Complexity in Interaction

Lena Kurzen

Complexity in Interaction

ILLC Dissertation Series DS–2011–10

Institute for Logic, Language and Computation

For further information about ILLC-publications, please contact

Institute for Logic, Language and Computation
Universiteit van Amsterdam
Science Park 904
1098 XH Amsterdam
phone: +31-20-525 6051
fax: +31-20-525 5206
e-mail: `illc@uva.nl`
homepage: `http://www.illc.uva.nl/`

Complexity in Interaction

Academisch Proefschrift

ter verkrijging van de graad van doctor aan de
Universiteit van Amsterdam
op gezag van de Rector Magnificus
prof.dr. D.C. van den Boom
ten overstaan van een door het college voor
promoties ingestelde commissie, in het openbaar
te verdedigen in de Aula der Universiteit
op vrijdag 18 november 2011, te 13.00 uur

door

Lena Maria Kurzen

geboren te Mettingen, Duitsland.

Promotiecommissie

Promotor:
Prof.dr. J.F.A.K. van Benthem

Co-promotor:
Prof.dr. P. van Emde Boas

Overige leden:
Prof.dr. P.W. Adriaans
Prof.dr. K.R. Apt
Dr. A. Baltag
Prof.dr. W. van der Hoek
Dr. C. Löding
Dr. M.E.J. Raijmakers
Dr.ir. J.W.H.M. Uiterwijk

Faculteit der Natuurwetenschappen, Wiskunde en Informatica
Universiteit van Amsterdam
Science Park 904
1098 XH Amsterdam

The investigations were supported by the Dutch Research Organization for Scientific Research (NWO), TACTICS project, grant number 612.000.525.

Printed and bound by Ponsen en Looijen B.V., Wageningen.

ISBN: 978–90–5776–232–1

Contents

Acknowledgments **vii**

1 Introduction **1**
1.1 Modal logics for interaction – the basics 2
1.2 The basics of computational complexity 10
1.3 Research questions 14
1.4 Outline 20

I Complexity of Reasoning about Strategic Ability using Modal Logic **25**

2 Reasoning about Cooperation, Actions and Preferences **27**
2.1 Cooperation, Actions and Preferences 28
2.2 Cooperation Logic with Actions (CLA) 29
2.3 Cooperation Logic with Actions and Preferences 33
2.4 Complexity of CLA+P 40
2.5 Conclusions and Further Questions 47

3 Complexity of Modal Logics for Cooperation **51**
3.1 Reasoning about efficiency and stability in modal logics 52
3.2 Three ways of modeling cooperation 53
3.3 Comparing modal logics for cooperation 67
3.4 Conclusions and Further Questions 85

II Complexity in the Interaction between Diverse Agents **93**

4 Variations on Sabotage – Obstruction and Cooperation **95**
4.1 Sabotage Games 96
4.2 From Obstruction to Cooperation 105
4.3 Allowing breaks 121

4.4 Conclusions and Further Questions 125

5 Exploring the Tractability Border in Epistemic Tasks 131

5.1 From the Complexity of Epistemic Logics to the Complexity of Epistemic Tasks 132

5.2 Complexity of comparing and manipulating information 135

5.3 Conclusions and Further Questions 154

6 The Complexity of Playing Eleusis 165

6.1 Eleusis: an inductive inference game 166

6.2 Eleusis: The rules 167

6.3 Complexities in Eleusis 171

6.4 Conclusions and Further Questions 185

7 Conclusion 189

7.1 Summary of the chapters 189

7.2 Conclusion 193

7.3 Further Work 196

Bibliography 199

Index 211

Samenvatting 217

Abstract 219

Acknowledgments

Writing this thesis would not have been possible, let alone enjoyable, for me without the help and support of many people. In this section, I would like to express my gratitude to them.

First and foremost, I would like to thank my supervisors Johan van Benthem and Peter van Emde Boas for all their support throughout the last four years. I could benefit a lot from their experience in supervising PhD students and making sure that somehow things will work out in the end.

I am very grateful to Johan for sharing his never-ending enthusiasm about logic and (un)related topics with me and for always directing me to interesting topics and people to work with. Even while constantly traveling to other continents, he managed to still seem to always have time for discussions and for giving me useful feedback on my work. I am especially grateful to him for even supporting me in exploring some topics only loosely related to the topic of my PhD project.

I would like to thank Peter for all his help and practical advise. Irrespectively of the topics that I had some questions about, he would magically retrieve highly relevant articles from some decades ago (that I would have never found otherwise) from the depth of his personal library. I am very impressed by his ability to find flaws at high speed, even when hidden in small details. Sometimes, when I had given him a draft of my work, almost at the same moment he would return a list of mistakes and clarifying questions.

Thanks also to Johan and Peter for their joint efforts in turning my attempt at writing a 'samenvatting' into proper Dutch.

I am extremely grateful to the members of my thesis committee, Pieter Adriaans, Krzysztof Apt, Alexandru Baltag, Wiebe van der Hoek, Christof Löding, Maartje Raijmakers and Jos Uiterwijk, for spending parts of their summer 2011 reading through the manuscript of this thesis and for giving very useful comments on it.

My PhD research has been carried out within the NWO project TACTICS on "Theoretical and Algorithmic Complexity Thresholds in Computer gameS", in which Maarten Schadd has been my parter. I am very grateful to him for providing me with insights into the Maastricht-perspective on our project.

I would like to thank my co-authors Cédric Dégremont, Nina Gierasimczuk, Jakub Szymanik and Fernando Velázquez-Quesada, who have directly contributed to the content of this thesis. Among them, Cédric has accompanied me in my PhD work from the beginning of my first year until some last-minute adjustments before submitting this thesis. Cédric, Nina, Jakub and Fer have not only been great collaborators for me but also very good friends with whom I could share much more than just ideas about logic, learning theory, complexity theory and cognitive science. Among many things, I will never forget Jakub's skiing lessons, in which Fer and I learned how to ski in a Polish snowstorm, and our adventures on and along the Yangtze.

Actually starting my PhD and also writing Chapter 2 would not have been possible without the help of Ulle Endriss and Eric Pacuit. Thanks also for giving me useful feedback on various parts of my work during the PhD.

