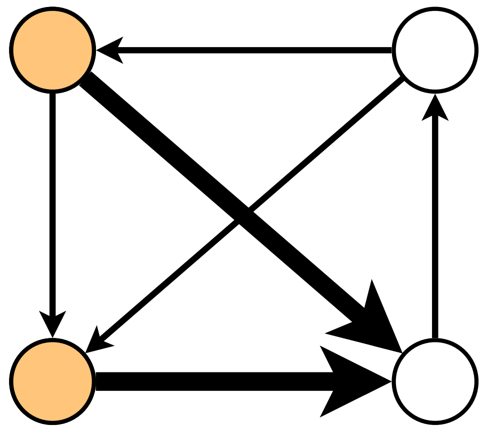
Components in Tournaments

- Alternatives in a tournament form a component if they bear the same relationship to all outside alternatives



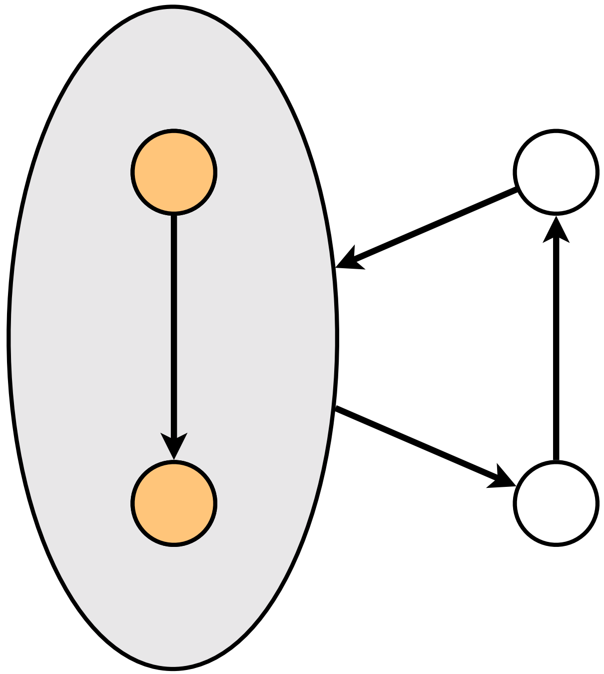
Components in Tournaments

- Alternatives in a tournament form a component if they bear the same relationship to all outside alternatives



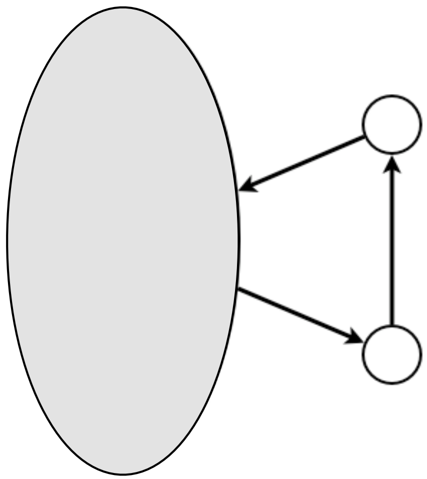
Components in Tournaments

- Alternatives in a tournament form a component if they bear the same relationship to all outside alternatives



Decompositions

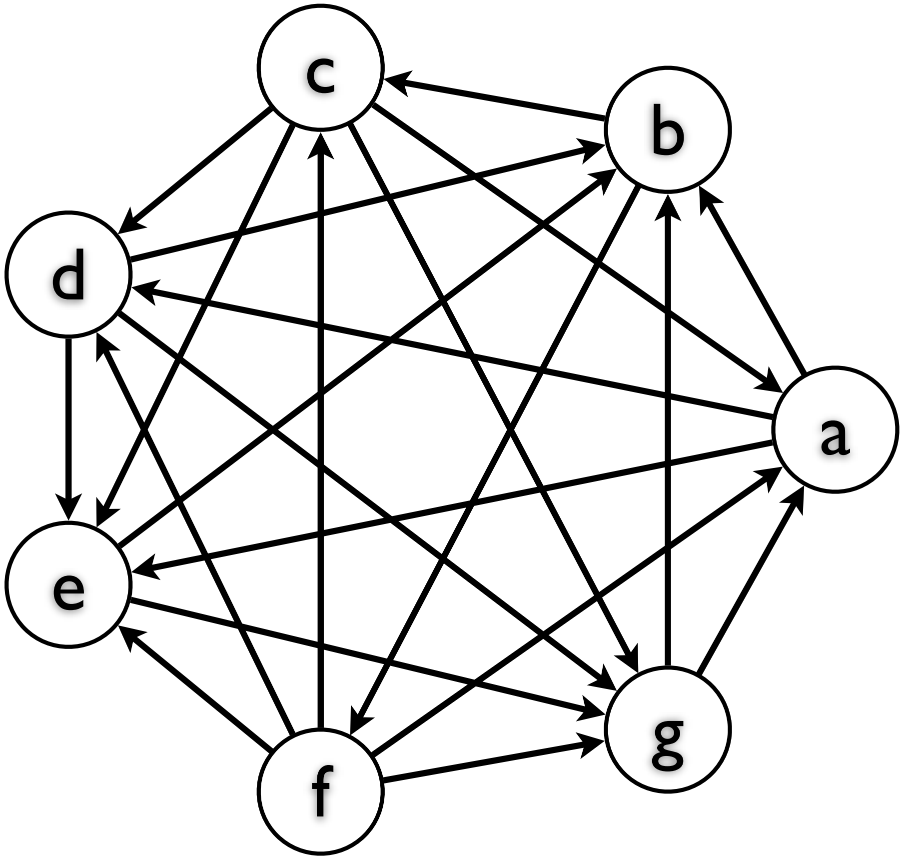
- A graph can be decomposed into components
- A decomposition of T=(A, >) is a set of pairwise disjoint components $\{B_1, B_2, \dots, B_k\}$ such that $\cup B_i = A$
- The summary of T w.r.t this decomposition is the tournament on the components Ť, induced by T.



