Université Fédérale

Toulouse Midi-Pyrénées

THÈSE

En vue de l'obtention du

DOCTORAT DE L'UNIVERSITÉ DE TOULOUSE

Délivré par :

Université Toulouse 3 Paul Sabatier (UT3 Paul Sabatier)

Présentée et soutenue par :
Arianna Novaro

le 12/11/2019

Titre :

Collective Decision-Making with Goals

École doctorale et discipline ou spécialité :
ED MITT : Domaine STIC : Intelligence Artificielle

Unité de recherche :
Institut de Recherche en Informatique de Toulouse - Logique, Interaction, Langue et Calcul

Directeur/trice(s) de Thèse :

Emiliano Lorini (Directeur), Umberto Grandi (Co-Directeur), Dominique Longin (Encadrant)

Jury :
Leila Amgoud (Présidente)
Edith Elkind (Examinatrice)
Umberto Grandi (Co-Directeur)
Emiliano Lorini (Directeur)
Nicolas Maudet (Rapporteur)
Gabriella Pigozzi (Examinatrice)
Michael Wooldridge (Rapporteur)

Abstract

Agents having to take a collective decision are often motivated by individual goals. In such scenarios, two key aspects need to be addressed. The first is defining how to select a winning alternative from the expressions of the agents. The second is making sure that agents will not manipulate the outcome. Agents should also be able to state their goals in a way that is expressive, yet not too burdensome. This dissertation studies the aggregation and the strategic component of multi-agent collective decisions where the agents use a compactly represented language. The languages we study are all related to logic: from propositional logic, to generalized CP-nets and linear temporal logic (LTL).

Our main contribution is the introduction of the framework of goal-based voting, where agents submit individual goals expressed as formulas of propositional logic. Classical aggregation functions from voting, judgment aggregation, and belief merging are adapted to this setting and studied axiomatically and computationally. Desirable axiomatic properties known in the literature of social choice theory are generalized to this new type of propositional input, as well as the standard complexity problems aimed at determining the result.

Another important contribution is the study of the aggregation of generalized CP-nets coming from multiple agents, i.e., CP-nets where the precondition of the preference statement is a propositional formula. We use different aggregators to obtain a collective ordering of the possible outcomes. Thanks to this thesis, two lines of research are thus bridged: the one on the aggregation of complete CP-nets, and the one on the generalization of CP-nets to incomplete preconditions. We also contribute to the study of strategic behavior in both collective decision-making and game-theoretic settings. The framework of goal-based voting is studied again under the assumption that agents can now decide to submit an untruthful goal if by doing so they can get a better outcome. The focus is on three majoritarian voting rules which are found to be manipulable. Therefore, we study restrictions on both the language of the goals and on the strategies allowed to the agents to discover islands of strategy-proofness.

We also present a game-theoretic extension of a recent model of opinion diffusion over networks of influence. In the influence games defined here, agents hold goals expressed as formulas of LTL and they can choose whether to use their influence power to make sure that their goal is satisfied. Classical solution concepts such as weak dominance and winning strategy are studied for influence games, in relation to the structure of the network and the goals of the agents. Finally, we introduce a novel class of concurrent game structures (CGS) in which agents can have shared control over a set of propositional variables. Such structures are used for the interpretation of formulas of alternating-time temporal logic, thanks to which we can express the existence of a winning strategy for an agent in a repeated game (as, for instance, the influence games mentioned above). The main result shows by means of a clever construction that a CGS with shared control can be represented as a CGS with exclusive control.

In conclusion, this thesis provides a valuable contribution to the field of collective decision-making by introducing a novel framework of voting based on individual propositional goals, it studies for the first time the aggregation of generalized CP-nets, it extends a framework of opinion diffusion by modelling rational agents who use their influence power as they see fit, and it provides a reduction of shared to exclusive control in CGS for the interpretation of logics of strategic reasoning. By using different logical languages, agents can thus express their goals and preferences over the decision to be taken, and desirable properties of the decision process can be ensured.

Résumé

Des agents devant prendre une décision collective sont souvent motivés par des buts individuels. Dans ces situations, deux aspects clés doivent être abordés : sélectionner une alternative gagnante à partir des voix des agents et s'assurer que les agents ne manipulent pas le résultat. Cette thèse étudie l'agrégation et la dimension stratégique des décisions collectives lorsque les agents utilisent un langage représenté de manière compacte. Nous étudions des langages de type logique : de la logique propositionnelle aux CP-nets généralisés, en passant par la logique temporelle linéaire (LTL).

Notre principale contribution est l'introduction d'un cadre de vote sur les buts, dans lequel les agents soumettent des buts individuels exprimés comme des formules de la logique propositionnelle. Les fonctions d'agrégation classiques issues du vote, de l'agrégation de jugements et de la fusion de croyances sont adaptées et étudiées de manière axiomatique et computationnelle. Les propriétés axiomatiques connues dans la littérature sur la théorie du choix social sont généralisées à ce nouveau type d'entrée, ainsi que les problèmes de complexité visant à déterminer le résultat du vote.

Une autre contribution importante est l'étude de l'agrégation des CP-nets généralisés, c'est-à-dire des CP-nets où la précondition de l'énoncé de préférence est une formule propositionnelle. Nous utilisons différents agrégateurs pour obtenir un classement collectif des résultats possibles. Grâce à cette thèse, deux axes de recherche sont ainsi reliés : l'agrégation des CP-nets classiques et la généralisation des CP-nets à des préconditions incomplètes.

Nous contribuons également à l'étude du comportement stratégique dans des contextes de prise de décision collective et de théorie des jeux. Le cadre du vote basé sur les buts est de nouveau étudié sous l'hypothèse que les agents peuvent décider de mentir sur leur but s'ils obtiennent ainsi un meilleur résultat. L'accent est mis sur trois règles de vote majoritaires qui se révèlent manipulables. Par conséquent, nous étudions des restrictions à la fois sur le langage des buts et sur les stratégies des agents en vue d'obtenir des résultats de votes non manipulables.

Nous présentons par ailleurs une extension stratégique d'un modèle récent de diffusion d'opinion sur des réseaux d'influence. Dans les jeux d'influence définis ici, les agents ont comme but des formules en LTL et ils peuvent choisir d'utiliser leur pouvoir d'influence pour s'assurer que leur but est atteint. Des solutions classiques telles que la stratégie gagnante sont étudiées pour les jeux d'influence, en relation avec la structure du réseau et les buts des agents.

Enfin, nous introduisons une nouvelle classe de concurrent game structures (CGS) dans laquelle les agents peuvent avoir un contrôle partagé sur un ensemble de variables propositionnelles. De telles structures sont utilisées pour interpréter des formules de logique temporelle en temps alternés (ATL), grâce auxquelles on peut exprimer l'existence d'une stratégie gagnante pour un agent dans un jeu itéré (comme les jeux d'influence mentionnés ci-dessus). Le résultat principal montre qu'un CGS avec contrôle partagé peut être représenté comme un CGS avec contrôle exclusif.

En conclusion, cette thèse contribue au domaine de la prise de décision collective en introduisant un nouveau cadre de vote basé sur des buts propositionnels. Elle présente une étude de l'agrégation des CP-nets généralisés et une extension d'un cadre de diffusion d'opinion avec des agents rationnels qui utilisent leur pouvoir d'influence. Une réduction du contrôle partagé à un contrôle exclusif dans les CGS pour l'interprétation des logiques du raisonnement stratégique est également proposée. Par le biais de langages logiques divers, les agents peuvent ainsi exprimer buts et préférences sur la décision à prendre, et les propriétés souhaitées pour le processus de décision peuvent en être garanties.

Acknowledgments

The dissertation that you are about to read would not exist without the support and encouragement that I have received during these three years of PhD fellowship. I was extremely lucky to be surrounded by great people, and I will do my best to do them justice by writing this acknowledgments page.

First and foremost I would like to thank all the committee members for having accepted to being part of the jury for my PhD defence: Leila Amgoud, Edith Elkind, Umberto Grandi, Emiliano Lorini, Nicolas Maudet, Gabriella Pigozzi and Michael Wooldridge. In particular, I want to thank the *rapporteurs* Nicolas Maudet and Michael Wooldridge for their time in carefully reading this thesis and writing detailed *rapports* filled with valuable feedback.

It is a recurring theme for PhD students to perennially complain about their supervisors: my main complaint is that I could never join these conversations, as I had the misfortune of receiving an excellent supervision. Thank you Dominique Longin for showing me how to think like a computer scientist, for your positive attitude and for offering useful comments and critiques to improve the work presented here. I also want to thank Emiliano Lorini for all the knowledge you shared about philosophical and game-theoretical questions, as well as your kindness for giving some final suggestions in the last steps of writing this manuscript during summer holidays. Last, but certainly not least, I want to express my gratitude to Umberto Grandi. Thank you for helping me improve my mathematical reasoning skills, for supporting and believing in my ideas when I felt discouraged, for your contagious enthusiasm and for your copious scientific, career and life advice. Not to mention, of course, for all the cups of real coffee you offered me during these years.

During my PhD years I had the amazing opportunity to travel abroad for conferences and workshops, which would not have been possible without the support of the administrative staff at the University of Toulouse. I want to thank Sébastien Konieczny and Pierre Marquis for hosting me for a short visiting at the Centre de Recherche en Informatique de Lens, and for the feedback they and Patricia Everaere gave me about preliminary versions of part of this work. I also want to thank Ulle Endriss for hosting me at the University of Amsterdam, where I could discuss many exciting ideas with him and his group, as well as for being the person who made me discover the field of computational social choice in the first place.

I want to thank the members of my research team at the Institut de Recherche en Informatique de Toulouse, and more broadly all my colleagues at the University of Toulouse 1 Capitole, for making me passionate about teaching. Thank you also to all my fellow PhD students, past and present, who shared an indefinite amount of hours with me in the mythical ME310, where so many discussions, laughs, cries (and, occasionally, some work) happened.

Finally, I want to thank all of my friends, those living in Toulouse or scattered around the world, former colleagues or those I met outside academia, those I have known since a couple years or dating back to elementary school, for their incredible support. Thanks also to my wonderful family, for their unconditional love and for always being there for me. I would not have achieved all of this without you.

(A meta-thank you to Rachael Colley for proofreading these acknowledgments).

Table of Contents

Chapter 1

Introduction

On a sunny morning, Lucy is having a coffee with her colleagues Bob and Ann at the Department of Computer Science. "Did you check your e-mails?" asks Bob. "Our bid for hosting the next edition of the Important Conference in Artificial Intelligence has been accepted!"

They are all thrilled about the perspective of holding such a prestigious event in their city. "That is great news!" says Lucy. "Do you think we should host it in the historical buildings in the center, or here in the new campus?"

"I live in between the center and here", replies Ann. "I'm indifferent about that. But I would like to avoid a poster session".

"If we have a poster session it's better to use the premises in the center" comments Bob, finishing his espresso. "We did it years ago and it worked quite smoothly!"

Lucy disagrees with Ann on the posters: she always learns a lot during these sessions by asking questions one-on-one to the researchers. She will try later to change Ann's mind. After all, Ann recognizes that Lucy has more experience in organizing conferences and she usually ends up listening to her on the topic.

"We could also split our tasks" suggests Bob, returning their cups to the counter. "For instance, I could build the website, Lucy could think about location, and Ann could book the gala dinner. But we all decide about the scientific issues together, of course".

While the three are waiting for the elevator to go back to their offices, Ann offers a sheet of paper to Lucy. "I was writing the syllabus for our course yesterday. In general, I prefer to listen to students' presentations rather than correcting weekly assignments, but if we are having a final oral exam it is better the other way around".

"Yes, I agree. And I also prefer the assignments to the presentations if we use the second textbook in the list" says Lucy, looking at the syllabus. "It already has some exercises we could use, and solutions are not available online".

The door of the elevator closes behind the researchers.

In the corridor, now empty, the artificial lights slowly start to dim, until they switch off completely.

A great variety of situations occurring every day resemble the ones in the short story above: a group of agents (i.e., Lucy, Bob, and Ann — but also the sensors regulating the lights in the corridor) hold complex goals and preferences, sometimes conflicting with one another, and they want to find a collective decision.

In this thesis, we will bring together ideas and techniques coming from multiple areas of computer science — ranging from multi-agent systems, logic, game theory and computational social choice — to formalize and propose solutions for a set of related problems in collective decision-making with rational agents.

1.1 Background and Context

This section wants to give a bird eye's view on the main areas of computer science to which this thesis is related. A comprehensive introduction to each one of these areas (as well as their intersections) would require multiple volumes, and is beyond the scope of this section. For this reason we point to some key references in the literature, often in the form of textbooks. Moreover, throughout the thesis we will give more exhaustive and detailed presentations of the related work when introducing each one of the problems we study.

The story in our incipit could be modeled in computer science as a **multi-agent system**, an area broadly defined as the study of the interactions of multiple agents (e.g., the three researchers) who can act autonomously towards the satisfaction of some goal or objective (see Shoham and Leyton-Brown, 2008, Wooldridge, 2009 and Weiss, 2013 for an introduction). In our case, in particular, the goals of the agents are individual. An important aspect of multi-agent systems is thus how to model intelligent agents, and many different approaches have been proposed in the literature (Wooldridge, 2013). We can mention, for instance, one of the first popular models originating from the philosophical analysis of intentions by Bratman (1987), where agents are modeled as holding beliefs, desires, and intentions — i.e., the BDI architectures (Cohen and Levesque, 1990; Rao and Georgeff, 1991, 1995; Dastani and van der Torre, 2002). In our story, however, the focus is not on how the agents derive their goals from beliefs and desires (see, e.g., Lorini, 2017), but rather on how they can obtain a collective decision based on (a suitable representation of) their goals and preferences.

In this respect, the field of *qualitative decision theory* comes into play in multi-agent systems for modeling agents in decision-making situations. While in classical decision theory an agent is modeled as having a probability distribution over possible outcomes (depending on the actions they take) which gives them a numerical utility that they then try to maximize, in qualitative decision theory agents use qualitative methods to express which outcomes they prefer and which ones they consider more probable (Wellman and Doyle, 1991; Boutilier, 1994; Brafman and Tennenholtz, 1996; Dubois et al., 2001). Qualitative descriptions of preferences over outcomes are usually easier to handle and to express for the agents than quantitative ones.

An important tool that has been used in multi-agent systems to formally represent agents and their interactions has been that of **logic** (see Van Dalen, 2004 for an introduction to propositional and predicate logic, and Blackburn et al., 2006 for an introduction to modal logic). We mentioned above some logics for BDI agents, but numerous other logical formalisms have been defined for multi-agent systems (see, e.g., Goranko and Jamroga, 2004 and Herzig, 2015 for some comparisons between logics for multi-agent systems and different semantics). The use of logic allows to rigorously represent a system, or some of its components (e.g., the agents' goals and preferences) in a formal language, and to reason about it.[1] In particular, a compact representation of goals and preferences given by a logical language is also important for research in **artificial intelligence** (Russell and Norvig, 2016), whose aim is

[1] Logic has proven useful for modeling and reasoning in a variety of other fields in computer science, an example being that of *planning*, where an agent in a certain initial state has to find a series of basic actions allowing them to attain a goal state (Fikes and Nilsson, 1971; Kautz and Selman, 1992; Bylander, 1994; Kautz and Selman, 1996; Kautz et al., 1996). We can describe the collective choice that our agents will take as an "abstract plan", but we won't focus on the actions needed for its actual realization (e.g., the phone call that the researchers need to make in order to book the gala dinner in the restaurant of their choice).

to create artificial agents displaying "intelligent" behavior, including for the areas of individual and collective decision-making.

In the scenario described in our initial example the goals of the agents are conflicting, and since they all want to achieve their goal regardless of the satisfaction of those of the others, their mutual interactions become adversarial. The field of **game theory** (see the books by Gibbons, 1992 and Leyton-Brown and Shoham, 2008 for reference) provides a mathematical model to study agents acting strategically towards the maximization of their (expected) individual utilities. If we are given the description of a game, in the form of the actions available to the agents and the payoffs they get for any possible combination of actions, and if we assume that the agents are rational, meaning that they want to maximize their payoff, the prediction of which strategies the agents will play is given by studying different solution concepts: for instance, Lucy has a winning strategy if no matter what the other agents do, she always knows when to speak (and when to stay silent) to change Ann's opinion.

Finally, **computational social choice** (see the recently published handbook by Brandt et al., 2016) collects multiple formal models to study the aggregation of opinions and preferences of a group of agents, as the ones expressed by Lucy, Ann, and Bob on the organization of the conference or the syllabus for the course.[2] This field of computer science studies various computational problems for voting rules, such as calculating the winner or manipulating the outcome (Hemaspaandra et al., 2005; Conitzer et al., 2007; Conitzer and Walsh, 2016), and it originates from the branch of economics known as social choice theory. Within social choice theory, seminal results in voting include those by Arrow (1951), Gibbard (1973) and Satterthwaite (1975), who showed that certain sets of desirable properties cannot be satisfied all together by any voting rule, as well as that of May (1952), who proved that the majority rule can be characterized by a set of axioms. In computational social choice, the field of *judgment aggregation* is concerned in particular with aggregating the yes/no opinions of multiple agents over binary issues (see the book by Grossi and Pigozzi, 2014 and the recent chapter by Endriss, 2016). Logic is used in judgment aggregation as a tool to model the questions that the agents have to approve or reject, as well as potential integrity constraints expressing some external law or physical constraint.

We have thus seen which areas of computer science can give us models and tools to study the situations as the one in the story at the beginning of this thesis. In Section 1.2 we will state our research questions and in Section 1.3 we will see how we tackle them by providing a chapter-by-chapter overview of this thesis.

1.2 Research Questions

We want to provide some answers to two research questions in collective decision-making, which are at the core of the interactions between the agents in our initial story. We can formulate them as follows:

1. How can we design **aggregation procedures** to help a group of agents having compactly expressed goals and preferences make a collective choice?

2. How can we model agents with conflicting goals who try to get a better outcome for themselves by **acting strategically**?

[2]The recent paper by Conitzer (2019) gives an account on how ideas from game theory, computational social choice and artificial intelligence can be used for the problem discussed above of designing agents with identities and beliefs.

For the first question we need to define some functions that can take as input goals and preferences as the ones stated by the researchers in our story to obtain a collective decision. In particular, we need to choose suitable languages for the agents to express their goals and preferences, which may be more complex than just accepting or rejecting an issue. We also want to know which properties are satisfied by the functions we define, and how difficult it is to compute their outcomes.

For the second question we need to consider two kinds of strategic behavior. On the one hand, an agent may realize that by submitting an untruthful goal she would get a better result for herself (possibly at the expenses of some other agent). On the other hand, the agents may exploit the fact that they are part of a common network, as they may know one another or be influenced by each other for the opinions they hold, to obtain their goals. We need to identify under which conditions such situations of strategic behavior may occur and which steps could be taken to limit them.[3]

1.3 Thesis Overview

The thesis is structured into two parts to address the two research questions we delineated in Section 1.2. The first part concerns the aggregation of compactly represented goals and preferences, while the second part studies the strategic behavior of agents in different multi-agent scenarios.

Chapter 2 introduces two known formal languages to express goals and preferences: propositional logic and conditional preference networks (CP-nets). We give the basic definitions of propositional logic (Section 2.1) and of CP-nets (Section 2.3), and we present a related literature review on goals in propositional logic and extensions of CP-nets (Sections 2.2 and 2.4, respectively). Finally, we compare the two languages in Section 2.5.

In **Chapter 3** we define the decision-making framework of *goal-based voting*, where agents express goals as propositional formulas over binary issues. We introduce different voting rules inspired from the fields of voting, judgment aggregation and belief merging in Section 3.1.2. In particular we define three novel adaptations of issue-wise majority and prove that the three rules are different in Proposition 3.1. In Section 3.2 we define a variety of axioms, i.e., desirable properties that we would like a rule to satisfy, and we prove numerous results on how the axioms relate to one another and whether our proposed rules satisfy them. In line with similar results in the literature, we prove in Theorem 3.1 that some combinations of axioms are mutually inconsistent (i.e., resoluteness, anonymity and duality). Our main result is Theorem 3.3 where we provide a characterization of *TrueMaj*, one of our majorities.

In Section 3.3 we study the computational complexity of the problem of computing the outcome of our rules. Goal-based voting rules turn out to be in general much harder to compute than their voting counterparts, especially the three generalizations of the majority rule which are all hard for the Probabilistic Polynomial Time class (Theorems 3.8, 3.9 and 3.10). We find however some polynomial results by restricting the language of goals to conjunctions (Theorem 3.11) and to disjunctions (Theorem 3.12). In Section 3.4 we compare the framework of goal-based voting with the close frameworks of belief merging and judgment aggregation (with or without

[3]Some arguments on why manipulation of voting rules is to be considered a negative phenomenon were given by Conitzer and Walsh (2016). For instance, it would be unfair towards agents who have less information or cognitive capacities to think about a possible manipulation, and in general it could be considered a waste of computational resources.

abstentions). In particular, we compare the frameworks with respect to the axioms, or *postulates* in belief merging, that the rules satisfy.

In **Chapter 4** we study the problem of aggregating individual preferences expressed as *generalized CP-nets*, a language with which agents can state their preference over the values for some variable given the same (partially specified) state of the world. We adapt four semantics known in the literature on the aggregation of classical CP-nets (Definition 4.3) and we study the computational complexity of different dominance problems for these semantics. In particular, we study the *Pareto* semantics (Section 4.2.1) which can be seen as a unanimous aggregation procedure, the *majority* semantics (Section 4.2.2) which corresponds to an absolute majority aggregator, the *max* semantics (Section 4.2.3) which is a majority with respect to the agents who did not abstain, and the *rank* semantics (Section 4.2.4) which is reminiscent of the Borda rule in voting. We find that for the most part the complexity problems do not become harder when switching from a single agent to multiple agents, with the sole exception of checking that an outcome is non-dominated, going from polynomial to PSPACE-complete for *Pareto* and *maj* semantics (Theorems 4.1 and 4.5, respectively) and to PSPACE membership for *max* semantics (Theorem 4.9), as well as existence of a non-dominated outcome for *Pareto* semantics, going from NP-complete to PSPACE-complete (Theorem 4.2).

Chapter 5 introduces the second part of this thesis, focused on agents behaving strategically. We first recall Boolean games and some classical definitions of solution concepts in game theory. In Section 5.1 we give the basic definitions of known logics for modelling time and strategic reasoning, i.e., ATL and LTL, which we intepret over the Concurrent Game Structures (CGS) of Definition 5.1. In Section 5.2 we present the related work at the intersection of logic, game theory and networks of agents.

In **Chapter 6** we study the setting of goal-based voting introduced in Chapter 3 under the assumption that now agents can manipulate the outcome by submitting untruthful goals. In Section 6.1 we add some definitions to model the manipulation problem, distinguishing in particular three types of agents (optimists, pessimists, and expected utility maximizers) and three types of manipulation strategies (unrestricted, erosion, and dilatation).

In Section 6.2 we establish some general manipulation results for the three issue-wise majority rules defined in Section 3.1.2: Theorem 6.1 proves that the three majoritarian rules are manipulable under all the types of manipulation we study. This negative result leads us to try to refine the bounds of manipulability by focussing on subclasses of goals: conjunctions (Section 6.2.1), disjunctions (Section 6.2.2) and exclusive disjunctions (Section 6.2.3). Theorem 6.7 gives us some conditions under which our majority rules become indeed strategy-proof for conjunctive and disjunctive goals. Finally, Section 6.3 studies the computational complexity of the manipulation problem for the resolute majoritarian voting rules, which is found to be as hard as computing their outcome.

In **Chapter 7** we introduce *influence games*, a class of iterated games where agents can control the spread of their opinions over a network. In Section 7.1 we present the opinion diffusion dynamics and the strategic actions available to the agents (i.e., the possibility to use or not their influence power). While agents can use any rule they like to aggregate the opinons of their influencers, we focus on the unanimous aggregation procedure of Definition 7.7. Agents hold goals expressed in a variant of LTL and we focus in particular on *consensus* and *influence* types of goals. In Section 7.3 we present results of both game-theoretic and computational nature for influence games. Propositions 7.1 and 7.2, as well as Example 7.5 give us an insight on which types of solutions can be attained depending on the influence network and on the goals of the agents, while Section 7.3.2 tells us that finding Nash equilibria for influence games is in general computationally hard.

Chapter 8 introduces a new subclass of CGS where agents have *shared* control over subsets of propositional variables, generalizing existing work on CGS with exclusive control. In Section 8.1 we give the definitions of both types of CGS, and in Section 8.2 we give some examples of iterated games that can be modeled by these structures, such as iterated Boolean games with exclusive control, the influence games of Chapter 7 and aggregation games. Our main result is Theorem 8.1 in Section 8.3, showing how it is possible to reduce the model-checking problem of ATL^* formulas on CGS with shared control to model-checking (a translation of) ATL^* formulas on CGS with exclusive control.

Chapter 9 concludes the thesis. In Section 9.1 we summarize what we achieved in the dissertation with respect to our research questions, and in Section 9.2 we provide some perspectives for future work.

Sources of the Chapters

The work presented in this thesis is based on the following publications:

- Chapter 3 is based on:

  Novaro, A., Grandi, U., Longin, D., and Lorini, E. (2018). Goal-Based Collective Decisions: Axiomatics and Computational Complexity. In *Proceedings of the 27th International Joint Conference on Artificial Intelligence and the 23rd European Conference on Artificial Intelligence (IJCAI-ECAI-2018)*.

  Novaro, A., Grandi, U., Longin, D., and Lorini, E. (2018). From Individual Goals to Collective Decisions (Extended Abstract). In *Proceedings of the 17th International Conference on Autonomous Agents and Multiagent Systems (AAMAS-2018)*.

- Chapter 4 is based on:

  Haret, A., Novaro, A., and Grandi, U. (2018). Preference Aggregation with Incomplete CP-nets. In *Proceedings of the 16th International Conference on Principles of Knowledge Representation and Reasoning (KR-2018)*.

- Chapter 6 is based on:

  Novaro, A., Grandi, U., Longin, D., and Lorini, E. (2019). Strategic Majoritarian Voting with Propositional Goals (Extended Abstract). In *Proceedings of the 18th International Conference on Autonomous Agents and Multiagent Systems (AAMAS-2019)*.

- Chapter 7 is based on:

  Grandi, U., Lorini, E., Novaro, A., and Perrussel, L. (2017). Strategic Disclosure of Opinions on a Social Network. In *Proceedings of the 16th International Conference on Autonomous Agents and Multiagent Systems (AAMAS-2017)*.

- Chapter 8 is based on:

  Belardinelli, F., Grandi, U., Herzig, A., Longin, D., Lorini, E., Novaro, A., and Perrussel, L. (2017). Relaxing Exclusive Control in Boolean Games. In *Proceedings of the 16th conference on Theoretical Aspects of Rationality and Knowledge (TARK-2017)*.

Part I

Aggregation

Chapter 2

Compact Languages for Goals and Preferences

The first part of this dissertation focuses on how propositional logic can be used to compactly represent goals and preferences. In many real-world applications, in fact, the space of alternatives from which a collective choice needs to be taken is combinatorial and it would be too burdensome for the agents to provide a full ranking of all possible outcomes (Chevaleyre et al., 2008).

An example is that of a group of agents having to decide which values to attribute to the features composing a complex object: think, for instance, to the three researchers in the story at the incipit of this thesis who have to decide over the different "features" composing the conference (i.e., location, posters, gala dinner, and so on), or a group of friends having to plan a common trip to a city with multiple points of interest. Observe that already for just three binary features, if agents were to rank all possible combinations of choices they would need to order eight potential outcomes. The challenge is then to find a language to represent the preferences and goals of the agents over all the possible alternatives which is at the same time compact and expressive (see also the survey by Lang and Xia, 2016).

In this chapter we present the basic definitions for propositional goals and CP-nets, which will be useful for Chapter 3, where we will introduce a framework for voting with propositional goals, and for Chapter 4, where we will aggregate preferences expressed as CP-nets (with a component in propositional logic).

2.1 Propositional Goals

In propositional logic we have a countable set of *propositional variables* $PV = \{p, q, \dots\}$ and formulas are built according to the following grammar:

$$\varphi ::= p \mid \neg\varphi \mid \varphi_1 \wedge \varphi_2 \mid \varphi_1 \vee \varphi_2$$

where $p \in PV$. Namely, we have the *negation* of φ ($\neg\varphi$), the *conjunction* of φ_1 and φ_2 ($\varphi_1 \wedge \varphi_2$) and the *disjunction* of φ_1 and φ_2 ($\varphi_1 \vee \varphi_2$). We write the *implication* $\varphi_1 \to \varphi_2$ as a shorthand for $\varphi_2 \vee \neg\varphi_1$; we write the *biconditional* $\varphi_1 \leftrightarrow \varphi_2$ in place of $(\varphi_1 \to \varphi_2) \wedge (\varphi_2 \to \varphi_1)$; we write the *exclusive disjunction* $\varphi_1 \oplus \varphi_2$ for $\neg(\varphi_1 \leftrightarrow \varphi_2)$. Moreover, we write $\top$ for $p \vee \neg p$ and $\bot$ for $p \wedge \neg p$.

A *literal* L_p is a propositional variable p or its negation $\neg p$: we call it *positive* in case $L_p = p$ and *negative* if $L_p = \neg p$. An *interpretation* is a function $v : \mathcal{I} \to \{0, 1\}$ associating a binary value to each propositional variable, such that value 1 is interpreted as 'true' and value 0 as 'false'. An interpretation v makes formula φ true, written $v \models \varphi$, according to the following conditions:

$$\begin{array}{rll} v \models p & \text{iff} & v(p) = 1 \\ v \models \neg\varphi & \text{iff} & \text{it is not the case that } v \models \varphi \\ v \models \varphi_1 \wedge \varphi_2 & \text{iff} & v \models \varphi_1 \text{ and } v \models \varphi_2 \\ v \models \varphi_1 \vee \varphi_2 & \text{iff} & v \models \varphi_1 \text{ or } v \models \varphi_2 \end{array}$$

The set of interpretations that make a certain formula φ true, i.e, the set of its *models*, is represented by the set $\mathrm{Mod}(\varphi) = \{v \mid v \models \varphi\}$. Two formulas φ and ψ are *equivalent*, denoted by $\varphi \equiv \psi$, if and only if $\mathrm{Mod}(\varphi) = \mathrm{Mod}(\psi)$.

A *propositional goal* is then a propositional formula expressed over variables denoting the issues at stake. For instance, if we have variable p for the statement "We will visit the park", variable s for "We will visit the museum" and variable r for "We will visit the old lighthouse", we can write formula $\varphi = (p \wedge r) \rightarrow s$ to express the complex goal "If we visit the lighthouse and the park, then we are also going to visit the museum". The models of φ are then $\mathrm{Mod}(\varphi) = \{(111), (011), (101), (001), (100), (010), (000)\}$.

We can also define *restrictions* on the language of goals. In particular, we define the languages $\mathcal{L}^\star$ as follows:

$$\varphi := p \mid \neg p \mid \varphi_1 \star \varphi_2$$

where $\star \in \{\wedge, \vee, \oplus\}$. We call $\mathcal{L}^\wedge$ the *language of conjunctions* (or cubes), where each formula is a conjunction of literals; $\mathcal{L}^\vee$ the *language of disjunctions* (or clauses), where each formula is a disjunction of literals; and $\mathcal{L}^\oplus$ the *language of exclusive disjunctions*, where each formula is an exclusive disjunction of literals. Note that negation only applies to propositional variables and not to complex formulas. Thus, we have that formula $\neg p \wedge q$ belongs to $\mathcal{L}^\wedge$, while formula $\neg(p \wedge q)$ does not.

2.2 Related Work on Propositional Goals

Propositional goals have been proposed as a compact representation of dichotomous preferences over combinatorial alternatives described by binary variables (e.g., think of *multiple referenda* where voters are asked to either accept or reject a number of issues). Lang (2004) has studied numerous representation languages for goals. The simplest representation of preferences R_{basic} is such that each agent expresses a goal in propositional logic and has utility either 1 or 0 (i.e., dichotomous) depending on whether her goal is satisfied or not.

In the same paper, Lang also proposes a representation for *weighted goals*, where propositional formulas in a *goal base* have a numerical value attached representing their weight.[1] Uckelman et al. (2009) also studied weighted goals, where the utility of an alternative is calculated as the sum of the weights of the formulas it satisfies. They defined different languages (with various restrictions on the syntax of the goal or on the range of the weights) to characterize classes of utility functions, to compare languages with respect to succinctness, and to study the computational complexity of finding the most preferred alternative given a certain utility function.

Finally, Lang also discussed *prioritized goals*, where propositional formulas in the goal base are associated with a weak order (a *priority relation*). Brewka (2004) proposed a Logical Preference Description (LPD) language to describe qualitatively the preferences of the agents over possible outcomes given their individual ranked knowledge base (which they use to represent prioritized goals in propositional logic).

[1]Observe that R_{basic} can be seen as a special case of weighted goals where the goal base contains a single goal, whose weight can be assumed to be 1.

The literature on *Boolean games* (Harrenstein et al., 2001; Bonzon et al., 2006) also studies situations where agents are endowed with propositional goals. More precisely, each agent i has a propositional goal γ_i over a set of variables Φ but they are also each assigned a subset of variables $\Phi_i \subseteq \Phi$ whose assignment they can exclusively control. We will see Boolean games again in more detail in the second part of this thesis (Chapter 5).

Two other important frameworks are strongly related to the idea of aggregating the propositional goals of a group of agents. The first one is *judgment aggregation* (List, 2012; Lang and Slavkovik, 2014; Endriss, 2016), in particular in its *binary aggregation* model (Dokow and Holzman, 2010a; Grandi and Endriss, 2011), which can be seen as a form of voting where agents express propositional goals having a single model (i.e., their goals are *complete* conjunctions over all the issues at stake). The generalization of judgment aggregation to abstentions (by Gärdenfors, 2006; Dietrich and List, 2008; Dokow and Holzman, 2010b, and more recently by Terzopoulou et al., 2018) corresponds to agents having *partial* conjunctions of literals as goals (i.e., goals in the language $\mathcal{L}^\wedge$ described in Section 2.1). Recent work in judgment aggregation has advanced some ideas to give agents more flexibility in expressing their opinions. For instance, Miller (2008) and Benamara et al. (2010) work in the framework of judgment aggregation where issues can be divided into premises and conclusions, and instead of asking all agents to respect some external *integrity constraint* (e.g., modeling a law) they let them follow individual rules to express their judgments. Endriss (2018) distinguishes between rationality and feasibility constraints, where the former has to be respected by the agents (and it is the same for all of them) while the latter has to be satisfied by the outcome of the aggregation process.

The second related framework is *belief merging* (Konieczny and Pérez, 2002), that aims at aggregating the beliefs coming from multiple agents in order to get the beliefs of the group. Each agent holds a set of beliefs in the form of a belief base K of propositional formulas. Belief bases are aggregated via functions called merging operators, satisfying a set of properties called IC postulates, and the aggregation takes into account the presence of integrity constraints. Dastani and van der Torre (2002) also proposed an attempt at using logic-based belief merging for goals, assessed by axiomatic properties from belief revision.

In Chapter 3 we will define a framework for the aggregation of propositional goals based on voting, building on the R_{basic} representation by Lang (2004) mentioned above. We will also specify in more details the relationship of our proposed framework with both judgment aggregation and belief merging in Section 3.4.

2.3 CP-nets

The framework of CP-nets has been introduced to model situations where the preference over the value for some variable depends on the choice made on another variable, everything else being equal (Boutilier et al., 2004). The classical example to explain CP-nets takes place at a restaurant. Lucy wants to order a main course and a beverage and let us assume that, for simplicity, the options for the main course in this restaurant are just fish or steak, and for beverages only white or red wine. Overall, Lucy prefers fish over steak. However, the preference over type of wine for Lucy depends on what she is eating: if she orders fish she would like to drink white wine rather than red, and vice-versa if she orders steak.

In order to model situations like the one above we need *variables* to express the features or components on which a choice has to be made. In our example, we have a variable B for the beverages and a variable M for the main course. The *domain* $D(X)$ of a variable X contains all the possible values that the variable can assume. The domain of beverages, for instance, would be $D(B) = \{w, r\}$ for white or red wine, and the one for main courses would be $D(M) = \{s, f\}$ for steak or fish. Agents express their preferences through *(conditional preference) statements*[2] of the form $xy \ldots : z_1 \triangleright z_2$. The left part $xy \ldots$, which we call *precondition*, describes some *state of the world* or *outcome*, and the right part $z_1 \triangleright z_2$ expresses the *preference* over the values of some variable in the given outcome. Lucy, for instance, expresses three statements:

$$
\begin{array}{ll}
(\varphi_0) & f \triangleright s \\
(\varphi_1) & f : w \triangleright r \\
(\varphi_2) & s : r \triangleright w
\end{array}
$$

whose meaning is precisely "Fish is always preferred to steak" (φ_0),[3] "If fish is served, white wine is preferred to red" (φ_1), and "If steak is served, red wine is preferred to white" (φ_2).

We have just seen in which sense for CP-nets the preference over the value for some variable depends on which value we chose for another variable. Another important aspect to consider is that we can compare two different outcomes by using some conditional preference statement only *ceteris paribus*, i.e., everything else being equal. For instance, suppose our restaurant decides to improve their menu by adding a (short) list of desserts, which we model by a new variable D. The desserts offered are chocolate cake or banana bread, and hence the domain of D is $D(D) = \{c, b\}$. According to φ_1, Lucy prefers a world *fwc* (where the menu is fish, white wine, and chocolate cake) to a world *frc* (where the menu is fish, red wine, and chocolate cake). However, we cannot say that Lucy prefers *fwc* to *frb* (where the menu is fish, red wine, and banana bread), as the two outcomes differ on the value of variable D. Hence, only a small number of comparisons are made between outcomes when using conditional preference statements.

This gives us an explanation of the first part of the name 'CP-nets': CP is an acronym which stands at the same time for *conditional preferences*, as the statements expressed by the agents, and *ceteris paribus*, as the comparisons made between outcomes. The second part of the name is short for *networks*, as it is possible to construct a graph illustrating how variables depend on one another as we explain below.

The nodes in the *dependency graph* (or *preference graph*), are the variables of our problem. For instance, in the first version of the menu of the restaurant we would have one node for variable M and one for variable B. Then, each node X is annotated with a *conditional preference table* $CPT(X)$ containing a preference statement over values of X for each possible combination of values for the variables upon which X depends. Figure 2.1 represents the CP-net of Lucy's preferences in our example.

2.4 Related Work on CP-nets

The literature on CP-nets includes many extensions and variations of the basic framework we presented above. We have, for instance, *tradeoff-enhanched* CP-nets (or TCP-nets), which extend CP-nets by allowing the agent to say that in a particular state of the world it is more important to get a better value for variable X rather than getting

[2]Alternatively called *conditional preference rules* (Goldsmith et al., 2008).

[3]In Chapter 4 this statement will rather be written as $\top : f \triangleright s$ for generalized CP-nets. We follow here the presentation given by Boutilier et al. (2004) in the case of complete CP-nets.

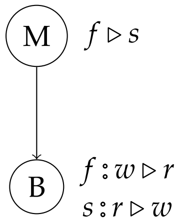

FIGURE 2.1: Lucy's CP-net.

a better value for variable Y (Brafman and Domshlak, 2002; Brafman et al., 2006). In *conditional preference theories* (CP-theories), an agent is allowed to state than in a given state of the world (which does not need to be completely specified), some value x_1 for variable X is preferred to value x_2, regardless of which values are assigned to the variables in some set W (Wilson, 2004). If agents are given a set of objects $\mathcal{S}_1$, and they can express that they prefer to also get the bundle of goods $\mathcal{S}_2$ rather than the bundle of goods $\mathcal{S}_3$, we are in the framework of *conditional importance networks* (CI-nets) by Bouveret et al. (2009).

Agents may also be allowed to express incomplete preference statements. For instance, Lucy may want to express only φ_1 in our previous example. Depending on the specific application, we may want to handle this lack of information about preferences over full outcomes by sticking to the *ceteris paribus* semantics or by departing from it, as done for instance in the *totalitarian* semantics studied by Ciaccia (2007) for incomplete databases. In Chapter 4 we will work in the framework of another type of incomplete CP-nets, i.e., *generalized* CP-nets (gCP-nets), where agents are allowed to express conditional preference statements by using arbitrary propositional formulas in the precondition. Moreover, we will be working in the context of multiple agents (mCP-nets), whose incomplete CP-nets have to be aggregated.

2.5 Comparing Propositional Goals and CP-nets

Propositional formulas allow us to compactly represent goals, and with CP-nets we can compactly represent *ceteris paribus* preferences over outcomes. Additionally, in gCP-nets agents can express any propositional formula as the precondition of their preference statements. We could then wonder whether it is possible to construct a propositional formula expressing the same preference ordering over outcomes induced by an arbitrary CP-net, and vice-versa to construct a CP-net expressing the same preference ordering induced by a propositional goal.

If we assume that the preferences induced by propositional formulas are dichotomous, the answer is no: in fact, an agent holding a propositional goal divides the outcomes into just two classes, with the models of her goal in one and the counter-models in the other.[4] The following example illustrates this point:

Example 2.1. Consider two binary variables A and B whose respective domains are $D(A) = \{a, \overline{a}\}$ and $D(B) = \{b, \overline{b}\}$. A propositional goal of the form $a \vee b$ induces an ordering where ab is preferred to $\overline{a}\overline{b}$: however, this relation cannot be reproduced by a CP-net as both variables A and B are changing their values in the two outcomes,

[4]In Section 6.1 we will briefly discuss how the preferences of agents having propositional goals could be alternatively defined based on the Hamming distance. The focus of the thesis will however be on a dicothomous interpretation of preferences induced by goals.

thus contradicting the *ceteris paribus* assumption. On the other hand, a CP-net composed by statements as $a \rhd \overline{a}$, $a : b \rhd \overline{b}$ and $\overline{a} : \overline{b} \rhd b$ induces the non-dichotomous preference relation over outcomes where ab is strictly preferred to $a\overline{b}$, which in turn is strictly preferred to $\overline{a}\overline{b}$.

There are however examples in the literature of more general languages able to express both propositional goals and CP-nets. In particular, Bienvenu et al. (2010) proposed the "prototypical" preference logic PL whose formulas are of the form $\Psi = \alpha \rhd \beta \parallel F$ where α and β are propositional formulas and F is a set of propositional formulas. Intuitively, the meaning of Ψ is that an outcome satisfying α is preferred to an outcome satisfying β provided that they agree on the interpretation of formulas in F. Both CP-nets and propositional goals (as a special case of prioritized goals) are then shown to be fragments of PL.

Chapter 3

Goal-based Voting

Social choice and voting have provided useful techniques for the design of rational agents that need to act in situations of collective choice (Brandt et al., 2016). In particular, judgment aggregation has been applied in various settings: from multi-agent argumentation (Awad et al., 2017) to the collective annotation of linguistic corpora (Qing et al., 2014). However, when considering collective decision-making in practice, the rigidity of judgment aggregation in asking individuals to provide complete judgments over issues can become an obstacle. Consider the following example, inspired by the *traveling group problem* (Klamler and Pferschy, 2007):

Example 3.1. An automated travel planner is organizing a city trip for a group of friends: Ann, Barbara, and Camille. They have to decide whether to visit the Lighthouse, the Museum, and the Park. Ann wants to see all the points of interest, Barbara prefers to have a walk in the Park, and Camille would like to visit a single place but she does not care about which one. A judgment-based planner would require all agents, including Camille, to specify a full valuation over all issues.

For instance, we could obtain the following situation:

	Lighthouse	Museum	Park
Ann	☑	☑	☑
Barbara	☒	☒	☑
Camille	☒	☑	☒

By taking a simple majority vote, the collective plan would be to visit both the Museum and the Park. This does not satisfy Camille: by being asked for a complete judgment, she was unable to express her truthful goal to "visit a single place, no matter which one" thus getting an outcome that she does not like.

In order to take care of situations such as the one presented in Example 3.1, we present a framework for the aggregation of individual goals expressed in propositional logic. We define some voting rules that could be used to return a collective decision, and we analyze what are the properties that they satisfy, giving an axiomatic characterization for one of them. We study how hard it is from a computational perspective to determine the outcome of the introduced rules. Finally, we compare our framework with belief merging and judgment aggregation.

3.1 Framework

In the first part of this section we present the basic definitions of the framework of *goal-based voting*, where agents express their goals as propositional formulas. In the second part we define and compare a selection of voting rules, inspired by voting, belief merging and judgment aggregation.

3.1.1 Basic Definitions

A group of *agents* in the finite set $\mathcal{N} = \{1, \ldots, n\}$ has to take a collective decision over some *issues* in the finite set $\mathcal{I} = \{1, \ldots, m\}$. We assume that issues have binary values: when they take value 1 it means they are accepted, and when they take value 0 it means they are rejected. Each agent $i \in \mathcal{N}$ has an *individual goal* γ_i, which is expressed as a consistent propositional formula. Formally, each γ_i is written in the propositional language $\mathcal{L}$ whose propositional variables are all the issues $j \in \mathcal{I}$ (see Section 2.1 for the basic definitions of propositional logic).

For simplicity, we sometimes write an interpretation v as vector $(v(1), \ldots, v(m))$. From a voting perspective, each interpretation corresponds to an alternative (a candidate), and the models of γ_i are those alternatives (candidates) that agent i supports. We also assume that issues are independent from one another, meaning that all possible interpretations over $\mathcal{I}$ are feasible (or, equivalently, that there are no integrity constraints).

For agent $i \in \mathcal{N}$ and issue $j \in \mathcal{I}$, we indicate by $m_{ij}^x = |\{v \in \text{Mod}(\gamma_i) \mid v(j) = x\}|$ for $x \in \{0, 1\}$ how many times issue j has value x in the models of agent i's goal. Then, we denote by vector $m_i(j) = (m_{ij}^0, m_{ij}^1)$ the two totals of zeroes and ones for issue j in the models of γ_i. A *goal-profile* (or simply, a *profile*) $\mathbf{\Gamma} = (\gamma_1, \ldots, \gamma_n)$ is a vector collecting all the goals of the agents in $\mathcal{N}$.

