

Control Complexity of Schulze Voting

Curtis Menton¹, Preetjot Singh²

¹Computer Science, University of Rochester, Rochester, NY, USA

²EECS, Northwestern University, Evanston, IL, USA



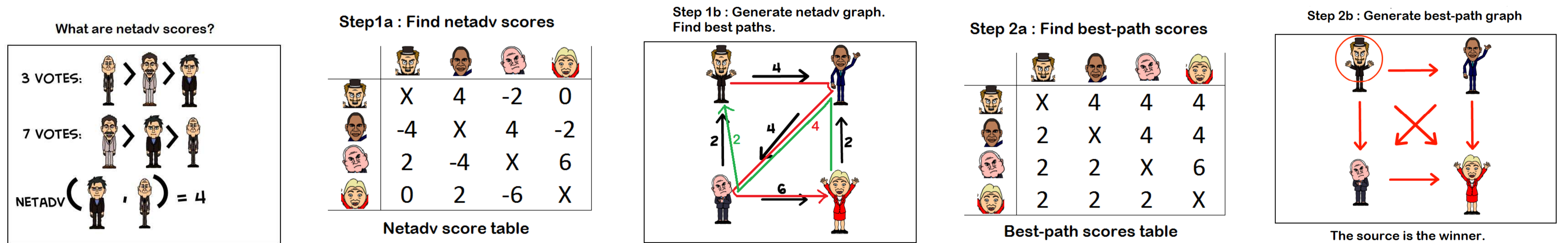
Overview

One critical area of Computational Social Choice is the study of voting rules that contain just the right amount of complexity - enough to discourage strategic behaviour, but not so much that calculating the winner is NP-hard. The Schulze voting rule, introduced recently by Marcus Schulze in 1997, has quickly gained a high degree of real-world use: Users include **Wikimedia**, the **Pirate Party**, **MTV**, and **Ubuntu**, among others. It is a Condorcet voting rule with a complex winner determination method that is nevertheless in polynomial time. Our paper resolves the complexity of several electoral *control* cases, demonstrating previously unknown vulnerabilities and resistances of the voting rule.

Definitions

An *election* is a pair (C, V) where C is a set of candidates and V is a collection of votes. *Votes* are preferences over candidates, expressed (in our case) as a strict linear ordering (such as $A > B > C$ indicating A is preferred to B is preferred to C). A *voting rule* is a mapping from an election (C, V) to a set $C' \subseteq C$ of winners.

