Some Funny Complexity Results for Judgment Aggregation

Wojtek Jamroga, University of Luxembourg
(joint work with Marija Slavkovik)

ESSLLI @ Duesseldorf, August 16, 2013
Outline

1 Introduction

2 Distance-Based Judgment Aggregation

3 Complexity: Bad News

4 Positive Results

5 Conclusions
Outline

1 Introduction

2 Distance-Based Judgment Aggregation

3 Complexity: Bad News

4 Positive Results

5 Conclusions
Introduction

- It is often convenient to ascribe mental attitudes to groups of agents.
- Examples: opinion of the government, belief of a religious group, goal of a company.

collective of agents $\rightarrow$ collective agent
Judgment Aggregation

- Similar to preference aggregation & voting
- Two approaches to judgment aggregation:
  - Idealistic: specify postulates and prove impossibility
  - Pragmatic: use a reasonably good procedure
- In the latter case, complexity is important!
Distance-Based Judgment Aggregation

- Distance-based judgment aggregation defines the collective opinion as a well-behaved compromise between individual opinions
- Aggregation rules must be “well behaved” mathematically
- Does that imply that they are well-behaved computationally?
Distance-Based Judgment Aggregation

- Distance-based judgment aggregation defines the collective opinion as a well-behaved compromise between individual opinions
- Aggregation rules must be “well behaved” mathematically
- Does that imply that they are well-behaved computationally?
- Not necessarily...
Outline

1 Introduction
2 Distance-Based Judgment Aggregation
3 Complexity: Bad News
4 Positive Results
5 Conclusions
Distance-Based Judgment Aggregation

Judgment Aggregation

Definition (Judgment aggregation)

Let $N$ be a finite set of agents, $A \subseteq \mathcal{L}$ a finite agenda of issues from a propositional language $\mathcal{L}$, $C \subseteq \mathcal{L}$ a finite set of admissibility constraints, and $T$ a set of truth values.

Judgment sets ($JS$) are consistent and admissible combinations of opinions on issues from $A$, that is, all $js : A \rightarrow T$ such that there is a valuation $v \in PV$ with: (i) $val_v(\varphi) = js(\varphi)$ for every $\varphi \in A$, and (ii) $val_v(\psi) = 1$ for every $\psi \in C$.

A judgment profile is a collection of $|N|$ judgment sets, one per agent.

A judgment aggregation rule $\nabla : JS^{|N|} \rightarrow \mathcal{P}(JS) \setminus \{\emptyset\}$ aggregates opinions from all the agents into a collective judgment set (or sets).
Example: Guarding Robots

3 robots are guarding a building, and have just observed a person. Each robot must assess whether the person is authorized to be there (proposition $auth$), if it has malicious intent ($mal$), and whether to classify the event as dangerous intrusion ($intr$). Additionally, it is assumed that a non-authorized person with malicious intent implies intrusion: $¬auth \land mal \rightarrow intr$. 

```plaintext
robot 1 | 1 1 0
robot 2 | 0 0 0
robot 3 | 0 1 1
majority | 0 1 0
```
Example: Guarding Robots

3 robots are guarding a building, and have just observed a person. Each robot must assess whether the person is authorized to be there (proposition $auth$), if it has malicious intent ($mal$), and whether to classify the event as dangerous intrusion ($intr$). Additionally, it is assumed that a non-authorized person with malicious intent implies intrusion: $\neg auth \land mal \rightarrow intr$.

<table>
<thead>
<tr>
<th></th>
<th>$auth$</th>
<th>$mal$</th>
<th>$intr$</th>
</tr>
</thead>
<tbody>
<tr>
<td>robot 1</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>robot 2</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>robot 3</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>majority</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
</tbody>
</table>

Note that the most obvious aggregation rule (majority) results in an inadmissible judgment set.
Distance-Based Aggregation

Definition (Distance-based judgment aggregation)

A **distance-based aggregation rule** looks for a collective opinion that does not stray too much from the individual judgments:

\[
\nabla_{d,aggr}(jp) = \arg\min_{j\in JS} \{aggr(d(js, jp[1]), \ldots, d(js, jp[|N|])) \},
\]

where \(d\) is a **distance metric**, and \(aggr\) an **aggregation function**.
Example: Guarding Robots