We now formalize Example 3.1 to illustrate all the above definitions:

Example 3.2. Agents Ann, Barbara, and Camille, are modeled by set $\mathcal{N} = \{1, 2, 3\}$. The three choices they have to make over the lighthouse, the museum, and the park, are modeled by the set $\mathcal{I} = \{1, 2, 3\}$. Ann's goal is $\gamma_1 = 1 \land 2 \land 3$, while Barbara's goal is $\gamma_2 = \neg 1 \land \neg 2 \land 3$ and Camille's goal is $\gamma_3 = (1 \land \neg 2 \land \neg 3) \lor (\neg 1 \land 2 \land \neg 3) \lor (\neg 1 \land \neg 2 \land 3)$. The models of the agents' goals correspond to $\text{Mod}(\gamma_1) = \{(111)\}$, $\text{Mod}(\gamma_2) = \{(001)\}$ and $\text{Mod}(\gamma_3) = \{(100), (010), (001)\}$. Ann only has one model for her goal (i.e., $|\gamma_1| = 1$) and for issue 3, for instance, we thus have $m_{13} = (m_{13}^0, m_{13}^1) = (0, 1)$. Camille has three models for her goal γ_3, and for issue 2 we have $m_{32} = (m_{32}^0, m_{32}^1) = (2, 1)$. Profile $\mathbf{\Gamma} = (\gamma_1, \gamma_2, \gamma_3)$ captures this situation.

The agents take a decision over the issues at stake by means of voting rules, whose input is a profile of n formulas (each one submitted by one agent in $\mathcal{N}$) and whose output is a set of interpretations over the m issues in $\mathcal{I}$, for all n and m. The formal definition is as follows:

Definition 3.1. A *goal-based voting rule* is a collection of functions for all $n, m \in \mathbb{N}$ defined as:

$$F : (\mathcal{L}_{\mathcal{I}})^n \to \mathcal{P}(\{0, 1\}^m) \setminus \{\emptyset\}$$

Vector $F(\mathbf{\Gamma})_j = (F(\mathbf{\Gamma})_j^0, F(\mathbf{\Gamma})_j^1)$ where $F(\mathbf{\Gamma})_j^x = |\{v \in F(\mathbf{\Gamma}) \mid v_j = x\}|$ for $x \in \{0, 1\}$ indicates the total number of zeroes and ones in the outcome of F for j in $\mathbf{\Gamma}$. For simplicity, we write $F(\mathbf{\Gamma})_j = x$ for $x \in \{0, 1\}$ in case $F(\mathbf{\Gamma})_j^{1-x} = 0$.

3.1.2 Goal-based Voting Rules

We start by presenting a rule inspired by simple approval voting (Brams and Fishburn, 1978, 2007; Laslier and Sanver, 2010). For all profiles the *Approval* rule chooses, among all the models of the agents' goals, those interpretations that satisfy a maximal number of agents' goals. Formally:

$$Approval(\mathbf{\Gamma}) = \underset{v \in \text{Mod}(\bigvee_{i \in \mathcal{N}} \gamma_i)}{\arg\max} |\{i \in \mathcal{N} \mid v \in \text{Mod}(\gamma_i)\}|.$$

The *Approval* rule has also been previously introduced in the work by Lang (2004) under the name of *plurality rule*, and in belief merging by Konieczny and Pérez (2002) as a $\Delta_{\mu}^{\sum,d}$ rule.[1] Variations of *Approval* have been used for multi-winner elections (Aziz et al., 2015), where the satisfaction of an agent is modeled by different vectors depending on how many of her candidates (in this case, models of her goal) are chosen by the rule. Despite its intuitive appeal, approval-based voting is not adapted to combinatorial domains in which a large number of alternatives may be approved by a few agents only, as illustrated by the following example:

Example 3.3. Consider a profile $\mathbf{\Gamma} = (\gamma_1, \gamma_2, \gamma_3)$ for three agents and three issues such that $\gamma_1 = 1 \wedge 2 \wedge 3$, $\gamma_2 = 1 \wedge 2 \wedge \neg 3$ and $\gamma_3 = \neg 1 \wedge \neg 2 \wedge \neg 3$. As each goal has exactly one model, and there is no interpretation which satisfies more than one goal at the same time, we have that the outcome is $Approval(\mathbf{\Gamma}) = \{(111), (110), (000)\}$ which leaves the agents with the same options they began with.

The next class of rules that we define is inspired by the quota rules from judgment aggregation (Dietrich and List, 2007a), where each issue needs a certain quota of votes to be approved in the outcome. First, let $\mu_\varphi : \text{Mod}(\varphi) \to \mathbb{R}$ be a function associating to each model v of φ some weight $\mu_\varphi(v)$. Observe that μ_φ may associate different weights to distinct models of the same formula φ.

Let *threshold rules* be defined as:

$$TrSh^{\mu}(\mathbf{\Gamma})_j = 1 \;\; \textit{iff} \;\; \big(\sum_{i \in \mathcal{N}} (w_i \cdot \sum_{v \in \text{Mod}(\gamma_i)} v(j) \cdot \mu_{\gamma_i}(v))\big) \geq q_j$$

such that $1 \leq q_j \leq n$ for all $j \in \mathcal{I}$ is the quota of issue j, for each $v \in \text{Mod}(\gamma_i)$ we have $\mu_{\gamma_i}(v) \neq 0$ and $w_i \in (0, 1]$ is the individual weight of agent i.

Intuitively, each issue j in threshold rules is considered independently from the others, and there is a certain quota q_j that has to be reached for the issue to be accepted. Additionally, each agent may be given a different weight w_i (think for instance of a committee where the vote of the president counts more than the vote of a member) and each model v of an individual goal may have a different weight $\mu_{\gamma_i}(v)$. To simplify notation we omit the vector $\vec{q} = (q_1, \ldots, q_m)$, specifying the particular choice of thresholds for the issues, from the name of $TrSh^{\mu}$ when clear from context.

The general definition of threshold rules can be used to provide a first adaptation for goal-based voting of the classical issue-wise majority rule (May, 1952). Consider a $TrSh^{\mu}$ rule having $\mu_{\gamma_i}(v) = \frac{1}{|\text{Mod}(\gamma_i)|}$ and $w_i = 1$ for all $v \in \text{Mod}(\gamma_i)$ and for all $i \in \mathcal{N}$: we call *EQuota* such procedures, as they are inspired by *equal and even cumulative voting* (Campbell, 1954; Bhagat and Brickley, 1984). Namely, the goal of each agent is given a total weight of 1 (echoing the "one person, one vote" principle), which they equally distribute over all the models of their goal. In this way, agents whose goals have more models are not favored by the rule with respect to goals with less models.

The *equal and even majority rule EMaj* is formally defined[2] as follows:

$$EMaj(\mathbf{\Gamma})_j = 1 \;\; \textit{iff} \;\; \big(\sum_{i \in \mathcal{N}} \frac{m_{ij}^1}{|\text{Mod}(\gamma_i)|}\big) > \frac{n}{2}$$

[1] In this case d is the *drastic* distance, such that for two interpretations v and v', $d(v, v') = 0$ if $v = v'$ and $d(v, v') = 1$ otherwise.

[2] The *EMaj* rule has been initially defined by using $\lceil \frac{n+1}{2} \rceil$ as quota (Novaro et al., 2018), to stay close to the intuition of majority where a quota corresponds to a certain number of 'heads' for or against an issue. We can, however, let the quota be a real number as fractions of votes come from different agents.

for all $j \in \mathcal{I}$. Observe that the *EMaj* adaptation of the (strict) majority rule is thus biased towards rejection of an issue: when there is a tie in the sum of acceptances and rejections the issue is rejected. To overcome this issue, we present a second adaptation of majority where comparisons are directly made between acceptances and rejections for each issue $j \in \mathcal{I}$.

The *TrueMaj* goal-based voting rule is formally defined as:

$$TrueMaj(\mathbf{\Gamma}) = \Pi_{j \in \mathcal{I}} M(\mathbf{\Gamma})_j$$

where for each $j \in \mathcal{I}$:

$$M(\mathbf{\Gamma})_j = \begin{cases} \{x\} & \text{if } \sum_{i \in \mathcal{N}} \frac{m^x_{ij}}{|\text{Mod}(\gamma_i)|} > \sum_{i \in \mathcal{N}} \frac{m^{1-x}_{ij}}{|\text{Mod}(\gamma_i)|} \\ \{0,1\} & \text{otherwise} \end{cases}$$

Intuitively, *TrueMaj* calculates for each issue a sum of the ones and a sum of the zeroes in all the models of the individual goals, weighting each model as in *EQuota* rules. If the weighted sum of ones (respectively, zeroes) is higher than the weighted sum of zeroes (respectively, ones), the result for that issue is one (respectively, zero). In case the sums are equal, the rule outputs all interpretations with either a 0 or a 1 for that issue.[3]

We now introduce our third and final adaptation of majority to goal-based voting. Let $Maj(v_1, \dots, v_n)_j = 1$ if $\sum_{i \in \mathcal{N}} v_i(j) > \frac{n}{2}$, and $Maj(v_1, \dots, v_n)_j = 0$ otherwise. Note that the function *Maj* has a vector of interpretations as input, and it outputs an interpretation. We can now formally define the *two-step majority* rule:

$$2sMaj(\mathbf{\Gamma}) = \{Maj(EMaj(\gamma_1), \dots, EMaj(\gamma_n))\}.$$

The *2sMaj* rule applies majority twice: first on the models of the agents' individual goals, and then again on the result obtained in the first step. This rule belongs to a wider class of voting functions that apply first a rule on each individual goal, and then a second (possibly different) rule on the results obtained in the first step.

We may think that the three adaptations of majority to goal-based voting that we introduced here (i.e., *EMaj*, *TrueMaj* and *2sMaj*) are equivalent in the sense that they simply represent via three definitions the same underlying function. The following proposition proves that this is not the case:

Proposition 3.1. The rules *EMaj*, *TrueMaj* and *2sMaj* are different.

Proof. Since *EMaj*, *TrueMaj* and *2sMaj* are collection of functions for any n and m, it suffices to provide three goal-profiles $\mathbf{\Gamma}$, $\mathbf{\Gamma}'$, and $\mathbf{\Gamma}''$, one for each pair of rules, on which their outcomes differ. The three profiles are shown in Table 3.1.

Consider profile $\mathbf{\Gamma}$ and the rules *EMaj* and *TrueMaj*. For issues 2 and 3 we have that $\sum_{i \in \mathcal{N}} \frac{m^1_{i2}}{|\text{Mod}(\gamma_i)|} = 2$ and $\sum_{i \in \mathcal{N}} \frac{m^1_{i3}}{|\text{Mod}(\gamma_i)|} = 3$, respectively, are the sums of the votes coming from the agents in favour of the issues. Since $\sum_{i \in \mathcal{N}} \frac{m^1_{i2}}{|\text{Mod}(\gamma_i)|} = 2 > 1.5$ and $\sum_{i \in \mathcal{N}} \frac{m^1_{i3}}{|\text{Mod}(\gamma_i)|} = 3 > 1.5$ we have that $EMaj(\mathbf{\Gamma})_2 = 1$ and $EMaj(\mathbf{\Gamma})_3 = 1$, meaning that both issues are accepted in the outcome of *EMaj*. Similarly, as $\sum_{i \in \mathcal{N}} \frac{m^1_{i2}}{|\text{Mod}(\gamma_i)|} = 2 > 1 = \sum_{i \in \mathcal{N}} \frac{m^0_{i2}}{|\text{Mod}(\gamma_i)|}$ and $\sum_{i \in \mathcal{N}} \frac{m^1_{i3}}{|\text{Mod}(\gamma_i)|} = 3 > 0 = \sum_{i \in \mathcal{N}} \frac{m^0_{i3}}{|\text{Mod}(\gamma_i)|}$, we have that $TrueMaj(\mathbf{\Gamma})_2 = 1$ and $TrueMaj(\mathbf{\Gamma})_3 = 1$, meaning again that both issues are accepted

[3] Observe that *EMaj* can be also defined as *TrueMaj* with a $\{0\}$ instead of $\{0,1\}$ in the second line.

	$\mathbf{\Gamma}$	$\text{Mod}(\gamma_i)$	$\mathbf{\Gamma}'$	$\text{Mod}(\gamma_i)$	$\mathbf{\Gamma}''$	$\text{Mod}(\gamma_i)$
Agent 1	$1 \wedge \neg 2 \wedge 3$	(101)	$1 \wedge 2 \wedge 3$	(111)	$\neg(1 \vee 2 \vee 3)$	(000)
Agent 2	$\neg 1 \wedge 2 \wedge 3$	(011)	$2 \wedge (1 \to 3)$	(111) (011) (010)	$2 \wedge (1 \vee 3)$	(111) (110) (011)
Agent 3	$2 \wedge 3$	(111) (011)	$2 \wedge (1 \to 3)$	(111) (011) (010)	$2 \wedge (1 \vee 3)$	(111) (110) (011)
EMaj		(011)	—			(010)
TrueMaj		(011) (111)		(111)	—	
2sMaj	—			(011)		(111)

TABLE 3.1: Profiles $\mathbf{\Gamma}, \mathbf{\Gamma}', \mathbf{\Gamma}''$ on which *EMaj*, *TrueMaj* and *2sMaj* differ.

in the outcome of *TrueMaj*. For issue 1, however, we have that $\sum_{i \in \mathcal{N}} \frac{m^1_{i2}}{|\text{Mod}(\gamma_i)|} = 1 + 0 + \frac{1}{2} = 1.5$, and thus $EMaj(\mathbf{\Gamma})_1 = 0$, meaning that the issue is rejected, while $TrueMaj(\mathbf{\Gamma})_1 = (1, 1)$, meaning that there will be one interpretation where issue 1 has value 1 and one where it has value 0.

Consider $\mathbf{\Gamma}'$ and the rules *TrueMaj* and *2sMaj*. For issues 2 and 3, it is easy to see that both rules will accept them in the outcome. For issue 1, observe that $\sum_{i \in \mathcal{N}} \frac{m^1_{i1}}{|\text{Mod}(\gamma_i)|} = 1 + \frac{2}{3} > 1 + \frac{1}{3} = \sum_{i \in \mathcal{N}} \frac{m^0_{i1}}{|\text{Mod}(\gamma_i)|}$ and thus $TrueMaj(\mathbf{\Gamma}')_1 = 1$. For *2sMaj*, the first step of aggregation results in $((111), (011), (011))$ which, when focusing on issue 1, gives outcome $2sMaj(\mathbf{\Gamma}')_1 = 0$.

The outcomes for profile $\mathbf{\Gamma}''$ and rules *EMaj* and *2sMaj* can be easily obtained through a similar reasoning to the one given for profiles $\mathbf{\Gamma}$ and $\mathbf{\Gamma}'$. □

We briefly discuss here how non-binary issues could be handled in goal-based voting. Suppose that Ann, Barbara and Camille of Example 3.2 also have to choose a restaurant, and on a popular recommendation website restaurants are categorized by price as cheap, medium, and expensive. One way to include this non-binary issue would be to add three variables (one for each price range) and a constraint for the outcome of the rule. In fact, while we might allow the agents to express goals such as "I want either a cheap or a medium priced restaurant", the final choice must indicate a specific price category. Concretely, we would need to add three new binary variables: 4 (cheap restaurant), 5 (medium restaurant) and 6 (expensive restaurant), and a constraint IC= $(4 \wedge \neg 5 \wedge \neg 6) \vee (\neg 4 \wedge 5 \wedge \neg 6) \vee (\neg 4 \wedge \neg 5 \wedge 6)$ such that for all $\mathbf{\Gamma}$ we have $F(\mathbf{\Gamma}) \subseteq \text{Mod}(\text{IC})$. As independent rules are susceptible of returning a result which does not satisfy the constraint, the definitions of *EMaj*, *TrueMaj* and *2sMaj* would need to be modified to provide an alternative result in such cases.[4]

In this section we have introduced aggregation rules for propositional goals inspired by existing rules in voting, judgment aggregation and belief merging. We

[4]Similar problems have been discussed for binary aggregation. In particular, the problem of *lifting integrity constraints* (Grandi and Endriss, 2013) consists in looking for classes of aggregation rules (defined by the axioms they satisfy) ensuring that for all integrity constraints of a certain syntactical class, if the constraint is satisfied by all the agents it will also be satisfied by the outcome of the rule; and the *most representative voter rules* (Endriss and Grandi, 2014) guarantee to always return an outcome consistent with the constraint since they select the ballot of a voter in the profile.

have also seen how the three definitions of majority that we provided designate indeed three different functions. In the next section we will compare the proposed rules based on the properties they satisfy.

3.2 Axiomatics

A great variety of functions can be used to take a collective decision, as we have seen in Section 3.1.2. Depending on the specific voting situation at hand, one rule may be preferred to another: but how can agents compare different rules if they are solely given their definitions? In this section we will follow the tradition in social choice theory of providing a series of desirable properties (called *axioms*) adapted to the setting of goal-based voting, and checking whether the rules satisfy them or not. In line with Arrow's theorem (Arrow, 1951) we will see that certain desiderata are incompatible with one another and no rule can satisfy every axiom.

3.2.1 Resoluteness

Imagine a group of agents who needs to take a collective decision, and they submit their individual goals to a voting rule. It would be disappointing if the voting rule returned a large set of possible alternatives as the result: in some sense they would still need to make a choice over this result, as they will concretely execute a single plan. The first two axioms that we introduce deal precisely with this idea of *resoluteness* of a rule, in a stronger or weaker way.

Definition 3.2. A rule F is *resolute* (R) if and only if for all profiles $\mathbf{\Gamma}$ it is the case that $|F(\mathbf{\Gamma})| = 1$.

By definition *EMaj* and *2sMaj* are resolute since their outcome is constructed by uniquely deciding for 0 or 1 one issue at a time. On the other hand, neither *Approval* nor *TrueMaj* are resolute, as shown by the following example:

Example 3.4. Consider $\mathcal{N} = \{1, 2, 3\}$ and $\mathcal{I} = \{1, 2\}$. Let $\mathbf{\Gamma}$ be such that $\gamma_1 = 1 \wedge 2$, $\gamma_2 = 2$, $\gamma_3 = \neg 1 \wedge 2$. We have that $\mathit{Approval}(\mathbf{\Gamma}) = \mathit{TrueMaj}(\mathbf{\Gamma}) = \{(11), (01)\}$.

Such a requirement may seem too strict in situations as the one of Example 3.4: after all, the agents have the same opinion on the second issue but they are equally split about the first one. In this case we may argue that an abstention over the first issue can be acceptable in the outcome. We thus introduce a novel definition of *weak resoluteness* as follows:

Definition 3.3. A rule F is *weakly resolute* (WR) if and only if for all profiles $\mathbf{\Gamma}$ and all $j \in \mathcal{I}$ we have $F(\mathbf{\Gamma})_j \in \{(a, 0), (b, b), (0, c)\}$ for $a, b, c \in \mathbb{N}^+$.

Intuitively, a weakly resolute rule always gives either a definite answer on an issue or it abstains: thus, while the rule is not resolute its outcomes still have some structure. An equivalent formulation of the (WR) axiom is that for all profiles $\mathbf{\Gamma}$ there exists a (possibly partial) conjunction φ such that $F(\mathbf{\Gamma}) = \mathrm{Mod}(\varphi)$. We can easily see that all resolute rules are weakly resolute:

Proposition 3.2. If a rule F is resolute, then it is also weakly resolute.

Proof. By definition of resoluteness, we have that $F(\mathbf{\Gamma}) = \{(x_1 \dots x_m)\}$ where $x_j \in \{0,1\}$ for all $j \in \mathcal{I}$. Therefore, for each $j \in \mathcal{I}$ we have that $F(\mathbf{\Gamma})_j \in \{(1,0),(0,1)\}$, and therefore F is weakly resolute.[5] □

A consequence of Proposition 3.2 is that *EMaj* and *2sMaj* are thus weakly resolute. Intuitively, *TrueMaj* is also weakly resolute as for each issue it either provides a 0 or a 1 in the outcome, or it provides both a 0 and a 1.[6]

3.2.2 Anonymity, Duality and Neutrality

In many situations of collective decision making, for instance in political elections, we want that each vote (and voter) is treated in an equal and fair way. Namely, we want *anonymity* in the sense that it does not matter which agent submitted which vote to compute the result. In goal-based voting, we express this axiom as:

Definition 3.4. A rule F is *anonymous* (A) if and only if for any profile $\mathbf{\Gamma}$ and permutation $\sigma : \mathcal{N} \to \mathcal{N}$, we have that $F(\gamma_1, \dots, \gamma_n) = F(\gamma_{\sigma(1)}, \dots, \gamma_{\sigma(n)})$.

By permuting in any possible way the agents' goals, the result of an anonymous rule does not change. Observe that *EMaj*, *TrueMaj*, *2sMaj* and *Approval* are anonymous, while an instance of *TrSh* rules where at least two agents $i, \ell \in \mathcal{N}$ have different weights $w_i \neq w_\ell$ would not be anonymous.

The following axiom of *duality* is inspired by the neutrality property of May (1952) and by the domain-neutrality axiom in binary aggregation (Grandi and Endriss, 2011). Essentially, we want a rule to not be biased towards the acceptance or rejection of the issues. Let $\varphi[j \mapsto k]$ for $j, k \in \mathcal{I}$ be the replacement of each occurrence of k by j in φ. For instance, if $\varphi = p \wedge (q \vee r)$ we have that $\varphi[p \mapsto q] = p \wedge (p \vee r)$. We define duality as:

Definition 3.5. A rule is *dual* (D) if and only if for all profiles $\mathbf{\Gamma}$, $F(\overline{\gamma}_1, \dots, \overline{\gamma}_n) = \{(1 - v(1), \dots, 1 - v(m)) \mid v \in F(\mathbf{\Gamma})\}$ where $\overline{\gamma} = \gamma[\neg 1 \mapsto 1, \dots, \neg m \mapsto m]$.

Both *Approval* and *TrueMaj* are easily seen to be dual. On the other hand, *EMaj* and *2sMaj* do not satisfy this axiom due to their being resolute, as shown by the following example:

Example 3.5. Consider a profile $\mathbf{\Gamma}$ for two agents and two issues such that $\gamma_1 = 1 \wedge 2$ and $\gamma_2 = \neg 1 \wedge 2$. We have that $EMaj(\mathbf{\Gamma}) = 2sMaj(\mathbf{\Gamma}) = \{(01)\}$. Take now the goals $\overline{\gamma_1} = \neg 1 \wedge \neg 2$ and $\overline{\gamma_2} = 1 \wedge \neg 2$ (we simplify the double negation) and observe that for this new profile $\mathbf{\Gamma}'$ we have $EMaj(\mathbf{\Gamma}') = 2sMaj(\mathbf{\Gamma}') = \{(00)\}$, while according to duality we should have $\{(10)\}$.

Unfortunately, the properties of resoluteness, anonymity and duality cannot be simultaneously satisfied, as shown by the following theorem:[7]

[5] We can provide an alternative proof of this proposition based on the equivalent formulation of (WR) stated above. For all profiles $\mathbf{\Gamma}$ we have by definition that $F(\mathbf{\Gamma}) = \{(x_1 \dots x_m)\}$ where $x_j \in \{0,1\}$ for $j \in \mathcal{I}$. Then, it suffices to consider the following conjunction φ:

$$\varphi = \bigwedge_{\substack{j \in \mathcal{I} \\ x_j = 1}} j \wedge \bigwedge_{\substack{j \in \mathcal{I} \\ x_j = 0}} \neg j.$$

[6] We will see in this section a proof of this fact, i.e., that *TrueMaj* is weakly resolute, as a consequence of the rule satisfying the axiom of independence.

[7] A related result in social choice theory states that there exists no resolute, anonymous, and neutral voting procedure for 2 alternatives and an even number of voters (see Moulin, 1983).

Theorem 3.1. There is no resolute rule F satisfying both anonymity and duality.

Proof. Consider a rule F and suppose towards a contradiction that F is resolute, anonymous and dual. Take profile $\mathbf{\Gamma}$ for $\mathcal{N} = \{1,2\}$ and $\mathcal{I} = \{1,2\}$ where $\gamma_1 = 1 \wedge \neg 2$ and $\gamma_2 = \neg 1 \wedge 2$. By anonymity of F, for profile $\mathbf{\Gamma}' = (\gamma_2, \gamma_1)$ we have $F(\mathbf{\Gamma}) = F(\mathbf{\Gamma}')$. Since F is resolute, $F(\mathbf{\Gamma}) = \{(x,y)\}$, for $x,y \in \{0,1\}$, and thus $F(\mathbf{\Gamma}') = \{(x,y)\}$. However, note that $\gamma_1 = \overline{\gamma_2}$ and $\gamma_2 = \overline{\gamma_1}$. Hence, $\mathbf{\Gamma}' = \overline{\mathbf{\Gamma}}$ and by duality we must have $F(\mathbf{\Gamma}') = \{(1-x, 1-y)\}$, bringing us a contradiction. □

We have seen how to ensure that agents are treated equally (anonymity) and that the decision for each issue is not biased towards acceptance or rejection (duality). We may also want to ensure that issues are treated in an equal way with respect to one another. To this purpose, we introduce the axiom of *neutrality*, which is inspired by its voting counterpart — stating that if two alternatives are swapped in every ballot, they should be swapped in the outcome of the social choice function (see Zwicker, 2016). An analogous property has been introduced by Endriss and Grandi (2017) for graph aggregation under the name of *permutation-neutrality*. We can now define *neutrality* as follows:

Definition 3.6. A rule F is *neutral* (N) if for all profiles $\mathbf{\Gamma}$ and permutations $\sigma : \mathcal{I} \to \mathcal{I}$, we get $F(\gamma_1^\sigma, \dots, \gamma_n^\sigma) = \{(v(\sigma(1)) \dots v(\sigma(m))) \mid v \in F(\mathbf{\Gamma})\}$ where we have that $\gamma_i^\sigma = \gamma_i[1 \mapsto \sigma(1), \dots, m \mapsto \sigma(m)]$.

The following example shows that *EQuota* rules, and thus more generally *TrSh* rules, are not neutral when the quotas for two issues differ:

Example 3.6. Let $\mathbf{\Gamma}$ be a profile for agents in $\mathcal{N} = \{1,2,3\}$ and issues in $\mathcal{I} = \{1,2\}$ such that $\gamma_1 = \gamma_2 = 1 \wedge 2$ and $\gamma_3 = \neg 1 \wedge 2$. Consider now the *EQuota* rule such that $q_1 = 2$ and $q_2 = 3$: we have that $EQuota(\mathbf{\Gamma}) = \{(11)\}$. Let σ be a permutation such that $\sigma(1) = 2$ and $\sigma(2) = 1$. We then have that $\gamma_1^\sigma = \gamma_2^\sigma = 2 \wedge 1$ and $\gamma_3^\sigma = \neg 2 \wedge 1$, and for $\mathbf{\Gamma}^\sigma = (\gamma_1^\sigma, \gamma_2^\sigma, \gamma_3^\sigma)$ we have that $EQuota(\mathbf{\Gamma}^\sigma) = \{(10)\}$ which is different from the permutation of (11) that should be the result according to neutrality.

As far as the other rules are concerned, *Approval* is neutral as it considers the models of agents' goals in their entirety and it simply permutes their values, and both *TrueMaj* and *2sMaj* are neutral since they have the same quota for all issues.

A different formulation of neutrality is the one commonly used in binary and judgment aggregation (Grandi and Endriss, 2011; Endriss, 2016), also similarly introduced for graph aggregation (Endriss and Grandi, 2017) which intuitively states that if two issues are treated in the same way in a profile, their respective outcomes on that profile should be the same. This definition however is not so easily adaptable to goal-based voting: consider a profile with a goal $\gamma = 1 \vee 2$, whose models are $\text{Mod}(\gamma) = \{(11), (10), (01)\}$. Should an axiom consider issues a and b as treated equivalently in γ (and thus in $\mathbf{\Gamma}$) or should it also impose that in each model they have the same truth-value at the same time? Depending on the answer, we may (or not) require the outcomes for a and b to be the same. We leave a deeper analysis of this axiom for future research.

3.2.3 Unanimity and Groundedness

The property of *unanimity* imposes to the outcome of a rule on a profile to agree with the agents if they agree with each other. In goal-based voting we can separate this property into two different axioms. The first focuses on the case where the unanimous choice of the agents for a specific issue is respected in the outcome, and it is a straightforward generalization of the unanimity axiom in binary aggregation (Grandi and Endriss, 2011). It is also closely related to the *unanimous* consensus class in voting (Elkind et al., 2010), stating that a candidate should win if it is ranked first by all voters in an election. We formally define unanimity as:

Definition 3.7. A rule F is *unanimous* (U) if for all profiles $\mathbf{\Gamma}$ and for all $j \in \mathcal{I}$, if $m^x_{ij} = 0$ for all $i \in \mathcal{N}$ then $F(\mathbf{\Gamma})_j = 1 - x$ for $x \in \{0, 1\}$.

It is fairly easy to see that if all agents accept or reject unanimously an issue the outcomes of *EMaj*, *TrueMaj* and *2sMaj* will agree with the profile. The *Approval* rule also satisfies it, since it will choose one of the models of the agents' goals (which by definition are all unanimous on that issue). For *TrSh* rules, the axiom is not satisfied for certain (degenerate) choices of quotas and weights for the agents' goals, as shown by the following example:

Example 3.7. Consider $\mathbf{\Gamma}$ for three agents and two issues such that $\gamma_1 = \gamma_2 = \gamma_3 = 1 \land 2$, where $q_1 = q_2 = 3$ are the quotas, and $\mu_{1\land 2}(11) = 0.3$ are the weights for the agents' goals. If $w_i = 1$ for all $i \in \mathcal{N}$ are the weights of the agents, we have $TrSh(\mathbf{\Gamma}) = \{(00)\}$. Similar examples can be found for agents' weights lower than 1.

The second type of unanimity captures the idea that if the agents all agree on some models of their goals, these models should be the outcome. More formally, we define *model-unanimity*[8] as follows:

Definition 3.8. A rule F is *model-unanimous* (MU) if on all profiles $\mathbf{\Gamma}$ we have $v \in F(\mathbf{\Gamma})$ if and only if $v \in \text{Mod}(\gamma_i)$ for all $i \in \mathcal{N}$.

The (MU) axiom can be alternatively formulated as stating that if for all $\mathbf{\Gamma}$ we have $\text{Mod}(\bigwedge_{i\in\mathcal{N}} \gamma_i) \neq \emptyset$ then we must have that $F(\mathbf{\Gamma}) = \text{Mod}(\bigwedge_{i\in\mathcal{N}} \gamma_i)$. Observe that in Definition 3.8, the right-to-left direction ensures that if agents are unanimous about a model it will be accepted in the outcome (but the outcome may include other models which are not unanimously supported), while the left-to-right direction ensures that all the models in the outcome are unanimously accepted by the agents (but the rule may be excluding from the outcome some models that all agents accept). Model-unanimity is also known as the (IC2) postulate in belief merging (Everaere et al., 2015) and since *Approval* can be expressed as an IC merging operator (see Section 3.1.2) it therefore satisfies model-unanimity. The following example shows that *EMaj*, *TrueMaj* and *2sMaj* do not satisfy this type of unanimity:

Example 3.8. Let $\mathbf{\Gamma}$ be a profile for three agents and three issues where the agents' goals are such that $\text{Mod}(\gamma_1) = \{(000), (101), (110)\}$, $\text{Mod}(\gamma_2) = \{(000), (111), (100)\}$ and $\text{Mod}(\gamma_3) = \{(000), (101), (110)\}$. Even though $\text{Mod}(\gamma_1 \land \gamma_2 \land \gamma_3) = \{(000)\}$, we have that $EMaj(\mathbf{\Gamma})$ = $TrueMaj(\mathbf{\Gamma})$ = $2sMaj(\mathbf{\Gamma}) = \{(100)\}$.

Unanimity does not imply model-unanimity since *EMaj*, *TrueMaj* and *2sMaj* satisfy the first but not the second. Interestingly, the opposite is also not the case:

Proposition 3.3. There exists a rule F that is model-unanimous and not unanimous.

[8]The rule $Conj_v$ defined in previous work (Novaro et al., 2018) captured precisely the intuition behind this axiom, in that its output was defined as the models of the conjunction of the agents' goals if they were mutually consistent, and a default option otherwise.

Proof. Let F be defined as such that $F(\mathbf{\Gamma}) = \text{Mod}(\bigwedge_{i\in\mathcal{N}} \gamma_i)$ if $\text{Mod}(\bigwedge_{i\in\mathcal{N}} \gamma_i) \neq \emptyset$, and $\{0\}^m$ otherwise. By definition F is model-unanimous. Consider the profile $\mathbf{\Gamma}$ for two agents and three issues where $\gamma_1 = 1 \wedge \neg 2$ and $\gamma_2 = 1 \wedge 2$. Since $\text{Mod}(\gamma_1 \wedge \gamma_2) = \emptyset$, we have that $F(\mathbf{\Gamma}) = \{(000)\}$. However, $m^0_{i1} = 0$ for all $i \in \mathcal{N}$ and yet $F(\mathbf{\Gamma})_1 = 0$. □

Observe that (MU) implies in particular that $F(\gamma, \dots, \gamma) = \text{Mod}(\gamma)$, which corresponds to the *strongly unanimous* consensus class in voting (Elkind et al., 2010).

The axioms of unanimity that we just saw require that the outcome agrees with the agents if they *all* agree (on issues or models). The next axiom that we introduce states that the outcome of a rule should be supported by *at least* one of the agents in the profile. We formally define *groundedness* as follows:

Definition 3.9. A rule F is *grounded* (G) if $F(\mathbf{\Gamma}) \subseteq \text{Mod}(\bigvee_{i\in\mathcal{N}} \gamma_i)$.

By definition, *Approval* is grounded, since it outputs a model approved by at least one agent in the profile. On the other hand, the same is not true for the majority rules *EMaj*, *TrueMaj* and *2sMaj*, as shown by the following result:

Proposition 3.4. *EMaj*, *TrueMaj* and *2sMaj* are not grounded.

Proof. Consider a profile $\mathbf{\Gamma}$ for three agents and issues where $\text{Mod}(\gamma_1) = \{(111)\}$, $\text{Mod}(\gamma_2) = \{(010)\}$ and $\text{Mod}(\gamma_3) = \{(001)\}$. The rules *EMaj*, *TrueMaj* and *2sMaj* all return $\{(011)\}$, and since $(011) \notin \text{Mod}(\bigvee_{i\in\mathcal{N}} \gamma_i)$ groundedness is not satisfied. □

Proposition 3.4 implies that the three majority rules do not guarantee that the collective choice will satisfy the goal of at least one agent. In some cases, however, this can be seen as the rules finding a compromise issue-by-issue between the conflicting views of the agents.

Since $\text{Mod}(\bigwedge_{i\in\mathcal{N}} \gamma_i) \subseteq \text{Mod}(\bigvee_{i\in\mathcal{N}} \gamma_i)$ we may think that model-unanimity implies groundedness. The following result however shows that the two axioms are not related:

Proposition 3.5. There exists rules F and F' such that F is grounded and not model-unanimous, while F' is model-unanimous and not grounded.

Proof. Consider F such that for all profiles $\mathbf{\Gamma}$, if $\text{Mod}(\bigvee_{i\in\mathcal{N}} \gamma_i) = \text{Mod}(\bigwedge_{i\in\mathcal{N}} \gamma_i)$ then $F(\mathbf{\Gamma}) = \text{Mod}(\bigvee_{i\in\mathcal{N}} \gamma_i)$ and $F(\mathbf{\Gamma}) = \text{Mod}(\bigvee_{i\in\mathcal{N}} \gamma_i) \setminus \text{Mod}(\bigwedge_{i\in\mathcal{N}} \gamma_i)$ otherwise. For F' it suffices to consider the same rule defined in the proof of Proposition 3.3. □

An analogous notion of groundedness has been previously defined for the aggregation of ontologies as well (Porello and Endriss, 2014).

3.2.4 Monotonicity, Egalitarianism and Independence

In judgment aggregation, the axiom of *monotonicity* informally states that the acceptance decision of an issue should not be reversed if more agents endorse it (List, 2012; Grandi and Endriss, 2011). Similarly, we say that a goal-based voting rule is *monotonic* if adding support for an issue j when the current result for j is equally irresolute or favoring acceptance, results in an outcome strictly favoring acceptance for j. The formal definition that we provide applies only to *comparable* profiles $\mathbf{\Gamma}$ and $\mathbf{\Gamma}'$ for which $|\text{Mod}(\gamma_i)| = |\text{Mod}(\gamma'_i)|$ for all $i \in \mathcal{N}$:

Definition 3.10. A rule F satisfies *monotonicity* (M) if for all comparable profiles $\mathbf{\Gamma} = (\gamma_1, \dots, \gamma_i, \dots, \gamma_n)$ and $\mathbf{\Gamma}^\star = (\gamma_1, \dots, \gamma_i^\star, \dots, \gamma_n)$, for all $j \in \mathcal{I}$ and $i \in \mathcal{N}$, if $m^{1\star}_{ij} > m^1_{ij}$, then $F(\mathbf{\Gamma})^1_j \geq F(\mathbf{\Gamma})^0_j$ implies $F(\mathbf{\Gamma}^\star)^1_j > F(\mathbf{\Gamma}^\star)^0_j$.

The rules *EMaj* and *2sMaj* are monotonic, since they have a threshold of acceptance for each issue and once the issue is accepted, adding support only confirms the current result. The same reasoning applies to *TrueMaj*, with the additional observation that a tie in the outcome only comes from a tie in the profile, and if more support is added (deleted) the issue is decided towards acceptance (rejection). The *Approval* rule however does not satisfy this axiom, as the following example shows:

Example 3.9. Let $\mathbf{\Gamma}$ be defined for three agents and issues with $\text{Mod}(\gamma_2) = \text{Mod}(\gamma_3) = \{(101), (001)\}$ and $\text{Mod}(\gamma_1) = \{(101), (111), (010), (000)\}$. We have $\mathit{Approval}(\mathbf{\Gamma}) = \{(101)\}$. Take now a goal $\gamma_1^\star$ with $\text{Mod}(\gamma_1^\star) = \{(110), (111), (001), (100)\}$. Since $m_{11}^1 = 2$ and $m_{11}^{1\star} = 3$ and $\mathit{Approval}(\mathbf{\Gamma})_1^1 = 1 > 0 = \mathit{Approval}(\mathbf{\Gamma})_1^0$, we should have $\mathit{Approval}(\mathbf{\Gamma}^\star)_1^1 > \mathit{Approval}(\mathbf{\Gamma}^\star)_1^0$. However, as $\mathit{Approval}(\mathbf{\Gamma}^\star) = \{(001)\}$ the axiom is not satisfied.

As for (MU), a *model-wise* monotonicity could be defined as follows: for every profile $\mathbf{\Gamma}$, if $v \in F(\mathbf{\Gamma})$ and there is $i \in \mathcal{N}$ such that $v \notin \text{Mod}(\gamma_i)$ and $v \in \text{Mod}(\gamma_i')$ then $v \in F(\gamma_1, \dots, \gamma_i', \dots, \gamma_n)$; and if $v \notin F(\mathbf{\Gamma})$ and there is $i \in \mathcal{N}$ such that $v \in \text{Mod}(\gamma_i)$ and $v \notin \text{Mod}(\gamma_i')$ then $v \notin F(\gamma_1, \dots, \gamma_i', \dots, \gamma_n)$.[9]

The next axiom that we introduce formalizes the "one person, one vote" principle, while at the same time ensuring that a rule is giving an *equal* weight to the models of each agent's goal. Namely, if we take a profile $\mathbf{\Gamma}$ in which agents have a varying number of models for their goals, and we transform it into a bigger profile $\mathbf{\Gamma}'$ where a set of agents (proportional to how many models a goal had in $\mathbf{\Gamma}$) has a goal satisfied by exactly one of the models in $\mathbf{\Gamma}$, the results in $\mathbf{\Gamma}$ and $\mathbf{\Gamma}'$ should be the same. Formally:

Definition 3.11. A rule F is *egalitarian* (E) if for all $\mathbf{\Gamma}$, on the profile $\mathbf{\Gamma}'$ such that:

(a) $|\mathcal{N}'| = |\mathcal{N}| \cdot \text{lcm}(|\text{Mod}(\gamma_1)|, \dots, |\text{Mod}(\gamma_n)|)$, and

(b) for $v \in \text{Mod}(\gamma_i)$ and $i \in \mathcal{N}$, $\frac{|\mathcal{N}'|}{|\mathcal{N}| \cdot |\text{Mod}(\gamma_i)|}$ agents have $\gamma' = \bigwedge_{\substack{j \in \mathcal{I} \\ v(j)=1}} j \wedge \bigwedge_{\substack{\ell \in \mathcal{I} \\ v(j)=0}} \neg \ell$

it holds that $F(\mathbf{\Gamma}) = F(\mathbf{\Gamma}')$.

In an egalitarian rule it is thus possible to turn every profile into an equivalent profile (with respect to the outcome) where each agent submits a goal in the form of a complete conjunction. From their definitions, we see that all *EQuota* rules and *TrueMaj* satisfy the axiom. It is not satisfied by some *TrSh* rules (where the weight of the models is unequal), by *2sMaj* and by *Approval*, as per the following example:

Example 3.10. Consider a profile $\mathbf{\Gamma}$ for three agents and issues such that $\text{Mod}(\gamma_1) = \{(111)\}$, $\text{Mod}(\gamma_2) = \{(110), (000), (001)\}$ and $\text{Mod}(\gamma_3) = \{(110), (011), (010)\}$. We have that $\mathit{Approval}(\mathbf{\Gamma}) = \{(110)\}$ and $\mathit{2sMaj}(\mathbf{\Gamma}) = \{(010)\}$. Consider now a new profile $\mathbf{\Gamma}'$ constructed according to Definition 3.11: it is a profile for three issues and nine agents such that $\text{Mod}(\gamma_1) = \text{Mod}(\gamma_2) = \text{Mod}(\gamma_3) = \{(111)\}$, $\text{Mod}(\gamma_4) = \{(110)\}$, $\text{Mod}(\gamma_5) = \{(000)\}$, $\text{Mod}(\gamma_6) = \{(001)\}$, $\text{Mod}(\gamma_7) = \{(110)\}$, $\text{Mod}(\gamma_8) = \{(011)\}$ and $\text{Mod}(\gamma_9) = \{(010)\}$. We have that $\mathit{Approval}(\mathbf{\Gamma}') = \mathit{2sMaj}(\mathbf{\Gamma}') = \{(111)\}$.

Consider now a *TrSh* rule such that $q_j = \frac{n}{2}$ for all $j \in \mathcal{I}$ and the weight function μ is defined as follows: $\mu_{\gamma_2}(110) = \mu_{\gamma_3}(110) = \frac{2}{3}$, $\mu_{\gamma_2}(000) = \mu_{\gamma_2}(001) = \mu_{\gamma_3}(011) =$

[9] Moreover, Novaro et al. (2018) also defined the axiom of *positive responsiveness* as a stronger version of monotonicity. Namely, a rule F is *positively responsive* (PR) if for all comparable profiles $\mathbf{\Gamma} = (\gamma_1, \dots, \gamma_i, \dots, \gamma_n)$ and $\mathbf{\Gamma}^\star = (\gamma_1, \dots, \gamma_i^\star, \dots, \gamma_n)$, for all $j \in \mathcal{I}$ and $i \in \mathcal{N}$, if $m_{ij}^{x\star} > m_{ij}^x$ for $x \in \{0,1\}$, then $F(\mathbf{\Gamma})_j^x \geq F(\mathbf{\Gamma})_j^{1-x}$ implies $F(\mathbf{\Gamma}^\star)_j^x > F(\mathbf{\Gamma}^\star)_j^{1-x}$.

$\mu_{\gamma_3}(010) = \frac{1}{6}$, and $\mu_\gamma(v) = 1$ for any other γ and v. We have that $TrSh(\mathbf{\Gamma}) = \{(110)\}$ while $TrSh(\mathbf{\Gamma}') = \{(111)\}$.

The final axiom that we present is related to a controversial yet well-known property used in both characterization and impossibility results in aggregation theory (List, 2012; Brandt et al., 2016). It states that the decision of acceptance or rejection for each issue j should be taken by looking uniquely at the acceptances or rejections that j received in the profile. First, let $\mathcal{D}_m = \{(a, b) \mid a, b \in \mathbb{N} \text{ and } a + b \leq 2^m\}$ and $\mathcal{C} = \{\{0\}, \{1\}, \{0, 1\}\}$. We formalize the principle of *independence* in goal-based voting as follows:

Definition 3.12. A rule F is *independent* (I) if there are functions $f_j : \mathcal{D}^n_m \to \mathcal{C}$ for $j \in \mathcal{I}$ and $n, m \in \mathbb{N}^+$ such that for all profiles $\mathbf{\Gamma}$ we have $F(\mathbf{\Gamma}) = \Pi_{j\in\mathcal{I}} f_j(m_1(j), \dots, m_n(j))$.

From the definitions we see that *TrSh*, *EQuota* and *TrueMaj* rules are independent, since they construct the outcome issue-by-issue. The *Approval* rule is not independent, as shown by the following example:

Example 3.11. Consider a profile $\mathbf{\Gamma}$ for three agents and three issues such that $\gamma_1 = 1 \land 2 \land 3$, $\gamma_2 = 1 \land 2 \land \neg 3$ and $\gamma_3 = 1 \land \neg 2 \land 3$. The outcome is $Approval(\mathbf{\Gamma}) = \{(111), (110), (101)\}$. From the outcome on issues 2 and 3 we see that it does not correspond to the cartesian product of functions with codomain $\mathcal{C} = \{\{0\}, \{1\}, \{0, 1\}\}$.

We conclude this section with another general result relating our axioms, namely those of independence and weak resoluteness:

Theorem 3.2. Each independent goal-based voting rule is weakly resolute.

Proof. Consider an arbitrary profile $\mathbf{\Gamma}$ and the outcome of an independent rule $F(\mathbf{\Gamma})$. As F is independent, we have $F(\mathbf{\Gamma}) = \Pi_{j\in\mathcal{I}} f(m_1(j), \dots, m_n(j))$, where each $m_x(j) \in \{\{0\}, \{1\}, \{0, 1\}\}$. We want to show that F is weakly resolute. We construct a conjunction ψ as follows: for all $j \in \mathcal{I}$, if $f(m_1(j), \dots, m_n(j)) = \{0\}$ add conjunct $\neg j$ to ψ; if $f(m_1(j), \dots, m_n(j)) = \{1\}$ add conjunct j to ψ; if $f(m_1(j), \dots, m_n(j)) = \{0, 1\}$ skip. Observe that for all $v \in \text{Mod}(\psi)$ and for all $j \in \mathcal{I}$ appearing as conjuncts in ψ, we have $v(j) = 1$ for a positive literal j, and $v(j) = 0$ for a negative literal $\neg j$. Moreover, for all $k \in \mathcal{I}$ which did not appear in ψ we have any possible combination of truth values. Therefore, $\text{Mod}(\psi) = F(\mathbf{\Gamma})$. □

The converse of Theorem 3.2 does not hold: consider an F that returns $\{(11 \dots 1)\}$ if in at least one $v \in \text{Mod}(\gamma_1)$ issue 1 is true, and returns $\{(00 \dots 0)\}$ otherwise.