How Schulze Finds The Winner



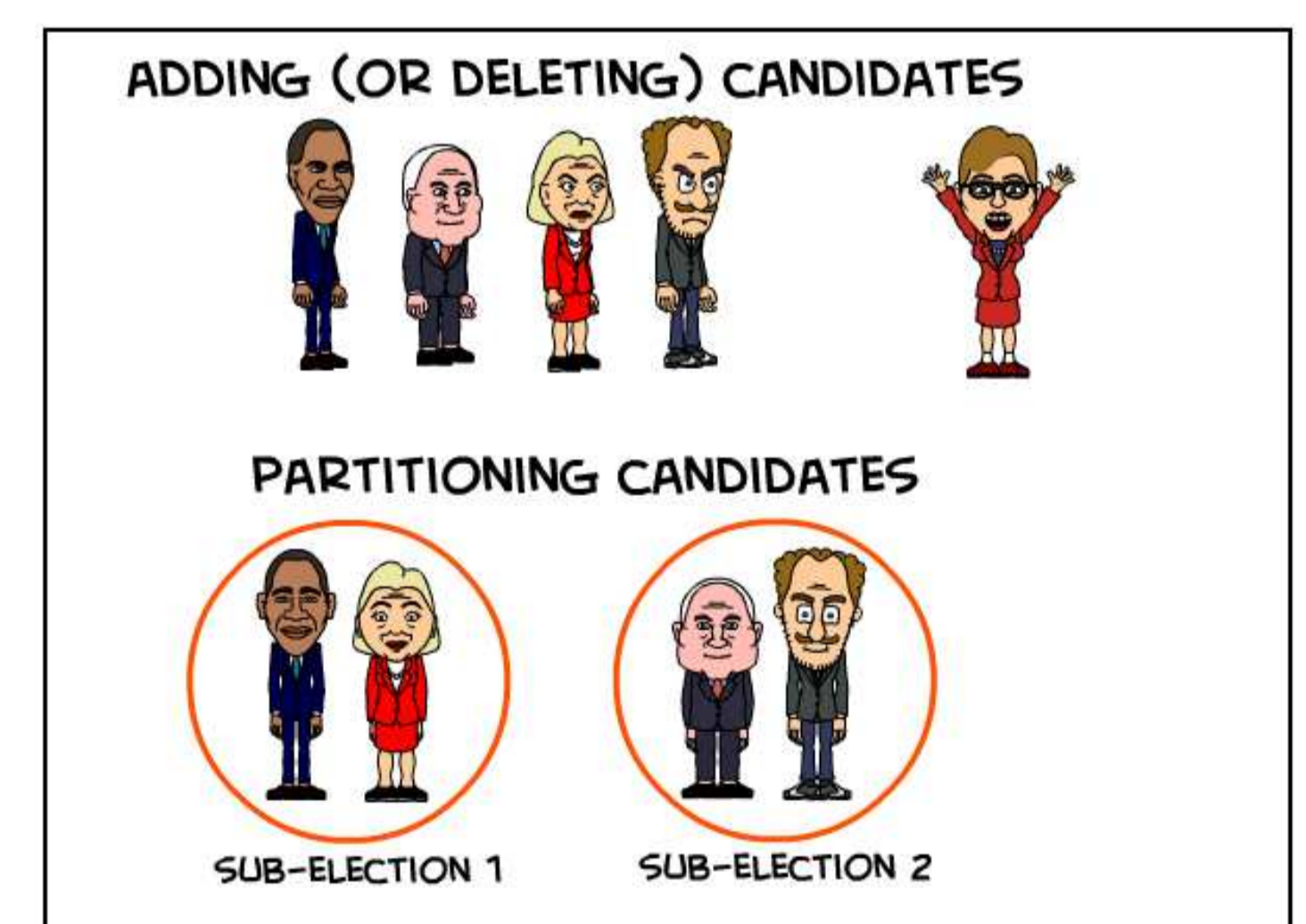
