On Problem Kernels for Possible Winner Determination Under the k-Approval Protocol

Nadja Betzler

Friedrich-Schiller-Universität Jena, Germany

3rd Workshop on Computational Social Choice
September 2010

Motivation

Typical voting scenario for joint decision making:

Voters give preferences over a set of candidates as linear orders.

Example: candidates: $C = \{a, b, c, d\}$

profile:	vote 1:	a	$>$	b	$>$	c	$>$	d
	vote 2:	a	$>$	d	$>$	c	$>$	b
	vote 3:	b	$>$	d	$>$	c	$>$	a

Aggregate preferences according to a voting rule

Partial information

Realistic settings: voters might only provide partial information.

For example:

- not all voters have given their preferences yet
- new candidates are introduced
- a voter cannot compare several candidates because of lack of information

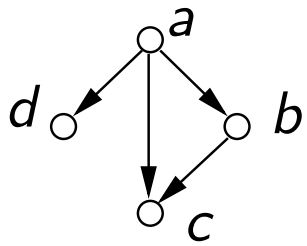
How to deal with partial information?

We consider whether a distinguished candidate can still win.

Partial vote

A **partial vote** is a transitive and antisymmetric relation.

Example: $C = \{a, b, c, d\}$
partial vote: $a \succ b \succ c,\ a \succ d$



possible **extensions**:

1. $a > \mathbf{d} > b > c$
2. $a > b > \mathbf{d} > c$
3. $a > b > c > \mathbf{d}$

An extension of a profile of partial votes extends every partial vote.

Computational Problem

Possible Winner

Input: A voting rule r, a set of candidates C, a profile of partial votes, and a distinguished candidate c.
Question: Is there an extension profile where c wins according to r?

Considered voting rule:

k-approval

In every vote, the best k candidates get one point each. A candidate with most points in total wins.

Known results for Possible Winner

Results for several voting systems, [Konczak and Lang, 2005], [Pini et al., IJCAI 2007], [Walsh, AAAI 2007], [Xia and Conitzer, AAAI 2008], ...

Results for k-approval

- Possible Winner is NP-hard for two (or more) partial votes
  [Betzler, Hemmann, and Niedermeier, IJCAI 2009]
- Possible Winner is NP-hard for any fixed $k \in \{2, \ldots, m-2\}$ for m candidates
  [Betzler and Dorn, JCSS 2010]

Known results for Possible Winner

Results for several voting systems, [Konczak and Lang, 2005], [Pini et al., IJCAI 2007], [Walsh, AAAI 2007], [Xia and Conitzer, AAAI 2008], ...

Results for k-approval

- Possible Winner is NP-hard for two (or more) partial votes
  [Betzler, Hemmann, and Niedermeier, IJCAI 2009]
- Possible Winner is NP-hard for any fixed $k \in \{2, \ldots, m-2\}$ for m candidates
  [Betzler and Dorn, JCSS 2010]

This work

Is the Possible Winner problem easy to compute when the number k of "one-positions" **and** the number of votes is small?

Combined parameters

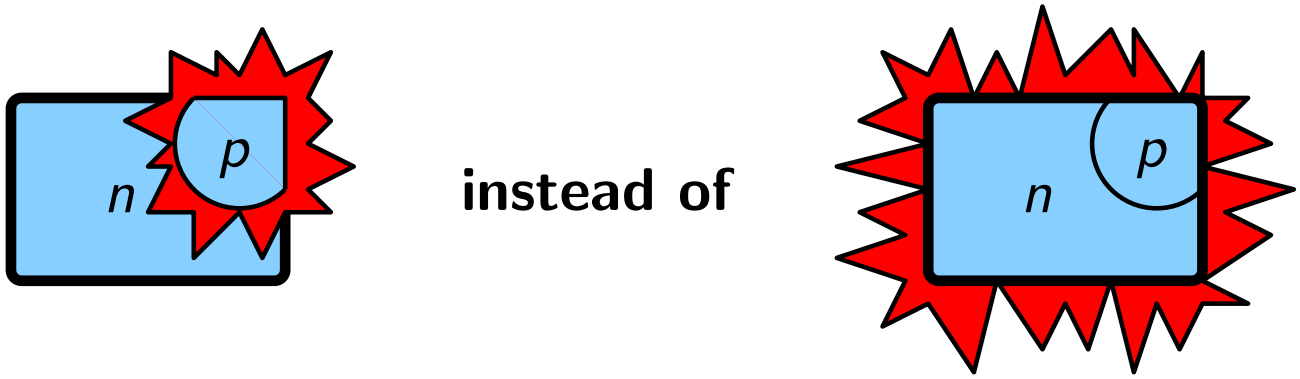
2 scenarios:

- "number of partial votes" and "number of one-positions" k
- "number of partial votes" and "number of zero-positions" k_0

Motivation: Small committee selects few winners/losers out of a large set of candidates (grants, graduate students, ...)

