Towards a Dichotomy of Finding Possible Winners in Elections Based on Scoring Rules

Nadja Betzler[1]

joint work with

Britta Dorn[2]

[1]Friedrich-Schiller-Universität Jena, Germany
[2]Eberhard-Karls-Universität Tübingen, Germany

MFCS 2009

Motivation

Voting scenarios arise in many situations:

- political elections
- committees: decisions about job applicants
- web site rankings
- recommender systems
- choice of restaurant
- ...

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

Kind of voting rules considered in this work: **Scoring rules**

Scoring rules

Examples:

- plurality: $(1, 0, \ldots, 0)$
- 2-approval: $(1, 1, 0, \ldots, 0)$
- veto: $(1, \ldots, 1, 0)$
- Borda: $(m-1, m-2, \ldots, 0)$
- Formula 1 scoring: $(10, 8, 6, 5, 4, 3, 2, 1, 0, \ldots, 0)$

For m candidates, a scoring vector $(\alpha_1, \alpha_2, \ldots, \alpha_m)$ with $\alpha_1 \geq \alpha_2 \geq \cdots \geq \alpha_m$ and $\alpha_m = 0$ provides a scoring value for every position.

The scoring values of every candidate are summed up and the candidate with the highest score wins.

Scoring rules

m candidates: scoring vector $(\alpha_1, \alpha_2, \dots, \alpha_m)$ with
$\alpha_1 \geq \alpha_2, \geq \dots \geq \alpha_m$ and $\alpha_m = 0$

Scoring rule

provides a scoring vector for every number of candidates.

- non-trivial: $\alpha_1 \neq 0$
- pure: the scoring vector for i candidates can be obtained from the scoring vector for $i - 1$ candidates by inserting an additional score value at an arbitrary position

Example:

3 candidates: $(6, 3, 0)$

4 candidates: pure: $(6, 3, 3, 0)$, $(6, 5, 3, 0)$, $(8, 6, 3, 0)$, ...
not pure: $(6, 6, 0, 0)$

Partial information

Recall: In the typical model, votes need to be presented as linear orders.

Realistic settings: voters may 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

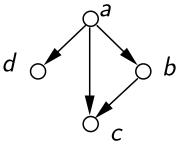
How to deal with partial information?

We consider the question if 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. $a > d > b > c$
2. $a > b > d > c$
3. $a > b > c > d$

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?

The NECESSARY WINNER problem asks for a candidate that wins in every extension.

For scoring rules, NECESSARY WINNER can be solved in polynomial time [XIA AND CONITZER, AAAI 2008] for unweighted voters.

Known results for scoring rules

Two studied scenarios for POSSIBLE WINNER:

1. weighted voters:
   Dichotomy for the special case of MANIPULATION
   [HEMASPAANDRA AND HEMASPAANDRA, JCSS 2007]
   $\Rightarrow$ NP-complete for all scoring rules except plurality (holds even for a constant number of candidates)
2. unweighted voters:
   a) constant number of candidates: always polynomial time
   [CONITZER, SANDHOLM, AND LANG, JACM 2007]
   b) unbounded number of candidates:

Known results for scoring rules

unweighted voters + unbounded number of candidates:

- NP-complete for scoring rules that fulfill the following:
  [Xia and Conitzer, AAAI 2008]
  there is a position b with

$$\alpha_b - \alpha_{b+1} = \alpha_{b+1} - \alpha_{b+2} = \alpha_{b+2} - \alpha_{b+3}$$

  and

$$\alpha_{b+3} > \alpha_{b+4}$$

  Examples: $(\ldots, 6, 5, 4, 3, 0, \ldots)$, $(\ldots, 17, 14, 11, 8, 7, \ldots)$

Known results for scoring rules

unweighted voters + unbounded number of candidates:

- NP-complete for scoring rules that fulfill the following:
  [Xia and Conitzer, AAAI 2008]
  there is a position b with
  $$\alpha_b - \alpha_{b+1} = \alpha_{b+1} - \alpha_{b+2} = \alpha_{b+2} - \alpha_{b+3}$$
  and
  $$\alpha_{b+3} > \alpha_{b+4}$$
  Examples: $(\dots, 6, 5, 4, 3, 0, \dots)$, $(\dots, 17, 14, 11, 8, 7, \dots)$
- Parameterized complexity study for some scoring rules:
  [Betzler, Hemmann, and Niedermeier, IJCAI 2009]
  k-approval is NP-hard for two partial votes when k is part of the input

Main Theorem

Theorem

For non-trivial pure scoring rules, Possible Winner is

- polynomial-time solvable for plurality and veto,
- open for $(2, 1, \dots, 1, 0)$, and
- NP-complete for all other cases.

Examples for new results:

- 2-approval: $(1, 1, 0, \dots)$
- voting systems in which one can specify a small group of favorites and a small group of disliked candidates, like $(2, 2, 2, 1, \dots, 1, 0, 0)$ or $(3, 1, \dots, 1, 0)$

