Computational Social Choice 2020

Ulle Endriss
Institute for Logic, Language and Computation
University of Amsterdam

[http://www.illc.uva.nl/~ulle/teaching/comsoc/2020/]

Plan for Today

The main purpose of today's lecture is to give you enough information to allow you to decide whether you want to take this course.

- What is Computational Social Choice? Why study COMSOC?
- Organisational Matters: planning, expectations, assessment, ...
- First Topic: Voting to elect a single representative

Opening Example

Five *voters* express their *preferences* over three *alternatives*. We need to find a *good* ranking of the alternatives to reflect this information:

Voter 1: $a \succ b \succ c$

Voter 2: $b \succ c \succ a$

Voter 3: $c \succ a \succ b$

Voter 4: $c \succ a \succ b$

Voter 5: $b \succ c \succ a$

?

What is Computational Social Choice?

Social choice theory is about methods for *collective decision making*, such as *political* decision making by groups of *economic* agents.

Its methodology ranges from the *philosophical* to the *mathematical*. It is traditionally studied in *Economics* and *Political Science* and it is a close cousin of both *decision theory* and *game theory*.

Its findings are relevant to multiple *applications*, such as these:

- How to fairly allocate resources to the members of a society?
- How to fairly divide computing time between several users?
- How to elect a president given people's preferences?
- How to combine the website rankings of multiple search engines?
- How to aggregate the views of different judges in a court case?
- How to extract information from noisy crowdsourced data?

Computational social choice, the topic of this course, emphasises the fact that any method of decision making is ultimately an *algorithm*.

Special Theme

For this edition of the course, I want to specifically focus on what you might call *innovations in democratic decision making*. Keywords:

- *liquid democracy*
- *participatory budgeting*

But we will start with some more established topics in *voting theory*.

Relationship with AI

Ideas from Economics entered AI when it became clear that we can use them to study interaction between agents in a *multiagent system*. Nowadays, the study of "*economic paradigms*" is all over AI.

The influential *One Hundred Year Study on Artificial Intelligence* (2016) singles out the following eleven "*hot topics*" in AI:

large-scale machine learning | deep learning | reinforcement learning
robotics | computer vision | natural language processing
collaborative systems | crowdsourcing and human computation
algorithmic game theory and *computational social choice*
internet of things | neuromorphic computing

And indeed, while COMSOC transcends several disciplines, about half of it gets published in AI conference proceedings and journals.

P. Stone et al. "Artificial Intelligence and Life in 2030". One Hundred Year Study on Artificial Intelligence. Stanford, 2016.

Prerequisites

This is an advanced *research-oriented* course: we'll move fast and often touch upon recent research. The focus is on *theory*.

I expect *mathematical maturity* (working out and writing up proofs), but little in terms of specific *mathematical knowledge:* just some basic concepts from combinatorics, probability theory, and logic.

Some background in the following is useful but not strictly required:

- *Game Theory:* We'll often reason about agents being strategic. Prior exposure to game theory helps with this kind of thinking.
- *Complexity Theory:* Required to analyse social choice mechanisms from an algorithmic point of view.
- *Programming:* For one homework assignment some very modest programming skills will be required (in Python). Help is available.

Organisational Matters

- **Website:** Lecture slides, literature, homework, project ideas, and other important information will get posted on the course website.

  Literature includes, in particular, the *Handbook of COMSOC*.

- **Canvas:** Used for announcements, submission of homework, discussions, and to make available the lecture recordings.

- **Assessment:** Homework (50%) + mini-project (50%).

  You worst HW grade won't count (so: ok if you need to skip one).

- **Commitment:** Be ready to invest ∼20h/week. Heavy HW regime for the first three weeks; after that the focus is on the projects.

  You should usually be present at (online) lecturers.

F. Brandt, V. Conitzer, U. Endriss, J. Lang, and A.D. Procaccia (eds), *Handbook of Computational Social Choice*. Cambridge University Press, 2016.

Homework

Most questions will be of the problem-solving sort, requiring:

- a good *understanding* of the topic to see what the question is
- some *creativity* to find the solution
- *mathematical maturity*, to write it up correctly, often as a proof
- good *taste*, to write it up in a reader-friendly manner

Solutions must be *typed* up professionally (LaTeX strongly preferred).

Of course, solutions should be *correct*. But just as importantly, they should be *short* and *easy to understand*. (This is the advanced level: it's not anymore just about you getting it, it's now about your reader!)

The usual rules on permissible *collaboration* apply: discussing with others to improve your *understanding* is fine (indeed, it is encouraged), but producing your *solutions* is something you do by yourself.