Parameterized Complexity

Given an NP-hard problem with input size n and a parameter p
Basic idea: Confine the combinatorial explosion to p



Definition

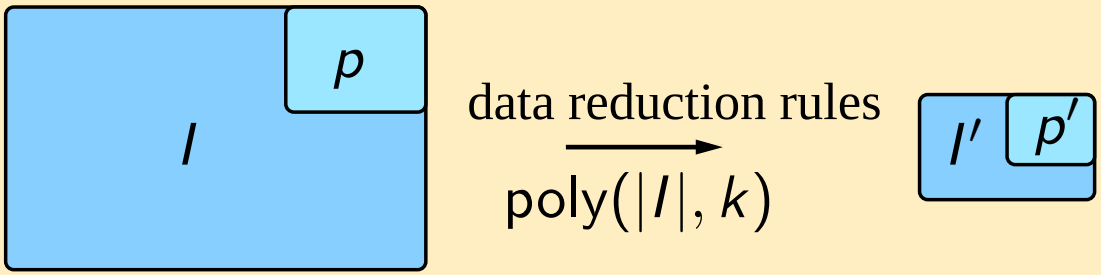
A problem of size n is called *fixed-parameter tractable* with respect to a parameter p if it can be solved in $f(p) \cdot n^{O(1)}$ time.

Parameters: pairs of integers

Problem kernel

Let $L \subseteq \Sigma^* \times \Sigma^*$ be a parameterized problem. An instance of L is denoted by (I, p).