I would like to thank Frank Radmacher for not only understanding some very vaguely formulated problems of mine regarding Sabotage Games, but also solving them almost instantly.

I am very grateful to Edith Hemaspaandra for taking the time to carefully go through some technical parts of Chapter 5. I have learnt a lot from her insights into how to tackle problems in computational complexity.

I would like to thank Joel Uckelman and Federico Sangati for the discussions we had about the game *Eleusis*, which is the topic of Chapter 6. I have been very impressed by how Fede turned our discussions and ideas into an online implementation of the game, while at the same time also finishing his thesis. Thanks to Pablo Seban for introducing me to this game at ESSLLI 2008.

Moreover, I would like to thank Davide Grossi, Fenrong Liu, Luigi Sauro, Sunil Simon and Rineke Verbrugge for interesting discussions and very useful comments on various parts of this dissertation.

I am grateful to Kai-Uwe Kühnberger for encouraging me to go to Amsterdam to study logic in the first place. Also, I would like to thank Stefan Bruda and Leen Torenvliet whose classes got me interested in complexity theory.

Throughout my PhD, I have been working in Johan's dynamic LIRA group, which has been an extremely inspiring environment for me, even when I was not working on dynamic epistemic logic. Special thanks to Fer for organizing the seminar together and to Davide, Ştefăn, Cédric, Alexandru, Sonja and Ben for taking over the organization when our seminar had turned into a nationwide event.

I would like to thank François Schwarzentruber, and Paolo Turrini for fruitful discussions on coalitions, complexity and modal logics. It was very reassuring for me when I met you when just having started my PhD and I discovered that I was not the only one who has just started working on those things.

Even though this might not be immediately present in the rest of this thesis, I have had the chance to also explore real strategic reasoning of people (children, in particular). I would like to thank Marian Counihan for her initiative to try to bring logicians and psychologists together and for the joint organization of the summer reading group in 2009. I am also very grateful to Han van der Maas and Maartje Raijmakers for giving me the opportunity to get involved in the *rekentuin*, which has been a great experience for me.

I would like to thank the AUC logic teaching team Cédric, Fernando, Jan, Nina and Ştefăn. I would have never thought that spending whole days on correcting and grading logic exams and could actually be fun.

I am greatly indebted to Jenny Batson, Karin Gigengack, Tanja Kassenaar, Ingrid van Loon, Peter van Ormondt and Marco Vervoort, for all their efforts to make the ILLC such a great place for doing research. Thanks to Karin for even digitalizing the content of some whiteboards during the reorganization at Science Park, making sure no idea will be lost. Special thanks to my favorite security guard René Goedman!

I am very grateful to the (former) members of the ILLC PhD council Inés Crespo, Nina Gierasimczuk, Umberto Grandi, Raul Leal, Gideon Maillette de Buy Wenniger and Federico Sangati for turning complaints and suggestions into concrete actions and improvements.

In the beginning of my PhD, I have benefited a lot from the support and help of Ans Rekers. I am also very grateful to Doris Dhuygelaere and Rosemarijn Franses whose 'loopbaan advies' and 'sollicitatietraining' has been extremely valuable for me in the final stages of my PhD. Thanks also to Niels Molenaar, the international office and the housing office of the Faculty of Science who have been extremely supportive throughout my stay in Amsterdam whenever I was about to be homeless.

I would like to thank my JK office mates Ştefăn, Hugo, Michael and Christian, and all my open space mates in the "Stiltezône" (which is actually a more fun place than the name might suggest). Special thanks to Ştefăn, who was always there to help me with all kinds of practical problems.

I would like to thank the extended COMSOC lunch family and the coffee gang for all the refreshing breaks. Thanks Daniele, Davide, Joel, Stéphane, Raquel, Ulle and Umberto!

Thanks to Mark, Hartmut, Christian, Silvia, Helle, Jens, Thomas, Lauri, Jarmo, Julian, Markos, Szymon, Alexandra, Sunil, Sara, Yanjing and Jonathan for inspiring discussions and various fun activities well beyond logic.

As much as I have really enjoyed most of the process of doing research and writing this thesis, I must say that I might have enjoyed the time that I have spent on other things even more.

The most important person for me to mention here is Raul, who has been with me for most of the six years of my Amsterdam life and thanks to whom Amsterdam very quickly started to feel like home for me. I want to thank him for his love, his honesty and for always coming up with a Plan B (which usually is better than the original plan) whenever things did not work out as I wanted.

I have written most of my dissertation during the summer, when the ILLC was rather empty and distractions were harder to find. Luckily, the 'USC universum' has provided me with great opportunities for quick escapes from writing this thesis.

I would have never thought that I would actually get into (rock) climbing after moving to a country that does not really have that many high mountains (yet). I would like to thank Remco, Elout, Bas and Coen from whom I have learnt a lot. Thanks to my climbing partners Anna, Raul, Michele, Christian and Szymon, the ASAC-ers and the boulder gang.

Anna has been a great partner in all kinds of sports activities throughout the last years. Our rock climbing and running trips have always been at least as refreshing for me as they have been exhausting.

Thanks to Emile, not only for teaching me probably more Dutch words than Mr. van Dale will ever do.

I would like to thank my friends from home, Dani, Jenni, Jessica, Markus, Susan, Torsten and Verena, for all the ice creams, city walks, running trainings, coffees, dinners and picnics, and for often making time to meet up with me on short notice.

Finally, I would like to thank my parents for all their support, and Christian for being the coolest brother ever.

Amsterdam Lena Kurzen
September, 2011.

Chapter 1

Introduction

Looking back at the most exciting and best days of your life so far, there is a good chance that most of these days will probably not be days of total isolation. Most probably, they will to some extent involve interaction with other beings. Moreover, it is not only the exciting life changing encounters but also very basic situations in everyday life in which interaction with other individuals plays an essential role. Examples of interaction in everyday life include having a meeting at work, having dinner with friends or family while discussing recent life developments and plans for the weekend, giving or attending lectures and engaging in some sports activity after work. More abstractly, much of politics and economics is also about the interaction of individuals or institutions.