3.2.5 Characterizing Goal-Based Majority

A seminal result in the characterization of aggregation rules is due to May (1952), who gave an axiomatization of the majority rule in the context of voting over two alternatives. Part of the appeal of the majority rule is its omnipresence in everyday decisions as well as its simple definition. We build on May's result and similar results in judgment aggregation (Endriss, 2016) to provide a characterization of our *TrueMaj* rule in the following theorem:[10]

Theorem 3.3. A rule F satisfies (E), (I), (N), (A), (M), (U) and (D) if and only if it is *TrueMaj*.

[10] A previous version of this result used the stronger axiom of positive responsiveness in place of monotonicity (Novaro et al., 2018).

Proof. The right-to-left direction follows from the discussion provided in Section 3.2. For the left-to-right direction, consider a rule F which satisfies this list of axioms. Let $\mathbf{\Gamma}$ be an arbitrary profile over n voters and m issues. Since F satisfies (E), we can construct a profile $\mathbf{\Gamma}'$ for m issues and $n' = |\mathcal{N}| \cdot \text{lcm}(|\text{Mod}(\gamma_1)|, \ldots, |\text{Mod}(\gamma_n)|)$ agents where each agent in $\mathcal{N}'$ submits a goal having a single model (a model of one of the goals in $\mathbf{\Gamma}$). Importantly, we know that $v \in F(\mathbf{\Gamma})$ if and only if $v \in F(\mathbf{\Gamma}')$.

Therefore, without loss of generality we can restrict our attention only to profiles where each agent submits a goal corresponding to a complete conjunction (having thus a unique model). We denote by $\mathcal{G}^\wedge$ such a set of profiles: in particular we have that $\mathbf{\Gamma}' \in \mathcal{G}^\wedge$. We now have to show that $F(\mathbf{\Gamma}') = \mathit{TrueMaj}(\mathbf{\Gamma}')$.

Since F satisfies (I), we know that there are functions $f_1, \ldots, f_m$ such that $F(\mathbf{\Gamma}') = \Pi_{j \in \mathcal{I}} f_j(m_1(j), \ldots, m_n(j))$. Moreover, as $\mathbf{\Gamma}' \in \mathcal{G}^\wedge$ we have that $m_i(j) \in \{(0,1), (1,0)\}$ for all $i \in \mathcal{N}'$ and $j \in \mathcal{I}$. Namely, since each goal has a unique model in $\mathbf{\Gamma}'$, either an issue has value 1 or 0 in it. Hence, each f_j on profiles in $\mathcal{G}^\wedge$ can equivalently be seen as a function from $\{0,1\}^n$ to $\{\{0\}, \{1\}, \{0,1\}\} = \mathcal{C}$. By the fact that F satisfies also the (N) axiom, we have that $f_1 = \cdots = f_m$ must be the case: i.e., the same function f is applied to all issues. The axiom (A) tell us that any permutation of the goals of the agents in $\mathbf{\Gamma}'$ gives the same result $F(\mathbf{\Gamma}')$.

Combining the (A), (I) and (N) axioms, we can see that only the number of ones (or zeroes) counts in determining the outcome of f, and not their position in the input. Hence, we can consider f as a function $f : \{0, \ldots, n\} \to \mathcal{C}$ from number of agents (say, all those that assign 1 to the issue) to $\{\{0\}, \{1\}, \{0,1\}\}$.

Given that F satisfies (U), we know that on a profile $\mathbf{\Gamma}^+$ such that for all i we have $m^0_{ij} = 0$ we have $F(\mathbf{\Gamma}^+)^0_j = 0$, i.e., $v(j) = 1$ for all $v \in F(\mathbf{\Gamma}^+)$. Consequently, we have that $f(n) = \{1\}$ and similarly that $f(0) = \{0\}$.

Let now s be a sequence of $\mathcal{G}^\wedge$-profiles $\mathbf{\Gamma}^- = \mathbf{\Gamma}^0, \mathbf{\Gamma}^1, \ldots, \mathbf{\Gamma}^n = \mathbf{\Gamma}^+$ where exactly one agent i changes her goal γ_i from profile $\mathbf{\Gamma}^k$, in which $m^1_{ij} = 0$, to profile $\mathbf{\Gamma}^{k+1}$, in which $m^1_{ij} = 1$. Namely, each agent i replaces $\neg j$ with j in their goal γ_i. By the (I) axiom and the definition of cartesian product, for any $\mathbf{\Gamma}$ and j we have that $F(\mathbf{\Gamma})_j$ is either equal to $(a, 0)$, (b, b) or $(0, c)$ for $a, b, c \in \mathbb{N}$. In fact, the interpretations in the outcome $F(\mathbf{\Gamma})$ can either have only zeroes for j $(a, 0)$, as many zeroes as ones for j (b, b), or only ones for j $(0, c)$. Since F satisfies (M), we have that the outcome on the $\mathbf{\Gamma}^k$ profiles in s can only possibly change:

- from $(a, 0)$ to (b, b) or $(0, c)$;
- from (b, b) to $(0, c)$.

Considering that $f(0) = \{0\}$ and $f(n) = \{1\}$, this means that there is some number q such that $f(0) = \{0\}, \ldots, f(q-1) = \{0\}$, $f(q) = \{0,1\}$ or $f(q) = \{1\}$, and $f(q+1) = \{1\}, \ldots, f(n) = \{1\}$. We now show that for n even we have $q = \frac{n}{2}$ and $f(q) = \{0,1\}$, while for n odd we have $q = \frac{n+1}{2}$ and $f(q) = \{1\}$.

For n even, consider the profile $\mathbf{\Gamma}^\ell$ were exactly half of the agents accept j and half reject it. If $F(\mathbf{\Gamma}^\ell)_j = (a, 0)$ or $(0, c)$, meaning that $f(\frac{n}{2}) = \{0\}$ or $f(\frac{n}{2}) = \{1\}$ respectively, the outcome would have to be reversed for $F(\overline{\mathbf{\Gamma}}^\ell)_j$ since F satisfies (D). However, both in $\mathbf{\Gamma}^\ell$ and $\overline{\mathbf{\Gamma}}^\ell$ the decision for j is determined by $f(\frac{n}{2})$, which therefore has to be equal to $\{0,1\}$.

For n odd, suppose for reductio that $q < \frac{n+1}{2}$ and consider a profile $\mathbf{\Gamma}$ where there are exactly q agents accepting j. From the discussion above, we have by definition of q that the outcome must be $F(\mathbf{\Gamma})_j = (0, c)$, i.e., issue j is accepted. Consider the profile $\overline{\mathbf{\Gamma}}$: by definition, all the agents that accept j in $\mathbf{\Gamma}$ reject it in $\overline{\mathbf{\Gamma}}$: that is,

$|\{i \mid m_i(j) = (0,1) \text{ for } \gamma_i \in \mathbf{\Gamma}\}| = |\{i \mid m_i(j) = (1,0) \text{ for } \gamma_i \in \overline{\mathbf{\Gamma}}\}| = q < \frac{n+1}{2}$. Thus, $|\{i \mid m_i(j) = (0,1) \text{ for } \gamma_i \in \overline{\mathbf{\Gamma}}\}| \geq \frac{n+1}{2} > q$. From definition of q we should have $F(\overline{\mathbf{\Gamma}})_j = (0, c')$ which however contradicts axiom (D). Suppose for reductio that $q > \frac{n+1}{2}$ and consider a profile $\mathbf{\Gamma}$ where $\frac{n+1}{2} \leq |\{i \mid m_i(j) = (0,1) \text{ for } \gamma_i \in \mathbf{\Gamma}\}| < q$. Again from definition of q we have that $F(\mathbf{\Gamma})_j = (a, 0)$ (i.e., issue j is rejected), and the same is true for the profile $\overline{\mathbf{\Gamma}}$ contradicting (D). Therefore, we have $q = \frac{n+1}{2}$.

Concluding, F is a rule defined as the cartesian product of the decisions for each $j \in \mathcal{I}$ taken by the same function $f : \{0, \dots, n\} \to \mathcal{C}$, with $f(k) = \{0,1\}$ for n even and $k = \frac{n}{2}$, $f(k) = \{0\}$ for $\sum_{i \in \mathcal{N}'} m^x_{ij} > \sum_{i \in \mathcal{N}'} m^{1-x}_{ij}$ for $x \in \{0,1\}$, corresponding to the definition of *TrueMaj*. As $\mathbf{\Gamma}$ was an arbitrary profile and *TrueMaj* satisfies axiom (E) this concludes the proof. □

The rules *EMaj* and *2sMaj* are based on analogous intuitions as *TrueMaj*. However, observe that *EMaj* has a bias towards rejection of the issues and thus it does not satisfy the (D) axiom. Without the (D) axiom we obtain (uniform) *EQuota* rules, of which *EMaj* is an instance. On the other hand, the rule *2sMaj* does not satisfy the egalitarian axiom (E), as we have seen with Example 3.10. Hence, the proof of Theorem 3.3 could not be directly modified to obtain a characterization result for *2sMaj*. The axioms used in the proof of Theorem 3.3 will be compared with their judgment aggregation counterparts in Section 3.4.2.

As we have seen, a general tension exists between the decisiveness or resoluteness of the rule — i.e., its ability to take a unique decision in all (or most) situations, essential in the development of decision-aid tools — and fairness requirements with respect to issues and individuals. The *TrueMaj* rule seems to strike a fair balance between these requirements.

3.3 Computational Complexity

In this section we study the computational complexity of a central problem in voting: that of determining the outcome of a rule on a given profile, i.e., the *winner determination* problem. We will see that while propositional logic helps the agents express compactly their goals, it carries over an increase in complexity with respect to the standard setting of voting. We will assume familiarity with the field of computational complexity: for an introduction to the topic and more details, see the manual by Arora and Barak (2009).

3.3.1 Definitions

In the literature on judgment aggregation we can find two definitions for the winner determination problem, in case the rule is resolute or not (Endriss et al., 2012; Baumeister et al., 2015; de Haan and Slavkovik, 2017). For both types of rules, the underlying idea is to construct an outcome (that in case of resolute rules will be unique) by finding the value of one issue at a time.

We adapt the definition of the winner determination problem for a resolute rule F in goal-based voting as follows:

WinDet(F)
Input: profile $\mathbf{\Gamma}$, issue $j \in \mathcal{I}$
Question: Is it the case that $F(\mathbf{\Gamma})_j = 1$?

As F is resolute, the outcome $F(\boldsymbol{\Gamma})$ on any profile will be a singleton. Hence, given a profile $\boldsymbol{\Gamma}$ in order to construct the outcome of F on $\boldsymbol{\Gamma}$ it suffices to ask the WINDET(F) question m times, once for every issue $j \in \mathcal{I}$.

Next, we present the definition of the winner determination problem for a non-resolute rule F (that can potentially be weakly resolute):

WINDET$^\star(F)$
Input: profile $\boldsymbol{\Gamma}$, subset $S \subseteq \mathcal{I}$, partial interpretation $\rho : S \to \{0, 1\}$
Question: Is there a $v \in F(\boldsymbol{\Gamma})$ with $v(j) = \rho(j)$ for $j \in S$?

By asking the WINDET$^\star(F)$ question for a profile $\boldsymbol{\Gamma}$ starting from a subset S of $\mathcal{I}$ containing a single issue, and progressively adding all the issues in $\mathcal{I}$ we can construct a complete outcome in $F(\boldsymbol{\Gamma})$. Note that we only need to ask the WINDET$^\star(F)$ question m times, as a negative answer implies that we must assign the opposite truth value in the partial interpretation ρ for the issue that we just added. This definition of the WINDET$^\star(F)$ problem is formulated by an existential quantifier, but an universal definition had been proposed as well (Lang and Slavkovik, 2014).[11]

Finally, we introduce a variation of WINDET(F) and WINDET$^\star(F)$ for weakly resolute rules:

WINDET$^{\mathrm{WR}}(F)$
Input: profile $\boldsymbol{\Gamma}$, issue $j \in \mathcal{I}$, value $x \in \{0, 1\}$
Question: Is there a $v \in F(\boldsymbol{\Gamma})$ with $v(j) = x$?

We will use this formulation of the winner determination problem for the rule *TrueMaj*. While we could also use WINDET$^\star(F)$, the fact that *TrueMaj* is independent and thus weakly resolute allows us to simplify the problem by only focusing on one issue at the time (instead of a subset of issues as for WINDET$^\star(F)$). Therefore, to build its entire outcome it suffices to ask the WINDET$^{\mathrm{WR}}(F)$ problem for each issue $j \in \mathcal{I}$ twice: one for $x = 1$ and one for $x = 0$.

3.3.2 Approval Rule

We start by determining the complexity of the WINDET$^\star(F)$ problem for the *Approval* rule. In the standard voting setting determining the winning candidate(s) for approval is computationally easy, as it amounts to summing for each candidate the number of approval votes they received and then looking for the highest total. When moving to goal-based voting, the problem becomes harder.

First, we need some preliminary definitions and results. Let $\Theta_2^p = \mathsf{P}^{\mathsf{NP}[\log]}$ be the class of decision problems solvable in polynomial time by a Turing machine that can make $\mathcal{O}(\log n)$ queries to an NP oracle, for n the size of the input.

Consider now the following decision problem:

MAX-MODEL
Input: satisfiable propositional formula φ, variable p of φ
Question: Is there a model $v \in \mathrm{Mod}(\varphi)$ that sets a maximal number of variables of φ to true and such that $v(p) = 1$?

To illustrate the MAX-MODEL problem, consider an instance for three propositional variables $\{p, q, s\}$ where $\varphi = p \wedge \neg q$. We have that $\mathrm{Mod}(\varphi) = \{(101), (100)\}$ are its models. Suppose we ask the MAX-MODEL question for q and φ: the answer

[11] As explained by Lang and Slavkovik (2014), for judgment aggregation rules the existential and universal definitions of the WINDET problem are *dual* to each other, in the sense that if the universal problem for a rule F is in a class C, its existential version is in the class CO-C.

will be negative, since in the model (101) with a maximal number of variables set to true, the variable q is false. On the other hand, the answer for p and φ is positive.

The complexity of the MAX-MODEL problem has been previously established:

Theorem 3.4 (Chen and Toda, 1995)**.** MAX-MODEL is Θ_2^p-complete.

We are now ready to prove the following result for winner determination of *Approval* in goal-based voting:

Theorem 3.5. WINDET*(*Approval*) is Θ_2^p-complete.

Proof. For membership, we reduce an arbitrary instance of WINDET*(*Approval*) to the ELECT-SAT$_{plurality,R_{basic}}$ problem, which has been shown to be Θ_2^p-complete in Proposition 4.(3) by Lang (2004). Intuitively, given a profile B and a formula ψ, the ELECT-SAT$_{plurality,R_{basic}}$ problem asks whether there is a model selected as the winner by plurality which satisfies ψ. It thus suffices to construct an instance for $B = \langle \gamma_1, \dots, \gamma_n \rangle$ and formula $\psi = \bigwedge_{\substack{j \in S \\ \rho(j)=1}} j \wedge \bigwedge_{\substack{j' \in S \\ \rho(j')=0}} \neg j'$.

For hardness we reduce from an instance of MAX-MODEL where $\varphi[p_1, \dots, p_m]$ is a satisfiable formula and p_i for $i \in \{1, \dots, m\}$ is one of its variables. Construct now an instance of WINDET*(*Approval*) where $\mathbf{\Gamma} = (\gamma_1, \dots, \gamma_m, \gamma_{m+1}, \gamma_{m+2}, \dots, \gamma_{2m+1})$ is a profile such that $\gamma_1 = \dots = \gamma_{m+1} = \varphi$ and $\gamma_{m+2} = p_1, \dots, \gamma_{2m+1} = p_m$.

We have that $Approval(\mathbf{\Gamma}) \subseteq \mathrm{Mod}(\varphi)$, since a strict majority of $\frac{m+1}{2m+1}$ agents already supports all the models of φ. Moreover, in this profile $\mathbf{\Gamma}$ precisely the models maximizing the number of variables set to true in φ win. In fact, consider a model $v \in \mathrm{Mod}(\varphi)$: as explained, v gets the support of all the first $m+1$ agents whose goal is φ, and then for all the agents in $\{m+2, \dots, 2m+1\}$ it gets the support of those agents whose goal-variable is true in v. Specifically, the support of v is $(m+1) + |\{p_i \mid v(p_i) = 1\}|$.

Hence, only those $v \in \mathrm{Mod}(\varphi)$ with a maximal number of 1s are in the outcome of $Approval(\mathbf{\Gamma})$. It now suffices to set $S = \{p_i\}$ for p_i the propositional variable in the instance of MAX-MODEL and $\rho(p_i) = 1$. Therefore, a formula φ has a model with a maximal number of variables set to true where p_i is true if and only if WINDET*(*Approval*) returns yes on the constructed profile $\mathbf{\Gamma}$. □

As stated in Section 3.1.2, the *Approval* rule is an instance of an IC merging operator from belief merging. The proof sketch of an equivalent result to that of Theorem 3.5 had been previously shown by Konieczny et al. (2002) in the context of belief merging. Our formulation however allows for a clearer comparison with the frameworks of voting and judgment aggregation.

3.3.3 Threshold Rule

We now study the winner determination problem for a $TrSh^\mu$ rules where the weight for each model of an agent's goal, and for each agent, is 1. Intuitively, this means that an agent submitting a goal having three models will have a global weight of 3, while an agent submitting a goal satisfied by a single model will have a weight of 1. This rule is of interest in case we want to reward an agent submitting a goal which can be satisfied in more cases (i.e., we interpret it as a sign of the agents being flexible versus them being "picky"). The individual weight of each agent is 1, however, meaning that no specific agent has more voting power by design.

Theorem 3.6. WINDET($TrSh^\mu$) is NP-complete, if $\mu_{\gamma_i}(v) = 1$ and $w_i = 1$ for all $i \in \mathcal{N}$ and $v \in 2^m$.

Proof. For membership in NP, consider an instance of WINDET($TrSh^{\mu}$) composed by a profile $\mathbf{\Gamma} = (\gamma_1, \dots, \gamma_n)$ and an issue j. Guess n sets of interpretations $X_1, \dots, X_n$ such that for all $i \in \mathcal{N}$ we have $X_i \subseteq 2^m$ and $|X_i| \leq q_j + 1$, and for each $v \in X_i$ we have $v(j) = 1$. Namely, we are guessing potential (sub)sets of the models of the agents goals having issue j set to true. The size of this guess is at most $n \cdot q_j$ and recall that $q_j \leq n$. It is then easy to check whether $|X_1| + \cdots + |X_n| > q_j$ and whether for all $v \in X_i$ we have $v \in \text{Mod}(\gamma_i)$, i.e., model-checking if $v \models \gamma_i$.

For hardness, we reduce from an instance φ of the SAT problem. Construct a profile $\mathbf{\Gamma} = (\varphi \vee \neg p)$ for a single agent and a fresh variable p. Set $q_p = 0$ (recall that in Threshold rules the operator used is $\geq$). Then, φ is satisfiable if and only if $\sum_{v \in \text{Mod}(\gamma_1)} v(p) > q_p$. □

The proof of Theorem 3.6 could be easily adapted to the case where the individual weights w_i for $i \in \mathcal{N}$ have different values, as they have to be multiplied with the corresponding cardinalities $|X_i|$. However, when the weights of the models of the goals are defined in a more complex manner, as for instance in *EQuota* rules, it may not be enough to know that the models are at least k to compute the result.

3.3.4 Majority Rules

An example of a jump in complexity brought by the weights assigned to the agents' goals is given by our generalizations of the majority rules. In particular, let PP, for Probabilistic Polynomial Time, be the class of decision problems solvable by a non-deterministic Turing machine that accepts in strictly more than half of all non-deterministic choices if and only if the answer to the problem is yes.

The typical complete problem for PP is MAJSAT, or simply MAJ in the original paper by Gill (1977):

MAJSAT
Input: propositional formula φ
Question: Is it the case that $|\text{Mod}(\varphi)| > |\text{Mod}(\neg\varphi)|$?

Namely, given a propositional formula φ we check whether φ has more models than its negation $\neg\varphi$: i.e., if it is the case that a majority of all possible interpretations is a model of φ.

Theorem 3.7 (Gill, 1977)**.** MAJSAT is PP-complete.

We can now establish the complexity bounds of the problem WINDET(*EMaj*):

Theorem 3.8. WINDET(*EMaj*) is in PSPACE and PP-hard.

Proof. We start by showing membership in PSPACE. Take the algorithm which considers one agent at a time and for each agent i it holds two counters a_i and b_i. Starting from interpretation $(00\dots0)$ and proceeding towards $(11\dots1)$, the algorithm checks whether the current interpretation satisfies $\gamma_i \wedge j$ and if it satisfies γ_i. In the former case, it increments counter a_i and in the latter case it increments counter b_i. Then, it erases the current interpretation and writes the next one. At the end, the algorithm outputs yes if and only if it it is the case that $\sum_{i \in \mathcal{N}} \frac{a_i}{b_i} > \frac{n}{2}$.

For hardness, we reduce from an instance φ of the MAJSAT problem. Consider a profile $\mathbf{\Gamma}$ for three agents such that $\gamma_1 = (\varphi \wedge p) \vee (\neg\varphi \wedge \neg p)$ and $\gamma_2 = \gamma_3 = \top$ for p a fresh variable. Since agent 1 is pivotal in $\mathbf{\Gamma}$, we then have that $EMaj(\mathbf{\Gamma})_p = 1$ if and only if $\frac{m^1_{1p}}{\text{Mod}(\varphi)} > \frac{1}{2}$ if and only if $|\text{Mod}(\varphi)| > |\text{Mod}(\neg\varphi)|$. □

We next move to the rule *2sMaj* and prove its complexity bounds for the WINDET problem. We will also use the class P^{PP} which contains problems solvable in polynomial time with access to a PP oracle. First, consider the following problem:

MAJSAT-p
Input: propositional formula φ, variable p of φ
Question: Is it the case that $|\text{Mod}(\varphi \wedge p)| > |\text{Mod}(\varphi \wedge \neg p)|$?

For example, consider the instance of MAJSAT-p for formula $\varphi = p \vee q$ and variable p. As $\text{Mod}(\varphi \wedge p) = \{(10), (11)\}$ and $\text{Mod}(\varphi \wedge \neg p) = \{(01)\}$ we have that the answer will be positive. We now establish the complexity for this problem:

Lemma 3.1. MAJSAT-p is PP-complete.

Proof. We start by showing membership in PP. Consider the non-deterministic Turing machine that guesses an interpretation v for φ, with *Var* the set of its variables. If $v \not\models \varphi$ the machine accepts with probability $\frac{1}{2}$, if $v \models \varphi \wedge p$ the machine accepts with probability 1 and if $v \models \varphi \wedge \neg p$ the machine accepts with probability 0. The probability that the machine accepts is thus given by $\frac{|\text{Mod}(\varphi \wedge p)|}{2^{|Var|}} + \frac{1}{2} \cdot \frac{|\text{Mod}(\neg\varphi)|}{2^{|Var|}}$, while the probability that it rejects is $\frac{|\text{Mod}(\varphi \wedge \neg p)|}{2^{|Var|}} + \frac{1}{2} \cdot \frac{|\text{Mod}(\neg\varphi)|}{2^{|Var|}}$. Therefore, $|\text{Mod}(\varphi \wedge p)| > |\text{Mod}(\varphi \wedge \neg p)|$ if and only if the probability that the TM accepts is higher than $\frac{1}{2}$.

For hardness, we reduce from an instance φ of the problem MAJSAT. Define $\psi = (\varphi \wedge p) \vee (\neg\varphi \wedge \neg p)$ for p a fresh variable. Now, $\psi \wedge p$ makes only the first disjunct true, while $\psi \wedge \neg p$ makes the second disjunct true. Therefore, $|\text{Mod}(\varphi)| > |\text{Mod}(\neg\varphi)|$ if and only if $|\text{Mod}(\psi \wedge p)| > |\text{Mod}(\psi \wedge \neg p)|$. □

We can now prove the complexity of WINDET for *2sMaj*:

Theorem 3.9. WINDET(*2sMaj*) is in P^{PP} and PP-hard.

Proof. In order to prove membership in P^{PP}, consider the Turing machine asking the n queries MAJSAT-j for formulas $\gamma_1, \dots, \gamma_n$ to the PP oracle. Observe that the queries allow us to know the result of the first aggregation step of *2sMaj* for j. The machine accepts if and only if strictly more than half of the queries have positive answer.

To prove hardness for the class PP, we reduce from an instance φ of the MAJSAT problem. Consider a profile $\mathbf{\Gamma}$ for three agents such that $\gamma_1 = (\varphi \wedge p) \vee (\neg\varphi \wedge \neg p)$, $\gamma_2 = p$ and $\gamma_3 = \neg p$ for p a fresh variable. Observe that since agent 1 is pivotal on p, if there are more models for φ than for $\neg\varphi$ there will be more models for p than for $\neg p$ in $\mathbf{\Gamma}$. We also have that $2sMaj(\mathbf{\Gamma})_p = 1$ if and only if $EMaj(\mathbf{\Gamma})_p = 1$, namely on this profile $\mathbf{\Gamma}$ the two rules will give the same outcome for p. The proof is then the same as for Theorem 3.8. □

Finally, we study the complexity of WINDET$^{\text{WR}}$ for the *TrueMaj* rule. First, note that in the paper by Gill (1977) it is shown that the class PP is closed under complement. This means that for any problem $X \in$ PP, the complement problem $\overline{X}$ is also in PP. In particular, we thus have that there is a polynomial reduction from any instance x of a problem $X \in$ PP or of its complement $\overline{X} \in$ PP into an instance of MAJSAT, as MAJSAT is PP-complete. Hence, we have that $x \in \overline{X}$ if and only if its translation is in MAJSAT, which means that $x \in X$ if and only if its translation is in $\overline{\text{MAJSAT}}$, which means that $\overline{\text{MAJSAT}}$ is PP-complete. This reasoning leads to proving the following result:

Theorem 3.10. WINDET$^{\text{WR}}$(*TrueMaj*) for $x = 1$ is in PSPACE and PP-hard.[12]

[12] A similar proof can be found for $x = 0$.

Proof. For membership, consider the same algorithm as the one of the proof of Theorem 3.8, with acceptance condition that $\sum_{i \in \mathcal{N}} \frac{a_i}{b_i} \geq \frac{n}{2}$.

For hardness, we provide a reduction from an instance φ of the $\overline{\text{MAJSAT}}$ problem. Construct an instance of WINDET$^{\text{WR}}$(*TrueMaj*) by taking a profile $\mathbf{\Gamma}$ for three agents, such that $\gamma_1 = (\varphi \wedge \neg p) \vee (\neg\varphi \wedge p)$ and $\gamma_2 = \gamma_3 = \top$ for p a fresh variable, and $x = 1$. Observe that when the models of φ are more than the models of its negation there will be more models for $\neg p$ than for p in $\mathbf{\Gamma}$. We then have that $|\text{Mod}(\varphi)| \geq |\text{Mod}(\neg\varphi)|$ if and only if WINDET$^{\text{WR}}$(*TrueMaj*) for $\mathbf{\Gamma} = (\gamma_1)$, $j = p$ and $x = 1$ gives a positive answer. □

As we have seen, the three variations of majority for goal-based voting all bring about a jump in complexity from the issue-wise majority rule of judgment aggregation, i.e., from P to PP. This is a direct consequence of using propositional logic as a way to compactly express the agents' goals. In fact, if the agents submitted the models of their goals as the input of a goal-based voting rule, the winner determination problem for the majorities *EMaj*, *2sMaj* and *TrueMaj* would become easy, since it would suffice to perform a (weighted) sum of the positive votes for an issue.

We can, however, obtain positive results for the winner determination problem of majority rules for certain *restrictions on the language* of the goals. The first concerns the language of conjunctions $\mathcal{L}^\wedge$:

Theorem 3.11. If $\gamma_i \in \mathcal{L}^\wedge$ for all $i \in N$, then WINDET(*EMaj*), WINDET(*2sMaj*) and WINDET$^{\text{WR}}$(*TrueMaj*) are in P.

Proof. Since $\gamma_i \in \mathcal{L}^\wedge$ for all $i \in N$, every goal is a (possibly incomplete) conjunction of literals L_k for $k \in \mathcal{I}$. Hence, for each issue k that does not appear in a goal γ exactly half of the models of γ have k true and the other half have k false; while if k appears as a positive literal all models of γ have k true, and if it appears as a negative literal all models of γ have k false.

For WINDET(*EMaj*) and WINDET$^{\text{WR}}$(*TrueMaj*), consider a counter `supp` starting at 0. For each goal γ_i where $i \in \mathcal{N}$, add 1 to `supp` if $L_j = j$ appears in γ_i, add 0 to `supp` if $L_j = \neg j$ appears in γ_i, and add 0.5 to `supp` if there is no literal L_j in γ_i. The answer of WINDET(*EMaj*) is positive if and only if `supp`$> \frac{n}{2}$, and for WINDET$^{\text{WR}}$(*TrueMaj*) with $x = 1$ the answer is positive if and only if `supp`$> \frac{n}{2}$.

For WINDET(*2sMaj*) the only difference is that also whenever there is no literal L_j in γ_i we add 0 to `supp`. Then the answer for WINDET(*2sMaj*) is positive if and only if `supp`$> \frac{n}{2}$. □

A similar positive result can be proven for the language of disjunctions $\mathcal{L}^\vee$, which includes (among other classes) Horn clauses:

Theorem 3.12. If $\gamma_i \in \mathcal{L}^\vee$ for all $i \in N$, then WINDET(*EMaj*), WINDET(*2sMaj*) and WINDET$^{\text{WR}}$(*TrueMaj*) are in P.

Proof. We follow a similar reasoning to the one of the proof of Theorem 3.11. For WINDET(*EMaj*) and WINDET$^{\text{WR}}$(*TrueMaj*), consider a counter `supp` starting at 0. For each goal γ_i where $i \in \mathcal{N}$, let k be the number of literals appearing in γ_i. First, we parse the agent's goal and in case we find $(p \vee \neg p)$ for some $p \in \mathcal{I}$ we add 0.5 to `supp`. Otherwise, add $\frac{2^{k-1}}{2^k-1}$ to `supp` if $L_j = j$ appears in γ_i, add $1 - \frac{2^{k-1}}{2^k-1}$ to `supp` if $L_j = \neg j$ appears in γ_i, and add 0.5 to `supp` if there is no literal L_j in γ_i. The answer of WINDET(*EMaj*) is positive if and only if `supp`$> \frac{n}{2}$, and for WINDET$^{\text{WR}}$(*TrueMaj*) with $x = 1$ the answer is positive if and only if `supp`$> \frac{n}{2}$.

For WinDet(*2sMaj*), the counter supp works differently: it adds 1 to supp if $L_j = j$ appears in γ_i and it adds 0 to supp if $L_j = \neg j$ appears in γ_i or if there is no literal L_j in γ_i.[13] The answer for WinDet(*2sMaj*) is positive if and only if supp$> \frac{n}{2}$. □

Theorems 3.11 and 3.12 thus open a direction of future research on restrictions over the language of goals to find tractable instances of the winner determination problem for the majority rules. Note also that the restrictions are not too limiting, as we can recover both binary aggregation (with abstentions) and Horn clauses.

3.4 Aggregating Beliefs, Judgments or Goals

In this section we compare in more details our framework of goal-based voting with both belief merging (Konieczny and Pérez, 2002) and judgment aggregation (List, 2012). The connection with the former comes from the fact that belief merging was proposed to combine the beliefs of multiple agents: though belief merging operators and axioms differ from those of goal-based voting, both settings are concerned with the problem of aggregating propositional formulas into a result (a set of interpretations) for the group. The connection with the latter comes from the fact that judgment aggregation is a setting where agents submit binary vectors over a set of issues to take a collective decision: goal-based voting adapts known rules in judgment aggregation to the more expressive propositional goals.

3.4.1 Goal-based Voting and Belief Merging

We here analyze the axioms and rules of goal-based voting with respect to the IC postulates of belief merging, following the formulation by Everaere et al. (2017). The first obvious observation is that since the *Approval* rule is an IC merging operator, as mentioned in Section 3.1.2, it satisfies all the IC postulates.

In belief merging, a group of n agents wants to aggregate their individual beliefs to obtain the beliefs of the group. Each agent i has a set $K_i = \{\varphi_1, \dots, \varphi_k\}$ of consistent propositional formulas as her beliefs. Profile $E = (K_1, \dots, K_n)$ is a vector including all the individual belief bases. An integrity constraint μ is a propositional formula posing a restriction on the possible outcomes. A merging operator $\Delta_\mu(E)$ is a function from profiles of belief bases and integrity constraints to a set of formulas. In goal-based voting terms, we have that belief bases correspond to goals: i.e., $\gamma_i = \varphi_1 \wedge \cdots \wedge \varphi_k$ for $\{\varphi_1, \dots, \varphi_k\} = K_i$. A profile E thus corresponds to a goal-profile $\mathbf{\Gamma}$ and a merging operator $\Delta_\mu(E)$ corresponds to $F(\mathbf{\Gamma})$, with the sole exception that the output of $\Delta_\mu(E)$ is a set of formulas while for $F(\mathbf{\Gamma})$ we have a set of interpretations.

The first two IC postulates are the following:

(IC0) $\Delta_\mu(E) \models \mu$
(IC1) If μ is consistent, then $\Delta_\mu(E)$ is consistent

The postulate (IC0) requires the outcome of a rule to satisfy the constraint, while (IC1) demands the outcome to be consistent if the constraint is consistent. As in goal-based voting there are no integrity constraints (equivalently, the integrity constraint is $\mu = \top$) and since $F(\mathbf{\Gamma}) \neq \emptyset$ on all $\mathbf{\Gamma}$, (IC0) and (IC1) are satisfied by design.

[13] Observe that the first step of aggregation of *2sMaj* does not distinguish between a goal expressed in $\mathcal{L}^\wedge$ and the corresponding goal in $\mathcal{L}^\vee$ where each occurrence of $\wedge$ is replaced by a $\vee$.

The next postulate determines a crucial difference between belief merging operators and goal-based voting rules (except, of course, for the *Approval* rule):

$$\text{(IC2) If } \bigwedge E \wedge \mu \text{ is consistent, then } \Delta_\mu(E) \equiv \bigwedge E \wedge \mu$$

Postulate (IC2) states that the outcome of a rule should coincide with the conjunction of the individual goals if they are consistent. In Section 3.2.3 we introduced this axiom under the name of model-unanimity and showed that *EMaj*, *TrueMaj* and *2sMaj* did not satisfy it. We now generalize this result by proving that (IC2) is incompatible with both resoluteness and weak resoluteness:

Proposition 3.6. No goal-based voting rule F satisfying (IC2) can satisfy (R) or (WR).

Proof. Consider a profile $\mathbf{\Gamma}$ for two issues such that $\gamma_1 = \cdots = \gamma_n = 1 \vee 2$. Since $\text{Mod}(\bigwedge_{i \in \mathcal{N}} \gamma_i) = \{(11), (10), (01)\}$ we have that $|\text{Mod}(\bigwedge_{i \in \mathcal{N}} \gamma_i)| > 1$, thus F is not resolute, but also $F(\mathbf{\Gamma}) = \text{Mod}(1 \vee 2)$ and thus F is not weakly resolute. □

The next postulate (IC3) expresses the idea of irrelevance of syntax:

$$\text{(IC3) If } E_1 \equiv E_2 \text{ and } \mu_1 \equiv \mu_2, \text{ then } \Delta_{\mu_1}(E_1) \equiv \Delta_{\mu_2}(E_2)$$

All the goal-based voting rules defined in Section 3.1.2 satisfy (IC3), as propositional logic is used for a compact representation of goals but then the rules look at the models of the goals to compute the outcome.

Postulate (IC4) has been introduced as a fairness axiom defined for two agents, aiming at treating equally two belief bases. Intuitively, it states that if the result of the merging is consistent with one belief base, it should also be consistent with the other, so that there is no bias towards either. The formal definition is:

(IC4) If $K_1 \models \mu$ and $K_2 \models \mu$, then $\Delta_\mu((K_1, K_2)) \wedge K_1$ is consistent if and only if $\Delta_\mu((K_1, K_2)) \wedge K_2$ is consistent

Goal-based voting rules *EMaj*, *2sMaj* and *2sMaj* do not satisfy (IC4), as shown by the following example:

Example 3.12. Consider a profile $\mathbf{\Gamma}$ for two agents and three issues such that $\gamma_1 = \neg 1 \wedge \neg 2 \wedge \neg 3$ and $\gamma_2 = (1 \wedge \neg 2 \wedge \neg 3) \vee (\neg 1 \wedge 2 \wedge \neg 3) \vee (\neg 1 \wedge \neg 2 \wedge 3)$. We have that $\mathit{EMaj}(\mathbf{\Gamma}) = \mathit{TrueMaj}(\mathbf{\Gamma}) = \mathit{2sMaj}(\mathbf{\Gamma}) = \{(000)\}$. The outcome is thus only consistent with the goal of agent 1 and not with that of agent 2.

Postulates (IC5) and (IC6) can be seen as analogous to the axiom of *reinforcement* in social choice theory (Young, 1974). Together they intuitively say that if we combine two profile into a new one, the outcome of the merging on the new profile should be the intersection of the outcomes on the two initial profiles. Formally:

(IC5) $\Delta_\mu(E_1) \wedge \Delta_\mu(E_2) \models \Delta_\mu(E_1 \sqcup E_2)$

(IC6) If $\Delta_\mu(E_1) \wedge \Delta_\mu(E_2)$ is consistent, then $\Delta_\mu(E_1 \sqcup E_2) \models \Delta_\mu(E_1) \wedge \Delta_\mu(E_2)$

We prove that the three majoritarian rules *EMaj*, *2sMaj* and *TrueMaj* satisfy the reinforcement postulates (IC5) and (IC6). For clarity, we express them in goal-based voting terms:[14]

[14] The same formulation of reinforcement has been introduced for binary aggregation as well (Costantini et al., 2016).

Theorem 3.13. For any $\mathbf{\Gamma}$ and $\mathbf{\Gamma}'$, *EMaj*, *2sMaj* and *TrueMaj* satisfy $F(\mathbf{\Gamma}) \cap F(\mathbf{\Gamma}') = S \neq \emptyset$ if and only if $F(\mathbf{\Gamma} \sqcup \mathbf{\Gamma}') = S$.

Proof. Consider two arbitrary profiles $\mathbf{\Gamma}$ and $\mathbf{\Gamma}'$. Let $EMaj(\mathbf{\Gamma}) = EMaj(\mathbf{\Gamma}') = \{w\}$. For all $j \in \mathcal{I}$: if $w(j) = 1$, then there were more than $\frac{n}{2}$ votes for j in both $\mathbf{\Gamma}$ and $\mathbf{\Gamma}'$ (and consequently in $\mathbf{\Gamma} \sqcup \mathbf{\Gamma}'$); if $w(j) = 0$, then in $\mathbf{\Gamma}$ and $\mathbf{\Gamma}'$ either there was a tie for j or there were less than $\frac{n}{2}$ votes for j. Any combination of ties or votes lower than $\frac{n}{2}$ for j in $\mathbf{\Gamma}$ and $\mathbf{\Gamma}'$ still leads to $EMaj(\mathbf{\Gamma} \sqcup \mathbf{\Gamma}')_j = 0$.

The reasoning for *2sMaj* is the same as for *EMaj* applied to the second step only. For *TrueMaj*, let $TrueMaj(\mathbf{\Gamma}) \cap TrueMaj(\mathbf{\Gamma}') = S$. For all $j \in \mathcal{I}$: if there are $w, w' \in S$ such that $w(j) = 1$ and $w'(j) = 0$, then $\mathbf{\Gamma}$ and $\mathbf{\Gamma}'$ had a tie in the votes for j and thus a tie will be in $\mathbf{\Gamma} \sqcup \mathbf{\Gamma}'$ (hence in the outcome). If $w(j) = 1$ for all $w \in S$ (analogously for 0), there may have been a tie in either $\mathbf{\Gamma}'$ or $\mathbf{\Gamma}$ for j, but not both, and so $\mathbf{\Gamma} \sqcup \mathbf{\Gamma}'$ will have no tie for j. □

Next, we have (IC7) and (IC8) which have been introduced as generalizations of belief revision axioms:

(IC7) $\Delta_{\mu_1}(E) \wedge \mu_2 \models \Delta_{\mu_1 \wedge \mu_2}(E)$
(IC8) If $\Delta_{\mu_1}(E) \wedge \mu_2$ is consistent, then $\Delta_{\mu_1 \wedge \mu_2}(E) \models \Delta_{\mu_1}(E)$

Again, since in the framework of goal-based voting presented here we do not consider integrity constraints, the two postulates are not applicable.

Finally, we have the (Maj) postulate defining majority merging operators: it states that we can always make any belief base the outcome by adding a certain number of its copies to a given profile. Formally:

$$\text{(Maj) } \exists n \Delta_\mu(E_1 \sqcup E_2^n) \models \Delta_\mu(E_2)$$

Interestingly, the (Maj) postulate is not satisfied by neither *EMaj* nor *TrueMaj*, as illustrated by the following example:

Example 3.13. Take a profile $\mathbf{\Gamma}$ for two issues such that $\gamma_1 = 1 \wedge 2$ and $\gamma_2 = 1 \leftrightarrow \neg 2$: by adding any number of copies of γ_2 to $\mathbf{\Gamma}$, in the presence of γ_1 the outcome will always be $\{(11)\}$ for both *EMaj* and *TrueMaj*.

In conclusion, belief merging and goal-based voting are technically similar frameworks to the point where some rules, like *Approval*, can be used in both. On the other hand, as we have seen in this section, the focus in each framework is on rules which satisfy some properties (being them called axioms or postulates) that are deemed more important if the object of the aggregation is a goal or a belief and that can be mutually exclusive.

3.4.2 Goal-based Voting and Judgment Aggregation

The framework of judgment aggregation has been introduced to model collective decision-making when agents have binary opinions over a set of issues. This framework can be expressed in two equivalent formulations: the *formula-based* model by List (2012) and the *binary aggregation* model by Grandi and Endriss (2011). The models are equivalent in the sense that it is possible to translate any problem expressed in one framework into a problem of the other framework (Endriss et al., 2016). Moreover, the relationship of judgment aggregation rules and voting rules has been studied by Lang and Slavkovik (2013). In this section we will first consider formula-based judgment aggregation, then the binary aggregation model (also with its extension to abstentions), and finally we will re-examine the axioms used in Theorem 3.3 in judgment aggregation terms.

We start by briefly describing formula-based judgment aggregation and compare it with goal-based voting. A set of *agents* $\mathcal{N} = \{1, \dots, n\}$ have to take a decision over a set $\Phi = \{\varphi_1, \dots, \varphi_m, \neg\varphi_1, \dots, \neg\varphi_m\}$ of propositional formulas called the *agenda*. Each agent i supports some formulas of the agenda by adding them to her individual consistent judgment set J_i which contains one of either φ_k or $\neg\varphi_k$ for $1 \leq k \leq m$.

A first intuitive idea would be to translate an instance of a goal-based voting problem into formula-based judgment aggregation as follows:

Example 3.14. Let $\mathcal{N} = \{1, \dots, n\}$ be a set of agents expressing the goals $\gamma_1, \dots, \gamma_n$ over the issues in $\mathcal{I} = \{1, \dots, m\}$. Construct now a formula-based judgment aggregation problem by considering the set of agents $\mathcal{N}^{\text{JA}} = \{1, \dots, n\}$ and the agenda $\Phi = \{\gamma_1, \dots, \gamma_n, \neg\gamma_1, \dots, \neg\gamma_n\}$.[15] Then, let $J_i = \{\gamma_i\} \bigcup_{k \neq i}\{\neg\gamma_k\}$ for all $i \in \mathcal{N}$.

There are a number of problems with the translation proposed in Example 3.14. From a conceptual point of view, the agenda should be a set in advance list of issues on which agents are asked to provide an opinion: it should not be the opinions themselves. From a technical point of view, individual judgment sets need to be consistent while the proposed translation may give inconsistent sets in case goals are mutually compatible (e.g., if $\gamma_1 = 1 \land 2$ and $\gamma_2 = 1$ we would have $J_1 = \{1 \land 2, \neg 1\}$ which is inconsistent). From a practical point of view, many rules in judgment aggregation calculate their outcome based on how many agents support a formula in the agenda: in the proposed translation, only agents having goal γ_i would include formula γ_i in their individual judgment set, and in profiles with n different goals it would result in a symmetric situation where only one agent supports each goal-formula and the negations of all the other goal-formulas.

Observe that even a translation where $\Phi = \mathcal{I} \bigcup_{j \in \mathcal{I}}\{\neg j\}$ would not work, as an agent with goal $\gamma = 1 \to 2$ would not be able to choose whether or not to include formulas 1 and 2 in her judgment set. For ease of presentation, given the difficulties just raised and since the two models are equivalent, we discuss how to translate a judgment aggregation problem into goal-based voting for the binary aggregation model only.

In binary aggregation, we have a set of *agents* $\mathcal{N} = \{1, \dots, n\}$ who have to express an opinion over a set $\mathcal{I} = \{1, \dots, m\}$ of binary issues. Each agent $i \in \mathcal{N}$ submits as her *ballot* a vector $B_i \in \{0,1\}^m$ with a binary decision for each issue in $\mathcal{I}$. The j-th position of vector B_i is denoted by b_{ij}. A *profile* is a vector $\boldsymbol{B} = (B_1, \dots, B_n)$ containing all the agents' ballots. Finally, we have an *integrity constraint* IC, that is a propositional formula written over variables in $\mathcal{I}$ to model an existing relationship among the issues.[16] Each agent i has to submit a ballot B_i satisfying IC.