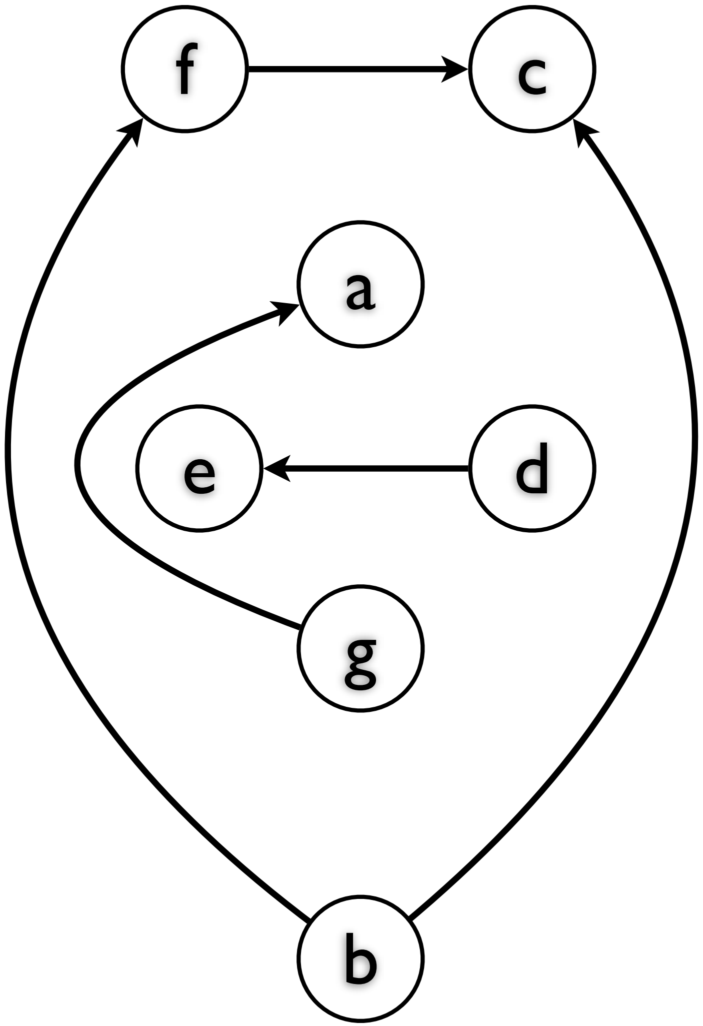
Decomposition Tree

- There is a unique minimal decomposition
- A component may be decomposable again
- Represent this recursive decompositions as a decomposition tree
- The decomposition degree δ is the maximum degree in the decomposition tree

