Computational Social Choice 2022

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

[http://www.illc.uva.nl/~ulle/teaching/comsoc/2022/]

Plan for Today

This will be a first lecture on questions of *computational complexity* in judgment aggregation. The main problem we shall consider is that of *outcome determination:* computing the outcome for a given profile.

Aggregation rules considered:

- quota rules
- premise-based aggregation
- Kemeny rule

For a comprehensive study of the problem, refer to the paper below.

U. Endriss, R. de Haan, J. Lang, and M. Slavkovik. The Complexity Landscape of Outcome Determination in Judgment Aggregation. *JAIR*, 2020.

Checking Rationality

Before we start, let's consider a much more basic problem . . .

Maybe the most fundamental problem in aggregation is to determine whether the information supplied by an individual agent is well-formed.

RATIONALITY
Input: Agenda Φ, integrity constraint Γ, judgment set $J \in 2^{\Phi}$
Question: Is J an element of $\mathcal{J}(\Phi, \Gamma)$?

For our two models of aggregation, this boils down to:

- For formula-based judgment aggregation (with $\Gamma = \top$): Is the given *judgment set* J *complete* and *consistent*?
- For binary aggregation with integrity constraints (just literals in Φ): Is the given *judgment set* J a *model* of the *integrity constraint* Γ?

Exercise: *Analyse the complexity of the problem!*

Complexity of Checking Rationality

The complexity differs for the two models of aggregation:

Proposition 1 RATIONALITY *is in P for binary aggregation with integrity constraints, but NP-complete for formula-based JA.*

Proof:

- BA: This is model checking for propositional logic, which is easy (in the truth-table for Γ, check the row corresponding to J).* ✓
- JA: Checking consistency is SAT (though completeness is easy). ✓

*[assuming Φ includes all propositional letters occurring in Γ]

The Outcome Determination Problem

For the remainder of today we deal only with *formula-based JA*.

We first focus on *resolute* aggregation rules F (= single winner).

The *search problem* ultimately of interest is the problem of computing $F(\boldsymbol{J})$, given a profile $\boldsymbol{J}$. Here is the corresponding *decision problem:*

> $\textsc{OutDet}(F)$
> **Input:** Agenda Φ, profile $\boldsymbol{J} \in \mathcal{J}(\Phi)^n$, formula $\varphi \in \Phi$
> **Question:** Is φ an element of $F(\boldsymbol{J})$?

Exercise: *Why is F part of the problem name but Φ part of the input?*

If you can solve $\textsc{OutDet}(F)$ efficiently, then you can also solve the search problem efficiently (by deciding on each formula in turn).

Exercise: *Why would the following not be a good formulation?*
"Given Φ, $\boldsymbol{J}$, and $J \subseteq \Phi$, decide whether $F(\boldsymbol{J}) = J$."

Quota Rules

A *quota rule* F_q is defined by a function $q : \Phi \to \{0, 1, \ldots, n{+}1\}$:

$$F_q(\boldsymbol{J}) \quad = \quad \{\varphi \in \Phi \mid \#N_\varphi^{\boldsymbol{J}} \geqslant q(\varphi)\}$$

This includes, for instance, the majority rule.

For any quota rule, outcome determination is tractable:

Proposition 2 $\textsc{OutDet}(F_q)$ *is in P for every quota rule* F_q.*

Proof: Obvious. You just need to count whether $\#N_\varphi^{\boldsymbol{J}} \geqslant q(\varphi)$. ✓

*[unless applying q itself is a super-polynomial problem]

Premise-Based Aggregation

The *premise-based rule* F_{pre} for premises Φ_p and conclusions Φ_c:

$$F_{\mathrm{pre}}(\boldsymbol{J}) \quad = \quad \Delta \cup \{\varphi \in \Phi_c \mid \Delta \models \varphi\}, \\ \text{where } \Delta = \{\varphi \in \Phi_p \mid \#N_\varphi^{\boldsymbol{J}} > \frac{n}{2}\}$$

Assume premises $=$ literals and Φ closed under propositional variables (guarantees consistency and completeness, at least for odd n).

Proposition 3 *Under above assumptions,* $\textsc{OutDet}(F_{\mathrm{pre}})$ *is in P.*

Proof:

- For *premises*, this is just counting. ✓
- For *conclusions*, this is model checking for propositional logic. ✓

Outcome Determination for Irresolute Rules

Most practically useful aggregation rules actually are *irresolute* and may return a (nonempty) set of winning judgment sets:

$$F : \mathcal{J}(\Phi)^n \to 2^{(2^\Phi)} \setminus \{\emptyset\}$$

Suppose that for the *search problem* we are content with computing *one* of the judment sets in the outcome.

Exercise: *Formulate the corresponding decision problem.*

The Outcome Determination Problem

Outcome determination for irresolute aggregation rules F:

$\textsc{OutDet}^\star(F)$
Input: Agenda Φ, profile $\boldsymbol{J} \in \mathcal{J}(\Phi)^n$, subset $L \subseteq \Phi$
Question: Is there a $J^\star \in F(\boldsymbol{J})$ such that $L \subseteq J^\star$?

