The Complexity Landscape of Outcome Determination in Judgment Aggregation

Ulle Endriss   ULLE.ENDRISS@UVA.NL
Ronald de Haan   ME@RONALDDEHAAN.EU
Institute for Logic, Language and Computation
University of Amsterdam, The Netherlands

Jérôme Lang   LANG@LAMSADE.DAUPHINE.FR
CNRS, LAMSADE, PSL, Paris-Dauphine University, France

Marija Slavkovik   MARIJA.SLAVKOVIK@UIB.NO
Department of Information Science and Media Studies
University of Bergen, Norway

Abstract

We provide a comprehensive analysis of the computational complexity of the outcome determination problem for the most important aggregation rules proposed in the literature on logic-based judgment aggregation. Judgment aggregation is a powerful and flexible framework for studying problems of collective decision making that has attracted interest in a range of disciplines, including Legal Theory, Philosophy, Economics, Political Science, and Artificial Intelligence. The problem of computing the outcome for a given list of individual judgments to be aggregated into a single collective judgment is the most fundamental algorithmic challenge arising in this context. Our analysis applies to several different variants of the basic framework of judgment aggregation that have been discussed in the literature, as well as to a new framework that encompasses all existing such frameworks in terms of expressive power and representational succinctness.

1. Introduction

Judgment aggregation is a powerful framework for modelling a multitude of collective decision making scenarios (see the recent surveys by List, 2012; Grossi & Pigozzi, 2014; Endriss, 2016). These scenarios are modelled in terms of an *agenda*, which consists of a number of propositions you may either accept or reject. Which combinations of propositions it would be admissible to accept at the same time can be subject to complex logical constraints. This makes the design—but also the use—of rules for the aggregation of several individual judgments into a single collective judgment a challenging undertaking. In this paper we focus on the most fundamental computational problem arising when we want to use judgment aggregation in practice. Suppose we have elicited the judgments of several agents regarding the status of the propositions in the agenda. Suppose further that we have selected a specific rule for aggregating individual judgments in a principled manner that will respect all relevant constraints. Then we need to ask: What is the status of each of the propositions in the outcome returned by the rule? This is the *outcome determination problem*. In this paper, we provide a detailed analysis of the computational complexity of this problem.

Following the seminal work of List and Pettit (2002), judgment aggregation has attracted the interest of scholars in a wide variety of disciplines, ranging from Legal Theory, to Philosophy, Economics, Political Science, and Artificial Intelligence (AI). Indeed, within AI—and more specifically, within the tradition of the AI literature on belief revision—similar questions have been pondered for some

time under the heading of *belief merging* (Baral, Kraus, Minker, & Subrahmanian, 1992; Liberatore & Schaerf, 1998; Konieczny & Pino-Pérez, 2002; Konieczny, Lang, & Marquis, 2004).[1]

Example 1.1. To illustrate some of the challenges involved in aggregating individual judgments in a principled manner, let us consider the following scenario. There is a malfunctioning system that has four components: c_1, c_2, c_3, c_4. Initial investigations lead to the conclusion that either component c_1 is faulty, or component c_3 is faulty, or that both components c_2 and c_4 are faulty. Using propositional variable x_i to denote that component c_i is functioning correctly, this knowledge can be expressed as a constraint $\Gamma = \neg x_1 \vee \neg x_3 \vee (\neg x_2 \wedge \neg x_4)$, or equivalently $(\neg x_1 \vee \neg x_2 \vee \neg x_3) \wedge (\neg x_1 \vee \neg x_3 \vee \neg x_4)$. Now a number of technicians are asked to give their opinion about whether each of the four components are faulty or functioning properly: three of them say $\{\neg x_1, x_2, x_3, x_4\}$ (only component c_1 is faulty), one says $\{x_1, x_2, \neg x_3, \neg x_4\}$ (components c_3 and c_4 are faulty), three say $\{x_1, x_2, \neg x_3, x_4\}$ (only component c_3 is faulty), and four say $\{x_1, \neg x_2, x_3, \neg x_4\}$ (components c_2 and c_4 are faulty). What is the most likely explanation for the malfunctioning of the system? If we use the *majority rule* to perform aggregation of the technicians' judgments, then we may run into difficulties:

	c_1 works	c_2 works	c_3 works	c_4 works
Technicians 1–3	No	Yes	Yes	Yes
Technician 4	Yes	Yes	No	No
Technicians 5–7	Yes	Yes	No	Yes
Technicians 8–11	Yes	No	Yes	No
Majority	Yes	Yes	Yes	Yes

Thus, for the specific *profile* of individual judgments considered here, even though each individual judgment is *consistent*, in the sense of satisfying the constraint $(\neg x_1 \vee \neg x_2 \vee \neg x_3) \wedge (\neg x_1 \vee \neg x_3 \vee \neg x_4)$, the outcome is not: it violates the constraint. In other words, even though the judgment of each individual technician is in line with the knowledge about the system, the outcome returned by the majority rule is not. It declares that none of the components are faulty.

Example 1.1, which is a variant of the famous *doctrinal paradox* (Kornhauser & Sager, 1993), demonstrates that the majority rule cannot guarantee that the outcome returned will always be consistent, not even for the fairly simple aggregation scenario we considered here.[2] So in practice we will have to use more sophisticated aggregation rules. If guaranteed consistency of the outcome is a desideratum we do not want to give up on, any aggregation rule of interest must choose from the set of all consistent judgments the one that—in some sense—is best reflecting the judgments of the group as a whole. And indeed, a large number of rules of this kind—all making subtly different choices when interpreting the notion of "best reflecting the judgments of the group"—have been put forward in the literature in recent years.

The most fundamental problem of an algorithmic nature arising in the context of judgment aggregation is the problem of computing the outcome for a given profile of judgments under a

---

1. We refer to the recent review by Everaere, Konieczny, and Marquis (2017) for a discussion of the differences and similarities between judgment aggregation and belief merging. To cut a long story short, the main difference is the notion of an agenda, which is central to judgment aggregation and absent from most work on belief merging.

2. Nehring and Puppe (2007) have provided a precise characterisation of the class of aggregation scenarios that are sufficiently simple for the majority rule to be guaranteed to always return a consistent outcome. In a nutshell, these are essentially the aggregation scenarios that can be modelled using only conjunctions of clauses with at most two literals each (see also Grandi, 2012; Endriss, 2018). Observe that Example 1.1 involves a conjunction of 3-clauses.

given aggregation rule. In reference to related problems in voting theory (see, e.g., Hemaspaandra, Hemaspaandra, & Rothe, 1997) and the study of combinatorial auctions (see, e.g., Sandholm, 2002), in earlier work the task of computing the outcome of a given judgment aggregation rule for a given input has been called the *winner determination problem*. But, while in the context of an election we face the task of determining which of the candidates is the election winner and while in the context of an auction we face the task of determining which bidder "wins" (obtains) which items, this terminology arguably is misplaced in the context of judgment aggregation. This is why we propose to use the term *outcome determination problem* instead.

Contribution. We provide a comprehensive overview of the computational complexity of the outcome determination problem in judgment aggregation. Our results cover all judgment aggregation rules—eleven in total—that have received significant attention in the literature and that meet the following two criteria. First, they must be *generally applicable*, i.e., their use should not be restricted to agendas of a certain type.[3] Second, they must *guarantee consistency* of outcomes for all admissible inputs, thereby avoiding the paradox we observed in Example 1.1. Our results apply to all important frameworks for modelling judgment aggregation that have been considered in the literature, starting with the original framework of List and Pettit (2002), in which the relationships between propositions in the agenda are encoded by means of the logical form of these propositions, as well as more recent proposals in which these relationships are modelled explicitly in the form of integrity constraints. To facilitate our analysis and to obtain the strongest possible results regarding upper complexity bounds, we also introduce a new framework for judgment aggregation that integrates the features of all frameworks that have previously been considered in the literature.

Our results robustly place the outcome determination problem beyond NP, yet within the Polynomial Hierarchy—for all of the eleven rules considered. Still, when considered at a finer level of inspection, our results reveal interesting and significant variation regarding the exact complexity of evaluating different aggregation rules and we obtain completeness results for the complexity classes Θ_2^p, Δ_2^p, Σ_2^p, and Θ_3^p.[4] These findings provide important insights into the mechanics of judgment aggregation and can offer guidance for the development of practical algorithms. In particular, they indicate what existing tools for combinatorial optimisation, such as Answer Set Programming or SAT solvers, can potentially be used to implement judgment aggregation solvers (see also Section 5).

Related work. Early work in judgment aggregation focused on the philosophical and practical relevance of the doctrinal paradox and similar concerns for group decision making (see, e.g., Chapman, 1998; Pettit, 2001). Following the work of List and Pettit (2002), who provided a first formal framework in which to study such questions, most technical results in subsequent years were of an axiomatic nature and included, in particular, a flurry of impossibility theorems showing that no aggregation rule can meet certain normative requirements while at the same time guaranteeing consistency on agendas of some logical richness (see, e.g., Pauly & van Hees, 2006; Gärdenfors, 2006; Dietrich & List, 2007a; Nehring & Puppe, 2007; Dokow & Holzman, 2010). In contrast to this early focus, several of the more recent contributions to the literature have instead addressed the design of new aggregation rules that can perform satisfactorily with respect to at least some of these requirements—we cite many of them when we review judgment aggregation rules in Section 2.3.

The study of the computational complexity of judgment aggregation was initiated by Endriss, Grandi, and Porello (2012). They suggested to apply the toolbox of complexity theory not only to

---

3. Part of the literature focusses on specific agendas, e.g., so-called *conjunctive agendas* (Dietrich & Mongin, 2010).
4. We are going to recall the definitions of these complexity classes in Section 3.

Judgment aggregation rule F	Complexity of the outcome determination problem	
Condorcet rule CON	Σ_2^p-complete	(Theorems 4.1 and 4.2)
Slater rule SLA	Θ_2^p-complete	(Theorems 4.3 and 4.4)
Kemeny rule KEM	Θ_2^p-complete	(Theorems 4.5 and 4.6)
MaxHamming rule MAXHAM	Θ_2^p-complete	(Theorems 4.7 and 4.8)
AVGGEO-rule	Θ_3^p-complete	(Theorems 4.11 and 4.14)
MAXGEO-rule	Θ_3^p-complete	(Theorems 4.16 and 4.17)
Reversal-scoring rule REVSCO	Θ_2^p-complete	(Theorems 4.18 and 4.19)
Ranked-agenda rule RAN	Σ_2^p-complete	(Theorems 4.20 and 4.21)
LEXIMAX-rule	Δ_2^p-complete	(Theorems 4.22 and 4.23)
Young rule YNG	Θ_2^p-complete	(Theorems 4.24 and 4.25)
Dodgson rule DOD	Θ_2^p-complete	(Theorems 4.26 and 4.27)

Table 1: Overview of the complexity results we establish in this paper for the outcome determination problem in judgment aggregation for different judgment aggregation rules F.

the outcome determination problem, but also to two other problems: deciding whether an agent can manipulate a rule in her favour and deciding whether a given agenda is *safe* in the sense of not admitting a doctrinal paradox. They analysed the complexity of outcome determination for one specific aggregation rule, the so-called Kemeny rule. In the present paper we provide a unified view on results we proved since then in a series of conference papers (Lang & Slavkovik, 2014; Endriss & De Haan, 2015; De Haan & Slavkovik, 2017), while also closing several gaps in our understanding of the complexity landscape of outcome determination that had still remained.

In closely related work, De Haan (2016) has begun a more fine-grained analysis of outcome determination complexity using the tools of parameterized complexity. He also demonstrated how techniques from knowledge compilation and algebraic model counting can be used to identify tractable special cases that are of interest to specific applications (De Haan, 2018). The kind of complexity analysis we provide here is arguably a necessary first step for undertaking such work.

While outcome determination is the most fundamental computational problem arising in judgment aggregation, complexity theory has been fruitfully applied to a number of other such problems as well, including the aforementioned manipulation and safety problems. The complexity of strategic manipulation has been analysed in more depth by Baumeister, Erdélyi, Erdélyi, and Rothe (2015), who also introduced the problem of bribery and its complexity-theoretic analysis into the literature on judgment aggregation. Baumeister, Erdélyi, Erdélyi, Rothe, and Selker (2020) furthermore investigated the complexity of a number of control problems, where an adversary is trying to obtain a more favourable outcome by changing the set of judges in an aggregation problem.

For an overview of complexity results about voting and preference aggregation, we refer to the *Handbook of Computational Social Choice* (Brandt, Conitzer, Endriss, Lang, & Procaccia, 2016b, especially Chapters 3, 4, 5, 6, 7, and 17).

Roadmap. We begin in Section 2 by explaining the different frameworks of judgment aggregation and the different variants of the outcome determination problem that have been studied in the literature—as well as the relation between them—and defining the judgment aggregation rules that we study in this paper. Then, in Section 3, we give an overview of the computational complexity tools that we use. We present the complexity results that we establish for the outcome determination problem for judgment aggregation in Section 4—an overview of the complexity results in this paper can be found in Table 1 on page 690. We conclude in Section 5.

2. The Model

In this section, we first define and compare the different frameworks for modelling judgment aggregation scenarios we are going to work with, and we then do the same for the different variants of the outcome determination problem we are going to consider. Finally, we briefly motivate and define the various judgment aggregation rules for which we are going to analyse the complexity of outcome determination.

2.1 Judgment Aggregation Frameworks

In the literature, several different judgment aggregation frameworks have been studied. The results in this paper are applicable to several of them. Next, we describe these different frameworks using one unified language, and we briefly discuss the relation between the different frameworks. In particular, we argue that one framework is strictly more general than all the others, and that one framework is strictly less general than all the others. Therefore, in the remainder of the paper, we will prove upper bounds on the complexity for the most general variant, and lower bounds on the complexity for the most restricted variant. Before we can turn to the definition of the setting of judgment aggregation, we briefly establish some common ground on relevant concepts from propositional logic.

Propositional logic. Let $\mathcal{L}_a$ be a (countably infinite) set of propositional *variables* (or *atoms*), e.g., $\mathcal{L}_a = \{x_1, x_2, \ldots\}$. A *literal* is a propositional variable x (a *positive literal*) or a negated variable $\neg x$ (a *negative literal*). Let $\mathcal{L}_\ell$ be the set of all literals over $\mathcal{L}_a$, e.g., $\mathcal{L}_\ell = \{x_1, \neg x_1, x_2, \neg x_2, \ldots\}$. Propositional formulas are constructed from propositional variables using the Boolean operators $\wedge, \vee, \rightarrow$, and $\neg$. Let $\mathcal{L}_f$ be the set of all propositional formulas over the atoms in $\mathcal{L}_a$.

A *clause* is a finite set of literals, not containing a complementary pair $x, \neg x$, and is interpreted as the disjunction of these literals. A formula in *conjunctive normal form (CNF)* is a finite set of clauses, interpreted as the conjunction of these clauses.

A propositional formula is *doubly-negated* if it is of the form $\neg\neg\psi$. For every propositional formula φ, we let $\sim\varphi$ denote the *complement* of φ, i.e., $\sim\varphi = \neg\varphi$ if φ is not of the form $\neg\psi$, and $\sim\varphi = \psi$ if φ is of the form $\neg\psi$. For a propositional formula φ, $\mathsf{var}(\varphi)$ denotes the set of all variables occurring in φ.

We use the standard notion of *(truth) assignments* $\alpha : \mathsf{var}(\varphi) \rightarrow \{0, 1\}$ for propositional formulas and *truth* of a formula under such an assignment—for more details, we refer to textbooks in logic (e.g., Van Dalen, 2013). For any formula φ and any truth assignment α, we let $\varphi[\alpha]$ denote the formula obtained from φ by instantiating variables x in the domain of α with $\alpha(x)$ and simplifying the formula accordingly. By a slight abuse of notation, if α is defined on all variables in $\mathsf{var}(\varphi)$, we let $\varphi[\alpha]$ denote the truth value of φ under α.

Judgment aggregation. A judgment aggregation scenario is specified by a tuple $(\Phi, \Gamma_{in}, \Gamma_{out}, \boldsymbol{J})$, consisting of the following elements. The set Φ is called the *agenda*, and is a finite set $\{\varphi_1, \neg\varphi_1, \ldots, \varphi_m, \neg\varphi_m\}$, where each φ_i is a non-negated propositional formula. We call $[\Phi] = \{\varphi_1, \ldots, \varphi_m\}$, the set of non-negated formulas in Φ, the *pre-agenda* corresponding to Φ.

Then there are two *integrity constraints* Γ_{in} and Γ_{out}, which are both propositional formulas: the *input constraint* Γ_{in}—that expresses which individual opinions are allowed—and the *output constraint* Γ_{out}—that expresses which collective opinions are allowed.

Opinions (both individual and collective) are represented by *judgment sets* $J \subseteq \Phi$, i.e., by subsets of the agenda. A judgment set J is *complete* if for each $\varphi \in [\Phi]$ it holds that either $\varphi \in J$ or $\neg\varphi \in J$. A judgment set J is Γ_{in}-*consistent* if $J \cup \{\Gamma_{in}\}$ is a satisfiable set of formulas, and J is Γ_{out}-*consistent* if $J \cup \{\Gamma_{out}\}$ is a satisfiable set of formulas. For judgment sets J representing individual opinions, we will say that J is *consistent* if it is Γ_{in}-consistent, and for judgment sets J representing collective opinions, we will say that J is *consistent* if it is Γ_{out}-consistent. We use $\mathcal{J}(\Phi, \Gamma)$ to denote all judgment sets $J \subseteq \Phi$ that are complete and Γ-consistent, for $\Gamma \in \{\Gamma_{in}, \Gamma_{out}\}$. When Φ and Γ are clear from the context, then for any set $S \subseteq \Phi$ we let $ext(S) = \{ J \in \mathcal{J}(\Phi, \Gamma) \mid J \supseteq S \}$ denote the set of all complete and consistent *extensions* of S.

Finally, we have a profile $\boldsymbol{J} = (J_1, \ldots, J_n)$, for some $n \in \mathbb{N}$, that consists of a sequence of individual judgment sets $J_i \in \mathcal{J}(\Phi, \Gamma_{in})$ representing the judgments of the individual agents. We use $\mathcal{J}(\Phi, \Gamma_{in})^+$ to denote the set of all such profiles (of any length $n \in \mathbb{N}$). By a slight abuse of notation, we will write $J_i \in \boldsymbol{J}$ to express that J_i is one of the individual judgment sets in the profile $\boldsymbol{J}$. For each $\varphi \in \Phi$, we let $N(\boldsymbol{J}, \varphi) = |\{ i \mid \varphi \in J_i \}|$ denote the number of judgment sets in the profile $\boldsymbol{J}$ that contain φ.

A *judgment aggregation rule* is a function F that maps every profile $\boldsymbol{J} \in \mathcal{J}(\Phi, \Gamma_{in})^+$ of complete and consistent individual judgment sets to a non-empty set $F(\boldsymbol{J})$ of collective judgment sets. For the sake of readability, we only write $F(\boldsymbol{J})$, even though the outcome of F also depends on Φ and typically also on Γ_{out} (and for certain rules even on Γ_{in}). Judgment aggregation rules are defined for profiles with any number of agents. When a rule F has the property that for every profile $\boldsymbol{J} \in \mathcal{J}(\Phi, \Gamma_{in})^+$ it holds that $F(\boldsymbol{J}) \subseteq \mathcal{J}(\Phi, \Gamma_{out})$—i.e., that every selected collective judgment set is (Γ_{out}-)consistent—we say that F is *consistent*. We say that F is *resolute* if it holds that $F(\boldsymbol{J})$ is a singleton for every profile $\boldsymbol{J} \in \mathcal{J}(\Phi, \Gamma_{in})^+$.

Frameworks. Having defined the general framework of judgment aggregation that we will use in this paper, we are now ready to identify various restricted variants that have been studied in the literature—including the most restricted variant that we will use to establish all lower bound results in this paper. We distinguish between the following frameworks:

(1) The agenda $\Phi \subseteq \mathcal{L}_\ell$ contains only literals, and $\Gamma_{in} = \Gamma_{out} \in \mathcal{L}_f$ is a formula containing only variables from $\mathsf{var}(\Phi)$.

   - This framework has been considered by (a.o.) Grandi (2012) and Grandi and Endriss (2013) under the name of *binary aggregation with integrity constraints*.

(2) The agenda $\Phi \subseteq \mathcal{L}_\ell$ contains only literals, and $\Gamma_{in} = \Gamma_{out} \in \mathcal{L}_f$ is an arbitrary constraint, which may in particular refer to variables that do not occur in the agenda.

   - This framework has been considered by (a.o.) Endriss, Grandi, De Haan, and Lang (2016) and De Haan (2018).

(3) The agenda $\Phi \subseteq \mathcal{L}_f$ contains arbitrary formulas, and $\Gamma_{\text{in}} = \Gamma_{\text{out}} = \top$ is a trivial constraint.

  • This is the classical "formula-based" framework that has been considered by (a.o.) List and Pettit (2002), Dietrich and List (2007a), Miller and Osherson (2009), Endriss et al. (2012), and Endriss et al. (2016).

(4) The agenda $\Phi \subseteq \mathcal{L}_f$ contains arbitrary formulas, and $\Gamma_{\text{in}} = \Gamma_{\text{out}} \in \mathcal{L}_f$ is an arbitrary constraint.

  • This framework has been considered by (a.o.) Dietrich and List (2008), Lang and Slavkovik (2014), Endriss and De Haan (2015), Endriss et al. (2016), and De Haan and Slavkovik (2017).

(5) The agenda $\Phi \subseteq \mathcal{L}_\ell$ contains only literals, and $\Gamma_{\text{in}}, \Gamma_{\text{out}} \in \mathcal{L}_f$ are (possibly distinct) formulas containing only variables from $\text{var}(\Phi)$.

  • This framework has been considered by Endriss (2018), who refers to Γ_{in} as the *rationality constraint* and to Γ_{out} as the *feasibility constraint*.

We note that the model of judgment aggregation we have defined in this paper is more general than any of the five specific frameworks defined above—and indeed more general than any framework that has previously been considered in the literature. Let us now explicitly add this most general variant to our list of frameworks:

(6) The agenda $\Phi \subseteq \mathcal{L}_f$ contains arbitrary formulas, and $\Gamma_{\text{in}}, \Gamma_{\text{out}} \in \mathcal{L}_f$ are arbitrary (possibly distinct) constraints.

Even though these six frameworks might seem very different, they are closely related to each other. Endriss et al. (2016) carried out a formal investigation of the relation between variants (1)–(4). They showed that variants (2)–(4) are in fact polynomial-time translatable to each other—under the assumption that at least one satisfying truth assignment to the constraint $\Gamma_{\text{in}} = \Gamma_{\text{out}}$ is given. In other words, variants (2)–(4) are equivalent in terms of generality. Moreover, they showed that variant (1) is strictly less general than variants (2)–(4), under the common complexity-theoretic assumption that the Polynomial Hierarchy is strict. Variant (5) of the framework is more general than variant (1): the latter can easily be seen to be a special case of variant (5), where $\Gamma_{\text{in}} = \Gamma_{\text{out}}$. Variant (6), finally, is easily seen to be more general than any of the other frameworks: variants (4) and (5) are special cases of variant (6). In Figure 1, we provide an overview of the relation between the different variants of the framework in terms of their generality.