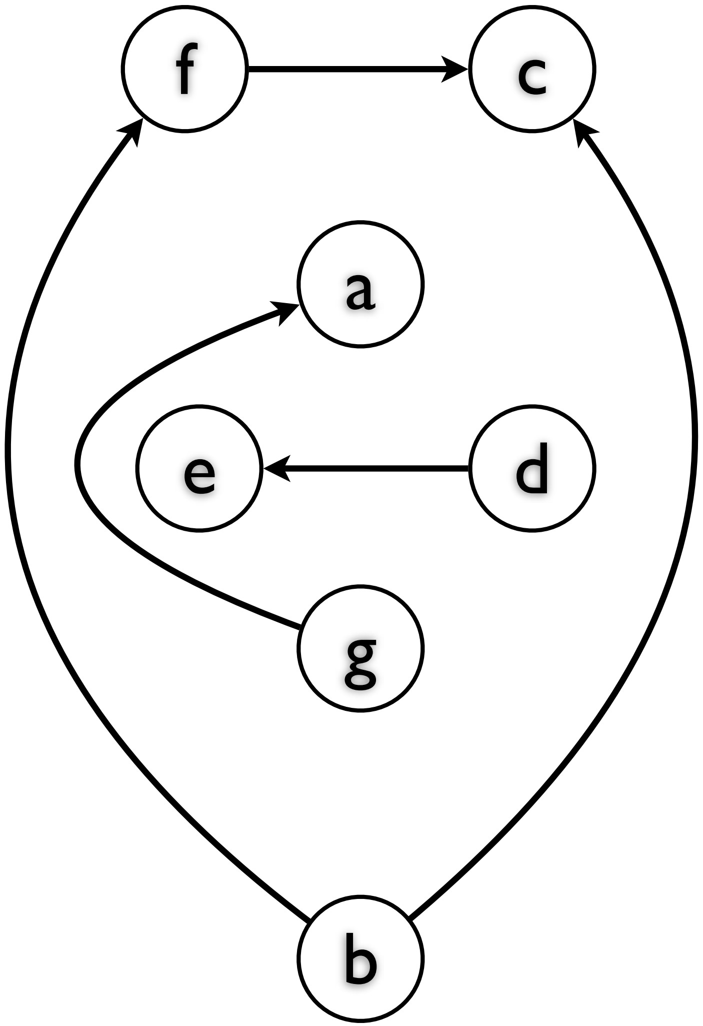
Example: Decomposition Tree

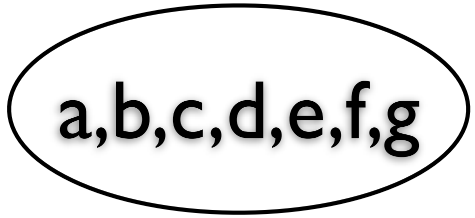


Example: Decomposition Tree

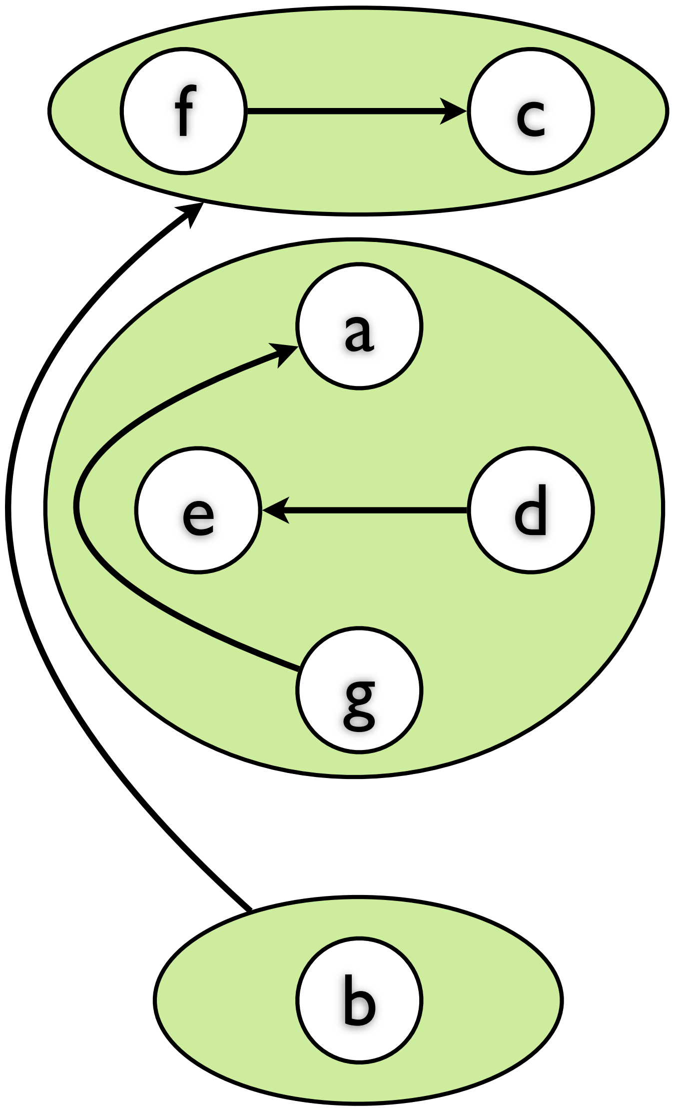


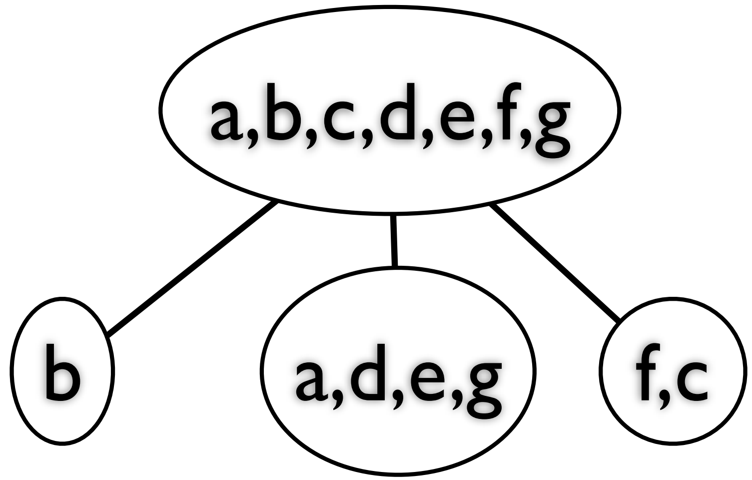
Example: Decomposition Tree



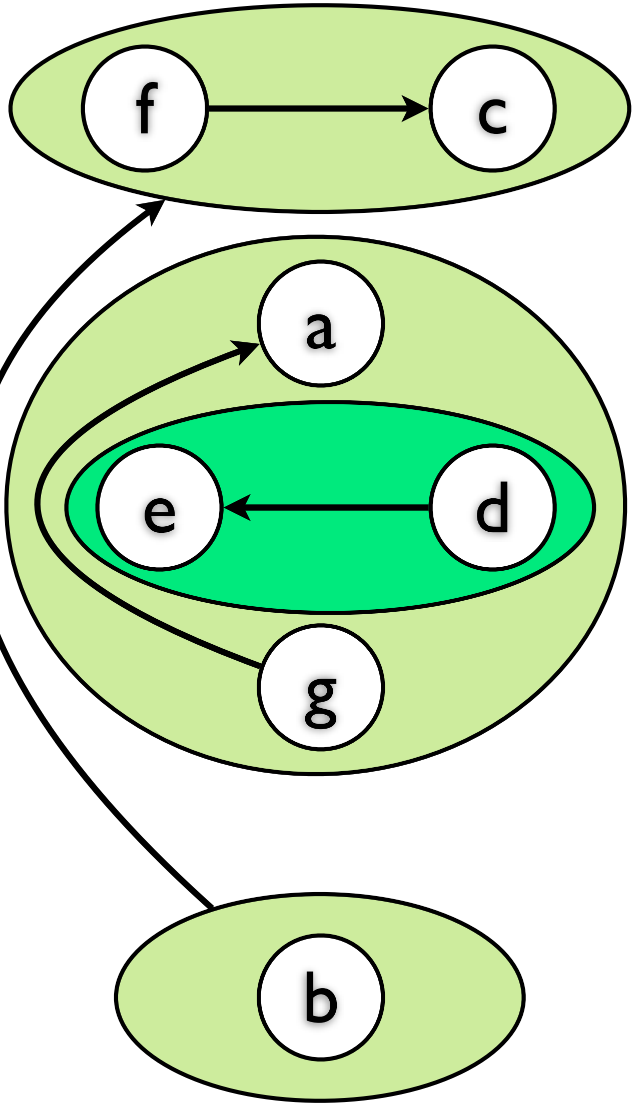


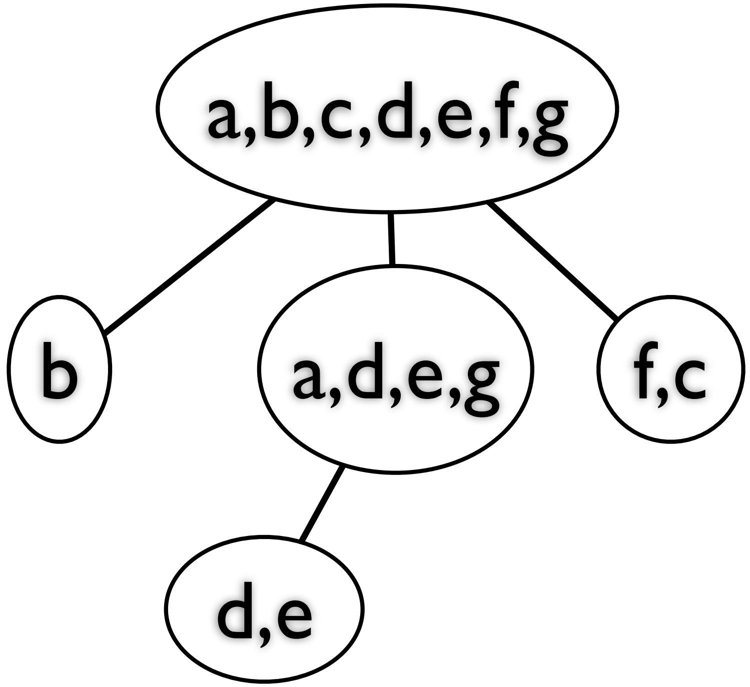
Example: Decomposition Tree



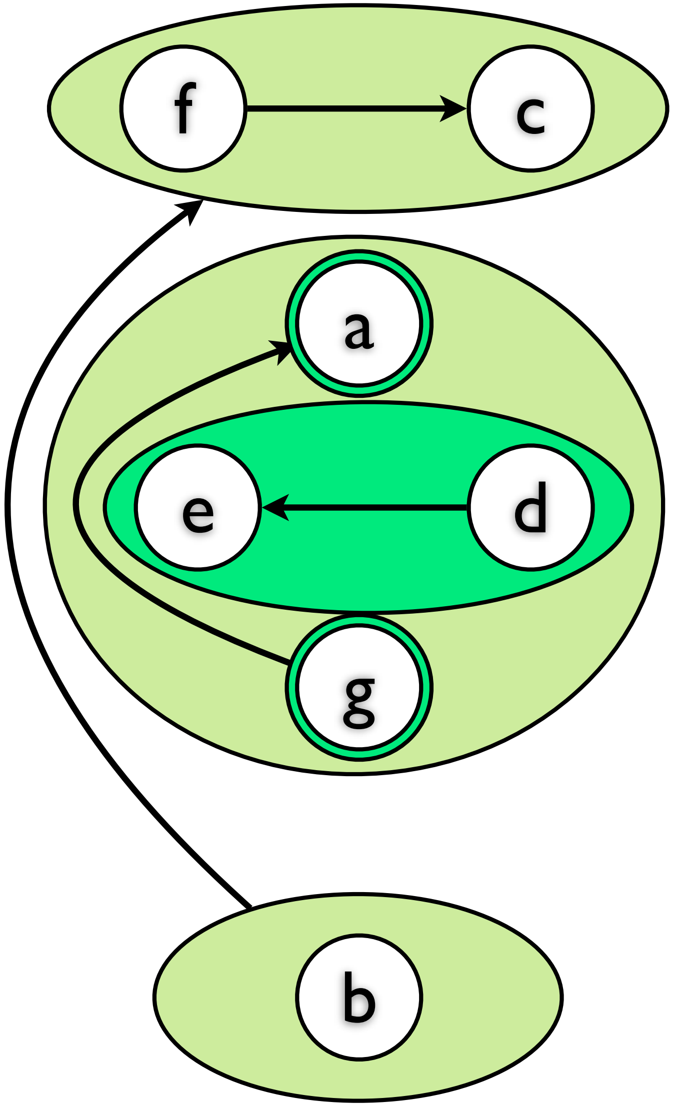


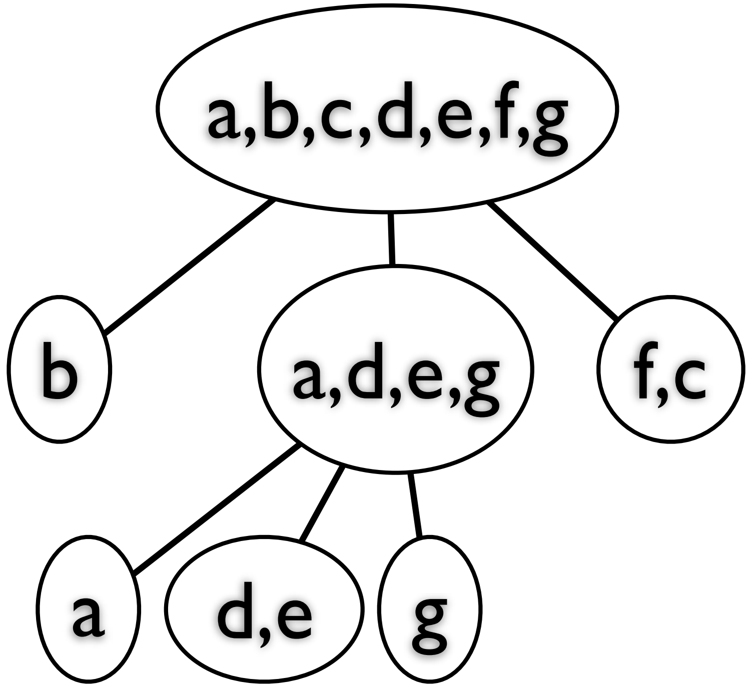
Example: Decomposition Tree



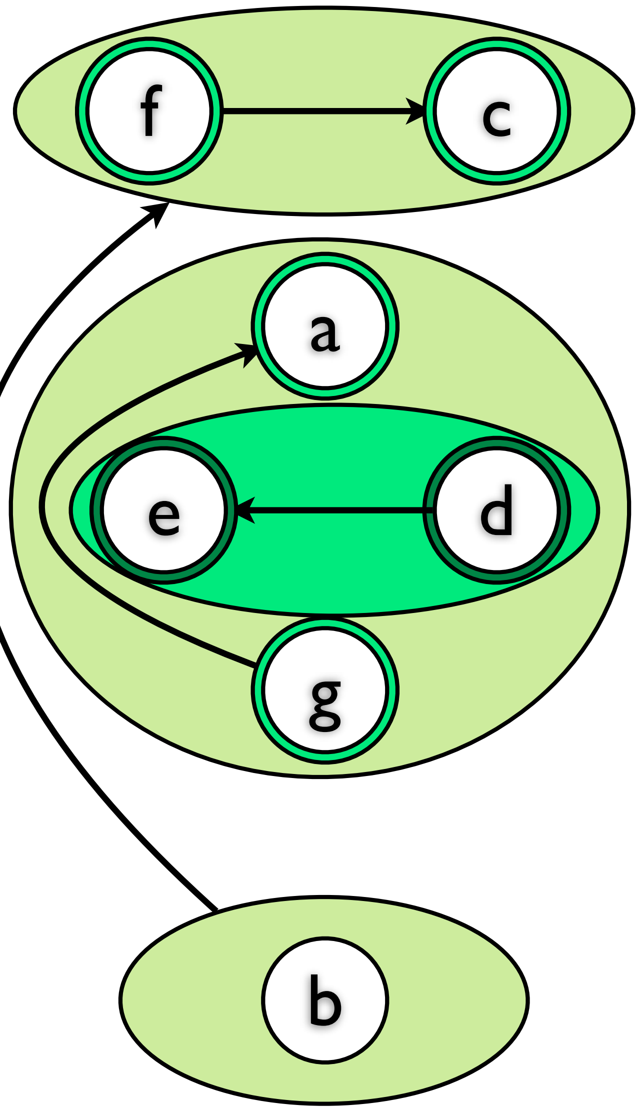


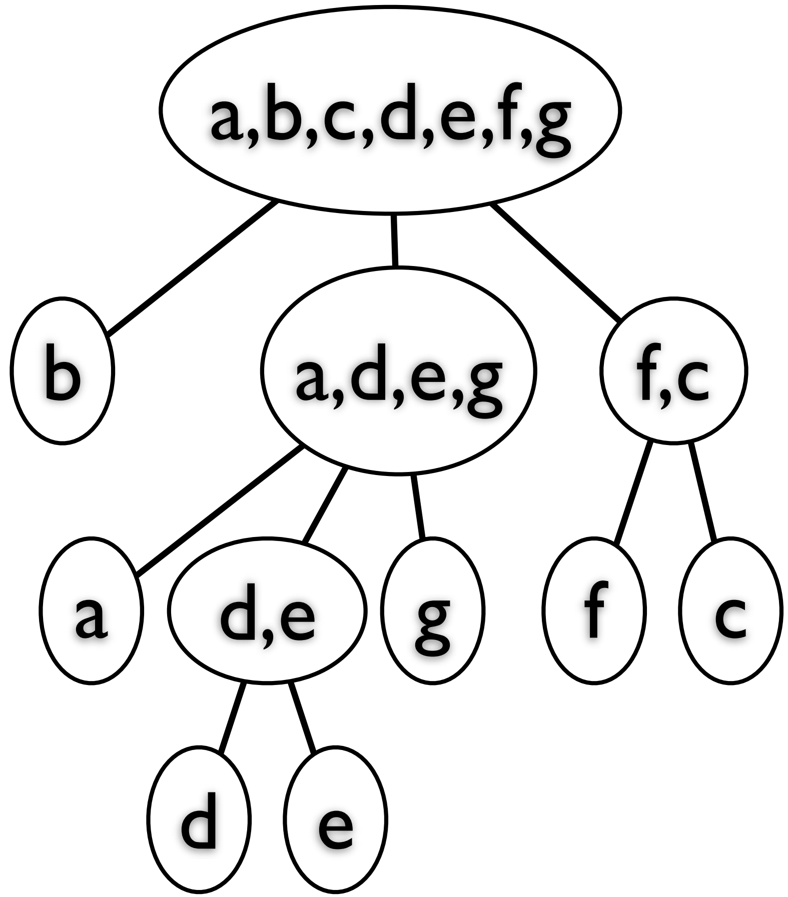
Example: Decomposition Tree



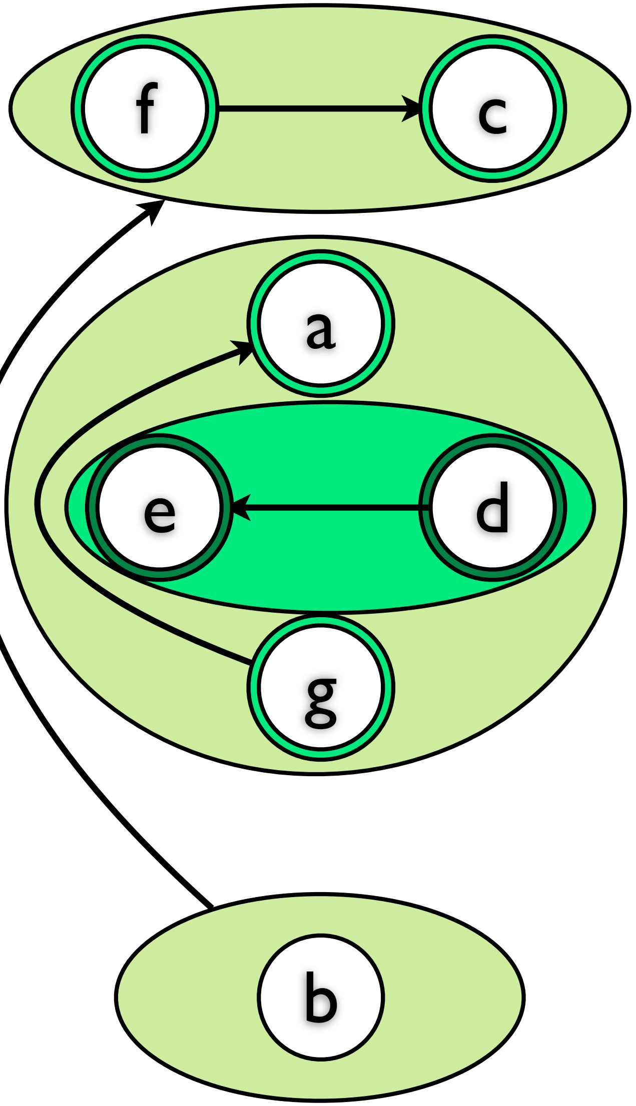


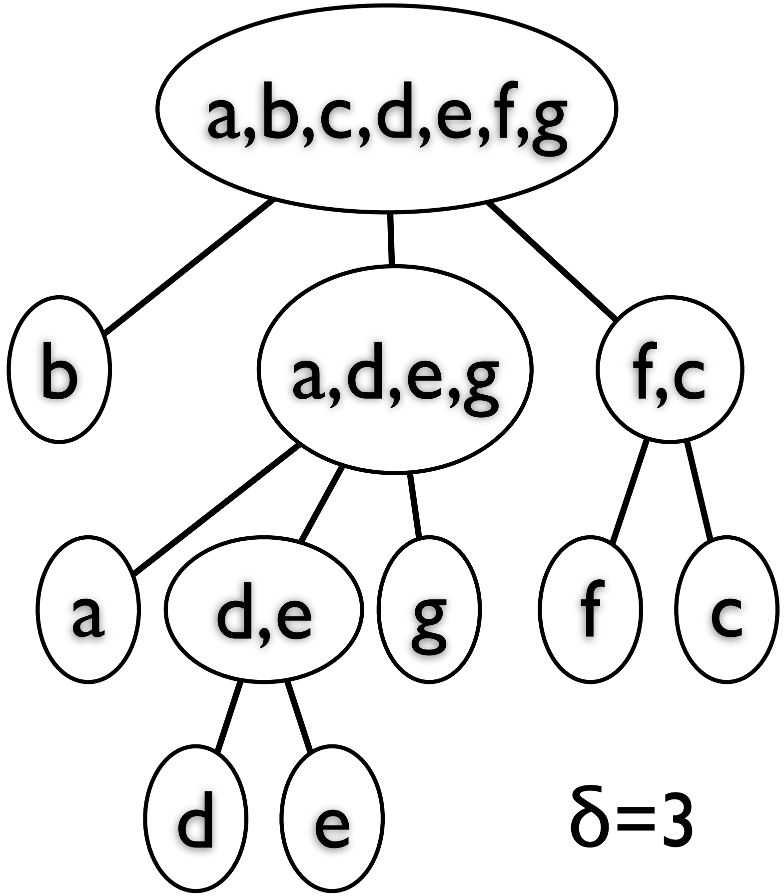
Example: Decomposition Tree