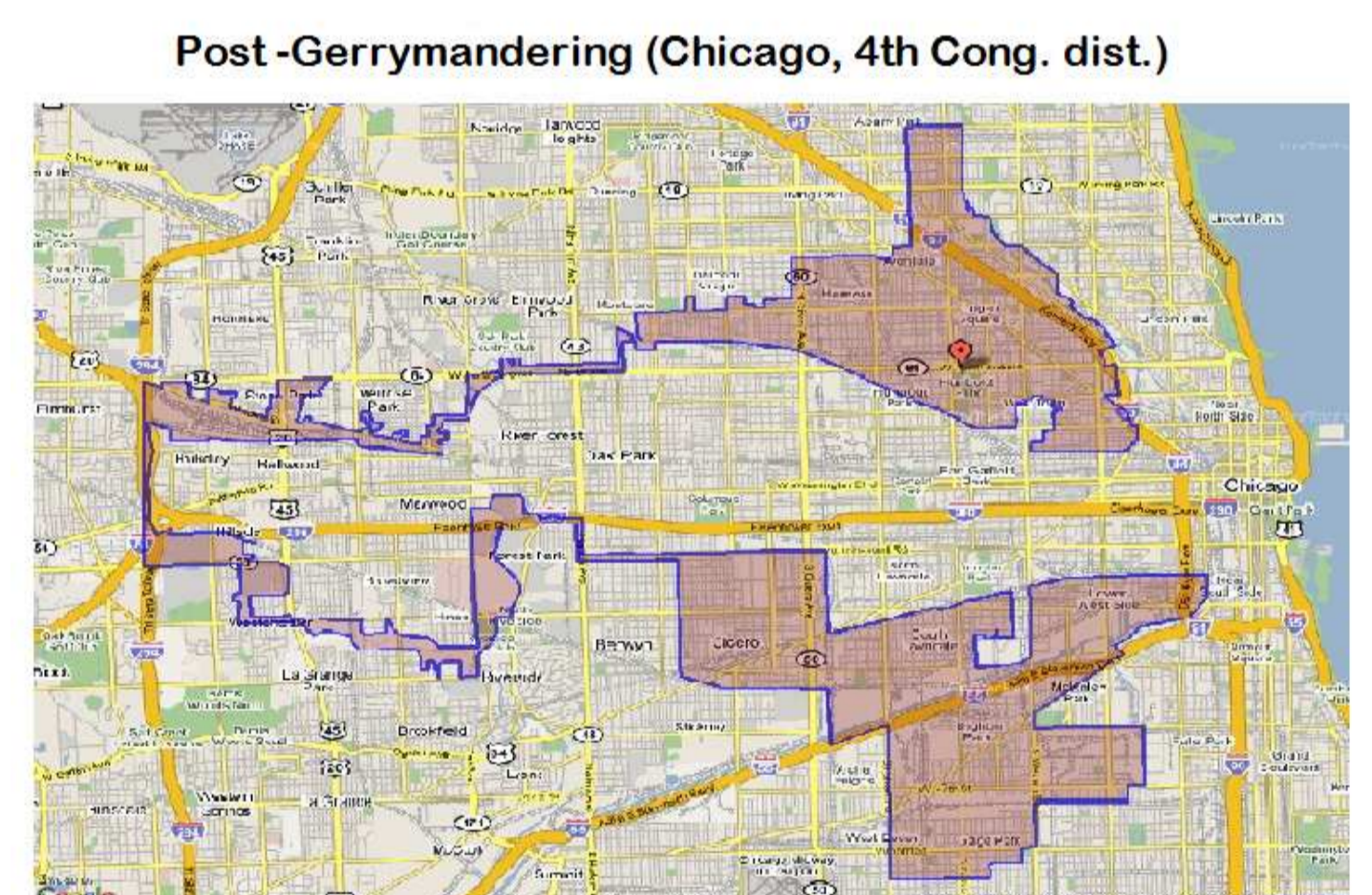
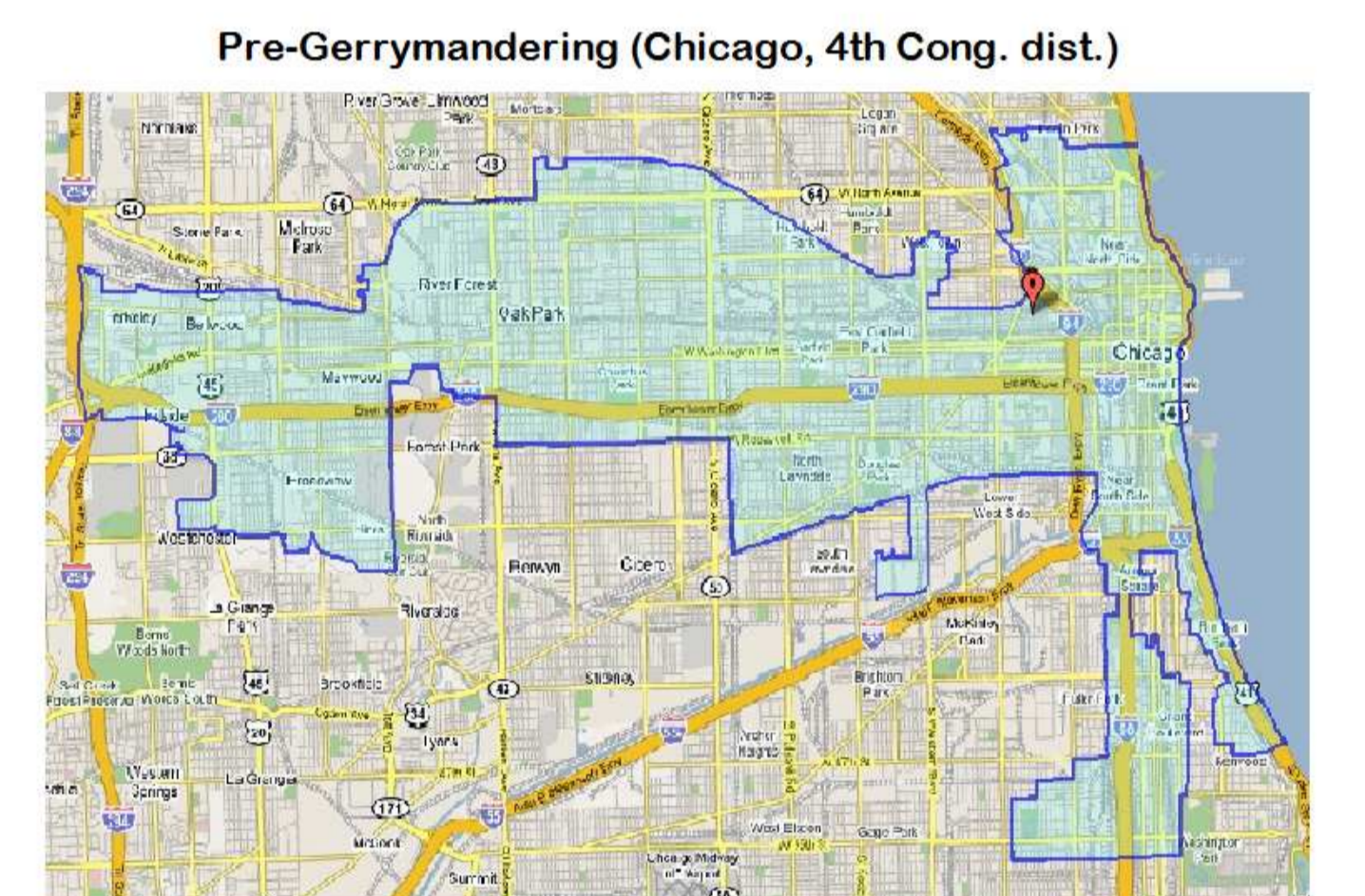
Control