Mini-Projects

During the second part of the course you'll work on your mini-project in a team of *three students*. Possible types of projects include:

- identify an interesting relevant paper not covered in class and fill in some gaps, or come up with an extension or a generalisation
- apply an algorithmic technique to a problem that to date has only been considered by economists/political scientists/philosophers
- explore an application domain: could be a literature review, an idea for a new application, or an experimental study

The purpose of this is to provide you with some research experience.

All projects must be related to the *special theme* and make use of some of the *COMSOC techniques* introduced during the course.

Deliverables: *paper* + *video* (instead of presentation)

Voting Theory

The core scenario of collective decision making studied in social choice theory concerns the election of a single "winner" given the (ranked) preferences of several voters over a set of alternatives.

Today we start by defining this *model* and many different *voting rules:*

- Positional scoring rules
- Rules based on the majority graph
- Rules based on the weighted majority graph
- Run-off rules (only one today)

For full details, consult Zwicker (2016) or other introductory texts.

W.S. Zwicker. Introduction to the Theory of Voting. In F. Brandt et al. (eds.), *Handbook of Computational Social Choice*. CUP, 2016.

Three Voting Rules

Suppose n *voters* choose from a set of m *alternatives* by stating their preferences in the form of *linear orders* over the alternatives.

Here are three *voting rules* (there are many more):

- *Plurality:* elect the alternative ranked first most often (i.e., each voter assigns 1 point to an alternative of her choice, and the alternative receiving the most points wins)
- *Plurality with runoff:* run a plurality election and retain the two front-runners; then run a majority contest between them
- *Borda:* each voter gives $m-1$ points to the alternative she ranks first, $m-2$ to the alternative she ranks second, etc.; and the alternative with the most points wins

Example: Choosing a Beverage for Lunch

Consider this election, with nine *voters* having to choose from three *alternatives* (namely what beverage to order for a common lunch):

2 Germans:	Beer $\succ$ Wine $\succ$ Milk
3 Frenchmen:	Wine $\succ$ Beer $\succ$ Milk
4 Dutchmen:	Milk $\succ$ Beer $\succ$ Wine

Which beverage *wins* the election for

- the plurality rule?
- plurality with runoff?
- the Borda rule?

The Model

Fix a finite set $A = \{a, b, c, \ldots\}$ of *alternatives*, with $|A| = m \geqslant 2$.

Let $\mathcal{L}(A)$ denote the set of all strict linear orders R on A. We use elements of $\mathcal{L}(A)$ to model (true) *preferences* and (declared) *ballots*.

Each member i of a finite set $N = \{1, \ldots, n\}$ of *voters* supplies us with a ballot R_i, giving rise to a *profile* $\boldsymbol{R} = (R_1, \ldots, R_n) \in \mathcal{L}(A)^n$.

A *voting rule* (or *social choice function*) for N and A selects (ideally) one or (in case of a tie) more winners for every such profile:

$$F : \mathcal{L}(A)^n \to 2^A \setminus \{\emptyset\}$$

If $|F(\boldsymbol{R})| = 1$ for all profiles $\boldsymbol{R}$, then F is called *resolute*.

Most natural voting rules are *irresolute* and have to be paired with a *tie-breaking rule* to always select a unique election winner.

Examples: random tie-breaking, lexicographic tie-breaking

Positional Scoring Rules

We can generalise the idea underlying the Borda rule as follows:

A *positional scoring rule* (PSR) is defined by a so-called *scoring vector* $\boldsymbol{s} = (s_1, \ldots, s_m) \in \mathbb{R}^m$ with $s_1 \geqslant s_2 \geqslant \cdots \geqslant s_m$ and $s_1 > s_m$.

Each voter submits a ranking of the m alternatives. Each alternative receives s_i points for every voter putting it at the ith position.

The alternative(s) with the highest score (sum of points) win(s).

<u>Examples:</u>

- *Borda rule* = PSR with scoring vector $(m-1, m-2, \ldots, 0)$
- *Plurality rule* = PSR with scoring vector $(1, 0, \ldots, 0)$
- *Antiplurality* (or *veto*) *rule* = PSR with scoring vector $(1, \ldots, 1, 0)$
- For any $k < m$, *k-approval* = PSR with $(\underbrace{1, \ldots, 1}_{k}, 0, \ldots, 0)$

<u>Exercise:</u> *Name the rule induced by $\boldsymbol{s} = (9, 7, 5)$! General idea?*

Rules Based on the Majority Graph

Tempting to want to use "the majority rule". *But what would that be?*

$$\begin{array}{ll} \text{Voter 1:} & a \succ b \succ c \\ \text{Voter 2:} & b \succ c \succ a \\ \text{Voter 3:} & c \succ a \succ b \end{array}$$

This is the famous *Condorcet Paradox* (the majority relation is cyclic).