Observations:

- Can solve the search problem by repeatedly solving $\textsc{OutDet}^\star(F)$.
- Working with φ instead of L (as we did before) would not work.

The Kemeny Rule

The *Kemeny rule* maximises agreement with the accepted formulas:

$$F_{\text{kem}}(\boldsymbol{J}) \quad = \quad \underset{J \in \mathcal{J}(\Phi)}{\text{argmax}} \sum_{\varphi \in J} |N_\varphi^{\boldsymbol{J}}|$$

Complexity Analysis

Clearly, *outcome determination* for the *Kemeny rule* is pretty hard.
A naïve algorithm would proceed as follows:

- Go though all (complement-free and complete) judgment sets (there are *exponentially many*).
- For each of them, check whether it is consistent (*NP-complete*).
- For the consistent ones, measure total agreement (easy).
- Return the maximum (or one of the maxima, in case of ties).

But maybe there is a smarter way?

Result

We can do better than using the naïve algorithm, but the problem is still highly intractable. Next, we will develop this result:

Theorem 4 (Endriss et al., 2012) *The outcome determination problem for the Kemeny rule is Θ_2^P-complete.*

Recall that $\Theta_2^p = \mathrm{P}_{||}^{\mathrm{NP}} = \mathrm{P}^{\mathrm{NP}}[\log]$ is the class of problems solvable in *polynomial time* with a *logarithmic* number of queries to an *NP-oracle*.

Hardness can be proved via reduction from the winner determination problem for the Kemeny rule in voting (Hemaspaandra et al., 2005). We will skip this proof and only show how to prove membership.

U. Endriss, U. Grandi, and D. Porello. Complexity of Judgment Aggregation. *Journal of Artificial Intelligence Research (JAIR)*, 2012.

E. Hemaspaandra, H. Spakowski, and J. Vogel. The Complexity of Kemeny Elections. *Theoretical Computer Science*, 2005.

The Kemeny Score Problem

Recall that the Kemeny rule is searching for a consistent judgment set J that maximises the *Kemeny score:*

$$\text{KS}^{\boldsymbol{J}}(J) \quad = \quad \sum_{\varphi \in J} |N_\varphi^{\boldsymbol{J}}|$$

So consider first this problem:

$\textsc{KemenyScore}$
Input: Agenda Φ, profile $\boldsymbol{J} \in \mathcal{J}(\Phi)^n$, subset $L \subseteq \Phi$, $K \in \mathbb{N}$
Question: Is there a $J^\star \in \mathcal{J}(\Phi)$ such that $L \subseteq J^\star$ and $\text{KS}^{\boldsymbol{J}}(J^\star) \geqslant K$?

Again: $\textsc{OutDet}^\star$ is looking for the maximal such K (not given).

Easy-to-prove upper bound:

Lemma 5 $\textsc{KemenyScore}$ *is in NP.*

Proof: A suitable witness is $J^\star$, together with a model for $J^\star$. ✓

Upper Bound

We can now establish an upper bound for $\textsc{OutDet}^{\star}$:

Lemma 6 $\textsc{OutDet}^{\star}$ *for the Kemeny rule is in* Θ_2^p.

Proof: Use an *NP-oracle* that can solve $\textsc{KemenyScore}$.
Then we can solve $\textsc{OutDet}^{\star}$ by simply trying all possible values for K.

But: the *number of queries* to the oracle would be super-logarithmic, as the maximal K could be any number between 1 and $K^{\star} = \frac{|\Phi|}{2} \cdot |N|$.

But we can do a smarter search of the space of all K's (*binary search*):

- query $\textsc{KemenyScore}$ with $K := \frac{1}{2} \cdot K^{\star}$
- if YES, continue with $K := \frac{3}{2} \cdot K$ $(= \frac{3}{4} \cdot K^{\star})$
- if NO, continue with $K := \frac{1}{2} \cdot K$ $(= \frac{1}{4} \cdot K^{\star})$
- and so on

Thus, the number of (adaptive) queries is *logarithmic:* $O(\log_2 K^*)$. ✓

Islands of Tractability

Whether using the Kemeny rule *actually* is hard in practice depends, to a large extent, on the agenda Φ at hand.

Are there agendas for which outcome determination is tractable? *Yes!*

Proposition 7 *For agendas that consist of literals only, the outcome determination problem for the Kemeny rule is in P.*

Exercise: *Prove it!*

The above result is overly simplistic and of little practical use, but there also are tracability islands for more interesting classes of agendas. Refer to the work of De Haan (2018) for an in-depth analysis.

R. de Haan. Hunting for Tractable Languages for Judgment Aggregation. KR-2018.

Summary

We have discussed basic complexity questions in judgment aggregation:

- Deciding whether an individual judgment is "rational"
  (significant difference between our two models of aggregation)
- Deciding whether given formulas are accepted by a rule:
  - quota rules: polynomial
  - (simple) premise-based rule: polynomial
  - Kemeny rule: complete for parallel access to NP (hard!)

What next? Strategic behaviour in JA (including complexity issues).