In order to translate an instance of a binary aggregation problem into goal-based voting we thus need to extend the latter framework with integrity constraints. Then, a possible translation is illustrated by the following example:

Example 3.15. Let $\boldsymbol{B} = (B_1, \dots, B_n)$ be a binary aggregation profile for issues $\mathcal{I} = \{1, \dots, m\}$, agents $\mathcal{N} = \{1, \dots, n\}$ and constraint IC. Consider now the goal-based

[15] To be consistent with the common definition of agendas in formula-based judgment aggregation, we let $\neg\gamma_i = \psi$ in case $\gamma_i = \neg\psi$ (i.e., we avoid double negation).

[16] When translating an instance of a formula-based judgment aggregation problem into a binary aggregation one, the constraint IC is used to model the logical structure of the agenda Φ.

voting translation whose issues and agents are $\mathcal{I}$ and $\mathcal{N}$, respectively, and the goal of an agent $i \in \mathcal{N}$ is defined as $\gamma_i = \bigwedge_{\substack{b_{ij}=1 \\ j \in \mathcal{I}}} j \wedge \bigwedge_{\substack{b_{ij}=0 \\ j \in \mathcal{I}}} \neg j$. Namely, we construct a goal-profile $\mathbf{\Gamma}$ where each goal γ_i is such that $\text{Mod}(\gamma_i) = \{B_i\}$. By construction we thus have that $\gamma_i \models \text{IC}$.

As far as rules are concerned, the issue-wise (strict) majority rule for binary aggregation (Dietrich and List, 2007b; Endriss, 2016), is formally defined as:

$$Maj(\boldsymbol{B})_j = 1 \textit{ iff } \sum_{i \in \mathcal{N}} b_{ij} \geq \left\lceil \frac{n}{2} \right\rceil.$$

It is then easy to prove that both *EMaj* and *2sMaj* are generalizations of *Maj*, in the sense that their outcomes coincide when agents have goals in the form of complete conjunctions of literals (as per Example 3.15). More precisely, we have that for *EMaj*:

$$\begin{aligned}
EMaj(\mathbf{\Gamma})_j = 1 &\textit{ iff } \sum_{i \in \mathcal{N}} \Big(\sum_{v \in \text{Mod}(\gamma_i)} \frac{v(j)}{|\text{Mod}(\gamma_i)|} \Big) \geq \left\lceil \frac{n}{2} \right\rceil \\
&\textit{ iff } \sum_{i \in \mathcal{N}} \Big(\sum_{v \in \text{Mod}(\gamma_i)} \frac{v(j)}{1} \Big) \geq \left\lceil \frac{n}{2} \right\rceil \\
&\textit{ iff } \sum_{i \in \mathcal{N}} v_i(j) \geq \left\lceil \frac{n}{2} \right\rceil \\
Maj(\boldsymbol{B})_j = 1 &\textit{ iff } \sum_{i \in \mathcal{N}} b_{ij} \geq \left\lceil \frac{n}{2} \right\rceil
\end{aligned}$$

And for *2sMaj* we have:

$$\begin{aligned}
2sMaj(\mathbf{\Gamma}) &= Maj(EMaj(\text{Mod}(\gamma_1)), \ldots, EMaj(\text{Mod}(\gamma_n))) \\
&= Maj(B_1, \ldots, B_n)
\end{aligned}$$

Since *TrueMaj* is weakly resolute, it does not directly correspond to *Maj*. Endriss and Grandi (2014) proposed an irresolute issue-wise majority defined as $\text{Maj}^\star(\boldsymbol{B}) = \{B^w, B^s\}$ where B^w is the outcome of weak majority and B^s is that of strict majority. This is a different rule from *TrueMaj*, even when restricted to profiles of binary aggregation, as shown by the following example:

Example 3.16. Take a profile $\mathbf{\Gamma}$ for two agents and issues such that $\text{Mod}(\gamma_1) = \{(01)\}$ and $\text{Mod}(\gamma_2) = \{(10)\}$. The result of $\text{Maj}^\star$ is $\{(00), (11)\}$, while the result of *TrueMaj* is $\{(11), (00), (10), (01)\}$.

In the framework of binary aggregation with abstentions by Dokow and Holzman (2010b) entries in a ballot can also have value $\star$ in case the agents is abstaining on those issues, i.e., $B_i \in \{0, 1, \star\}^m$. We can keep the translation of ballots into goals of Example 3.15: if a ballot B_i for some $i \in \mathcal{N}$ has value $b_{ij} = \star$ for some $j \in \mathcal{I}$, the issue will not appear as a conjunct in the corresponding γ_i. Terzopoulou et al. (2018) recently defined classes of *scoring functions* for formula-based judgment aggregation with abstentions, where different weights are given to the formulas in an individual judgment set J_i depending on its size: an interesting direction for future work would be to translate such rules as goal-based voting rules.

We conclude this section by comparing the axioms used in Theorem 3.3, i.e., (I), (N), (A), (M) and (U), with those used for characterizing the majority rule in resolute binary aggregation (Grandi and Endriss, 2011). From the point of view of goal-based voting, we will thus focus on the special case of goals which are complete conjunctions of literals: for all $i \in \mathcal{N}$ we have $|\text{Mod}(\gamma_i)| = 1$. Hence, for simplicity we can write $m_i(j) = 0$ instead of $m_i(j) = (1,0)$, and $m_i(j) = 1$ instead of $m_i(j) = (0,1)$. As in the proof of Theorem 3.3, we denote by $\mathcal{G}^\wedge$ the class of profiles $\mathbf{\Gamma}$ where goals are complete conjunctions. Note however that in binary aggregation functions are defined for fixed $\mathcal{N}$ and $\mathcal{I}$, while in goal-based voting functions are defined for any $\mathcal{N}$ and $\mathcal{I}$.

Independence. A rule F is independent if and only if there are functions $f_j : \{0,1\}^m \to \{\{0\},\{1\},\{0,1\}\}$ for $j \in \mathcal{I}$ and $m \in \mathbb{N}^+$ such that for all profiles $\mathbf{\Gamma}$ we have $F(\mathbf{\Gamma}) = \Pi_{j\in\mathcal{I}} f_j(m_1(j), \ldots, m_n(j))$. If this is the case, it means that in particular if in two profiles $\mathbf{\Gamma}$ and $\mathbf{\Gamma}'$ we have that $m_i(j) = m'_i(j)$ for all $i \in \mathcal{N}$, $f_j(m_1(j), \ldots, m_n(j)) = f_j(m'_1(j), \ldots, m'_n(j))$ and thus $F(\mathbf{\Gamma})_j = F(\mathbf{\Gamma}')_j$. Thus, the restriction of (I) to profiles in $\mathcal{G}^\wedge$, i.e., profiles of binary aggregation,[17] implies the binary aggregation axiom of independence $(\text{I})^{\text{BA}}$:

$(\text{I})^{\text{BA}}$: For any issue $j \in \mathcal{I}$ and profiles $\boldsymbol{B}$ and $\boldsymbol{B}'$, if $b_{ij} = b'_{ij}$ for all $i \in \mathcal{N}$, then $F(\boldsymbol{B})_j = F(\boldsymbol{B}')_j$.

The formulation of the axiom of independence in goal-based voting is close to the original formulation by List and Pettit (2002) of the axiom of *systematicity* for judgment aggregation. Besides being defined for *resolute* rules and for a *given* set of agents and issues, the notion of systematicity differs from independence as it also includes neutrality, since the function f used to decide the outcome is the same for all issues.[18]

Neutrality. The goal-based voting notion of neutrality when restricted to profiles in $\mathcal{G}^\wedge$ implies the binary aggregation notion of issue-neutrality $(\text{I-N})^{\text{BA}}$:

$(\text{I-N})^{\text{BA}}$: For any two issues $j, j' \in \mathcal{I}$ and any profile $\boldsymbol{B}$, if for all $i \in \mathcal{N}$ we have that $b_{ij} = b'_{ij}$ then $F'(\boldsymbol{B})_j = F(\boldsymbol{B})_{j'}$.

In fact, if we consider a profile $\mathbf{\Gamma}$ in which two issues $j, k \in \mathcal{I}$ are such that $m_i(j) = m_i(k)$ for all $i \in \mathcal{N}$, and we consider the permutation σ such that $\sigma(j) = k$, $\sigma(k) = j$, and $\sigma(\ell) = \ell$ (for $\ell \in \mathcal{I} \setminus \{j,k\}$) we obtain a profile $\mathbf{\Gamma}'$ which is identical to $\mathbf{\Gamma}$. Therefore, $F(\mathbf{\Gamma})_j = F(\mathbf{\Gamma}')_k = F(\mathbf{\Gamma})_k$.

Anonymity. The axiom of anonymity in goal-based voting is a direct generalization of anonymity in binary aggregation $(\text{A})^{\text{BA}}$:

$(\text{A})^{\text{BA}}$: For any profile $\boldsymbol{B}$ and any permutation $\sigma : \mathcal{N} \to \mathcal{N}$, it is the case that $F(B_1, \ldots, B_n) = F(B_{\sigma(1)}, \ldots, B_{\sigma(n)})$.

It suffices to substitute the propositional goals with their unique model to obtain the axiom of binary aggregation.

[17] To be more precise, a profile in binary aggregation is $(v_1, \ldots, v_n$ for $v_i \in \text{Mod}(\gamma_i)$ for $i \in \mathcal{N}$ rather than $(\gamma_1, \ldots, \gamma_n)$, as we have seen above in this section. We use the two formulations interchangeably given that we focus on goals in $\mathcal{G}^\wedge$.

[18] Similarly to the discussion presented here, in the paper by List and Pettit (2004) it is stated that the original notion of systematicity implies the preference aggregation notion of *independence of non-welfare characteristics* (INW). The generalization of the definition of (INW) from preferences to judgments has then been used as the standard definition of systematicity, and thus of independence (List, 2012).

Unanimity. A rule F is unanimous if for all profiles $\mathbf{\Gamma}$ and for all $j \in \mathcal{I}$, if $m_i(j) = x$ for all $i \in \mathcal{N}$, then $F(\mathbf{\Gamma})_j = x$ for $x \in \{0,1\}$. Again, this is a direct generalization of the axiom of unanimity in binary aggregation (U)$^{\text{BA}}$:

(U)$^{\text{BA}}$: For any profile $\boldsymbol{B}$ and any $x \in \{0,1\}$, if $b_{ij} = x$ for all $i \in \mathcal{N}$ then $F(\boldsymbol{B})_j = x$.

Monotonicity. A rule F is monotonic if for all profiles $\mathbf{\Gamma} = (\gamma_1, \dots, \gamma_i, \dots, \gamma_n)$ and $\mathbf{\Gamma}^\star = (\gamma_1, \dots, \gamma_i^\star, \dots, \gamma_n)$, for all $j \in \mathcal{I}$ and $i \in \mathcal{N}$, if $m_i(j)^\star = 1$ and $m_i(j) = 0$ then $F(\mathbf{\Gamma})^1_j \geq F(\mathbf{\Gamma})^0_j$ implies $F(\mathbf{\Gamma}^\star)^1_j > F(\mathbf{\Gamma}^\star)^0_j$. First, observe that the condition of *comparable* profiles is trivially satisfied in binary aggregation, as agents always have a goal with a single model. Goal-based monotonicity implies issue-monotonicity in binary aggregation (M)$^{\text{BA}}$:

(M)$^{\text{BA}}$: For any issue $j \in \mathcal{I}$ and profiles $\boldsymbol{B} = (B_1, \dots, B_i, \dots, B_n)$ and $\boldsymbol{B'} = (\boldsymbol{B}_{-i}, B'_i)$ if $b_{ij} = 0$ and $b_{ij'} = 1$, then $F(\boldsymbol{B})_j = 1$ entails $F(\boldsymbol{B'})_j = 1$.

Goal-based monotonicity is however more general, as it also considers possible ties in the outcome (which for the *TrueMaj* rule derive from a tie in the input).

Duality. The notion of duality in goal-based voting does not directly imply the axiom of domain neutrality (D-N)$^{\text{BA}}$ in binary aggregation:

(D-N)$^{\text{BA}}$: For any two issues $j, j' \in \mathcal{I}$ and any profile $\boldsymbol{B}$, if $b_{ij} = 1 - b_{ij'}$ for all $i \in \mathcal{N}$ then $F(\boldsymbol{B})_j = 1 - F(\boldsymbol{B})_{j'}$.

Duality and independence together imply domain-neutrality. Consider a profile $\mathbf{\Gamma}$ where for two issues $j, j' \in \mathcal{I}$ we have $m_i(j) = 1 - m_i(j')$ for all $i \in \mathcal{N}$. In the profile $\mathbf{\Gamma}^\star$ where for all $i \in \mathcal{N}$ we have $\gamma_i^\star = \overline{\gamma_i}$, the resolute outcome $F(\overline{\mathbf{\Gamma}})_j$ is equal to $1 - F(\mathbf{\Gamma})_j$. Observe however that $m_i(j)^\star = m_i(j')$ and thus by independence we have that $F(\mathbf{\Gamma})_{j'} = 1 - F(\overline{\mathbf{\Gamma}})_j$.

In binary aggregation, we have the following result:

Theorem 3.14 (Grandi and Endriss, 2011)**.** If the number of individuals is odd, an aggregation procedure F satisfies (A)$^{\text{BA}}$, (I-N)$^{\text{BA}}$, (D-N)$^{\text{BA}}$, (I)$^{\text{BA}}$, (M)$^{\text{BA}}$ if and only if it is the majority rule.

Theorem 3.3 can thus be seen as a generalization of Theorem 3.14, with the important distinctions that (i) we consider rules defined for all $\mathcal{N}$ and $\mathcal{I}$, (ii) we consider odd and even number of agents, and (iii) *TrueMaj* is not resolute (unlike *Maj*).

3.5 Conclusions

From the observation that judgment aggregation falls short in modeling some real-life examples of collective decision-making in multi-issue domains, such as finding a shared travel plan for a group of friends (Example 3.1), we introduced voting rules to aggregate a set of compactly represented goals in propositional logic into a collectively satisfying alternative. Some of our rules were inspired by belief merging operators or they were generalizations of known voting rules in social choice theory.

In order to formally analyze them, we explored many possible formulations of axioms describing desirable properties and studied how they relate to one another. We noticed a tension between resoluteness and fairness for a voting rule, as certain sets of axioms are not satisfiable together (see, e.g., Theorem 3.1). A rule that tries to find a compromise between these two needs is the adaptation of the majority rule that we call *TrueMaj*, for which we provided an axiomatization in Theorem 3.3.

We also studied the computational complexity of determining the outcome for our rules, with an expected increase in complexity due to the fact that our agents use compactly expressed preferences. In particular, we find that *Approval* is Θ_2^p-complete (as per analogous results in the literature on belief merging), a special case of *TrSh* rules is NP-complete, and our generalizations of majority are all hard for the PP class. Such a result opens up a path for future research in studying restrictions on the language of goals that may determine islands of tractability for the WinDet problem, or develop tractable approximations for their computation. We obtained first positive results in this direction with Theorems 3.11 and 3.12.

Finally, we compared our framework with both belief merging and judgment aggregation, in order to give a precise assessment of how the settings relate to one another. We find that some of the properties we study are incompatible with properties of other frameworks (in particular, belief merging) and that the rules in different frameworks focus on different types of properties. For the reader interested in a comparison between the frameworks of belief merging and judgment aggregation we refer to the paper by Everaere et al. (2017).

Chapter 4

Aggregation of Incomplete CP-nets

In Chapter 3 we have seen how goals of different agents who look for a collective decision can be compactly represented by a propositional formula. Propositional logic can be used also to compactly represent preferences of agents over possible outcomes that are conditional on specific assumptions, as in the scenario presented in the following example:

Example 4.1. Lucy wants to rent an apartment and to make her decision she deems three factors to be important: location, price, and whether she will have flatmates or not. In case the apartment is expensive or it is located in the suburbs, she would prefer having flatmates (to share utilities or get some company) rather than living alone. Hence, if an apartment is *expensive* and located in the city center she would rather share it with flatmates. Lucy does not say what her preference is over having flatmates if she sees the listing for a *cheap and central* apartment. Nor do we know what she would prefer between an expensive shared apartment in the center and a cheap shared apartment in the suburbs.

In Example 4.1, Lucy needs a compact way to express her preferences instead of simply ordering all possible configurations of apartments. Moreover, she does not specify her preference over some variable for all possible combinations of the remaining variables, which would be required by the classical framework of CP-nets introduced in Section 2.3. We thus need to work in the framework of *generalized* CP-nets by Goldsmith et al. (2008), gCP-nets for short, where agents do not have the burden of providing complete preference tables but they can state preconditions of their statements as propositional formulas.

In this chapter we thus bridge two lines of research in CP-nets: the one on gCP-nets and the one on the aggregation of CP-nets coming from *multiple* agents (mCP-nets). We will consider the case where multiple agents express a gCP-net and they want to find a collective ordering that reflects their individual preferences. We will use four semantics to aggregate individual gCP-nets, inspired by the work on mCP-nets by Rossi et al. (2004) and Lukasiewicz and Malizia (2016), but adapted to the fact that CP-nets in our case are incomplete. Analogously to the winner determination problem in goal-based voting that we saw in Section 3.3.1, when studying the computational complexity for mgCP-nets we will not aim at generating the full order on outcomes (its number being in general exponential in the number of variables). Rather, we will focus on determining dominance over outcomes for different semantics, starting from complexity results for *individual* gCP-nets.

Grandi et al. (2014) also studied the aggregation of CP-nets but in the presence of constraints delimiting *feasible* outcomes, and defined a procedure based on sequential voting that can be applied on CP-net profiles having special characteristics (i.e., profiles where we can find an ordering of the variables such that the agents can vote for the value of the current variable given what has already been decided for the previous variables). Another approach is to use heuristics and techniques from constraint satisfaction problems for computing local winners in the individual CP-nets and then rule out those that are majority-dominated by some other alternative (Li et al., 2014, 2015). Even *probabilistic* CP-nets (PCP-nets), where for each variable X there is a probability distribution over the set of all possible preference orderings for the domain of X, have been used to model the aggregation of multiple CP-nets (Cornelio et al., 2013, 2015). Finally, the aggregation of classical CP-nets in a context of social influence has been studied by Maran et al. (2013), with focus in particular on the issue of bribery.

4.1 Framework

We start by providing the syntax and the semantics for gCP-nets (Section 4.1.1). Then, we give some complexity results for individual CP-nets (Section 4.1.2) and present *m*gCP-nets (Section 4.1.3).

4.1.1 Preliminaries

Let $\mathcal{V} = \{X, Y, Z, \dots\}$ be a finite set of *variables*. Each variable $X \in \mathcal{V}$ has a finite domain $D(X) = \{x_1, \dots, x_k\}$ of possible *values*. With a slight abuse of notation, we write simply x_i to indicate that the variable X has been assigned value $x_i \in D(X)$. Agents express conditional preferences, meaning that the value for one variable may depend on the values assigned to other variables. For X a variable and $x_i, x_j \in D(X)$ two of its possible values, $x_i \rhd x_j$ expresses a *preference over* X intuitively stating that value x_i is preferred to x_j for X. A preference can only be expressed over values of a single variable: i.e., we cannot write $x_i \rhd y_i$ for $x_i \in D(X)$ and $y_i \in D(Y)$ for $X \neq Y$.

A *propositional formula* ψ *over* $\mathcal{W} \subseteq \mathcal{V}$ is defined over $\bigcup_{X \in \mathcal{W}} D(X)$ by using the standard propositional connectives, though the domains are not necessarily binary. Therefore, a formula such as $\neg x_i$, for $x_i \in D(X)$, is equivalent to $\bigvee_{x_j \in D(X), x_j \neq x_i} x_j$ and formulas such as $x_i \wedge x_j$ for x_i and x_j different values of variable X are inconsistent, as illustrated by the following example.

Example 4.2. Lucy is considering three neighborhoods in Toulouse: Capitole, Esquirol and Rangueil. We thus have a variable, let us call it B, whose domain $D(B)$ has three values, one for each neighborhood: b_1, b_2 and b_3. If Lucy wants to exclude Capitole, it means that she either considers Esquirol or Rangueil: i.e., formula $\neg b_1$ is equivalent to $b_2 \vee b_3$. Moreover, she cannot state that she wants her apartment to be in two neighborhoods at the same time, hence a formula like $b_2 \wedge b_1$ is inconsistent.

Let ψ be a propositional formula over variables in $\mathcal{W}$ and $X \notin \mathcal{W}$ be a variable. Let also $\pi = x_i \rhd x_j$ be a preference over two possible values $x_i, x_j \in D(X)$. A *conditional preference statement* φ defined as $\psi : \pi$ expresses the fact that if ψ is true, then the preferences over X behave as stated by π. In particular, we call formula ψ the *precondition of* φ. The condition $X \notin \mathcal{W}$ is required to rule out statements such as "If the apartment is in Rangueil, then an apartment in Capitole is to be preferred to an apartment in Esquirol". A *gCP-net* $N = \{\varphi_1, \dots, \varphi_n\}$ is a finite set of conditional preference statements. We rewrite more compactly transitive statements of the form $\psi : (x_i \rhd x_j)$ and $\psi : (x_j \rhd x_k)$ as $\psi : (x_i \rhd x_j \rhd x_k)$.

A gCP-net can be represented by a dependency graph, where each node is a variable and an edge from X to Y indicates that some values of X occur in the precondition of a preference over values for Y. A gCP-net N is *acyclic* if its dependency graph is acyclic. The following example shows the concepts introduced above:

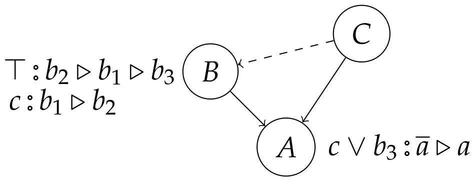

FIGURE 4.1: Dependency graphs and conditional preference statements for the gCP-nets N_1 and N_2 of Example 4.3. The additional dependency given by statement φ_3 is shown with a dashed line.

Example 4.3. Lucy is looking for an apartment in Toulouse on a popular website for local listings. She is interested by three features: the type of accommodation she can rent (A), which can be either an entire apartment (a) or a room in a shared apartment ($\overline{a}$); the neighborhood where the house is located (B), that for her only includes Capitole (b_1), Esquirol (b_2) or Rangueil (b_3); the cost (C), which can be high (c) or normal ($\overline{c}$). The set of variables is $\mathcal{V} = \{A, B, C\}$ and the domains of the variables are $D(A) = \{a, \overline{a}\}$, $D(B) = \{b_1, b_2, b_3\}$ and $D(C) = \{c, \overline{c}\}$. Lucy submits to this service her gCP net $N_1 = \{\varphi_1, \varphi_2\}$, where the statements are as follows:

$$\begin{array}{lrl} (\varphi_1) & \top : & b_2 \rhd b_1 \rhd b_3, \\ (\varphi_2) & c \vee b_3 : & \overline{a} \rhd a. \end{array}$$

Later on, Lucy updates her gCP-net by adding statement φ_3:

$$(\varphi_3) \quad c : \quad b_1 \rhd b_2.$$

Statement φ_1 says that Lucy unconditionally prefers to live in Esquirol over Capitole over Rangueil. Statement φ_2 says that if the apartment is expensive or in Rangueil, a shared place is preferred. Statement φ_3 says that if the apartment is costly, Capitole is better than Esquirol.

The dependency graphs for Lucy's gCP-nets N_1 and $N_2 = N_1 \cup \{\varphi_3\}$ are depicted in Figure 4.1.

An *outcome* is an assignment to each variable $X \in \mathcal{V}$ of a value in their domain $D(X)$. For instance, outcome $x_1 y_2 z_3 \ldots$ is such that variable X is assigned value x_1, Y is assigned y_2, Z is assigned z_3, and so on. In our example, an outcome would be a specific apartment of which we know location, price and presence of flatmates. The set of all outcomes is denoted by $\mathcal{O} = \Pi_{X_i \in \mathcal{V}} D(X_i)$. If $X \in \mathcal{V}$ is a variable, we write as $o[X]$ the value of outcome o on X; if $\mathcal{W} \subseteq \mathcal{V}$ is a set of variables, we write $o[\mathcal{W}]$ for the values of outcome o on $\mathcal{W}$. We also write $o \models \psi$ to say that outcome o satisfies propositional formula ψ (taking into account as explained before how we handle variables with non-binary values).

Preferences over outcomes are represented by a binary relation $>$ over $\mathcal{O}$. If $o_1 > o_2$ we say that o_1 *dominates* o_2 *with respect to* $>$, meaning that o_1 is preferred to o_2 in $>$. If $o_1 > o_2$ and $o_2 \not> o_1$, we say that o_1 *strictly dominates* o_2 *with respect to* $>$. In case $o_1 \not> o_2$ and $o_2 \not> o_1$ we say that o_1 *and* o_2 *are incomparable with respect to* $>$, and we write it as $o_1 \bowtie o_2$. The following notion of worsening flips allows us to define the semantics of conditional preference statements over binary relations on $\mathcal{O}$.

Definition 4.1. If $x_i, x_j \in D(X)$, $\varphi = \psi : x_i \rhd x_j$ is a conditional preference statement with respect to X, and o_1 and o_2 are two outcomes, then there is a *worsening flip from* o_1 *to* o_2 *sanctioned by* φ if $o_1 \models \psi$, $o_2 \models \psi$, $o_1[Y] = o_2[Y]$, for any $Y \in \mathcal{V} \setminus \{X\}$ and $o_1[X] = x_i$, $o_2[X] = x_j$.

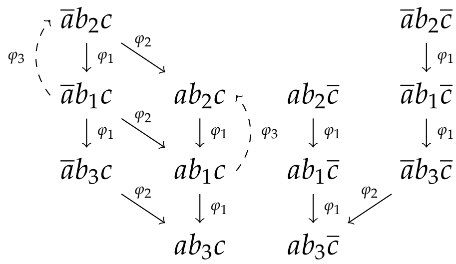

FIGURE 4.2: The induced model from Example 4.4. An arrow from o_1 to o_2 indicates a worsening flip sanctioned by N_1 and is labeled by the preference statement inducing it. Transitivity arrows are omitted.

Intuitively, we have a worsening flip from o_1 to o_2 sanctioned by $\varphi = \psi : (x_i \triangleright x_j)$ if outcomes o_1 and o_2 both satisfy precondition ψ, and they differ only in that variable X has value x_i in o_1 and value x_j in o_2. Namely, if ψ is true, all else being equal (*ceteris paribus*) it is better to have x_i than x_j. We say that there is an *improving flip from o_2 to o_1 sanctioned by φ* if there is a worsening flip from o_1 to o_2 sanctioned by φ. If N is a gCP-net, a worsening (improving) flip from o_2 to o_1 sanctioned by N is a worsening (improving) flip from o_2 to o_1 sanctioned by some $\varphi \in N$.

Definition 4.2. If N is a gCP-net and o, o' are two outcomes, then *o dominates o' with respect to N*, written $o >_N o'$, if there exists a sequence of outcomes $o_1, \dots, o_k$ such that $o_1 = o$, $o_k = o'$ and, for every $i \in \{1, \dots, k-1\}$, there exists a worsening flip from o_i to o_{i+1} sanctioned by N.

We call $>_N$ the *induced model of N* and often write $>_i$ instead of $>_{N_i}$ when clear from context. We say that a gCP-net is *consistent* if there is no chain of worsening flips starting with some outcome o and ending back on o. Since $>_N$ is transitive, this is equivalent to saying that there is no outcome o such that $o >_N o$.

Example 4.4. For the scenario described in Example 4.3 there are 12 possible outcomes. The outcome $o = ab_2\bar{c}$ refers to a reasonably priced and private apartment in Esquirol. For the variables A and B, $o[\{A, B\}] = ab_2$ refers to the values of outcome o on variables A and B. For the gCP-net $N_1 = \{\varphi_1, \varphi_2\}$ provided by Lucy, the induced model $>_1$, as well as the worsening flips induced by adding φ_3, is depicted in Figure 4.2. Adding statement φ_3 to N_1 results in an inconsistent gCP-net, as the induced model $>_2$ of $N_2 = N_1 \cup \{\varphi_3\}$ contains the sequence of worsening flips ab_2c, ab_1c, ab_2c, which implies that $ab_2c >_2 ab_2c$.

Observe that Definition 4.1 assumes that the *ceteris paribus* assumption holds even for the models of ψ in statements as $\psi : \pi$. For instance, in Example 4.3 we induce the rankings $\bar{a}b_3c > ab_3c$ and $\bar{a}b_3\bar{c} > ab_3\bar{c}$ from statement $\varphi_2 = c \vee b_3 : \bar{a} \triangleright a$, but not $\bar{a}b_3\bar{c} > ab_2c$, though both $\bar{a}b_3\bar{c}$ and ab_2c satisfy precondition $c \vee b_3$. While it induces less comparisons between outcomes, we believe this interpretation to be justified here as (i) it does not infer more than what is strictly warranted by the agent's statements, and (ii) it gives the agents more freedom to refine their orders without thereby creating inconsistencies, as the following example illustrates:

Example 4.5. Consider $\mathcal{V}$ as in Example 4.3, but Lucy now submits the gCP-net $N = \{\bar{a} \vee \bar{c} : b_3 \triangleright b_2\}$. If the semantics was not interpreted only *ceteris paribus*, we could derive that $\bar{a}b_3c >_N \bar{a}b_2\bar{c}$, meaning that Lucy prefers an expensive shared apartment in Rangueil to a cheap shared apartment in Esquirol. Suppose Lucy later adds some statements to N, leading to $N' = \{\bar{a} \vee \bar{c} : b_3 \triangleright b_2, b_2 \wedge \bar{c} : \bar{a} \triangleright a, b_3 \wedge c : a \triangleright$

DOMINANCE:	$o_1 >_N o_2$.
CONSISTENCY:	N is consistent.
WNON-DOM'ED:	o is weakly non-dominated in $>_N$.
NON-DOM'ED:	o is non-dominated in $>_N$.
DOM'ING:	o is dominating in $>_N$.
STR-DOM'ING:	o is strongly dominating in $>_N$.
∃NON-DOM'ED:	there is a non-dominated outcome in $>_N$.
∃DOM'ING:	there is a dominating outcome in $>_N$.
∃STR-DOM'ING:	there is a strongly dominating outcome in $>_N$.

TABLE 4.1: Reasoning tasks for a gCP-net N and outcomes o, o_1, o_2.

$\bar{a}, a \wedge c : b_2 \rhd b_3, a \wedge b_2 : \bar{c} \rhd c\}$. From N' we now derive $\bar{a}b_2\bar{c} >_{N'} \bar{a}b_3c$, i.e., N' is now inconsistent, which would not have happened under the *ceteris paribus* assumption.

4.1.2 Individual gCP-nets

We are interested in studying some key computational problems about notions of consistency, dominance, and optimality, rather than providing a full order over outcomes. Let $>$ be a binary relation on $\mathcal{O}$ and o an outcome: we say that o is *weakly non-dominated* if, for any outcome o', it holds that $o' > o$ implies $o > o'$. If there is no outcome o' such that $o' > o$, including $o' = o$, then we say that o is simply *non-dominated*. If $o > o'$ for all outcomes o', then o is a *dominating* outcome. If o is dominating as well as non-dominated, then it is *strongly dominating*. Observe that if o is weakly non-dominated, then it is possible that o is part of a cycle in $>$, as long as the cycle is not dominated by an outcome outside it. Likewise, if o is dominating, then it can be involved in a cycle in $>$. Table 4.1 lists the reasoning tasks of interest.

The computational complexity of these tasks for a single gCP-net has been established in previous work (Goldsmith et al., 2008)[1]. In particular, DOMINANCE, CONSISTENCY, WNON-DOM'ED, DOM'ING, STR-DOM'ING, ∃DOM'ING as well as ∃STR-DOM'ING have been shown to be PSPACE-complete in the general case, with the result for DOMINANCE holding even when the gCP-net N is consistent. The NON-DOM'ED problem is in P, while ∃NON-DOM'ED is NP-complete. If N is consistent, then the DOM'ING and ∃DOM'ING problems are in coNP. Moreover, the SELF-DOMINANCE problem, i.e., the problem of determining for a given gCP-net N and outcome o whether $o >_N o$, is also PSPACE-complete.

Let us consider an additional problem. In CP-nets, an acyclic dependency graph guarantees consistency (Boutilier et al., 2004). However, in gCP-nets this is not true anymore, as shown by the following example:

Example 4.6. Take a gCP-net N over $\mathcal{V} = \{A, B, C, D\}$, such that $D(A) = \{a\}$, $D(B) = \{b\}$, $D(C) = \{c\}$ and $D(D) = \{d_1, d_2, d_3\}$. Consider the following statements composing the gCP-net $N = \{a : d_1 \rhd d_2,\ b : d_2 \rhd d_3,\ c : d_3 \rhd d_1\}$. The dependency graph of N is acyclic, yet N induces the ordering $abcd_1 >_N abcd_2 >_N abcd_3 >_N abcd_1$, hence $abcd_1 >_N abcd_1$.[2]

Example 4.6 shows that a gCP-net can be acyclic, and yet contain some preference statements whose preconditions can be triggered by the same assignment, and for which a cycle in the ordering over outcomes is derived. Therefore, acyclic gCP-nets

[1]Importantly, the results by Goldsmith et al. (2008) assume that the precondition ψ of a gCP-net is represented in disjunctive normal form. As our results will depend on theirs, we follow the same assumption — which does not limit the expressivity of the precondition.

[2]An analogous example was pointed out by Wilson (2004).

may or may not be consistent: we thus define the computational problem of checking consistency for acyclic gCP-nets. Namely, given an acyclic gCP-net N for which we can assume, without loss of generality,[3] that all preferences π in the statements are over the same variable, to determine whether

$$a\text{CONSISTENCY:} \quad N \text{ is consistent.}$$

Intuitively, we look for a $N' \subseteq N$ for which the conjunction of all preconditions is satisfiable and the preference statements lead to a cycle over the outcomes.

Proposition 4.1. aCONSISTENCY is coNP-complete.

Proof. For membership, we show that the complement of aCONSISTENCY, i.e., deciding whether an acyclic gCP-net $N = \{\psi_1 : \pi_1, \ldots, \psi_n : \pi_n\}$ is inconsistent, is in NP. We guess a subset $N' \subseteq N$ and an assignment for the ψ_i's in N' (a partial assignment over $\mathcal{V}$). Then, for all possible completion of the guessed assignment with values in the π_i's we check if the statements $\psi_i : \pi_i$ lead to a cycle. For hardness we reduce from UNSAT. Consider an instance of UNSAT, i.e., a propositional formula φ whose unsatisfiability we want to check. Construct now a gCP-net $N = \{\varphi : a \triangleright a\}$ for A a fresh variable whose values $D(A) = \{a, \bar{a}\}$ do not occur in φ. If φ is unsatisfiable, then $\varphi : a \triangleright a$ is discarded when constructing $>_N$, and thus N is consistent, since it has no cycles. On the other hand, suppose φ is satisfiable. Then $\varphi : a \triangleright a$ leads to a cycle in $>_N$. □

4.1.3 *m*gCP-nets

An *mgCP-net* M is a multi-set $M = \langle N_1, \ldots, N_m \rangle$ of m gCP-nets over the set $\mathcal{V}$ of variables. We think of the semantics for mgCP-nets as a binary relation over outcomes, reflecting the domination relationships induced by the individual gCP-nets in M. For every mgCP-net M we thus define a binary relation $>_M$ on outcomes, called the *induced collective model of* M, which is obtained by aggregating the induced models of the gCP-nets in M, with notions such as dominance and consistency analogous to the ones for single gCP-nets (though transitivity is not guaranteed anymore).

Given an mgCP-net $M = \langle N_1, \ldots, N_m \rangle$ and two outcomes o_1, o_2, we define the following sets of agents supporting a certain dominance or incomparability relation between outcomes:

$$\begin{aligned} s_M^{o_1 > o_2} &= \{N_i \in M \mid o_1 >_i o_2\}, \\ s_M^{o_1 \bowtie o_2} &= \{N_i \in M \mid o_1 \not>_i o_2 \text{ and } o_2 \not>_i o_1\}. \end{aligned}$$

Given a gCP-net N and two outcomes o_1 and o_2, we define the *equivalence relation* $o_1 \approx_N^d o_2$ if $o_1 = o_2$, or $o_1 >_N o_2$ and $o_2 >_N o_1$. Namely, we partition the set of outcomes into equivalence classes. If o is an outcome, its equivalence class (i.e., the set of outcomes that includes o and, if they exist, all outcomes with which o forms a cycle in $>_N$) is called *the dominance class of o with respect to N* and is denoted by $[o]_N$, with the subscript omitted when clear from context. The dominance classes form a strict partial order, which we denote by $>_N^d$. We say that $[o_1]$ *dominates* $[o_2]$ *with respect to* $>_N^d$, written $[o_1] >_N^d [o_2]$, if $o_1 >_N o_2$ and $o_2 \not>_N o_1$ for o_1 and o_2 representative elements of their equivalence class. A dominance class $[o]$ is *non-dominated with respect to* $>_N^d$ if there is no dominance class which dominates it with respect to $>_N^d$.

[3]We can ignore statements on other variables as they would not be part of this inconsistency cycle.

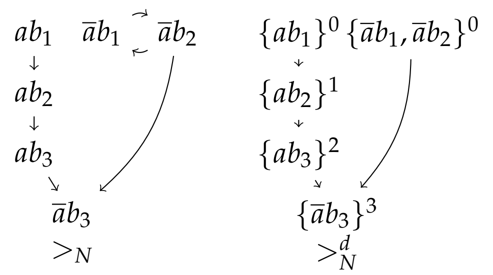

FIGURE 4.3: Dominance classes and $r_N^{\mathtt{lp}}$ (in superscripts).

A *ranking function r with respect to a gCP-net N* (mgCP-net M, respectively) assigns to every outcome o a non-negative number $r_N(o)$ ($r_M(o)$, respectively). We sometimes write $r_i(o)$ instead of $r_{N_i}(o)$ for simplicity. The *longest path rank function* r_N^{lp} assigns to an outcome o the length of the longest path from $[o]$ to a non-dominated dominance class in $>^d_N$.[4] Observe that this always exists because all the outcomes forming a cycle belong to the same equivalence class.

Example 4.7. Consider $\mathcal{V} = \{A, B\}$ where $D(A) = \{a, \overline{a}\}$ and $D(B) = \{b_1, b_2, b_3\}$. Take a gCP-net $N = \{\top : b_1 \rhd b_2 \rhd b_3, b_3 : a \rhd \overline{a}, \overline{a} : b_2 \rhd b_1\}$. Figure 4.3 shows the induced model $>_N$, the dominance classes, the strict partial order $>^d_N$ on dominance classes, and the longest path ranks assigned by $r^{\mathtt{lp}}$.

We now define the semantics of mgCP-nets, generalizing the semantics defined for mCP-nets (Rossi et al., 2004; Lukasiewicz and Malizia, 2016).

Definition 4.3. If $M = \langle N_1, \ldots, N_m \rangle$ is an mgCP-net, the *Pareto relation* $>^P_M$, *majority relation* $>^{maj}_M$, *maximality relation* $>^{max}_M$ and *rank relation* $>^r_M$ *with respect to* M are defined, for any o_1 and o_2, as follows:

$$
\begin{array}{ll}
o_1 >^P_M o_2 & \text{if } o_1 >_i o_2 \text{, for every } N_i \in M; \\
o_1 >^{maj}_M o_2 & \text{if } o_1 >_i o_2 \text{, for } \left\lceil \frac{m+1}{2} \right\rceil N_i \in M; \\
o_1 >^{max}_M o_2 & \text{if } |s_M^{o_1 > o_2}| > max\{|s_M^{o_2 > o_1}|, |s_M^{o_1 \bowtie o_2}|\}; \\
o_1 >^r_M o_2 & \text{if } r_M(o_1) \leq r_M(o_2).
\end{array}
$$

A dominance relation is preserved in Pareto semantics if all agents were unanimous about it in their individual gCP-nets. For majority semantics, it is sufficient to have a majority of agents with that dominance relation in their gCP-nets. For max semantics, a dominance relation is preserved if there are more agents wanting it than agents against or indifferent towards it. Finally, a dominance relation holds in rank semantics if the rank of an outcome is better (lower) than the rank of the other.

If S is a semantics, we call $>^S_M$ the *S-induced (collective) model of M*. Given an mgCP-net M and a semantics S, if $o_1 >^S_M o_2$ we say that o_1 *S-dominates* o_2 *with respect to M*. We say that M is *S-consistent*, or simply *consistent*, if there is no set of outcomes $o_1, \ldots, o_k$ such that $o_1 >_M \ldots >_M o_k >_M o_1$. S-non-dominated, S-weakly non-dominated, S-dominating and S-strongly dominating outcomes for an mgCP-net M are defined analogously as for individual gCP-nets. In the following we will use a specific type of rank relation r_M, obtained by summing up the $r^{\mathtt{lp}}$ score for all agents in M. Thus, for an mgCP-net $M = \langle N_1, \ldots, N_m \rangle$, we will take the rank of o with respect to M to be $r_M(o) = \sum_{N_i \in M} r_i^{\mathtt{lp}}(o)$.

[4] The rank semantics has been used before to aggregate acyclic CP-nets (Rossi et al., 2004; Lukasiewicz and Malizia, 2016). However, their definition differs from ours since acyclic CP-nets feature no cycles between outcomes.

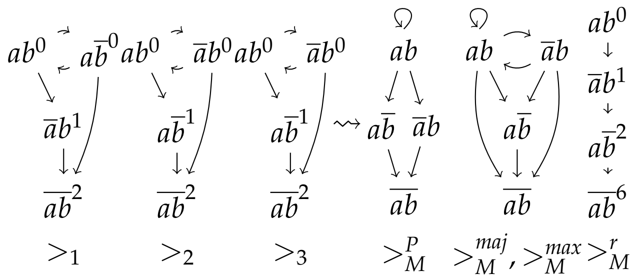

FIGURE 4.4: Individual and collective semantics for the mgCP-net M of Example 4.8. Edges in the induced models $>_{1-3}$ indicate worsening flips given by the gCP-nets N_{1-3}, respectively, whereas edges in the induced collective models indicate domination relations obtained through aggregation. Arrows inferred through transitivity in $>^P_M$ and $>^r_M$ are omitted; since $>^{maj}_M$ and $>^{max}_M$ are not guaranteed to be transitive, every domination relation in them is explicit. Longest path ranks assigned by $r_i^{\mathtt{lp}}$ are shown as a superscript.

Example 4.8. Alice, Bob and Carol are looking for a shared apartment in Toulouse. The online service of Example 4.3 can now handle preferences submitted by multiple agents. The variables are $\mathcal{V} = \{A, B\}$ as in Example 4.3, though for simplicity we now assume each variable is binary. Alice submits $N_1 = \{\top : a \rhd \overline{a}, \top : b \rhd \overline{b}, a : \overline{b} \rhd b\}$, while Bob and Carol submit $N_2 = N_3 = \{\top : a \rhd \overline{a}, \top : b \rhd \overline{b}, b : \overline{a} \rhd a\}$, with the corresponding 3gCP-net being $M = \langle N_1, N_2, N_3 \rangle$. The induced models $>_1$, $>_2$ and $>_3$ together with the induced collective models $>^P_M$, $>^{maj}_M$, $>^{max}_M$ and $>^r_M$, are shown in Figure 4.4. None of the induced individual models has a strongly dominating outcome, though ab is *weakly* non-dominated, as well as dominating, in each, and thus a prime candidate for being at the top of the list of suggested apartments. Since ab self-dominates in each of the individual induced models, this domination relation carries over to the induced collective models. The rank of an outcome in $>^r_M$ is computed by summing up its ranks in the individual induced models. Thus, $r_M(\overline{a}\overline{b}) = \sum_{i=1}^{3} r_i^{\mathtt{lp}}(\overline{a}\overline{b}) = 6$.

Observe that for different outcomes o_1 and o_2, if $o_1 >^P_M o_2$, then $o_1 >^{maj}_M o_2$, and if $o_1 >^{maj}_M o_2$, then $o_1 >^{max}_M o_2$. The fact that the semantics are not mutually exclusive suggests that they can be used alongside each other, e.g., to deliver results when the Pareto semantics is undecided. Secondly, the *maj*- and *max*-induced models $>^{maj}_M$ and $>^{max}_M$, respectively, are not guaranteed to be transitive. However, the Pareto-induced model $>^P_M$ is transitive, since if $o_1 >^P_M o_2$ and $o_2 >^P_M o_3$, then $o_1 >^P_i o_3$, for every $N_i \in M$, and thus $o_1 >^P_M o_3$. Hence, if there is a set of outcomes such that $o_1 >^P_M \dots >^P_M o_k >^P_M o_1$, then we have that $o_1 >^P_M o_1$. It follows that the condition for Pareto-consistency of mgCP-nets coincides with consistency for individual gCP-nets, i.e., M is Pareto-consistent if and only if there is no outcome o for which $o >^P_M o$. By definition we cannot have $o >^{max}_M o$, though other forms of inconsistency are possible. Finally, the longest-path induced by $>^r_M$ is a total pre-order on outcomes, since every outcome gets a rank in $>^r_M$ and two outcomes can have the same rank. Therefore, $>^r_M$ is transitive by design.

	single gCP-nets	*m*gCP-nets			
		Pareto	*maj*	*max*	*r*
$S-$DOMINANCE	PSPACE-c	PSPACE-c	PSPACE-c	PSPACE-c	PSPACE-h
$S-$CONSISTENCY	PSPACE-c	PSPACE-c	PSPACE-h	PSPACE-h	—
$S-$wNON-DOM'ED	PSPACE-c	PSPACE-c	PSPACE-c	PSPACE-h	PSPACE-h
$S-$NON-DOM'ED	in P	PSPACE-c	PSPACE-c	in PSPACE	—
$S-$DOM'ING	PSPACE-c	PSPACE-c	PSPACE-c	PSPACE-c	PSPACE-h
$S-$STR-DOM'ING	PSPACE-c	PSPACE-c	PSPACE-c	PSPACE-c	—
$S-\exists$NON-DOM'ED	NP-c	PSPACE-c	NP-h (*)	NP-h	—
$S-\exists$DOM'ING	PSPACE-c	PSPACE-c	PSPACE-c	PSPACE-c	—
$S-\exists$STR-DOM'ING	PSPACE-c	PSPACE-c	PSPACE-c	PSPACE-c	—

TABLE 4.2: Complexity results for single gCP-nets (by Goldsmith et al., 2008) and *m*gCP-nets; '-c' and '-h' are short for -complete and -hard, respectively, for a given class; '—' means that the answer is trivial; 'S' stands for the corresponding semantics; '(*)' indicates the presence of a non-tight upper bound.