Additionally, interaction is an interesting and up-to-date topic due to technological developments which have the effect of interaction changing form, making the world in some sense more connected and allowing for faster communication and exchange of information between individuals (cf. e.g. Friedewald and Raabe (2011)). Increased speed and number of participants of interactive processes can lead to an increased difficulty of predicting what will be the outcome of such processes, e.g. who will be the winner of an Internet auction, or how long it will take until some new piece of information has reached a certain part of the population.

The emerging networks involving computer- and communication technologies are becoming increasingly heterogeneous and complicated, which calls for a solid foundational theory for the analysis of such systems that will help us to get a deeper understanding of interactive processes (Ministry of Public Management, Home Affairs, Posts and Telecommunications, Japan 2004; Egawa et al. 2007). Such a theory should provide us with

1. tools to formalize different interactive processes and

2. a measure of complexity of interaction.

In this dissertation, we propose a formal study of the complexity of networks of interaction. We choose an approach based on *modal logic* and *computational complexity theory*. The main motivation for this the following.

1. Using modal logic we can model interaction networks on an abstract level, and moreover various more specific frameworks have been built upon modal logic to model more particular concepts involved in interaction.

2. Computational complexity theory provides us with tools to classify problems and tasks according to their intrinsic difficulty, abstracting away from particular implementations.

We now present the basics of both of them and illustrate how they can be used to study interactive processes. We start with modal logic.

1.1 Modal logics for interaction – the basics

Taking an abstract perspective on modern interaction, much of it can be represented as networks: sets of nodes with different kinds of connections between them. Examples of this include communication networks and social networks.

In general, modal logic is a formal framework for reasoning about *relational structures*. Basic relational structures are closely related to graphs as they basically consist of a set of vertices (states, worlds, points) and binary relations on this set; the binary relations can be seen as edges or transitions. As an example of such a structure, think about all your friends and construct a graph by taking all friends as vertices and adding an edge between any two persons who are friends of each other. This will be a relational structure.

For a detailed introduction to modal logic, we refer the reader to Blackburn et al. (2001), and to van Benthem (2010) for an introduction to different ways to use modal logic for reasoning about interaction.

To fix the notation, in what follows, we let PROP be a countable set of propositional variables, typical members of which are denoted by $p, q, \ldots$ etc. They will be used to represent simple facts, which can be either true or false.

Definition 1.1 (Basic Kripke models) A *Basic Kripke model* $\mathcal{M}$ is of the form (W, R, V), where

- $W \neq \varnothing$ is a set of worlds (states), also referred to as the *domain* of $\mathcal{M}$ and also denoted by $Dom(\mathcal{M})$.

- the accessibility relation R is a binary relation on W, i.e., $R \subseteq W \times W$, and

- $\mathsf{V} : \textsc{prop} \to \wp(W)$ is a propositional valuation function, assigning to each propositional letter a subset of W. ◀

Such a model can be used to represent a network in which we have one kind of connection. As an example, consider a computer network; i.e.,

- W is a set of computers,
- for all $w, v \in W$ we have $(w, v) \in R$ if and only if there is a cable connecting the computers w and v.
- Propositional letters could then e.g. represent different properties of computers.

In most of this dissertation, we will be concerned with Kripke models that have more than one accessibility relation. Mostly, we will deal with models that are designed for formalizing a situation involving a set of individuals (which we refer to as *agents*), and there will be an accessibility relation for each agent.

Definition 1.2 (Multi-agent Kripke models) A *Kripke model* $\mathcal{M}$ for a finite set of agents $\mathtt{N}$ is of the form $(W, (R_i)_{i \in \mathtt{N}}, \mathsf{V})$, where

- $W \neq \emptyset$ is a set of worlds (states), also referred to as the *domain* of $\mathcal{M}$, denoted by $Dom(\mathcal{M})$.
- for each $i \in \mathtt{N}$, R_i is a binary relation on W, i.e., $R_i \subseteq W \times W$, and
- $\mathsf{V} : \textsc{prop} \to \wp(W)$ is a propositional valuation function. ◀

The above definition is very general and allows for many different interpretations as to what the accessibility relations of the agents should mean. Some examples of possible interpretations of $(w, v) \in R_i$ – sometimes also denoted by wR_iv – are the following:

- Agent i finds the state v at least as good as w (cf. Chapters 2 and 3).
- At w, agent i considers it possible that he actually is in v (cf. Chapter 5).
- At w, agent i can
  - change the situation from w into v (cf. Chapter 3),
  - move to v (cf. Chapter 4).

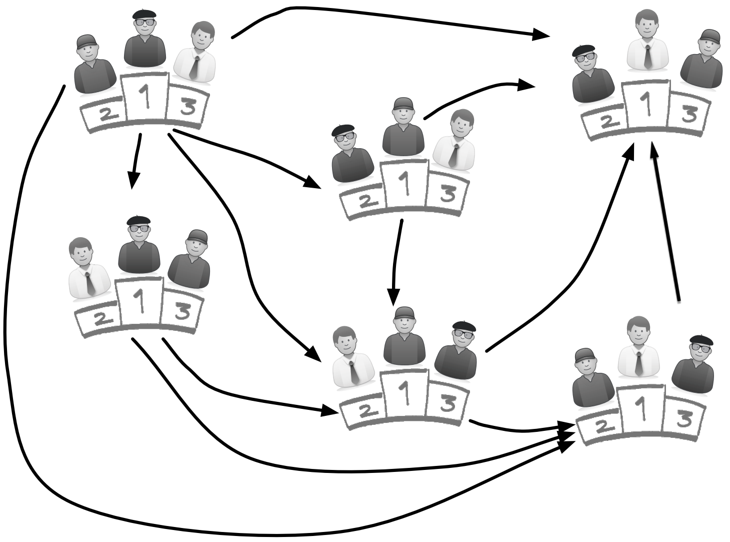
A straightforward generalization is to take an accessibility relation for each subset of $\mathtt{N}$, i.e., for each group (sometimes referred to as *coalition*) of agents (cf. Chapters 2 and 3). Additionally, in Chapters 2 and 3, we will also consider models in which we have accessibility relations for actions. These relations then show how performing actions can change the state of a system.

