Computational Social Choice: Spring 2015

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

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 *winner determination:* computing the outcome for a given profile.

Aggregation rules considered:

- quota rules
- the premise-based procedure
- the max-sum rule, a.k.a. Kemeny or distance-based rule

But we start with an even more basic problem . . .

Most of the definitions and results today come from the paper below.

U. Endriss, U. Grandi, and D. Porello. Complexity of Judgment Aggregation. *Journal of Artificial Intelligence Research (JAIR)*, 45:481–514, 2012.

Rationality

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

For our two frameworks of aggregation, RATIONALITY is defined as:

- For formula-based judgment aggregation (JA):
  Is a given *judgment set* J in fact *complete* and *consistent*?
- For binary aggregation with integrity constraints (BA):
  Is a given *ballot* B in fact a *model* of the *integrity constraint* Γ?

Proposition 1 RATIONALITY *is in* P *for BA, but* NP*-complete for JA.*

Proof:

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

Comparison of Frameworks

To date, most work on the complexity of judgment aggregation has focused on the formula-based framework, not on binary aggregation with integrity constraints.

- This is why we will focus on formula-based JA from now on.
- Most results for BA can be expected to be similar (the radical difference in complexity for RATIONALITY is an exception), but most of the details have not yet been worked out in the literature.

The Winner Determination Problem

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:*

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

If you can solve $\textsc{WinDet}(F)$, you can also solve the search problem (by deciding for all formulas in turn).

Discussion: 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.

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

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

Premise-Based Aggregation

The *premise-based procedure* $F_{\textsf{pre}}$ for premises Φ_p and conclusions Φ_c:

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

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

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

Proof:

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

Winner Determination for Irresolute Rules

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

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

How to formulate the *winner determination problem* now?

- *Search problem:* compute *one* of the winning judgment sets!
- *Decision problem:* ?

The Winner Determination Problem

Winner determination for irresolute aggregation rules F:

$\textsc{WinDet}^\star(F)$
Instance: 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 $\textsc{WinDet}^\star(F)$.
- Just checking $\varphi \in F(\boldsymbol{J})$, as we did before, would not be enough.

The Max-Sum Rule

The *max-sum rule* F (so far only defined for binary aggregation) maximises the sum of the majority strengths of the accepted formulas:

$$F(\boldsymbol{J}) \quad = \quad \underset{J \in \mathcal{J}(\Phi)}{\operatorname{argmax}} \sum_{\varphi \in J} |N_{\varphi}^{\boldsymbol{J}}| - |N_{\sim\varphi}^{\boldsymbol{J}}|$$

Note how this guarantees *consistency* by definition.

Also known under several other names. One is the *Kemeny rule*, because of its similarity to the Kemeny rule in voting theory.

Another is "the" *distance-based rule*, because it can also be defined in terms of distances (and is the best known of such rules):

$$F(\boldsymbol{J}) \quad = \quad \underset{J \in \mathcal{J}(\Phi)}{\operatorname{argmin}} \sum_{i \in \mathcal{N}} H(J, J_i),$$

where $H(J, J') = |J \setminus J'|$ is the *Hamming distance*

Complexity Analysis

Clearly, *winner determination* for the *max-sum 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 the sum of the weights (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 winner determination problem for the max-sum 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)*, 45:481–514, 2012.

E. Hemaspaandra, H. Spakowski, and J. Vogel. The Complexity of Kemeny Elections. *Theoretical Computer Science*, 349(3):382–391, 2005.

The Kemeny Score Problem

Recall that the max-sum rule is looking for a consistent judgment set J that maximises the *Kemeny score:*

$$\mathrm{KS}^{\boldsymbol{J}}(J) \quad := \quad \sum_{\varphi \in J} |N^{\boldsymbol{J}}_{\varphi}| - |N^{\boldsymbol{J}}_{\sim\varphi}|$$

So consider first this problem:

KemenyScore
Instance: 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 $\mathrm{KS}^{\boldsymbol{J}}(J^\star) \geqslant K$?

Again: WinDet* is looking for the maximal such K (not given).

Easy-to-prove upper bound:

Lemma 5 *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 WinDet:

Lemma 6 WinDet *for the max-sum rule is in* Θ_2^P.

Proof: Use an NP-*oracle* that can solve KemenyScore.
Then WinDet can be solved in *poly-time* (just try all K's).

But: the *number of queries* to the oracle would be polynomial, as the maximal K could be any number between 1 and $K^\star := \frac{|\Phi|}{2} \cdot |\mathcal{N}|$.
(This is actually ok for the "parallel access to NP reading of Θ_2^P.)

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

- query 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*: $O(\log_2 K^*)$. ✓

Summary

We have discussed basic complexity questions in judgment aggregation:

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

What next?

Later on in the course, we will see several other examples for problems arising in the context of judgment aggregation for which it is useful to understand their complexity.

Next: the problem of the *safety of the agenda*