But when *n is odd*, given a profile, we can construct the corresponding *majority graph* $\langle A, \succ_M \rangle$, with $x \succ_M y$ iff $> \frac{n}{2}$ voters rank x above y.

Several rules can be defined in terms of the majority graph. Examples:

- *Copeland rule:* score of alternative $=$ out-degree in the graph (thus: you get a point for every won one-to-one majority contest)
- *Slater rule:* Find a ranking minimising number of edges in majority graph we'd have to switch. Elect top alternative in that ranking.

Refinements of these definitions for *even n* exist.

Example

Most voting rules are *not* definable in terms of the majority graph.

The *plurality rule* is an example. Consider these two profiles:

$$\begin{array}{ccc} a \succ_1 b \succ_1 c & \qquad & a \succ_1 b \succ_1 c \\ a \succ_2 b \succ_2 c & \qquad & b \succ_2 a \succ_2 c \\ c \succ_3 b \succ_3 a & \qquad & c \succ_3 a \succ_3 b \end{array}$$

Same majority graph ($a \succ_M b \succ_M c$), yet plurality winners differ!

Rules Based on the Weighted Majority Graph

For simplicity, let's again assume that *n is odd*. (But can generalise.)

Given a profile, the corresponding *weighted majority graph* $\langle A, \succ_M, w\rangle$ is defined by the majority graph $\langle A, \succ_M\rangle$ and the weight function w that maps any pair (x, y) in $\succ_M$ to the difference between the number of voters ranking x above y and those ranking y above x.

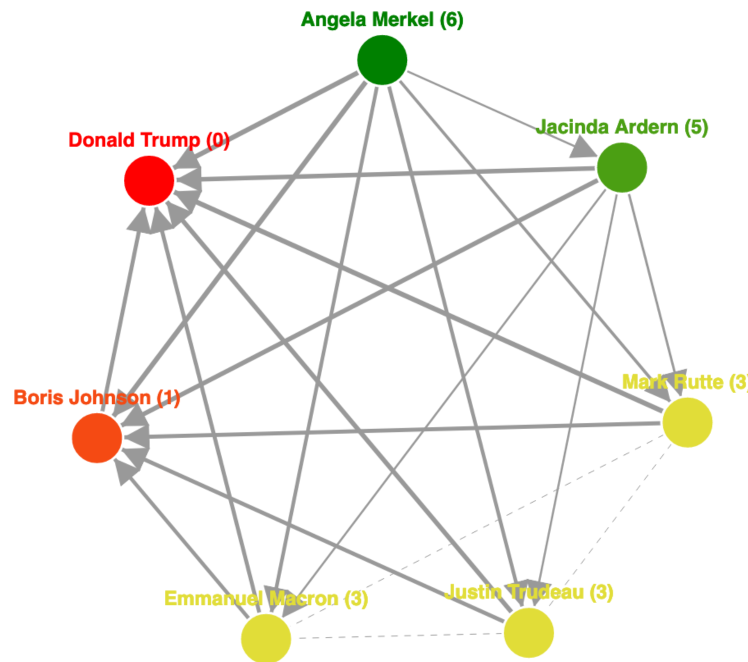
Several rules can be defined in terms of the WMG. Examples:

- *Kemeny rule:* Find a ranking that minimises the sum of weights of the edges in the weighted majority graph we would have to switch. Then elect the top alternative in that ranking.
- *Ranked-Pairs:* Build a full ranking by "locking in" ordered pairs in order of majority strength (but avoid cycles). Elect top alternative.

Exercise: *Show that the Borda rule is definable in terms of the WMG!*

WHich ALternative is Elected?

Whale (`whale.imag.fr`) is a great tool to collect ballots, compute election outcomes for several rules, and visualise your data.



Exercise: *What do some of the voting rules currently not implemented in Whale (Slater, Kemeny, ranked-pairs) have in common?*

S. Bouveret. Social Choice on the Web. In U. Endriss (ed.), *Trends in Computational Social Choice*. AI Access, 2017.

Summary

We have introduced the basic model of voting theory concerned with electing a single alternative based on a profile of ranked preferences and we have seen a first selection of voting rules:

- Positional scoring rules: Borda, plurality, antiplurality, k-approval
- Based on majority graph: Copeland, Slater
- Based on weighted majority graph: Kemeny, ranked-pairs, (Borda)
- Plurality with runoff (generalisation to follow)

Read Chapter 1 of the Handbook to get an understanding of the nature and history of the field of computational social choice.

What next? Characterising voting rules, to help choose the right one.