More generally, we have Kripke models of the form

$$\mathcal{M} = (W, (R_a)_{a \in \Sigma}, \mathsf{V}),$$

where Σ is a finite set of labels, which are used to represent different kinds of connections between the elements in W. We will now give some examples to illustrate different kinds of Kripke models

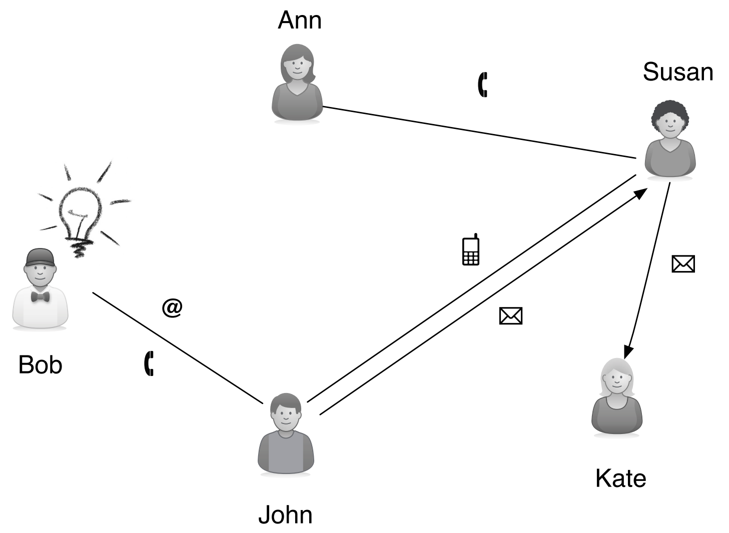
Example 1.3 Consider a situation in which we have a competition between three competitors C_1, C_2, C_3 and the outcome of the competition will be a ranking of the three candidates. As an external observer, we can then have preferences over the six possible outcomes. The following is a graphical representation of the situation, with the arrows representing our preferences over the different outcomes.



A Kripke model $\mathcal{M} = (W, R, \mathsf{V})$ can then be defined accordingly.

- W contains the six possible outcomes.
- R is defined in such a way that wRv if and only if v is preferred over w, and
- the valuation V defined such that each propositional variable $p_{C_1} p_{C_2}, p_{C_3}$, is made true in exactly those outcomes in which the respective candidate wins.

With such a formalization, we can reason e.g. about whether for some outcome it is the case that there is some outcome which is better and for which p_{C_1} is true (i.e., C_1 wins). ◀



Example 1.4 Consider the situation in which five individuals can communicate via a communication network represented as follows.

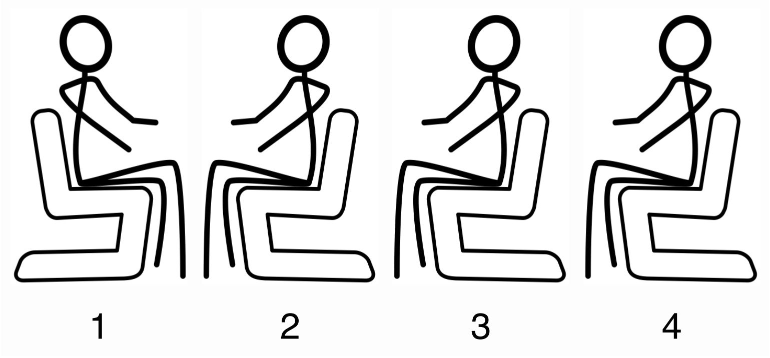
We can define a Kripke model $\mathcal{M} = (W, (R_a)_{a\in\Sigma}, \mathsf{V})$, with Σ being the set of labels representing the four types of connections (landline, mobile, email, letter).

- $W = \{\text{Ann, Bob, John, Kate, Susan}\}$,
- each R_a is defined as in the diagram, and
- we have one propositional letter p_{idea}, with $\mathsf{V}(p_{idea}) = \{\text{Bob}\}$.

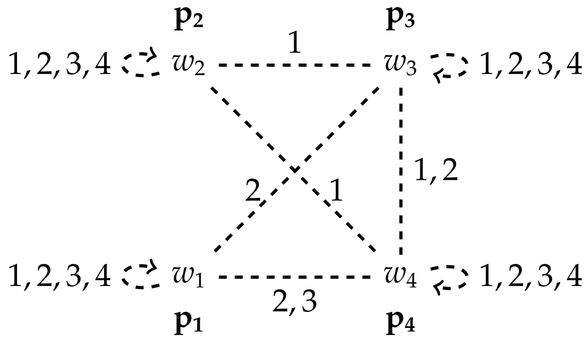
Interesting questions that arise here could e.g. be whether Bob's idea can be communicated to Kate even if after each communication step some connection breaks down. We will consider such a setting in Chapter 4. ◀

Example 1.5 (Cards on a train) Consider a situation with four agents traveling by train, sitting in positions as depicted below. They want to play a game, and in order to determine who is the first to play they distribute four cards among them. Three are of the suit ♤ and one is of the suit ♧. The player who gets the ♧ wins this pre-game and will be the first to start the real game. Suppose that each player has gotten one of the four cards. They each can see their own card, and due to the positions in which they are sitting, additionally

- both 3 and 4 can see the card of 2,
- 4 can see the card of 3.



Now, we construct a Kripke model for the four agents which represents the information they have about who might have the ♧. We use propositional letters $\mathbf{p_i}$ to represent the fact that agent i has the card ♧.



Formally, the model is defined as follows. $\mathcal{M} = (W, (R_i)_{i \in \mathtt{N}}, \mathsf{V})$ with $\mathtt{N} = \{1,2,3,4\}$, where

- $W = \{w_1, w_2, w_3, w_4\}$ is the set of four possible situations; w_i represents the situation in which agent i has the ♧,
- the accessibility relations R_i representing the uncertainty of the agents are defined as follows
  - $R_1 = Id \cup \{(w_2, w_3), (w_3, w_2), (w_2, w_4), (w_4, w_2), (w_3, w_4), (w_4, w_3)\}$,
  - $R_2 = Id \cup \{(w_1, w_3), (w_3, w_1), (w_4, w_1), (w_1, w_4), (w_3, w_4), (w_4, w_3)\}$,
  - $R_3 = Id \cup \{(w_1, w_4), (w_4, w_1)\}$,
  - $R_4 = Id$,

with $Id = \{(w, w) \mid w \in W\}$,

- $\mathsf{V}(p_i) = w_i$ for all $i \in \{1, 2, 3, 4\}$.