Example: Decomposition Tree





$\delta=3$

- Decomposition degree δ is max. degree in decomp. tree

Tournament Solutions

- Given a tournament, what is the set of winners?
- Intuitively easy if one alternative c dominates all others
  - c is a Condorcet winner
  - does not exist in most tournaments
- A tournament solution S returns a non-empty subset of A, i.e., $S(T) \subseteq A$
- Many solution concepts have been proposed in the past
- Axiomatic approach: Do they have desirable properties?

Zoo of Tournament Solutions

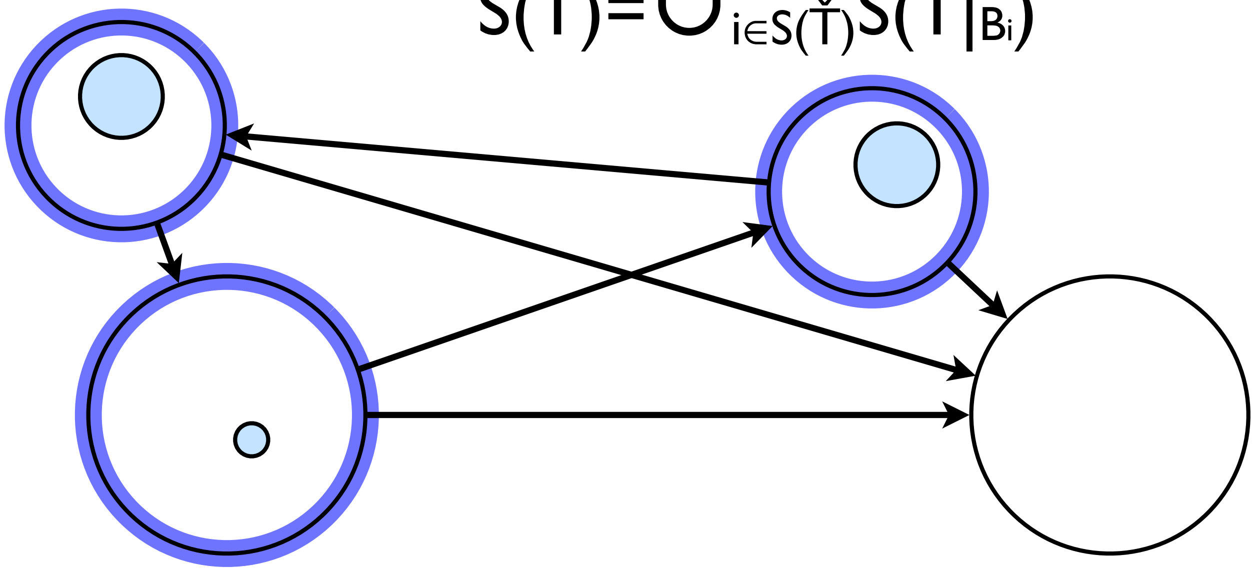
- Copeland set
- Slater set
- Banks set
- Uncovered Set
- Minimal Covering Set (MC)
- Bipartisan Set (BP)
- TEQ

- Many tournament solutions are computationally hard