4.2 Computational Complexity

We will study here the reasoning tasks of Table 4.1, now focusing on *m*gCP-nets and on the semantics introduced in Section 4.1.3. An overview of our results, alongside existing results for single gCP-nets, is given in Table 4.2. We also recall that PSPACE = coPSPACE and that, by Savitch's Theorem, NPSPACE = PSPACE (Arora and Barak, 2009): both facts will be used in our proofs.

4.2.1 Pareto semantics

For Pareto semantics (abbreviated by P), all the tasks considered turn out to be PSPACE-complete. Intuitively, we cannot leverage the polynomial algorithm for checking whether an outcome o_1 is non-dominated in individual gCP-nets: if o_1 is found to be dominated by o_2 in some gCP-net N_i from M, this is of no help in deciding whether o_1 is dominated in $>^P_M$. The outcome o_2 would have to dominate o_1 in *every* $>_i$ for this to hold. Our first two results will thus determine a jump in complexity with respect to the single-agent case from P and NP, respectively, to PSPACE; while the rest of our results will stay in PSPACE.

Theorem 4.1. The $P-$NON-DOM'ED problem for *m*gCP-nets is PSPACE-complete.

Proof. We show that the complement, i.e., checking whether o is dominated in $>^P_M$, is PSPACE-complete. For membership, guess an outcome o' and check if $o' >^P_M o$, which amounts to at most m PSPACE tasks.

For hardness, we do a reduction from the SELF-DOMINANCE problem for single gCP-nets. Thus, given a gCP-net N and an outcome $o = xyz\cdots$, take the 2gCP-net $M = \langle N_1, N_2 \rangle$, where $N_1 = N$ and $N_2 = \{(y \wedge z \wedge \dots) : x \triangleright x\}$: i.e., N_2 is such that the only induced comparison in $>_2$ is one in which o self-dominates. The claim, then, is that o is self-dominating in $>_N$ iff o is dominated in $>^P_M$. To see this, assume first that $o >_N o$, hence $o >_1 o$. Since $o >_2 o$ by design, it follows that $o >^P_M o$. Conversely, if o is dominated in $>^P_M$, then, as o is not dominated by another outcome in $>_2$, this can only be because it is dominated by itself in $>^P_M$, and thus it self-dominates in $>_N$. □

Theorem 4.2. The $P-\exists$NON-DOM'ED problem for *m*gCP-nets is PSPACE-complete.

Proof. For membership, we guess an outcome o and ask the PSPACE−complete problem P−NON-DOM'ED for M and o, where M is the given mgCP-net. This is in NPSPACE, and as NPSPACE = PSPACE it is thus in PSPACE.

For hardness, we reduce from P−NON-DOM'ED. Consider an instance with mgCP-net $M = \langle N_1, \dots, N_m \rangle$ and outcome $o = v_1 \dots v_k$ for $\mathcal{V} = \{V_1, \dots, V_k\}$ and $v_i \in D(V_i)$ for $i \in \{1, \dots, k\}$. We now construct a mgCP-net $M' = \langle N'_1, \dots, N'_m \rangle$, where $N'_i = N_i \cup \{\top : v'_i \triangleright v'_i \mid v'_i \in D(V_i), v'_i \neq v_i\}$. The intuitive idea is that any outcome $o' \neq o$ is now self-dominating in $>_{N'_i}$, and hence self-dominating in $>^P_{M'}$. Thus, if there is a non-dominated outcome at all in $>^P_{M'}$, then it must be o: and this only happens if o is non-dominated in M. In other words, o is non-dominated in M if and only if there is a non-dominated outcome in M', concluding the proof. □

Theorem 4.3. The problems P−STR-DOM'ING, P−∃STR-DOM'ING, P−DOMINANCE, P−CONSISTENCY, P−WNON-DOM'ED, P−DOM'ING, P−∃DOM'ING are all PSPACE-complete.

Proof. For hardness, it suffices to reduce from all the corresponding problems for a single gCP-net.

For membership, consider an mgCP-net $M = \langle N_1, \dots, N_m \rangle$. For P−DOMINANCE, we have to check whether $o_1 >_i o_2$, for every $N_i \in M$. This amounts to solving m PSPACE tasks, which is also in PSPACE. For P−CONSISTENCY, recall that M being Pareto-consistent is equivalent to $o \not>^P_M o$, for any outcome o, i.e., $o \not>_i o$, for some $N_i \in M$. We thus ask of every outcome whether $o >_i o$, for all $N_i \in M$, which amounts to a (potentially exponential) number of PSPACE tasks. A similar algorithm works for P−WNON-DOM'ED, where we need to take every outcome o' and check whether $o' >_i o$ and $o \not>_i o'$, for every $N_i \in M$. Existence of such an outcome o' implies that o is not weakly non-dominated in $>_M$, while lack of existence implies the contrary. For S−DOM'ING, we have that o is a dominating outcome iff $o >^P_M o'$, for any outcome o'. This is equivalent to o being dominating in every induced model $>_i$, for $N_i \in M$. Determining this involves solving m PSPACE tasks. For P−STR-DOM'ING we have to check that o is dominating in every $>_i$, for $N_i \in M$ and, in addition, that o is strongly dominating in at least one $>_i$. In fact, suppose o was dominating in every $>_i$, but strongly dominating in neither of them: then $o >_i o$ for all $N_i \in M$, and thus $o >^P_M o$, which means that o is not strongly dominating in $>^P_M$. Checking whether o is strongly-dominating in some $>_i$ is in PSPACE, thus our task is in PSPACE. For P−∃DOM'ING and P−∃STR-DOM'ING, respectively, we can go through every outcome and ask whether it is dominating and strongly dominating, respectively, in $>^P_M$. This consists entirely of PSPACE tasks. □

4.2.2 Majority Semantics

The majority semantics *maj* is inspired by the well-known majority rule in preference aggregation. By definition 4.3, when the number of agents is even we get a *strict* version of majority. The results can be however easily adapted to a *weak* majority. Our first result includes all problems which remain PSPACE-complete when moving from a single gCP-net to many. Then, we have a result whose complexity increases from polynomial to PSPACE-complete from gCP-nets to mgCP-nets. Finally, we have hardness results which again follow from the single-agent case.

Theorem 4.4. Problems *maj*−DOMINANCE, *maj*−∃DOM'ING, *maj*−STR-DOM'ING, *maj*−DOM'ING, *maj*−WNON-DOM'ED, *maj*−∃STR-DOM'ING are PSPACE-complete.

Proof. PSPACE-hardness is inherited from the corresponding single-agent problems, by considering a mgCP-net with $m = 1$. We now establish membership.

For $maj-$DOMINANCE, consider an algorithm counting whether there are more than $\lceil \frac{m+1}{2} \rceil$ agents in M such that for each i it holds that $o_1 >_i o_2$. If this is the case answer yes, and no otherwise. This algorithm need to keep track of the yes/no answer of at most m PSPACE problems, and hence it is in PSPACE. For $maj-$DOM'ING, consider an algorithm checking for each $o' \in \mathcal{O}$ whether there is a set S of agents, such that $|S| \geq \lceil \frac{m+1}{2} \rceil$, where each agent $i \in S$ has $o >_i o'$. Hence, if for some o' such a set S is found, the algorithm answers yes, and no otherwise. We thus need to repeat at most $|\Pi_{X \in \mathcal{V}} D(X)|$ times (for all possible outcomes), at most m (for all agents) dominance PSPACE tasks. For $maj-$STR-DOM'ING, consider an algorithm which solves the problems $maj-$DOM'ING and $maj-$NON-DOM'ED, that are both in PSPACE, and answers yes if and only if for both problems it gets a positive answer. For $maj-$wNON-DOM'ED, given an outcome o we do the following procedure for all outcomes o' in M. We check if *(a)* there is a majority of agents for which $o' >_i o$ (at most m PSPACE tasks) and *(b)* there is a majority of agents for which $o >_i o'$ (again, at most m PSPACE tasks). If we find an outcome o' such *(a)* is true while *(b)* is false, the algorithm gives a negative answer (o is weakly dominated), and positive otherwise. For $maj-\exists$DOM'ING and $maj-\exists$STR-DOM'ING, consider an algorithm solving the problems $maj-$DOM'ING and $maj-$STR-DOM'ING, respectively, for all outcomes $o \in \mathcal{O}$, and that says yes if at least for one instance the answer is positive. This amounts to solving a (possibly exponential) number of PSPACE tasks. □

Theorem 4.5. The $maj-$NON-DOM'ED problem for mgCP-nets is PSPACE-complete.

Proof. Let $\overline{maj-\text{NON-DOM'ED}}$ be the complement of $maj-$NON-DOM'ED, i.e., the problem asking whether there is some outcome o' such that $o' >_M^{maj} o$, for given o and M. Consider the algorithm that for every outcome o' checks whether for at least $\lceil \frac{m+1}{2} \rceil$ (and at most m) agents i in M it is the case that $o' >_i o$, and it outputs yes if so. This problem is in PSPACE and thus its complement $maj-$NON-DOM'ED is in coPSPACE, which means that $maj-$NON-DOM'ED is in PSPACE.

Proof of hardness is identical to that of Theorem 4.1, since in maj semantics for $m = 2$ majority corresponds to the total number of agents. □

Theorem 4.6. Problem $maj-$CONSISTENCY is PSPACE-hard; $maj-\exists$NON-DOM'ED is NP-hard and in PSPACE.

Proof. The hardness reductions are from the single-agent problems where $m = 1$. For $maj-\exists$NON-DOM'ED, membership in PSPACE comes from the algorithm asking the PSPACE$-$complete problem $maj-$NON-DOM'ED for all outcomes o, and giving a positive answer if at least one is non-dominated. □

4.2.3 Max Semantics

The semantics *max* refines *maj* by taking into account also incomparabilities and thus expressing a relative majority. The *max* semantics does not admit cycles of length at most 2. In fact, for $>_M^{max}$ to be inconsistent there would need to be two outcomes o_1 and o_2 such that $o_1 >_M^{max} o_2$ and $o_2 >_M^{max} o_1$, implying a contradiction between $|s_M^{o_1 > o_2}| > |s_M^{o_2 > o_1}|$ and $|s_M^{o_2 > o_1}| > |s_M^{o_1 > o_2}|$.

Theorem 4.7. Problems $max-$DOMINANCE, $max-$DOM'ING, $max-$STR-DOM'ING, $max-\exists$DOM'ING and $max-\exists$STR-DOM'ING are PSPACE-complete.

Proof. Hardness is obtained from the corresponding single-agent problems, by considering a mgCP-net with $m = 1$. We thus focus on PSPACE-membership.

For $max-$DOMINANCE, consider an algorithm that stores $|s_M^{o_1 > o_2}|$ as supp, i.e., the number of agents i in M for whom $o_1 >_i o_2$. Observe that supp $\leq m$. Then, the algorithm stores $|s_M^{o_2 > o_1}|$ as opp, i.e., the number of agents in M such that $o_2 >_i o_1$. Again, opp $\leq m$. Then, it stores $m -$ supp $-$ opp as inc; i.e., the number of agents in M for whom o_1 and o_2 are incomparable. Finally, if inc $\geq$ opp and supp $\geq$ inc, or if inc $\leq$ opp and supp $\geq$ opp, the algorithm answers yes, and no otherwise. For $max-$DOM'ING, consider an algorithm that for any $o' \in \mathcal{O}$ it stores as supp the number of agents i in M such that $o' >_i o$, then the number of agents k in M such that $o >_k o'$ as opp, and the number of agents considering o and o' as incomparable in inc $= m -$ supp $-$ opp. Analogously to $max-$DOMINANCE, the algorithm checks if inc $\geq$ opp and supp $\geq$ inc, or if inc $\leq$ opp and supp $\geq$ opp, in which cases it answers yes, and no otherwise. The algorithm solves a (potentially exponential) number of PSPACE tasks, which is in PSPACE. For $max-$STR-DOM'ING, it suffices to design an algorithm which runs the algorithms for $max-$DOM'ING and $max-$NON-DOM'ED (both in PSPACE), and which answers yes if and only if both tasks return yes. For $max-\exists$DOM'ING and $max-\exists$STR-DOM'ING, consider an algorithm asking the problems $max-$DOM'ING and $max-$STR-DOM'ING for all outcomes $o \in \mathcal{O}$, and answering yes if for at least one of the outcomes the answer is yes. This amounts to solving a (possibly exponential) number of PSPACE tasks. □

Theorem 4.8. Problems $max-$CONSISTENCY and $max-$wNON-DOM'ED are PSPACE-hard, while $max-\exists$NON-DOM'ED is NP-hard.

Proof. We reduce from the corresponding single-agent problems where $m = 1$. □

Theorem 4.9. The $max-$NON-DOM'ED problem for mgCP-nets is in PSPACE.

Proof. Consider an algorithm that checks for all $o' \in \mathcal{O}$ whether $o' >_M^{max} o$, which is a PSPACE problem according to Theorem 4.7, and it answers yes if every one of these tasks gives a no answer. □

Observe that we cannot reduce from SELF-DOMINANCE as done in the proof of Theorem 4.1 for $max-$NON-DOM'ED since $>_M^{max}$ has no self-dominating outcomes by definition.

4.2.4 Rank semantics

Since the r-induced model of an mgCP-net M is a total preorder, the notions of strongly dominating outcome and non-dominated outcome are vacuous, as $o >_M^r o$ for all o. Furthermore, weakly non-dominated outcomes always exist, and they coincide with dominating outcomes. Thus, the $r-$CONSISTENCY, $r-$NON-DOM'ED, $r-$STR-DOM'ING, $r-\exists$NON-DOM'ED, $r-\exists$DOM'ING, and $r-\exists$STR-DOM'ING problems have trivial answers. We will focus on $r-$DOMINANCE and $r-$wNON-DOM'ED, with the understanding that a solution to the latter problem is also a solution to the $r-$DOM'ING problem. We first show, using results on single gCP-nets and some intermediary results that finding the rank of an outcome and comparing two outcomes with respect to their rank is PSPACE-hard even in the single gCP-case (Proposition 4.10) and this carries over to the multi-agent case (Theorem 4.11).

Lemma 4.1. If o and o' are two outcomes and their Hamming distance, i.e., the number of variables on which they differ, is $d_H(o, o') = p$, then there exists a gCP-net $N(o, o')$ such that $|N| = p$ and a sequence of length p of worsening flips from o to o'.

Proof. Since worsening flips exist only between outcomes o_i and o_j differing on just one variable, we can create a chain of outcomes $o_1, \ldots, o_p$ of length p of worsening flips sanctioned by the following gCP-net. Let $o = v_1 \ldots v_k a_1 \ldots a_p$ and $o' = v_1 \ldots v_k a'_1 \ldots a'_p$ such that there are p variables $A_1, \ldots, A_p$ where $o[A_i] \neq o'[A_i]$ for each i and on the V_i variables the two outcomes agree. Consider now the statements:

$$\begin{aligned} v_1 \wedge \cdots \wedge v_k \wedge a_2 \wedge \cdots \wedge a_p : a_1 \rhd a'_1 \\ v_1 \wedge \cdots \wedge v_k \wedge a'_1 \wedge a_3 \wedge \cdots \wedge a_p : a_2 \rhd a'_2 \\ \cdots \\ v_1 \wedge \cdots \wedge v_k \wedge a'_1 \wedge \cdots \wedge a'_{p-1} : a_p \rhd a'_p \end{aligned}$$

The gCP-net $N(o, o')$ is simply the collection of all the statements above. □

If o is an outcome over a set of variables $\mathcal{V}$ and $X \notin \mathcal{V}$ is a variable such that $D(X) = \{x, \overline{x}\}$, we write ox^* for the outcome o' over $\mathcal{V} \cup \{X\}$ such that $o'[Y] = o[Y]$, for any $Y \in \mathcal{V}$, and $o'[X] = x^*$, for $x^* \in D(X)$. We can think of o' as o concatenated with x^*. We now show that given an outcome o of rank 0 in N, we can construct a new gCP-net N' where (a suitable copy of) o has rank k, for any $k \geq 0$.

Lemma 4.2. If N_1 is a gCP-net and o is an outcome over variables in $\mathcal{V}$, and $k \geq 0$, then there exists a gCP-net N_2 over variables $\mathcal{V} \cup \{X_1, \ldots, X_k\}$, where all the variables in $\{X_1, \ldots, X_k\}$ are binary and none of them occurs in $\mathcal{V}$, such that $r_1^{\mathtt{lp}}(o) = 0$ iff $r_2^{\mathtt{lp}}(ox_1 \ldots x_k) = k$.

Proof. Consider the following gCP-net:

$$\begin{aligned} N_2 = N_1 \cup \{ & \bigwedge_{X \in \mathcal{V}} o[X] \wedge \bigwedge_{i=1}^{k-1} x_i : \overline{x_k} \rhd x_k, \bigwedge_{X \in \mathcal{V}} o[X] \wedge \bigwedge_{i=1}^{k-2} x_i \wedge \overline{x_k} : \overline{x_{k-1}} \rhd x_{k-1}, \\ & \bigwedge_{X \in \mathcal{V}} o[X] \wedge \bigwedge_{i=1}^{k-3} x_i \wedge \bigwedge_{j=k}^{k-1} \overline{x_j} : \overline{x_{k-2}} \rhd x_{k-2}, \ldots, \\ & \bigwedge_{X \in \mathcal{V}} o[X] \wedge \bigwedge_{i=2}^{k} \overline{x_i} : \overline{x_1} \rhd x_1 \}. \end{aligned}$$

It holds that $r_1^{\mathtt{lp}}(o) = 0$ iff $r_2^{\mathtt{lp}}(ox_1 \ldots x_k) = k$. See Example 4.9 for an illustration of the construction. □

Example 4.9. Take a gCP-net N_1 over a single binary variable A, with $N_1 = \{\top : a \rhd \overline{a}\}$. Suppose we want to construct N_2 such that (a suitable copy of) outcome a has rank 2. We add two new binary variables, X and Y, and define N_2 over variables A, X and Y. As for proof of Lemma 4.2, we first import all the conditional preference statements from N_1: by the *ceteris paribus* semantics, this creates four copies of the dominance relation from $>_1$, one for each assignment to variables X and Y. We then add statements $a \wedge x : \overline{y} \rhd y$ and $a \wedge \overline{y} : \overline{x} \rhd x$, which create a chain improving flips from axy to $a\overline{x}\overline{y}$ of length 2 (see Figure 4.5). It is easy to see now that $r_1^{\mathtt{lp}}(a) = 0$ iff $r_2^{\mathtt{lp}}(axy) = 2$.

Theorem 4.10. For a gCP-net N, two outcomes o_1 and o_2, and $k \geq 0$, then it is PSPACE-hard to check:

(a) whether $r_N^{\mathtt{lp}}(o_1) = k$;

(b) whether $r_N^{\mathtt{lp}}(o_1) = r_N^{\mathtt{lp}}(o_2)$;

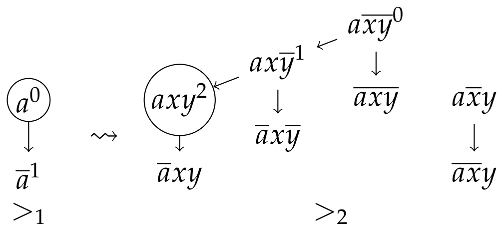

FIGURE 4.5: The induced models $>_1$ and $>_2$ from Example 4.9; the outcome whose rank goes from 0 to 2 is circled.

(c) whether $r^{\texttt{lp}}_N(o_1) < r^{\texttt{lp}}_N(o_2)$.

Proof. For (a), observe that o_1 is weakly non-dominated in $>_N$ if and only if it has $r^{\texttt{lp}}_1(o_1) = 0$. Then, Lemma 4.2 gives us a reduction from wNON-DOM'ED for individual gCP-nets under $>_N$.

For (b) we do a reduction from DOMINANCE. Let N_1 be a gCP-net, we define $N_2 = N_1 \cup N(o_2, o_1)$, where $N(o_2, o_1)$ is a gCP-net constructed as in Lemma 4.1, which induces a dominance relation from o_2 to o_1. Note that if $o_1 >_N o_2$, for some gCP-net N, then o_1 must be on a path from $[o_2]$ to a non-dominated class in $>^d_N$, and thus $r^{\texttt{lp}}_N(o_1) \leq r^{\texttt{lp}}_N(o_2)$. It follows now that $o_1 >_1 o_2$ iff $r^{\texttt{lp}}_2(o_1) = r^{\texttt{lp}}_2(o_2)$.

For (c), we do a reduction from DOMINANCE. Let N_1 be a gCP-net and o_1, o_2 outcomes over variables in $\mathcal{V}$. We now construct a gCP-net N_2 over variables in $\mathcal{V} \cup \{X\}$, where $X \notin \mathcal{V}$ is binary, such that $o_1 >_1 o_2$ iff $r^{\texttt{lp}}_2(o_1x) < r^{\texttt{lp}}_2(o_2x)$. Namely, given a gCP-net N_1, we can create a gCP-net N_2, where a dominance relation in $>_1$ is reflected by a difference between the ranks of two (copies of the) outcomes in $>_2$. First, note that $d_H(o_1, o_2) = p$ implies $d_H(o_1x, o_2x) = p$. By Lemma 4.1 we construct the gCP-net $N(o_2\overline{x}, o_1x)$ of size p which sanctions a chain of worsening flips from $o_2\overline{x}$ to o_1x. We now take $N_2 = N_1 \cup N(o_2\overline{x}, o_1x)$. If $o_1 >_1 o_2$, then $r^{\texttt{lp}}_2(o_1x) < r^{\texttt{lp}}_2(o_2x)$. Conversely, if $o_1 \not>_1 o_2$, then $r^{\texttt{lp}}_2(o_1x) \geq r^{\texttt{lp}}_2(o_2x)$, since o_1 inherits all of o_2's ancestors, and thus its rank is at least as great. $\square$

The following example illustrates the point (c) in the previous proof:

Example 4.10. Take $N_1 = \{\overline{b} : a \triangleright \overline{a}\}$ and we are interested in outcome ab and $\overline{ab}$ (circled in Figure 4.6). Namely, we want to construct N_2 such that $ab >_1 \overline{ab}$ iff $r^{\texttt{lp}}_2(abx) < r^{\texttt{lp}}_2(\overline{ab}x)$. Taking N_2 as described in the proof of Theorem 4.10 we get, via the *ceteris paribus* semantics, an extra copy of every dominance relation in $>_1$, one for x and one for $\overline{x}$. The added preference statements take the dominance block $a\overline{bx} >_2 \overline{abx}$ and place it on top of abx: since this block is a copy of the block $a\overline{b}x >_2 \overline{ab}x$, we get that abx inherits all of $\overline{ab}x$'s ancestors. Thus, the rank of abx in $>_2$ can be smaller than the rank of $\overline{ab}x$ if and only if the path of ancestors of $\overline{ab}x$ goes through abx, i.e., only if $abx >_2 \overline{ab}x$. But this means that this edge must have been in $>_1$ originally (which is not the case here).

From Theorem 4.10 we get that for a single gCP-net weak non-dominance (point (a) with $k = 0$) and domination (either (b) or (c)) for rank semantics are PSPACE-hard.

Theorem 4.11. The r−DOMINANCE and r−wNON-DOM'ED problems for mgCP-nets are PSPACE-hard.

Proof. Inherited from the single gCP-net case. $\square$

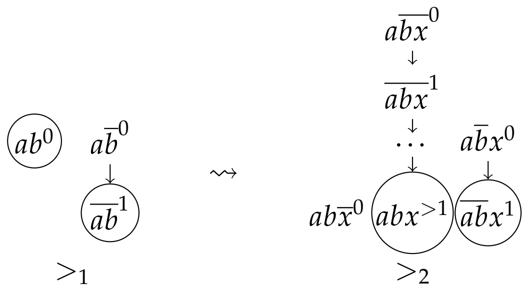

FIGURE 4.6: The induced models from Example 4.10.

4.3 Conclusions

In this chapter we studied mgCP-nets, i.e., profiles of generalized CP-nets (Goldsmith et al., 2008) aggregated under semantics adapted from the literature of mCP-nets (Rossi et al., 2004; Lukasiewicz and Malizia, 2016). Our motivation has been illustrated by the Examples 4.1 and 4.8, where an online renting service allows agents to only partially specify their preconditions for conditional preference statements. We have hereby bridged the line of research on gCP-nets, which provides a framework for expressing general types of preferences under the *ceteris paribus* semantics, and that on the aggregation of CP-nets coming from different agents. In particular, the *Pareto* semantics we study here is reminiscent of the unanimity axiom of Section 3.2.3 and the majority semantics of the *Maj* rule of Section 3.4.2.

As the main barriers to implementation are computational, our focus has been that of studying a variety of complexity problems for mgCP-nets. In particular, we analyzed the consequences of moving from one to multiple agents with respect to the complexity of some known consistency and optimality problems. Our findings show that these problems fall mostly in the PSPACE complexity class. On the positive side, for most cases the complexity does not increase when moving to the multi-agent case, one exception being the problems for non-dominated outcomes.

The results of this chapter lead to many interesting directions for future work. Analogously to what we did at the end of Section 3.3.4 for the complexity results of the winner determination problem for majority rules in goal-based voting, we could explore restrictions (possibly syntactical) on gCP-nets to make reasoning tasks tractable. Moreover, alternative ways to aggregate individual gCP-nets could be explored, such as voting directly on formulas as in judgment aggregation (Endriss, 2016), or adapting rules from goal-based voting as the ones of Section 3.1.2.

Part II

Strategic Behavior

Chapter 5

Logic, Goals, and Strategic Agents

In the second part of this thesis we study some game-theoretic and strategic problems in settings where agents have individual goals represented in a logical language. In particular, we will start in Chapter 6 by studying the strategic component of goal-based voting (introduced in Chapter 3): i.e., an agent i will not be assumed to be truthful anymore and she will submit a goal γ_i' different from her honest goal γ_i if by doing so she can obtain a better result of the aggregation process.

A fundamental example in the literature on propositional goals in a strategic setting is that of Boolean games (Harrenstein et al., 2001; Bonzon et al., 2006). In such games, each player i wants to achieve a goal γ_i represented by a formula of propositional logic and her preferences are dichotomous. Goals are expressed over a set of propositional variables Φ which is partitioned in as many subsets as the number of agents and each agent has *exclusive* control over the variables in her subset $\Phi_i \subseteq \Phi$. The strategic actions at her disposal consist in affecting the truth values of the variables she controls (i.e., those in Φ_i). The following simple example illustrates a situation captured by a Boolean game:

Example 5.1. Consider three agents, Diana, Bacchus and Flora, who want to organize a potluck, i.e., a meal where everyone brings something to share. In order to avoid the common scenario where everyone just brings a bag of chips and a beer, they decide to make each one of them in charge of a specific category. Diana will bring the meat (either steak or chicken), Bacchus will bring the wine (either white or red), and Flora will bring the appetizer (either spring rolls or fried squash blossom). Each one of them can only control the part of the meal that they are assigned to: e.g., Diana cannot decide if they will be drinking white or red wine. However, each one of them has some goal regarding the meal as a whole. In particular, Diana would like to have red wine if they have steak, Bacchus would be happy if at the meal there will be chicken or spring rolls, and Flora would like to have white wine and chicken.

In Boolean games the strategic actions are thus different from those of goal-based voting: in the former agents act by assigning a truth value only to the propositional variables they control, while in the latter agents directly submit their (possibly untruthful) goal. Boolean games have later been generalized to their iterated version (Gutierrez et al., 2013, 2015), in which agents play the game infinitely many times. Therefore, the goals of the agents are expressed in Linear Temporal Logic (LTL), that we introduce in Section 5.1, which extends propositional logic by making it possible to refer to future states of the world. In Chapter 7 we will see another example of iterated game where agents have goals expressed in LTL, i.e., we will introduce the class of *influence games*. Here, agents will be connected by a network of influence and they will try to change the opinions of the other agents to achieve their goals by using their influence power as an action.

Given a class of games, classical problems in game theory consist in studying whether the game admits certain types of solutions. Examples of solution concepts are, for instance, whether an agent has a strategy allowing her to obtain her goal no matter what the other agents do (*winning strategy*), whether an agent has a strategy which is better than any other strategy (*weakly dominant strategy*), and finally whether a strategy profile is such that no single agent would benefit from a unilateral deviation (*Nash equilibrium*). An important tool that can be used to check the existence of a winning strategy for an agent in an iterated game is Alternating-Time Temporal Logic (ATL), that we introduce in Section 5.1, which extends LTL by allowing to talk about the actions of a coalition of agents. As we will see, the semantics of ATL can be defined over so-called Concurrent Game Structures (CGS): in Chapter 8 we will introduce a special type of CGS where agents have shared control over propositional variables, which is useful for instance to capture the dynamics of influence games.

5.1 Logics for Time and Strategic Reasoning: ATL and LTL

A multitude of logics have been defined to reason about different scenarios in multi-agent systems, in particular game-theoretic ones (see, e.g., the book by Shoham and Leyton-Brown, 2008). We present here two such logics, suited to reason about agents having propositional goals in iterated games: the Linear-time Temporal Logic (LTL) by Pnueli (1977) and the Alternating-time Temporal Logic (ATL) by Alur et al. (2002). In particular, temporal logics with strategic operators allow us to express properties of games, such as the existence of a winning strategy for an agent.

We start by defining the syntax of ATL*, which is composed by *state* formulas (φ) and *path* formulas (ψ), for a given set of variables Φ and a set of agents N:

$$\begin{array}{rcl} \varphi & ::= & p \mid \neg\varphi \mid \varphi_1 \vee \varphi_2 \mid \langle\langle C \rangle\rangle \psi \\ \psi & ::= & \varphi \mid \neg\psi \mid \psi_1 \vee \psi_2 \mid \bigcirc\psi \mid \psi_1 \mathcal{U} \psi_2 \end{array}$$

where $p \in \Phi$ and $C \in 2^N$. The intuitive reading of $\langle\langle C \rangle\rangle \psi$ is "coalition C has a strategy to enforce ψ", that of $\bigcirc\psi$ is "ψ holds at the next state" and that of $\psi_1 \mathcal{U} \psi_2$ is "ψ_1 will hold until ψ_2 holds". Conjunction ($\wedge$) and implication ($\rightarrow$) are defined in the usual way. The language of ATL consists of all state formulas where ψ in $\langle\langle C \rangle\rangle \psi$ is either $\bigcirc\varphi$ or $\varphi \mathcal{U} \varphi$.

The language of LTL is defined by the following BNF and it consists of all path formulas in ATL* whose state formulas are propositional atoms only:

$$\psi \quad ::= \quad p \mid \neg\psi \mid \psi \vee \psi \mid \bigcirc\psi \mid \psi_1 \mathcal{U} \psi_2$$

In order to clarify the syntax of ATL, ATL* and LTL, we provide here a small example of formulas which we can or cannot express in the three logics:

Example 5.2. Let $\Phi = \{p, q\}$ be our propositional variables and $N = \{1, 2, 3\}$ be our set of agents. Consider the following four formulas:

$$\begin{array}{ll} \varphi_1 = p \vee q & \varphi_2 = \bigcirc p \vee q \\ \varphi_3 = \langle\langle \{1,2\} \rangle\rangle p \vee q & \varphi_4 = \langle\langle \{1,2\} \rangle\rangle \bigcirc p \vee q \end{array}$$

We have that φ_1 is a formula of ATL, ATL* and LTL; φ_2 is a formula of ATL* and LTL; φ_3 is a formula of ATL*; and φ_4 is a formula of ATL and ATL*.

We now introduce Concurrent Game Structures (CGS), which will give us a way to define truth conditions for LTL and ATL* formulas:

Definition 5.1. Given a set N of agents and a set Φ of propositional variables, a *concurrent game structure* (CGS) is a tuple $\mathcal{G} = \langle S, Act, d, \tau, \pi \rangle$ such that:

- S is a non-empty set of *states*;
- Act is a non-empty set of individual *actions*;
- $d : N \times S \to (2^{Act} \setminus \{\emptyset\})$ is the *protocol* function, returning the actions available to the agents at each state;
- $\tau : S \times Act^{|N|} \to S$ is the *transition* function, such that for every $s \in S$ and $\alpha \in Act^{|N|}$, $\tau(s, \alpha)$ is defined if and only if $\alpha_i \in d(i, s)$ for all $i \in N$;
- $\pi : S \to 2^{\Phi}$ is the *state labelling* function.

In order to define the semantics for LTL and ATL* we still need to provide some additional notation. The set of *enabled joint actions* at some state s is defined as $Act(s) = \{\alpha \in \mathcal{A}^n \mid \alpha_i \in d(i, s) \text{ for every } i \in N\}$. The set of *successors* of s is $Succ(s) = \{\tau(s, \alpha) \mid \alpha \in Act(s)\}$. An infinite sequence of states $\lambda = s_0 s_1 \dots$ is a *computation* or a *path* if $s_{k+1} \in Succ(s_k)$ for all $k \geq 0$. For every computation λ and $k \geq 0$, $\lambda[k, \infty] = s_k, s_{k+1}, \dots$ denotes the suffix of λ starting from s_k. Observe that $\lambda[k, \infty]$ is also a computation. When λ is clear from context, we denote with $\alpha[k]$ the action such that $\lambda[k+1] = \tau(\lambda[k], \alpha[k])$.

The following is a simple example to illustrate all the concepts introduced so far:

Example 5.3. Consider the sets of variables $\Phi = \{p, q\}$ and agents $N = \{1, 2, 3\}$. Define a CGS $\mathcal{G}$ with states $S = \{s_1, s_2, s_3, s_4\}$, actions $Act = \{a_1, a_2, a_3, \overline{a_1}, \overline{a_2}, \overline{a_3}\}$, protocol function d where for all $s \in S$ and $i \in N$ we have $d(i, s) = \{a_i, \overline{a_i}\}$, transition function τ where for all $s \in S$ if $\alpha = a_1 a_2 a_3$ then $\tau(s, \alpha) = s_1$, if $\alpha = \overline{a_1 a_2 a_3}$ then $\tau(s, \alpha) = s_4$, and for any other α we have $\tau(s, \alpha) = s_2$, and finally labelling function π such that $\pi(s_1) = \{p, q\}$, $\pi(s_2) = \{p\}$, $\pi(s_3) = \{q\}$ and $\pi(s_4) = \{\emptyset\}$.

We have that $Act(s) = \{a_1 a_2 a_3, a_1 a_2 \overline{a_3}, a_1 \overline{a_2} a_3, \overline{a_1} a_2 a_3, a_1 \overline{a_2 a_3}, \overline{a_1} a_2 \overline{a_3}, \overline{a_1 a_2} a_3, \overline{a_1 a_2 a_3}\}$ and $Succ(s) = \{s_1, s_2, s_4\}$ for all $s \in S$. An example of a computation is the path $\lambda = s_1 s_2 s_4 s_1 s_2 s_4 \dots$, and also its suffix $\lambda[2, \infty] = s_4 s_1 s_2 s_4 \dots$ is a computation as well. We have that $\alpha[2] = a_1 a_2 a_3$ is the action such that $\lambda[3] = \tau(\lambda[2], \alpha[2]) = s_1$.

A *(memoryless) strategy* for agent $i \in N$ is a function $\sigma_i : S \to Act$ such that $\sigma_i(s) \in d(i, s)$, returning an action for each state. We let $\boldsymbol{\sigma}_C$ be a *joint strategy* for coalition $C \subseteq N$, i.e., a function returning for each agent $i \in C$, the individual strategy σ_i. For simplicity we write $\boldsymbol{\sigma}$ for $\boldsymbol{\sigma}_N$. The set $out(s, \boldsymbol{\sigma}_C)$ includes all computations $\lambda = s_0 s_1 \dots$ such that (*a*) $s_0 = s$; and (*b*) for all $k \geq 0$, there is $\alpha \in Act(s)$ such that $\boldsymbol{\sigma}_C(i)(s_k) = \alpha_i$ for all $i \in C$, and $\tau(s_k, \alpha) = s_{k+1}$. Note that $out(s, \boldsymbol{\sigma})$ is a singleton.

We continue Example 5.3 to explain these new pieces of notation just defined:

Example 5.4. The strategy of agent 1 is defined as $\sigma_1(s_1) = \sigma_1(s_2) = a_1$ and $\sigma_1(s_3) = \sigma_1(s_4) = \overline{a_1}$. For agent 2, let $\sigma_2(s_1) = \sigma_2(s_3) = a_2$ and $\sigma_2(s_2) = \sigma_2(s_4) = \overline{a_2}$. Finally, for agent 3 let $\sigma_3(s_1) = \sigma_3(s_4) = a_3$ and $\sigma_3(s_2) = \sigma_3(s_3) = \overline{a_3}$. Given these strategies, we have for instance that $out(s_1, \boldsymbol{\sigma}) = \{s_1 s_1 s_1 s_1 \dots\}$ and $out(s_4, \boldsymbol{\sigma}) = \{s_4 s_2 s_2 s_2 \dots\}$

We are now ready to define truth conditions for LTL and ATL* formulas with respect to a CGS $\mathcal{G}$:

$$
\begin{array}{lll}
(\mathcal{G}, s) \models p & \text{iff} & s(p) = 1 \\
(\mathcal{G}, s) \models \neg\varphi & \text{iff} & (\mathcal{G}, s) \not\models \varphi \\
(\mathcal{G}, s) \models \varphi_1 \vee \varphi_2 & \text{iff} & (\mathcal{G}, s) \models \varphi_1 \text{ or } (\mathcal{G}, s) \models \varphi_2 \\
(\mathcal{G}, s) \models \langle\langle C \rangle\rangle \psi & \text{iff} & \text{for some } \boldsymbol{\sigma}_C, \text{ for all } \lambda \in out(s, \boldsymbol{\sigma}_C), (\mathcal{G}, \lambda) \models \psi \\
(\mathcal{G}, \lambda) \models \varphi & \text{iff} & (\mathcal{G}, \lambda[0]) \models \varphi \\
(\mathcal{G}, \lambda) \models \neg\psi & \text{iff} & (\mathcal{G}, \lambda) \not\models \psi \\
(\mathcal{G}, \lambda) \models \psi_1 \vee \psi_2 & \text{iff} & (\mathcal{G}, \lambda) \models \psi_1 \text{ or } \mathcal{G}, \lambda \models \psi_2 \\
(\mathcal{G}, \lambda) \models \bigcirc\psi & \text{iff} & (\mathcal{G}, \lambda[1, \infty]) \models \psi \\
(\mathcal{G}, \lambda) \models \psi_1 \mathcal{U} \psi_2 & \text{iff} & \text{for some } i \geq 0, (\mathcal{G}, \lambda[i, \infty]) \models \psi_2 \text{ and} \\
 & & (\mathcal{G}, \lambda[j, \infty]) \models \psi_1 \text{ for all } 0 \leq j < i
\end{array}
$$

In particular, observe that formulas in ATL* are interpreted on states, while formulas in LTL are interpreted on computations.

5.2 Related Work on Games, Logic, and Networks

A game-theoretic setting which is close to strategic goal-based voting is that of aggregation games (Grandi et al., 2019). In aggregation games, agents hold as propositional goals conjunctions over the variables at stake, and contrary to Boolean games agents have shared control over all the variables. The strategic actions consist in submitting a binary ballot over the issues. If transposed to the framework of goal-based voting, this would mean that agents would be allowed to submit only one of the models of their goal (as Camille in Example 3.1).

Influence games are part of the literature on formal models of opinion diffusion in multi-agent systems, combining techniques from social network analysis with those of belief merging and judgment aggregation. In particular, in Belief Revision Games (Schwind et al., 2015, 2016) agents connected via a network hold belief bases over some issues. At each step of the game, agents update their beliefs on the issues based on the opinions of the agents connected to them if the opinions do not conflict with their current beliefs; otherwise, agents update by means of belief merging operators to choose the most plausible opinion based on how "close" it is to the opinions of their connected agents (and themselves). Other examples include propositional opinion diffusion and pairwise preference diffusion on social networks (Grandi et al., 2015; Brill et al., 2016). In the propositional opinion diffusion model (later extended by Botan et al., 2019 with the inclusion of constrains on the issues), the focus is on studying how the opinions of the agents evolve over time due to the influence of other agents in the network. An agent's opinion at a given time results from aggregating the opinions at the previous time step of the agens from which she is influenced. Our contribution with influence games consists in adding a strategic component to these models, where agents can rationally choose whether or not to disclose their opinon to the people they influence, if by doing so they can work towards the satisfaction of their goal.

The CGS with shared control that we define extend CGS with exclusive control which have been introduced by Belardinelli and Herzig (2016). Another example of a logical framework for reasoning about capabilities of agents with exclusive propositional control is that of Coalition Logic of Propositional Control (CL-PC), introduced by van der Hoek and Wooldridge (2005). When ATL formulas are interpreted over the class of *strong* CGS with exclusive control defined by Belardinelli and Herzig

(2016), where control is exhaustive and all actions are available at all states, the expressive power is the same as that of CL-PC.[1] Gerbrandy (2006) studied variations of CL-PC without exclusive control, i.e., where a propositional variable p may be controlled by more than one agent. In particular, the logic of *positive control* requires that at least one agent sets p to true to assign the truth value true to p, while in the logic of *consensus games* all agents have to set p to true to change its value to true.[2]

Finally, we also mention *simple games*, a class of cooperative games where each coalition of agents is labeled as either winning or losing, and *weighted majority games*, a class of simple games where we can associate to each agent some weight such that for every winning coalition the sum of agents' weights is higher than the sum of weights in every losing coalition (Isbell, 1959). Observe however that in these cooperative games agents do not hold conflicting logical goals.

[1] The respective model-checking problems however differ in complexity: Δ_3^P-complete for CGS with exclusive control and PSPACE-complete for CL-PC (Belardinelli and Herzig, 2016).

[2] Observe the similarity of these notions with, respectively, the *TrSh* rules with quota 1 (introduced in Section 3.1.2) and the unanimity axiom (introduced in Section 3.2.3).

Chapter 6

Strategic Goal-Based Voting

Manipulation of voting rules has been amply studied in voting theory, starting from the seminal result of Gibbard (1973) and Satterthwaite (1975) to more recent studies aimed at finding barriers to manipulation (see, e.g., the recent survey by Conitzer and Walsh, 2016). When introducing the framework of goal-based voting in Chapter 3 we assumed that agents always provide their truthful goal. However, there may be situations where agents would benefit (i.e., getting a better outcome) by lying about their goals. Consider the following example:

Example 6.1. Three automated personal assistants need to arrange a business meal for their owners. They have to decide whether the restaurant should be fancy (F), if it should be in the center (C), and if they should meet for lunch (L) instead of dinner. Each owner gives to their assistant a propositional goal with respect to these issues. The goal of the first agent is that if they go to a restaurant in the suburbs, then they should have a casual lunch: $\gamma_1 = \neg C \rightarrow (\neg F \wedge L)$. The second agent wants that the meeting is either in the suburbs or casual, but not both: $\gamma_2 = \neg C \oplus \neg F$. The third agent wants a fancy lunch in the center: $\gamma_3 = F \wedge L \wedge C$. If the agents use the *TrueMaj* rule (defined in Section 3.1.2), the result would be $\{(111)\}$. However, the second agent notices that by submitting the goal $\gamma_2 = C \wedge \neg F \wedge \neg L$ the outcome of *TrueMaj* would be $\{(101)\}$ which is one of the models of the goal γ_2 unlike the truthful outcome $\{(111)\}$.

In this chapter we will thus study the strategic behavior of goal-oriented agents when there is no assumption on their truthfulness. In particular, we will focus on the three majoritarian rules that we have introduced in Section 3.1.2: i.e., *EMaj*, *TrueMaj* and *2sMaj*. The appeal of majority lies not only in its intuitive definition and extensive application in real-world scenarios, but also on having been widely studied in the related fields of voting theory and judgment aggregation (May, 1952; Dietrich and List, 2007a), including from a strategic point of view (Dietrich and List, 2007c).

Each of our majoritarian goal-based voting rules will be analyzed with respect to its resistance to several manipulation strategies, inspired by similar work on strategy-proofness in belief merging (Everaere et al., 2007). Negative results, i.e., finding that a rule *can* be manipulated in the general case, lead us to study the computational complexity of manipulation, as well as restricting the language of individual goals in the hope of discovering niches of strategy-proofness.[1]

6.1 Framework

The basic definitions of the framework of goal-based voting have been introduced in Section 3.1.1. As we have mentioned, we assume that agents holding propositional

[1] Conitzer and Walsh (2016) discuss the limitations of the worst-case complexity approach when dealing with manipulation, as the instances making the problem hard may be uncommon.

goals leads to a dicothomous preference relation on outcomes: an agent equally prefers any model of her goal to any counter-model. For resolute rules, this implies that the *unique* outcome satisfies an agent if and only if it is a model of her goal. If a rule is not resolute (in the stronger sense), however, different notions of satisfaction with respect to the outcome can be defined depending on how an agent compares two *sets* of interpretations.

Let $sat : \mathcal{L}_\mathcal{I} \times (\mathcal{P}(\{0,1\}^m) \setminus \emptyset) \to [0,1]$ be a function expressing the *satisfaction* of agent i towards the outcome of a rule F on profile $\mathbf{\Gamma}$, with the intuitive reading that 1 indicates maximal satisfaction of the agent while 0 indicates maximal unsatisfaction. We simply write $sat(i, F(\mathbf{\Gamma}))$ instead of $sat(\gamma_i, F(\mathbf{\Gamma}))$. The *optimistic*, *pessimistic* and *expected utility maximizer* are three notions of satisfaction an agent may hold:

$$opt(i, F(\mathbf{\Gamma})) = \begin{cases} 1 & \text{if } F(\mathbf{\Gamma}) \cap \text{Mod}(\gamma_i) \neq \emptyset \\ 0 & \text{otherwise} \end{cases}$$

$$pess(i, F(\mathbf{\Gamma})) = \begin{cases} 1 & \text{if } F(\mathbf{\Gamma}) \subseteq \text{Mod}(\gamma_i) \\ 0 & \text{otherwise} \end{cases}$$

$$eum(i, F(\mathbf{\Gamma})) = \frac{|\text{Mod}(\gamma_i) \cap F(\mathbf{\Gamma})|}{|F(\mathbf{\Gamma})|}$$

An optimistic agent is satisfied if in the outcome of F there is at least one model of her goal. A pessimistic agent wants all the interpretations in the outcome of F to be models of her goal. The notions of optimistic and pessimistic agents also appear in the work by Jimeno et al. (2009) for orderings of candidates. More precisely, when comparing two sets of candidates A and B, an optimist ordering prefers A to B if the best candidate in A is better than the best candidate in B. Conversely, a pessimistic ordering prefers A to B if the worst candidate in A is better than the worst candidate in B. These notions correspond to ours when considering sets A and B containing interpretations over variables. Expected utility maximizers assume that a unique interpretation will be chosen at random among those tied in the outcome of F, and thus the higher the proportion of models of the agent's goal in $F(\mathbf{\Gamma})$ over the total number of interpretations in $F(\mathbf{\Gamma})$, the better.[2]

Another approach to model the satisfaction of an agent with repect to an outcome $F(\mathbf{\Gamma})$ would be by using preferences based on the Hamming distance, as done for instance in judgment aggregation (Dietrich and List, 2007c; Endriss et al., 2012). The Hamming distance between two interpretations corresponds to the number of issues on which they have different values: for instance, interpretations (011) and (101) have an Hamming distance of 2 as they only differ on the values assigned to the first two issues. When comparing a set of interpretations (e.g., the models of an agent's goal) with a single interpretation (e.g., the outcome of a resolute rule) or another set of interpretations (e.g., the outcome of a non-resolute rule) we would need to define again different types of agents, depending on whether the Hamming distance is computed between the interpretations that are closer, those that are further away, or an average of them. We leave the study of manipulation of goal-based voting with Hamming preferences for future work.