Plurality

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

$v_1 : a \succ c \succ d, b \succ c$

$v_2 : c \succ a \succ b$

$v_3 : a \succ d \succ b$

$v_4 : a \succ b \succ c$

$v_5 : a \succ c, b \succ d$

Plurality

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

$v_1 : a \succ c \succ d, b \succ c$

$v_2 : c \succ a \succ b$ $\Rightarrow c > a > b > d$

$v_3 : a \succ d \succ b$ $\Rightarrow c > a > d > b$

$v_4 : a \succ b \succ c$

$v_5 : a \succ c, b \succ d$

Step I: Maximize score of c

Plurality

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

$v_1 : a \succ c \succ d, b \succ c$

$v_2 : c \succ a \succ b$ $\Rightarrow c > a > b > d$

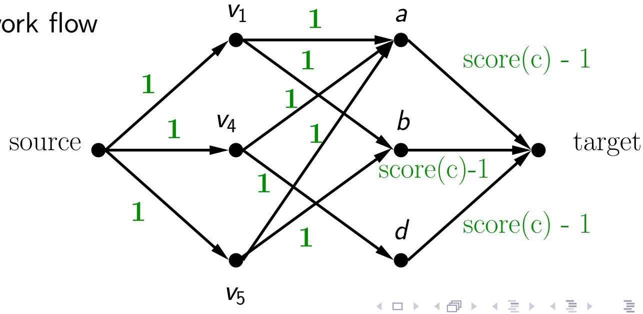
$v_3 : a \succ d \succ b$ $\Rightarrow c > a > d > b$

$v_4 : a \succ b \succ c$

$v_5 : a \succ c, b \succ d$

Step I: Maximize score of c

Step II: Network flow



Plurality

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

$v_1 : a \succ c \succ d, b \succ c \quad \Rightarrow a > b > c > d$

$v_2 : c \succ a \succ b \quad \Rightarrow c > a > b > d$

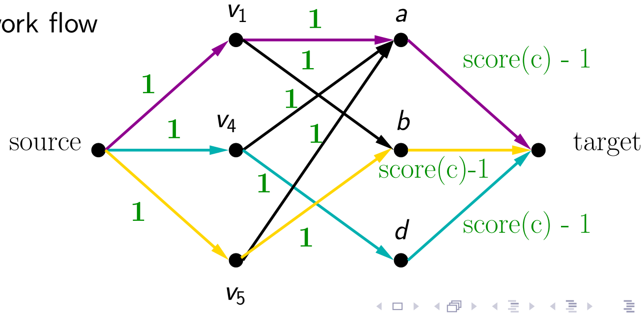
$v_3 : a \succ d \succ b \quad \Rightarrow c > a > d > b$

$v_4 : a \succ b \succ c \quad \Rightarrow d > a > b > c$

$v_5 : a \succ c, b \succ d \quad \Rightarrow b > a > c > d$

Step I: Maximize score of c

Step II: Network flow



2-approval

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

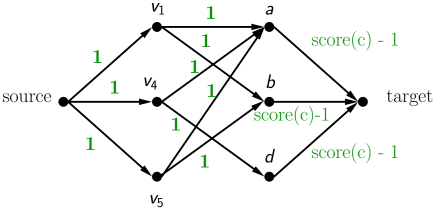
$v_1 : a \succ c \succ d, b \succ c$

$v_2 : c \succ a \succ b$

$v_3 : a \succ d \succ b$

$v_4 : a \succ b \succ c$

$v_5 : a \succ c, b \succ d$



Why does the network flow strategy not work for 2-approval, that is $(1, 1, 0, \dots)$?

2-approval

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

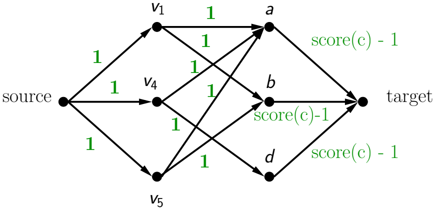
$v_1 : a \succ c \succ d, b \succ c$

$v_2 : c \succ a \succ b$

$v_3 : a \succ d \succ b$

$v_4 : a \succ b \succ c$

$v_5 : a \succ c, b \succ d$



Why does the network flow strategy not work for 2-approval, that is $(1, 1, 0, \ldots)$?

The two "one-point positions" within a partial vote cannot be chosen independently

for example, in v_3:
setting d to a one-point position implies that candidate a must also take a one-point position

Basic idea of the NP-hardness proofs

Different types of many-one reductions are combined to one reduction that works for all of the stated scoring rules.

Types of reductions:

1. for an "unbounded number of positions with different score values",
   e.g. $(m-1, m-2, \ldots, 1, 0)$
2. for an "unbounded number h of equal positions",

Basic idea of the NP-hardness proofs

Different types of many-one reductions are combined to one reduction that works for all of the stated scoring rules.

Types of reductions:

1. for an "unbounded number of positions with different score values",
   e.g. $(m-1, m-2, \dots, 1, 0)$
2. for an "unbounded number h of equal positions",
   that is, there must be a position i such that one of the following cases hold
   a) $\alpha_i \geq \alpha_{i+1} > \alpha_{i+2} = \alpha_{i+h}$, e.g. $(1, 1, 0, \dots)$
   b) $\alpha_{i-h} = \alpha_i > \alpha_{i+1} \geq \alpha_{i+2}$, e.g. $(1, \dots, 1, 0, 0)$
   c) $\alpha_1 \neq \alpha_2 = \alpha_{m-1} \neq \alpha_m, \alpha_1 \neq 2 \cdot \alpha_2$, e.g. $(3, 1, \dots, 1, 0)$

What about non-pure scoring rules?

Theorem

For non-trivial pure scoring rules, Possible Winner is

- polynomial-time solvable for plurality and veto,
- open for $(2, 1, \ldots, 1, 0)$, and
- NP-complete for all other cases.

Problem: scoring rules which have "easy" scoring vectors for nearly all number of candidates and still "hard" scoring vectors for some unbounded numbers of candidates

Property of pure scoring rules: can never go back to an easy vector
Examples: $(1, 0, 0)$, $(1, 1, 0, 0) \rightarrow$ not $(1, 0, 0, 0, 0)$
$(1, 1, 1, 0)$, $(2, 1, 1, 1, 0), \ldots$

Open questions

What about $(2, 1, \ldots, 1, 0)$?

In how many extension does a distinguished candidate win?

Parameter number of candidates: combinatorial algorithm?