<table>
<thead>
<tr>
<th></th>
<th>auth</th>
<th>mal</th>
<th>intr</th>
</tr>
</thead>
<tbody>
<tr>
<td>robot 1</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>robot 2</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>robot 3</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

Take $aggr = \sum$, $d = d_H$
A distance over $X$ is a function $d : X \times X \to \mathbb{R}^+ \cup \{0\}$ such that:

- **(minimality)** $d(x, y) = 0$ iff $x = y$,
- **(symmetry)** $d(x, y) = d(y, x)$, and
- **(triangle inequality)** $d(x, y) + d(y, z) \geq d(x, z)$.

Two well known distances over $\{0, 1\}^m$ are: the Hamming distance $d_H$, and the drastic distance $d_D$. 
Distance-Based Aggregation

Definition (Distance metric)

A distance over $X$ is a function $d : X \times X \rightarrow \mathbb{R}^+ \cup \{0\}$ such that:

- (minimality) $d(x, y) = 0$ iff $x = y$,
- (symmetry) $d(x, y) = d(y, x)$, and
- (triangle inequality) $d(x, y) + d(y, z) \geq d(x, z)$.

Two well known distances over $\{0, 1\}^m$ are: the Hamming distance $d_H$, and the drastic distance $d_D$.

Definition (Aggregation function)

An aggregation is a function $aggr : (\mathbb{R}^+)^n \rightarrow \mathbb{R}^+$ such that:

- (minimality) $aggr(0^n) = 0$, and
- (monotonicity) if $x \leq y$, then $aggr(\ldots, x, \ldots) \leq aggr(\ldots, y, \ldots)$.

Well known aggregators are: min, max, sum, and product.
Outline

1. Introduction
2. Distance-Based Judgment Aggregation
3. Complexity: Bad News
4. Positive Results
5. Conclusions
Complexity of winner set verification

Definition (winner set verification)

WINVER is the decision problem defined as follows:

Input: Agents $N$, agenda $A$, constraints $C$, judgment profile $jp \in JS^{N}(A,C)$, and judgment set $js \in JS(A,C)$;

Output: true if $js \in \nabla (jp)$, else false.
Complexity of winner set verification

Definition (winner set verification)

\( \text{WINVER} \) is the decision problem defined as follows:

**Input:** Agents \( N \), agenda \( A \), constraints \( C \), judgment profile \( j_p \in JS^{|N|}(A, C) \), and judgment set \( j_s \in JS(A, C) \);

**Output:** \textit{true} if \( j_s \in \nabla(j_p) \), else \textit{false}.

What is the complexity of \text{WINVER}?
Bad News

Theorem

There is a distance which is not Turing computable.
Bad News

Proof. We construct the Turing distance $d_{TR}$ as follows. First, we assume a standard encoding of Turing machines in binary strings; we use $TM(X)$ to refer to the machine represented by the string of bits $X \in \{0, 1\}^m$. We also assume by convention that strings starting with 0 or ending with 1 represent only machines that always halt (e.g., some TM’s with only accepting states).

Let $halts(X) = 0$ if the $TM(X)$ halts, and 1 otherwise. Now, for any $js, js' \in \{0, 1\}^m$, we take

$$d_{TR}(js, js') = d_D(js, js') + halts(h(js, js')),$$

where $d_D$ is the drastic distance, and $h(js, js')$ is the Hamming sequence for $(js, js')$. 
Proof ctd. We check that $d_{TR}$ is a distance metric:

1. $d_{TR}(js, js) = d_D(js, js) + \text{halts}(0^m) = 0$;
2. $d_{TR}(js, js') = 0 \Rightarrow d_D(js, js') = 0 \Rightarrow js = js'$;
3. $d_{TR}(js, js') = d_{TR}(js', js)$: straightforward;
4. Triangle inequality: the nontrivial case is $js \neq js' \neq js''$, then
   
   $$d_{TR}(js, js') + d_{TR}(js', js'') \geq 2 \geq d_{TR}(js, js'').$$

For incomputability, we observe that $TM(X)$ halts iff $d_{TR}(X, 0^{|X|}) \leq 1$. 
Bad News

**Theorem**

*There is a distance and an aggregation function for which WINVER is undecidable.*
**Bad News**

**Proof.** We construct a *reduction from the halting problem*. Given is a representation $X \in \{0, 1\}^m$ of a Turing machine (same assumptions on the encoding). We take $d = d_{TR}$, $aggr = \sum$.