In view of these relationships between the different frameworks, in the body of this paper we will use the restrictions of variant (1) for the lower bounds (hardness results) we establish, and we will use the general setting of variant (6) to show upper bounds (membership results).

2.2 The Problem of Outcome Determination

Similar to the variety in judgment aggregation frameworks that have been studied in the literature, also a variety of different formalisations of the computational problem of outcome determination have been put forward and analysed. Next, we will describe these different formalisations, discuss the relation between them, and identify (once more) a most general and a most restricted variant that we will use for the upper and lower bounds, respectively. We will focus mostly on decision problems, and
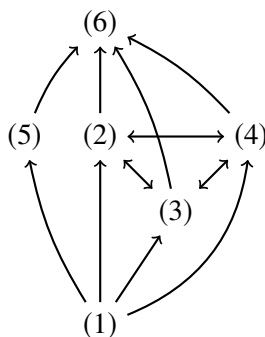
693

Figure 1: An overview of judgment aggregation frameworks (1)–(6) in terms of their generality. An arrow from x to y indicates that variant y is at least as general as x.

only briefly discuss their relationship to search problems. For such decision problems, we distinguish between credulous and skeptical variants, and consider more and less restricted variants. Fix a judgment aggregation rule F.

(a) *Credulous outcome determination, general variant.*

> **Input:** An agenda Φ, constraints Γ_{in} and Γ_{out}, a profile $\boldsymbol{J} \in \mathcal{J}(\Phi, \Gamma_{\text{in}})$, and subsets $L, L_1, \ldots, L_u \subseteq \Phi$ of the agenda, for $u \geq 0$.
> **Question:** Is there a judgment set $J^* \in F(\boldsymbol{J})$ such that $L \subseteq J^*$ and $L_i \nsubseteq J^*$ for each $i \in \{1, \ldots, u\}$?

This variant has been considered by Endriss and De Haan (2015), De Haan (2016), and De Haan and Slavkovik (2017).

A restricted version of this variant, where $u = 0$, has been considered by Endriss et al. (2012), Endriss et al. (2016), and De Haan (2018).

(b) *Credulous outcome determination, restricted variant.*

> **Input:** An agenda Φ, constraints Γ_{in} and Γ_{out}, a profile $\boldsymbol{J} \in \mathcal{J}(\Phi, \Gamma_{\text{in}})$, and a formula $\varphi^* \in \Phi$ from the agenda.
> **Question:** Is there a judgment set $J^* \in F(\boldsymbol{J})$ such that $\varphi^* \in J^*$?

This variant has not been explicitly identified in the literature, but it has been used implicitly to establish lower bounds—e.g., by Endriss and De Haan (2015), De Haan (2016, 2018), and De Haan and Slavkovik (2017).

(c) *Skeptical outcome determination, restricted variant.*

> **Input:** An agenda Φ, constraints Γ_{in} and Γ_{out}, a profile $\boldsymbol{J} \in \mathcal{J}(\Phi, \Gamma_{\text{in}})$, and a formula $\varphi^* \in \Phi$ from the agenda.
> **Question:** Is it the case that $\varphi^* \in J^*$ for all judgment sets $J^* \in F(\boldsymbol{J})$?

This variant has been considered by Lang and Slavkovik (2014).

Variant (a) of the decision problem is useful if one is in fact interested in solving a search variant of the problem—that is, computing one or more judgment sets in the outcome $F(\boldsymbol{J})$—as we will explain in more detail below, after defining search variants of the problem. Variants (b) and (c) are directed at properties of the entire set of outcomes, rather than at a single judgment set in the outcome: they ask whether a given formula φ^* is possibly or certainly accepted, respectively, if the ultimate collective judgment set is selected from the set of outcomes. Thus, variants (b) and (c) are useful if one is interested in such global properties.[5]

As pointed out by Lang and Slavkovik (2014), since each judgment aggregation rule F selects a set of complete judgment sets, it holds for every agenda Φ, every profile $\boldsymbol{J}$, and every formula $\varphi^* \in A$ that $\varphi^* \in J^*$ for all $J^* \in F(\boldsymbol{J})$ if and only if it is not the case that $\sim\!\varphi^* \in J^*$ for some $J^* \in F(\boldsymbol{J})$. Therefore, variant (b) is straightforwardly polynomial-time reducible to the co-problem of variant (c), and *vice versa:* map any input $(\Phi, \Gamma_{\text{in}}, \Gamma_{\text{out}}, \boldsymbol{J}, \varphi^*)$ to the the input $(\Phi, \Gamma_{\text{in}}, \Gamma_{\text{out}}, \boldsymbol{J}, \sim\!\varphi^*)$. Thus, if the problem variant (b) is complete for some complexity class K—for some variant of the judgment aggregation framework and some judgment aggregation rule F—then the problem variant (c) is complete for coK—for the same variant and rule. In other words, variants (b) and (c) are equivalent from the perspective of computational complexity (modulo complementation of complexity classes). Moreover, variant (b) is a restricted case of variant (a): when $u = 0$ and $L = \{\varphi^*\}$, the two problem variants coincide. Therefore, variant (a) is more general than variant (b).

In the remainder of the paper, we will only consider variants (a) and (b). We will use variant (a) of the problem to establish upper bounds (membership results) and variant (b) to establish lower bounds on the complexity (hardness results).

Search Problems. As pointed out by Endriss and De Haan (2015), the computational complexity obtained for decision problems formalising the task of computing outcomes does not directly lead to corresponding (tight) results on the computational complexity of the (search) problem of computing an outcome.[6] Next, we consider two search variants of the computational problem of outcome determination, defined along the lines of the credulous variants (a) and (b) of the decision problem.

(d) *Search, general variant.*

> **Input:** An agenda Φ, constraints Γ_{in} and Γ_{out}, a profile $\boldsymbol{J} \in \mathcal{J}(\Phi, \Gamma_{\text{in}})$, and subsets $L, L_1, \ldots, L_u \subseteq \Phi$ of the agenda, for $u \geq 0$.
> **Output:** A judgment set $J^* \in F(\boldsymbol{J})$ such that $L \subseteq J^*$ and $L_i \nsubseteq J^*$ for each $i \in \{1, \ldots, u\}$, if one exists.

---

5. We note that defining a general, skeptical variant of the outcome determination problem would not be useful: if we are interested in deciding whether all outcomes include or exclude certain sets of formulas, we can always reduce this to a series of decision problems on individual formulas (the restricted variant of the problem). The same is not possible for the credulous variant of the problem, because we cannot be certain that the queries to the restricted variant would always be answered with respect to *the same* specific outcome.

6. In the related setting of voting, there are in fact cases where the complexity of search problems and their corresponding decision problems differs (Hemaspaandra, Hemaspaandra, & Menton, 2020).

(e) *Search, restricted variant.*

> **Input:** An agenda Φ, constraints Γ_{in} and Γ_{out}, a profile $\boldsymbol{J} \in \mathcal{J}(\Phi, \Gamma_{in})$, and a formula $\varphi^* \in \Phi$ from the agenda.
> **Output:** A judgment set $J^* \in F(\boldsymbol{J})$ such that $\varphi^* \in J^*$, if one exists.

We can solve both of these search variants of the problem if we have access to an oracle (or an efficient algorithm) for the decision variant (a). We can do this by iteratively constructing the set J^* and querying whether the partially constructed set J' can be extended with some $\varphi \in \Phi$ by letting $L = J' \cup \{\varphi\}$. This gives us a polynomial-time Turing reduction[7] from the variants (d) and (e) to the decision problem (a). (In fact, we can extend this approach to enumerate different judgment sets $J^* \in F(\boldsymbol{J})$ with polynomial delay by using an oracle for the decision problem of variant (a)—this is what the sets $L_1, \ldots, L_u$ are useful for.) Variant (b) cannot be similarly used to solve the search variants (d) or (e). This is because in variant (b) we can only ask about the inclusion of a single formula in a judgment set in the outcome, and so we cannot iteratively construct the set J^*.

The Turing reducibility from variants (d) and (e) to variant (a) is enough to establish that, if variant (a) is polynomial-time solvable, then the search problems (d) and (e) are also polynomial-time solvable. Conversely, it holds that whenever variant (a) is not polynomial-time solvable unless $\mathsf{P} = \mathsf{NP}$—e.g., if variant (a) is NP-hard—then also the search problems (d) and (e) are not polynomial-time solvable unless $\mathsf{P} = \mathsf{NP}$.

However, as it will turn out, for many judgment aggregation rules, the computational complexity of outcome determination lies at some level in the Polynomial Hierarchy between NP and Δ_2^p—e.g., completeness for NP, Θ_2^p, or Δ_2^p.[8] When using polynomial-time Turing reductions (rather than many-to-one reductions[9]), the classes NP, Θ_2^p, and Δ_2^p all coincide. Thus, if we were to rely only on the Turing reducibility from variants (d) and (e) to variant (a) for establishing the complexity of the search problem, we would lose some complexity-theoretic insights and distinctions that we establish by showing, say, Θ_2^p-completeness.

To preserve these subtle distinctions also for the search problem formalisations, we would need to prove these results directly for the search problems, using appropriate notions (e.g., classes, reductions, complete problems)—as has been done by Endriss and De Haan (2015). We will not go into detail on this in the remainder of this paper. It suffices to note that all proofs we establish in this paper (both lower and upper bounds) can be extended to the search variants (d) and (e) of the problem of outcome determination for judgment aggregation.

---

7. A polynomial-time Turing reduction from problem A to problem B is a polynomial-time algorithm that has access to an oracle for problem B and that solves problem A. (We consider algorithms with access to an oracle in more detail in Section 3.)

8. We will define the Polynomial Hierarchy and the complexity classes Θ_2^p and Δ_2^p in Section 3.

9. A polynomial-time many-to-one reduction from a decision problem A to a decision problem B is a polynomial-time algorithm R that, when given an input x for A, computes an input $f(x)$ for B such that $x \in A$ if and only if $f(x) \in B$. This is a special case of a polynomial-time Turing reduction. Given a many-to-one reduction R, we can construct a polynomial-time Turing reduction R' as follows: to solve A, for any input x, R' first computes $f(x)$ and then uses the oracle to decide if $f(x) \in B$, which is the case if and only if $x \in A$.

2.3 Judgment Aggregation Rules

Next, we recall the definitions of the most important judgment aggregation rules that have been considered in the literature. We restrict attention to rules that are (i) consistent and (ii) applicable to any kind of agenda.[10] As to date for several of these rules there is no commonly accepted terminology to identify them, we are going to reference the mostly widely used names in each such case. For a more extensive presentation of the landscape of judgment aggregation rules we refer to the taxonomy published by Lang, Pigozzi, Slavkovik, van der Torre, and Vesic (2017).

To illustrate the judgment aggregation rules, we will use the following running example.

Example 2.1. Consider the judgment aggregation scenario $(\Phi^\dagger, \Gamma^\dagger_{in}, \Gamma^\dagger_{out}, J^\dagger)$, where the agenda $\Phi^\dagger$ is based on the pre-agenda $[\Phi^\dagger] = \{x_1, x_2, x_3, x_4\}$ containing four propositional variables, and the input and output constraints $\Gamma^\dagger_{in} = \Gamma^\dagger_{out} = (\neg x_1 \vee \neg x_2 \vee \neg x_3) \wedge (\neg x_1 \vee \neg x_3 \vee \neg x_4)$ coincide. The profile $J^\dagger = (J_1, \ldots, J_{11})$ is given in Figure 2, and consists of the judgment sets $J_1, J_2, J_3 = \{\neg x_1, x_2, x_3, x_4\}$, $J_4 = \{x_1, x_2, \neg x_3, \neg x_4\}$, $J_5, J_6, J_7 = \{x_1, x_2, \neg x_3, x_4\}$, and $J_8, J_9, J_{10}, J_{11} = \{x_1, \neg x_2, x_3, \neg x_4\}$. (This is the same scenario as used in Example 1.1, formalised in the framework of judgment aggregation.)

$J^\dagger$	J_1–J_3	J_4	J_5–J_7	J_8–J_{11}
x_1	0	1	1	1
x_2	1	1	1	0
x_3	1	0	0	1
x_4	1	0	1	0

Figure 2: The profile $J^\dagger$ in the running example that accompanies the description of judgment aggregation rules in Section 2.3.

As we have seen in the introduction, the majority rule is not consistent. Nevertheless, the notion of majority support is certainly very appealing, which is why several important rules are defined in terms of this notion. We define the *majoritarian judgment set* $m(J)$ associated with a profile $J = (J_1, \ldots, J_n)$ of length n as the set of agenda elements supported by a strict majority of agents:

$$m(J) \quad = \quad \{\, \varphi \in \Phi \mid N(J, \varphi) > \tfrac{n}{2} \,\}$$

Several judgment aggregation rules amount to selecting consistent judgment sets that agree with $m(J)$ as much as possible. For any given profile J, let $\max(m(J), \subseteq)$ denote the set of all consistent sets $S \subseteq m(J)$ for which there is no consistent $S' \subseteq m(J)$ with $S' \supset S$, and let $\max(m(J), \leq)$ denote the set of all consistent sets $S \subseteq m(J)$ for which there is no consistent $S' \subseteq m(J)$ with $|S'| > |S|$.

---

10. The former restriction rules out the popular *quota rules* (Dietrich & List, 2007b), under which a formula is accepted if and only if the number of agents supporting that formula exceeds a certain quota. The latter restriction rules out the popular *premise-based rule* (Pettit, 2001), which is applicable only when the agenda can be divided into premises and conclusions, and which accepts those premises that are supported by a majority of the agents and those conclusions that are logically entailed by the accepted premises. We note that outcome determination is algorithmically trivial for the quota rules. For the premise-based rule the complexity of outcome determination reduces to the complexity of determining whether a given conclusion follows from a given set of premises, which is a trivial task for the *conjunctive agendas* considered in essentially all of the literature discussing applications of this rule (see, e.g., Dietrich & Mongin, 2010; Hartmann & Sprenger, 2012; Terzopoulou & Endriss, 2018).

Definition 1 (Condorcet rule). *The Condorcet rule* CON *returns, for any given profile $\boldsymbol{J}$, exactly those complete and consistent judgment sets that agree with the majoritarian judgment set of $\boldsymbol{J}$ to a maximal extent (with respect to set inclusion):*

$$\text{CON}(\boldsymbol{J}) \quad = \quad \{\, J \in ext(S) \mid S \in \max(m(\boldsymbol{J}), \subseteq) \,\}$$

For the profile $\boldsymbol{J}^{\dagger}$ in Example 2.1, the Condorcet rule returns $\text{CON}(\boldsymbol{J}^{\dagger}) = \{\{x_1, x_2, \neg x_3, x_4\}, \{x_1, \neg x_2, x_3, \neg x_4\}, \{\neg x_1, x_2, x_3, x_4\}\}$.

We stress that the definition of this rule is parametric in both the agenda Φ and the output constraint Γ_{out}, even if these parameters do not explicitly show up in our simplified notation. By definition, the Condorcet rule will always return a nonempty subset of $\mathcal{J}(\Phi, \Gamma_{\text{out}})$. The same is true for all other aggregation rules to be defined in the sequel. The Condorcet rule is named so due to its connection to the notion of a Condorcet winner in the context of preference aggregation (Nehring, Pivato, & Puppe, 2014). It has also been called the *maximal subagenda rule* (Lang, Pigozzi, Slavkovik, & van der Torre, 2011) and the *max-set rule* (Endriss, 2018).

Definition 2 (Slater rule). *The Slater rule* SLA *returns, for any given profile $\boldsymbol{J}$, exactly those complete and consistent judgment sets that agree with the majoritarian judgment set of $\boldsymbol{J}$ on a maximal number of formulas:*

$$\text{SLA}(\boldsymbol{J}) \quad = \quad \{\, J \in ext(S) \mid S \in \max(m(\boldsymbol{J}), \leq) \,\}$$

For the profile $\boldsymbol{J}^{\dagger}$ in Example 2.1, the Slater rule returns the outcome $\text{SLA}(\boldsymbol{J}^{\dagger}) = \{\{x_1, x_2, \neg x_3, x_4\}, \{\neg x_1, x_2, x_3, x_4\}\}$.

The name of this rule reflects the fact that it is a direct generalisation of the Slater rule for preference aggregation (Slater, 1961; Brandt, Brill, & Harrenstein, 2016a). It is also known as the *endpoint rule* (Miller & Osherson, 2009), the *maxcard subagenda rule* (Lang et al., 2011), the *maxcard Condorcet rule* (Lang et al., 2017), and the *max-num rule* (Endriss, 2018).

Rather than trying to maximise agreement with the majoritarian judgment set, we may also try to maximise agreement with (or minimise distance to) the profile itself. For any two judgment sets J and J' (that each include exactly one of φ and $\neg \varphi$ for every $\varphi \in [\Phi]$), their *Hamming distance* $H(J, J')$ is defined as the number of elements of the pre-agenda $[\Phi]$ on which they disagree:

$$H(J, J') \quad = \quad |J \setminus J'| \quad = \quad |J' \setminus J|$$

Definition 3 (Kemeny rule). *The Kemeny rule* KEM *returns, for any given profile $\boldsymbol{J}$, exactly those complete and consistent judgment sets that maximise agreement with $\boldsymbol{J}$ and thus minimise the sum of the Hamming distances to the individual judgments in $\boldsymbol{J}$:*

$$\text{KEM}(\boldsymbol{J}) \quad = \quad \operatorname*{argmax}_{J \in \mathcal{J}(\Phi, \Gamma_{\text{out}})} \sum_{\varphi \in J} N(\boldsymbol{J}, \varphi) \quad = \quad \operatorname*{argmin}_{J \in \mathcal{J}(\Phi, \Gamma_{\text{out}})} \sum_{J_i \in \boldsymbol{J}} H(J, J_i)$$

For the profile $\boldsymbol{J}^{\dagger}$ in Example 2.1, the Kemeny rule returns $\text{KEM}(\boldsymbol{J}^{\dagger}) = \{\{x_1, x_2, \neg x_3, x_4\}\}$. The winning judgment set $\{x_1, x_2, \neg x_3, x_4\}$ has a total Hamming distance of 19 to the profile $\boldsymbol{J}$. The two judgment sets that follow are $\{x_1, \neg x_2, x_3, \neg x_4\}$—with a total distance of 20—and $\{\neg x_1, x_2, x_3, x_4\}$— with a total distance of 21.

This rule generalises the Kemeny rule for preference aggregation (Kemeny, 1959; Fischer, Hudry, & Niedermeier, 2016). It is also known as the *distance-based rule* (Pigozzi, 2006; Endriss et al.,

2012), the *prototype rule* (Miller & Osherson, 2009), the *median rule* (Nehring et al., 2014), the *maximum-weight subagenda rule* (Lang et al., 2011), and the *max-sum rule* (Endriss, 2018).

The Kemeny rule may be interpreted as minimising the *average* (Hamming) distance to the individual judgment sets. Alternatively, we may minimise the *maximum* distance to any of the individual judgment sets to, in some sense, help the agent that is worst off (Lang et al., 2011).

Definition 4 (MaxHamming rule). *The MaxHamming rule* MaxHam *returns, for any given profile $\boldsymbol{J}$, exactly those complete and consistent judgment sets that minimise the maximum of the Hamming distance to the individual judgments in $\boldsymbol{J}$:*

$$\text{MaxHam}(\boldsymbol{J}) = \operatorname*{argmin}_{J \in \mathcal{J}(\Phi, \Gamma_{\text{out}})} \max_{J_i \in \boldsymbol{J}} H(J, J_i)$$

For the profile $\boldsymbol{J}^{\dagger}$ in Example 2.1, the MaxHamming rule returns the following set:

$$
\begin{aligned}
\text{MaxHam}(\boldsymbol{J}^{\dagger}) = \quad &\{\{x_1, x_2, \neg x_3, x_4\}, \{x_1, x_2, \neg x_3, \neg x_4\}, \{x_1, \neg x_2, x_3, \neg x_4\}, \\
&\{x_1, \neg x_2, \neg x_3, x_4\}, \{\neg x_1, x_2, x_3, x_4\}, \{\neg x_1, x_2, x_3, \neg x_4\}, \\
&\{\neg x_1, x_2, \neg x_3, \neg x_4\}, \{\neg x_1, \neg x_2, \neg x_3, x_4\}, \{\neg x_1, \neg x_2, \neg x_3, \neg x_4\}\}
\end{aligned}
$$

Variants of the MaxHamming rule have been studied in preference aggregation—under the names *maximum rank aggregation* (Bachmaier, Brandenburg, Gleißner, & Hofmeier, 2015) and *MinMax rank aggregation* (Li & Milenkovic, 2017)—and in approval (committee) elections—under the name *minimax approval voting* (Brams, Kilgour, & Sanver, 2007).

The Hamming distance is but one way of measuring the distance between two judgment sets. It makes the implicit assumption that the most elementary way of changing a judgment set is to "flip" the judgment on exactly one agenda item. But one could argue that this is not appropriate when such a flip would render the judgment set inconsistent. An alternative approach, first proposed by Duddy and Piggins (2012), is to measure distance in terms of elementary changes that each amount to moving to a minimally different—yet consistent—judgment set. This idea leads to the following definition. Consider the undirected graph with the set of vertices $\mathcal{J}(\Phi, \Gamma_{\text{out}})$ in which there is an edge between vertices J and J' if and only if $2^{J \cup J'} \cap \mathcal{J}(\Phi, \Gamma_{\text{out}}) = \{J, J'\}$, i.e., if and only if there is no other complete and consistent judgment set "between" J and J' that we could construct by mixing the elements of these two sets. Call this graph the *geodesic graph*. Let G be defined as the function that returns the *geodesic distance* (the length of the shortest path) between any two vertices in the geodesic graph.