Control refers to a class of manipulative actions that alter the structure of an election to change the result. We model these as decision problems where we determine whether a case of control is possible in a given instance.

Control cases include:

- Adding or deleting voters (voter registration drives or voter suppression efforts).
- Adding or deleting candidates (ballot-access procedures).
- Partitioning voters (gerrymandering).
- Partitioning candidates (primary elections or runoff elections).

Control cases



Results

Control by	Tie Model	Plurality		Approval		Fallback		Schulze	
		C	D	C	D	C	D	C	D
Adding Candidates		R	R	I	V	R	R	R	S
Adding Candidates (unlimited)		R	R	I	V	R	R	R	S
Deleting Candidates		R	R	V	I	R	R	R	S
Partition of Candidates	TE	R	R	V	I	R	R	R	V
	TP	R	R	I	I	R	R	R	V
Run-off Partition of Candidates	TE	R	R	V	I	R	R	R	V
	TP	R	R	I	I	R	R	R	V
Adding Voters		V	V	R	V	R	V	R	R
Deleting Voters		V	V	R	V	R	V	R	R
Partition of Voters	TE	V	V	R	V	R	R	R	R
	TP	R	R	R	V	R	R	R	R

Figure 1: Control behavior under Schulze voting and other voting systems for comparison. V, R, S, and I stand for vulnerable, resistant, susceptible, and immune. Results proved in this paper in bold, other results from [3, 1, 2, 4, 5].

References

- [1] G. Erdélyi and M. Fellows. Parameterized control complexity in bucklin voting and in fallback voting. In *Proceedings of the 3rd International Workshop on Computational Social Choice*, pages 163–174, 2010.
- [2] G. Erdélyi, L. Piras, and J. Rothe. The complexity of voter partition in Bucklin and fallback voting: Solving three open problems. In *Proceedings of the 10th International Conference on Autonomous Agents and Multiagent Systems*, pages 837–844, May 2011.
- [3] G. Erdélyi and J. Rothe. Control complexity in fallback voting. In *Proceedings of the 16th Australasian Theory Symposium*, pages 39–48, Jan. 2010.
- [4] E. Hemaspaandra, L. Hemaspaandra, and J. Rothe. Anyone but him: The complexity of precluding an alternative. *Artificial Intelligence*, 171(5–6):255–285, 2007.
- [5] D. Parkes and L. Xia. A complexity-of-strategic-behavior comparison between Schulze’s rule and ranked pairs. In *Proceedings of the 26th AAAI Conference on Artificial Intelligence*, pages 1429–1435, 2012.