  - Slater, Banks and TEQ are NP-hard. MC and BP are in P but existing algorithms rely on linear programming and are thus rather inefficient.
  - All of these except Copeland and Slater satisfy composition-consistency.

Composition-Consistency

- A tournament solution S is composition-consistent if it chooses the 'best' alternatives from the 'best' components.
- Formally: S is composition-consistent if for all T, $\check{T}$ summary of T w.r.t. some decomposition $\{B_1,\ldots,B_k\}$

$$S(T)=\bigcup_{i\in S(\check{T})} S(T|_{B_i})$$



Fixed-Parameter Tractability

- Use parametrized complexity to analyze whether the hardness of a problem depends on the size of a certain parameter
- Consider a problem with parameter k fixed-parameter tractable (FPT) if there is an algorithm that solves it in time $f(k) \cdot poly(InputLength)$ where f is independent of the input length

Algorithm

1. Compute the decomposition tree
2. Recursively compute tournament solution on components

- Decomposition tree computable in linear time!
  - Follows from results by McConnell and de Montgolfier (2005); Capelle et al. (2002) on modular decomposition of directed graphs
- Number of tournaments to solve is bounded by |A|-1
- Size of the largest tournament to solve equals the decomposition degree

Main Result

Given composition-consistent tournament solution S where computing S(T) with $|T| \leq k$ takes time $\leq f(k)$