Definition 5 (Rules based on the geodesic distance). *The aggregation rules based on the geodesic distance G,* AvgGeo *and* MaxGeo*, are defined as follows, for any given profile $\boldsymbol{J}$:*

$$\text{AvgGeo}(\boldsymbol{J}) = \operatorname*{argmin}_{J \in \mathcal{J}(\Phi, \Gamma_{\text{out}})} \sum_{J_i \in \boldsymbol{J}} G(J, J_i)$$

$$\text{MaxGeo}(\boldsymbol{J}) = \operatorname*{argmin}_{J \in \mathcal{J}(\Phi, \Gamma_{\text{out}})} \max_{J_i \in \boldsymbol{J}} G(J, J_i)$$

For the profile $\boldsymbol{J}^{\dagger}$ in Example 2.1, the AvgGeo rule returns $\text{AvgGeo}(\boldsymbol{J}^{\dagger}) = \{\{x_1, x_2, \neg x_3, x_4\}\}$, while the MaxGeo rule returns the following outcome:

$$
\begin{aligned}
\text{MaxGeo}(\boldsymbol{J}^{\dagger}) = \quad &\{\{x_1, x_2, \neg x_3, x_4\}, \{x_1, x_2, \neg x_3, \neg x_4\}, \{x_1, \neg x_2, x_3, \neg x_4\}, \\
&\{x_1, \neg x_2, \neg x_3, x_4\}, \{\neg x_1, x_2, x_3, x_4\}, \{\neg x_1, x_2, x_3, \neg x_4\}, \\
&\{\neg x_1, x_2, \neg x_3, \neg x_4\}, \{\neg x_1, \neg x_2, \neg x_3, x_4\}, \{\neg x_1, \neg x_2, \neg x_3, \neg x_4\}\}
\end{aligned}
$$

The geodesic graph for the agenda $\Phi^\dagger$ and the output constraint $\Gamma^\dagger_{out}$ are shown in Figure 3. The cumulative geodesic distance from $\{x_1, x_2, \neg x_3, x_4\}$ to the judgment sets J_i in the profile $\boldsymbol{J}^\dagger$ is 19. All other judgment sets in $\mathcal{J}(\Phi, \Gamma_{out})$ have a larger cumulative geodesic distance to the judgment sets in $\boldsymbol{J}^\dagger$. The judgment sets in $\text{MAXGEO}(\boldsymbol{J}^\dagger)$ all have geodesic distance at most three to each judgment set in $\boldsymbol{J}^\dagger$. Moreover, there exists no judgment set in $\mathcal{J}(\Phi, \Gamma_{out})$ that has a geodesic distance of two or less to each judgment set in $\boldsymbol{J}^\dagger$.
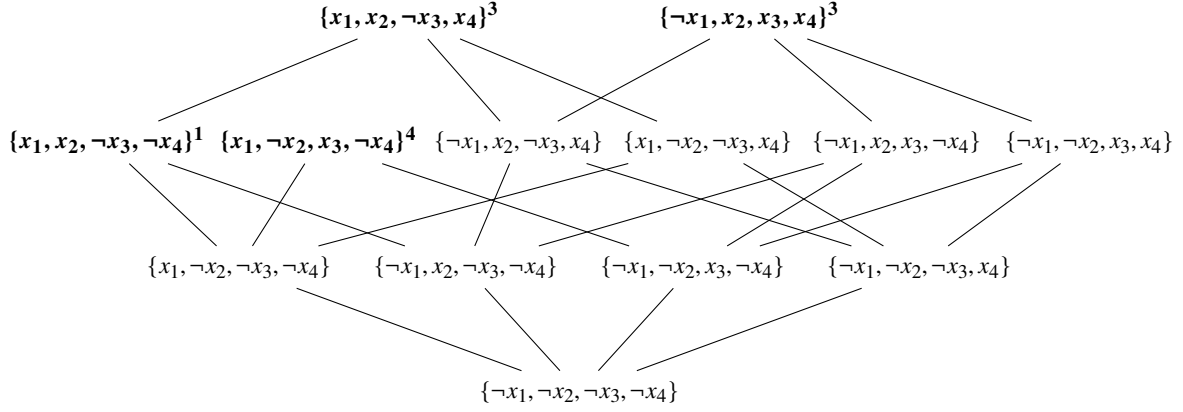

Figure 3: The geodesic graph for the agenda $\Phi^\dagger$ and the output constraint $\Gamma^\dagger_{out}$ from Example 2.1. The judgment sets from the agenda $\boldsymbol{J}^\dagger$ are highlighted in bold, and for each of them the number of times that they appear in $\boldsymbol{J}^\dagger$ is indicated.

Thus, the Kemeny rule is to the MaxHamming rule what AVGGEO is to MAXGEO. Of course, other distance measures, besides H and G, could be considered as well (Slavkovik & Ågotnes, 2014), but doing so here would go beyond the scope of the present paper.

Another approach to defining an aggregation rule is to associate every pair of a judgment set J_i and a formula φ with a *score* that indicates how strongly J_i supports φ, and to then return exactly those complete and consistent judgment sets that maximise the sum of those scores (Dietrich, 2014). Note that the Kemeny rule is such a scoring rule, namely for the trivial scoring function that assigns score 1 to (J_i, φ) if and only $\varphi \in J_i$ (and score 0 otherwise). Besides the Kemeny rule, the most important representative of the family of scoring rules is the *reversal-scoring rule*, first proposed by Dietrich (2014) and defined next. Its interest stems from the fact that it may be interpreted as a generalisation of the Borda voting rule (Borda, 1781; Zwicker, 2016).

Definition 6 (Reversal-scoring rule). *The reversal-scoring rule* REVSCO *is the scoring rule under which the score a formula receives from an individual judgment set is equal to the number of agenda items on which judgment has to be inverted before that formula can be rejected:*

$$\text{REVSCO}(\boldsymbol{J}) \;=\; \operatorname*{argmax}_{J \in \mathcal{J}(\Phi, \Gamma_{out})} \sum_{J_i \in \boldsymbol{J}} \sum_{\varphi \in J} rev(J_i, \varphi), \quad \text{where } rev(J_i, \varphi) = \min_{\substack{J' \in \mathcal{J}(\Phi, \Gamma_{out}) \\ s.t. \; \varphi \notin J'}} H(J_i, J')$$

For the profile $\boldsymbol{J}^\dagger$ in Example 2.1, the reversal-scoring rule returns the set $\text{REVSCO}(\boldsymbol{J}^\dagger) = \{\{x_1, \neg x_2, \neg x_3, \neg x_4\}\}$. The total reversal score $\sum_{J_i \in \boldsymbol{J}^\dagger} rev(J_i, x_1)$ for x_1 over all J_i in the profile $\boldsymbol{J}^\dagger$ is 8, and the total reversal score $\sum_{J_i \in \boldsymbol{J}^\dagger} rev(J_i, \neg x_1)$ for $\neg x_1$ is 6. In addition, $\sum_{J_i \in \boldsymbol{J}^\dagger} rev(J_i, x_2) = 7$,

$\sum_{J_i \in \boldsymbol{J}^\dagger} rev(J_i, \neg x_2) = 8$, $\sum_{J_i \in \boldsymbol{J}^\dagger} rev(J_i, x_3) = 7$, $\sum_{J_i \in \boldsymbol{J}^\dagger} rev(J_i, \neg x_3) = 8$, $\sum_{J_i \in \boldsymbol{J}^\dagger} rev(J_i, x_4) = 6$, and $\sum_{J_i \in \boldsymbol{J}^\dagger} rev(J_i, \neg x_4) = 9$. Since the judgment set $\{x_1, \neg x_2, \neg x_3, \neg x_4\}$ satisfies Γ_{out}, and each of the formulas φ in it have a larger total reversal score than their negation, it is the unique judgment set with highest total reversal score.

For the next definition, we return to the idea that the collective judgment set should agree with the majoritarian judgment set as much as possible. Suppose we order the agenda items in terms of the strengths of the majorities that support them (breaking ties in some way). Then we might want to accept formulas in that order, rejecting a formula with majority support only when accepting it would render the outcome inconsistent. Let $\mathcal{L}(\Phi)$ denote the set of all strict linear orders on Φ; we are going to use such orders to break ties between agenda items with equal support. Furthermore, for a given set J, profile $\boldsymbol{J}$, tie-breaking order $\gg$, and formula φ, define the set of formulas in J that in $\boldsymbol{J}$ either have more support than φ or the same amount of support as φ but precede φ in $\gg$:

$$J|^{\boldsymbol{J}}_{\gg \varphi} \;=\; \{\psi \in J \mid N(\boldsymbol{J}, \psi) > N(\boldsymbol{J}, \varphi)\} \cup \{\psi \in J \mid N(\boldsymbol{J}, \psi) = N(\boldsymbol{J}, \varphi) \text{ and } \psi \gg \varphi\}$$

Definition 7 (Ranked-agenda rule). *The ranked-agenda rule* RAN *returns, for any given profile $\boldsymbol{J}$, exactly those complete and consistent judgment sets that can be obtained by accepting formulas in an order that respects the relative strengths of the support they enjoy in $\boldsymbol{J}$:*

$$\text{RAN}(\boldsymbol{J}) \;=\; \bigcup_{\gg \in \mathcal{L}(\Phi)} \text{RAN}_\gg(\boldsymbol{J}) \quad \text{where}$$

$$\text{RAN}_\gg(\boldsymbol{J}) \;=\; \{ J \in \mathcal{J}(\Phi, \Gamma_{\text{out}}) \mid J|^{\boldsymbol{J}}_{\gg \varphi} \cup \{\varphi\} \text{ is } \Gamma_{\text{out}}\text{-inconsistent for every } \varphi \in \Phi \setminus J \}$$

For the profile $\boldsymbol{J}^\dagger$ in Example 2.1, the ranked-agenda rule returns $\text{RAN}(\boldsymbol{J}^\dagger) = \{\{x_1, x_2, \neg x_3, x_4\}, \{x_1, \neg x_2, x_3, \neg x_4\}\}$. In Example 2.1, the elements of the agenda ordered by support, are x_1 (8), x_2 and x_3 (7), x_4 (6), then $\neg x_4$ (5), $\neg x_2$ and $\neg x_3$ (4), and finally $\neg x_1$ (3). If, on the one hand, the priority relation $\gg$ is such that $x_2 \gg x_3$, then the computation of $\text{RAN}(\boldsymbol{J}^\dagger)$ starts with taking x_1, then x_2, then x_4 (because x_3 cannot be added without violating the consistency constraint), and finally $\neg x_3$. If, on the other hand, the priority relation $\gg$ is such that $x_3 \gg x_2$, then the computation of $\text{RAN}(\boldsymbol{J}^\dagger)$ starts with taking x_1, then x_3, and then $\neg x_2$ and $\neg x_4$.

Observe that $\text{RAN}_\gg$ is itself a (resolute) judgment aggregation rule, parameterised by a fixed tie-breaking order $\gg$. The name of the ranked-agenda rule, first suggested by Lang et al. (2011), derives from the ranked-pairs rule for preference aggregation (Tideman, 1987). It has also been called *Tideman rule* (Endriss & De Haan, 2015) and *support-based rule* (Porello & Endriss, 2014).

The next rule, proposed both by Nehring and Pivato (2013) and by Everaere, Konieczny, and Marquis (2014), is a refinement of the ranked-agenda rule. It is based on the following lexicographic order on judgment sets (determined by a given profile $\boldsymbol{J}$):

$$J' >^{\boldsymbol{J}}_{\text{lex}} J \quad \textit{iff} \quad \text{there exists a } k \in \mathbb{N} \text{ such that these two conditions hold:}$$

- $|\{ \varphi \in J' \mid N(\boldsymbol{J}, \varphi) \geq j \}| = |\{ \varphi \in J \mid N(\boldsymbol{J}, \varphi) \geq j \}|$ *for all $j > k$*
- $|\{ \varphi \in J' \mid N(\boldsymbol{J}, \varphi) \geq k \}| > |\{ \varphi \in J \mid N(\boldsymbol{J}, \varphi) \geq k \}|$

Definition 8 (LexiMax rule). *The* LexiMax*-rule returns, for any given profile $\boldsymbol{J}$, exactly those complete and consistent judgment sets that maximally agree with the majority decisions made in $\boldsymbol{J}$ regarding the agenda items in a lexicographic sense:*

$$\text{LexiMax}(\boldsymbol{J}) \;=\; \{ J \in \mathcal{J}(\Phi, \Gamma_{\text{out}}) \mid J' >^{\boldsymbol{J}}_{\text{lex}} J \text{ for no } J' \in \mathcal{J}(\Phi, \Gamma_{\text{out}}) \}$$

For the profile $J^\dagger$ in Example 2.1, the LEXIMAX-rule returns $\text{LEXIMAX}(J^\dagger) = \{\{x_1, x_2, \neg x_3, x_4\}\}$.

Recall that the majoritarian judgment set may not always be consistent. We could try to correct this by making a small modification to the profile. The rules defined next are based on this idea: move to a different profile close by for which the majoritarian judgment set is consistent and then return that set as the outcome of the rule. For the first of these rules the "small modification" to the profile involves removing some of the agents, while for the second rule it involves swapping some of the individual judgments on agenda items. Note that the definition of any such rule will be parametric not only in the agenda Φ and the output constraint Γ_{out} but also in the input constraint Γ_{in}.

Definition 9 (Rules based on minimal profile modifications). *The Young rule* YNG *and the Dodgson rule* DOD *are defined as follows, for any given profile J:*

$$\text{YNG}(J) = \{ J \in ext(m(J^*)) \mid J^* \in \underset{\substack{J' \in \mathcal{J}(\Phi, \Gamma_{\text{in}})^+ \\ s.t.\ J' \subseteq J\ and \\ m(J')\ is\ \Gamma_{\text{out}}\text{-consistent}}}{\text{argmax}} |J'| \}$$

$$\text{DOD}(J) = \{ J \in ext(m(J^*)) \mid J^* \in \underset{\substack{J' \in \mathcal{J}(\Phi, \Gamma_{\text{in}})^+ \\ s.t.\ |J'|=|J|\ and \\ m(J')\ is\ \Gamma_{\text{out}}\text{-consistent}}}{\text{argmin}} \sum_{i \leq |J|} H(J_i, J'_i) \}$$

For the profile $J^\dagger$ in Example 2.1, the Young rule returns $\text{YNG}(J^\dagger) = \{\{x_1, x_2, \neg x_3, x_4\},$ $\{x_1, x_2, \neg x_3, \neg x_4\}, \{x_1, \neg x_2, x_3, \neg x_4\}, \{x_1, \neg x_2, \neg x_3, \neg x_4\}\}$, and the Dodgson rule returns $\text{DOD}(J^\dagger) = \{\{x_1, x_2, \neg x_3, x_4\}\}$.

The Young rule was first introduced by Lang et al. (2011), named so due its similarity to the rule of the same name familiar from voting theory (Young, 1977; Caragiannis, Hemaspaandra, & Hemaspaandra, 2016). The Dodgson rule is the most important instance of the *"full" distance-based rules* defined by Miller and Osherson (2009). This rule is named after the Dodgson voting rule (Dodgson, 1876; Caragiannis et al., 2016)—when using an input constraint that expresses linear orders over a set of candidates and an output constraint that expresses that one candidate is preferred over all other candidates, we can simulate the Dodgson voting rule using the rule DOD.

A further family of aggregation rules of some interest are the *representative-voter rules*, which are rules that always return judgment sets that are part of the *support* of the input profile, i.e., that are themselves individual judgment sets submitted by at least one agent (Endriss & Grandi, 2014). The idea is that an aggregation rule should determine which of the agents is most representative of the group and then return that agent's judgment set. The most natural implementations of this idea are the *average-voter rule* (returning the individual judgment sets that minimise the average Hamming distance to the profile), the *majority-voter rule* (returning the individual judgment sets that minimise the Hamming distance to the majoritarian judgment set), and the *plurality-voter rule* (returning the individual judgment sets occurring most frequently in the profile). Note how the first two of these rules closely resemble the Kemeny rule and the Slater rule, respectively (with the support of J now taking the role of $\mathcal{J}(\Phi, \Gamma_{\text{out}})$). Any representative-voter rule is consistent by construction as long as the output constraint is a logical consequence of (or simply equal to) the input constraint. A great advantage of the representative-voter rules is that they are computationally easy: we simply have to choose the most representative judgment set found in the input rather than scanning through the potentially exponentially large set of *all* consistent judgment sets. While this makes them attractive for certain applications, this also means that they are of no further interest for this particular paper, given our focus on questions of computational complexity.

3. Complexity-Theoretic Preliminaries

In this section, we survey some basic notions from the theory of computational complexity. We assume that the reader is familiar with the complexity classes P and NP, and with basic notions such as polynomial-time reductions.[11] For more details, we refer to textbooks on computational complexity theory (see, e.g., Arora & Barak, 2009). We begin by recalling the archetypal NP-complete problem SAT.

SAT
Instance: A propositional formula φ.
Question: Is there a truth assignment that satisfies φ?

We briefly revisit some of the classes of the Polynomial Hierarchy (PH) (Meyer & Stockmeyer, 1972; Stockmeyer, 1976; Wrathall, 1976). These classes can be conveniently characterised using quantified Boolean formulas. A *quantified Boolean formula (in prenex form)* is a formula of the form $Q_1 x_1 Q_2 x_2 \ldots Q_n x_n.\psi$, where all x_i are propositional variables, each Q_i is either an existential quantifier ($\exists$) or a universal quantifier ($\forall$), and ψ is a (quantifier-free) propositional formula over the variables $x_1, \ldots, x_n$. The notion of truth for such formulas is defined in the usual way (see, e.g., Kleine Büning & Bubeck, 2009).

The PH contains the complexity class Σ_2^p, that consists of all decision problems for which there exists a polynomial-time reduction to the problem QSAT$_2$, which is defined as follows. Instances of the problem QSAT$_2$ are quantified Boolean formulas of the form $\exists x_1 \ldots \exists x_n \forall y_1 \ldots \forall y_m.\psi$, where ψ is quantifier-free. The problem is to decide whether the quantified Boolean formula φ is true. The PH also contains the complexity class Π_2^p. The class Π_2^p consists of all decision problems for which there exists a polynomial-time reduction to the problem coQSAT$_2$, which is complementary to the problem QSAT$_2$.

One can also characterise the class Σ_2^p using nondeterministic polynomial-time algorithms with access to an oracle for a problem in NP—e.g., an oracle for an NP-complete problem. Let R be a decision problem. A Turing machine $\mathbb{M}$ with access to an *R-oracle* is a Turing machine with a dedicated *oracle tape* and dedicated states q_{query}, q_{yes} and q_{no}. Whenever $\mathbb{M}$ is in the state q_{query}, it does not proceed according to the transition relation, but instead it transitions into the state q_{yes} if the oracle tape contains a string x that is a yes-instance for the problem R, i.e., if $x \in R$, and it transitions into the state q_{no} if $x \notin R$. Intuitively, the oracle solves arbitrary instances of R in a single time step. The class Σ_2^p consists of all decision problems that can be solved in polynomial time by a nondeterministic Turing machine that has access to an R-oracle, for some $R \in$ NP.

Two other classes that are contained in the PH are Δ_2^p and Θ_2^p. The complexity class Δ_2^p consists of all decision problems that can be solved by a polynomial-time deterministic Turing machine that has access to an R-oracle, for some $R \in$ NP. The complexity class Θ_2^p consists of all decision problems that can be solved by a polynomial-time deterministic Turing machine that has access to an R-oracle, for some $R \in$ NP, and that on each input queries this oracle $O(\log n)$ times, where n denotes the input size (Papadimitriou & Zachos, 1982). The class Θ_2^p can alternatively be characterised as the set of all decision problems that can be solved in polynomial time by an algorithm that has access to an R-oracle, for some $R \in$ NP, and that makes all queries to the oracle in parallel—i.e., first all queries

---

11. From this point on, we will only refer to polynomial-time many-to-one reductions, and we will refer to them simply as polynomial-time reductions.

are written, and then the answers to all these queries are given at the same time (Köbler, Schöning, & Wagner, 1987; Hemachandra, 1989; Buss & Hay, 1991).

The following problem is complete for the class Δ_2^p under polynomial-time reductions (Krentel, 1992). Let φ be a (satisfiable) propositional formula with $\mathsf{var}(\varphi) = \{x_1, \dots, x_n\}$, and take the natural order $x_1 < \cdots < x_n$ over these variables. The *lexicographically maximal model α* of φ is the lexicographically maximal truth assignment that satisfies φ—that is, α is the (unique) truth assignment $\alpha : \mathsf{var}(\varphi) \to \{0, 1\}$ for which it holds that (1) α satisfies φ, and (2) no truth assignment $\beta : \mathsf{var}(\varphi) \to \{0, 1\}$ that precedes α in the lexicographical order satisfies φ. A truth assignment β precedes α in the lexicographical order (w.r.t. $x_1 < \cdots < x_n$) if there exists some $1 \le k < n$ such that (i) $\beta(x_i) = \alpha(x_i)$ for all $1 \le i < k$ and (ii) $\beta(x_k) = 0$ and $\alpha(x_k) = 1$.

LEX-MAX-MODEL

Instance: A (satisfiable) propositional formula φ and a variable $x^* \in \mathsf{var}(\varphi)$.
Question: Is it the case that the lexicographically maximal model of φ sets x^* to true?

The following problem is complete for the class Θ_2^p under polynomial-time reductions (Krentel, 1988; Wagner, 1990; Chen & Toda, 1995).

MAX-MODEL

Instance: A satisfiable propositional formula φ, and a variable $x^* \in \mathsf{var}(\varphi)$.
Question: Is there a model α of φ that sets x^* to true, such that there is no other model of φ that sets more variables in $\mathsf{var}(\varphi)$ to true than α?

Additionally, the PH contains the complexity class Θ_3^p, which consists of all decision problems that can be solved by a polynomial-time Turing machine with access to an R-oracle, for some $R \in \Sigma_2^p$, that on each input queries this oracle oracle $O(\log n)$ times, where n denotes the input size. The following problem is complete for Θ_3^p under polynomial-time reductions (Chen & Toda, 1995).

QSAT$_2$-MIN-MODEL

Instance: A true instance φ of QSAT$_2$, where $\varphi = \exists X.\forall Y.\psi$, and a variable $x^* \in X$.
Question: Is there a truth assignment $\alpha : X \to \{0, 1\}$ that sets x^* to true and for which $\psi[\alpha]$ is valid, such that there is no truth assignment $\beta : X \to \{0, 1\}$ that sets more variables in X to true and for which $\psi[\beta]$ is valid?

An overview of the relation between the complexity classes that we discussed (and the other well-known complexity classes coNP and PSPACE) can be found in Figure 4.

4. Results

In this section, we present computational complexity results for the outcome determination problem in judgment aggregation, for the different judgment aggregation rules and for the different judgment aggregation frameworks we discussed in Section 2. These results are summarised in Table 1 on page 690. We give full proofs for all the results that we present. Moreover, for each result we give a reference to where restricted versions of these results—i.e., for only some variants of the judgment aggregation framework and for only some of the problem variants—have appeared in the literature (Lang & Slavkovik, 2014; Endriss & De Haan, 2015; De Haan & Slavkovik, 2017; De Haan, 2018).
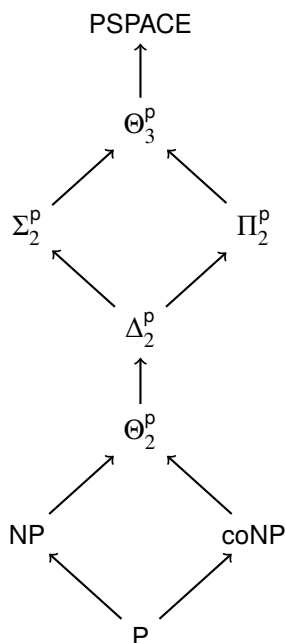
Figure 4: An overview of the computational complexity classes that are relevant for the results in this paper—and their relation to some other prominent complexity classes. Arrows indicate inclusion relations.

4.1 Outcome Determination for the Condorcet Rule

We begin by proving membership in Σ_2^p for the outcome determination problem for the Condorcet rule Con.

Theorem 4.1. *The outcome determination problem for the Condorcet rule* Con *is in* Σ_2^p.