Let $A = \{p_1, \ldots, p_m\}$ consist of $n$ unrelated atomic propositions, $C = \emptyset$, and $jp = \{0^m, X\}$. Now, for $X = 1 \ldots 0$ (the other cases of $X$ trivially halt), we have that $TM(X)$ halts iff $js = 0^m$, $X$ are the only winners. This is because the aggregate scores of $0^m$ and $X$ are 1 if $TM(X)$ halts and 2 otherwise, and no score can be less than 1. Moreover, for all other candidates $Y \in \{0, 1\}^m$ the score is at least 2, and in particular for $Y = (1)^m$ it is always 2.

Suppose now that deciding $WINVER$ terminates in finite time. Then, the halting of $TM(X)$ could be verified by $2^m$ $WINVER$ checks, i.e., also in finite time – which is a contradiction.
Outline

1. Introduction
2. Distance-Based Judgment Aggregation
3. Complexity: Bad News
4. Positive Results
5. Conclusions
Positive Results

**Theorem**

If $aggr$ and $d$ are computable in polynomial time then $WINVER$ for $\nabla_{d,aggr}$ is in $P^{NP}[2]$. 
Positive Results

Theorem

If \( \text{aggr} \) and \( d \) are computable in polynomial time then \( \text{WINVER} \) for \( \nabla_{d, \text{aggr}} \) is in \( P^{NP}[2] \).

\( P^{NP}[k] \) is the class of problems solvable by a polynomial-time deterministic Turing machine asking at most \( k \) adaptive queries to an \( NP \) oracle.

For complexity freaks: \( NP \subseteq P^{NP}[k] \subseteq \Delta^P_2 = P^{NP} \)
Algorithm: $\text{Winver}(js, jp, N, A, C, d, aggr)$

1. if $\text{Consistent}(js, A, C)$ and not $\text{ExistsBetter}(js, jp, N, A, C, d, aggr)$ then return(true) else return(false);

Oracle: $\text{Consistent}(js, A, C)$

1. guess a valuation $v \in PV$ for the atomic propositions in $A$;
2. if $\text{val}_v(\varphi) = js(\varphi)$ for every $\varphi \in A$ and $\text{val}_v(\psi) = 1$ for every $\psi \in C$ then return(true) else return(false);

Oracle: $\text{ExistsBetter}(js, jp, N, A, C, d, aggr)$

1. guess $js' \in JS$;
2. guess a valuation $v' \in PV$ for the atomic propositions in $A$;
3. if $\text{val}_{v'}(\varphi) = js'(\varphi)$ for every $\varphi \in A$ and $\text{val}_{v'}(\psi) = 1$ for every $\psi \in C$ and $\text{aggr}(d(js', jp[1]), \ldots, d(js', jp[|N|])) < \text{aggr}(d(js, jp[1]), \ldots, d(js, jp[|N|]))$ then return(true) else return(false);
**Algorithm:** $Winver(js, jp, N, A, C, d, aggr)$

Idea: ask the oracle if $js$ is consistent and admissible and whether there is no set with a better score.
Good News

For typical distances and aggregation functions, we get the following as a straightforward consequence:

Corollary

If $aggr \in \{\min, \max, \sum, \prod\}$ and $d \in \{d_H, d_D\}$ then $WINVER$ for $\nabla_{d,aggr}$ is in $P^{NP}[2]$. 
Outlines

1. Introduction
2. Distance-Based Judgment Aggregation
3. Complexity: Bad News
4. Positive Results
5. Conclusions
Conclusions

- We explore complexity bounds for judgment aggregation based on minimization of aggregate distance.

- Winner set verification for typical distance-based rules is NP-complete or slightly harder (couldn’t be easier!)

- In the general case, the complexity can be as wild as you like (=undecidable)

- Standard structural conditions on distance and aggregation functions are not enough to tame complexity – constraints on computability were needed.
Thank you for your attention!