Now, some interesting questions come up such as e.g.

- Can we make a public announcement to the agents that has the effect that after this all agents have the same information that they would have if agent 1 was facing the other direction (i.e., if he was facing left instead of right)?
- Does agent 1 have the same information about agent 2 as 2 has about 1?

Questions of this type will be investigated in Chapter 5. ◂

A related framework more general than the Kripke models defined above is that of *neighborhood* models, which instead of a binary accessibility relation on $Dom(\mathcal{M})$ have a function $\nu : Dom(\mathcal{M}) \to \wp(\wp(Dom(\mathcal{M})))$ assigning to every state a set of sets of states. For the details we refer to Hansen et al. (2009). In Chapters 2 and 3, we will consider neighborhood frameworks specifically designed for reasoning about the strategic ability of groups.

Then a formal language for Kripke models is specified as follows.

Definition 1.6 (Multi-agent modal language) The basic modal language for the finite set of agents $\mathtt{N}$ is defined as follows.

$$\varphi ::= p \mid \bot \mid \neg\varphi \mid \varphi \vee \psi \mid \Diamond_i \varphi$$

with $p \in \textsc{prop}$ and $i \in \mathtt{N}$. ◂

Additionally, we will use $\Box_i$, the dual operator of $\Diamond_i$: $\Box_i\varphi := \neg\Diamond_i\neg\varphi$. For notational convenience, we also use the following (standard) abbreviations. $\varphi \wedge \psi := \neg(\neg\varphi \vee \neg\psi)$, $\varphi \to \psi := \neg\varphi \vee \psi$, $\varphi \leftrightarrow \psi := (\varphi \to \psi) \wedge (\psi \to \varphi)$ and $\top := \neg\bot$.

The language is interpreted at a state w in a multi-agent Kripke model $\mathcal{M}$ as follows.

$$\begin{array}{lll}
\mathcal{M}, w \models p & \text{iff} & w \in \mathsf{V}(p). \\
\mathcal{M}, w \models \bot & & \text{never.} \\
\mathcal{M}, w \models \neg\varphi & \text{iff} & \text{it is not the case that } \mathcal{M}, w \models \varphi. \\
\mathcal{M}, w \models \varphi \vee \psi & \text{iff} & \mathcal{M}, w \models \varphi \text{ or } \mathcal{M}, w \models \psi. \\
\mathcal{M}, w \models \Diamond_i\varphi & \text{iff} & \text{there is some } v \text{ with } (w, v) \in R_i \text{ and } \mathcal{M}, v \models \varphi.
\end{array}$$

Definition 1.7 (Multi-agent modal logic $\mathbf{K}_{\mathtt{N}}$) The basic multi-agent modal logic $\mathbf{K}_{\mathtt{N}}$ for a finite set of agents $\mathtt{N}$ has the following axioms.

$$\text{(K)} \quad \Box_i(p \to q) \to (\Box_i p \to \Box_i q)$$

for each $i \in \mathtt{N}$ and the following three rules of inference:

Modus ponens: From φ and $\varphi \to \psi$ conclude that ψ.

Uniform substitution: From φ conclude that χ, with χ being obtained from φ by uniformly replacing propositional letters by formulas.

Necessitation: From φ, conclude that $\Box_i\varphi$. ◀

We write $|\mathcal{M}|$ to refer to the size of the model $\mathcal{M}$ – to be more precise, the size of a reasonable representation of the model. We refer to a pair $(\mathcal{M}, w)$ with $w \in Dom(\mathcal{M})$ as a *pointed* model. We will come back to this in Chapter 4.

Comparing models and reasoning about submodels

In various parts of our investigation we will need a reasonable notion of two models being similar. We make use of the notions of simulation, simulation equivalence and bisimulation.

Definition 1.8 (Simulation) We say that a pointed Kripke model $(\mathcal{M}, w)$, with $\mathcal{M} = (W, (R_i)_{i\in\mathtt{N}}, \mathsf{V})$ and $w \in W$, is simulated by another pointed Kripke model $(\mathcal{M}', w')$ (denoted by $(\mathcal{M}, w) \sqsubseteq (\mathcal{M}', w')$) with $\mathcal{M}' = (W', (R'_i)_{i\in\mathtt{N}}, \mathsf{V}')$ and $w' \in W'$ if the following holds.

There exists a binary relation $Z \subseteq W \times W'$ such that wZw' and for any pair of states $(x, x') \in W \times W'$, whenever xZx' then for all $i \in \mathtt{N}$:

1. x, x' verify the same proposition letters.
2. if xR_iz in $\mathcal{M}$ then there exists $z' \in W'$ with $x'R'_iz'$ and zZz'. ◀

We say that $\mathcal{M} = (W, (R_i)_{i\in\mathtt{N}}, \mathsf{V})$ is simulated by $\mathcal{M}' = (W', (R'_i)_{i\in\mathtt{N}}, \mathsf{V}')$ (denoted by $\mathcal{M} \sqsubseteq \mathcal{M}'$) if there are $w \in W$ and $w' \in W'$ such that $(\mathcal{M}, w) \sqsubseteq (\mathcal{M}', w')$. We say that a simulation $Z \subseteq W \times W'$ is *total* if for every $w \in W$, there is some $w' \in W'$ such that wZw', and for every $w' \in W'$, there is some $w \in W$ such that wZw'. If $\mathcal{M}$ is simulated by $\mathcal{M}'$ by means of a total simulation, we say $\mathcal{M} \sqsubseteq_{total} \mathcal{M}'$. Moreover, we say that $\mathcal{M} = (W, (R_i)_{i\in\mathtt{N}}, \mathsf{V})$ and $\mathcal{M}' = (W', (R'_i)_{i\in\mathtt{N}}, \mathsf{V}')$ are simulation equivalent if $\mathcal{M}$ simulates $\mathcal{M}'$ and $\mathcal{M}'$ simulates $\mathcal{M}$.