[2]Expected utility maximizers, optimistic, and pessimistic agents correspond to the *probabilistic*, *weak drastic* and *strong drastic* satisfaction indexes in the work of Everaere et al. (2007), whose work however did not study the issue-wise majority rules defined here.

The preference of agent i over outcomes is a complete and transitive relation $\succcurlyeq_i$, whose strict part is $\succ_i$, which is defined based on the agent's satisfaction:

$$F(\mathbf{\Gamma}) \succcurlyeq_i F(\mathbf{\Gamma}') \text{ \textit{iff} } sat(i, F(\mathbf{\Gamma})) \geq sat(i, F(\mathbf{\Gamma}')).$$

Namely, agent i prefers the outcome of F on profile $\mathbf{\Gamma}$ to the outcome of F on profile $\mathbf{\Gamma}'$ if and only if the satisfaction of agent i for $F(\mathbf{\Gamma})$ is higher than her satisfaction for $F(\mathbf{\Gamma}')$. Generalizing standard definitions of manipulation in judgment aggregation, we can now define what does it mean for a goal-based voting rule to be immune to manipulation.

Given $\mathbf{\Gamma} = (\gamma_i)_{i \in N}$, let $(\mathbf{\Gamma}_{-i}, \gamma'_i) = (\gamma_1, \ldots, \gamma'_i, \ldots, \gamma_n)$ be the profile where only agent i changed her goal from γ_i to γ'_i. Agent i has an *incentive to manipulate* by submitting goal γ'_i instead of γ_i if and only if $F(\mathbf{\Gamma}_{-i}, \gamma'_i) \succ_i F(\mathbf{\Gamma})$. A rule F is *strategy-proof* if and only if for all profiles $\mathbf{\Gamma}$ there is no agent i who has an incentive to manipulate.

Even though in some scenarios an agent may have an incentive to manipulate, there could be some constraints on the type of manipulation she can actually perform. For instance, consider the example below:

Example 6.2. Ann, Barbara and Camille are deciding the details of their next meeting and, among other things, whether they will meet in the morning or in the afternoon. Barbara and Camille know Ann very well, and they are aware that she is definitely a morning person. Therefore, even if there may be cases in which Ann would benefit by submitting a goal favouring a meeting in the afternoon, Barbara and Camille would immediately suspect that Ann is up to something.

For this reason, we will focus on three manipulation strategies previously introduced by Everaere et al. (2007):

- *Unrestricted*: i can send any goal γ'_i instead of her truthful goal γ_i
- *Erosion*: i can only send a goal γ'_i such that $\text{Mod}(\gamma'_i) \subseteq \text{Mod}(\gamma_i)$
- *Dilatation*: i can only send a goal γ'_i such that $\text{Mod}(\gamma_i) \subseteq \text{Mod}(\gamma'_i)$

Observe that a rule F that is manipulable by erosion and dilatation is therefore manipulable unrestrictedly. Conversely, a rule that is strategy-proof for unrestricted manipulation is also strategy-proof for erosion and dilatation strategies.

Another type of restriction that we will study for manipulation is on the individual goals themselves. Goals have been defined as arbitrary consistent formulas of propositional logic in Section 3.1.1, but the fact that a goal-based voting rule F is manipulable may depend on which fragment of propositional logic the agent is using to express her goal. Hence, we will investigate what happens when the goals are restricted to the languages $\mathcal{L}^\star$ defined at the end of Section 2.1.

In the rest of the chapter we will focus on the three issue-wise majoritarian rules introduced in Section 3.1.2, i.e., *EMaj*, *TrueMaj* and *2sMaj*.

6.2 Manipulation of Majoritarian Rules

A well-known result in judgment aggregation proven by Dietrich and List (2007c) is that the issue-wise majority rule is single-agent strategy-proof.[3] Our first result

[3] More precisely, their result in formula-based judgment aggregation holds for *closeness-respecting* preferences: i.e., preferences where $J \succcurlyeq_i J'$ if $(J_i \cap J') \subset (J_i \cap J)$ for judgment sets J_i, J, J'. For goal-based voting, we could rewrite it as $F(\mathbf{\Gamma}) \succcurlyeq_i F(\mathbf{\Gamma}')$ if $(\text{Mod}(\gamma_i) \cap F(\mathbf{\Gamma}')) \subset (\text{Mod}(\gamma_i)) \cap F(\mathbf{\Gamma}))$, which is implied by our definition of preference for resolute rules.

shows that in the context of goal-based voting strategy-proofness is not guaranteed for the three adaptations of the majority rule:

Theorem 6.1. *EMaj*, *TrueMaj* and *2sMaj* are manipulable by erosion and dilatation.

Proof. We provide goal-profiles where an agent can get a better result by submitting an untruthful goal. For ease of presentation we display the models of the agents' goals, but recall that the input of a rule F consists of propositional formulas. Consider the profiles $\mathbf{\Gamma}$, $\mathbf{\Gamma}'$ and $\mathbf{\Gamma}''$ for three agents and three issues, together with the results of *EMaj*, *TrueMaj* and *2sMaj*:

	$\mathbf{\Gamma}$	$\mathbf{\Gamma}'$	$\mathbf{\Gamma}''$
Mod(γ_1)	(111)	(111)	(111)
Mod(γ_2)	(001)	(001)	(001)
Mod(γ_3)	(101) (010) (000)	(101)	(101) (010) (000) (100) (110)
{*E*/*True*/*2s*}*Maj*	(001)	(101)	(101)

Let $\mathbf{\Gamma}$ be the profile where each agent submits their truthful goal: $\gamma_1 = 1 \wedge 2 \wedge 3$, $\gamma_2 = \neg 1 \wedge \neg 2 \wedge 3$, $\gamma_3 = (\neg 1 \wedge \neg 3) \vee (1 \wedge \neg 2 \wedge 3)$. For erosion manipulation, agent 3 prefers the result of *EMaj*, *2sMaj* and *TrueMaj* (which happen to coincide) when applied to $\mathbf{\Gamma}'$ rather than when applied to $\mathbf{\Gamma}$. For dilatation, agent 3 prefers the result of *EMaj*, *TrueMaj* and *2sMaj* when applied to $\mathbf{\Gamma}''$ rather than to $\mathbf{\Gamma}$. □

Observe that the three majoritarian rules on the profiles used in the proof of Theorem 6.1 return singleton outcomes: hence, the result holds for optimists, pessimists and expected utility maximizers (since the three notions coincide on such profiles).

Theorem 6.1 is thus in contrast with the result of judgment aggregation, painting a more negative picture for manipulation in goal-based voting. In the next sections we thus study three restrictions on the goal-language in a quest to find islands of strategy-proofness. Results are summarized in Table 6.1.

6.2.1 Conjunctions

The language of conjunctions $\mathcal{L}^\wedge$ captures the framework of judgment aggregation with abstentions (Gärdenfors, 2006; Dietrich and List, 2008; Dokow and Holzman, 2010b; Terzopoulou et al., 2018), as explained in Section 3.4.2. This means, in particular that agents having a goal in $\mathcal{L}^\wedge$ either support, reject or abstain on each issue. For this restriction on the language we find positive results of strategy-proofness:

Theorem 6.2. For any profile $\mathbf{\Gamma}$ where $\gamma_i \in \mathcal{L}^\wedge$ for some $i \in \mathcal{N}$, agent i has no incentive to manipulate unrestrictedly the rules *2sMaj* and *EMaj*.

Proof. Consider a profile $\mathbf{\Gamma}$ with $\gamma_i \in \mathcal{L}^\wedge$ for some agent $i \in \mathcal{N}$. Since *2sMaj* and *EMaj* are resolute, we have a unique outcome $\{w\}$ on $\mathbf{\Gamma}$ (which may be different for the two rules). If $w \in \text{Mod}(\gamma_i)$ agent i has no incentive to manipulate as her goal is already satisfied. Hence, suppose that $w \notin \text{Mod}(\gamma_i)$. As $\gamma_i = L_{j_1} \wedge \cdots \wedge L_{j_k}$ for $j_1, \ldots, j_k \in \mathcal{I}$, we have for all $j_\ell \in \mathcal{I}$:

- If L_{j_ℓ} is not in γ_i, then $m_i(j_\ell) = (\frac{|\text{Mod}(\gamma_i)|}{2}, \frac{|\text{Mod}(\gamma_i)|}{2})$;
- If $L_{j_\ell} = \ell$ is in γ_i, then $m_i(\ell) = (0, |\text{Mod}(\gamma_i)|)$;
- If $L_{j_\ell} = \neg\ell$ is in γ_i, then $m_i(\ell) = (|\text{Mod}(\gamma_i)|, 0)$.

Therefore, if $w \notin \text{Mod}(\gamma_i)$ there must be x literals in γ_i such that w satisfies their negation. Consider an arbitrary such literal L_x.

For *2sMaj*, let $\mathit{EMaj}(\gamma_i) = \{w_i\}$ be the result of the first step of majority applied to γ_i. We have that $w_i(x) = 1 - w(x)$, and therefore agent i cannot influence the outcome of *2sMaj* towards $w_i(x)$, and thus towards her goal.

For *EMaj*, if $w(x) = 1$ (analogously for 0), $L_x = \neg x$. Since $m_i(x) = (|\text{Mod}(\gamma_i)|, 0)$, we have $\sum_{v \in \text{Mod}(\gamma_i)} \frac{v(x)}{|\text{Mod}(\gamma_i)|} = 0$ and thus $\sum_{k \in N \setminus \{i\}} \sum_{v \in \text{Mod}(\gamma_k)} \frac{v(x)}{|\text{Mod}(\gamma_k)|} \geq \frac{n}{2}$. Agent i is already giving no support to x and yet x is accepted in the outcome. Therefore, *EMaj* cannot be manipulated. $\square$

For *TrueMaj* we can provide a similar proof, but we also need to consider the type of satisfaction of the agents, i.e., optimist, pessimist or expected utility maximizer:

Theorem 6.3. For any profile $\mathbf{\Gamma}$ where $\gamma_i \in \mathcal{L}^\wedge$ for some $i \in \mathcal{N}$, agent i has no incentive to manipulate unrestrictedly the rule *TrueMaj*.

Proof. First, observe that on the resolute profiles on which *TrueMaj* and *EMaj* coincide, we can apply the same reasoning as for the proof of Theorem 6.2. Consider now a profile $\mathbf{\Gamma}$ such that $\gamma_i \in \mathcal{L}^\wedge$, and let $\mathit{TrueMaj}(\mathbf{\Gamma}) = \{w_1, \dots, w_\ell\}$.

Let us start by considering an optimist agent i. By definition, agent i manipulates if and only if $\{w_1, \dots, w_\ell\} \cap \text{Mod}(\gamma_i) = \emptyset$, and she seeks to include at least one $v \in \text{Mod}(\gamma_i)$ into $\mathit{TrueMaj}(\mathbf{\Gamma})$. As $v \notin \mathit{TrueMaj}(\mathbf{\Gamma})$, there exists $\ell \in \mathcal{I}$ with literal L_ℓ appearing in γ_i such that $M(\mathbf{\Gamma}) = \{1 - x\}$ for $x = v(\ell)$. Thus $\sum_{i \in N} \frac{m_{i\ell}^{1-x}}{|\text{Mod}(\gamma_i)|} > \sum_{i \in N} \frac{m_{i\ell}^{x}}{|\text{Mod}(\gamma_i)|}$. By being truthful, however, agent i is already giving her full support to L_ℓ and hence cannot change the outcome in her favor.

An expected utility maximizer agent i manipulates when she can increase the value of $\frac{|\text{Mod}(\gamma_i) \cap \mathit{TrueMaj}(\mathbf{\Gamma})|}{|\mathit{TrueMaj}(\mathbf{\Gamma})|}$. As shown above, it is not possible for agent i to include an arbitrary $v \in \text{Mod}(\gamma_i)$ that is not in $\mathit{TrueMaj}(\mathbf{\Gamma}) = \{w_1, \dots, w_\ell\}$ in the outcome $\mathit{TrueMaj}(\mathbf{\Gamma})$. However, agent i may want to remove some counter-models of γ_i from $\mathit{TrueMaj}(\mathbf{\Gamma})$. All counter-models differ from the models of γ_i on some of the issues appearing as literals in γ_i. Hence, for any of these issues j the agent has to change $M(\mathbf{\Gamma})_j = \{0, 1\}$ into $\{0\}$ if $L_j = \neg j$, or $\{1\}$ if $L_j = j$. However, at $\mathbf{\Gamma}$ we have $\sum_{i \in N} \frac{m_{ij}^1}{|\text{Mod}(\gamma_i)|} = \sum_{i \in N} \frac{m_{ij}^0}{|\text{Mod}(\gamma_i)|}$, even though agent i is giving the full support of 1 to L_j. Thus, she cannot break the tie as any γ_i' would have models lowering her support.

Finally, a pessimist agent i cannot manipulate since she wants at least one $v \in \text{Mod}(\gamma_i)$ in the outcome, which is impossible as shown for the optimist case, and to remove all counter-models, which is also impossible as shown for the expected utility maximizers. Hence, the rule *TrueMaj* is strategy-proof. $\square$

A consequence of Theorems 6.2 and 6.3 is that goals in $\mathcal{L}^\wedge$ make the three majorities for goal-based voting strategy-proof:

Corollary 6.1. For any $\mathcal{I}$ and N, if $\gamma_i \in \mathcal{L}^\wedge$ for all $i \in N$ then *EMaj*, *TrueMaj* and *2sMaj* are strategy-proof for unrestricted manipulation.

Hence, Corollary 6.1 gives us a strategy-proofness result for majority rules in judgment aggregation with abstentions.

6.2.2 Disjunctions

Agents with goals in the language $\mathcal{L}^\vee$ want at least one of the literals in their goal to be accepted. The language of disjunctions is not enough, however, to guarantee strategy-proofness for *EMaj* and *TrueMaj*, as the following result shows for the erosion manipulation strategy:

Theorem 6.4. There exists a profile $\mathbf{\Gamma}$ and an agent $i \in N$ with $\gamma_i \in \mathcal{L}^\vee$ such that agent i has an incentive to manipulate *EMaj* and *TrueMaj* by erosion.

Proof. Consider a profile $\mathbf{\Gamma}$ for three agents and two issues such that $\gamma_1 = \neg 1 \wedge \neg 2$ and $\gamma_2 = \gamma_3 = 1 \vee 2$. The outcome on $\mathbf{\Gamma}$ for both *EMaj* and *TrueMaj* is $\{(00)\}$. However, if agent 3 now submits goal $\gamma'_3 = 1 \wedge 2$ we obtain $EMaj(\mathbf{\Gamma}') = TrueMaj(\mathbf{\Gamma}') = \{(11)\}$. Therefore, agent 3 has an incentive to manipulate $\mathbf{\Gamma}$ and the rules *EMaj* and *TrueMaj* are not strategy-proof. □

We can however obtain a positive result for *EMaj* and *TrueMaj* by restricting the set of available manipulation strategies to dilatation:

Theorem 6.5. For any profile $\mathbf{\Gamma}$ with $\gamma_i \in \mathcal{L}^\vee$ for $i \in N$, agent i has no incentive to manipulate *EMaj* and *TrueMaj* by dilatation.

Proof. Consider a profile $\mathbf{\Gamma}$ with $\gamma_i \in \mathcal{L}^\vee$ for some $i \in N$. First of all, observe that an agent having $p \vee \neg p$ for some $p \in \mathcal{I}$ in her goal already supports all interpretations and thus has no incentive to manipulate. For *EMaj*, let $EMaj(\mathbf{\Gamma}) = \{w\}$ such that $w \notin \mathrm{Mod}(\gamma_i)$. Since agent i can only use a dilatation strategy, the goals γ'_i such that $\mathrm{Mod}(\gamma_i) \subseteq \mathrm{Mod}(\gamma'_i)$ available to agent i are those whose models would lower i's support for each literal L_ℓ in γ_i. Thus she cannot manipulate.

Consider now $TrueMaj(\mathbf{\Gamma}) = \{w_1, \dots, w_k\}$. Similarly to the reasoning above, an optimist agent i does not have any dilatation strategy which can allow her to include one of the models of γ_i in the outcome of *TrueMaj*. An expected utility maximizer agent i may want to remove some $w_k \in TrueMaj(\mathbf{\Gamma})$ such that $w_k \models \neg L_j$ for all L_j in γ_i. However, this is only possible when $\{w_1, \dots, w_\ell\} \cap \mathrm{Mod}(\gamma_i) = \emptyset$, since the agent is restricted to dilatation strategies, and any other γ'_i would have more models increasing the votes against the literals in her sincere goal γ_i. By combining the reasoning for optimists and expected utility maximizers we can obtain an analogous result for pessimist agents as well, thus concluding the proof. □

For *2sMaj* we have a positive result for any type of manipulation strategy:

Theorem 6.6. For any profile $\mathbf{\Gamma}$ where $\gamma_i \in \mathcal{L}^\vee$ for some $i \in N$, agent i has no incentive to manipulate unrestrictedly the rule *2sMaj*.

Proof. Consider an arbitrary $\mathbf{\Gamma}$ where for some $i \in \mathcal{N}$ we have $\gamma_i \in \mathcal{L}^\vee$. First, an agent having $p \vee \neg p$ for some $p \in \mathcal{I}$ in her goal already supports all interpretations and thus has no incentive to manipulate. The result w_i of $EMaj(\mathrm{Mod}(\gamma_i))$ is such that $w_i(x) = 1$ if $L_x = x$ appears in γ_i, and $w_i(x) = 0$ if $L_x = \neg x$ appears in γ_i. Hence, it coincides with the result of $EMaj(\mathrm{Mod}(\gamma'_i))$ for $\gamma'_i \in \mathcal{L}^\wedge$ where every occurrence of $\vee$ in γ_i has been replaced by $\wedge$ in γ'_i. The proof of Theorem 6.2 can thus be applied, obtaining that agent i is already maximizing her chances of getting γ_i satisfied by submitting γ_i. □

By combining the results of Theorems 6.5 and 6.6 we get the following corollary:

Corollary 6.2. If $\gamma_i \in \mathcal{L}^\vee$ for all $i \in N$ then *EMaj* and *TrueMaj* are strategy-proof for dilatation manipulation, and *2sMaj* is strategy-proof for unrestricted manipulation.

By combining Corollaries 6.1 and 6.2 we get the following positive result:

Theorem 6.7. For any profile $\mathbf{\Gamma}$ such that $\gamma_1, \ldots, \gamma_n \in \mathcal{L}^\wedge \cup \mathcal{L}^\vee$, *EMaj* and *TrueMaj* are strategy-proof for dilatation manipulation and *2sMaj* is strategy-proof for unrestricted manipulation.

Therefore, we have found a restriction on the language of goals that gives us a strategy-proofness result for the three issue-wise majority rules *EMaj*, *TrueMaj* and *2sMaj*. If agents have as goals conjunctions or disjunctions they will not have an incentive to manipulate.

6.2.3 Exclusive Disjunctions

The final restriction that we study is that on the language of exclusive disjunctions $\mathcal{L}^\oplus$, where each formula is an exclusive disjunction of literals. The models of a formula $\varphi = ((p \oplus q) \oplus r)$ over three variables are $\text{Mod}(\varphi) = \{(111), (010), (100), (001)\}$. While our intuitive understanding of exclusive disjunction (i.e., "exactly one variable is true at the same time") may be in contrast with the presence of model (111), the results in this section are all for profiles on only two variables that thus do not pose any problem of interpretation.

Unfortunately, such a restriction does not give strategy-proofness for majority rules, as the following result shows:

Theorem 6.8. There exists profiles $\mathbf{\Gamma}^0, \mathbf{\Gamma}^1$ and $\mathbf{\Gamma}^2$, and agent $i \in N$ with $\gamma_i \in \mathcal{L}^\oplus$ such that agent i has an incentive to manipulate rules *2sMaj*, *EMaj* and *TrueMaj* by erosion and dilatation.

Proof. All profiles are for three agents and two issues. For *2sMaj* and *EMaj* and erosion manipulation, consider profile $\mathbf{\Gamma}^0$ where $\gamma_1 = 1 \wedge 2$, $\gamma_2 = \neg 1 \wedge \neg 2$ and $\gamma_3 = 1 \oplus 2$. We have that $2sMaj(\mathbf{\Gamma}^0) = EMaj(\mathbf{\Gamma}^0) = \{(00)\}$. Consider now $\gamma_3' = \neg 1 \wedge 2$. The result for both rules is $\{(01)\}$, and thus agent 3 has an incentive to manipulate.

For dilatation, consider the profile $\mathbf{\Gamma}^1$ where $\gamma_1 = \neg 1 \wedge 2$, $\gamma_2 = \neg 1 \wedge \neg 2$ and $\gamma_3 = 1 \oplus 2$. We have that $2sMaj(\mathbf{\Gamma}^1) = EMaj(\mathbf{\Gamma}^1) = \{(00)\}$. If we consider $\gamma_3^\star = 1 \vee 2$, the result is $2sMaj(\mathbf{\Gamma}^{1\star}) = EMaj(\mathbf{\Gamma}^{1\star}) = \{(01)\}$, and thus agent 3 has again an incentive to manipulate.

For *TrueMaj*, consider $\mathbf{\Gamma}^2$ with $\gamma_1 = 1 \wedge 2$, $\gamma_2 = 1 \wedge \neg 2$ and $\gamma_3 = 1 \oplus 2$. We have $TrueMaj(\mathbf{\Gamma}^2) = \{(10), (11)\}$. Agent 3 can manipulate by erosion with $\gamma_3^\star = 1 \wedge \neg 2$, and by dilatation with $\gamma_3^{\star\star} = \neg 1 \vee \neg 2$. In both cases the result is $\{(10)\}$. □

Therefore, the language of exclusive disjunctions does not guarantee strategy-proofness for the rules *2sMaj*, *EMaj* and *TrueMaj* for any of the manipulation strategies we introduced. From the proofs of Theorems 6.6 and 6.8 we can observe that the rule *2sMaj* behaves in counter-intuitive ways for the goals in $\mathcal{L}^\vee$ and $\mathcal{L}^\oplus$, and hence its use should not be prioritized in such cases.

6.3 Computational Complexity

We have seen in Section 6.2 that majoritarian goal-based voting rules are in general subject to the threat of manipulation, as shown by Theorem 6.1. However we may be interested in checking how hard would it be for an agent to actually *know* that by submitting another goal she would get a better outcome.

The problem of manipulability of resolute rules is defined as follows, akin to its definition for judgment aggregation (Endriss et al., 2012):

	$\mathcal{L}^\wedge$		$\mathcal{L}^\vee$		$\mathcal{L}^\oplus$	
	E	D	E	D	E	D
EMaj	SP	SP	M	SP	M	M
TrueMaj	SP	SP	M	SP	M	M
2sMaj	SP	SP	SP	SP	M	M

TABLE 6.1: Manipulation and strategy-proofness results for the majoritatian rules *EMaj*, *TrueMaj* and *2sMaj* and the language restrictions $\mathcal{L}^\wedge$, $\mathcal{L}^\vee$ and $\mathcal{L}^\oplus$. In particular, 'E' stands for erosion, 'D' for dilatation, 'SP' for strategy-proof and 'M' for manipulable.

MANIP(F)
Input: Profile $\mathbf{\Gamma} = (\gamma_1, \dots, \gamma_n)$, agent i
Question: If $\text{Mod}(\gamma_i) \cap F(\mathbf{\Gamma}) = \emptyset$, is there γ_i' so that $\text{Mod}(\gamma_i) \cap F(\mathbf{\Gamma}_{-i}, \gamma_i') \neq \emptyset$?

As we have seen in Section 3.3.4, the winner determination problem WINDET of majority rules was hard for the class PP (Probabilistic Polynomial Time). Our results will make use of Lemma 3.1 where we proved that the problem MAJSAT-p is PP-complete.

Theorem 6.9. MANIP(*2sMaj*) is PP-hard.

Proof. We reduce from the PP-complete problem MAJSAT-p. Consider an instance of MAJSAT-p with a formula $\varphi[p_1, \dots, p_k]$ and p_1 one of its variables. Construct an instance of MANIP(*2sMaj*) where $\mathcal{I} = \{p_1, \dots, p_k, q, r\}$, and a profile $\mathbf{\Gamma} = (\gamma_1, \gamma_2, \gamma_3)$ with $\gamma_1 = (p_2 \wedge \dots \wedge p_k) \wedge p_1 \wedge q \wedge r$, and $\gamma_2 = \varphi \wedge \neg q \wedge \neg r$, and $\gamma_3 = (p_2 \wedge \dots \wedge p_k) \wedge (p_1 \oplus q) \wedge (r \rightarrow q)$.

We show that $|\text{Mod}(\varphi \wedge p_1)| > |\text{Mod}(\varphi \wedge \neg p_1)|$ if and only if agent 3 can manipulate *2sMaj* on $\mathbf{\Gamma}$. The following table represents some features of $\mathbf{\Gamma}$, where question marks indicate the (possibly many) models of φ over $p_1, \dots, p_k$:

	$p_2 \dots p_k$	p_1	q	r
$\text{Mod}(\gamma_1)$	$1 \dots 1$	1	1	1
	?	?	0	0
$\text{Mod}(\gamma_2)$	$\vdots$	$\vdots$	0	0
	?	?	0	0
	$1 \dots 1$	0	1	1
$\text{Mod}(\gamma_3)$	$1 \dots 1$	0	1	0
	$1 \dots 1$	1	0	0

The result on $p_2, \dots, p_k$ is decided by agents 1 and 3: all issues will be accepted regardless of agent 2's vote. Let us now focus on p_1, q and r. Applying strict majority to the models of γ_3 leads to the first-step result (010). Agent 3 is pivotal on issues q and r (since agents 1 and 2 give one vote for and one vote against them after the first step). There are now two cases to consider:

a) If $|\text{Mod}(\varphi \wedge p_1)| > |\text{Mod}(\varphi \wedge \neg p_1)|$, the result of *2sMaj*$(\mathbf{\Gamma})$ is $(1 \dots 1110)$, that is not a model of γ_3. However, by submitting $\gamma_3' = (p_2 \wedge p_k) \wedge p_1 \wedge \neg q \wedge \neg r$, we have *2sMaj*$(\mathbf{\Gamma}') = (1 \dots 1100)$ which is a model of γ_3. Hence, agent 3 has an incentive to manipulate.

b) If $|\text{Mod}(\varphi \wedge p_1)| \leq |\text{Mod}(\varphi \wedge \neg p_1)|$, we have that $2sMaj(\mathbf{\Gamma}) = (1\dots1010)$, which is a model of γ_3. Agent 3 has thus no incentive to manipulate.

This completes the reduction, showing that MANIP(*2sMaj*) is PP-hard. □

For the rule *EMaj* we can provide the following similar result:

Theorem 6.10. MANIP(*EMaj*) is PP-hard.

Proof. We reduce from MAJSAT-p by constructing an instance of MANIP(*EMaj*) from a given instance (φ, p_1) of MAJSAT-p. Let $\mathcal{I} = \{p_1, \dots, p_k, q, r\}$, and profile $\mathbf{\Gamma} = (\gamma_1, \gamma_2, \gamma_3)$ with $\gamma_1 = (p_2 \wedge \dots \wedge p_k) \wedge p_1 \wedge q \wedge r$, $\gamma_2 = \varphi \wedge \neg q \wedge r$ and $\gamma_3 = (p_2 \wedge \dots \wedge p_k) \wedge ((p_1 \wedge q \wedge r) \vee (p_1 \wedge \neg q \wedge \neg r) \vee (\neg p_1 \wedge \neg q \wedge r) \vee (\neg p_1 \wedge \neg q \wedge \neg r))$.

Analogously to the proof of Theorem 6.9, we show that on profile $\mathbf{\Gamma}$ agent 3 can manipulate *EMaj* if and only if $|\text{Mod}(\varphi \wedge p_1)| > |\text{Mod}(\varphi \wedge \neg p_1)|$:

	$p_2 \dots p_k$	p_1	q	r
$\text{Mod}(\gamma_1)$	$1\dots1$	1	1	1
	?	?	0	1
$\text{Mod}(\gamma_2)$	$\vdots$	$\vdots$	0	1
	?	?	0	1
	$1\dots1$	1	1	1
$\text{Mod}(\gamma_3)$	$1\dots1$	1	0	0
	$1\dots1$	0	0	1
	$1\dots1$	0	0	0

Regardless of what agent 2 is voting for $p_2, \dots, p_k$, the result is decided by agents 1 and 3: all issues will be accepted in the outcome. For variables p_1, q and r we do a case study to show that $|\text{Mod}(\varphi \wedge p_1)| > |\text{Mod}(\varphi \wedge \neg p_1)|$ if and only if agent 3 can manipulate *EMaj* on $\mathbf{\Gamma}$:

1. If $|\text{Mod}(\varphi \wedge p_1)| > |\text{Mod}(\varphi \wedge \neg p_1)|$, then we have $EMaj(\mathbf{\Gamma}) = (1\dots1101)$, which is not a model of γ_3. By submitting $\gamma_3' = (p_2 \wedge \dots \wedge p_k) \wedge p_1 \wedge q \wedge r$, we have $EMaj(\mathbf{\Gamma}') = (1\dots1111)$ which is a model of γ_3. Hence, agent 3 has an incentive to manipulate.

2. If $|\text{Mod}(\varphi \wedge p_1)| \leq |\text{Mod}(\varphi \wedge \neg p_1)|$, then $EMaj(\mathbf{\Gamma}) = (1\dots1001)$, which is a model of γ_3. Agent 3 has thus no incentive to manipulate.

Therefore, MANIP(*EMaj*) is PP-hard. □

The result of Theorem 6.10 holds for a manipulator i having a goal γ_i in propositional logic, without restrictions on the language: the next step would be to study if the problem stays hard when we focus on goals in $\mathcal{L}^{\vee}$. Given that the goals of the other agents can be arbitrary formulas, we conjecture it to be PP-hard as well. The manipulation problem for *TrueMaj* is also probably at least as hard as the WINDET problem (and the MANIP problem for *EMaj*), but its definition should include another parameter for the satisfaction of the agents.

6.4 Conclusions

This chapter has studied the strategic component of the framework of goal-based voting presented in Chapter 3. In particular, the focus has been on the three rules that we have proposed as adaptations of the issue-wise majority rule with varying degrees of resoluteness. We found that in the general case *EMaj*, *TrueMaj* and *2sMaj* are not immune to manipulation, even when the manipulator can only use limited strategies on their truthful goal (i.e., erosion and dilatation).

Looking for ways to make manipulation impossible, we restricted the language of an agent's goal to fragments of propositional logic: the language of conjunctions $\mathcal{L}^{\wedge}$, the language of disjunctions $\mathcal{L}^{\vee}$ and the language of exclusive disjunctions $\mathcal{L}^{\oplus}$. Restricting goals to the language $\mathcal{L}^{\wedge}$ makes manipulation impossible, as well as dilatation manipulation for the language $\mathcal{L}^{\vee}$. This suggests a promising direction for further research, on minimal restrictions to the language of goals to guarantee strategy-proofness of majoriatrian rules.

Finally, we also found that while they are not strategy-proof in the general case, the *EMaj* and *2sMaj* rules are PP-hard for an agent to manipulate, i.e., as hard as their winner determination problem which has been studied in Section 3.3.4. Therefore, even though the manipulation problem is in general hard to solve for an agent, it is not harder than computing the winner for the majority rules. We thus suspect the manipulation problem to become easy when the agents *all* have a goal in a restricted language for which the winner determination problem is easy.

Chapter 7

Influence Games over a Network

We have seen in Chapter 6 the strategic component of goal-based voting, where agents submit (possibly untruthful) goals to take a collective decision. In this chapter we focus on the process of how opinions are formed when agents are connected by a network. More precisely, we study the process of social influence, or propositional opinion diffusion, as per the work by Grandi et al. (2015) adding a game-theoretic component. In propositional opinion diffusion (POD) the agents' opinions over binary variables are propagated on a network by means of a chosen aggregation rule, and agents have no goals or strategic actions available. We define here the *games of influence* as a new class of infinite repeated games: at each step a player can choose whether to make her private opinion on some issues public or not, and she will update her current opinion according to the public opinions of the agents she trusts and via some aggregation procedure. As in Chapter 6 each player has a goal that she wants to achieve, this time expressed in a variant of the logic LTL which allows to consider the future opinions of the agents.

The following example illustrates the situations captured by influence games:

Example 7.1. Ann, Bob and Jesse will be asked to participate in a referendum about issue p at some point in the coming months. Ann thinks at the moment that the vote should go in favor of p, and she also knows that Bob and Jesse always listen to what she thinks on political matters. Before the referendum comes, Ann would like to find a consensus with Bob and Jesse about p (they either all approve p or they all approve $\neg p$), since she does not like to argue with her friends about politics, and in order to do so she will make use of her influence power over them.

Influence games provide a basic abstraction to explore the effects of a trust network on the behavior of agents, as well as allowing us to study game-theoretic solution concepts such as winning strategy, weak dominance and Nash equilibrium. The interplay between the network structure and the type of goal an agent holds has an impact on her strategic ability to achieve her goal.

In this chapter we will focus on two goal schemas: that of reaching *consensus* among a group of agents, akin to the one of Ann in Example 7.1, and that of *influencing* individuals towards one's opinion. As we shall see, introducing a network into an apparently simple influence problem makes positive results hard to find both from a strategic and computational perspective.

7.1 Framework

We generalize the POD model by separating the notions of private and public opinions via a notion of visibility (Section 7.1.1), and we adapt the diffusion process through aggregation to this more complex setting (Section 7.1.2).

7.1.1 Private Opinions and Unanimous Diffusion

Let $\mathcal{I} = \{p_1, \dots, p_m\}$ be a finite set of propositions or *issues* and let $N = \{1, \dots, n\}$ be a finite set of individuals or *agents*. Agents have opinions on issues in $\mathcal{I}$ in the form of a propositional interpretation or, equivalently, a binary vector:

Definition 7.1. The *private opinion* of agent i is a function $B_i : \mathcal{I} \to \{1, 0\}$ where $B_i(p) = 1$ and $B_i(p) = 0$ express, respectively, the agent's opinion that p is true and the agent's opinion that p is false.

Let $\boldsymbol{B} = (B_1, \dots, B_n)$ denote the profile of private opinions of agents in N, analogously to what we have seen for binary aggregation in Section 3.4.2. For simplicity we do not introduce any integrity constraint. We assume that each agent has the choice of using (or not) her influence power over her private opinion on each issue, as formalized in the following definition:

Definition 7.2. We call *visibility function* of agent i any map $V_i : \mathcal{I} \to \{1, 0\}$ where $V_i(p) = 1$ expresses that agent i's opinion on p is visible.

We denote by $\boldsymbol{V} = (V_1, \dots, V_n)$ the profile composed of the agents' visibility functions. By combining the private opinion with the visibility function of an agent, we can build her public opinion as a three-valued function on the issues.

Definition 7.3. Let B_i be the opinion of agent i and V_i her visibility function. The *public opinion* of i is a function $P_i : \mathcal{I} \to \{1, 0, ?\}$ such that

$$P_i(p) = \begin{cases} B_i(p) & \text{if } V_i(p) = 1 \\ ? & \text{if } V_i(p) = 0 \end{cases}$$

Again, $\boldsymbol{P} = (P_1, \dots, P_n)$ is the profile of public opinions of all the agents in N. We denote by $\boldsymbol{P}_C$ the restriction of public profile $\boldsymbol{P}$ to individuals in $C \subseteq N$.

Definition 7.4. A *state* is a tuple $S = (\boldsymbol{B}, \boldsymbol{V})$ where $\boldsymbol{B}$ is a profile of private opinions and $\boldsymbol{V}$ is a profile of visibility functions. The set of all states is denoted by $\mathcal{S}$.

A state thus consists of the profiles of private opinions and of visibilities of all the agents in N. The agents, in particular, are linked by an *influence network* modeled as a directed graph:

Definition 7.5. An *influence network* is a directed irreflexive graph $E \subseteq N \times N$, where $(i, j) \in E$ means that agent j is influenced by agent i.

We also refer to E as the *influence graph* and to individuals in N as the nodes of the graph. Let $Inf(i) = \{k \in N \mid (k, i) \in E\}$ be the set of *influencers* of agent i in the network E. Given a state S, this definition can be refined by considering $Inf^S(i, p) = \{k \in N \mid (k, i) \in E \text{ and } P_k(p) \neq ?\}$ to be the subset of i's influencers that are actually showing their private opinion about issue p.

Given a profile of public opinions and an influence network, we model the process of opinion diffusion by means of an aggregation function, which modifies the private opinion of an agent from the public opinions of other agents.

Definition 7.6. An *aggregation procedure* for agent i is a class of functions

$$F_{i,C} : \{0,1\}^{\mathcal{I}} \times \{0,1,?\}^{\mathcal{I} \times C} \longrightarrow \{0,1\}^{\mathcal{I}} \text{ for all } C \subseteq N \setminus \{i\}$$

that maps agent i's individual opinion and the public opinions of a set of agents C to agent i's individual opinion.

To simplify notation, we drop C from the subscript when clear from context. Many aggregation procedures have been considered in the literature on judgment aggregation, and they can be adapted to our setting. Notable examples are quota rules, where agents change their opinion if the number of people disagreeing with them is higher than a given quota, such as the majority rule (see Section 3.1.2 for other examples of aggregation rules). Unanimity is another instance of a quota rule, which we define here as we will use it in our results:

Definition 7.7. The *unanimous issue-by-issue aggregation procedure* is defined as:

$$F_i^U(B_i, \boldsymbol{P}_C)(p) = \begin{cases} B_i(p) & \text{if } C = \emptyset \\ x \in \{0,1\} & \text{if } P_k(p) = x \text{ for all } k \in C \text{ s.t. } P_k(p) \neq ? \\ B_i(p) & \text{otherwise} \end{cases}$$

That is, an individual will change her private opinion about issue p if and only if all agents in C (usually among her influencers, as we will see) publicly expressing their opinion are unanimous in disagreeing with her own.

7.1.2 Actions, Transitions and Individual Goals

In our model, agents can use their influence power over the issues by means of specific actions of type $\mathsf{reveal}(J)$ — i.e., the action of showing the opinion on issues in J, and $\mathsf{hide}(J)$ — i.e., the action of hiding the opinion on issues in J. We allow for simultaneous disclosure on multiple propositions. Let thus:

$$\mathcal{A} = \{(\mathsf{reveal}(J), \mathsf{hide}(J')) \mid J, J' \subseteq \mathcal{I} \text{ and } J \cap J' = \emptyset\}$$

be the set of individual actions. Each joint action $\boldsymbol{a} = (a_1, \ldots, a_n) \in \mathcal{A}^n$ induces a new state from a given one via a transition function:

Definition 7.8. The *transition function succ* : $\mathcal{S} \times \mathcal{A}^n \longrightarrow \mathcal{S}$ associates to each state S and joint action $\boldsymbol{a}$ a new state $S' = (\boldsymbol{B}', \boldsymbol{V}')$ as follows, for all $i \in N$ and $p \in \mathcal{I}$. For $a_i = (\mathsf{reveal}(J), \mathsf{hide}(J')) \in \mathcal{A}$:

- $V_i'(p) = \begin{cases} 1 & \text{if } p \in J \\ 0 & \text{if } p \in J' \\ V_i(p) & \text{otherwise} \end{cases}$
- $B_i' = F_i^U(B_i, \boldsymbol{P}'_{Inf(i)^S})$

where $\boldsymbol{P'}$ is the public profile obtained from private profile $\boldsymbol{B}$ and visibility profile $\boldsymbol{V'}$.

By a slight abuse of notation we denote with $\boldsymbol{a}(S)$ the state $succ(S, \boldsymbol{a})$ obtained from S and $\boldsymbol{a}$ by applying the transition function. We also use the following abbreviations: $\mathsf{skip} = (\mathsf{reveal}(\emptyset), \mathsf{hide}(\emptyset))$ for doing nothing, $\mathsf{reveal}(J) = (\mathsf{reveal}(J), \mathsf{hide}(\emptyset))$, $\mathsf{hide}(J) = (\mathsf{reveal}(\emptyset), \mathsf{hide}(J))$, and we drop curly parentheses in $\mathsf{reveal}(\{p\})$ and $\mathsf{hide}(\{p\})$. Our definition assumes that the influence process occurs after the actions have changed the visibility of the agents' opinions. Specifically, first, actions affect the visibility of opinions, and then each agent modifies her private opinion on the basis of those opinions of her influencers that have become public.

We are now ready to define the concept of *history*, describing the temporal aspect of agents' opinion dynamic:

Definition 7.9. Given a set of issues $\mathcal{I}$, a set of agents N, and aggregation procedures F_i for $i \in N$ over a network E, an *history* is an infinite sequence of states $H = (H_0, H_1, \dots)$ such that for all $t \in \mathbb{N}$ there exists a joint action $\boldsymbol{a}_t \in \mathcal{A}^n$ such that $H_{t+1} = \boldsymbol{a}_t(H_t)$. The set of all histories is denoted by $\mathcal{H}$.

Observe that Definition 7.9 restricts the set of all possible histories to those that correspond to a run of the influence dynamic described above. For notational convenience, for any $i \in N$ and for any $t \in \mathbb{N}$, we denote with $H^B_{i,t}$ agent i's private opinion in state H_t and with $H^V_{i,t}$ agent i's visibility function in state H_t.

Example 7.2. Consider the example in Figure 7.1, where the initial state is H_0 and the agents are $N = \{1, 2, 3\}$ such that $\mathit{Inf}(1) = \{2, 3\}$. Let $\boldsymbol{a}_0 = (\mathsf{skip}, \mathsf{skip}, \mathsf{reveal}(p))$ and $\boldsymbol{a}_1 = (\mathsf{skip}, \mathsf{hide}(p), \mathsf{skip})$ be the joint actions of the agents at the first two states. Namely, agent 3 reveals her opinion on p, and at the next step agent 2 hides hers. If all individuals are using the unanimous aggregation procedure, then states H_1 and H_2 result from applying the joint actions from state H_0. In state H_1, agent 1's private opinion about p has changed to 1, as all her influencers are publicly unanimous on p, while in H_2 no opinion is updated.

$$
\begin{array}{ccccc}
(\boldsymbol{B}_0, \boldsymbol{V}_0) & & (\boldsymbol{B}_1, \boldsymbol{V}_1) & & (\boldsymbol{B}_2, \boldsymbol{V}_2) \\
((011), (110)) & \xrightarrow[\boldsymbol{a}_0]{\qquad\qquad} & ((111), (111)) & \xrightarrow[\boldsymbol{a}_1]{\qquad\qquad} & ((111), (101)) \\
H_0 & & H_1 & & H_2
\end{array}
$$

FIGURE 7.1: The first three states of a history: in each state H_{0-2}, the private opinions of the three agents ($\boldsymbol{B}$) as well as their choice for visibility ($\boldsymbol{V}$) are displayed.

We define a language $\mathcal{L}_{\mathsf{LTL\text{-}I}}$ to express the goals of the agents using the temporal logic LTL introduced in Chapter 5. In particular, atoms in $\mathcal{L}_{\mathsf{LTL\text{-}I}}$ will be of the form $\mathsf{op}(i, p)$, to be read as "agent i's opinion is that p is true" (and since opinions are binary, $\neg\mathsf{op}(i, p)$ has to be read "agent i's opinion is that p is not true") and $\mathsf{vis}(i, p)$ to be read as "agent i's opinion about p is shown". We can also define the temporal operators 'eventually' ($\Diamond$) and 'henceforth' ($\Box$) as $\Diamond\varphi = \top\mathcal{U}\varphi$ and $\Box\varphi = \neg\Diamond\neg\varphi$.

The interpretation of $\mathcal{L}_{\mathsf{LTL\text{-}I}}$-formulas relative to histories is defined as follows:

Definition 7.10. Let H be a history, let φ be a formula of $\mathcal{L}_{\mathsf{LTL\text{-}I}}$ and let $k, k', k'' \in \mathbb{N}$. Then:

$$
\begin{array}{rcl}
H, k \models \mathsf{op}(i, p) & \Leftrightarrow & H^B_{i,k}(p) = 1 \\
H, k \models \mathsf{vis}(i, p) & \Leftrightarrow & H^V_{i,k}(p) = 1 \\
H, k \models \neg\varphi & \Leftrightarrow & H, k \not\models \varphi \\
H, k \models \varphi_1 \wedge \varphi_2 & \Leftrightarrow & H, k \models \varphi_1 \text{ and } H, k \models \varphi_2 \\
H, k \models \bigcirc\varphi & \Leftrightarrow & H, k+1 \models \varphi \\
H, k \models \varphi_1 \mathcal{U} \varphi_2 & \Leftrightarrow & \exists k' : (k \leq k' \text{ and } H, k' \models \varphi_2 \text{ and} \\
& & \forall k'' : \text{if } k \leq k'' < k' \text{ then } H, k'' \models \varphi_1)
\end{array}
$$

Formulas of $\mathcal{L}_{\mathsf{LTL\text{-}I}}$ will be used to express agents' goals on the iterative diffusion process. Since individuals do not have any influence on the initial state of the history, we will consider only goals of the form $\bigcirc\varphi$ and $\varphi\mathcal{U}\psi$, for any φ and ψ in $\mathcal{L}_{\mathsf{LTL\text{-}I}}$, which we denote as *goal formulas*.