Proof. As described in Sections 2.1 and 2.2, we will use variant (6) of the framework and variant (a) of the problem to establish the upper bound on the complexity. This proof proceeds along the same lines as a proof given by Lang and Slavkovik (2014, Proposition 1). We describe a nondeterministic polynomial-time algorithm with access to an NP oracle that solves the problem. The algorithm guesses a complete judgment set $J^* \subseteq \Phi$, and guesses a truth assignment α to the variables in Φ and in Γ_{out}. The algorithm then checks that α satisfies Γ_{out} and satisfies φ for each $\varphi \in J^*$, and that $L \subseteq J^*$ and $L_i \not\subseteq J^*$ for each $i \in \{1, \ldots, u\}$. Then, the algorithm uses the NP oracle to check whether there exists a complete and Γ_{out}-consistent judgment set $J' \subseteq \Phi$ such that $J' \cap m(\boldsymbol{J}) \supsetneq J^* \cap m(\boldsymbol{J})$. If there exists such a judgment set J', the algorithm rejects the input; otherwise, it accepts the input. By definition of the Condorcet rule, the algorithm accepts the input if and only if $J^* \in \mathrm{Con}(\boldsymbol{J})$. □

Next, we prove Σ_2^p-hardness for the outcome determination problem for the Condorcet rule.[12]

---

12. The proof of Theorem 4.2 is not based on the original hardness proof for the outcome determination problem for the Condorcet rule, that appeared in preliminary work (Lang & Slavkovik, 2014, Proposition 1)—the original proof only applies to variants (2), (3) and (4) of the framework.

Theorem 4.2. *The outcome determination problem for the Condorcet rule* CON *is* Σ_2^p-*hard.*

Proof. As described in Sections 2.1 and 2.2, we will use variant (1) of the framework and variant (b) of the problem to establish the lower bound on the complexity. We give a reduction from the Σ_2^p-complete problem QSAT$_2$. Let $\varphi = \exists x_1 \ldots \exists x_n \forall y_1 \ldots \forall y_m.\psi$ be an instance of QSAT$_2$. We construct an instance $(\Phi, \Gamma_{\text{in}}, \Gamma_{\text{out}}, \boldsymbol{J}, \varphi^*)$ of the outcome determination problem for the Condorcet rule as follows. We define Φ by letting $[\Phi] = \{ z_{x_i}, z_{\overline{x_i}} \mid 1 \leq i \leq n \} \cup \{ z_{y_j}, z_{\overline{y_j}} \mid 1 \leq j \leq m \} \cup \{w_1, w_2, w_3, u\}$. We define $\Gamma_{\text{in}} = \Gamma_{\text{out}}$ as follows:

$$\Gamma_{\text{in}} = \Gamma_{\text{out}} = \; (w_1 \wedge w_2 \wedge w_3) \to$$
$$\left(\bigwedge_{i=1}^{n} \left(z_{x_i} \leftrightarrow \neg z_{\overline{x_i}} \right) \wedge \left(\left(u \wedge \bigwedge_{j=1}^{m} \left(\neg z_{y_j} \wedge \neg z_{\overline{y_j}} \right) \right) \vee \left(\neg \psi' \wedge \bigwedge_{j=1}^{m} \left(z_{y_j} \leftrightarrow \neg z_{\overline{y_j}} \right) \right) \right) \right),$$

where ψ' is obtained from ψ by replacing each variable x_i by z_{x_i} and replacing each variable y_j by z_{y_j}. We define the profile $\boldsymbol{J} = (J_1, J_2, J_3)$ as indicated in Figure 5. Finally, we let $\varphi^* = u$.

$\boldsymbol{J}$	J_1	J_2	J_3	$m(\boldsymbol{J})$
z_{x_1}	1	1	1	1
$z_{\overline{x_1}}$	1	1	1	1
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
z_{x_n}	1	1	1	1
$z_{\overline{x_n}}$	1	1	1	1
z_{y_1}	1	1	1	1
$z_{\overline{y_1}}$	1	1	1	1
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
z_{y_m}	1	1	1	1
$z_{\overline{y_m}}$	1	1	1	1
w_1	1	1	0	1
w_2	1	0	1	1
w_3	0	1	1	1
u	0	0	0	0

Figure 5: Construction of the profile $\boldsymbol{J}$ in the proof of Theorem 4.2.

We show that φ is true if and only if there is some $J^* \in \text{CON}(\boldsymbol{J})$ with $\varphi^* \in J^*$.

($\Rightarrow$) Suppose that φ is true—that is, that there is a truth assignment $\alpha : \{x_1, \ldots, x_n\} \to \{0, 1\}$ such that for all truth assignments $\beta : \{y_1, \ldots, y_m\} \to \{0, 1\}$ it holds that $\alpha \cup \beta$ satisfies ψ. Consider the judgment set $J^* = \{w_1, w_2, w_3, u\} \cup \{ z_{x_i} \mid 1 \leq i \leq n, \alpha(x_i) = 1 \} \cup \{ z_{\overline{x_i}} \mid 1 \leq i \leq n, \alpha(x_i) = 0 \} \cup \{ \neg z_{y_j}, \neg z_{\overline{y_j}} \mid 1 \leq j \leq m \}$. This set J^* is Γ_{out}-consistent. We show that there is no Γ_{out}-consistent J' such that $m(\boldsymbol{J}) \cap J' \supsetneq m(\boldsymbol{J}) \cap J^*$. To do so, we first observe that $m(\boldsymbol{J}) \cap J^* = \{w_1, w_2, w_3\} \cup \{ z_{x_i} \mid 1 \leq i \leq n, \alpha(x_i) = 1 \} \cup \{ z_{\overline{x_i}} \mid 1 \leq i \leq n, \alpha(x_i) = 0 \}$. Suppose, to derive a contradiction, that such a J' exists—that is, $m(\boldsymbol{J}) \cap J' \supsetneq m(\boldsymbol{J}) \cap J^*$. Then either J' contains $\neg u$, or J' contains z_{y_j} or $z_{\overline{y_j}}$ for some $1 \leq j \leq m$. In both cases, because J' is Γ_{out}-consistent, we know that J' is consistent with $\neg \psi' \wedge \bigwedge_{j=1}^{m}(z_{y_j} \leftrightarrow \neg z_{\overline{y_j}})$. Moreover, the set J' must agree with J^* on the issues $z_{x_i}, z_{\overline{x_i}}$. Consider the truth assignment $\beta' : \{y_1, \ldots, y_m\} \to \{0, 1\}$

that is defined by letting $\beta'(y_j) = 1$ if and only if $z_{y_j} \in J'$, for each $1 \leq j \leq m$. Since J' is consistent with $\neg\psi'$, we get that $\alpha \cup \beta'$ satisfies $\neg\psi$. This is a contradiction with our assumption that $\psi[\alpha]$ is a tautology. Therefore, we can conclude that no such J' exists, and thus that $J^* \in \mathrm{Con}(J)$.

($\Leftarrow$) Suppose that there is some $J^* \in \mathrm{Con}(J)$ with $u = \varphi^* \in J^*$. We then know that $\{w_1, w_2, w_3\} \subseteq J^*$—if this were not the case, then $J^* \setminus \{u\} \cup \{\neg u\}$ would be Γ_{out}-consistent and would agree with $m(J)$ on a strict superset of issues (w.r.t. J^*), which would contradict $J^* \in \mathrm{Con}(J)$. Then, since J^* is Γ_{out}-consistent, we know that for each $1 \leq i \leq n$ it holds that $|J^* \cap \{z_{x_i}, z_{\overline{x_i}}\}| = 1$. Moreover, we know that $\{\neg z_{y_j}, \neg z_{\overline{y_j}} \mid 1 \leq j \leq m\} \subseteq J^*$—again, if this were not the case, then $J^* \setminus \{u\} \cup \{\neg u\}$ would be Γ_{out}-consistent and would agree with $m(J)$ on a strict superset of issues (w.r.t. J^*), which would contradict $J^* \in \mathrm{Con}(J)$. Consider the truth assignment $\alpha : \{x_1, \ldots, x_n\} \to \{0, 1\}$ that is defined by letting $\alpha(x_i) = 1$ if and only if $z_{x_i} \in J^*$, for each $1 \leq i \leq n$. We show that $\psi[\alpha]$ is a tautology. Suppose, to derive a contradiction, that $\psi[\alpha]$ is not a tautology, that is, that there exists some $\beta : \{y_1, \ldots, y_m\} \to \{0, 1\}$ such that $\alpha \cup \beta$ falsifies ψ. Consider the judgment set $J' = \{w_1, w_2, w_3, u\} \cup (J^* \cap \{z_{x_i}, \neg z_{x_i}, z_{\overline{x_i}}, \neg z_{\overline{x_i}} \mid 1 \leq i \leq n\}) \cup \{z_{y_j} \mid 1 \leq j \leq m, \beta(y_j) = 1\} \cup \{z_{\overline{y_j}} \mid 1 \leq j \leq m, \beta(y_j) = 0\}$. It holds that $m(J) \cap J' \supsetneq m(J) \cap J^*$. Moreover, because $\alpha \cup \beta$ satisfies $\neg\psi$, we get that J' is Γ_{out}-consistent. This is a contradiction with our assumption that $J^* \in \mathrm{Con}(J)$. Thus, we can conclude that no such β exists, and thus that $\psi[\alpha]$ is a tautology and that φ is true. $\square$

4.2 Outcome Determination for the Slater Rule

We now turn to the membership result for the outcome determination problem for the Slater rule.

Theorem 4.3. *The outcome determination problem for the Slater rule* SLA *is in* Θ_2^{p}.

Proof. As described in Sections 2.1 and 2.2, we will use variant (6) of the framework and variant (a) of the problem to establish the upper bound on the complexity. We describe a polynomial-time algorithm that queries an NP oracle $O(\log |\Phi|)$ times and that solves the problem. The algorithm uses oracle queries to check if there exists a complete and Γ_{out}-consistent judgment set J that agrees with $m(J)$ on a given number k of formulas $\varphi \in \Phi$. This is a problem in NP, so by picking an NP-complete problem for the oracle, we can use a single oracle query to solve any instance of the above problem in polynomial time. The maximal number k of formulas φ on which any judgment set $J \subseteq \Phi$ agrees with $m(J)$ is $|\Phi|$. Therefore, we can determine the maximal number $k_{\max}$ of formulas φ on which any complete and Γ_{out}-consistent judgment set $J \subseteq \Phi$ agrees with $m(J)$ using $O(\log |\Phi|)$ oracle queries—by using binary search. Then, after having identified this number $k_{\max}$, the algorithm queries the oracle one additional time to determine whether there exists a complete and Γ_{out}-consistent judgment set J^* that agrees with $m(J)$ on $k_{\max}$ formulas $\varphi \in \Phi$ such that $L \subseteq J^*$ and $L_i \nsubseteq J^*$ for all $i \in \{1, \ldots, u\}$. Again, since this is a problem in NP, one oracle query suffices to determine this.[13] The algorithm outputs "yes" if and only if such a judgment set J^* exists. This algorithm runs in polynomial time, queries the NP oracle $O(\log |\Phi|)$ times, and correctly solves the problem. Therefore, the problem is in Θ_2^{p}. $\square$

To show that outcome determination for the Slater rule is Θ_2^{p}-hard, one can use the standard embedding of voting into judgment aggregation (Dietrich & List, 2007a; Endriss, 2016). Performing

---

13. We use a single oracle to solve instances of two different NP problems. By picking an NP complete problem R for the oracle, we know that we can solve any problem in NP with a single oracle call—by using the fact that each problem in NP can be polynomial-time reduced to R.

the Slater rule on the result of this embedding corresponds to carrying out the Slater voting rule (Endriss, 2018). As computing the winner of a Slater election is Θ_2^p-hard (see, e.g., Conitzer, 2006), this embedding can be used to provide a reduction to show Θ_2^p-hardness of outcome determination for the Slater judgment aggregation rule. We give an alternative proof of Θ_2^p-hardness that is self-contained and does not rely on work in voting theory.[14]

Theorem 4.4. *The outcome determination problem for the Slater rule* SLA *is* Θ_2^p*-hard.*

Proof. As described in Sections 2.1 and 2.2, we will use variant (1) of the framework and variant (b) of the problem to establish the lower bound on the complexity. We give a reduction from the Θ_2^p-complete problem MAX-MODEL. Let (ψ, x^*) be an instance of MAX-MODEL, with $\text{var}(\psi) = \{x_1, \ldots, x_n\}$. Without loss of generality, suppose that $x^* = x_1$. We construct an instance $(\Phi, \Gamma_{\text{in}}, \Gamma_{\text{out}}, J, \varphi^*)$ of the outcome determination problem for the Slater rule as follows.

We firstly construct the agenda Φ. For each $1 \leq i \leq n$ and each $1 \leq j \leq n+1$, we introduce a fresh propositional variable $z_{i,j}$. We then let $\Phi = \{ x_i, \neg x_i, z_{i,j}, \neg z_{i,j} \mid 1 \leq i \leq n, 1 \leq j \leq n+1 \}$. We define the constraints $\Gamma_{\text{in}} = \Gamma_{\text{out}}$ as follows:

$$\Gamma_{\text{in}} = \Gamma_{\text{out}} = \bigvee_{1 \leq i \leq n} \left(\bigwedge_{1 \leq j \leq n+1} z_{i,j} \right) \vee \left(\psi \wedge \bigwedge_{\substack{1 \leq i \leq n \\ 1 \leq j \leq n+1}} \neg z_{i,j} \right).$$

We construct the profile $J = (J_1, \ldots, J_n)$ as indicated in Figure 6. Finally, we let $\varphi^* = x_1$.

J	J_1	J_2	$\cdots$	J_{n-1}	J_n	$m(J)$
x_1	0	1	1	$\cdots$	1	1
x_2	1	0	1	$\cdots$	1	1
$\vdots$	$\vdots$		$\ddots$		$\vdots$	$\vdots$
x_{n-1}	1	$\cdots$	1	0	1	1
x_n	1	$\cdots$	1	1	0	1
$z_{1,1}$	1	0	0	$\cdots$	0	0
$z_{2,1}$	0	1	0	$\cdots$	0	0
$\vdots$	$\vdots$		$\ddots$		$\vdots$	$\vdots$
$z_{n-1,1}$	0	$\cdots$	0	1	0	0
$z_{n,1}$	0	$\cdots$	0	0	1	0
$\vdots$			$\vdots$			$\vdots$
$z_{1,n+1}$	1	0	0	$\cdots$	0	0
$z_{2,n+1}$	0	1	0	$\cdots$	0	0
$\vdots$	$\vdots$		$\ddots$		$\vdots$	$\vdots$
$z_{n-1,n+1}$	0	$\cdots$	0	1	0	0
$z_{n,n+1}$	0	$\cdots$	0	0	1	0

Figure 6: Construction of the profile J in the proof of Theorem 4.4.

---

14. The proof of Theorem 4.4 is based on a proof that appeared in preliminary work (Endriss & De Haan, 2015).

The intuition behind this construction is that for an outcome to agree with the majoritarian judgment set $m(\boldsymbol{J})$ on as many issues in Φ as possible, it is "cheaper" to do so by satisfying ψ than $\bigwedge_{1 \leq j \leq n+1} z_{i,j}$ for some $1 \leq i \leq n$. The former can be done by deviating from $m(\boldsymbol{J})$ on at most n issues, whereas the latter requires deviating from $m(\boldsymbol{J})$ on at least $n + 1 > n$ issues.

We show that there is some $J^* \in \text{SLA}(\boldsymbol{J})$ with $\varphi^* \in J^*$ if and only if there is a model of ψ that sets a maximal number of variables in $\text{var}(\psi)$ to true and that sets x_1 to true.

($\Rightarrow$) Suppose that there is some $J^* \in \text{SLA}(\boldsymbol{J})$ with $\varphi^* \in J^*$. Define the truth assignment $\alpha : \text{var}(\psi) \rightarrow \{0, 1\}$ by letting $\alpha(x_i) = 1$ if and only if $x_i \in J^*$. We know that J^* is Γ_{out}-consistent—thus either (i) it is consistent with $\bigvee_{1 \leq i \leq n} \bigwedge_{1 \leq j \leq n+1} z_{i,j}$ or (ii) it is consistent with $\psi \wedge \bigwedge_{1 \leq i \leq n, 1 \leq j \leq n+1} \neg z_{i,j}$. In case (i), it must disagree with $m(\boldsymbol{J})$ on at least $n + 1$ issues, and in case (ii), it can disagree with $m(\boldsymbol{J})$ on at most n issues. Thus, since J^* maximises agreement with $m(\boldsymbol{J})$, we can rule out case (i). Therefore, we know that α satisfies ψ. Also, we know that α sets x_1 to true, because $x_1 = \varphi^* \in J^*$. We show that there is no $\beta : \text{var}(\psi) \rightarrow \{0, 1\}$ that satisfies ψ and that sets more variables in $\text{var}(\psi)$ to true than α. Suppose, to derive a contradiction, that such a β exists. Consider the judgment set $J' = \{ \neg z_{i,j} \mid 1 \leq i \leq n, 1 \leq j \leq n + 1 \} \cup \{ \ell \mid 1 \leq i \leq n, \ell \in \{x_i, \neg x_i\}, \beta(\ell) = 1 \}$. This set J' is Γ_{out}-consistent, and agrees with $m(\boldsymbol{J})$ on strictly more issues than J^*. This is a contradiction with our assumption that $J^* \in \text{SLA}(\boldsymbol{J})$. Thus, we can conclude that no such β exists. We then know that α sets a maximal number of variables in $\text{var}(\psi)$ to true and sets x_1 to true.

($\Leftarrow$) Suppose that there is a truth assignment $\alpha : \text{var}(\psi) \rightarrow \{0, 1\}$ that satisfies ψ, sets a maximal number of variables in $\text{var}(\psi)$ to true and sets x_1 to true. Consider the judgment set $J^* = \{ \neg z_{i,j} \mid 1 \leq i \leq n, 1 \leq j \leq n + 1 \} \cup \{ \ell \mid 1 \leq i \leq n, \ell \in \{x_i, \neg x_i\}, \alpha(\ell) = 1 \}$. This set J^* is Γ_{out}-consistent. Moreover, it contains $\varphi^* = x_1$, because $\alpha(x_1) = 1$. We show that $J^* \in \text{SLA}(\boldsymbol{J})$. Suppose, to derive a contradiction, that there is some complete and Γ_{out}-consistent judgment set J' that agrees with $m(\boldsymbol{J})$ on more issues than J^*. Without loss of generality, we can take J' so that it minimizes disagreement with $m(\boldsymbol{J})$. Since J' is Γ_{out}-consistent, it holds that either (i) it is consistent with $\bigvee_{1 \leq i \leq n} \bigwedge_{1 \leq j \leq n+1} z_{i,j}$ or (ii) it is consistent with $\psi \wedge \bigwedge_{1 \leq i \leq n, 1 \leq j \leq n+1} \neg z_{i,j}$. In case (i), J' must disagree with $m(\boldsymbol{J})$ on at least $n+1$ issues, and in case (ii), it can disagree with $m(\boldsymbol{J})$ on at most n issues. Then, because J' minimizes disagreement with $m(\boldsymbol{J})$, we can rule out case (i). Therefore, we know that $\{ \neg z_{i,j} \mid 1 \leq i \leq n, 1 \leq j \leq n + 1 \} \subseteq J'$, and that $J' \cup \{\psi\}$ is satisfiable. The only way that J' can agree with $m(\boldsymbol{J})$ on more issues than J^* is for $|J' \cap \{ x_i \mid 1 \leq i \leq n \}| > |J^* \cap \{ x_i \mid 1 \leq i \leq n \}|$. Consider the truth assignment $\beta : \text{var}(\psi) \rightarrow \{0, 1\}$ such that for each $1 \leq i \leq n$, $\beta(x_i) = 1$ if and only if $x_i \in J'$. Then β sets more variables in $\text{var}(\psi)$ to true than α. Moreover, since $J' \cup \{\psi\}$ is satisfiable, β satisfies ψ. This is a contradiction with our assumption that α sets a maximal number of variables in $\text{var}(\psi)$ to true for any satisfying assignment of ψ. Thus, we can conclude that no such J' exists, and thus that $J^* \in \text{SLA}(\boldsymbol{J})$. $\qquad \square$

4.3 Outcome Determination for the Kemeny Rule

Next is the membership result for the outcome determination problem for the Kemeny rule KEM.

Theorem 4.5. *The outcome determination problem for the Kemeny rule* KEM *is in* Θ_2^p.

Proof. As described in Sections 2.1 and 2.2, we will use variant (6) of the framework and variant (a) of the problem to establish the upper bound on the complexity. The proof is similar to the proof of Theorem 4.3. We describe a polynomial-time algorithm that queries an NP oracle $O(\log |\Phi| + \log n)$ times and that solves the problem. The algorithm uses oracle queries to check if there exists a

complete and Γ_{out}-consistent judgment set J such that $\sum_{J_i \in \boldsymbol{J}} H(J, J_i) \leq k$ for a given value k. This is a problem in NP, so by picking an NP-complete problem for the oracle, we can use a single oracle query to solve any instance of the above problem in polynomial time. The maximum value of $\sum_{J_i \in \boldsymbol{J}} H(J, J_i)$ for any judgment set $J \subseteq \Phi$ is at most $n \cdot |\Phi|$, where n is the number of judgment sets in the profile $\boldsymbol{J}$. Therefore, we can determine the minimum value $k_{\min}$ of $\sum_{J_i \in \boldsymbol{J}} H(J, J_i)$ for any complete and Γ_{out}-consistent judgment set J using $O(\log |\Phi| + \log n)$ oracle queries—by using binary search. Then, after having identified this number $k_{\min}$, the algorithm queries the oracle one additional time to determine whether there exists a complete and Γ_{out}-consistent judgment set J^* such that $\sum_{J_i \in \boldsymbol{J}} H(J^*, J_i) = k_{\min}$, such that $L \subseteq J^*$ and such that $L_i \nsubseteq J^*$ for all $i \in \{1, \ldots, u\}$. Again, since this is a problem in NP, one oracle query suffices to determine this. The algorithm outputs "yes" if and only if such a judgment set J^* exists. This algorithm runs in polynomial time, queries the NP oracle $O(\log |\Phi| + \log n)$ times, and correctly solves the problem. Therefore, the problem is in Θ_2^{p}. □

Similarly as for the Slater rule, for the following result one can use the standard embedding of voting into judgment aggregation (Dietrich & List, 2007a; Endriss, 2016). Performing the Kemeny rule on the result of this embedding corresponds to carrying out the Kemeny voting rule (Endriss, 2018). As computing the winner of a Kemeny election is Θ_2^{p}-hard (Hemaspaandra, Spakowski, & Vogel, 2005), this embedding can be used to provide a reduction to show Θ_2^{p}-hardness of outcome determination for the Kemeny judgment aggregation rule. Also in this case, we give an alternative proof of Θ_2^{p}-hardness that is self-contained and does not rely on work in voting theory.[15]

Theorem 4.6. *The outcome determination problem for the Kemeny rule* KEM *is* Θ_2^{p}-*hard.*

Proof. As described in Sections 2.1 and 2.2, we will use variant (1) of the framework and variant (b) of the problem to establish the lower bound on the complexity. We give a reduction from the Θ_2^{p}-complete problem MAX-MODEL. We use the same reduction as used for the Slater rule in the proof of Theorem 4.4—that is, for each instance (ψ, x^*) of MAX-MODEL we construct the same instance $(\Phi, \Gamma_{\text{in}}, \Gamma_{\text{out}}, \boldsymbol{J}, \varphi^*)$ as described in the proof of Theorem 4.4. To see that this reduction also works correctly for the Kemeny rule, it suffices to see that each issue in the majoritarian judgment set $m(\boldsymbol{J})$ is supported by the same number of judgment sets in the profile. Thus, for instances that are constructed by the reduction, the Slater rule coincides with the Kemeny rule. □

4.4 Outcome Determination for the MaxHamming Rule

We now turn our attention to the membership result for the outcome determination problem for the MaxHamming rule MAXHAM.