The truth of positive existential formulas in pointed models is preserved under simulations.

Example 1.9 In order to get an intuitive idea of simulation, consider two pointed Kripke models $(\mathcal{M}, w), (\mathcal{M}', w')$ both with one agent (Bob), and the accessibility relations representing the uncertainty of Bob. Then

$$(\mathcal{M}, w) \sqsubseteq (\mathcal{M}', w')$$

means that in $(\mathcal{M}, w)$ Bob has more refined information than in $(\mathcal{M}', w)$, i.e., in $(\mathcal{M}', w')$ Bob has more uncertainty. ◀

The following notion is stronger than simulation equivalence.

Definition 1.10 (Bisimulation) A local bisimulation between two pointed Kripke models with set of agents $\mathtt{N}$, $(\mathcal{M}, w)$ with $\mathcal{M} = (W, (R_i)_{i \in \mathtt{N}}, \mathsf{V})$ and $(\mathcal{M}', w')$ with $\mathcal{M}' = (W', (R_i')_{i \in \mathtt{N}}, \mathsf{V}')$ is a binary relation $Z \subseteq W \times W'$ such that wZw' and also for any pair of worlds $(x, x') \in W \times W'$ whenever xZx' then for all $i \in \mathtt{N}$:

1. x, x' verify the same proposition letters.

2. if xR_iu in $\mathcal{M}$ then there exists $u' \in W'$ with $x'R_i'u'$ and uZu'.

3. if $x'R_i'u'$ in $\mathcal{M}'$ then there exists $u \in W$ with xR_iu and uZu'. ◀

We say that $\mathcal{M} = (W, (R_i)_{i \in \mathtt{N}}, \mathsf{V})$ and $\mathcal{M}' = (W', (R_i')_{i \in \mathtt{N}}, \mathsf{V}')$ are bisimilar ($\mathcal{M} \underline{\leftrightarrow} \mathcal{M}'$) if there are $w \in W$ and $w' \in W'$ such that $(\mathcal{M}, w) \underline{\leftrightarrow} (\mathcal{M}', w')$. A bisimulation $Z \subseteq Dom(\mathcal{M}) \times Dom(\mathcal{M}')$ is *total* if for every $w \in Dom(\mathcal{M})$, there is some $w' \in Dom(\mathcal{M}')$ such that wZw', and for every $w' \in Dom(\mathcal{M}')$, there is some $w \in Dom(\mathcal{M})$ such that wZw'. Then we write $\mathcal{M} \underline{\leftrightarrow}_{total} \mathcal{M}'$.

Bisimilarity is an equivalence relation. Bisimilarity implies modal equivalence: If two pointed models are bisimilar, they satisfy the same modal formulas. We will use this primarily in Chapters 3 and 5. In Chapter 3, we also use that invariance under bisimulation characterizes exactly those formulas of first-order logic with one free variable that are equivalent to the standard translation of a modal formula into first-order logic formulas (van Benthem 1976).

In order to illustrate why the notion of bisimulation is interesting for our investigation, let us go back to Example 1.5 where we formalized the situation of the four travelers playing cards on a train. For determining if this situation is equivalent (with respect to the information of the players) to a situation in which they sit in a different configuration, we basically have to check if the two models are bisimilar.

In Example 1.5, we were also interested in whether it was possible to make a public announcement that has the effect of reducing the uncertainty of some agent in a certain way. A public announcement can have the effect that some situations that were considered possible before can be eliminated. This motivates why we would like to have a way to talk about parts of a model, e.g. only those states in which some proposition is true. For this, we need the notion of a *submodel*.

Definition 1.11 (Submodel) We say that $\mathcal{M}'$ is a submodel of $\mathcal{M}$ iff $W' \subseteq W$, $\forall i \in \mathtt{N}$, $R_i' = R_i \cap (W' \times W')$, $\forall p \in \textsc{prop}$, $\mathsf{V}'(p) = \mathsf{V}(p) \cap W'$. ◀

Definition 1.12 (Generated submodel) We say that $\mathcal{M}' = (W', (R_i)'_{i \in \mathtt{N}}, \mathsf{V}')$ is a generated submodel of $\mathcal{M} = (W, (R_i)_{i \in \mathtt{N}}, \mathsf{V})$ iff $W' \subseteq W$ and $\forall i \in \mathtt{N}$, $R_i' = R_i \cap (W' \times W')$, $\forall p \in \textsc{prop}$, $\mathsf{V}'(p) = \mathsf{V}(p) \cap W'$ and if $w \in W'$ and wR_iv then $v \in W'$.

The submodel of $\mathcal{M}$ generated by $X \subseteq W$ is the smallest generated submodel $\mathcal{M}'$ of $\mathcal{M}$ with $X \subseteq Dom(\mathcal{M}')$. ◀

A generated submodel is actually bisimilar to the original model: for all $w' \in Dom(\mathcal{M}')$ we have that $(\mathcal{M}, w') \underline{\leftrightarrow} (\mathcal{M}', w')$. Thus, the truth of modal formulas is invariant under taking generated submodels.

To summarize, we have thus seen the basic ideas of how the framework of modal logic can be used to model interaction. In Chapter 3, we will additionally introduce some *hybrid* and *Boolean modal* extensions of modal logic which can explicitly refer to particular states in the model and e.g. talk about the intersection of relations.

In general, we are interested in what kind of social phenomena we can model in modal logic. However, it is not only important whether some situation can be modeled but if we want to actually use the formal framework of modal logic to reason about social phenomena, computational properties also play an important role as we want to be able to reason efficiently. This leads us to computational complexity theory, which we will use to capture the complexity of interaction.

1.2 The basics of computational complexity – from the perspectives of modal logic and interaction

In our investigation of the complexity of interaction, we will use *computational complexity theory*. We will now briefly present the basics that underlie the complexity analyses that are given in Chapters 2 to 6.

