Parameterized Computational Complexity of Dodgson and Young Elections

Nadja Betzler

joint work with

Jiong Guo and Rolf Niedermeier

Friedrich-Schiller-Universität Jena
Institut für Informatik

SWAT 2008

Election and Condorcet Winner

Election

Set of votes V, set of candidates C.
A vote is preference list (total order) over all candidates.

Condorcet winner

A candidate c that defeats every other candidate in pairwise comparison, that is, c is better in more than half of the votes.

Example:

vote 1:	a	>	c	>	b
vote 2:	c	>	a	>	b
vote 3:	b	>	c	>	a

$\Rightarrow$ c is the Condorcet winner!

Condorcet winner

Problem

A Condorcet winner does not always exist.

Example:

vote 1:	a	>	b	>	c
vote 2:	c	>	a	>	b
vote 3:	b	>	c	>	a

then:

a is preferred to b

b is preferred to c

c is preferred to a

Condorcet winner

Problem

A Condorcet winner does not always exist.

Way out: Take the candidate that is "closest" to a Condorcet winner. We consider two ways:

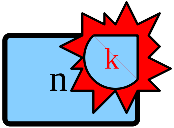
Dodgson and Young elections

In both we compute the score of every candidate and the candidate with highest/lowest score wins.

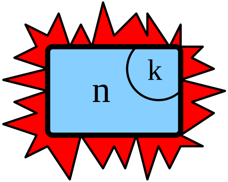
Both problems are NP-hard: Parameterized complexity?

Parameterized Complexity — Introduction I

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



instead of



Definition

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

Parameterized Complexity — Introduction II

Completeness program developed by Downey and Fellows [Downey & Fellows 1999]

$$\text{FPT} \subseteq \overbrace{\text{W[1]} \subseteq \text{W[2]} \subseteq \ldots \subseteq \text{W[P]}}^{\text{Presumably fixed-parameter intractable}}$$

Assumption: FPT $\neq$ W[1]

If W[1]=FPT then 3-SAT for a Boolean formula F with n variables can be solved in $2^{o(n)} \cdot |F|^{O(1)}$ time.

Outline of the talk

- Main part of this talk: Algorithm for Dodgson Score
- Results for extensions of Dodgson Score
- Results for Young Score

Dodgson Score: Definition

Let a *switch* be the swapping of the two positions of two neighboring candidates in a vote.

Dodgson Score

Given: An election (V, C), a distinguished candidate $c \in C$, and an integer $k \geq 0$.
Question: Can c be made a Condorcet winner by at most k switches?

Example:

v_1 : $a > b > d > c$
v_2 : $b > d > c > a$
v_3 : $c > a > b > d$

No Condorcet winner:
a is beaten by c,..
b is beaten by a,..
c is beaten by d,..
d is beaten by b,..

Dodgson Score: Definition

Let a *switch* be the swapping of the two positions of two neighboring candidates in a vote.

DODGSON SCORE

Given: An election (V, C), a distinguished candidate $c \in C$, and an integer $k \geq 0$.
Question: Can c be made a Condorcet winner by at most k switches?

Example:

$$
\begin{aligned}
v_1 &: \quad a > b > d > c \\
v_2 &: \quad b > d > c > a \\
v_3 &: \quad c > a > b > d
\end{aligned}
$$

Candidate c can become Condorcet winner by 2 switches.

Dodgson Score: Known results

- NP-complete

  [Bartholdi III, Tovey & Trick, Social Choice and Welfare 1989]

- Winner and ranking variant are Θ_2^p-complete

  [Hemaspaandra, Hemaspaandra, & Rothe, J. ACM 1997]

Algorithms:

- Greedy heuristic with frequent success guarantee

  [Homan & L. Hemaspaandra, J. Heuristics 2007]

- Approximability

  [McCabe-Dansted, Pritchard & Slinko, Social Choice and Welfare 2007]

  [Covey & Homan, manuscript 2008]

  [Caragiannis & al. , working paper 2008]

Parameterized Complexity of Dodgson Score

Number of votes n
Number of candidates m

Theorem

DODGSON SCORE can be solved in $O(2^k \cdot nk + nm)$ time.