For some subset of agents $C \subseteq \mathcal{N}$ and some issues $J \subseteq \mathcal{I}$ consider the following goals on consensus and influence in situations of opinion diffusion:

$$\begin{aligned}
\mathsf{cons}(C,J) &:= \Diamond\Box(\mathsf{pcons}(C,J) \vee \mathsf{ncons}(C,J)) \\
\mathsf{influ}(i,C,J) &:= \Diamond\Box \bigwedge_{p\in J} \big((\mathsf{op}(i,p) \to \bigcirc\mathsf{pcons}(C,p)) \\
&\qquad \wedge(\neg\mathsf{op}(i,p) \to \bigcirc\mathsf{ncons}(C,p))\big)
\end{aligned}$$

where:

$$\begin{aligned}
\mathsf{pcons}(C,J) &:= \bigwedge_{i\in C}\bigwedge_{p\in J} \mathsf{op}(i,p) \\
\mathsf{ncons}(C,J) &:= \bigwedge_{i\in C}\bigwedge_{p\in J} \neg\mathsf{op}(i,p).
\end{aligned}$$

Intuitively, an agent holding the $\mathsf{cons}(C,J)$ goal wants at some point in the history to reach a stable consensus either for or against the issues in J with the agents in C. The $\mathsf{influ}(i,C,J)$ goal expresses instead the idea that agent i wants to eventually gain a stable influence over the people in C about the issues in J (i.e., they will always hold her opinion at the next step).

7.2 Influence Games

We are now ready to combine all concepts introduced in the previous sections to give the definition of an influence game:

Definition 7.11. An *influence game* is a tuple $IG = (N, \mathcal{I}, E, F_i, S_0, \gamma_1, \ldots, \gamma_n)$ where $N, \mathcal{I}, E$ and S_0 are, respectively, a set of agents, a set of issues, an influence network, and an initial state, F_i for $i \in N$ is an aggregation procedure, and γ_i is agent i's goal.

For the sake of simplicity, in the remainder we will consider that all agents use the unanimous aggregation procedure of Definition 7.7.

In Section 7.2.1 we introduce the strategies that agents use in order to attain their goals, and in Section 7.2.2 we give the definitions of some solution concepts for the games of influence. The strategies and solution concepts we introduce here are common notions in game theory (see, e.g., the book by Gibbons, 1992), adapted to the context of agents having goals inducing dichotomous preferences.

7.2.1 Strategies

The individual strategies our agents will used are defined on states and are called *memory-less*, as the agent only consider the current state of the world to decide her next move:

Definition 7.12. A *memory-less strategy* for player i is a function $Q_i : \mathcal{S} \to \mathcal{A}$ that associates an action to every state.

A strategy profile is a tuple $\boldsymbol{Q} = (Q_1, \ldots, Q_n)$. For notational convenience, we also use $\boldsymbol{Q}$ to denote the function $\boldsymbol{Q} : \mathcal{S} \longrightarrow \mathcal{A}^n$ such that $\boldsymbol{Q}(S) = \boldsymbol{a}$ if and only if $Q_i(S) = a_i$, for all $S \in \mathcal{S}$ and $i \in N$. As the following definition highlights, every strategy profile induces a history if combined with an initial state:

Definition 7.13. Let S_0 be an initial state and let $\boldsymbol{Q}$ be a strategy profile. The *induced history* $H_{S_0,\boldsymbol{Q}} \in \mathcal{H}$ is defined as follows:

$$\begin{aligned} H_0(S_0, \boldsymbol{Q}) &= S_0 \\ H_{n+1}(S_0, \boldsymbol{Q}) &= succ(S_n, \boldsymbol{Q}(S_n)) \text{ for all } n \in \mathbb{N} \end{aligned}$$

It is important to observe that we study influence games as repeated games of complete information. The hide action can thus be interpreted in our model as "not exercising one's own influence", without any epistemic interpretation, and the reveal action as "persuade".

7.2.2 Solution Concepts

We start by defining the solution concept of *winning strategy*. Intuitively, Q_i is a winning strategy for player i if and only if by playing this strategy she will achieve her goal no matter what the other players do.

Definition 7.14. Let IG be an influence game and let Q_i be a strategy for player i. We say that Q_i is a *winning strategy* for player i in state S_0 if

$$H_{S_0,(Q_i,\boldsymbol{Q}_{-i})} \models \gamma_i$$

for all profiles $\boldsymbol{Q}_{-i}$ of strategies of players other than i.

Example 7.3. Let Ann, Bob and Jesse be three agents and suppose that $B_{Ann}(p) = 1$, $B_{Bob}(p) = 0$, $B_{Jesse}(p) = 0$ for $p \in \mathcal{I}$. Their influence network is as follows:

$$\begin{tikzcd} Jesse & \\ & Bob \arrow[ul] \\ Ann \arrow[uu] \arrow[ur] & \end{tikzcd}$$

Suppose Ann's goal is $\Diamond\Box\mathsf{op}(Jesse, p)$. Her winning memory-less strategy is to play $\mathsf{reveal}(p)$ in all states. Bob will be influenced to believe p at the second stage in the history and similarly for Jesse at the third stage, since her influencers are unanimous even if Bob plays $\mathsf{hide}(p)$.

As we will see in Section 7.3.1, the concept of winning strategy is rather strong in our setting. We then define the less demanding notion of *weak dominance*:

Definition 7.15. Let IG be an influence game and Q_i a strategy for player i. We say that Q_i is a *weakly dominant strategy* for player i and initial state S_0 if and only if for all profiles $\boldsymbol{Q}_{-i}$ of strategies of players other than i and for all strategies Q'_i we have:

$$H_{S_0,(Q'_i,\boldsymbol{Q}_{-i})} \models \gamma_i \Rightarrow H_{S_0,(Q_i,\boldsymbol{Q}_{-i})} \models \gamma_i$$

Example 7.4. Suppose that in Example 7.3 Ann thinks p, but does not influence Jesse any longer. Here, Ann does not have a winning strategy: if neither Bob nor Jesse believe p, it is sufficient for Bob to play $\mathsf{hide}(p)$ to make sure that Ann will never satisfy her goal. However, playing action $\mathsf{reveal}(p)$ is weakly dominant for Ann.

Finally, we introduce the concept of *Nash equilibrium* for influence games:

Definition 7.16. Let IG be an influence game and $\boldsymbol{Q}$ a strategy profile. We say that $\boldsymbol{Q}$ is a *Nash equilibrium* for initial state S_0 if and only if for all $i \in N$ and for all $Q'_i \in \mathcal{Q}_i$:

$$H_{S_0,(Q_i,\boldsymbol{Q}_{-i})} \models \gamma_i \text{ or } H_{S_0,(Q'_i,\boldsymbol{Q}_{-i})} \not\models \gamma_i.$$

Results on the computational problems of membership, existence and uniqueness of Nash equilibria will be described in Section 7.3.2.

7.3 Game-Theoretic and Computational Results

In this section we will present game-theoretic and computational results for influence games. In Section 7.3.1 we study the interplay between network structure and existence of solutions concepts for certain types of goals. Then, in Section 7.3.2 we summarize some results concerning the computational complexity of checking the presence of winning strategies or Nash equilibria in influence games.

7.3.1 Influence Network and Solution Concepts

In this section we show how a certain type of network may or may not ensure the existence of a certain solution concept given the goals of the agents. Our first result tells us that in a network without cycles of influence, if an agent wants to achieve the consensus goal then she has a winning strategy to do so.

Proposition 7.1. If E is a directed acyclic graph (DAG) such that $|Inf(i)| \leq 1$ for all agents $i \in N$, and if agent a has goal $\gamma_a := \mathsf{cons}(C_a, J)$ where $J \subseteq \mathcal{I}$ and $C_a := \{k \in N \mid a \in Inf(k)\} \cup \{a\}$, then agent a has a winning strategy.

Proof. Consider a DAG E and an agent a with goal γ_a. Let Q_a be the strategy associating to every state S action $\mathsf{reveal}(J)$. We want to show that $H_{S_0,(Q_a,\boldsymbol{Q}_{-a})} \models \gamma_a$ holds for all S_0 and $\boldsymbol{Q}_{-a}$. Consider the position of agent a in the graph for arbitrary S_0. In case there is no agent b such that $a \in Inf(b)$, the goal reduces to $\mathsf{cons}(\{a\}, J)$ which is always trivially satisfied. In case $Inf(a) = \emptyset$, by playing $\mathsf{reveal}(J)$ in S_0 and since every agent uses the unanimous aggregation rule, at stage 1 all child nodes of a will update their beliefs on J by copying a's opinion (she is their only influencer). Moreover, they can't change their opinions on J later on in the history.

On the other hand, suppose there is some agent b such that $a \in Inf(b)$ and some agent $c \in Inf(a)$. By assumption on E we thus have that $Inf(a) = \{c\}$ and $Inf(b) = \{a\}$. Hence, either at some point k in the history all ancestors of a will have reached consensus, such that by playing $\mathsf{reveal}(J)$ from point $k+1$ onwards the consensus among a and her child nodes will be maintained, or there is no such k. Since there is a unique path linking a to one of the source nodes of E, if her ancestors always disagree in the history it means that there is some agent among them who has a different opinion and who will never play $\mathsf{reveal}(J)$. Therefore, the opinion of a will nonetheless be stable and γ_a will be attained. $\square$

The assumption of acyclicity in the above result rules out the situation where all nodes in a cycle play $\mathsf{reveal}(J)$ and they start in S_0 by having alternating positive and negative opinions on the issues in J. Moreover, having at most one influencer per agent ensures each agent to have full control over their child nodes.

Proposition 7.1 also implies that for $\gamma_a := \Box\Diamond(\mathsf{pcons}(C_a, J) \vee \mathsf{ncons}(C_a, J))$ the same result holds. In fact, since eventually a will reach a stable consensus with her child nodes, it is always true that we can find some later point in the history where consensus holds. In general, however, Proposition 7.1 suggests that winning strategies are a strong solution concept, and the type of goals which can be ensured have a narrow scope.

If we focus on the less demanding concept of weak dominance, we may intuitively think that a strategy associating action $\mathsf{reveal}(J)$ to all states is weakly dominant for an agent a having goal $\gamma_a := \mathsf{influ}(a, C, J)$ for $C \subseteq N$, regardless of the network E or the initial state S_0. In fact, all agents use the monotonic aggregation rule F_i^U. Yet, we show in the following example that to satisfy goals of type γ_a as described, an agent could sometimes benefit from hiding her opinion.

Example 7.5. For four agents $N = \{1, 2, 3, 4\}$ and one issue $\mathcal{I} = \{p\}$ consider the network $E = \{(1, 2), (2, 3), (3, 4)\}$. Suppose agent 1 and 2 associate action $\mathsf{reveal}(p)$ to all states, and agent 3 associates action $\mathsf{hide}(p)$ only to the states where $1, 2$ and 3 agree on p. Let the goal of agent 2 be $\gamma_2 = \mathsf{influ}(2, \{4\}, \{p\})$. Consider the history below for these strategies, where goal γ_2 is not attained (we only represent $\boldsymbol{B}$):

$$\underset{H_0}{(0101)} \xrightarrow[\boldsymbol{a}_0]{} \underset{H_1}{(0010)} \xrightarrow[\boldsymbol{a}_1]{} \underset{H_2}{(0001)}$$

From state H_2 onward, given the strategies of the agents, the profile of opinions $\boldsymbol{B} = (0001)$ won't change. On the other hand, consider a strategy for agent 2 identical to the previous one, but for the fact that it associates to state H_0 action $\mathsf{hide}(p)$. This is what would happen:

$$\underset{H_0}{(0101)} \xrightarrow[\boldsymbol{a}_0]{} \underset{H_1}{(0000)}$$

From state H_1 onwards, given the strategies of the agents, the profile of opinions won't change. Thus, we found a network, an initial state H_0, and strategies for the other agents, such that agent 2 can be better off by hiding her opinion on p to satisfy her influence goal γ_2.

We can now see an easy example of how the network structure and the agents' goals can yield a Nash equilibrium.

Proposition 7.2. Let E be a cycle for $N = \{1, 2\}$. If $\gamma_1 = \gamma_2 = \mathsf{cons}(N, J)$, where $J \subseteq \mathcal{I}$, then there exists a Nash equilibrium for any initial state S_0.

Proof. To attain their goal the agents must coordinate on the issues in J on which they disagree in S_0. In fact, in case at some stage k of the history they both play $\mathsf{hide}(p)$ for $p \in J$ their private opinion would stay the same at stage $k+1$. If they both play $\mathsf{reveal}(p)$, at the next stage they would just swap their opinions on p (since they are each other's only influencers and they both use the unanimous rule). Hence, agent 1 has to play $\mathsf{reveal}(p)$ whenever the other agent is playing $\mathsf{hide}(p)$ so that at the next stage in the history he will have copied her opinion, while she would have not changed hers — and similarly if the other agent is playing $\mathsf{reveal}(p)$.

Consider thus an arbitrary strategy Q_1 for agent 1 and an initial state S_0. Construct now strategy Q_2 for agent 2 associating action $\mathsf{reveal}(J^S)$ to all states where strategy Q_1 associates action $\mathsf{hide}(J^S)$ for $J^S = \{p \in J \mid b_{1p} = 1 - b_{2p} \text{ in } S\}$, and viceversa for $\mathsf{reveal}(J^S)$. By the above reasoning, the strategy profile $\boldsymbol{Q} = (Q_1, Q_2)$ generates a history that satisfies both γ_1 and γ_2, and therefore is a Nash equilibrium. The same construction can be done for an arbitrary strategy of agent 2. □

The results in this section shed light on how the different components of an influence game (i.e., the network, the strategies and the goals) interact with one another.

In particular, we have seen in Proposition 7.1 how an agent can achieve her consensus goal over the people she influences given that the influence network she belongs to is a directed acyclic graph. We also have a positive result for the consensus goal in case two agents mutually influence one another, as Proposition 7.2 tells us that there is a Nash equilibrium no matter the initial state.

The most interesting result is perhaps the one illustrated by Example 7.5. In fact, even though the model of influence given by influence games is pretty simple, its dynamic is non-trivial: an agent can sometimes benefit from not using her influence power in order to attain her influence goal.

7.3.2 Computational Complexity

We briefly present some preliminary complexity results that have been obtained for influence games and Nash equilibria, leaving a more detailed analysis for future work. There are three standard computational problems to study for games and the Nash equilibrium solution concept. The first one asks, given an influence game and a strategy profile, whether the profile is a Nash equilibrium of the game (*membership*). The second one asks, given an influence game, if there is a Nash equilibrium for the game (*existence*). Finally, the third one asks, given an influence game, if there is a unique Nash equilibrium for the game (*uniqueness*).

For memory-less strategies, the three problems are in PSPACE. The result is obtained by providing a translation of memory-less strategies, the unanimous aggregation function and of the opinion diffusion process as formulas of the logic LTL–I. Then, the problem is reduced to validity checking for LTL, which is in PSPACE. Additional details and proofs can be found in the paper by Grandi et al. (2017).[1]

We will see again influence games in Chapter 8, where we will study the computational problem of checking, given a specific game and an agent, if the agent has a winning strategy in that game (see Proposition 8.2).

7.4 Conclusions

In this chapter we extended a model of opinion diffusion on networks by Grandi et al. (2015) with a strategic component modeling agents who can choose to use or not their influence power. The diffusion of opinions is an important process to study, especially if these are linked to some future collective decision agents have to take (such as the ones of Chapter 3). We defined influence games, a class of iterated games where agents have goals expressed in LTL. Our results on the interaction between the type of goals, the network structure and the game-theoretic solution concepts showed that agents were greatly empowered by these basic actions. Interestingly, for the influence goal and the unanimous aggregation rule, a strategy always using an agent's influence power is not weakly dominant. From a computational point of view, the problems of checking membership, existence and uniqueness of a Nash equilibrium in influence games are all in PSPACE. In line with the work of Chapter 6, we could enrich the work presented here by adding actions for the agents to lie about their private opinions. This would give them even more sophisticated strategies to attain their goals. Moreover, we could use different aggregation procedures to update the opinions of the agents.

[1] The paper also studies the complexity problems for agents having *perfect-recall* strategies (i.e., where agents choose an action based on the history up to that stage), which we did not touch here but are a standard notion for iterated games.

Chapter 8

Shared Control in Concurrent Game Structures

In Chapter 5 we have seen the definition of Boolean games as a strategic model of agents with propositional goals having exclusive control over variables. Boolean games have also been generalized to their iterated version (Gutierrez et al., 2015, 2013), where the agents' goals are formulas of Linear Temporal Logic LTL, and an agent's strategy determines an assignment to the variables she controls for every round of the game. Intuitively, an iterated Boolean game would be a situation like the one of Example 5.1, with the sole difference that the potluck now is a recurring event and thus the three agents can have goals concerning the future. In Chapter 7 we have seen another example of an iterated game, i.e., influence games, where agents holding goals in LTL influence each other about opinions over issues.

In order to reason about such iterated games we can use Alternating-time Temporal Logic ATL^* (introduced in Section 5.1) to express, for instance, the fact that an agent has a winning strategy to make her goal true. Formulas of ATL^* can be interpreted over the Concurrent Game Structures (CGS) of Definition 5.1, but observe how there the actions and the transition functions are left completely general. Belardinelli and Herzig (2016) defined a particular subclass of CGS in order to model situations as the ones presented in Example 5.1, where the control over propositional variables is exclusive (CGS-EPC): their definition will be given in Section 8.1.

In this chapter we study models in which we relax the assumption that propositional control is exclusive: consider for instance Example 3.1, where the control over the variables is *shared* among the agents and no single agent can unilaterally decide the truth value of some variable. We thus introduce Concurrent Game Structures with Shared Propositional Control (CGS-SPC) and show their relationship with different classes of games studied in literature, including (variations of) iterated Boolean games. The main result will be that verification of ATL^* formulas on CGS-SPC can be reduced to verification in CGS-EPC, via a polynomial translation of a CGS-SPC into a CGS-EPC by means of a dummy agent who controls the value of the shared variables and simulates the transition function.

8.1 Framework

In this section we present two classes of concurrent game structures with propositional control, used for the interpretation of logics for individual and collective strategies introduced in Chapter 5.

The *concurrent game structures with exclusive propositional control* (CGS-EPC) are formally defined by Belardinelli and Herzig (2016) as:[1]

Definition 8.1. A *concurrent game structure with exclusive propositional control* is a tuple $\mathcal{G} = \langle N, \Phi_1, \dots, \Phi_n, S, d, \tau \rangle$, where:

- $N = \{1, \dots, n\}$ is a set of *agents;*
- $\Phi = \Phi_1 \cup \cdots \cup \Phi_n$ is a set of *propositional variables* partitioned in n disjoint subsets, one for each agent;
- $S = 2^{\Phi}$ is the set of *states*, corresponding to all valuations over Φ;
- $d : N \times S \to (2^{\mathcal{A}} \setminus \emptyset)$, for $\mathcal{A} = 2^{\Phi}$, is the *protocol function*, with $d(i,s) \subseteq \mathcal{A}_i$ for $\mathcal{A}_i = 2^{\Phi_i}$;
- $\tau : S \times \mathcal{A}^n \to S$ is the *transition function* such that $\tau(s, \alpha_1, \dots, \alpha_n) = \bigcup_{i \in N} \alpha_i$.

Intuitively, a CGS-EPC describes the interactions of a group N of agents, each one of them controlling (exclusively) a set $\Phi_i \subseteq \Phi$ of propositional atoms. The state of the CGS is an interpretation of the atoms in Φ. In each such state the protocol function returns which actions an agent can execute.

The intuitive meaning of action $\alpha_i \in d(i,s)$ is "assign true to all atoms in α_i, and false to all atoms in $\Phi_i \setminus \alpha_i$". We introduce the $idle_s$ action as $\{p \in \Phi_i \mid s(p) = 1\}$, for every $i \in N$, $s \in S$. With an abuse of notation we write $d(i,s) = \alpha$ whenever $d(i,s)$ is a singleton $\{\alpha\}$.

Each state $s \in S$ can be equivalently seen as a function $s : \Phi \to \{0,1\}$ returning the truth value of a propositional variable in s, so that $s(p) = 1$ iff $p \in s$. Given $\alpha = (\alpha_1, \dots, \alpha_n) \in \mathcal{A}^n$, we equally see each $\alpha_i \subseteq \Phi_i$ as a function $\alpha_i : \Phi_i \to \{0,1\}$ returning the choice of agent i for p under action α.

We now generalize concurrent game structures for propositional control by relaxing the exclusivity requirement on the control of propositional variables. We thus introduce concurrent game structures with shared propositional control CGS-SPC:

Definition 8.2. A *concurrent game structure with shared propositional control* is a tuple $\mathcal{G} = \langle N, \Phi_0, \dots, \Phi_n, S, d, \tau \rangle$ such that:

- N, S, and d are defined as in Definition 8.1 with $\mathcal{A} = 2^{\Phi \setminus \Phi_0}$;
- $\Phi = \Phi_0 \cup \Phi_1 \cup \cdots \cup \Phi_n$ is a set of *propositional variables*, where $\Phi_0 \cup \Phi_1 \cup \cdots \cup \Phi_n$ is not necessarily a partition and $\Phi_0 = \Phi \setminus (\Phi_1 \cup \cdots \cup \Phi_n)$;
- $\tau : S \times \mathcal{A}^n \to S$ is the *transition function*.

In CGS-SPC the same atom can be controlled by multiple agents, and propositional control is not exhaustive. Additionally, the actions in $\mathcal{A}$ do not take into account propositional variables in Φ_0 because they are not controlled by anyone (though their truth value might change according to the transition function). The transition function combines the various actions and produces a consistent successor state according to some rule.[2] Simple examples of such rules include introducing a threshold $m_p \in \mathbb{N}$ for every variable p, thus setting $p \in \tau(s, \alpha)$ if and only if the

[1] More precisely, the CGS-EPC we consider here correspond to the *weak* version defined by Belardinelli and Herzig (2016), as opposed to a strong version where $d(i,s) = \mathcal{A}_i$ for all $i \in N$ and $s \in S$.

[2] The definition of τ as an arbitrary function is needed to represent complex aggregation procedures such as those used in the games described in Section 8.2.2.

number of agents i with $p \in \alpha_i$ is greater than m_p: these rules are analogous to the *TrSh* rules introduced in Section 3.1.2.

A CGS-EPC can be seen as a special case of CGS-SPC in which every atom is controlled by exactly one agent, and therefore $\{\Phi_0, \dots, \Phi_n\}$ is a partition of Φ. Moreover, τ is given in a specific form as per Definition 8.1.

8.2 Examples of CGS in Iterated Games

In order to illustrate the concepts introduced in Section 8.1 we present three examples of how to use CGS-SPC to model iterated games, i.e., *iterated boolean games* (Gutierrez et al., 2015), *influence games* (Grandi et al., 2017), and *aggregation games* (Grandi et al., 2019). We then relate the satisfiability of a certain ATL^* formula on a CGS-SPC to an agent having a winning strategy for her goal in the game.

8.2.1 Iterated Boolean Games

An *iterated Boolean game* is a tuple $\langle \mathcal{G}, \gamma_1, \dots, \gamma_n \rangle$ such that (*i*) $\mathcal{G}$ is a CGS-EPC with a trivial protocol (i.e., for every $i \in N$, $s \in S$, $d(i,s) = \mathcal{A}_i$); and (*ii*) for every $i \in N$, the *goal* γ_i is an LTL-formula.

We can generalize the above to *iterated Boolean games with shared control* as follows:

Definition 8.3. An *iterated Boolean game with shared control* is a tuple $\langle \mathcal{G}, \gamma_1, \dots, \gamma_n \rangle$ where:

(i) $\mathcal{G}$ is a CGS-SPC;

(ii) for every $i \in N$, the *goal* γ_i is an LTL-formula.

Just like CGS-SPC generalize CGS-EPC, iterated Boolean games with shared control generalize standard iterated Boolean games. For both types of control (exclusive or shared) we can thus link the existence of a winning strategy for an agent in an iterated Boolean game (with exclusive or shared control, respectively) to the model-checking problem of a simple ATL^* formula, as expressed in the following straightforward proposition:

Proposition 8.1. An agent i in an iterated Boolean game has a winning strategy for goal γ_i and state s if and only if formula $\langle\langle \{i\} \rangle\rangle \gamma_i$ is satisfied in ($\mathcal{G}$,s).

8.2.2 Influence Games

We associate here to an influence game $IG = \langle N, \Phi, E, S_0, \{F_{i,Inf(i)}\}_{i \in N}, \{\gamma_i\}_{i \in N} \rangle$ as per Definition 7.11 a CGS-SPC $\mathcal{G}' = \langle N', \Phi'_0, \dots, \Phi'_n, S', d', \tau' \rangle$ by letting $N' = N$; $\Phi'_0 = \{\mathsf{op}(i,p) \mid i \in N, p \in \Phi\}$; $\Phi'_i = \{\mathsf{vis}(i,p) \mid p \in \Phi\}$ for $i \in N'$; $S' = 2^{\Phi'}$; $d'(i,s') = 2^{\Phi'_i}$ for $s' \in S'$; and finally for state $s' \in S'$ and action α' we let:

$$\tau'(s', \alpha')(\varphi) = \begin{cases} \alpha'_i(\mathsf{vis}(i,p)) & \text{if } \varphi = \mathsf{vis}(i,p) \\ F_{i,Inf(i)}(\vec{a}, \vec{b})_{|p} & \text{if } \varphi = \mathsf{op}(i,p) \end{cases}$$

where vectors $\vec{a} = (a_1, \ldots, a_{|\Phi|})$ and $\vec{b} = (b_1, \ldots, b_{|\Phi|})$ are defined as follows, for $k \in$ *Inf(i)*:

$$a_p = \begin{cases} 1 & \text{if } \mathsf{op}(i,p) \in s' \\ 0 & \text{otherwise} \end{cases}$$

$$b_p = \begin{cases} 1 & \text{if } \alpha_k(\mathsf{vis}(k,p)) = 1 \text{ and } \mathsf{op}(k,p) \in s' \\ 0 & \text{if } \alpha_k(\mathsf{vis}(k,p)) = 1 \text{ and } \mathsf{op}(k,p) \notin s' \\ ? & \text{if } \alpha_k(\mathsf{vis}(k,p)) = 0 \end{cases}$$

Vector $\vec{a}$ represents the opinion of agent i over the issues at state s', while vector $\vec{b}$ represents the opinions of i's influencers over the issues, in case they are using their influence power. In particular, '?' indicates that the influencers of i in *Inf(i)* are not using their influence power.

Proposition 8.2. Agent i in influence game IG has a winning strategy for goal γ_i and state S_0 if and only if formula $\langle\langle\{i\}\rangle\rangle\gamma_i$ is satisfied in the associated CGS-SPC and state s' corresponding to S_0.

Proof. Let IG be an influence game and let $\mathcal{G}'$ be the CGS-SPC associated to it. Consider now an arbitrary agent i and suppose that i has a winning strategy in IG for her goal γ_i in S_0. A memory-less strategy σ_i for agent i in an influence game maps to each state actions of type $(\mathsf{reveal}(J), \mathsf{hide}(J'))$, where $J, J' \subseteq \Phi$ and $J \cap J' = \emptyset$. For any state s in IG, consisting of opinions and visibilities, consider the state s' in $\mathcal{G}'$ where $B_i(p) = 1$ if and only if $\mathsf{op}(i,p) \in s'$ and $V_i(p) = 1$ if and only if $\mathsf{vis}(i,p) \in s'$. We now construct the following strategy for $\mathcal{G}'$:

$$\sigma_i'(s') = \{\mathsf{vis}(i,p) \mid p \in J \text{ for } \sigma_i(s) = (\mathsf{reveal}(J), \mathsf{hide}(J'))\}$$

By the semantics of the $\langle\langle\{i\}\rangle\rangle$ operator provided in Section 5.1, and by the definition of winning strategy, the statement follows easily from our construction of $\mathcal{G}'$. □

The above translation sheds light over the control structure of the variables of type $\mathsf{op}(i,p)$. In fact, we can now see that $\mathsf{op}(i,p) \in \Phi_0'$ for all $i \in N$ and $p \in \Phi$, i.e., opinions are not controlled directly by agents.

8.2.3 Aggregation Games

Individuals facing a collective decision have goals on the outcome of the voting process, outcome that is jointly controlled by all individuals in the group. For instance, a vote on a single binary issue using the majority rule corresponds to a game with a single variable controlled by all individuals, the majority rule playing the role of the transition function.

Similar situations have been modelled as one-shot games called *aggregation games* (Grandi et al., 2019), and we here define them for iterated decision processes:

Definition 8.4. An *iterated aggregation game* is a tuple $AG = \langle N, \Phi, F, \gamma_1, \ldots, \gamma_n \rangle$ where:

- N is a set of *agents*;
- $\Phi = \{p_1, \ldots, p_m\}$ are variables representing *issues*;
- $F : \{0,1\}^{N \times \Phi} \to \{0,1\}$ is an *aggregation function*, i.e., a Boolean function associating a collective decision to the opinions of the agents on the issues;

- γ_i for $i \in N$ is an *individual goal* for each agent, i.e., a LTL formula over Φ.

Individuals at each stage of an aggregation game only have information about the current valuation of variables in Φ, resulting from the aggregation of their individual opinions. Analogously to Proposition 8.2, we can obtain the following result:

Proposition 8.3. Agent i in AG has a winning strategy for goal γ_i in s if and only if formula $\langle\langle\{i\}\rangle\rangle\gamma_i$ is satisfied in the associated CGS-SPC in the corresponding state s'.

Proof. From an iterated aggregation game $AG = \langle N, \Phi, F, \gamma_1, \ldots, \gamma_n\rangle$, construct a CGS-SPC $\mathcal{G}' = \langle N', \Phi', S', d', \tau'\rangle$ as follows. Let $N' = N$; $\Phi'_i = \Phi$ for all $i = 1, \ldots, n$; and $\Phi'_0 = \emptyset$. Hence, each agent controls all variables. Let the set of actions available to each player be $d'(i, s) = 2^{\Phi'}$ for all i and s, and the transition function τ' be such that $\tau'(s, \alpha_1, \ldots, \alpha_n) = F(\alpha_1, \ldots, \alpha_n)$. The statement then follows easily. □

An example of an iterated aggregation game is the setting of iterative voting, where individuals hold preferences about a set of candidates and iteratively change the result of the election in their favor until a converging state is reached.

8.3 Relationship between Exclusive and Shared CGS

We prove here our main result, namely that the shared control of a CGS-SPC can be simulated in a CGS-EPC having exclusive control. In particular, any formula in ATL^* satisfied in some CGS-SPC can be translated in polynomial time into an ATL^*-formula satisfied in a CGS-EPC. To do so, we introduce a dummy agent to simulate the aggregation function and we make use of an additional turn-taking atom which allows us to distinguish the states where the agents choose their actions from those in which the aggregation process takes place. Agents have memory-less strategies.

We begin by inductively defining a translation function tr within ATL^*. Intuitively, tr translates every ATL^*-formula χ into a formula $tr(\chi)$ having roughly the same meaning, but where the one-step "next" operator is doubled:

$$\begin{array}{lcl} tr(p) & = & p \\ tr(\neg\chi) & = & \neg tr(\chi) \\ tr(\chi \vee \chi') & = & tr(\chi) \vee tr(\chi') \\ tr(\bigcirc\chi) & = & \bigcirc\bigcirc\, tr(\chi) \\ tr(\chi\mathcal{U}\chi') & = & tr(\chi)\mathcal{U}tr(\chi') \\ tr(\langle\langle C\rangle\rangle\chi) & = & \langle\langle C\rangle\rangle tr(\chi) \end{array}$$

where $p \in \Phi$, $C \subseteq N$, and χ, χ' are either state- or path-formulas. The translation is easily seen to be polynomial.

We then map a given CGS-SPC to a CGS-EPC:

Definition 8.5. Let $\mathcal{G} = \langle N, \Phi_0, \ldots, \Phi_n, S, d, \tau\rangle$ be a CGS-SPC. The CGS-EPC corresponding to $\mathcal{G}$ is $\mathcal{G}' = \langle N', \Phi'_1, \ldots, \Phi'_n, S', d', \tau'\rangle$ where:

- $N' = N \cup \{*\}$;
- $\Phi' = \Phi \cup \{turn\} \cup \{c_{ip} \mid i \in N \text{ and } p \in \Phi_i\}$ and Φ' is partitioned as follows, for agents in N':

$$\Phi'_i = \{c_{ip} \in \Phi' \mid p \in \Phi_i\}$$
$$\Phi'_* = \{turn\} \cup \Phi$$

- $S' = 2^{\Phi'}$. For every $s' \in S'$, let $s = (s' \cap \Phi) \in S$ be the *restriction* of s' on Φ;
- d' is defined according to the truth value of $turn$ in s'. Specifically, given $\alpha_i \in \mathcal{A}_i$, let $\alpha'_i = \{c_{ip} \in \Phi'_i \mid p \in \alpha_i\} \in \mathcal{A}'_i$. Then, for $i \in N$ we let:

$$d'(i, s') = \begin{cases} \{\alpha'_i \in \mathcal{A}'_i \mid \alpha_i \in d(i, s)\} & \text{if } s'(turn) = 0 \\ \emptyset & \text{if } s'(turn) = 1 \end{cases}$$

For agent * we define:

$$d'(*, s') = \begin{cases} +turn & \text{if } s'(turn) = 0 \\ \tau(s, \alpha), \text{ for } \alpha_i(p) = s'(c_{ip}) & \text{if } s'(turn) = 1 \end{cases}$$

where $+turn = idle_s \cup \{turn\}$.

- τ' is defined as per Definition 8.1, i.e., $\tau'(s', \alpha') = \bigcup_{i \in N'} \alpha'_i$.

Intuitively, in the CGS-EPC $\mathcal{G}'$ every agent $i \in N$ controls local copies c_{ip} of atoms $p \in \Phi$. The aggregation function τ in $\mathcal{G}$ is mimicked by the dummy agent $*$, whose role is to observe the values of the various c_{ip}, then perform an action to aggregate them and set the value of p accordingly. Observe that agent $*$ acts only when the $turn$ variable is true, in which case all the other agents set all their variables to false, i.e., they all play $\emptyset$. This is to ensure the correspondence between memory-less strategies of $\mathcal{G}$ and $\mathcal{G}'$, as shown in Lemma 8.2.

The size of game $\mathcal{G}'$ is polynomial in the size of $\mathcal{G}$, and $\mathcal{G}'$ can be constructed in polynomial time from $\mathcal{G}$. To see this, observe that an upper bound on the number of variables is $N \times \Phi$. While we can associate to each state $s' \in S'$ a state $s = s' \cap \Phi$ in S, when given a state $s \in S$ there are multiple states s' that agree with s on Φ. The purpose of the next definition is to designate one such state as the *canonical* one:

Definition 8.6. For every $s \in S$, we define the *canonical state* $s'_\star = \{s' \in S' \mid s' \cap \Phi = s \text{ and } s(p) = 0 \text{ for } p \notin \Phi\}$.

Observe that, in particular, in all canonical states atom $turn$ is false. The following example illustrates this concept:

Example 8.1. Consider $\Phi = \{p, q\}$ and $N = \{1, 2\}$. Let then $\Phi_1 = \{p\}$ and $\Phi_2 = \{p, q\}$. We thus have that $\Phi' = \{p, q, c_{1p}, c_{2p}, c_{2q}, turn\}$. If $s = \{p\}$, we have for instance that $s' \cap \Phi = s$ for $s' = \{p, c_{1p}\}$. On the other hand, $s'_\star = \{p\}$.

We now define a correspondence between paths of $\mathcal{G}$ and $\mathcal{G}'$. For notational convenience, we indicate with $\lambda[k]_{|\Phi} = \lambda[k] \cap \Phi$, the restriction of state $\lambda[k]$ to variables in Φ. Given a path λ' of $\mathcal{G}'$, consider the unique infinite sequence of states λ associated to λ' defined as follows:

$$\lambda[k] = \lambda'[2k]_{|\Phi} = \lambda'[2k{+}1]_{|\Phi} \text{ for all } k \in \mathbb{N}. \quad (\dagger)$$

On the other hand, there are multiple sequences λ' that can be associated with a path λ, so that $(\dagger)$ holds true. In fact, we only know how the variables in Φ behave, while the truth values of the other variables can vary. We now use condition $(\dagger)$ to characterize the paths of $\mathcal{G}$ and $\mathcal{G}'$ that can be associated:

Lemma 8.1. Given a CGS-SPC $\mathcal{G}$ and the corresponding CGS-EPC $\mathcal{G}'$, the following is the case:

$$\begin{tikzcd}
\lambda[0] \arrow[rr, "(\alpha_1 \ldots \alpha_n)[0]"] \arrow[d, dashed] & & \lambda[1] \arrow[rr, "(\beta_1 \ldots \beta_n)[1]"] \arrow[d, dashed] & & \lambda[2] \arrow[rr, "(\delta_1 \ldots \delta_n)[2]"] \arrow[d, dashed] & & \cdots \\
\lambda'[0] \arrow[r, "\alpha'_1 \ldots \alpha'_n", "+turn"'] & \lambda'[1] \arrow[r, "\emptyset", "\tau(\lambda'[1]_{|\Phi}, \alpha)"'] & \lambda'[2] \arrow[r, "\beta'_1 \ldots \beta'_n", "+turn"'] & \lambda'[3] \arrow[r, "\emptyset", "\tau(\lambda'[3]_{|\Phi}, \beta)"'] & \lambda'[4] \arrow[r, "\delta'_1 \ldots \delta'_n", "+turn"'] & \lambda'[5] \arrow[r, "\emptyset", "\tau(\lambda'[5]_{|\Phi}, \delta)"'] & \cdots
\end{tikzcd}$$

FIGURE 8.1: A path λ in a CGS-SPC $\mathcal{G}$ and its associated path λ' in a CGS-EPC $\mathcal{G}'$.

1. for all paths λ' of $\mathcal{G}'$, sequence λ satisfying condition (†) is a path of $\mathcal{G}$;

2. for all paths λ of $\mathcal{G}$, for all sequences λ' satisfying (†), λ' is a path of $\mathcal{G}'$ if and only if for all k there exists a $\mathcal{G}$-action $\alpha[k]$ such that $\lambda[k] \xrightarrow{\alpha[k]} \lambda[k{+}1]$ and states $\lambda'[2k{+}1]$ and $\lambda'[2k{+}2]$ can be obtained from state $\lambda'[2k]$ by performing actions $(\alpha'_1, \ldots, \alpha'_n, +turn)$ and then $(\emptyset_1, \ldots, \emptyset_n, \tau(\lambda'[2k{+}1]_{|\Phi}, \alpha))$.

Proof. We first prove (1) by showing that λ is a path of $\mathcal{G}$, i.e., that for every k there is an action α that leads from $\lambda[k]$ to $\lambda[k{+}1]$. Suppose that $\lambda'[2k] \xrightarrow{\alpha'[2k]} \lambda'[2k{+}1] \xrightarrow{\alpha'[2k+1]} \lambda'[2(k{+}1)]$ for action $\alpha'[2k] = (\alpha'_1, \ldots, \alpha'_n, +turn)$ and action $\alpha'[2k{+}1] = (\emptyset_1, \ldots, \emptyset_n, \tau(\lambda'[2k{+}1]_{|\Phi}, \alpha))$. Then, we observe that we can move from state $\lambda[k] = \lambda'[2k]_{|\Phi} = \lambda'[2k{+}1]_{|\Phi}$ to $\lambda[k{+}1] = \lambda'[2k{+}2]_{|\Phi}$ by performing action $(\alpha_1, \ldots, \alpha_n)$ such that $\alpha_i = \{p \in \Phi \mid c_{ip} \in \alpha'_i\}$ for every $i \in N$.

As for (2), the right-to-left direction is clear. For the left-to-right direction, let λ' be a path associated to λ. From (†) we know that for any k we have that $\lambda'[2k]_{|\Phi} = \lambda[k]$ and $\lambda'[2k{+}2]_{|\Phi} = \lambda[k{+}1]$. Now by Definition 8.5, the only actions available to the players at $\lambda'[2k]$ are of the form $(\alpha'_1, \ldots, \alpha'_n, +turn)$, and the only action available at $\lambda'[2k{+}1]$ is $(\emptyset_1, \ldots, \emptyset_n, \tau(\lambda'[2k{+}1]_{|\Phi}, \alpha))$. We can thus obtain the desired result by considering action $\alpha[k] = (\alpha_1, \ldots, \alpha_n)$, where $\alpha_i = \{p \in \Phi_i \mid c_{ip} \in \alpha'_i\}$ for each $i \in N$, and by observing that by (†) we have $\tau(\lambda'[2k+1]_{|\Phi}, \alpha) = \tau(\lambda[k], \alpha)$. □

Figure 8.1 illustrates the two paths λ and λ' in the proof of Lemma 8.1. The second part of the lemma characterizes the set of $\mathcal{G}'$-paths λ' associated to a $\mathcal{G}$-path λ: for any sequence of $\mathcal{G}$-actions that can generate path λ, we can construct a distinct $\mathcal{G}'$-path λ' that corresponds to λ, where the sequence of actions can be reconstructed by reading the values of the variables in Φ'_i in odd states $\lambda[2k+1]$.

From this set of $\mathcal{G}'$-paths λ' we can specify a subset of *canonical* paths as follows:

Definition 8.7. For a path λ of $\mathcal{G}$, a *canonical associated path* $\lambda'_\star$ of $\mathcal{G}'$ is any path λ' such that (†) holds and $\lambda'[0] = \lambda[0]'_\star$.

That is, a canonical path λ' associated to λ starts from the canonical state $\lambda[0]'_\star$ associated to $\lambda[0]$. The following example clarifies the concepts just introduced:

Example 8.2. Consider a CGS-SPC $\mathcal{G}$ with $N = \{1,2\}$ and $\Phi = \{p,q\}$ such that $\Phi_1 = \{p\}$ and $\Phi_2 = \{p,q\}$. Let $d(i,s) = 2^{\Phi_i}$ for all $i \in N$ and $s \in S$, and let $\tau(s,\alpha)(p) = 0$ if and only if $\alpha_1(p) = \alpha_2(p) = 0$, while $\tau(s,\alpha)(q) = \alpha_2(q)$ for all $s \in S$. Namely, issue p becomes true if at least one agent makes it true, while issue q follows the decision of agent 2. Let now $\lambda = s_0 s_1 \ldots$ be a path of $\mathcal{G}$ such that $s_0 = \{p\}$ and $s_1 = \{p,q\}$. Observe that there are many actions α such that $\tau(s_0,\alpha) = s_1$: namely, the one where both agents set p to true, or where just one of them does (and agent 2 sets q to true).

Construct now the CGS-EPC $\mathcal{G}'$ as in Definition 8.5 and consider the following four sequences $\lambda' = s'_0 s'_1 s'_2 \dots$ where:

(a) $s'_0 = \{p\}, s'_1 = \{c_{1p}, c_{2p}, c_{2q}, p, turn\}, s'_2 = \{p, q\}, \dots$

(b) $s'_0 = \{p\}, s'_1 = \{c_{1p}, c_{2q}, p, turn\}, s'_2 = \{p, q\}, \dots$

(c) $s'_0 = \{p, c_{1p}\}, s'_1 = \{c_{1p}, c_{2q}, p, turn\}, s'_2 = \{p, q\}, \dots$

(d) $s'_0 = \{p\}, s'_1 = \{c_{2q}, p, turn\}, s'_2 = \{p, q\}, \dots$

Observe that (a) and (b) are both examples of canonical paths (up to the considered state), corresponding to two actions that might have led from s_0 to s_1 in $\mathcal{G}$. On the other hand, (c) is a non-example while being a path of $\mathcal{G}'$ satisfying (†), since s'_0 is not canonical. Finally, sequence (d) satisfies (†) but it is not a path of $\mathcal{G}'$, since it is not possible to obtain s'_2 from s'_1.

The next result extends the statement of Lemma 8.1 to paths generated by a specific strategy. Given a $\mathcal{G}'$-strategy σ'_C and a state $s' \in S'$, let $\Pi(out(s', \sigma'_C)) = \{\lambda \mid \lambda' \in out(s', \sigma'_C)\}$, i.e., all the "projections" of paths λ' in $out(s', \sigma'_C)$ to paths λ in $\mathcal{G}$, obtained through (†).

Lemma 8.2. Given a CGS-SPC $\mathcal{G}$, the corresponding CGS-EPC $\mathcal{G}'$ is such that:

1. for every joint strategy σ_C in $\mathcal{G}$, there exists a strategy σ'_C in $\mathcal{G}'$ such that for every state $s \in S$ we have that $\Pi(out(s'_\star, \sigma'_C)) = out(s, \sigma_C)$;

2. for every joint strategy σ'_C in $\mathcal{G}'$, there exists a strategy σ_C in $\mathcal{G}$ such that for all *canonical* states $s' \in S'$ we have that $\Pi(out(s', \sigma'_C)) = out(s'_{|\Phi}, \sigma_C)$.

Proof. We first prove (1). Given strategy σ_C in $\mathcal{G}$, for $i \in C$ define σ'_i as follows:

$$\sigma'_i(s') \quad = \quad \begin{cases} \{c_{ip} \mid p \in \sigma_i(s) \text{ and } s = s'_{|\Phi}\} & \text{if } s'(turn) = 0 \\ \emptyset & \text{otherwise} \end{cases}$$

Observe that if $s'(turn) = 1$ agents in C are obliged to play action $\emptyset$ by Definition 8.5, since it is their only available action. By combining all definitions above, we get that $\Pi(out(s'_\star, \sigma'_C)) = out(s, \sigma_C)$ for an arbitrary state $s \in S$.