Theorem 4.7. *The outcome determination problem for the MaxHamming rule* MAXHAM *is in* Θ_2^{p}.

Proof. As described in Sections 2.1 and 2.2, we will use variant (6) of the framework and variant (a) of the problem to establish the upper bound on the complexity. The proof is similar to the proofs of Theorems 4.3 and 4.5. We describe a polynomial-time algorithm that queries an NP oracle $O(\log |\Phi|)$ times and that solves the problem. The algorithm uses oracle queries to check if there exists a complete and Γ_{out}-consistent judgment set J such that $\max_{J_i \in \boldsymbol{J}} H(J, J_i) \leq k$ for a given value k. This is a problem in NP, so by picking an NP-complete problem for the oracle, we can use a single

---

15. The proof of Theorem 4.6 is based on a proof that appeared in preliminary work (Endriss & De Haan, 2015).

oracle query to solve any instance of the above problem in polynomial time. The maximum value of $\max_{J_i \in \mathbf{J}} H(J, J_i)$ for any judgment set $J \subseteq \Phi$ is at most $|\Phi|$, where n is the number of judgment sets in the profile $\mathbf{J}$. Therefore, we can determine the minimum value $k_{\min}$ of $\max_{J_i \in \mathbf{J}} H(J, J_i)$ for any complete and Γ_{out}-consistent judgment set J using $O(\log |\Phi|)$ oracle queries—by using binary search. Then, after having identified this number $k_{\min}$, the algorithm queries the oracle one additional time to determine whether there exists a complete and Γ_{out}-consistent judgment set J^* such that $\max_{J_i \in \mathbf{J}} H(J^*, J_i) = k_{\min}$, such that $L \subseteq J^*$ and such that $L_i \nsubseteq J^*$ for all $i \in \{1, \ldots, u\}$. Again, since this is a problem in NP, one oracle query suffices to determine this. The algorithm outputs "yes" if and only if such a judgment set J^* exists. This algorithm runs in polynomial time, queries the NP oracle $O(\log |\Phi|)$ times, and correctly solves the problem. Therefore, the problem is in Θ_2^p. $\qquad\square$

We modify the proof of Theorem 4.4 to give us a proof of Θ_2^p-hardness for the outcome determination problem for the MaxHamming rule.[16]

Theorem 4.8. *The outcome determination problem for the MaxHamming rule* MaxHam *is Θ_2^p-hard.*

Proof. As described in Sections 2.1 and 2.2, we will use variant (1) of the framework and variant (b) of the problem to establish the lower bound on the complexity. We give a reduction from the Θ_2^p-complete problem Max-Model. We use the same reduction as used for the Slater rule in the proof of Theorem 4.4—that is, for each instance (ψ, x^*) of Max-Model we construct the same instance $(\Phi, \Gamma_{\text{in}}, \Gamma_{\text{out}}, \mathbf{J}, \varphi^*)$ as described in the proof of Theorem 4.4. We will argue that this reduction also works correctly for the MaxHamming rule MaxHam.

When considering judgment sets J that are Γ_{out}-consistent we can distinguish two cases: either (i) for some $1 \le i \le n$, J contains $z_{i,j}$ for all $1 \le j \le u$, or (ii) J is consistent with ψ. In case (i), we know that there is a judgment set J_i in the profile $\mathbf{J}$ such that the Hamming distance between J and J_i is at least $2u$. In case (ii), we know that J is consistent with $\bigwedge_{1 \le i \le n, 1 \le j \le u} \neg z_{i,j}$, and thus that the maximum Hamming distance between J and any judgment set J_i in the profile $\mathbf{J}$ is at most $u + n$. Because we chose u so that $u < n$, we know that $u + n < 2u$. From this, we get that all $J^* \in \text{MaxHam}(\mathbf{J})$ are consistent with ψ.

For any judgment set J that is consistent with ψ it holds that the maximum Hamming distance between J and any judgment set J_i in the profile $\mathbf{J}$ is $u + n - b + 1$, where b is the maximal number of variables in $\text{var}(\psi)$ that are set to true by any satisfying truth assignment of ψ. Moreover, the judgment sets $J^* \in \text{MaxHam}(\mathbf{J})$ are in one-to-one correspondence with the truth assignments satisfying ψ that set a maximal number of variables in $\text{var}(\psi)$ to true. From this, we get that there is a $J^* \in \text{MaxHam}(\mathbf{J})$ with $x_1 = \varphi^* \in J^*$ if and only if there is a model of ψ that sets a maximal number of variables in $\text{var}(\psi)$ to true and that sets x_1 to true. $\qquad\square$

4.5 Outcome Determination for Rules Based on the Geodesic Distance

Next, we show Θ_3^p-membership for the outcome determination problem for the AvgGeo rule.[17] To establish membership in Θ_3^p, we use two lemmas.

---

16. The proof of Theorem 4.8 is not based on hardness proofs for the outcome determination problem for the MaxHamming rule that appeared in preliminary work (De Haan and Slavkovik, 2017, Theorem 3; De Haan, 2018, Proposition 4). One of the hardness proofs that appeared in preliminary work (De Haan and Slavkovik, 2017, Theorem 3) does not apply to variant (1) of the framework. The other (De Haan, 2018, Proposition 4) is aimed at a much more restricted setting and has a much more involved proof. To avoid an unnecessarily involved proof, we provide a new proof of Theorem 4.8.

17. The proof of Theorem 4.11 is based on the original membership proof for the outcome determination problem for the AvgGeo rule that appeared in preliminary work (De Haan & Slavkovik, 2017, Lemma 2).

Lemma 4.9. *The following problem is in* coNP. *Given an agenda* Φ, *an output constraint* Γ_{out}, *a profile* $\mathbf{J}$, *and two judgment sets* $J, J' \in \mathcal{J}(\Phi, \Gamma_{\text{out}})$, *decide if there is an edge from* J *to* J' *in the geodesic graph for* Φ *and* Γ_{out}.

Proof. We show that the complementary problem is in NP. There is no edge from J to J' in the geodesic graph if and only if there is some complete and Γ_{out}-consistent judgment set $J'' \subseteq \Phi$ such that for each $\varphi \in \Phi$, J'' agrees with J or J' (or both). We can nondeterministically guess such a set J'', together with a truth assignment witnessing that J'' is Γ_{out}-consistent, and verifying in polynomial time that for each $\varphi \in \Phi$, J'' agrees with at least one of J or J'. This shows that the complementary problem is in NP, and thus the problem is in coNP. □

Lemma 4.10. *The following problem is in* Σ_2^{p}. *Given an instance* $(\Phi, \Gamma_{\text{in}}, \Gamma_{\text{out}}, \mathbf{J}, L, L_1, \ldots, L_u)$ *of variant (a) of the outcome determination problem, and an integer* k *(given in unary), decide if there is a judgment set* $J \in \mathcal{J}(\Phi, \Gamma_{\text{out}})$ *such that* $L \subseteq J$, *such that* $L_i \nsubseteq J$ *for each* $1 \leq i \leq u$, *and such that* $\sum_{J_i \in \mathbf{J}} G(J, J_i) \leq k$.

Proof. We show this by describing a nondeterministic polynomial-time algorithm that has access to an NP oracle, and that decides this problem. The algorithm guesses a judgment set $J^* \subseteq \Phi$ and a truth assignment $\alpha : \text{var}(\Phi) \cup \text{var}(\Gamma_{\text{out}}) \to \{0, 1\}$, and it checks that J^* is complete that α satisfies each $\varphi \in J^*$ and Γ_{out}. In addition, it checks whether $L \subseteq J^*$ and $L_i \nsubseteq J^*$ for each $1 \leq i \leq u$. Moreover, the algorithm guesses $|\mathbf{J}|$ natural numbers $0 \leq \ell_1, \ldots, \ell_{|\mathbf{J}|} \leq k$ such that $\ell = \sum_{1 \leq i \leq |\mathbf{J}|} \ell_i \leq k$, and it guesses a judgment set $J_{i,j} \subseteq \Phi$ and a truth assignment $\alpha_{i,j} : \text{var}(\Phi) \cup \text{var}(\Gamma_{\text{out}}) \to \{0, 1\}$ for each $1 \leq i \leq |\mathbf{J}|$ and each $1 \leq j \leq \ell_i$. The algorithm then verifies (1) whether each judgment set $J_{i,j}$ is complete, and (2) whether the assignment $\alpha_{i,j}$ satisfies each $\varphi \in J_{i,j}$ as well as Γ_{out}, for each $1 \leq i \leq |\mathbf{J}|$ and $1 \leq j \leq \ell_i$. Moreover, it verifies (3) whether for each $1 \leq i \leq |\mathbf{J}|$ and $0 \leq j \leq \ell_i$, there is an edge from $J_{i,j}$ to $J_{i,j+1}$ in the geodesic graph. It does so by querying an oracle for the problem in Lemma 4.9 for each edge. We know that condition (3) is satisfied if and only if the oracle answers "yes" for all queries. Since the problem in Lemma 4.9 is in coNP, we know that an NP oracle suffices. The algorithm accepts if and only if all conditions (1)–(3) are satisfied. It is straightforward to verify that $\sum_{J_i \in \mathbf{J}} G(J^*, J_i) \leq k$ if and only if there exists some guess that satisfies conditions (1)–(3). Thus, the problem is in Σ_2^{p}. □

Theorem 4.11. *The outcome determination problem for the* AVGGEO *rule is in* Θ_3^{p}.

Proof. As described in Sections 2.1 and 2.2, we will use variant (6) of the framework and variant (a) of the problem to establish the upper bound on the complexity. In order to show membership in Θ_3^{p}, we describe how to solve the outcome determination problem for the AVGGEO rule in polynomial time by querying a Σ_2^{p} oracle at most $O(\log |\mathbf{J}| + \log |\Phi|)$ times. We know that the maximum value of $\sum_{J_i \in \mathbf{J}} G(J, J_i)$ for any judgment set J is upper bounded by $|\mathbf{J}| \cdot |\Phi|$. Therefore, by using a logarithmic number of queries to an oracle for the problem in Lemma 4.10, we can determine the minimum value k^* for $\sum_{J_i \in \mathbf{J}} G(J, J_i)$ using binary search. Then, with one additional query to this oracle we can determine whether there exists a judgment set $J^* \in \mathcal{J}(\Phi, \Gamma_{\text{out}})$ such that (1) $\sum_{J_i \in \mathbf{J}} G(J, J_i) \leq k^*$, (2) $L \subseteq J^*$, and (3) $L_i \nsubseteq J^*$ for all $1 \leq i \leq u$. Thus, the outcome determination problem for the AVGGEO rule is in Θ_3^{p}. □

To show Θ_3^{p}-hardness for the outcome determination problem for the AVGGEO rule, we introduce the following auxiliary problem—which is a variant of QSAT$_2$-MIN-MODEL—and show that it is

Θ_3^p-hard. Let $\chi = \exists X.\forall Y.\psi$ be a quantified Boolean formula. Moreover, let $\beta_0 : Y \rightarrow \{0,1\}$ be the truth assignment that sets all variables in Y to false, and let $\beta_1 : Y \rightarrow \{0,1\}$ be the truth assignment that sets all variables in Y to true. We say that χ is *normal* if $\psi[\beta_0]$ and $\psi[\beta_1]$ are both unsatisfiable. We say that a truth assignment $\beta : Y \rightarrow \{0,1\}$ is *normal* if $\beta \neq \beta_0$ and $\beta \neq \beta_1$. If χ is normal, we say that χ is *true* if there exists a truth assignment $\alpha : X \rightarrow \{0,1\}$ such that for all normal truth assignments $\beta : Y \rightarrow \{0,1\}$ it holds that $\alpha \cup \beta$ satisfies ψ.[18]

QSAT$_2$-MIN-MODEL$'$
Instance: A true instance $\chi = \exists X.\forall Y.\psi$ of QSAT$_2$ that is normal, and a variable $x^* \in X$.
Question: Is there a truth assignment $\alpha : X \rightarrow \{0,1\}$ with the property that
  ($\star$) for all normal truth assignments $\beta : Y \rightarrow \{0,1\}$ it holds that $\alpha \cup \beta$ satisfies ψ,
such that α sets a minimal number of variables in X to true—among all assignments $\alpha' : X \rightarrow \{0,1\}$ with property ($\star$)—and such that α sets x^* to true?

The following example illustrates the problem QSAT$_2$-MIN-MODEL$'$.

Example 4.12. Consider the following two quantified Boolean formulas χ_1 and χ_2, involving the sets $X = \{x_1, x_2\}$ and $Y = \{y_1, y_2\}$ of variables:

$$\chi_1 = \exists X.\forall Y.\psi_1 \quad \text{where} \quad \psi_1 = \Big((y_1 \wedge \neg y_2) \vee (y_1 \wedge y_2)\Big) \wedge (x_1 \vee y_1) \wedge (x_1 \vee x_2)$$

$$\chi_2 = \exists X.\forall Y.\psi_2 \quad \text{where} \quad \psi_2 = \Big((y_1 \wedge \neg y_2) \vee (\neg y_1 \wedge y_2)\Big) \wedge (x_1 \vee y_1) \wedge (x_1 \vee x_2)$$

The formula χ_1 is not normal, because $\psi_1[\beta_1]$ is satisfiable. The formula χ_2 is normal, because $\psi_2[\beta_0]$ and $\psi_2[\beta_1]$ are both unsatisfiable. Moreover, χ_2 is true, because the truth assignment α with $\alpha(x_1) = 1$ and $\alpha(x_2) = 1$ has property ($\star$)—that is, α combined with any normal truth assignment $\beta : Y \rightarrow \{0,1\}$ satisfies ψ_2. Therefore, (χ_2, x_1) and (χ_2, x_2) are valid instances of the problem QSAT$_2$-MIN-MODEL$'$.

In the case of χ_2, there are exactly two truth assignments to the variables in X that have property ($\star$)—namely α_1 and α_2, where $\alpha_1(x_1) = \alpha_2(x_1) = 1$, $\alpha_1(x_2) = 0$ and $\alpha_2(x_2) = 1$. Among these two, only α_1 sets a minimal number of variables in X to true (namely one). Thus, (χ_2, x_1) is a yes-instance of the problem QSAT$_2$-MIN-MODEL$'$, and (χ_2, x_2) is a no-instance.

Lemma 4.13. QSAT$_2$-MIN-MODEL$'$ *is* Θ_3^p-*hard.*

Proof. We show that QSAT$_2$-MIN-MODEL$'$ is Θ_3^p-hard by giving a reduction from QSAT$_2$-MIN-MODEL. Let (χ, x^*) be an instance of QSAT$_2$-MIN-MODEL, where $\chi = \exists X.\forall Y.\psi$ and $x^* \in X$. We construct the following instance (χ', x) of QSAT$_2$-MIN-MODEL$'$, defined by letting $Y' = Y \cup \{y_0, y_0'\}$ for fresh variables y_0, y_0', and by letting $\chi' = \exists X.\forall Y'.\psi'$, where:

$$\psi' = \left(\bigvee_{y \in Y'} y\right) \wedge \left(\bigvee_{y \in Y'} \neg y\right) \wedge \big((y_0 \leftrightarrow \neg y_0') \rightarrow \psi\big).$$

It is straightforward to verify that χ' is normal. Take an arbitrary $\alpha : X \rightarrow \{0,1\}$. We show that $\forall Y.\psi[\alpha]$ is true if and only if α has property ($\star$) for the instance χ'.

---

18. In other words, the notion of truth for quantified Boolean formulas $\exists X.\forall Y.\psi$ that we call normal is restricted to normal truth assignments to the variables in Y (and still refers to all truth assignments to the variables in X).

($\Rightarrow$) Suppose that $\forall Y.\psi[\alpha]$ is true. We show that α has property ($\star$) for the instance χ'. Take an arbitrary truth assignment $\beta : Y' \to \{0,1\}$ such that $\beta \neq \beta_0$ and $\beta \neq \beta_1$. Since $\forall Y.\psi[\alpha]$ is true and $Y' \supseteq Y$, we know that $\alpha \cup \beta$ satisfies ψ, and thus that $\alpha \cup \beta$ satisfies ψ'.

($\Leftarrow$) Conversely, suppose that α has property ($\star$) for the instance χ'. We show that $\forall Y.\psi[\alpha]$ is true. Take an arbitrary truth assignment $\beta : Y \to \{0,1\}$. Then consider the truth assignment $\beta' : Y' \to \{0,1\}$ such that $\beta'(y_0) = 0$, $\beta'(y_0') = 1$, and $\beta'(y) = \beta(y)$ for all $y \in Y$. Since α has property ($\star$) and since $\beta' \neq \beta_0$ and $\beta' \neq \beta_1$, we know that $\alpha \cup \beta'$ satisfies ψ'. Moreover, since β' satisfies $(y_0 \leftrightarrow \neg y_0')$, we know that β' satisfies $\psi[\alpha]$. Because β' agrees with β on all variables in $\psi[\alpha]$, we know that β satisfies $\psi[\alpha]$. Since we chose β arbitrarily, we can conclude that $\forall Y.\psi[\alpha]$ is true. $\qquad\square$

We are now ready to show Θ_3^p-hardness of outcome determination for the AvGGEO rule.[19]

Theorem 4.14. *The outcome determination problem for the* AvGGEO *rule is* Θ_3^p*-hard.*

Proof. As described in Sections 2.1 and 2.2, we will use variant (1) of the framework and variant (b) of the problem to establish the lower bound on the complexity. We give a polynomial-time reduction from QSAT$_2$-MIN-MODEL′. Take an instance $(\exists X.\forall Y.\psi, x^*)$ of QSAT$_2$-MIN-MODEL′, where $\exists X.\forall Y.\psi$ is a true quantified Boolean formula, where ψ is quantifier-free, where $x^* \in X = \{x_1, \ldots, x_n\}$ and $Y = \{y_1, \ldots, y_m\}$, and where $\exists X.\forall Y.\psi$ is normal. We say that the *weight* of a truth assignment $\omega : X \to \{0,1\}$ is the number of variables $x \in X$ such that $\omega(x) = 1$. We construct an instance $(\Phi, \Gamma_{\text{in}}, \Gamma_{\text{out}}, \boldsymbol{J}, \varphi^*)$ of the winner determination problem for the AvGGEO rule as follows.

We introduce variables $y_{j,i}$ and $z_{j,i}$ for each $1 \leq j \leq m$ and each $1 \leq i \leq 4n$. We define the agenda Φ by letting $[\Phi] = \{x_1, \ldots, x_n\} \cup \{y_{j,i}, z_{j,i} \mid 1 \leq j \leq m, 1 \leq i \leq 4n\}$. We then construct the integrity constraints $\Gamma_{\text{in}} = \Gamma_{\text{out}}$ as follows:

$$
\Gamma_{\text{in}} = \Gamma_{\text{out}} = \left(\begin{array}{l} \left(\neg\psi_1 \wedge \bigwedge_{\substack{1 \leq j \leq m \\ 1 \leq i \leq 4n}} \neg y_{j,i} \wedge \left(\left(\bigwedge_{1 \leq j \leq m} \zeta_j \right) \to \bigwedge_{\substack{1 \leq j \leq m \\ 1 \leq i \leq 4n}} z_{j,i} \right) \right) \vee \\[2em] \left(\neg\psi_2 \wedge \bigwedge_{\substack{1 \leq j \leq m \\ 1 \leq i \leq 4n}} \neg z_{j,i} \wedge \left(\left(\bigwedge_{1 \leq j \leq m} \zeta_j' \right) \to \bigwedge_{\substack{1 \leq j \leq m \\ 1 \leq i \leq 4n}} y_{j,i} \right) \right) \end{array} \right),
$$

where $\zeta_j = \bigvee_{1 \leq i \leq 4n} z_{j,i}$ and $\zeta_j' = \bigvee_{1 \leq i \leq 4n} y_{j,i}$ for each $1 \leq j \leq m$, where ψ_1 is obtained from ψ by replacing each occurrence of $y_j \in Y$ with ζ_j and where ψ_2 is obtained from ψ by replacing each occurrence of $y_j \in Y$ with ζ_j'. We let $\boldsymbol{J} = (J_1, J_2, J_3)$, where $J_1 = \{\neg x_i \mid 1 \leq i \leq n\} \cup \{y_{j,i}, \neg z_{j,i} \mid 1 \leq j \leq m, 1 \leq i \leq 4n\}$, where $J_2 = \{\neg x_i \mid 1 \leq i \leq n\} \cup \{z_{j,i}, \neg y_{j,i} \mid 1 \leq j \leq m, 1 \leq i \leq 4n\}$, and where $J_3 = \{\neg\varphi \mid \varphi \in [\Phi]\}$. It is straightforward to verify that J_1, J_2 and J_3 are Γ_{in}-consistent. Finally, we let $\varphi^* = x^*$.

In the remainder of the proof, we will show that the judgment sets in AvGGEO($\boldsymbol{J}$) are in correspondence with the truth assignments $\omega : X \to \{0,1\}$ with property ($\star$) of minimal weight. Take any truth assignment $\omega : X \to \{0,1\}$. Consider the following judgment set:

$$
\begin{aligned}
J^\omega = \; & \{x_i \mid 1 \leq i \leq n, \omega(x_i) = 1\} \cup \{\neg x_i \mid 1 \leq i \leq n, \omega(x_i) = 0\} \cup \\
& \{\neg y_{j,i}, z_{j,i} \mid 1 \leq j \leq m, 1 \leq i \leq 4n\}.
\end{aligned}
$$

---

19. The proof of Theorem 4.14 is not based on the original hardness proof for the outcome determination problem for the AvGGEO rule that appeared in preliminary work (De Haan & Slavkovik, 2017, Theorem 4)—the original proof only applies to variants (2), (3) and (4) of the framework.

For each $\omega : X \to \{0,1\}$ of weight w that has property $(\star)$, it holds that $G(J^\omega, J_3) = w$ and $G(J^\omega, J_1) = G(J^\omega, J_2) = w + 1$, because there is no Γ_{out}-consistent judgment set between J^ω and J_1^ω or J_2^ω in the geodesic graph, where:

$$
\begin{aligned}
J_1^\omega = \ & \{\, x_i \mid 1 \le i \le n, \omega(x_i) = 1 \,\} \cup \{\, \neg x_i \mid 1 \le i \le n, \omega(x_i) = 0 \,\} \cup \\
& \{\, y_{j,i}, \neg z_{j,i} \mid 1 \le j \le m, 1 \le i \le 4n \,\}, \text{ and} \\
J_2^\omega = \ & \{\, x_i \mid 1 \le i \le n, \omega(x_i) = 1 \,\} \cup \{\, \neg x_i \mid 1 \le i \le n, \omega(x_i) = 0 \,\} \cup \\
& \{\, \neg y_{j,i}, z_{j,i} \mid 1 \le j \le m, 1 \le i \le 4n \,\}.
\end{aligned}
$$

Then the judgment sets J^ω for truth assignments $\omega : X \to \{0,1\}$ with property $(\star)$ of minimal weight w_0 have cumulative geodesic distance $3w_0 + 2$ to the profile $\boldsymbol{J}$.