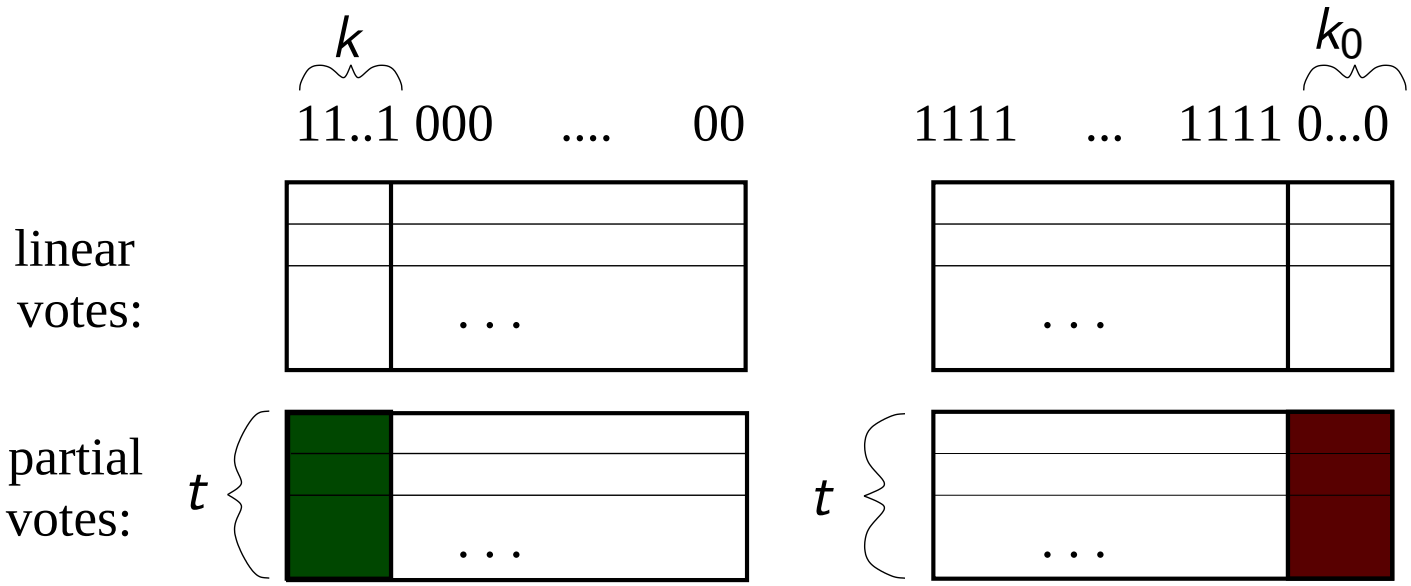
Kernelization



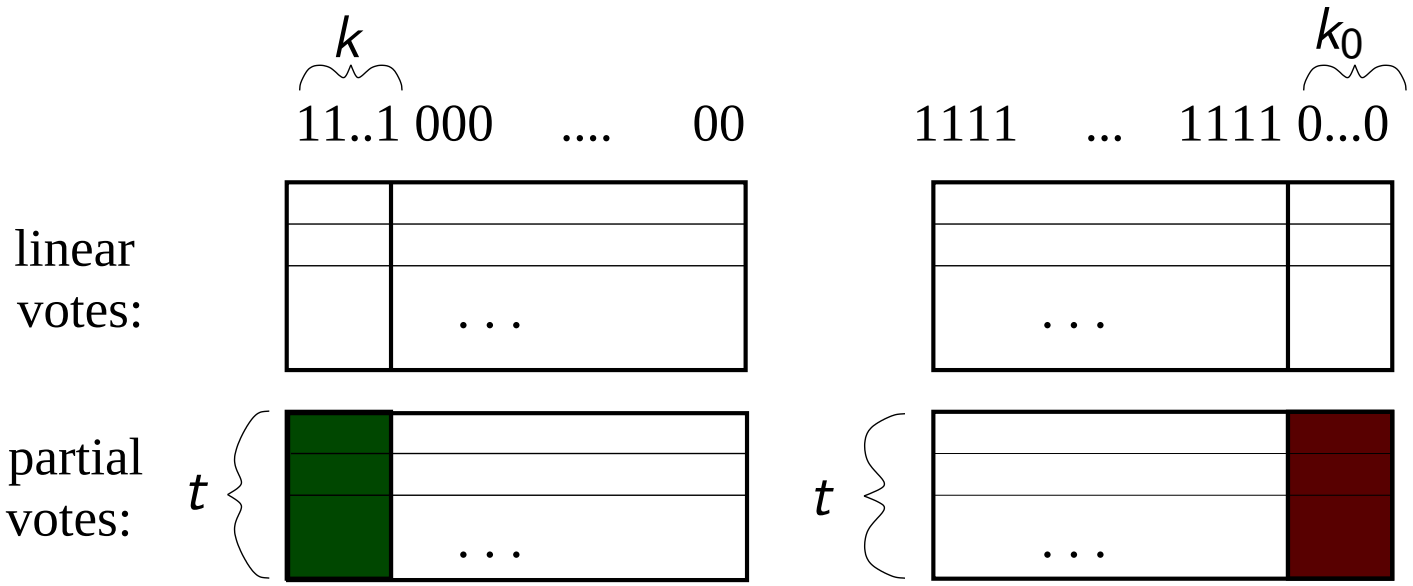
- $(I, p) \in L \iff (I', p') \in L$
- $|p'| \leq |p|$
- $|I'| \leq g(|p|)$

If g is a polynomial, we say L admits a *polynomial problem kernel*.

Main results



Main results



Parameter	(t, k)	(t, k_0)
	FPT	FPT
	"Tower of k's"-kernel	
	no polynomial kernel	**polynomial kernel**

Polynomial kernel for (t, k_0)

1. Fix the distinguished candidate c as good as possible
2. $z(c') :=$ # zero-positions such that c' is beaten by c
3. **If** $\sum_{c' \in C} z(c') > t \cdot k_0$, **then** return "no"
   else replace "irrelevant" candidates by a bounded number.

Example: $C := \{a, b, c, d_1, \ldots, d_s\}$, $k_0 = 2$

candidate	points in linear votes
d_i	≤ 10
b	11
a	12
c	12

partial votes:

$v_1 : a \succ c,\ b \succ d_1,\ d_2 \succ d_3$

$v_2 : d_1 \succ d_2 \succ \cdots \succ d_s \succ b \succ c$

$v_3 : d_s \succ c,\ a \succ d_2$

Polynomial kernel for (t, k_0)

1. Fix the distinguished candidate c as good as possible
2. $z(c') :=$ # zero-positions such that c' is beaten by c
3. **If** $\sum_{c' \in C} z(c') > t \cdot k_0$, **then** return "no"
   else replace "irrelevant" candidates by a bounded number.

