ETH
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Distributed Computing

Prof. R. Wattenhofer

Topics in Computational Voting and Game Theory

Voting usually means electing a leader or governing body, but we also see it in our day-to-day decisions: maybe a team needs to pick a good meeting time, or a group of referees have to decide on a contest winner. Some computational problems also resolve to voting: selecting a diverse set of movies to offer plane passengers, or choosing the location of a fire station for rapid response.

Among the first formulated problems in voting was the problem of determining the election winner(s). Many answers have been proposed, each with its own downsides, like susceptibility to strategic voting, where voters misrepresent their true opinions to gain an advantage. Unfortunately, the Gibbard–Satterthwaite Theorem tells us that any system perfectly immune to strategic behaviour is neccesarily dictatorial, so there is no "best" voting system.

Most voting systems are easy to implement: voters cast their votes, then the authority aggregates them in a simple way and declares the winner(s). Surprisingly, this is not always the case: Lewis Carroll proposed a system where the winner is the candidate who, with the fewest changes in voters' preferences, beats all other candidates in pairwise one-on-one races.[1] As later shown, it is NP-hard to determine the winner in this system.

Computational Social Choice deals with many challenging algorithmic questions: manipulating the outcome of the election, voter bribery, proportional representation, multi-winner elections, voting with uncertainty, voting under left-right wing assumptions, as well as axiomatic results, like proving that no voting system can satisfy a number of desirable properties. Algorithmic Game Theory adds related questions to the mix: fair division, housing allocation, stable matchings, coalition formation, computing equilibria, participatory budgeting, etc.

Requirements. Ability to work independently and interest in conducting new research. Solid algorithmic and mathematical background (emphasis on writing proofs). Experience with voting theory not required. Depending on the extent of the project, coding skills are a plus. We will have weekly meetings to discuss open questions and the next steps.

Interested? Please contact us for more details!

- Andrei Constantinescu: aconstantine@ethz.ch, ETZ G93

[1] Wait a minute, doesn't such a candidate always exist from the get-go? No, look up "Condorcet paradox".