In order to show that there are no Γ_{out}-consistent judgment sets J with smaller cumulative geodesic distance to the profile $\boldsymbol{J}$, we argue that every path in the geodesic graph from J_1 to J_2 that does not go through any judgment set J^ω—where ω is a truth assignment that has property $(\star)$—has length at least $4n$. This is the case because (i) this path has to go through some judgment set J' with $\{\, \neg y_{j,i}, z_{j,i} \mid 1 \le j \le m, 1 \le i \le n \,\} \subseteq J'$, (ii) there is some Γ_{out}-consistent J'' between J' and J_1 that includes $\{\, y_{j,i} \mid 1 \le i \le n \,\}$ for some y_j, and (iii) the distance from J'' to J' is at least $4n$. We may assume without loss of generality that $n > 2$, and thus that $4n > 3w_0 + 2$.

Because for every $\omega : X \to \{0,1\}$, the judgment set J^ω is Γ_{out}-consistent, we know that any path from J_1 to J_2 must go through J^ω for some $\omega : X \to \{0,1\}$. Moreover, for each path from J_1 to J_2 it holds that the closest judgment set on this path to J_3 is a set of the form J^ω, for some ω. Therefore, we know that any judgment set that minimizes the cumulative distance to the profile $\boldsymbol{J} = (J_1, J_2, J_3)$ is of the form J^ω for some $\omega : X \to \{0,1\}$ that has property $(\star)$. Moreover, the cumulative distance to such a set J^ω is $3w + 2$, where w is the weight of ω. Thus, AvgGeo$(\boldsymbol{J})$ consists of the judgment sets J^ω for all those ω with property $(\star)$ of minimal weight.

This means that there is a judgment set $J^* \in$ AvgGeo$(\boldsymbol{J})$ with $x^* = \varphi^* \in J^*$ if and only if there is a truth assignment $\alpha : X \to \{0,1\}$ with property $(\star)$ that sets a minimal number of variables in X and that sets x^* to true. $\qquad \square$

The membership proof for the outcome determination problem for the MaxGeo rule is similar to the proof of the membership result for the AvgGeo rule (Theorem 4.11). To establish membership in Θ_3^{p}, we use an additional lemma.

Lemma 4.15. *The following problem is in Σ_2^{p}. Given an instance $(\Phi, \Gamma_{\text{in}}, \Gamma_{\text{out}}, \boldsymbol{J}, L, L_1, \ldots, L_u)$ of variant (a) of the outcome determination problem, and an integer k (given in unary), decide if there is a judgment set $J \in \mathcal{J}(\Phi, \Gamma_{\text{out}})$ such that $L \subseteq J$, such that $L_i \not\subseteq J$ for each $1 \le i \le u$, and such that $\max_{J_i \in \boldsymbol{J}} G(J, J_i) \le k$.*

Proof. We show this by describing a nondeterministic polynomial-time algorithm that has access to an NP oracle, and that decides this problem. The algorithm guesses a judgment set $J^* \subseteq \Phi$ and a truth assignment $\alpha : \text{var}(\Phi) \cup \text{var}(\Gamma_{\text{out}}) \to \{0,1\}$, and it checks that J^* is complete that α satisfies each $\varphi \in J^*$ and Γ_{out}. In addition, it checks whether $L \subseteq J^*$ and $L_i \not\subseteq J^*$ for each $1 \le i \le u$. Moreover, the algorithm guesses $|\boldsymbol{J}|$ natural numbers $0 \le \ell_1, \ldots, \ell_{|\boldsymbol{J}|} \le k$ such that $\ell = \max_{1 \le i \le |\boldsymbol{J}|} \ell_i \le k$, and it guesses a judgment set $J_{i,j} \subseteq \Phi$ and a truth assignment $\alpha_{i,j} : \text{var}(\Phi) \cup \text{var}(\Gamma_{\text{out}}) \to \{0,1\}$ for each $1 \le i \le |\boldsymbol{J}|$ and each $1 \le j \le \ell_i$. The algorithm then verifies (1) whether each judgment set $J_{i,j}$ is complete, and (2) whether the assignment $\alpha_{i,j}$ satisfies each $\varphi \in J_{i,j}$ as well as Γ_{out}, for each $1 \le i \le |\boldsymbol{J}|$ and $1 \le j \le \ell_i$. Moreover, it verifies (3) whether for each $1 \le i \le |\boldsymbol{J}|$ and $0 \le j \le \ell_i$,

there is an edge from $J_{i,j}$ to $J_{i,j+1}$ in the geodesic graph. It does so by querying an oracle for the problem in Lemma 4.9 for each edge. We know that condition (3) is satisfied if and only if the oracle answers "yes" for all queries. Since the problem in Lemma 4.9 is in coNP, we know that an NP oracle suffices. The algorithm accepts if and only if all conditions (1)–(3) are satisfied. It is straightforward to verify that $\max_{J_i \in \boldsymbol{J}} G(J^*, J_i) \leq k$ if and only if there exists some guess that satisfies conditions (1)–(3). Thus, the problem is in Σ_2^{p}. $\square$

Theorem 4.16. *The outcome determination problem for the* MAXGEO *rule is in* Θ_3^{p}.

Proof. As described in Sections 2.1 and 2.2, we will use variant (6) of the framework and variant (a) of the problem to establish the upper bound on the complexity. This proof is entirely similar to the proof of Theorem 4.11. In order to show membership in Θ_3^{p}, we describe how to solve the outcome determination problem for the MAXGEO rule in polynomial time by querying a Σ_2^{p} oracle at most $O(\log |\Phi|)$ times. We know that the maximum value of $\max_{J_i \in \boldsymbol{J}} G(J, J_i)$ for any judgment set J is upper bounded by $|\Phi|$. Therefore, by using a logarithmic number of queries to an oracle for the problem in Lemma 4.15, we can determine the minimum value k^* for $\max_{J_i \in \boldsymbol{J}} G(J, J_i)$ using binary search. Then, with one additional query to this oracle we can determine whether there exists a judgment set $J^* \in \mathcal{J}(\Phi, \Gamma_{\mathrm{out}})$ such that (1) $\max_{J_i \in \boldsymbol{J}} G(J, J_i) \leq k^*$, (2) $L \subseteq J^*$, and (3) $L_i \nsubseteq J^*$ for all $1 \leq i \leq u$. Thus, the outcome determination problem for the MAXGEO rule is in Θ_3^{p}. $\square$

We show Θ_3^{p}-hardness for the outcome determination problem for the MAXGEO rule similarly to the hardness proof for the AVGGEO rule (Theorem 4.14).

Theorem 4.17. *The outcome determination problem for the* MAXGEO *rule is* Θ_3^{p}-*hard.*

Proof. As described in Sections 2.1 and 2.2, we will use variant (1) of the framework and variant (b) of the problem to establish the lower bound on the complexity. We use the same reduction as used for the AVGGEO rule in the proof of Theorem 4.14—that is, for each instance $(\exists X.\forall Y.\psi, x^*)$ of QSAT$_2$-MIN-MODEL$'$ we construct the same instance $(\Phi, \Gamma_{\mathrm{in}}, \Gamma_{\mathrm{out}}, \boldsymbol{J}, \varphi^*)$ as described in the proof of Theorem 4.14.

We argue that this reduction also works correctly for the MAXGEO rule. For any $\omega : X \to \{0, 1\}$ with weight w that has property $(\star)$, it holds that $\max_{J_i \in \boldsymbol{J}} G(J^\omega, J_i) = w + 1$. For any $\omega : X \to \{0, 1\}$ with weight w that does not have property $(\star)$, it holds that $\max_{J_i \in \boldsymbol{J}} G(J^\omega, J_i) > w_0 + 1$, where w_0 is the minimal weight of any truth assignment $\omega' : X \to \{0, 1\}$ that has property $(\star)$. Similarly, for any Γ_{out}-consistent judgment set J that does not coincide with J^ω for any $\omega : X \to \{0, 1\}$, it holds that $\max_{J_i \in \boldsymbol{J}} G(J^\omega, J_i) > w_0 + 1$. From this, we can conclude that for the constructed profile $\boldsymbol{J}$ it holds that MAXGEO($\boldsymbol{J}$) = AVGGEO($\boldsymbol{J}$), and thus that the reduction is correct. $\square$

4.6 Outcome Determination for the Reversal-Scoring Rule

Next, we show Θ_2^{p}-membership for the outcome determination problem for the reversal-scoring rule REVSCO.

Theorem 4.18. *The outcome determination problem for the reversal-scoring rule* REVSCO *is in* Θ_2^{p}.

Proof. As described in Sections 2.1 and 2.2, we will use variant (6) of the framework and variant (a) of the problem to establish the upper bound on the complexity. We describe a polynomial-time algorithm that queries an NP oracle $O(\log |\Phi| + \log n)$ times and that solves the problem. Roughly, the algorithm

does the following. It first computes the maximum total reversal score for any $J^* \in \mathcal{J}(\Phi, \Gamma_{\text{out}})$—that is, the maximum value of $\text{REV}(\boldsymbol{J}, J^*) = \sum_{J_i \in \boldsymbol{J}, \varphi \in J^*} rev(J_i, \varphi)$ for some $J^* \in \mathcal{J}(\Phi, \Gamma_{\text{out}})$. Then, using this maximum value, the algorithm checks whether there exists some $J^* \in \mathcal{J}(\Phi, \Gamma_{\text{out}})$ such that (i) $\text{REV}(\boldsymbol{J}, J^*)$ reaches the maximum value, such that (ii) $L \subseteq J^*$ and such that (iii) $L_i \nsubseteq J^*$ for all $i \in \{1, \ldots, u\}$.

The algorithm proceeds to find the maximum value of $\text{REV}(\boldsymbol{J}, J^*)$ in two phases. In the first phase, it queries the NP oracle to determine the following value:

$$k_0 = \sum_{J_i \in \boldsymbol{J}} \sum_{\varphi \in \Phi} rev(J_i, \varphi).$$

To do so, the algorithm uses oracle queries to check whether there exist judgment sets $J^{\bullet}_{i,\varphi} \in \mathcal{J}(\Phi, \Gamma_{\text{out}})$, for each $J_i \in \boldsymbol{J}$ and each $\varphi \in \Phi$, such that $\sum_{J_i \in \boldsymbol{J}, \varphi \in \Phi} H(J_i, J^{\bullet}_{i,\varphi}) \leq k$ for a given value k. This is a problem in NP, so by picking an NP-complete problem for the oracle, we can use a single oracle query to solve any instance of the above problem in polynomial time. The maximum value of $\sum_{J_i \in \boldsymbol{J}, \varphi \in \Phi} H(J_i, J^{\bullet}_{i,\varphi})$ is at most $|\Phi|^2 \cdot n$, where n is the number of judgment sets in the profile $\boldsymbol{J}$. Therefore, we can determine the minimum value k_0 of $\sum_{J_i \in \boldsymbol{J}, \varphi \in \Phi} H(J_i, J^{\bullet}_{i,\varphi})$ using $O(\log |\Phi| + \log n)$ oracle queries—by using binary search.

Then, in the second phase, the algorithm queries the NP oracle to determine the following value:

$$k_{\max} = \max_{J^* \in \mathcal{J}(\Phi, \Gamma_{\text{out}})} \text{REV}(\boldsymbol{J}, J^*) \quad \text{where } \text{REV}(\boldsymbol{J}, J^*) = \sum_{J_i \in \boldsymbol{J}} \sum_{\varphi \in J^*} rev(J_i, \varphi).$$

To do so, the algorithm uses oracle queries to check whether there exists a judgment set $J^* \in \mathcal{J}(\Phi, \Gamma_{\text{out}})$ such that $\text{REV}(\boldsymbol{J}, J^*) \geq k$ for a given value k. When the value of k_0 is known, this is a problem in NP. To solve the problem, one can guess judgment sets $J^{\bullet}_{i,\varphi} \in \mathcal{J}(\Phi, \Gamma_{\text{out}})$, for each $J_i \in \boldsymbol{J}$ and each $\varphi \in \Phi$, and verify that the sets $J^{\bullet}_{i,\varphi}$ correspond to the value for k_0. If this check succeeds, the value of $rev(J_i, \varphi)$ is equal to $H(J_i, J^{\bullet}_{i,\varphi})$, for each $J_i \in \boldsymbol{J}$ and each $\varphi \in \Phi$. Then, one can guess a judgment set $J^* \in \mathcal{J}(\Phi, \Gamma_{\text{out}})$ and use the sets $J^{\bullet}_{i,\varphi}$ to verify that $\text{REV}(\boldsymbol{J}, J^*) \geq k$ in polynomial time. Thus, by picking an NP-complete problem for the oracle, we can use a single oracle query to solve any instance of the above problem in polynomial time. The maximum value of $\text{REV}(\boldsymbol{J}, J^*)$ is at most $|\Phi|^2 \cdot n$, where n is the number of judgment sets in the profile $\boldsymbol{J}$. Therefore, we can determine the maximum value $k_{\max}$ of $\text{REV}(\boldsymbol{J}, J^*)$ using $O(\log |\Phi| + \log n)$ oracle queries—by using binary search.

Then, after having identified the maximum value $k_{\max}$ of $\text{REV}(\boldsymbol{J}, J^*)$ for any $J^* \in \mathcal{J}(\Phi, \Gamma_{\text{out}})$, the algorithm queries the oracle one additional time to determine whether there exists a complete and Γ_{out}-consistent judgment set J^* such that $\text{REV}(\boldsymbol{J}, J^*) = k_{\max}$, such that $L \subseteq J^*$ and such that $L_i \nsubseteq J^*$ for all $i \in \{1, \ldots, u\}$. Again, since this is a problem in NP, one oracle query suffices to determine this. The algorithm outputs "yes" if and only if such a judgment set J^* exists. This algorithm runs in polynomial time, queries the NP oracle $O(\log |\Phi| + \log n)$ times, and correctly solves the problem. Therefore, the problem is in Θ^p_2. □

We modify the proof of Theorem 4.4 to give us a proof of Θ^p_2-hardness for the outcome determination problem for the reversal scoring rule.[20]

---

20. The proof of Theorem 4.19 is not based on the original hardness proof for the outcome determination problem for the reversal scoring rule that appeared in preliminary work (De Haan & Slavkovik, 2017, Theorem 2). We find that a proof that is based on a modification of the proof of Theorem 4.4 is easier to follow.

Theorem 4.19. *The outcome determination problem for the reversal-scoring rule* REVSCO *is* Θ_2^p*-hard.*

Proof. As described in Sections 2.1 and 2.2, we will use variant (1) of the framework and variant (b) of the problem to establish the lower bound on the complexity. We give a reduction from the Θ_2^p-complete problem MAX-MODEL. To do so, we will modify the reduction used in the proof of Theorem 4.4. Let (ψ, x^*) be an instance of MAX-MODEL, with $\mathrm{var}(\psi) = \{x_1, \ldots, x_n\}$. Without loss of generality, suppose that $x^* = x_1$. Moreover, without loss of generality, suppose that the satisfying assignment of ψ that sets the maximal number of variables in $\mathrm{var}(\psi)$ to true sets at least $\frac{1}{2}n$ and at most $\frac{3}{4}n$ variables in $\mathrm{var}(\psi)$ to true. We can ensure this property by (conjunctively) adding an appropriate number of fresh unit literals to ψ.

We construct an instance $(\Phi, \Gamma_{\mathrm{in}}, \Gamma_{\mathrm{out}}, \boldsymbol{J}, \varphi^*)$ of the outcome determination problem for the reversal scoring rule by following the construction in the proof of Theorem 4.4, with one modification: to make the analysis easier, we use $u = n$ (rather than $u = n + 1$).

We argue that $\mathrm{REVSCO}(\boldsymbol{J})$ is in one-to-one correspondence with the set of truth assignments satisfying ψ that set a maximal number of variables in $\mathrm{var}(\psi)$ to true. To do so, we analyse the value of $\sigma(\varphi) = \sum_{J_i \in \boldsymbol{J}} rev(J_i, \varphi)$, for each $\varphi \in \Phi$. For each $1 \leq i \leq n$, it holds that $\sigma(x_i) = n - 1$ and $\sigma(\neg x_i) = 1$. Moreover, for each $1 \leq i \leq n$ and each $1 \leq j \leq u$, it holds that $\sigma(z_{i,j}) = n - b - 1$ and $\sigma(\neg z_{i,j}) = n - 1$, where b is the maximal number of variables in $\mathrm{var}(\psi)$ that are set to true by any truth assignment satisfying ψ. The majoritarian judgment set $m(\boldsymbol{J})$ maximizes the value of $\sum_{\varphi \in \Phi} \sigma(\varphi)$, but it does not satisfy Γ_{out}. To achieve Γ_{out}-consistency, while deviating as little as possible from $m(\boldsymbol{J})$, judgment sets can either (i) satisfy ψ or (ii) satisfy $\bigwedge_{1 \leq j \leq u} z_{i,j}$ for some $1 \leq i \leq n$. The optimal total reversal score of a judgment set J that satisfies ψ (and agrees with $m(\boldsymbol{J})$ on $\bigwedge_{1 \leq i \leq n, 1 \leq j \leq u} \neg z_{i,j}$) is $S_1 = n + bn - 2b + nu(n - 1)$. The optimal total reversal score of a judgment set J that satisfies $\bigwedge_{1 \leq j \leq u} z_{i,j}$ for some $1 \leq i \leq n$ (and agrees with $m(\boldsymbol{J})$ on all other issues) is $S_2 = n(n - 1) + u(n - b - 1) + u(n - 1)^2$. By our assumption that $n/2 \leq b \leq 3n/4$, and by our choice that $u = n$, we get that $S_1 \geq n^3 - n^2/2 - 3n/2$ and $S_2 \leq n^3 - n^2/2 - n$. This means that for $n \geq 1$ we get that $S_1 \geq S_2$, and thus the judgment sets J^* with optimal total reversal score satisfy ψ by setting a maximal number of variables in $\mathrm{var}(\psi)$ to true. Therefore, we get that there is some $J^* \in \mathrm{REVSCO}(\boldsymbol{J})$ with $x_1 = \varphi^* \in J^*$ if and only there is a satisfying truth assignment for ψ that satisfies x_1 and that sets a maximal number in $\mathrm{var}(\psi)$ to true. $\square$

4.7 Outcome Determination for the Ranked-Agenda and the LexiMax Rule

We show Σ_2^p-membership for the outcome determination problem for the ranked-agenda rule RAN.

Theorem 4.20. *The outcome determination problem for the ranked-agenda rule* RAN *is in* Σ_2^p.

Proof. As described in Sections 2.1 and 2.2, we will use variant (6) of the framework and variant (a) of the problem to establish the upper bound on the complexity. We describe a nondeterministic polynomial-time algorithm with access to an NP oracle that solves the problem. The algorithm guesses a complete judgment set $J^* \subseteq \Phi$, and guesses a truth assignment α to the variables in Φ and in Γ_{out}. The algorithm then checks that α satisfies Γ_{out} and satisfies φ for each $\varphi \in J^*$, and the algorithm checks that $L \subseteq J^*$ and $L_i \nsubseteq J^*$ for each $i \in \{1, \ldots, u\}$. Moreover, the algorithm guesses a tie-breaking order $\gg \in \mathcal{L}(\Phi)$. Then, for each $\varphi \in \Phi \setminus J^*$, the algorithm constructs the set $(J^*)|_{\gg \varphi}^{\boldsymbol{J}}$—this can be done in polynomial time, given J^* and $\gg$. The algorithm then uses the NP oracle to determine that $(J^*)|_{\gg \varphi}^{\boldsymbol{J}} \cup \{\varphi\}$ is Γ_{out}-inconsistent for each $\varphi \in \Phi \setminus J^*$. This is a problem in coNP, so by picking an NP-complete problem for the oracle, we can use a single oracle query to solve

any instance of the above problem in polynomial time. The algorithm accepts the input if and only if the sets $(J^*)|_{\gg_\varphi}^J \cup \{\varphi\}$ for all $\varphi \in \Phi \setminus J^*$ are Γ_{out}-inconsistent. By definition of the ranked-agenda rule, the algorithm accepts the input if and only if $J^* \in \text{RAN}(J)$. □

Next, we prove Σ_2^p-hardness for the outcome determination problem for the ranked agenda rule RAN.[21]

Theorem 4.21. *The outcome determination problem for the ranked-agenda rule* RAN *is* Σ_2^p-*hard.*

Proof. As described in Sections 2.1 and 2.2, we will use variant (1) of the framework and variant (b) of the problem to establish the lower bound on the complexity. We give a reduction from the Σ_2^p-complete problem QSAT$_2$. Let $\varphi = \exists x_1 \ldots \exists x_n \forall y_1 \ldots \forall y_m.\psi$ be an instance of QSAT$_2$. We construct an instance $(\Phi, \Gamma_{\text{in}}, \Gamma_{\text{out}}, J, \varphi^*)$ of the outcome determination problem for the ranked-agenda rule as follows. We define Φ by letting $[\Phi] = \{ z_{x_i}, z_{\overline{x_i}} \mid 1 \leq i \leq n \} \cup \{ z_{y_j}, z_{\overline{y_j}} \mid 1 \leq j \leq m \} \cup \{w_1, \ldots, w_9, u\}$. We define $\Gamma_{\text{in}} = \Gamma_{\text{out}}$ as follows:

$$\Gamma_{\text{in}} = \Gamma_{\text{out}} = \bigvee_{i=1}^{9} w_i \vee \left(\bigwedge_{i=1}^{n} (z_{x_i} \leftrightarrow \neg z_{\overline{x_i}}) \wedge \bigwedge_{j=1}^{m} (z_{y_j} \leftrightarrow \neg z_{\overline{y_j}}) \wedge (u \rightarrow \neg \psi') \right),$$

where ψ' is obtained from ψ by replacing each variable x_i by z_{x_i} and replacing each variable y_j by z_{y_j}. We define the profile $J = (J_1, \ldots, J_9)$ as indicated in Figure 7. Finally, we let $\varphi^* = \neg u$.

We have that $\{\neg w_1, \ldots, \neg w_9\} \subseteq J^*$ for every $J^* \in \text{RAN}(J)$: the formulas $\neg w_i$, for $1 \leq i \leq 9$, have the highest support $N(J, \neg w_i) = 8$ in the profile, and $\{\neg w_1, \ldots, \neg w_9\}$ is consistent with Γ_{out}. Therefore, each $J^* \in \text{RAN}(J)$ must be consistent with $\bigwedge_{i=1}^{n}(z_{x_i} \leftrightarrow \neg z_{\overline{x_i}}) \wedge \bigwedge_{j=1}^{m}(z_{y_j} \leftrightarrow \neg z_{\overline{y_j}}) \wedge (u \rightarrow \psi')$.

We show that φ is true if and only if there is some $J^* \in \text{RAN}(J)$ with $\varphi^* \in J^*$.