First of all, we note that in this dissertation, we use complexity theory purely as a tool for analyzing different problems with respect to how difficult it is to solve them computationally. For the reader interested in the details of computational complexity theory itself, we refer to the literature (e.g. Papadimitriou (1994)). To be more precise, the way in which we use computational complexity theory here is to classify *decision problems* according to their "difficulty". By decision problems we mean problems that consist of some input and a yes/no-question about the input. In order to give the reader an idea of such problems, we will now give those decision problems which are usually used to evaluate a logic with respect to how efficiently it can be used for reasoning.

Decision Problem 1.13 (Satisfiability of a logic) (SAT)
Input: *Formula φ in the language $\mathcal{L}$ of a logic* **L**.

Question: *Is there a model of* **L** *that satisfies the formula φ?* ◀

What does the satisfiability problem actually mean for modal logics for reasoning about interaction?

In the context of modal logic frameworks for interaction, the satisfiability problem is concerned with deciding whether it is possible to design a system according to the specifications given by the formula φ, i.e., a system in which φ holds. This is somewhat in line of what *mechanism design* (cf. Osborne and Rubinstein (1994)) is about.

Decision Problem 1.14 (Model checking of a logic) (combined complexity)
Input: *Formula φ in the language $\mathcal{L}$ of a logic* **L**, *a finite model $\mathcal{M}$ of* **L**.
Question: *Does $\mathcal{M}$ satisfy φ?* ◄

Decision Problem 1.15 (Model checking a formula φ of some logic) (data complexity)
Input: *A finite model $\mathcal{M}$ of a logic* **L**.
Question: *Does $\mathcal{M}$ satisfy the formula φ?* ◄

Note the difference between the two versions of model checking: In Problem 1.14, the formula is part of the input while in Problem 1.15 the formula is fixed.

What do the two model checking problems actually mean for modal logics for reasoning about interaction?

Both problems ask whether a system has a certain property φ. More concretely, in the context of interaction, the problems are concerned with whether an interactive situation has the property φ. Interesting choices for φ could be

- the property of the current state of the interaction being stable in the sense that none of the participants has an incentive to act in order to change it (cf. Chapter 3),
- the property that a particular agent has the ability to ensure that the interaction leads to success for her (cf. Chapter 4),
- the property that some particular fact is common knowledge among all the agents (cf. Chapter 5).

As for Problem 1.15 the formula is not part of the input, but fixed, this problem is independent from how succinctly the formula φ expresses a certain property we are interested in. Investigating the complexity of this problem for a given logic and a given formula φ of the logic's language, thus tells us how difficult it is to check whether the particular property φ holds.

Decision problem 1.14 is of a different nature. Investigating its complexity for a given logic tells us how difficult it is *in general* to check if systems have certain properties definable in the logic.

In computational complexity theory, the difficulty of decision problems is understood in an abstract computational way: it is understood in terms of how much more resources (in terms of time and space) are needed to compute an answer to the question as the size of the input gets bigger. For a classification of problems according to their computational difficulty, a model of computation is needed. Here, we use the *Church-Turing Thesis* (Church 1936; Turing 1936), which says that a problem can be solved algorithmically if and only if this can be done using a *Turing machine*.

Deterministic Turing machines are a theoretical model of computation; they consist of an infinite tape, a head which is used to read the input from the tape and to write on it, a transition function that specifies for every state the machine can be in and for every symbol that it can read on the tape what it should do: i.e., what symbol it should write or whether it should move one step further left or right on the tape. The computation stops once a final state is reached. Nondeterministic Turing machines differ in the way that instead of transition functions they can have arbitrary transition relations.

The *time* needed for a computation is measured in terms of the number of steps of the computation. The (memory) *space* needed for a computation refers to the number of tape cells needed for the computation.

For a categorization of problems according to their complexity independently of particular implementations of algorithms, we use the *invariance thesis* which says that for reasonably encoded input and two reasonable machines the complexity of the computation of the machines given that input differs by at most polynomial time and constant memory space (van Emde Boas 1990).

We now give the complexity classes into which most of the problems considered in this dissertation fall.

1.2.1 Complexity classes

Deterministic logarithmic space (L) The class L (LOGSPACE) contains very easy problems which can be solved by a deterministic Turing machine using only memory space logarithmic in the size of the input.

Nondeterministic logarithmic space (NL) The easiest problems that we explicitly encounter in this dissertation are in the class NL, which is the class of problems that can be solved using a nondeterministic Turing machine that uses an amount of memory space that is logarithmic in the size of the input. An example of such a problem is the *Reachability* problem, the problem of deciding if there is a path between two given points in a graph.

As example instance of Reachability consider the following input: the graph drawn in Example 1.4 and the vertices *Bob* and *Ann*. Then the reachability problem asks whether there is a path from Bob to Ann in the communication network.

The reachability problem will be used in Chapters 4 and 5, where it will appear in a game on a graph, and in the reasoning about whether a fact is commonly known among a group of agents, respectively.

Deterministic polynomial time (P) The class P (PTIME) contains NL and is the class of all problems that are solvable by a deterministic Turing machine within polynomial many time steps with respect to the size of the input. P is one of the most important complexity classes; problems which are in P are said to be *tractable* as they can be solved in an efficient way.

An example of a problem in this class is the decision problem of given two Kripke models to decide whether there is a bisimulation between them (Balcázar et al. 1992), which is a problem we will examine more closely in Chapter 5 when investigating the complexity of comparing the information that different agents have.

Nondeterministic polynomial time (NP) NP contains all the previously mentioned classes. It is the class of problems solvable by a nondeterministic Turing machine in polynomial time with respect to the size of the input. Intuitively, this class contains problems for which a proposed positive answer can be verified (deterministically) in polynomial time.

Problems which are at least as hard as all the problems in NP are called NP-*hard*. A problem P' is NP-hard if every problem P in NP can be reduced to it by a polynomial many-one reduction: a function which can be computed in polynomial time that transforms instances (input) x of P into an instance $f(x)$ of P' such that x is a positive instance of P if and only if $f(x)$ is a positive instance of P'. If a problem is NP-hard and in NP, it is NP-*complete*. For the details, we refer to Garey and Johnson (1990). A very prominent example of an NP-complete problem is the satisfiability problem of propositional logic.

