Complexity Results on Preference Aggregation: Winner Determination

1 Introduction, Terminology and Definitions

Social Choice Theory referees to the study and design of mechanisms for collective decision making. It involves models and results concerned with aggregating *individual inputs* into a *collective output*. The main questions asked are *how to design voting systems* and *what are the properties of different voting systems*. For more see, [1], [3] and [2].

Computational Complexity Theory aims to classify problems into different classes based on their difficulty and asks what makes some problems *hard* and some *easy* **??**. We consider an algorithm efficient if the number of steps is polynomial to the input size.

We focus on *Algorithmic* and *Computational* aspects of Social Choice Theory. The question - *can the voting mechanism guarantee to identify a winner efficiently* is natural and is known as the *Winner Determination* Problem. This is given formally below. First we require some definitions.

Definition 1 (Election)**.** *An election is given by a pair (C, V), where C is the set of candidates and V is the set of votes. A voter is expressed by his/her votes which are a preference ordering on C*

Definition 2 (Voting System)**.** *A voting system is a function $f : (C, V) \to 2^C$*

Many different voting systems exist. We focus on:

- Scoring Protocols: Let m be the number of candidates. In a scoring protocol, we have a *scoring vector* $\mathbf{a} \in \mathbb{R}^m$ that determines the number of points that each candidate gets. The basic idea is that the candidate in the ith position in the preference list of voter i receives a_i points. The difference in voting rules based on this idea lies in the way the a_is are given.
  - Plurality: Here only the top candidate receives a point and other candidates get zero.
  - k-Approval: Let $k \leq m$. The first k candidates get one point - the remaining zero. For $k = 1$ this is *Plurality*.

- Borda Count: In the Borda Count method, with m candidates we give scores $m-1, m-2, ..., 0$ for the candidate in the first, second , and so on, position of each voter.

- Pairwise Comparisons: Under these systems, candidates do not receive points immediately. Instead, we perform pairwise comparisons to determine which candidate is preferred and then check a majority condition to see who wins.
  - Condorcet: Here the candidate who is preferred to each other candidate by more than half of the voters wins.
  - Copeland

- Sequential Voting - voting proceeding in stages
  - Single Transferable Voting: Given the complete orderings of each voter, we proceed in rounds; This is not fixed but is at most the number of candidates. At each stage, give one point to each voter's top candidate. Then, if we can find a candidate who scores a point for more than half of the voters, that candidate wins. Otherwise, delete the candidate with the least score and repeat. If a candidate that is the top is deleted, then transfer the votes to the next candidate.

2 Polynomial Time Results

Theorem 2.1. *$\mathcal{E}$-WINNER $\in$ **P** if $\mathcal{E}$ is a Scoring Protocol.*

Proof. For any method relying on the scoring protocol, set a $n \times m$ array where on the horizontal axis we have some ordering of the m candidates and on the vertical axis some ordering of the voters. Then, entry i, j is given by either 1 or $m-k$, for some k depending on the method used. We loop through the array to find the sum of each column - this gives us the scoring vector. Then, we find the maximum value in the vector and we are done. Both of these can be done in polynomial time, hence the problem is in **P**

$n * m$ □

Theorem 2.2. *$\mathcal{E}$-WINNER $\in$ **P** if $\mathcal{E}$ is the Condorcet Voting Protocol.*

Proof. Represent the results of the comparisons by a directed graph such that there is an edge going from X to Y if and only if X wins Y by a majority of the voters. To determine the winner (if it exists[1]), it suffices to find the vertex with the most out-neighbours. It takes

[1] A Condorcet Winner is *not* guaranteed to exist

$O(n)$ steps to do this, where $n = |V| + |E|$, hence the problem is in **P** *relate to the number of candidates!! $\square$

Theorem 2.3. *$\mathcal{E}$-WINNER $\in$ **P** if $\mathcal{E}$ in the Copeland Family*

Proof. to add. $\square$

Theorem 2.4. *$\mathcal{E}$-WINNER $\in$ **P** if $\mathcal{E}$ is the Single Transferable Voting Protocol*

Proof. STV into pseudocode. Analyze running time: is quadratic time. Therefore, it is in **P** $\square$

Bibliography

[1] Yann Chevaleyre, Ulle Endriss, Jérôme Lang, and Nicolas Maudet. A short introduction to computational social choice. In Jan van Leeuwen, Giuseppe F. Italiano, Wiebe van der Hoek, Christoph Meinel, Harald Sack, and František Plášil, editors, *SOFSEM 2007: Theory and Practice of Computer Science*, pages 51–69, Berlin, Heidelberg, 2007. Springer Berlin Heidelberg.

[2] Christian List. Social Choice Theory. In Edward N. Zalta and Uri Nodelman, editors, *The Stanford Encyclopedia of Philosophy*. Metaphysics Research Lab, Stanford University, Winter 2022 edition, 2022.

[3] Jörg Rothe et al. *Economics and Computation: An Introduction to Algorithmic Game Theory, Computational Social Choice and Fair Division*, volume 4. Springer, 2015.