Then, S(T) can be computed in $O(|T|^2) + f(\delta(T)) \cdot (|T|-1)$

compute decomposition tree

worst-case time for solving a tournament

max. no. of tournaments

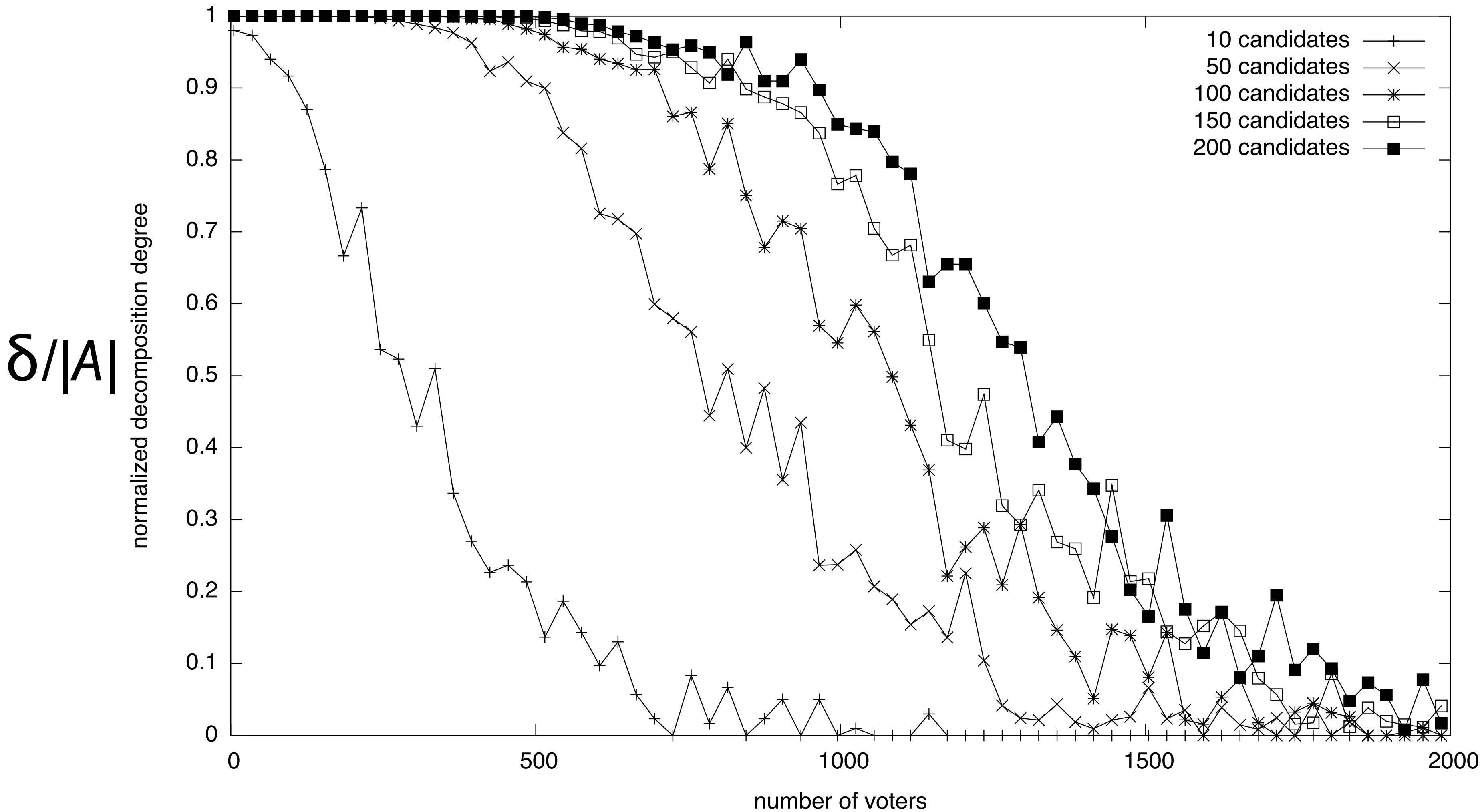
<u>Corollary</u>

Computing S(T) is fixed-parameter tractable w.r.t. $\delta(T)$.