($\Rightarrow$) Suppose that φ is true—that is, that there exists some truth assignment $\alpha : \{x_1, \ldots, x_n\} \rightarrow \{0, 1\}$ such that $\psi[\alpha]$ is a validity. Consider the judgment set $J^* = \{\neg w_1, \ldots, \neg w_9, \neg u\} \cup \{ z_{x_i}, \neg z_{\overline{x_i}} \mid 1 \leq i \leq n, \alpha(x_i) = 1 \} \cup \{ z_{\overline{x_i}}, \neg z_{x_i} \mid 1 \leq i \leq n, \alpha(x_i) = 0 \} \{ z_{y_j}, \neg z_{\overline{y_j}} \mid 1 \leq j \leq m \}$. We argue that $J^* \in \text{RAN}(J)$. We go over the formulas $\varphi \in \Phi$ in order of their support in the profile. The formulas $\neg w_1, \ldots, \neg w_9$ are consistent with Γ_{out}. The formulas $z_{x_1}, \ldots, z_{x_n}, z_{\overline{x_1}}, \ldots, z_{\overline{x_n}}$ all have equal support, and Γ_{out} requires that for each $1 \leq i \leq n$ there is exactly one of z_{x_i} and $z_{\overline{x_i}}$ in each judgment set. Since $\psi[\alpha]$ is a validity, and Γ_{out} entails $(u \rightarrow \neg \psi')$, it holds that $J^* \cap \{ z_{x_i}, \neg z_{x_i}, z_{\overline{x_i}}, \neg z_{\overline{x_i}} \mid 1 \leq i \leq n \} \cup \{u\}$ is Γ_{out}-inconsistent. Finally, including z_{y_j} for each $1 \leq j \leq m$ satisfies Γ_{out}, and ensures that no formula $z_{\overline{y_j}}$ can be added while keeping Γ_{out}-consistency. From this, we get that $J^* \in \text{RAN}(J)$—and we know that $\varphi^* = \neg u \in J^*$.

($\Leftarrow$) Suppose that there is some $J^* \in \text{RAN}(J)$ with $\varphi^* = \neg u \in J^*$. We show that φ is true. We know that $\{\neg w_1, \ldots, \neg w_9\} \subseteq J^*$. Since J^* is Γ_{out}-consistent, we know that J^* must be consistent with $\bigwedge_{1 \leq i \leq n}(z_{x_i} \leftrightarrow \neg z_{\overline{x_i}})$. Because the formulas $z_{x_i}, z_{\overline{x_i}}$ have the next highest support in the profile J, we know that $|J^* \cap \{z_{x_i}, z_{\overline{x_i}}\}| = 1$ for each $1 \leq i \leq n$. Consider the truth assignment $\alpha : \{x_1, \ldots, x_n\} \rightarrow \{0, 1\}$ that is defined by letting $\alpha(x_i) = 1$ if and only if $z_{x_i} \in J^*$, for each $1 \leq i \leq n$. We show that $\psi[\alpha]$ is a validity. Suppose, to derive a contradiction that $\neg \psi[\alpha]$ is satisfiable—that is, that there is a truth assignment $\beta : \{y_1, \ldots, y_m\} \rightarrow \{0, 1\}$ such that $\alpha \cup \beta$

J	J_1	J_2	J_3	J_4	J_5	J_6	J_7	J_8	J_9	$N(J, \varphi)$	$N(J, \neg\varphi)$
z_{x_1}	0	0	1	1	1	1	1	1	1	7	2
$z_{\overline{x_1}}$	0	0	1	1	1	1	1	1	1	7	2
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
z_{x_n}	0	0	1	1	1	1	1	1	1	7	2
$z_{\overline{x_n}}$	0	0	1	1	1	1	1	1	1	7	2
u	0	0	0	1	1	1	1	1	1	6	3
z_{y_1}	0	0	0	0	1	1	1	1	1	5	4
$z_{\overline{y_1}}$	0	0	0	0	1	1	1	1	1	5	4
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
z_{y_m}	0	0	0	0	1	1	1	1	1	5	4
$z_{\overline{y_m}}$	0	0	0	0	1	1	1	1	1	5	4
w_1	1	0	0	0	0	0	0	0	0	1	8
w_2	0	1	0	0	0	0	0	0	0	1	8
w_3	0	0	1	0	0	0	0	0	0	1	8
w_4	0	0	0	1	0	0	0	0	0	1	8
w_5	0	0	0	0	1	0	0	0	0	1	8
w_6	0	0	0	0	0	1	0	0	0	1	8
w_7	0	0	0	0	0	0	1	0	0	1	8
w_8	0	0	0	0	0	0	0	1	0	1	8
w_9	0	0	0	0	0	0	0	0	1	1	8

Figure 7: Construction of the profile J in the proof of Theorem 4.21.

falsifies ψ. Then we get that the set $\{w_1, \ldots, w_9\} \cup (J^* \cap \{z_{x_i}, \neg z_{x_i}, z_{\overline{x_i}}, \neg z_{\overline{x_i}} \mid 1 \le i \le n\}) \cup \{u\}$ is Γ_{out}-consistent. Since u is the formula in Φ with the next highest support, we get a contradiction with the fact that $\neg u \in J^*$. Thus, we can conclude that no such β exists, and therefore that $\psi[\alpha]$ is a validity. Thus, φ is true. $\qquad\square$

We show that the outcome determination problem for the LEXIMAX-rule is in Δ_2^p.

Theorem 4.22. *The outcome determination problem for the* LEXIMAX*-rule is in* Δ_2^p.

Proof. As described in Sections 2.1 and 2.2, we will use variant (6) of the framework and variant (a) of the problem to establish the upper bound on the complexity. We describe a (deterministic) polynomial-time algorithm with access to an NP oracle that solves the problem. The algorithm uses oracle queries to decide, for a given sequence $(\ell_m, \ldots, \ell_{m-k})$ of natural numbers, whether there exists a judgment set $J \in \mathcal{J}(\Phi, \Gamma_{\text{out}})$ such that for each $0 \le i \le k$ it holds that $|\{\varphi \in J \mid N(J, \varphi) \ge \ell_{m-i}\}| = \ell_{m-i}$. This is a problem in NP, so by picking an NP-complete problem for the oracle, we can use a single oracle query to solve any instance of the above problem in polynomial time. The algorithm iteratively computes the sequences $\boldsymbol{\ell}^* = (\ell_m^*, \ldots, \ell_{m-k}^*)$—for increasing $0 \le k < m$—such that the oracle returns "yes" for $\boldsymbol{\ell}^*$, and there is no sequence $\boldsymbol{\ell}' = (\ell_m', \ldots, \ell_{m-k}')$ with $\ell_{m-k'}' > \ell_{m-k'}^*$ for some $0 \le k' \le k$ and $\ell_{m-k''}' = \ell_{m-k''}^*$ for all $0 \le k' \le k''$ such that the oracle returns "yes" for $\boldsymbol{\ell}'$. For $k = 0$, we can find the value of ℓ_m^* by querying the oracle several times—using sequences $(\ell_m'^*)$. Then, for $k > 1$, we

have already identified $(\ell_m^*, \ldots, \ell_{m-k+1}^*)$. We can then find the value of ℓ_{m-k}^* by querying the oracle several times—using sequences $(\ell_m^*, \ldots, \ell_{m-k+1}^*, \ell'_{m-k})$ for different values of ℓ'_{m-k}.

After having identified $(\ell_m^*, \ldots, \ell_1^*)$, we can solve the problem by using another query to the NP oracle. We use the oracle to check whether there exists a judgment set $J^* \in \mathcal{J}(\Phi, \Gamma_{\text{out}})$ such that $L \subseteq J^*$ and $L_j \nsubseteq J^*$ for each $1 \leq j \leq u$, such that for each $0 \leq i < m$ it holds that $|\{ \varphi \in J \mid N(J, \varphi) \geq \ell_{m-i} \}| = \ell_{m-i}^*$. This is a problem in NP, so by picking an NP-complete problem for the oracle, we can use a single oracle query to solve this problem in polynomial time. The oracle returns "yes" to this last query if and only if there exists a judgment set $J^* \in \textsc{LexiMax}(J)$ such that $L \subseteq J^*$ and $L_j \nsubseteq J^*$ for each $1 \leq j \leq u$. Thus, by returning the answer to the last oracle query, this algorithm solves the problem in polynomial time with access to an NP oracle. This concludes our proof that the problem is in Δ_2^p. □

We complement the above membership proof with a Δ_2^p-hardness proof for the outcome determination problem for the LexiMax-rule.

Theorem 4.23. *The outcome determination problem for the* LexiMax-*rule is* Δ_2^p-*hard.*

Proof. As described in Sections 2.1 and 2.2, we will use variant (1) of the framework and variant (b) of the problem to establish the lower bound on the complexity. We give a polynomial-time reduction from the Δ_2^p-complete problem Lex-Max-Model. Let (ψ, x^*) be an instance of Lex-Max-Model, where ψ is a satisfiable formula with $\text{var}(\psi) = \{x_1, \ldots, x_n\}$ and the variables are ordered $x_1 < \cdots < x_n$. We construct an instance $(\Phi, \Gamma_{\text{in}}, \Gamma_{\text{out}}, J, \varphi^*)$ of the outcome determination problem for the LexiMax-rule as follows. We let $\Phi = \{ x_i, \neg x_i \mid 1 \leq i \leq n \} \cup \{ y_j, \neg y_j \mid 1 \leq j \leq 2n+3 \}$. We let $\Gamma_{\text{in}} = \Gamma_{\text{out}} = \psi \vee \bigvee_{1 \leq j \leq 2n+3} y_j$. We construct $J = (J_1, \ldots, J_{2n+3})$ as described in Figure 8. Finally, we let $\varphi^* = x^*$.

J	J_1	$\cdots$	J_{n+1}	J_{n+2}	J_{n+3}	J_{n+4}	$\cdots$	J_{2n-1}	J_{2n}	J_{2n+1}	J_{2n+2}	J_{2n+3}	$N(J, \varphi)$
x_1	1	$\cdots$	1	1	1	1	$\cdots$	1	1	1	0	0	$2n+1$
x_2	1	$\cdots$	1	1	1	1	$\cdots$	1	1	0	0	0	$2n$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
x_{n-1}	1	$\cdots$	1	1	1	0	$\cdots$	0	0	0	0	0	$n+3$
x_n	1	$\cdots$	1	1	0	0	$\cdots$	0	0	0	0	0	$n+2$
y_1	1	0	0	$\cdots$							$\cdots$	0	1
y_2	0	1	0	$\cdots$							$\cdots$	0	1
y_3	0	0	1	$\cdots$							$\cdots$	0	1
$\vdots$	$\vdots$			$\ddots$								$\vdots$	1
$\vdots$	$\vdots$						$\ddots$					$\vdots$	1
y_{2n+1}	0	$\cdots$							1	0	0		1
y_{2n+2}	0	$\cdots$							0	1	0		1
y_{2n+3}	0	$\cdots$							0	0	1		1

Figure 8: Construction of the profile J in the proof of Theorem 4.23.

We show that $\textsc{LexiMax}(J) = \{J^*\}$ with $J^* = \{ \neg y_j \mid 1 \leq j \leq 2n+3 \} \cup \{ \ell \mid 1 \leq i \leq n, \ell \in \{x_i, \neg x_i\}, \alpha(\ell) = 1 \}$, where α is the lexicographically maximal model of ψ. The formulas in Φ with highest support in J are the formulas $\neg y_j$—these are supported by $2n+2$ judgment sets in J.

Because $\Gamma_{\text{out}} \wedge \bigwedge_{1 \le j \le 2n+3} \neg y_j$ is satisfiable, we get that $\neg y_j \in J^*$ for each $1 \le j \le 2n + 3$. The formulas $x_1, \ldots, x_n$ have support $2n + 1, \ldots, n + 2$, respectively, and all other formulas have support less than $n + 2$. Therefore, J^* will agree with α on the formulas in $\{x_i, \neg x_i \mid 1 \le i \le n\}$. From this it follows that $x^* \in J^*$ if and only if w is set to true in α. $\qquad \square$

4.8 Outcome Determination for Rules based on Minimal Profile Modifications

We now turn our attention to showing Θ_2^{p}-membership for the outcome determination problem for the Young rule YNG.

Theorem 4.24. *The outcome determination problem for the Young rule* YNG *is in* Θ_2^{p}.

Proof. As described in Sections 2.1 and 2.2, we will use variant (6) of the framework and variant (a) of the problem to establish the upper bound on the complexity. The proof is similar to the proofs of Theorems 4.3, 4.5 and 4.7. We describe a polynomial-time algorithm that queries an NP oracle $O(\log |\boldsymbol{J}|)$ times and that solves the problem. The algorithm uses oracle queries to check if there exists a subprofile $\boldsymbol{J}' \subseteq \boldsymbol{J}$ containing at most a given number k of judgment sets such that $m(\boldsymbol{J}')$ is Γ_{out}-consistent. This is a problem in NP, so by picking an NP-complete problem for the oracle, we can use a single oracle query to solve any instance of the above problem in polynomial time. The maximum size $|\boldsymbol{J}'|$ of any such subprofile $\boldsymbol{J}'$ is $|\boldsymbol{J}|$. Therefore, we can determine the maximum value k_{max} of $|\boldsymbol{J}'|$ for any $\boldsymbol{J}' \subseteq \boldsymbol{J}$ such that $m(\boldsymbol{J}')$ is Γ_{out}-consistent using $O(\log |\boldsymbol{J}|)$ oracle queries—by using binary search. Then, after having identified this number k_{max}, the algorithm queries the oracle one additional time to determine whether there exists a subprofile $\boldsymbol{J}' \subseteq \boldsymbol{J}$ such that $|\boldsymbol{J}'| = k_{\text{max}}$, such that $m(\boldsymbol{J}') = J^*$ is Γ_{out}-consistent, such that $L \subseteq J^*$ and such that $L_i \not\subseteq J^*$ for all $i \in \{1, \ldots, u\}$. Again, since this is a problem in NP, one oracle query suffices to determine this. The algorithm outputs "yes" if and only if such a subprofile $\boldsymbol{J}' \subseteq \boldsymbol{J}$ exists. This algorithm runs in polynomial time, queries the NP oracle $O(\log |\boldsymbol{J}|)$ times, and correctly solves the problem. Therefore, the problem is in Θ_2^{p}. $\qquad \square$

Next, we prove Θ_2^{p}-hardness for the outcome determination problem for the Young rule YNG. To do this, one can use the standard embedding of voting into judgment aggregation (Dietrich & List, 2007a; Endriss, 2016). Performing the Young rule on the result of this embedding corresponds to carrying out the Young voting rule. As computing the winner of a Young election is Θ_2^{p}-hard (Rothe, Spakowski, & Vogel, 2003), this embedding can be used to show Θ_2^{p}-hardness of variant (a) of the outcome determination for the Young judgment aggregation rule. It is not straightforward to modify this reduction so that it works also for variant (b) of the outcome determination problem. We give an alternative proof of Θ_2^{p}-hardness that is self-contained, and that works for variant (b) of the outcome determination problem.[22]

Theorem 4.25. *The outcome determination problem for the Young rule* YNG *is* Θ_2^{p}-*hard.*

Proof. As described in Sections 2.1 and 2.2, we will use variant (1) of the framework and variant (b) of the problem to establish the lower bound on the complexity. We give a reduction from the Θ_2^{p}-complete problem MAX-MODEL. Let (ψ, x^*) be an instance of MAX-MODEL, with $\text{var}(\psi) = \{x_1, \ldots, x_n\}$. Moreover, without loss of generality, we may assume that the truth assignment that sets all variables to false does not satisfy ψ and that the truth assignment that sets all variables to true also does not

---

22. The proof of Theorem 4.25 is based on a proof appearing in preliminary work (Endriss & De Haan, 2015, Proposition 9).

satisfy ψ. We construct an instance $(\Phi, \Gamma_{\text{in}}, \Gamma_{\text{out}}, \boldsymbol{J}, \varphi^*)$ of the outcome determination problem for the Young rule $\textsc{Yng}$ as follows. We let $\Phi = \{\, x_i, \neg x_i \mid 1 \leq i \leq n \,\} \cup \{\, y_j, \neg y_j \mid 1 \leq j \leq 2n+1 \,\}$. We let $\Gamma_{\text{in}} = \Gamma_{\text{out}} = \psi \vee \bigvee_{1 \leq j \leq 2n+1} y_j$. We construct $\boldsymbol{J} = (J_1, \ldots, J_{2n+1})$ as described in Figure 9. Finally, we let $\varphi^* = x^*$.

$\boldsymbol{J}$	J_1	$\cdots$	J_{n+1}	J_{n+2}	J_{n+3}	J_{n+4}	$\cdots$	J_{2n-1}	J_{2n}	J_{2n+1}
x_1	1	$\cdots$	1	1	0	0	$\cdots$	0	0	0
x_2	1	$\cdots$	1	0	1	0	$\cdots$	0	0	0
x_3	1	$\cdots$	1	0	0	1	$\cdots$	0	0	0
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$	$\vdots$
x_{n-2}	1	$\cdots$	1	0	0	0	$\cdots$	1	0	0
x_{n-1}	1	$\cdots$	1	0	0	0	$\cdots$	0	1	0
x_n	1	$\cdots$	1	0	0	0	$\cdots$	0	0	1
y_1	1	0	0	$\cdots$					$\cdots$	0
y_2	0	1	0	$\cdots$					$\cdots$	0
y_3	0	0	1	$\cdots$					$\cdots$	0
$\vdots$	$\vdots$			$\ddots$						$\vdots$
$\vdots$	$\vdots$					$\ddots$				$\vdots$
y_{2n-1}	0	$\cdots$						1	0	0
y_{2n}	0	$\cdots$						0	1	0
y_{2n+1}	0	$\cdots$						0	0	1

Figure 9: Construction of the profile $\boldsymbol{J}$ in the proof of Theorem 4.25.

Intuitively, there are two ways to satisfy Γ_{out}: satisfy ψ or satisfy y_j for some $1 \leq j \leq 2n+1$. To get that $y_j \in m(\boldsymbol{J}')$ for some $1 \leq j \leq 2n+1$, one needs to delete $2n$ judgment sets from $\boldsymbol{J}$ to obtain $\boldsymbol{J}'$. We know that there exists a satisfying truth assignment α for ψ that sets at least one variable among $x_1, \ldots, x_n$ to true. Therefore, we can obtain some $\boldsymbol{J}'$ such that $m(\boldsymbol{J}')$ is Γ_{out}-consistent, by deleting at most $2n - 2$ judgment sets from $\boldsymbol{J}$: for each $1 \leq i \leq n$ such that $\alpha(x_i) = 0$, we delete J_i and J_{n+i+1}. Thus, we know that all judgment sets $J^* \in \textsc{Yng}(\boldsymbol{J})$ are consistent with ψ.

We show that there is some $J^* \in \textsc{Yng}(\boldsymbol{J})$ with $\varphi^* \in J^*$ if and only if there is a model of ψ that sets a maximal number of variables in $\text{var}(\psi)$ to true and that sets x^* to true.

($\Rightarrow$) Suppose that there is some $J^* \in \textsc{Yng}(\boldsymbol{J})$ with $\varphi^* \in J^*$. We know that J^* must be consistent with ψ. Consider the truth assignment $\alpha : \text{var}(\psi) \rightarrow \{0, 1\}$ that is defined by letting $\alpha(x_i) = 1$ if and only if $x_i \in J^*$, for each $1 \leq i \leq n$. Then α satisfies ψ and sets x^* to true. We show that there is no β that satisfies ψ and that sets more variables in $\text{var}(\psi)$ to true than α. Suppose, to derive a contradiction, that such a β does exist. Then consider the profile $\boldsymbol{J}'$ that is obtained from $\boldsymbol{J}$ by deleting J_i and J_{n+i+1} for each $1 \leq i \leq n$ such that $\beta(x_i) = 0$. Because β satisfies ψ and sets more variables in $\text{var}(\psi)$ to true than α, we get that $m(\boldsymbol{J}')$ is Γ_{out}-consistent and that $\boldsymbol{J}'$ contains more judgment sets than any subprofile $\boldsymbol{J}''$ of $\boldsymbol{J}$ for which $m(\boldsymbol{J}'') = J^*$. This is a contradiction with our assumption that $J^* \in \textsc{Yng}(\boldsymbol{J})$. Thus, we can conclude that no such β exists, and thus that α sets a maximal number of variables in $\text{var}(\psi)$ to true (for any truth assignment satisfying ψ).

($\Leftarrow$) Suppose that there is a satisfying truth assignment $\alpha : \text{var}(\psi) \rightarrow \{0, 1\}$ for ψ that sets a maximal number of variables in $\text{var}(\psi)$ to true and that sets x^* to true. Then consider the profile $\boldsymbol{J}'$ obtained from $\boldsymbol{J}$ by by deleting J_i and J_{n+i+1} for each $1 \leq i \leq n$ such that $\alpha(x_i) = 0$. Then $J^* = m(\boldsymbol{J}')$

is Γ_{out}-consistent, because it is consistent with ψ. Because α sets x^* to true, we get that $\varphi^* \in J^*$. We show that $J^* \in \text{YNG}(J)$. Suppose, to derive a contradiction, that there is a subprofile J'' of J that can be obtained by deleting a smaller number of judgment sets (than the number of judgment sets needed to obtain J') such that $m(J'')$ is Γ_{out}-consistent. Without loss of generality, we can take J'' to be the largest subprofile of J with this property. Since $m(J'')$ is Γ_{out}-consistent, we know that either (i) $m(J'')$ satisfies ψ, or (ii) $m(J'')$ satisfies y_j for some $1 \leq j \leq 2n + 1$. In case (ii), we know that at least $2n$ judgment sets need to be deleted from J to obtain J''. In case (i), we know that at most $2n - 2$ judgment sets need to be deleted from J to obtain J''. Therefore, since J'' is the largest subprofile of J such that $m(J'')$ is Γ_{out}-consistent, we know that $m(J'')$ is consistent with ψ.

Consider the truth assignment $\beta : \text{var}(\psi) \to \{0, 1\}$ that is defined by letting $\beta(x_i) = 1$ if and only if $x_i \in m(J'')$, for each $1 \leq i \leq n$. Because J'' is obtained from J by deleting fewer judgment sets than the number of sets that are deleted to obtain J', we know that β sets more variables in $\text{var}(\psi)$ to true than α. Also, because J'' is consistent with ψ, we know that β satisfies ψ. This is a contradiction with our assumption that α sets a maximal number of variables in $\text{var}(\psi)$ to true (for any satisfying assignment of ψ). Thus, we can conclude that no such J'' exists, and thus that $J^* = m(J') \in \text{YNG}(J)$. $\qquad \square$

Next, we establish Θ_2^p-membership for the outcome determination problem for the Dodgson rule DOD.

Theorem 4.26. *The outcome determination problem for the Dodgson rule* DOD *is in* Θ_2^p.

