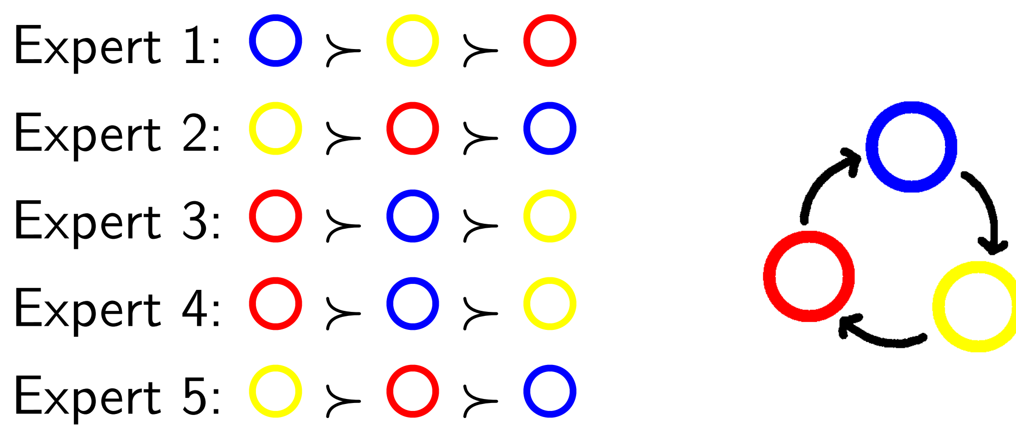
Computational Social Choice

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

[http://www.illc.uva.nl/COMSOC/]

Classic Example: The Condorcet Paradox

Social Choice Theory asks: how should we aggregate the preferences of the members of a group to obtain a "social preference"?



Marie Jean Antoine Nicolas de Caritat (1743–1794), better known as the **Marquis de Condorcet**: Highly influential Mathematician, Philosopher, Political Scientist, Political Activist. Observed that the *majority rule* may produce inconsistent outcomes ("Condorcet Paradox").

Classic Result: Arrow's Impossibility Theorem

In 1951, K.J. Arrow published his famous *Impossibility Theorem:*

Any preference aggregation mechanism for *three* or more alternatives that satisfies the axioms of *unanimity* and *IIA* must be *dictatorial*.

- Unanimity: if everyone says $A \succ B$, then so should society.
- Independence of Irrelevant Alternatives (IIA): if society says $A \succ B$ and someone changes their ranking of C, then society should still say $A \succ B$.

Kenneth J. Arrow (born 1921): American Economist; Professor Emeritus of Economics at Stanford; Nobel Prize in Economics 1972 (youngest recipient ever). His 1951 PhD thesis started modern Social Choice Theory. Google Scholar lists 9897 citations of the thesis.

Modern Applications of Social Choice Theory

Social choice-like problems arise in many applications. Examples:

- *Job Markets:* allocate junior doctors to hospitals, etc.
- *Search Engines:* determine the most important sites based on links ("votes") + to aggregate the output of several search engines
- *Semantic Web:* aggregate information from distinct sources in a consistent manner
- Others: grid computing, e-governance, e-commerce, live organ exchange, social networks, recommender systems, ...

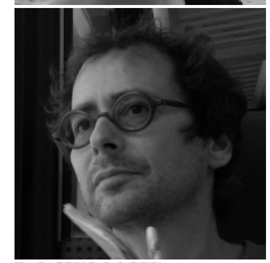
But not all of the classical assumptions will fit these new applications. So we need to develop *new models* and *ask new questions*.

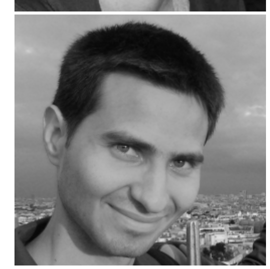
Computational Methods in Social Choice

Vice versa, techniques from computer science are useful for advancing the state of the art in social choice. Examples:

- *Algorithms and Complexity:* to develop algorithms for (complex) voting procedures + to understand the hardness of "using" them
- *Knowledge Representation:* to compactly represent the preferences of individual agents over large spaces of alternatives
- *Logic and Automated Reasoning:* to formally model problems in social choice + to automatically verify (or discover) theorems

Session Overview

Computational Social Choice =

> *looking at social choice through the "computational lens", aiming for (computational) applications*

Rest of the programme:

(1) Britta Dorn (Ulm)
Multivariate Algorithmics for Voting

(2) Jérôme Lang (Paris)
Voting in Combinatorial Domains

(3) Ioannis Caragiannis (Patras)
Computational Challenges in Fair Division

(4) Francesca Rossi (Padova)
Automated Design of Social Choice Mechanisms

(5) Péter Biró (Budapest)
Matching Schemes in Practice