To prove (2) we start from a strategy σ'_C in $\mathcal{G}'$. For any state $s \in S$, define $\sigma_i(s) = \{p \in \Phi_i \mid c_{ip} \in \sigma'_i(s'_\star)\}$. Note that the assumption in Definition 8.5 that all variables outside of Φ are put to false at stage $2k+1$ in $\mathcal{G}'$ is crucial here. In fact, without this assumption we would only be able to prove that $\Pi(out(s', \sigma'_C)) \supseteq out(s'_{|\Phi}, \sigma_C)$, as a strategy σ'_C may associate a different action to states s'_1 and s'_2 that coincide on Φ and that are realized in a path $\lambda' \in out(s', \sigma'_C)$. □

By means of Lemma 8.2 we are able to prove the following main result:

Theorem 8.1. Given any CGS-SPC $\mathcal{G}$, the corresponding CGS-EPC $\mathcal{G}'$ is such that for all state-formulas φ and path-formulas ψ in ATL^* the following holds:

$$\begin{array}{rlll} \text{for all } s \in S & (\mathcal{G}, s) \models \varphi & \text{iff} & (\mathcal{G}', s'_\star) \models tr(\varphi) \\ \text{for all } \lambda \text{ of } \mathcal{G} & (\mathcal{G}, \lambda) \models \psi & \text{iff} & (\mathcal{G}', \lambda'_\star) \models tr(\psi) \text{ for any } \lambda'_\star. \end{array}$$

Proof. The proof is by induction on the structure of formulas φ and ψ. The base case for $\varphi = p$ follows from the fact that $s = s'_{|\Phi}$ for all s' associated to s, including $s'_\star$.

As to the inductive cases for Boolean connectives, they follow immediately from the induction hypothesis.

Suppose that $\varphi = \langle\langle C\rangle\rangle\psi$. For the left-to-right direction, assume that $(\mathcal{G}, s) \models \varphi$. By the semantics, for some strategy σ_C, for all $\lambda \in out(s, \sigma_C)$, $(\mathcal{G}, \lambda) \models \psi$. By Lemma 8.2.1 we can find a strategy σ'_C in $\mathcal{G}'$ such that $\Pi(out(s'_\star, \sigma'_C)) = out(s, \sigma_C)$. By induction hypothesis, we know that for all $\lambda \in out(s, \sigma_C)$ we have that $(\mathcal{G}', \lambda'_\star) \models tr(\psi)$. These two facts combined imply that for all $\lambda' \in out(s'_\star, \sigma'_C)$ we have that $(\mathcal{G}', \lambda'_\star) \models tr(\psi)$, i.e., by the semantics, that $(\mathcal{G}', s'_\star) \models \langle\langle C\rangle\rangle tr(\psi)$, obtaining the desired result. The right-to-left direction can be proved similarly with Lemma 8.2(2).

Further, if φ is a state formula, $(\mathcal{G}, \lambda) \models \varphi$ if and only if $(\mathcal{G}, \lambda[0]) \models \varphi$, if and only if by induction hypothesis $(\mathcal{G}', \lambda[0]'_\star) \models tr(\varphi)$, i.e., $(\mathcal{G}', \lambda'_\star) \models tr(\varphi)$.

For $\psi = \bigcirc\psi_1$, suppose that $(\mathcal{G}, \lambda[1, \infty]) \models \psi_1$. By induction hypothesis, this is the case if and only if $(\mathcal{G}', (\lambda[1, \infty])'_\star) \models tr(\psi_1)$. Recall that by (†), we have that $(\lambda[1, \infty])'_\star = \lambda'_\star[2, \infty]$. This is the case because, when moving from λ to $\lambda'_\star$, we include an additional state $\lambda'_\star[1]$ in which the aggregation takes place. Therefore, $(\mathcal{G}', \lambda'_\star[2, \infty]) \models tr(\psi_1)$, that is, $(\mathcal{G}', \lambda'_\star) \models \bigcirc\bigcirc tr(\psi_1) = tr(\psi)$. The case for $\psi = \psi_1 \mathcal{U} \psi_2$ is proved similarly. □

As a consequence of Theorem 8.1, if we want to model-check an ATL^*-formula φ at a state s of an CGS-SPC $\mathcal{G}$, we can check its translation $tr(\varphi)$ at the related state $s'_\star$ of the associated CGS-EPC $\mathcal{G}'$. By observing that both the associated game $\mathcal{G}'$ and the translation φ are polynomial in the size of $\mathcal{G}$ and φ, we obtain that:

Corollary 8.1. The ATL^* model-checking problem for CGS-SPC can be reduced to the ATL^* model-checking problem for CGS-EPC.

8.4 Computational Complexity

In this section we prove complexity results for the model checking problem of an ATL^* (or ATL) formula φ on a CGS-SPC.

First, we define the model-checking problem for ATL over CGS-SPC:

MODELCHECK
Input: a CGS-SPC $\mathcal{G}$, a state $s \in S$, and an ATL formula φ
Question: Is it the case that $(\mathcal{G}, s) \models \varphi$?

We can then obtain the following result:

Theorem 8.2. Problem MODELCHECK (CGS-SPC and ATL formulas) is Δ_3^p-complete.

Proof. For membership, given a pointed CGS-SPC $(\mathcal{G}, s)$ and an ATL specification φ, by the translation tr introduced in Section 8.3 and Theorem 8.1 we have that $(\mathcal{G}, s) \models \varphi$ iff $(\mathcal{G}', s') \models tr(\varphi)$. Also, we observe that the CGS-EPC $\mathcal{G}'$ is of size polynomial in the size of $\mathcal{G}$, and that model checking ATL with respect to CGS-EPC is Δ_3^p-complete (Belardinelli and Herzig, 2016). For hardness, it is sufficient to observe that CGS-EPC are a subclass of CGS-SPC. □

For the verification of ATL^* we define the following problem:

MODELCHECK*
Input: a CGS-SPC $\mathcal{G}$, a state $s \in S$, and an ATL^* formula φ
Question: Is it the case that $(\mathcal{G}, s) \models \varphi$?

The model-checking problem for ATL^* on general concurrent game structures is 2EXPTIME-complete (Alur et al., 2002). We prove the following for CGS-SPC:

Theorem 8.3. Problem MODELCHECK* (CGS-SPC and ATL* formulas) is PSPACE-complete.

Proof. Membership follows from the PSPACE algorithm for ATL* on general CGS (Bulling et al., 2010). As for hardness, we observe that satisfiability of an LTL formula φ can be reduced to the model checking of the ATL* formula $\langle\langle 1 \rangle\rangle \varphi$ on a CGS-SPC with a unique agent 1. □

In Section 8.2 we showed how some examples of iterated games from the literature can be modelled as CGS-SPC, and how the problem of determining the existence of a winning strategy can therefore be reduced to model checking an ATL* formula. We now define and study the problem of checking existence of a winning strategy:

E-WIN(G, i)
Input: a game G, an agent i
Question: Does i have a memory-less winning strategy in G?

As an immediate consequences of Theorem 8.3 we obtain:

Corollary 8.2. E-WIN(G, i) is in PSPACE for G an iterated Boolean game with shared control.

An analogous result cannot be obtained for influence games and aggregation games directly. Decision problems in these structures are typically evaluated with respect to the number of agents and issues, and the size of the CGS-SPC associated to these games are already exponential in these parameters.

Corollary 8.3. E-WIN(G, i) is in PSPACE in the size of the associated CGS-SPC for G an influence game.

Corollary 8.3 is in line with previous results in the literature (Grandi et al., 2017).

8.5 Conclusions

In this chapter we introduced a class of concurrent game structures with shared propositional control (CGS-SPC), which we used to interpret popular logics for strategic reasoning ATL and ATL*. We have shown that CGS-SPC are general enough to capture iterated Boolean games and their generalization to shared control, as well as the influence games presented in Chapter 7 and aggregation games. Our main result shows that the model-checking problem for CGS-SPC and ATL* formulas can be reduced to the model-checking of ATL* formulas over standard CGS with exclusive control, allowing to establish some computational complexity results for the games mentioned above.

Even though our main result shows that the generalization to shared control structures can still be expressed by exclusive control structures, using CGS-SPC allows to model in a natural way complex interactions between agents on the assignment of truth values to propositional variables (as shown by the examples provided for Boolean games, influence games and aggregation games). Therefore, we have a way to model and reason about strategic situations of multi-agent decision-making as the ones at the core of this thesis.

An assumption on our CGS (both with exclusive and shared control) is that agents have perfect knowledge of the environment they are interacting with. Indeed, in our construction of Definition 8.5 the dummy agent $*$ is able to mimick the aggregation function τ as she can observe the values of c_{ip} for any other agent i. An interesting question is whether our reduction of CGS-SPC to CGS-EPC goes through even when imperfect information is assumed.

Chapter 9

Conclusions

At the incipit of this thesis we told the story of three researchers motivated by individual goals and preferences who were facing multiple situations of collective decision-making. In our quest to help them find such a collective choice, we delineated in Section 1.2 two main research questions, that we recall here:

1. How can we design **aggregation procedures** to help a group of agents having compactly expressed goals and preferences make a collective choice?
2. How can we model agents with conflicting goals who try to get a better outcome for themselves by **acting strategically**?

To conclude this thesis we provide a detailed summary of the main results that we achieved as an answer to these questions (Section 9.1), as well as some paths for future research that our contribution has opened (Section 9.2).

9.1 Summary of Contributions

The first three chapters were concerned with answering the first research question, on the aggregation of compactly represented goals and preferences coming from different agents. In Chapter 2 we presented two languages introduced in the literature, i.e., propositional logic and CP-nets (conditional preference networks), and provided a literature review for both. We also compared the preference orderings over outcomes generated by propositional goals (assuming dichotomous preferences) and CP-nets, observing that some orderings cannot be expressed by both languages.

In Chapter 3 we introduced the framework of goal-based voting, where agents express their goals as propositional formulas over binary issues, and they take a collective decision via a voting rule taking the goals as input and returning a set of interpretations as outcome. We then proceeded to an axiomatic study of our rules, by introducing properties for goal-based voting and then proving whether our rules satisfy them or not. Given our objective of helping agents find a collective decision, the first important axiom that we introduced was that of resoluteness of a rule (declined in a stronger and weaker notion). In this respect, we found the *Approval* rule to be inadequate, as it outputs a large set of possible plans whenever each of them is approved by a few agents only in the profile.

We then turned to issue-wise rules, warranted in particular by the absence of integrity constraints in our model, and we proposed three different generalizations of the well-known majority rule (i.e., *EMaj*, *TrueMaj* and *2sMaj*) all satisfying the weaker or stronger notion of resoluteness. Nevertheless, Theorem 3.1 exposed the fact that not all desirable properties can be achieved at the same time (specifically resoluteness, anonymity and duality) and compromises need to be taken when choosing a rule with respect to its properties. Our main result consisted in an axiomatic characterization of *TrueMaj*, obtained from independence, egalitarianism, neutrality, anonymity, monotonicity, unanimity and duality, thus striking a good balance among the (jointly achievable) properties of interest.

We then studied the computational complexity of determining the outcome for our rules, i.e., the WINDET problem. We found that having as input propositional formulas made the problem much harder than in voting. In particular, the three majorities *EMaj*, *TrueMaj* and *2sMaj* were found to be hard for the class PP (Probabilistic Polynomial Time). We however found some positive (tractable) results for the WINDET problem for our majorities by restricting the language of goals to conjunctions or disjunctions. This limitation is however still general enough to include binary aggregation with abstentions and Horn clauses. Finally, we compared the framework of goal-based voting with judgment aggregation and belief merging, with respect to properties (i.e., axioms or postulates) of aggregation rules. We found in particular that our notions of resoluteness conflicted with the (IC2) postulate of belief merging (Proposition 3.6), but this should not be seen too negatively as depending on the nature of the input (e.g., beliefs or goals) different sets of properties can be deemed more important by the user.

In Chapter 4 we merged two lines of research in CP-nets by investigating for the first time the aggregation of incomplete (generalized) CP-nets. Agents were thus given more flexibility in expressing their preferences (with respect to classical CP-nets), and they could find a collective decision via the *Pareto*, *maj*, *max* and *rank* semantics, which were known in the literature on complete CP-nets. We studied numerous computational problems related to dominance of outcomes in the preference ordering induced by the aggregation of incomplete CP-nets. For most problems we got hardness for the class PSPACE from the corresponding single-agent case, often leading to completeness results. The complexity can thus be seen as inherent to the studied problems and not given by the switch from one to multiple agents, two exceptions being the problem of checking if an outcome is non-dominated (jumping from P to PSPACE) and the problem of checking if it exists a non-dominated outcome (going from NP to PSPACE).

The second part of this thesis studied problems pertaining to our second research question, on the strategic behavior of goal-oriented agents. In Chapter 5 we recalled the definition of (iterated) Boolean games, a class of games where each agent controls exclusively a subset of the propositional variables and they hold logic-based goals. We provided the syntax and semantics of the logics LTL and ATL, interpreting them over Concurrent Game Structures (CGS). We then discussed the related work on game theory, logic and networks.

In Chapter 6 we considered again our framework of goal-based voting, assuming now that agents may submit untruthful goals if by doing so they can get a better result for themselves. We focused on the issue-by-issue majority rules *EMaj*, *TrueMaj* and *2sMaj*, which were found to be manipulable in the general case. This first negative result lead us to refine our study of manipulation by modulating different parameters of the problem: we considered agents with goals belonging to restrictions of the language (conjunctions, disjunctions, and exclusive disjunctions), agents with different satisfaction attitudes (optimistic, pessimistic, expected utility maximizer) and finally agents who were allowed to perform only certain manipulation strategies (unrestricted, erosion, and dilatation).

Regardless of the agent's satisfaction attitude, we found all our majoritarian rules to be strategy-proof for erosion and dilatation when the goals are conjunctions, and for dilatation (and erosion, for *2sMaj*) when the goals are disjunctions. Therefore, even though our majoritarian rules are in general subject to manipulation, some instances of strategy-proofness can be recovered by tuning two relevant parameters: the allowed manipulation strategy and the language of the agents' goals. From a computational perspective, our results indicate that checking if an agent can profitably manipulate is as hard as the WinDet problem for majoritarian rules. Hence, while we can consider it a positive result that a rule is PP-hard to manipulate (in the worst case), from a practical perspective we would ideally want a rule to be easy to compute and hard to manipulate — as it is the case, for instance, for the premise-based procedure in judgment aggregation (Endriss et al., 2012). This leads to a promising direction for future work: studying the Manip problem for goals restrictions for which WinDet is easy (e.g., *EMaj* and *TrueMaj* for disjunctions).

In Chapter 7 we defined influence games, a class of iterated games for modeling rational agents who decide whether to use their influence power in order to attain their LTL goals. Agents in influence games can thus act strategically by deciding to influence (or not) their neighbors in the influence network on the issues at stake, but they cannot lie about their opinions. We studied solution concepts for games where agents use the unanimous procedure to aggregate their influencers' opinions (to update their own). We found that an agent having a consensus goal has a winning strategy when the influence network is a DAG and each agent has at most one influencer. We also found a Nash equilibrium strategy profile for two agents having a consensus goal. Against our intuition, always using one's influence power is not a weakly dominant strategy to attain the influence goal. Therefore, the addition of a simple strategic action (i.e., using the influence power or not) to a model of opinion diffusion makes the dynamics of the spread of opinions much more difficult to capture and it is not straightforward to obtain general results.

In Chapter 8 we generalized the definition of a previously introduced CGS, where agents have exclusive control over a set of propositional variables, to a CGS where agents may have shared control over some variables. Our main result proved that verification of ATL^* formulas over CGS with shared control can be reduced to verification of ATL^* formulas over CGS with exclusive control. We then studied the computational complexity of model checking ATL^* formulas over CGS with shared control, which also allows us to express the problem of checking whether an agent has a winning strategy for one of the aforementioned iterated games: our results lie for the most part at the level of the PSPACE complexity class. Hence, while CGS-SPC offer a natural way to represent known iterated games in the literature where the control over variables is shared (including influence games), their verification problem of ATL^* formulas can still be reduced to structures with exclusive control.

In conclusion, with respect to the first research question we managed to provide two frameworks the agents can use to aggregate their compactly expressed goals and preferences (i.e., goal-based voting and mgCP-nets), whose rules satisfy a number of desirable properties, though the related computational problems are in general of high complexity and need to be restricted to become tractable. For the second research question, we found that (i) goal-based majoritarian rules are manipulable in general, but islands of strategy-proofness can be found under some restrictions of the setting, (ii) our strategic model of influence generates a complex opinions dynamics for which general results are hard to obtain, but fixing the type of graph and goal gives results for common solution concepts, and finally (iii) we understood the relationship between shared and exclusive control in concurrent game structures.

9.2 Perspectives and Future Work

In this thesis we achieved numerous results in the field of collective decision-making with compactly expressed goals and preferences. Nonetheless, there are extensions of this work that we believe would be worth investigating in the future, many of which we already mentioned in the respective concluding sections of each chapter. We list here some additional ideas.

In the first place, we could think about designing a **more general framework** which is able to model all the different problems that we studied in this thesis. For instance, the three researchers in our initial story can use propositional goals to decide about the conference, and generalized CP-nets to decide about the syllabus, but the aggregation procedures that we defined do not allow them to provide as input a mix of propositional goals and incomplete CP-nets. This research direction would go towards the work by Bienvenu et al. (2010) on the "prototypical" preference logic *PL* that we mentioned in Section 2.5, giving agents even more flexibility in how they express their goals and preferences.

We could also extend the aggregation frameworks of goal-based voting and *mg*CP-nets by considering that decisions may be recurring (e.g., the syllabus for a course that is taught every year) or that agents may want to change their vote after they have seen the current outcome or they have discussed it with their peers. Analogously to what has been done for classical voting (see the recent chapter by Meir, 2017), we could study the **iterative** versions of goal-based voting and aggregation of incomplete CP-nets. This would be in line with the recent work by Terzopoulou and Endriss (2018) who studied an iterative model for judgment aggregation.

Additionally, we could study in more depth possible **restrictions** over our frameworks. We have seen that certain classes of propositional goals for some rules give us tractable results for the winner determination problem or strategy-proofness results. We could then look for a precise characterization of the restrictions over the language which give us bounds for WINDET tractability and strategy-proofness. Analogously, we could look for restrictions on the precondition of incomplete CP-nets to make dominance tasks easier to compute than the PSPACE class.

In order to prove that our issue-wise majority rules are manipulable in general, we provided in the proof of Theorem 6.1 specific profiles for which an agent has an incentive to manipulate. Since for this kind of results a single counter-example suffices, it may paint an overly negative picture of the manipulation landscape — as such manipulable profiles may be rarely occurring in practice. Another direction for future work would be then to **implement** voting situations under majoritarian rules to get some statistics on which percentage of profiles are actually manipulable. Observe, however, that even checking all possible profiles for just three agents and issues would give us a total of $(2^{2^3} - 1) \times (2^{2^3} - 1) \times (2^{2^3} - 1) = 255 \times 255 \times 255 =$ 16581375 profiles to consider. For each of them we would need to check not only that there is some agent whose goal is not satisfied in the outcome, but also that they can *actually* manipulate the current result by submitting a different goal — accounting also for the different types of satisfaction for the rule *TrueMaj*.

A topic that has received increasing attention in recent years is that of **explanation** in artificial intelligence, i.e., the ability to explain to the (human) users the behavior and output of algorithms (see, e.g., the paper by Miller, 2019). In the area of computational social choice in particular, Cailloux and Endriss (2016) proposed a framework aimed at helping humans choose the "best" voting rule by modeling arguments for or against them based on the axioms they satisfy. We could pursue a similar approach for goal-based voting: while the rules that we use are not *per se* difficult to understand (e.g., the three generalizations of majority), they satisfy different axioms that are mutually incompatible and thus a choice needs to be taken over them by the user, depending on their priorities.

Finally, we could bring together our goal-based voting (or mgCP-nets) framework and the idea of agents connected by a trust network in influence games, into a framework where the dynamics that is modeled is that of **delegation** instead of diffusion. For instance, in goal-based voting we could imagine that an agent who is not able to form a goal over the issues could instead decide to "+1" the goal of another agent she trusts. In this research direction we can mention the Blue Sky paper by Brill (2018) on interactive democracy and vote delegation where some ideas and potential issues on the topic are presented.

Index

Bibliography

Alur, R., Henzinger, T. A., and Kupferman, O. (2002). Alternating-Time Temporal Logic. *Journal of the ACM*, 49(5):672—713. [Cited on pages 64 and 97]

Arora, S. and Barak, B. (2009). *Computational Complexity: A Modern Approach*. Cambridge University Press. [Cited on pages 30 and 53]

Arrow, K. J. (1951). *Social Choice and Individual Values*. Yale University Press. [Cited on pages 3 and 22]

Awad, E., Booth, R., Tohmé, F., and Rahwan, I. (2017). Judgement Aggregation in Multi-Agent Argumentation. *Journal of Logic and Computation*, 27(1):227—259. [Cited on page 17]

Aziz, H., Gaspers, S., Gudmundsson, J., Mackenzie, S., Mattei, N., and Walsh, T. (2015). Computational Aspects of Multi-Winner Approval Voting. In *Proceedings of the 2015 International Conference on Autonomous Agents and Multiagent Systems (AAMAS-2015)*. [Cited on page 19]

Baumeister, D., Erdélyi, G., Erdélyi, O. J., and Rothe, J. (2015). Complexity of Manipulation and Bribery in Judgment Aggregation for Uniform Premise-Based Quota Rules. *Mathematical Social Sciences*, 76:19—30. [Cited on page 30]

Belardinelli, F. and Herzig, A. (2016). On Logics of Strategic Ability Based on Propositional Control. In *Proceedings of the 25th International Joint Conference on Artificial Intelligence (IJCAI-2016)*. [Cited on pages 66, 67, 89, 90, and 97]

Benamara, F., Kaci, S., and Pigozzi, G. (2010). Individual Opinions-Based Judgment Aggregation Procedures. In *Proceedings of the 7th International Conference on Modeling Decisions for Artificial Intelligence (MDAI-2010)*. [Cited on page 13]

Bhagat, S. and Brickley, J. A. (1984). Cumulative Voting: The Value of Minority Shareholder Voting Rights. *The Journal of Law and Economics*, 27(2):339—365. [Cited on page 19]

Bienvenu, M., Lang, J., and Wilson, N. (2010). From Preference Logics to Preference Languages, and Back. In *Proceedings of the 12th International Conference on the Principles of Knowledge Representation and Reasoning (KR-2010)*. [Cited on pages 16 and 102]

Blackburn, P., van Benthem, J. F., and Wolter, F. (2006). *Handbook of Modal Logic*, volume 3. Elsevier. [Cited on page 2]

Bonzon, E., Lagasquie-Schiex, M.-C., Lang, J., and Zanuttini, B. (2006). Boolean Games Revisited. In *Proceedings of the 17th European Conference on Artificial Intelligence (ECAI-2006)*. [Cited on pages 13 and 63]

Botan, S., Grandi, U., and Perrussel, L. (2019). Multi-Issue Opinion Diffusion under Constraints. In *Proceedings of the 18th International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS-2019)*. [Cited on page 66]

Boutilier, C. (1994). Toward a Logic for Qualitative Decision Theory. In *Proceedings of the 4th International Conference on Principles of Knowledge Representation and Reasoning (KR-1994)*. [Cited on page 2]

Boutilier, C., Brafman, R. I., Domshlak, C., Hoos, H. H., and Poole, D. (2004). CP-nets: A Tool for Representing and Reasoning with Conditional Ceteris Paribus Preference Statements. *Journal of Artificial Intelligence Research*, 21:135—191. [Cited on pages 13, 14, and 49]

Bouveret, S., Endriss, U., and Lang, J. (2009). Conditional Importance Networks: A Graphical Language for Representing Ordinal, Monotonic Preferences over Sets of Goods. In *Proceedings of the 21st International Joint Conference on Artificial Intelligence (IJCAI-2009)*. [Cited on page 15]

Brafman, R. I. and Domshlak, C. (2002). Introducing Variable Importance Tradeoffs into CP-Nets. In *Proceedings of the 18th Conference in Uncertainty in Artificial Intelligence (UAI-2002)*. [Cited on page 15]

Brafman, R. I., Domshlak, C., and Shimony, S. E. (2006). On Graphical Modeling of Preference and Importance. *Journal of Artificial Intelligence Research*, 25:389—424. [Cited on page 15]

Brafman, R. I. and Tennenholtz, M. (1996). On the Foundations of Qualitative Decision Theory. In *Proceedings of the 13th AAAI Conference on Artificial Intelligence (AAAI-1996)*. [Cited on page 2]

Brams, S. and Fishburn, P. C. (1978). Approval Voting. *American Political Science Review*, 72(3):831—847. [Cited on page 18]

Brams, S. and Fishburn, P. C. (2007). *Approval Voting*. Springer Science & Business Media. [Cited on page 18]

Brandt, F., Conitzer, V., Endriss, U., Procaccia, A. D., and Lang, J. (2016). *Handbook of Computational Social Choice*. Cambridge University Press. [Cited on pages 3, 17, and 28]

Bratman, M. (1987). *Intention, Plans, and Practical Reason*, volume 10. Harvard University Press. [Cited on page 2]

Brewka, G. (2004). A Rank Based Description Language for Qualitative Preferences. In *Proceedings of the 16th European Conference on Artificial Intelligence (ECAI-2004)*. [Cited on page 12]

Brill, M. (2018). Interactive Democracy. In *Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems (AAMAS-2018) / Blue Sky Track*. [Cited on page 103]

Brill, M., Elkind, E., Endriss, U., and Grandi, U. (2016). Pairwise Diffusion of Preference Rankings in Social Networks. In *Proceedings of the 25th International Joint Conference on Artificial Intelligence (IJCAI-2016)*. [Cited on page 66]

Bulling, N., Dix, J., and Jamroga, W. (2010). Model Checking Logics of Strategic Ability: Complexity*. In Dastani, M., Hindriks, K. V., and Meyer, J.-J. C., editors, *Specification and Verification of Multi-agent Systems*, pages 125—159. Springer US. [Cited on page 98]

Bylander, T. (1994). The Computational Complexity of Propositional STRIPS Planning. *Artificial Intelligence*, 69(1-2):165—204. [Cited on page 2]

Cailloux, O. and Endriss, U. (2016). Arguing about Voting Rules. In *Proceedings of the 15th International Conference on Autonomous Agents and Multiagent Systems (AAMAS-2016)*. [Cited on page 102]

Campbell, W. (1954). The Origin and Growth of Cumulative Voting for Directors. *The Business Lawyer*, 10(3):3—16. [Cited on page 19]

Chen, Z.-Z. and Toda, S. (1995). The Complexity of Selecting Maximal Solutions. *Information and Computation*, 119(2):231—239. [Cited on page 32]

Chevaleyre, Y., Endriss, U., Lang, J., and Maudet, N. (2008). Preference Handling in Combinatorial Domains: From AI to Social Choice. *AI Magazine, Special Issue on Preferences*, 29(4):37—46. [Cited on page 11]

Ciaccia, P. (2007). Conditional Preferences: A New Semantics for Database Queries. In *Proceedings of the 15th Italian Symposium on Advanced Database Systems (SEBD-2007)*. [Cited on page 15]

Cohen, P. R. and Levesque, H. J. (1990). Intention is Choice with Commitment. *Artificial Intelligence*, 42(2-3):213—261. [Cited on page 2]

Conitzer, V. (2019). Designing Preferences, Beliefs, and Identities for Artificial Intelligence. In *Proceedings of the 33rd AAAI Conference on Artificial Intelligence (AAAI-19) Senior Member / Blue Sky Track*. [Cited on page 3]

Conitzer, V., Sandholm, T., and Lang, J. (2007). When are Elections with Few Candidates Hard to Manipulate? *Journal of the ACM*, 54(3):14. [Cited on page 3]

Conitzer, V. and Walsh, T. (2016). Barriers to Manipulation in Voting. In Brandt, F., Conitzer, V., Endriss, U., Lang, J., and Procaccia, A. D., editors, *Handbook of Computational Social Choice*, chapter 6. Cambridge University Press. [Cited on pages 3, 4, and 69]

Cornelio, C., Goldsmith, J., Mattei, N., Rossi, F., and Venable, K. B. (2013). Updates and Uncertainty in CP-nets. In *Proceedings of the 26th Australasian Joint Conference on Artificial Intelligence (AI-2013)*. [Cited on page 46]

Cornelio, C., Grandi, U., Goldsmith, J., Mattei, N., Rossi, F., and Venable, K. B. (2015). Reasoning with PCP-nets in a Multi-Agent Context. In *Proceedings of the 14th International Conference on Autonomous Agents and Multiagent Systems (AAMAS-2015)*. [Cited on page 46]

Costantini, M., Groenland, C., and Endriss, U. (2016). Judgment Aggregation under Issue Dependencies. In *Proceedings of the 30th AAAI Conference on Artificial Intelligence (AAAI-2016)*. [Cited on page 37]

Dastani, M. and van der Torre, L. (2002). Specifying the Merging of Desires into Goals in the Context of Beliefs. In *Proceedings of the 1st EurAsian Conference on Information and Communication Technology (EurAsiaICT-2002)*. [Cited on pages 2 and 13]

Dietrich, F. and List, C. (2007a). Judgment Aggregation by Quota Rules: Majority Voting Generalized. *Journal of Theoretical Politics*, 19(4):391—424. [Cited on pages 19 and 69]

Dietrich, F. and List, C. (2007b). Judgment Aggregation with Consistency Alone. *Department of Government, London School of Economics and Political Science*. [Cited on page 40]

Dietrich, F. and List, C. (2007c). Strategy-Proof Judgment Aggregation. *Economics & Philosophy*, 23(3):269—300. [Cited on pages 69, 70, and 71]

Dietrich, F. and List, C. (2008). Judgment Aggregation without Full Rationality. *Social Choice and Welfare*, 31(1):15—39. [Cited on pages 13 and 72]

Dokow, E. and Holzman, R. (2010a). Aggregation of Binary Evaluations. *Journal of Economic Theory*, 145(2):495—511. [Cited on page 13]

Dokow, E. and Holzman, R. (2010b). Aggregation of Binary Evaluations with Abstentions. *Journal of Economic Theory*, 145(2):544—561. [Cited on pages 13, 40, and 72]

Dubois, D., Prade, H., and Sabbadin, R. (2001). Decision-Theoretic Foundations of Qualitative Possibility Theory. *European Journal of Operational Research*, 128(3):459—478. [Cited on page 2]

Elkind, E., Faliszewski, P., and Slinko, A. (2010). On the Role of Distances in Defining Voting Rules. In *Proceedings of the 9th International Conference on Autonomous Agents and Multiagent Systems (AAMAS-2010)*. [Cited on pages 25 and 26]

Endriss, U. (2016). Judgment Aggregation. In Brandt, F., Conitzer, V., Endriss, U., Lang, J., and Procaccia, A. D., editors, *Handbook of Computational Social Choice*, chapter 17, pages 399—426. Cambridge University Press. [Cited on pages 3, 13, 24, 28, 40, and 59]

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)*. [Cited on page 13]

Endriss, U. and Grandi, U. (2014). Binary Aggregation by Selection of the Most Representative Voter. In *Proceedings of the 28th AAAI Conference on Artificial Intelligence (AAAI-2014)*. [Cited on pages 21 and 40]

Endriss, U. and Grandi, U. (2017). Graph Aggregation. *Artificial Intelligence*, 245:86—114. [Cited on page 24]

Endriss, U., Grandi, U., de Haan, R., and 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)*. [Cited on page 38]

Endriss, U., Grandi, U., and Porello, D. (2012). Complexity of Judgment Aggregation. *Journal of Artificial Intelligence Research*, 45:481—514. [Cited on pages 30, 70, 75, and 101]

Everaere, P., Konieczny, S., and Marquis, P. (2007). The Strategy-Proofness Landscape of Merging. *Journal of Artificial Intelligence Research*, 28:49—105. [Cited on pages 69, 70, and 71]

Everaere, P., Konieczny, S., and Marquis, P. (2015). Belief Merging versus Judgment Aggregation. In *Proceedings of the 2015 International Conference on Autonomous Agents and Multiagent Systems (AAMAS-2015)*. [Cited on page 25]

Everaere, P., Konieczny, S., and Marquis, P. (2017). An Introduction to Belief Merging and its Links with Judgment Aggregation. In Endriss, U., editor, *Trends in Computational Social Choice*, chapter 7, pages 123—143. AI Access. [Cited on pages 36 and 43]

Fikes, R. E. and Nilsson, N. J. (1971). STRIPS: A New Approach to the Application of Theorem Proving to Problem Solving. *Artificial Intelligence*, 2(3-4):189—208. [Cited on page 2]

Gärdenfors, P. (2006). A Representation Theorem for Voting with Logical Consequences. *Economics & Philosophy*, 22(2):181—190. [Cited on pages 13 and 72]

Gerbrandy, J. (2006). Logics of Propositional Control. In *Proceedings of the 5th International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS-2006)*. [Cited on page 67]

Gibbard, A. (1973). Manipulation of Voting Schemes: A General Result. *Econometrica*, pages 587—601. [Cited on pages 3 and 69]

Gibbons, R. (1992). *A Primer in Game Theory*. Harvester Wheatsheaf New York. [Cited on pages 3 and 83]

Gill, J. (1977). Computational Complexity of Probabilistic Turing Machines. *SIAM Journal on Computing*, 6(4):675—695. [Cited on pages 33 and 34]

Goldsmith, J., Lang, J., Truszczynski, M., and Wilson, N. (2008). The Computational Complexity of Dominance and Consistency in CP-Nets. *Journal of Artificial Intelligence Research*, 33:403—432. [Cited on pages 14, 45, 49, 53, and 59]

Goranko, V. and Jamroga, W. (2004). Comparing Semantics of Logics for Multi-Agent Systems. In *Information, Interaction and Agency*, pages 77—116. Springer. [Cited on page 2]

Grandi, U. and Endriss, U. (2011). Binary Aggregation with Integrity Constraints. In *Proceedings of the 22nd International Joint Conference on Artificial Intelligence (IJCAI-2011)*. [Cited on pages 13, 23, 24, 25, 26, 38, 40, and 42]

Grandi, U. and Endriss, U. (2013). Lifting integrity constraints in binary aggregation. *Artificial Intelligence*, 199–200:45—66. [Cited on page 21]

Grandi, U., Grossi, D., and Turrini, P. (2019). Negotiable Votes. *Journal of Artificial Intelligence Research*, 64:895—929. [Cited on pages 66, 91, and 92]

Grandi, U., Lorini, E., Novaro, A., and Perrussel, L. (2017). Strategic Disclosure of Opinions on a Social Network. In *Proceedings of the 16th International Conference on Autonomous Agents and Multiagent Systems (AAMAS-2017)*. [Cited on pages 87, 91, and 98]

Grandi, U., Lorini, E., and Perrussel, L. (2015). Propositional Opinion Diffusion. In *Proceedings of the 14th International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS-2015)*. [Cited on pages 66, 79, and 87]

Grandi, U., Luo, H., Maudet, N., and Rossi, F. (2014). Aggregating CP-nets with Unfeasible Outcomes. In *Proceedings of the 20th International Conference on Principles and Practice of Constraint Programming (CP-2014)*. [Cited on page 45]

Grossi, D. and Pigozzi, G. (2014). *Judgment Aggregation: A Primer*. Morgan and Claypool. [Cited on page 3]

Gutierrez, J., Harrenstein, P., and Wooldridge, M. (2013). Iterated Boolean Games. In *Proceedings of the 23th International Joint Conference on Artificial Intelligence (IJCAI-2013)*. [Cited on pages 63 and 89]

Gutierrez, J., Harrenstein, P., and Wooldridge, M. (2015). Iterated Boolean Games. *Information and Computation*, 242:53—79. [Cited on pages 63, 89, and 91]

de Haan, R. and Slavkovik, M. (2017). Complexity Results for Aggregating Judgments using Scoring or Distance-Based Procedures. In *Proceedings of the 16th Conference on Autonomous Agents and MultiAgent Systems (AAMAS-2017)*. [Cited on page 30]

Harrenstein, P., van der Hoek, W., Meyer, J.-J., and Witteveen, C. (2001). Boolean Games. In *Proceedings of the 8th Conference on Theoretical Aspects of Rationality and Knowledge (TARK-2001)*. [Cited on pages 13 and 63]

Hemaspaandra, E., Spakowski, H., and Vogel, J. (2005). The Complexity of Kemeny Elections. *Theoretical Computer Science*, 349(3):382—391. [Cited on page 3]

Herzig, A. (2015). Logics of Knowledge and Action: Critical Analysis and Challenges. *Autonomous Agents and Multi-Agent Systems*, 29(5):719—753. [Cited on page 2]

van der Hoek, W. and Wooldridge, M. (2005). On the Logic of Cooperation and Propositional Control. *Artificial Intelligence*, 164(1-2):81—119. [Cited on page 66]

Isbell, J. R. (1959). On the Enumeration of Majority Games. *Mathematical Tables and Other Aids to Computation*, 13(65):21—28. [Cited on page 67]

Jimeno, J. L., Pérez, J., and García, E. (2009). An Extension of the Moulin No Show Paradox for Voting Correspondences. *Social Choice and Welfare*, 33(3):343—359. [Cited on page 70]

Kautz, H., McAllester, D., and Selman, B. (1996). Encoding Plans in Propositional Logic. [Cited on page 2]

Kautz, H. and Selman, B. (1996). Pushing the Envelope: Planning, Propositional Logic, and Stochastic Search. In *Proceedings of the 13th National Conference on Artificial Intelligence (AAAI-1996)*. [Cited on page 2]

Kautz, H. A. and Selman, B. (1992). Planning as Satisfiability. In *Proceedings of the10th European Conference on Artificial Intelligence (ECAI-1992)*. [Cited on page 2]

Klamler, C. and Pferschy, U. (2007). The Traveling Group Problem. *Social Choice and Welfare*, 29(3):429—452. [Cited on page 17]

Konieczny, S., Lang, J., and Marquis, P. (2002). Distance Based Merging: A General Framework and Some Complexity Results. In *Proceedings of the 8th International Conference on Principles and Knowledge Representation and Reasoning (KR-2002)*. [Cited on page 32]

Konieczny, S. and Pérez, R. P. (2002). Merging Information Under Constraints: A Logical Framework. *Journal of Logic and Computation*, 12(5):773—808. [Cited on pages 13, 19, and 36]

Lang, J. (2004). Logical Preference Representation and Combinatorial Vote. *Annals of Mathematics and Artificial Intelligence*, 42(1-3):37—71. [Cited on pages 12, 13, 19, and 32]

Lang, J. and Slavkovik, M. (2013). Judgment Aggregation Rules and Voting Rules. In *Proceedings of the 3rd International Conference on Algorithmic Decision Theory (ADT-2013)*. [Cited on page 39]

Lang, J. and 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)*. [Cited on pages 13 and 31]

Lang, J. and Xia, L. (2016). Voting in Combinatorial Domains. In Brandt, F., Conitzer, V., Endriss, U., Lang, J., and Procaccia, A., editors, *Handbook of Computational Social Choice*, chapter 9, pages 197—222. Cambridge University Press. [Cited on page 11]

Laslier, J.-F. and Sanver, R. M. (2010). *Handbook on Approval Voting*. Springer Science & Business Media. [Cited on page 18]

Leyton-Brown, K. and Shoham, Y. (2008). Essentials of Game Theory: A Concise Multidisciplinary Introduction. *Synthesis Lectures on Artificial Intelligence and Machine Learning*, 2(1):1—88. [Cited on page 3]

Li, M., Vo, Q. B., and Kowalczyk, R. (2014). A Distributed Social Choice Protocol for Combinatorial Domains. *Journal of Heuristics*, 20(4):453—481. [Cited on page 46]

Li, M., Vo, Q. B., and Kowalczyk, R. (2015). Aggregating Multi-Valued CP-nets: a CSP-Based Approach. *Journal of Heuristics*, 21(1):107—140. [Cited on page 46]

List, C. (2012). The Theory of Judgment Aggregation: An Introductory Review. *Synthese*, 187(1):179—207. [Cited on pages 13, 26, 28, 36, 38, and 41]

List, C. and Pettit, P. (2002). Aggregating Sets of Judgments: An Impossibility Result. *Economics & Philosophy*, 18(1):89—110. [Cited on page 41]

List, C. and Pettit, P. (2004). Aggregating Sets of Judgments: Two Impossibility Results Compared. *Synthese*, 140(1-2):207—235. [Cited on page 41]

Lorini, E. (2017). Logics for Games, Emotions and Institutions. *IfCoLog Journal of Logics and their Applications*, 4(9):3075—3113. [Cited on page 2]

Lukasiewicz, T. and Malizia, E. (2016). On the Complexity of mCP-nets. In *Proceedings of the 30th AAAI Conference on Artificial Intelligence (AAAI-2016)*. [Cited on pages 45, 51, and 59]

Maran, A., Maudet, N., Pini, M. S., Rossi, F., and Venable, K. B. (2013). A Framework for Aggregating Influenced CP-nets and its Resistance to Bribery. In *Proceedings of the 27th AAAI Conference on Artificial Intelligence (AAAI-2013)*. [Cited on page 46]

May, K. O. (1952). A set of Independent Necessary and Sufficient Conditions for Simple Majority Decision. *Econometrica: Journal of the Econometric Society*, pages 680—684. [Cited on pages 3, 19, 23, 28, and 69]

Meir, R. (2017). Iterative Voting. In Endriss, U., editor, *Trends in Computational Social Choice*, chapter 4, pages 69—86. AI Access. [Cited on page 102]

Miller, M. K. (2008). Judgment Aggregation and Subjective Decision-Making. *Economics & Philosophy*, 24(2):205—231. [Cited on page 13]

Miller, T. (2019). Explanation in Artificial Intelligence: Insights from the Social Sciences. *Artificial Intelligence*, 267:1—38. [Cited on page 102]

Moulin, H. (1983). *The Strategy of Social Choice*. North-Holland. [Cited on page 23]

Novaro, A., Grandi, U., Longin, D., and Lorini, E. (2018). Goal-Based Collective Decisions: Axiomatics and Computational Complexity. In *Proceedings of the 27th International Joint Conference on Artificial Intelligence (IJCAI-2018)*. [Cited on pages 19, 25, 27, and 28]

Pnueli, A. (1977). The Temporal Logic of Programs. In *Proceedings of the 18th IEEE Symposium on Foundations of Computer Science*. [Cited on page 64]

Porello, D. and Endriss, U. (2014). Ontology Merging as Social Choice: Judgment Aggregation under the Open World Assumption. *Journal of Logic and Computation*, 24(6):1229—1249. [Cited on page 26]

Qing, C., Endriss, U., Fernández, R., and Kruger, J. (2014). Empirical Analysis of Aggregation Methods for Collective Annotation. In *Proceedings of the 25th International Conference on Computational Linguistics (COLING-2014)*. [Cited on page 17]

Rao, A. S. and Georgeff, M. P. (1991). Modeling Rational Agents within a BDI-Architecture. In *Proceedings of the 2nd International Conference on Principles of Knowledge Representation and Reasoning (KR-1991)*. [Cited on page 2]

Rao, A. S. and Georgeff, M. P. (1995). BDI Agents: From Theory to Practice. In *Proceedings of the 1st International Conference on Multiagent Systems (ICMAS-1995)*. [Cited on page 2]

Rossi, F., Venable, K. B., and Walsh, T. (2004). mCP Nets: Representing and Reasoning with Preferences of Multiple Agents. In *Proceedings of the 19th National Conference on Artificial Intelligence, 16th Conference on Innovative Applications of Artificial Intelligence (AAAI-2004)*. [Cited on pages 45, 51, and 59]

Russell, S. J. and Norvig, P. (2016). *Artificial Intelligence: A Modern Approach*. Pearson, 3rd edition. [Cited on page 2]

Satterthwaite, M. A. (1975). Strategy-Proofness and Arrow's Conditions: Existence and Correspondence Theorems for Voting Procedures and Social Welfare Functions. *Journal of Economic Theory*, 10(2):187—217. [Cited on pages 3 and 69]

Schwind, N., Inoue, K., Bourgne, G., Konieczny, S., and Marquis, P. (2015). Belief Revision Games. In *Proceedings of the 29th AAAI Conference on Artificial Intelligence (AAAI-2015)*. [Cited on page 66]

Schwind, N., Inoue, K., Bourgne, G., Konieczny, S., and Marquis, P. (2016). Is Promoting Beliefs Useful to Make Them Accepted in Networks of Agents? In *Proceedings of the 25th International Joint Conference on Artificial Intelligence (IJCAI-2016)*. [Cited on page 66]

Shoham, Y. and Leyton-Brown, K. (2008). *Multiagent Systems: Algorithmic, Game-Theoretic, and Logical Foundations*. Cambridge University Press. [Cited on pages 2 and 64]

Terzopoulou, Z. and Endriss, U. (2018). Modelling Iterative Judgment Aggregation. In *Proceedings of the 32nd AAAI Conference on Artificial Intelligence (AAAI-2018)*. [Cited on page 102]

Terzopoulou, Z., Endriss, U., and de Haan, R. (2018). Aggregating Incomplete Judgments: Axiomatisations for Scoring Rules. In *Proceedings of the 7th International Workshop on Computational Social Choice (COMSOC-2018)*. [Cited on pages 13, 40, and 72]

Uckelman, J., Chevaleyre, Y., Endriss, U., and Lang, J. (2009). Representing Utility Functions via Weighted Goals. *Mathematical Logic Quarterly*, 55(4):341—361. [Cited on page 12]

Van Dalen, D. (2004). *Logic and Structure*. Springer, 5th edition. [Cited on page 2]

Weiss, G. (2013). *Multiagent Systems*. MIT Press, 2nd edition. [Cited on page 2]

Wellman, M. P. and Doyle, J. (1991). Preferential Semantics for Goals. In *Proceedings of the 9th National Conference on Artificial Intelligence (AAAI-1991)*. [Cited on page 2]

Wilson, N. (2004). Extending CP-Nets with Stronger Conditional Preference Statements. In *Proceedings of the 19th National Conference on Artificial Intelligence, 16th Conference on Innovative Applications of Artificial Intelligence (AAAI-2004)*. [Cited on pages 15 and 49]

Wooldridge, M. (2009). *An Introduction to MultiAgent Systems*. John Wiley & Sons, 2nd edition. [Cited on page 2]

Wooldridge, M. (2013). Intelligent Agents. In Weiss, G., editor, *Multiagent Systems*, chapter 1, pages 3—50. MIT Press, 2nd edition. [Cited on page 2]

Young, H. (1974). An Axiomatization of Borda's Rule. *Journal of Economic Theory*, 9(1):43—52. [Cited on page 37]

Zwicker, W. S. (2016). Introduction to the Theory of Voting. In Brandt, F., Conitzer, V., Endriss, U., Lang, J., and Procaccia, A., editors, *Handbook of Computational Social Choice*, chapter 2, pages 23—56. Cambridge University Press. [Cited on page 24]