Proof. As described in Sections 2.1 and 2.2, we will use variant (6) of the framework and variant (a) of the problem to establish the upper bound on the complexity. The proof is similar to the proofs of Theorems 4.3, 4.5, 4.7, and 4.24. We describe a polynomial-time algorithm that queries an NP oracle $O(\log |\Phi| + \log |J|)$ times and that solves the problem. The algorithm uses oracle queries to check if there exists a profile $J' = (J_1', \ldots, J_n') \in \mathcal{J}(\Phi, \Gamma_{in})^n$, where $n = |J|$, such that $m(J')$ is Γ_{out}-consistent and such that $\sum_{i \leq |J|} H(J_i, J_i') \leq k$ for a given value k. This is a problem in NP, so by picking an NP-complete problem for the oracle, we can use a single oracle query to solve any instance of the above problem in polynomial time. The maximum value of $\sum_{i \leq |J|} H(J_i, J_i')$ for any profile J' of size $|J|$ is at most $|\Phi| \cdot |J|$. Therefore, we can determine the minimum value k_{min} of $\sum_{i \leq |J|} H(J_i, J_i')$ for any $J' = (J_1', \ldots, J_n')$ such that $m(J')$ is Γ_{out}-consistent using $O(\log |\Phi| + \log |J|)$ oracle queries— by using binary search. Then, after having identified this number k_{min}, the algorithm queries the oracle one additional time to determine whether there exists a profile $J' = (J_1', \ldots, J_n')$ such that $\sum_{i \leq |J|} H(J_i, J_i') = k_{min}$, such that $m(J') = J^*$ is Γ_{out}-consistent, such that $L \subseteq J^*$ and such that $L_i \nsubseteq J^*$ for all $i \in \{1, \ldots, u\}$. Again, since this is a problem in NP, one oracle query suffices to determine this. The algorithm outputs "yes" if and only if such a profile J' exists. This algorithm runs in polynomial time, queries the NP oracle $O(\log |\Phi| + \log |J|)$ times, and correctly solves the problem. Therefore, the problem is in Θ_2^p. $\qquad \square$

Finally, we modify the proof of Theorem 4.4 to obtain a proof of Θ_2^p-hardness for the outcome determination problem for the Dodgson rule.[23]

Theorem 4.27. *The outcome determination problem for the Dodgson rule* DOD *is* Θ_2^p*-hard.*

---

23. The proof of Theorem 4.27 is not based on the original hardness proof for the outcome determination problem for the Dodgson rule that appeared in preliminary work (Lang & Slavkovik, 2014, Proposition 5). This is because the original hardness proof does not apply to variant (1) of the framework.

Proof. As described in Sections 2.1 and 2.2, we will use variant (1) of the framework and variant (b) of the problem to establish the lower bound on the complexity. We give a reduction from the Θ_2^p-complete problem MAX-MODEL. We will use the reduction used in the proof of Theorem 4.4. Let (ψ, x^*) be an instance of MAX-MODEL, with $\mathsf{var}(\psi) = \{x_1, \ldots, x_n\}$. Without loss of generality, suppose that $x^* = x_1$. Moreover, without loss of generality, suppose that n is odd. We construct an instance $(\Phi, \Gamma_{\mathrm{in}}, \Gamma_{\mathrm{out}}, \boldsymbol{J}, \varphi^*)$ of the outcome determination problem for the reversal scoring rule by following the construction in the proof of Theorem 4.4.

To see that this reduction is correct, we observe that judgment sets J^* satisfying ψ (and agreeing with $m(\boldsymbol{J})$ on $\bigwedge_{1 \le i \le n, 1 \le j \le u} \neg z_{i,j}$) can be obtained as majoritarian judgment set $m(\boldsymbol{J}')$ by changing the profile on $(n - b)(n - 1)/2$ issues, where b is the maximal number of variables in $\mathsf{var}(\psi)$ that are set to true by any truth assignment satisfying ψ. To obtain a profile $\boldsymbol{J}'$ such that $m(\boldsymbol{J}')$ satisfies $\bigwedge_{1 \le j \le u} z_{i,j}$ for some $1 \le i \le n$, one needs to make $u(n - 1)/2$ on individual issues. Since $u > n - b$, we know that $\textsc{Dod}(\boldsymbol{J})$ are in one-to-one correspondence with the set of truth assignments satisfying ψ that set a maximal number of variables in $\mathsf{var}(\psi)$ to true. Therefore, we get that there is some $J^* \in \textsc{Dod}(\boldsymbol{J})$ with $x_1 = \varphi^* \in J^*$ if and only there is a satisfying truth assignment for ψ that satisfies x_1 and that sets a maximal number in $\mathsf{var}(\psi)$ to true. $\qquad\square$

5. Conclusion

We have provided a complete picture of the computational complexity of the outcome determination problem for all eleven judgment aggregation rules that (i) have received significant attention in the literature, (ii) are applicable for any agenda, and (iii) guarantee the consistency of outcomes. For each of these rules the outcome determination problem is complete for either Θ_2^p, Δ_2^p, Σ_2^p, or Θ_3^p. Our results hold for all important judgment aggregation frameworks considered in the literature.

For several of the judgment aggregation rules that have a clear counterpart in preference aggregation, our results are in line with complexity results for computing aggregated preferences (or computing the top-ranked candidates in the aggregated preferences). For example, aggregating preferences using the Kemeny, Slater, Young, or Dodgson rule is Θ_2^p-complete (see, e.g., Hemaspaandra et al., 2005; Conitzer, 2006; Rothe et al., 2003; Hemaspaandra et al., 1997)—which mirrors our Θ_2^p-completeness results for the judgment aggregation versions of these rules. Similarly, the MaxHamming judgment aggregation rule has parallels to the minimax approval procedure for committee elections (Brams et al., 2007), and our Θ_2^p-completeness result that for the MaxHamming judgment aggregation rule is in line with complexity results for the minimax approval rule (LeGrand, Markakis, & Mehta, 2007; De Haan, 2018). But for other preference aggregation rules, the complexity of computing the outcome of the analogous judgment aggregation rule has a higher computational complexity. For example, the ranked-pairs voting rule is polynomial-time computable (see, e.g., Fischer et al., 2016), which is not in line with our Σ_2^p-completeness result for the analogous ranked-agenda judgment aggregation rule.

Our results provide useful insights for the development of practical algorithmic approaches to computing the outcome in a given judgment aggregation scenario. For example, the complexity results we established in this paper indicate that we cannot avoid exponential running times in the worst case when developing algorithms that work for the fully general setting of judgment aggregation—at least not under the widely believed assumption that $\mathsf{P} \ne \mathsf{NP}$. Our results also indicate that automated reasoning tools aimed at higher levels of the Polynomial Hierarchy might provide a fruitful approach for developing practical algorithms. Research along these lines has

been initiated very recently, encoding the outcome determination problem for various judgment aggregation rules into the automated reasoning framework of *Answer Set Programming* (De Haan & Slavkovik, 2019). Our results indicate the amount of expressiveness that is needed in such automated reasoning languages for computing the outcome of judgment aggregation scenarios. For example, Θ_3^p-completeness indicates that the full expressivity of disjunctive answer set programming with optimisation is required, whereas Θ_2^p-completeness indicates that (non-disjunctive) answer set programming with optimisation suffices (see, e.g., Brewka & Eiter, 1999; Buccafurri, Leone, & Rullo, 2000). More generally, for judgment aggregation rules for which the outcome determination problem is Θ_2^p-complete or Δ_2^p-complete, one can use optimisation or so-called incremental variants of solving algorithms for NP problems—e.g., propositional satisfiability (SAT), integer programming (IP), constraint programming (CP), satisfiability-modulo-theories (SMT)—whereas for rules for which outcome determination is Σ_2^p-complete or Θ_3^p-complete, one can only encode the problem into harder problems (in polynomial time)—e.g., disjunctive answer set programming, or quantified Boolean satisfiability (QBF-SAT).

Future research should include studying the computational complexity of the outcome determination problem for different judgment aggregation rules in more detail. The results in this paper are worst-case results for the setting where the propositional logic formulas involved can be any arbitrary formulas. In many cases where the framework of judgment aggregation is applied to model a particular aggregation scenario, the logic formulas involved (and other parameters of the scenario) are likely to exhibit some structure. Therefore, it will be useful to study what the exact contribution of these problem parameters is for the computational complexity of the problem. Such an investigation has been initiated in recent work for a handful of judgment aggregation rules (De Haan, 2016, 2018)—both using the classical theory of computational complexity and the more fine-grained theory of parameterized complexity—but many questions remain.

Acknowledgments

References

Arora, S., & Barak, B. (2009). *Computational Complexity – A Modern Approach*. Cambridge University Press, Cambridge.

Bachmaier, C., Brandenburg, F. J., Gleißner, A., & Hofmeier, A. (2015). On the hardness of maximum rank aggregation problems. *Journal of Discrete Algorithms*, *31*, 2–13.

Baral, C., Kraus, S., Minker, J., & Subrahmanian, V. S. (1992). Combining knowledge bases consisting of first-order theories. *Computational Intelligence*, *8*(1), 45–71.

Baumeister, D., Erdélyi, G., Erdélyi, O. J., & Rothe, J. (2015). Complexity of manipulation and bribery in judgment aggregation for uniform premise-based quota rules. *Mathematical Social Sciences*, *76*, 19–30.

Baumeister, D., Erdélyi, G., Erdélyi, O. J., Rothe, J., & Selker, A.-K. (2020). Complexity of control in judgment aggregation for uniform premise-based quota rules. *Journal of Computer and System Sciences*, *112*, 13–33.

Borda, J.-C. d. (1781). *Mémoire sur les Élections au Scrutin*. Histoire de l'Académie Royale des Sciences, Paris.

Brams, S. J., Kilgour, D. M., & Sanver, M. R. (2007). A minimax procedure for electing committees. *Public Choice*, *132*(3–4), 401–420.

Brandt, F., Brill, M., & Harrenstein, P. (2016a). Tournament solutions. In Brandt, F., Conitzer, V., Endriss, U., Lang, J., & Procaccia, A. D. (Eds.), *Handbook of Computational Social Choice*, chap. 3, pp. 57–84. Cambridge University Press.

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

Brewka, G., & Eiter, T. (1999). Preferred answer sets for extended logic programs. *Artificial intelligence*, *109*(1–2), 297–356.

Buccafurri, F., Leone, N., & Rullo, P. (2000). Enhancing disjunctive datalog by constraints. *IEEE Transactions on Knowledge and Data Engineering*, *12*(5), 845–860.

Buss, S. R., & Hay, L. (1991). On truth-table reducibility to SAT. *Information and Computation*, *91*(1), 86–102.

Caragiannis, I., Hemaspaandra, E., & Hemaspaandra, L. A. (2016). Dodgson's rule and Young's rule. In Brandt, F., Conitzer, V., Endriss, U., Lang, J., & Procaccia, A. D. (Eds.), *Handbook of Computational Social Choice*, chap. 5, pp. 103–126. Cambridge University Press.

Chapman, B. (1998). More easily done than said: Rules, reasons and rational social choice. *Oxford Journal of Legal Studies*, *18*(2), 293–329.

Chen, Z.-Z., & Toda, S. (1995). The complexity of selecting maximal solutions. *Information and Computation*, *119*(2), 231–239.

Conitzer, V. (2006). Computing Slater rankings using similarities among candidates. In *Proceedings of the 21st AAAI Conference on Artificial Intelligence (AAAI-2006)*, pp. 613–619. AAAI Press.

van Dalen, D. (2013). *Logic and Structure* (5th edition). Springer.

Dietrich, F. (2014). Scoring rules for judgment aggregation. *Social Choice and Welfare*, *42*(4), 873–911.

Dietrich, F., & List, C. (2007a). Arrow's theorem in judgment aggregation. *Social Choice and Welfare*, *29*(1), 19–33.

Dietrich, F., & List, C. (2007b). Judgment aggregation by quota rules: Majority voting generalized. *Journal of Theoretical Politics*, *19*(4), 391–424.

Dietrich, F., & List, C. (2008). Judgment aggregation without full rationality. *Social Choice and Welfare*, *31*(1), 15–39.

Dietrich, F., & Mongin, P. (2010). The premiss-based approach to judgment aggregation. *Journal of Economic Theory*, *145*(2), 562–582.

Dodgson, C. (1876). A method of taking votes on more than two issues. Reprinted in McLean and Urken (1995), chap. 12, pp. 288–297.

Dokow, E., & Holzman, R. (2010). Aggregation of binary evaluations. *Journal of Economic Theory*, *145*(2), 495–511.

Duddy, C., & Piggins, A. (2012). A measure of distance between judgment sets. *Social Choice and Welfare*, *39*, 855–867.

Endriss, U. (2016). Judgment aggregation. In Brandt, F., Conitzer, V., Endriss, U., Lang, J., & Procaccia, A. D. (Eds.), *Handbook of Computational Social Choice*, chap. 17, pp. 399–426. Cambridge University Press.

Endriss, U., Grandi, U., & Porello, D. (2012). Complexity of judgment aggregation. *Journal of Artificial Intelligence Research*, *45*, 481–514.

Endriss, U. (2018). Judgment aggregation with rationality and feasibility constraints. In *Proceedings of the 17th International Conference on Autonomous Agents and Multiagent Systems (AAMAS-2018)*, pp. 946–954. IFAAMAS.

Endriss, U., & Grandi, U. (2014). Binary aggregation by selection of the most representative voter. In *Proceedings of the 28th AAAI Conference on Artificial Intelligence (AAAI-2014)*, pp. 668–674. AAAI Press.

Endriss, U., Grandi, U., de Haan, R., & Lang, J. (2016). Succinctness of languages for judgment aggregation. In *Proceedings of the 15th International Conference on the Principles of Knowledge Representation and Reasoning (KR-2016)*, pp. 176–186. AAAI Press.

Endriss, U., & de Haan, R. (2015). Complexity of the winner determination problem in judgment aggregation: Kemeny, Slater, Tideman, Young. In *Proceedings of the 14th International Conference on Autonomous Agents and Multiagent Systems (AAMAS-2015)*, pp. 117–125. IFAAMAS.

Everaere, P., Konieczny, S., & Marquis, P. (2014). Counting votes for aggregating judgments. In *Proceedings of the 13th International Conference on Autonomous Agents and Multiagent Systems (AAMAS-2014)*, pp. 1177–1184. IFAAMAS.

Everaere, P., Konieczny, S., & Marquis, P. (2017). An introduction to belief merging and its links with judgment aggregation. In Endriss, U. (Ed.), *Trends in Computational Social Choice*, chap. 7, pp. 123–143. AI Access.

Fischer, F., Hudry, O., & Niedermeier, R. (2016). Weighted tournament solutions. In Brandt, F., Conitzer, V., Endriss, U., Lang, J., & Procaccia, A. D. (Eds.), *Handbook of Computational Social Choice*, chap. 4, pp. 85–102. Cambridge University Press.

Gärdenfors, P. (2006). A representation theorem for voting with logical consequences. *Economics and Philosophy*, *22*(2), 181–190.

Grandi, U. (2012). *Binary Aggregation with Integrity Constraints*. Ph.D. thesis, University of Amsterdam.

Grandi, U., & Endriss, U. (2013). Lifting integrity constraints in binary aggregation. *Artificial Intelligence*, *199–200*, 45–66.

Grossi, D., & Pigozzi, G. (2014). *Judgment Aggregation: A Primer*. Morgan and Claypool Publishers.

de Haan, R. (2016). Parameterized complexity results for the Kemeny rule in judgment aggregation. In *Proceedings of the 22nd European Conference on Artificial Intelligence, ECAI-2016*, pp. 1502–1510. IOS Press.

de Haan, R. (2018). Hunting for tractable languages for judgment aggregation. In *Proceedings of the 16th International Conference on Principles of Knowledge Representation and Reasoning (KR-2018)*, pp. 194–203. AAAI Press.

de Haan, R., & Slavkovik, M. (2017). Complexity results for aggregating judgments using scoring or distance-based procedures. In *Proceedings of the 16th International Conference on Autonomous Agents and Multiagent Systems (AAMAS-2017)*, pp. 952–961. IFAAMAS.

de Haan, R., & Slavkovik, M. (2019). Answer set programming for judgment aggregation. In *Proceedings of the 28th International Joint Conference on Artificial Intelligence (IJCAI-2019)*, pp. 1668–1674. IJCAI.

Hartmann, S., & Sprenger, J. (2012). Judgment aggregation and the problem of tracking the truth. *Synthese*, *187*(1), 209–221.

Hemachandra, L. A. (1989). The strong exponential hierarchy collapses. *Journal of Computer and System Sciences*, *39*(3), 299–322.

Hemaspaandra, E., Hemaspaandra, L. A., & Rothe, J. (1997). Exact analysis of Dodgson elections: Lewis Carroll's 1876 system is complete for parallel access to NP. *Journal of the ACM*, *44*(6), 806–825.

Hemaspaandra, E., Hemaspaandra, L. A., & Menton, C. (2020). Search versus decision for election manipulation problems. *ACM Transactions on Computation Theory*, *12*(1).

Hemaspaandra, E., Spakowski, H., & Vogel, J. (2005). The complexity of Kemeny elections. *Theoretical Computer Science*, *349*(3), 382–391.

Kemeny, J. G. (1959). Mathematics without numbers. *Daedalus*, *88*(4), 577–591.

Kleine Büning, H., & Bubeck, U. (2009). Theory of quantified boolean formulas. In Biere, A., Heule, M., van Maaren, H., & Walsh, T. (Eds.), *Handbook of Satisfiability*, chap. 23, pp. 735–760. IOS Press.

Köbler, J., Schöning, U., & Wagner, K. W. (1987). The difference and truth-table hierarchies for NP. *RAIRO Theoretical Informatics and Applications*, *21*(4), 419–435.

Konieczny, S., & Pino-Pérez, R. (2002). Merging information under constraints: A logical framework. *Journal of Logic and Computation*, *12*(5), 773–808.

Konieczny, S., Lang, J., & Marquis, P. (2004). DA^2 merging operators. *Artificial Intelligence*, *157*(1–2), 49–79.

Kornhauser, L. A., & Sager, L. G. (1993). The one and the many: Adjudication in collegial courts. *California Law Review*, *81*(1), 1–59.

Krentel, M. W. (1988). The complexity of optimization problems. *Journal of Computer and System Sciences*, *36*(3), 490–509.

Krentel, M. W. (1992). Generalizations of OptP to the Polynomial Hierarchy. *Theoretical Computer Science*, *97*(2), 183–198.

Lang, J., Pigozzi, G., Slavkovik, M., & van der Torre, L. (2011). Judgment aggregation rules based on minimization. In *Proceedings of the 13th Conference on Theoretical Aspects of Rationality and Knowledge (TARK-2011)*, pp. 238–246. Institute of Mathematical Sciences, Chennai.

Lang, J., Pigozzi, G., Slavkovik, M., van der Torre, L., & Vesic, S. (2017). A partial taxonomy of judgment aggregation rules and their properties. *Social Choice and Welfare*, *48*(2), 327–356.

Lang, J., & Slavkovik, M. (2014). How hard is it to compute majority-preserving judgment aggregation rules? In *Proceedings of the 21st European Conference on Artificial Intelligence (ECAI-2014)*, pp. 501–506. IOS Press.

LeGrand, R., Markakis, E., & Mehta, A. (2007). Some results on approximating the minimax solution in approval voting.. In *Proceedings of the 6th International Conference on Autonomous Agents and Multiagent Systems (AAMAS-2007)*, pp. 1185–1187. IFAAMAS.

Li, P., & Milenkovic, O. (2017). Multiclass MinMax rank aggregation. In *Proceedings of the 2017 IEEE International Symposium on Information Theory (ISIT)*, pp. 3000–3004. IEEE Computer Society.

Liberatore, P., & Schaerf, M. (1998). Arbitration (or how to merge knowledge bases). *IEEE Transactions on Knowledge and Data Engineering*, *10*(1), 76–90.

List, C. (2012). The theory of judgment aggregation: An introductory review. *Synthese*, *187*(1), 179–207.

List, C., & Pettit, P. (2002). Aggregating sets of judgments: An impossibility result. *Economics and Philosophy*, *18*(1), 89–110.

McLean, I., & Urken, A. (1995). *Classics of Social Choice*. University of Michigan Press.

Meyer, A. R., & Stockmeyer, L. J. (1972). The equivalence problem for regular expressions with squaring requires exponential space. In *Proceedings of the 13th Annual Symposium on Switching and Automata Theory (SWAT-1972)*, pp. 125–129. IEEE Computer Society.

Miller, M. K., & Osherson, D. (2009). Methods for distance-based judgment aggregation. *Social Choice and Welfare*, *32*(4), 575–601.

Nehring, K., & Pivato, M. (2013). Majority rule in the absence of a majority. MPRA Paper 46721, Munich Personal RePEc Archive.

Nehring, K., Pivato, M., & Puppe, C. (2014). The Condorcet set: Majority voting over interconnected propositions. *Journal of Economic Theory*, *151*, 268–303.

Nehring, K., & Puppe, C. (2007). The structure of strategy-proof social choice. Part I: General characterization and possibility results on median spaces. *Journal of Economic Theory*, *135*(1), 269–305.

Papadimitriou, C. H., & Zachos, S. K. (1982). Two remarks on the power of counting. In Cremers, A. B., & Kriegel, H.-P. (Eds.), *Theoretical Computer Science*, Vol. 145 of *Lecture Notes in Computer Science*, pp. 269–275. Springer.

Pauly, M., & van Hees, M. (2006). Logical constraints on judgement aggregation. *Journal of Philosophical Logic*, *35*(6), 569–585.

Pettit, P. (2001). Deliberative democracy and the discursive dilemma. *Philosophical Issues*, *11*(1), 268–299.

Pigozzi, G. (2006). Belief merging and the discursive dilemma: An argument-based account to paradoxes of judgment aggregation. *Synthese*, *152*(2), 285–298.

Porello, D., & Endriss, U. (2014). Ontology merging as social choice: Judgment aggregation under the open world assumption. *Journal of Logic and Computation*, *24*(6), 1229–1249.

Rothe, J., Spakowski, H., & Vogel, J. (2003). Exact complexity of the winner problem for Young elections. *Theory of Computing Systems*, *36*(4), 375–386.

Sandholm, T. (2002). Algorithm for optimal winner determination in combinatorial auctions. *Artificial Intelligence*, *135*(1-2), 1–54.

Slater, P. (1961). Inconsistencies in a schedule of paired comparisons. *Biometrika*, *48*(3–4), 303–312.

Slavkovik, M., & Ågotnes, T. (2014). Measuring dissimilarity between judgment sets. In *Proceedings of the 14th European Conference on Logics in Artificial Intelligence (JELIA-2014)*, pp. 609–617. Springer.

Stockmeyer, L. J. (1976). The polynomial-time hierarchy. *Theoretical Computer Science*, *3*(1), 1–22.

Terzopoulou, Z., & Endriss, U. (2018). Modelling iterative judgment aggregation. In *Proceedings of the 32nd AAAI Conference on Artificial Intelligence (AAAI-2018)*, pp. 1234–1241. AAAI Press.

Tideman, T. N. (1987). Independence of clones as a criterion for voting rules. *Social Choice and Welfare*, *4*(3), 185–206.

Wagner, K. W. (1990). Bounded query classes. *SIAM Journal of Computing*, *19*(5), 833–846.

Wrathall, C. (1976). Complete sets and the polynomial-time hierarchy. *Theoretical Computer Science*, *3*(1), 23–33.

Young, H. P. (1977). Extending Condorcet's rule. *Journal of Economic Theory*, *16*(2), 335–353.

Zwicker, W. S. (2016). Introduction to the theory of voting. In Brandt, F., Conitzer, V., Endriss, U., Lang, J., & Procaccia, A. D. (Eds.), *Handbook of Computational Social Choice*, chap. 2, pp. 23–56. Cambridge University Press.