Hence: For a candidate that is close to be a Condorcet winner the Dodgson Score can be computed efficiently!

Proof: Dynamic programming algorithm

Dynamic Programming Algorithm

Example:

vote 1 :	c_2	$>$	c_3	$>$	c	$>$	c_1
vote 2 :	c_1	$>$	c_2	$>$	c	$>$	c_3
vote 3 :	c_1	$>$	c_2	$>$	c	$>$	c_3
vote 4 :	c_2	$>$	c	$>$	c_3	$>$	c_1
vote 5 :	c_3	$>$	c_2	$>$	c_1	$>$	c

Deficit:

candidate c_1: $d_1 = 1$
candidate c_2: $d_2 = 3$
candidate c_3: $d_3 = 0$

2 Observations

- Only switches that "improve" c need to be considered.
- One switch can decrease the deficit of at most one candidate by one. Thus:
  Number of candidates with positive deficit is bounded by k.
  The sum of all deficits is bounded by k.

Dynamic Programming Algorithm

Example:

vote 1 :	c_2	$>$	c_3	$>$	c	$>$	c_1
vote 2 :	c_1	$>$	c_2	$>$	c	$>$	c_3
vote 3 :	c_1	$>$	c_2	$>$	c	$>$	c_3
vote 4 :	c_2	$>$	c	$>$	c_3	$>$	c_1
vote 5 :	c_3	$>$	c_2	$>$	c_1	$>$	c

Deficit:

candidate c_1: $d_1 = 1$
candidate c_2: $d_2 = 3$
candidate c_3: $d_3 = 0$

Dynamic Programming Algorithm

Example:

vote 1 :	c_2	$>$	c_3	$>$	c	$>$	c_1
vote 2 :	c_1	$>$	c_2	$>$	c	$>$	c_3
vote 3 :	c_1	$>$	c_2	$>$	c	$>$	c_3
vote 4 :	c_2	$>$	c	$>$	c_3	$>$	c_1
vote 5 :	c_3	$>$	c_2	$>$	c_1	$>$	c

Deficit:

candidate c_1: $d_1 = 1$
candidate c_2: $d_2 = 3$
candidate c_3: $d_3 = 0$

Deficit list $(d_1, d_2, d_3) = (1, 3, 0)$.

Partial deficit lists : list of decreased deficits

Basic idea

Decompose the solution by decomposing the deficits.

Dynamic programming table

vote 1 :	c_2	>	c_3	>	c	>	c_1
vote 2 :	c_1	>	c_2	>	c	>	c_3
vote 3 :	c_1	>	c_2	>	c	>	c_3
vote 4 :	c_2	>	c	>	c_3	>	c_1
vote 5 :	c_3	>	c_2	>	c_1	>	c

Deficit:

candidate c_1: $d_1 = 1$
candidate c_2: $d_2 = 3$
candidate c_3: $d_3 = 0$

Table entry: Minimum number of switches needed to achieve a partial deficit list in a set of votes.

Initialization:

	$(0,0)$	$(0,1)$	$(1,0)$	$(1,1)$	$(0,2)$	$(1,2)$	$(0,3)$	$(1,3)$
$\{v_1\}$	∞	∞	∞	∞	∞	2	∞	0
$\{v_1, v_2\}$	...							
$\{v_1, v_2, v_3\}$	...							

Dynamic programming table

vote 1 : $c_2 > c_3 > c > c_1$
vote 2 : $c_1 > c_2 > c > c_3$
vote 3 : $c_1 > c_2 > c > c_3$
vote 4 : $c_2 > c > c_3 > c_1$
vote 5 : $c_3 > c_2 > c_1 > c$

Deficit:

candidate c_1: $d_1 = 1$
candidate c_2: $d_2 = 3$
candidate c_3: $d_3 = 0$

Table entry: Minimum number of switches needed to achieve a partial deficit list in a set of votes.