Example: $C := \{a, b, c, d_1, d_2, \ldots, d_s\}$, $k_0 = 2$

candidate	points in linear votes
d_i	≤ 10
b	11
a	12
c	12

partial votes:

$v_1 : a \succ c,\ b \succ d_1,\ d_2 \succ d_3,\ c \succ C \setminus \{c, a\} \quad \Rightarrow a > c > \ldots$

$v_2 : d_1 \succ d_2 \succ \cdots \succ d_s \succ b \succ c$

$v_3 : d_s \succ c,\ a \succ d_2,\ c \succ C \setminus \{c, d_s\} \quad \Rightarrow d_s > c > \ldots$

Polynomial kernel for (t, k_0)

1. Fix the distinguished candidate c as good as possible
2. $z(c') :=$ # zero-positions such that c' is beaten by c
3. **If** $\sum_{c' \in C} z(c') > t \cdot k_0$, **then** return "no"
   else replace "irrelevant" candidates by a bounded number.

Example: $C := \{a, b, c, d_1, d_2, \ldots, d_s\}$, $k_0 = 2$

candidate	points in linear votes	# zero-positions
d_i	≤ 10	0
b	11	1
a	12	2
c	12	—

partial votes:

$v_1 : a \succ c,\ b \succ d_1,\ d_2 \succ d_3,\ c \succ C \setminus \{c, a\}$

$v_2 : d_1 \succ d_2 \succ \cdots \succ d_s \succ b \succ c$

$v_3 : d_s \succ c,\ a \succ d_2,\ c \succ C \setminus \{c, d_s\}$

Polynomial kernel for (t, k_0)

1. Fix the distinguished candidate c as good as possible
2. $z(c') :=$ # zero-positions such that c' is beaten by c
3. **If** $\sum_{c' \in C} z(c') > t \cdot k_0$, **then** return "no"
   else replace "irrelevant" candidates by a bounded number.

Example: $C := \{a, b, c, \mathbf{d}, \cancel{d_1, d_2 \ldots, d_s}\}$, $k_0 = 2$

candidate	points in linear votes	# zero-positions
d_i	≤ 10	0
b	11	1
a	12	2
c	12	—

partial votes:

$v_1 : a \succ c,\ b \succ d_1,\ d_2 \succ d_3,\ c \succ C \setminus \{c, a\} \quad \Rightarrow\ a \succ c \succ b \succ d$

$v_2 : d_1 \succ d_2 \succ \cdots \succ d_s \succ b \succ c \quad \Rightarrow\ d \succ b \succ c$

$v_3 : d_s \succ c,\ a \succ d_2,\ c \succ C \setminus \{c, d_s\} \quad \Rightarrow\ c \succ a \succ d$

Polynomial kernels

Theorem

For k-approval, Possible Winner with t partial votes and k_0 zero-positions admits a polynomial kernel with $O(t \cdot k_0^2)$ candidates.

Crucial idea: Number of candidates that **have to** take **zero-positions** is bounded in a yes-instance.

Why does this not work for (t, k)?

Polynomial kernels

Theorem

For k-approval, Possible Winner with t partial votes and k_0 zero-positions admits a polynomial kernel with $O(t \cdot k_0^2)$ candidates.

Crucial idea: Number of candidates that **have to** take **zero-positions** is bounded in a yes-instance.

Why does this not work for (t, k)?

In a yes-instance, there might be an unbounded number of candidates that **may** take a **one-position**.

Theorem

For k-approval, Possible Winner with t partial votes does not admit a polynomial kernel wrt. (t, k) unless coNP $\subseteq$ NP/poly.

Overview of kernelization results

Possible Winner for k-approval with t partial votes

k_0 denotes the number of zero-positions

(t, k_0)	(t, k)
polynomial kernel (FPT)	superexponential kernel (FPT) no polynomial kernel 2-approval : polynomial kernel with $O(t^2)$ candidates

Future work

- Counting variant: In how many extensions does a distinguished candidate win?
  Some first results in
  [Bachrach, Betzler, and Faliszewski, AAAI 2010]
- Can the results from this work be transferred to other voting rules.
- kernelization for related problems

Future work

- Counting variant: In how many extensions does a distinguished candidate win?
  Some first results in
  [Bachrach, Betzler, and Faliszewski, AAAI 2010]
- Can the results from this work be transferred to other voting rules.
- kernelization for related problems

Thank you