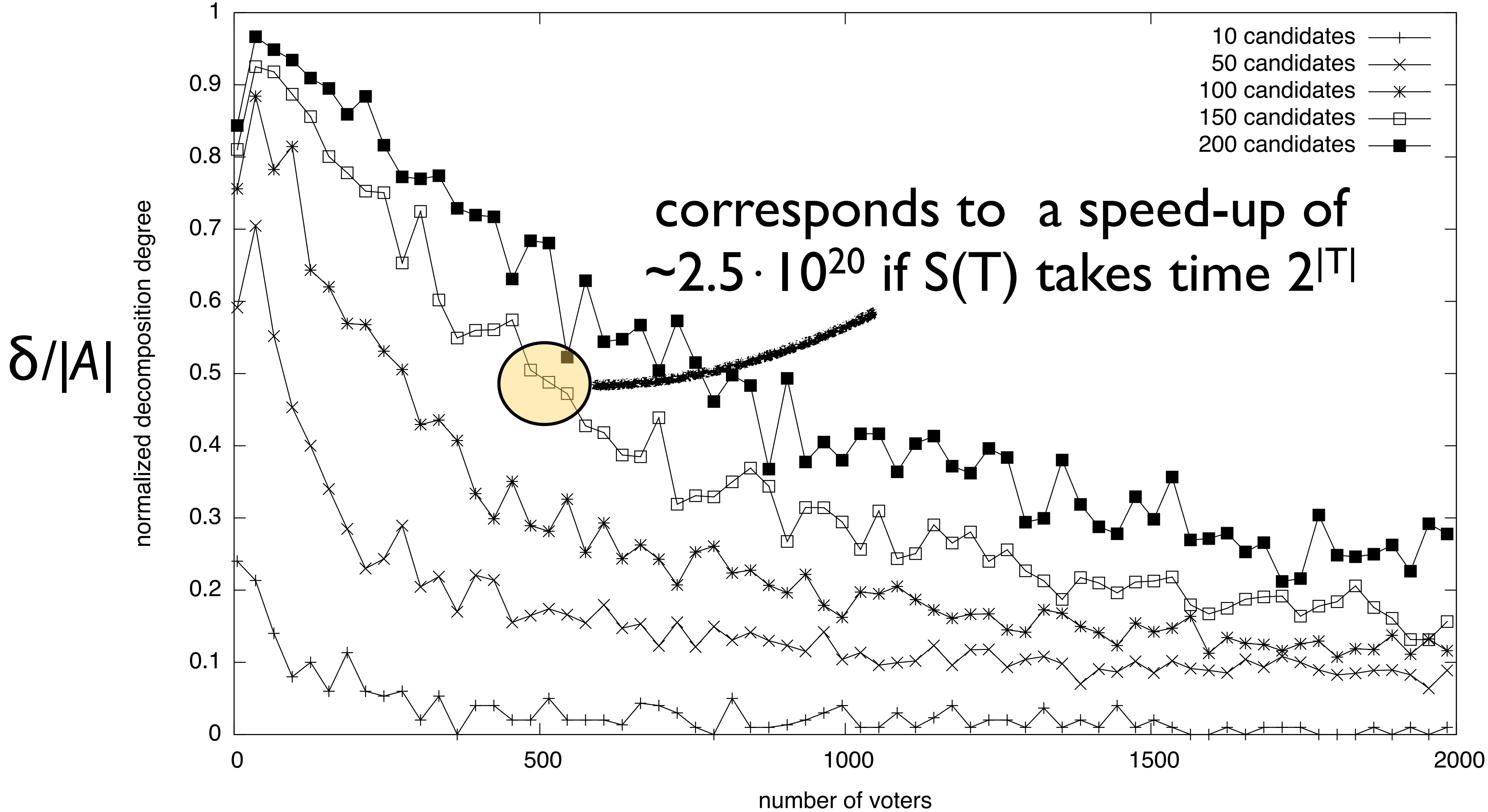
Experiments

- Generate majority tournaments according to voting models
  - Noise model: Voters give "correct" ranking of each pair of alternatives with probability $p > \frac{1}{2}$
  - Spatial model: Alternatives and voters are located in $[0,1]^d$. Preferences according to Euclidian distances between voters and alternatives.
- a > b iff a majority prefers a to b

Noise model with p=0.55

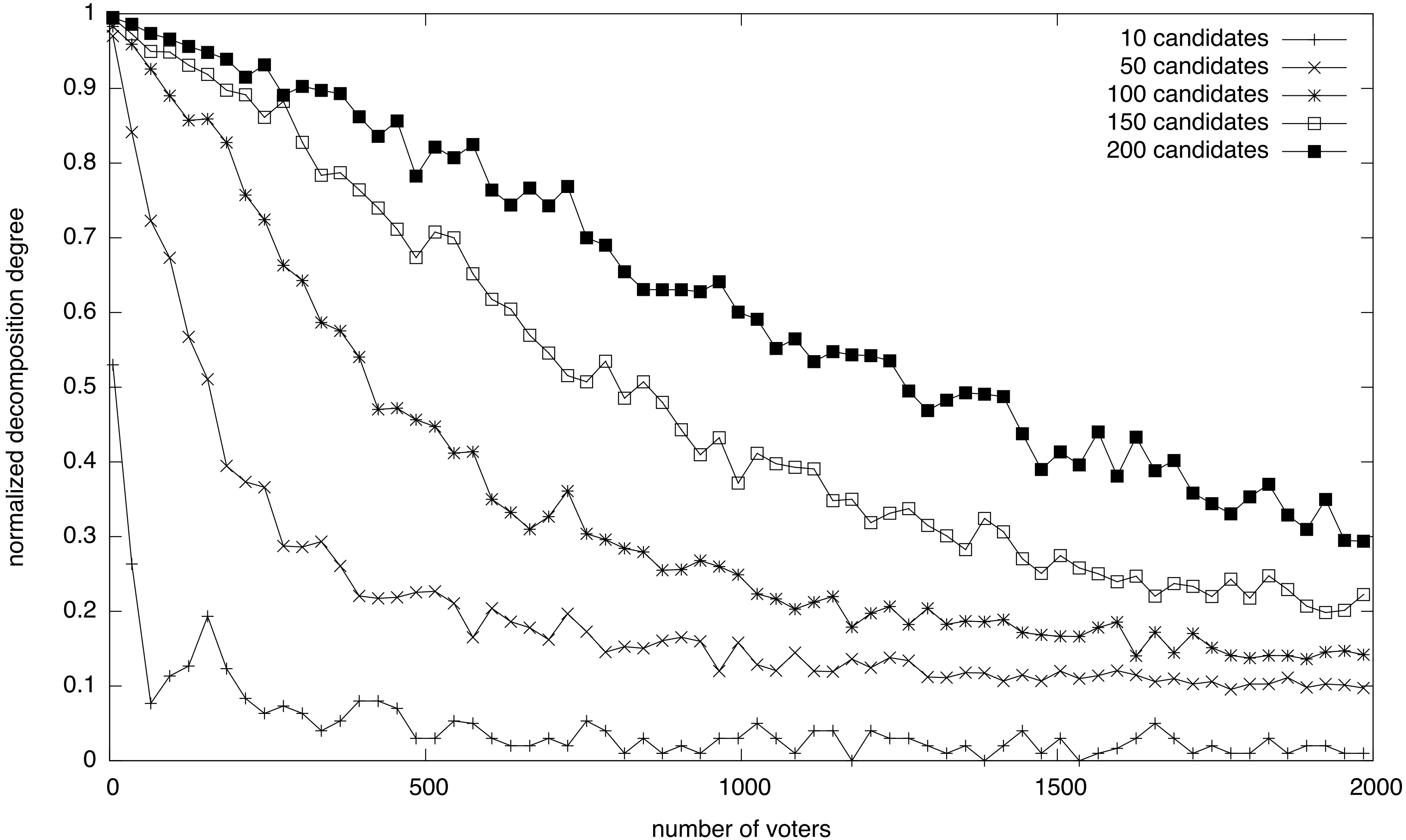


Spatial model with d=2



Spatial model with d=20

$\delta/|A|$



Conclusion

- Exploiting composition-consistency can lead to dramatical speed ups in algorithms for tournament solutions
- All tournament solutions satisfying composition-consistency are fixed-parameter tractable w.r.t. the decomposition degree
- $\delta = O(\log^k |A|)$ for some k allows polynomial-time algorithms for tournament solutions that in general only admit algorithms of time $O(2^n)$
- Future work
  - Measure positive effect by actual computation of composition-consistent tournament solutions.
  - Use parallelization and lookup tables.