Update:

	(0,0)	(0,1)	(1,0)	(1,1)	(0,2)	(1,2)	(0,3)	(1,3)
$\{v_1\}$	∞	∞	∞	∞	∞	2	∞	0
$\{v_1, v_2\}$	∞	4	∞	3	2	1	2	0
$\{v_1, v_2, v_3\}$	...							

$$T(\{v_1, v_2\}, (1,2)) = \min \begin{cases} T(\{v_1\}, (1,2)) \\ T(\{v_1\}, (1,3)) + \text{cost of improvement in} v_2 \end{cases}$$

For n votes and m candidates we can show:

- The size of the dynamic programming table is bounded by $2^k \cdot n$. (Since the overall sum of deficits is bounded by k.)
- Computation of a new entry can be done in time linear in k.

Theorem

DODGSON SCORE can be solved in $O(2^k \cdot nk + nm)$ time.

Dodgson Score: Allowing ties

Two natural ways going from total to partial orders of the votes:

Transformation of

$$b = a > c \quad \text{into} \quad c > b = a$$

- requires one switch (Model 1).
- requires two switches (Model 2). Here one can choose upon which candidate to improve first.

Dodgson Score: Allowing ties

Two natural ways going from total to partial orders of the votes:

Transformation of

$$b = a > c \quad \text{into} \quad c > b = a$$

- requires one switch (Model 1).
- requires two switches (Model 2). Here one can choose upon which candidate to improve first.

Theorem

Dodgson Tie Score 2 can be solved in $O(4^k \cdot nk + nm)$ time.

Theorem

Dodgson Tie Score 1 is $W[2]$-hard with respect to k.

Young Score: Definition

Young Score

Given: An election (V, C), a candidate $c \in C$, an integer $s \geq 0$.
Question: A subset $V' \subseteq V$ of size at least s such that (V', C) has the Condorcet winner c?

Dual Young Score asks if c can become Condorcet winner by deleting at most k votes. [Young, J. Econ. Theo. 1977]

Example:

vote 1:	b	>	a	>	c
vote 2:	a	>	c	>	b
vote 3:	a	>	c	>	b
vote 4:	c	>	a	>	b
vote 5:	b	>	c	>	a

Delete vote 1 and 2 !

Young Score: Parameterized Complexity

Theorem

DUAL YOUNG SCORE is W[2]-complete with respect to the number of deleted votes.

- W[2]-hardness: Reduction from a variant of Dominating Set.
- W[2]-membership: Reduction to Optimal Lobbying.
  (W[2]-completeness proven by [CHRISTIAN ET AL., REVIEW OF ECONOMIC DESIGN 2007])

Theorem

YOUNG SCORE is W[2]-complete with respect to the solution size.

- W[2]-hardness: Follows from the reduction of the Θ_2^p-completeness for the winner and ranking problems.
  [ROTHE, SPAKOWSKI, & VOGEL, TOCS 2003]
- W[2]-membership: Reduction to Optimal Lobbying.

Conclusion

Main observation

In contrast to classic complexity theory, the parameterized complexity of Dodgson Score and Young Score differs.

Whereas Dogson Score and both variants that allow ties are NP-complete, they have different parameterized complexity.

Outlook

Only W[2]-hardness of DODGSON TIE SCORE 1 is proven. Is it also W[2]-complete?

Other parameterizations:

parameter	Dodgson Score	dual Young Score	Young Score
# votes n	W[1]-h	FPT (2^n)	FPT (2^n)
# candidates m	FPT (ILP)	FPT (ILP)	FPT (ILP)
# steps k	**FPT (2^k)**	W[2]-**complete**	W[2]-**complete**

ILP DODGSON SCORE: [BARTHOLDI III, TOVEY & TRICK, SCW 1989] and improved by [MACCABE-DANSTED 2006]
ILP YOUNG SCORE: [YOUNG, J. ECON. THEO. 1977]
new result