Polynomial space (PSPACE) PSPACE contains the problems which can be solved by a Turing machine using only a polynomial amount of memory space, with respect to the size of the input. PSPACE-completeness and -hardness are defined analogously as for NP. This complexity class also plays a central role in this dissertation as first of all for many game-like interaction processes for two players the problem of deciding which of the players can win is PSPACE-complete. An example of such a game will be studied in Chapter 4. Moreover, the class also plays an important role in modal logic, as for the basic modal logic the satisfiability problem is PSPACE-complete (Ladner 1977).

We remark that allowing for nondeterminism does not add any computational power for deterministic PSPACE Turing computations (Savitch 1970).

Deterministic exponential time (EXPTIME), nondeterministic exponential time (NEXPTIME)

EXPTIME is the class of problems solvable by a deterministic Turing machine within exponentially many time steps, with respect to the input size.

NEXPTIME is the class of problems solvable by a nondeterministic Turing machine within exponentially many time steps, with respect to the size of the input. In Chapters 2 and 3, we will encounter problems for which we only know procedures that can solve them in nondeterministic exponential time

Undecidable problems There is also a whole hierarchy of classes of problems which cannot be solved algorithmically. We will encounter some of those problems in Chapter 3, when investigating the satisfiability problem of some logics for reasoning about cooperation of agents. Also, in Chapter 6 we will find an undecidable problem that can come up in the play of a particular card game.

To summarize, we have thus seen some examples of how modal logic can be used for reasoning about relevant concepts in the interaction of agents. Examples of such concepts include preferences, actions and information. Computational complexity provides us with tools to classify decision problems according to their intrinsic difficulty.

From the examples we considered, we could already see that

> *the choice of decision problem crucially determines to what extent we can draw conclusions about the complexity of interactive situations.*

This is something that we will pay particular attention to throughout the remainder of this dissertation. After having introduced the setting in which our investigations take place, we will now present the main questions that we address.

1.3 Research questions

The basic frameworks previously introduced give rise to a variety of interesting questions to be investigated with respect to the complexity of interaction. In the current work, we will start with an abstract perspective, focusing on logical theories for social actions and from there move to more concrete settings in which we focus on the algorithmic tasks that interacting agents face.

1.3.1 Complexity of reasoning about interaction in modal logics

We have seen that using modal logic we can model various aspects of networks of interaction. For capturing an interactive situation involving multiple agents on an abstract level, the *strategic ability* of agents plays a crucial role. We can describe an interactive situation in terms of what the abilities of the agents are, i.e., what results they can achieve. This can then be done on different levels of abstraction. We could for instance just focus on the abilities of different companies to achieve a certain market share themselves and also ensure that a certain competitor does does not get a share bigger than some percentage. This way we will get an abstract and external description of how the power is distributed in the interactive scenario involving the different companies. This way, we could make predictions of what are the possible outcomes that can arise. But sometimes this might not be enough, as an analysis at this level only reveals what could eventually be achieved. As soon as we want to reason about the situation on a more concrete level, the question that will come up is how exactly the results can be achieved. If we want to be able to reason about this, we also need to represent actions by which results can be achieved.

This illustrates that motivated by conceptual considerations, there are various ways to model strategic interaction of individuals and groups.

Then the question arises as to what kinds of modal logic frameworks are "best" for modeling which aspects of interaction, which is our first research question.

> **Research Question 1** *What formal frameworks are best suited for reasoning about which concepts involved in interaction?*
>
> - *What should be the primitive notions a formal approach should be based on?*

We will address this question in Chapters 2 and 3 on an abstract level, for very general modal logic frameworks for reasoning about the abilities of individuals and groups to achieve something. Evaluating how good some framework is for reasoning about certain aspects of cooperation can be done in different ways. A natural way to evaluate a system for reasoning about interaction is to determine which interesting concepts can be expressed and reasoned about. In the context of logics for the interaction of agents, it is often desired that the formal system can express concepts from game theory (Osborne and Rubinstein 1994), e.g. the property that a certain state of the system is stable in the sense that there is no incentive to act in order to change the state.

Apart from being able to express interesting concepts, also computational properties of the logical framework play an important role in its evaluation. The complexities of model checking and satisfiability problems tell us how much computational resources are needed to check if a given system has some property, and how difficult it is to determine if there is a system that has certain properties, respectively. For the design of modal logics for reasoning about interacting agents, the challenge is to develop systems which are sufficiently expressive given the particular situations to be modeled while still having good computational properties so that reasoning can be done efficiently. This illustrates the need for a systematic study of the connection between expressivity of logics with respect to concepts from game theory and social choice theory and the computational complexity of the logics. In Chapter 3, we study this by a semantic comparison of modal logic systems for reasoning about group ability.

Both model checking (combined complexity) and satisfiability are problems concerned with *every* formula of a logic's language. As our aim is to investigate the complexity of interactive processes, let us recall the main objective of this dissertation, which is to investigate the complexity of interaction. An analysis in terms of the complexity of satisfiability and model checking of logical frameworks for interaction provides a very abstract view on the complexity of interaction. This leads us to the following question.

> *What do the complexity of satisfiability and model checking really tell us about the complexity of interaction?*

Describing the complexity of interaction in terms of satisfiability and model checking might not be very accurate in the sense that it might just be the case that model checking and satisfiability for formulas that express interesting concepts about interaction all live on one side of the complexity spectrum. In other words, it might e.g. just be the case that the formulas interesting for interaction are not the ones which make satisfiability or model checking hard.

From complexity of logics for interaction to complexity of interaction itself. In order to get closer to the complexity of interaction itself, rather than the general complexity of formal systems used to reason about interaction, there are basically two possible paths to take.

1. Focus the complexity theoretical analysis of logical frameworks on those properties that are relevant for interaction.

2. Investigate the algorithmic complexity of tasks that arise in interaction.

Let us start with the first one. For this, we will move to more concrete settings of interaction of individual agents.

1.3.2 Complexity of interaction of diverse agents

Coming back to frameworks of reasoning about the power and abilities of agents, if we want to analyze how the power of agents is distributed among the participants, the ability to achieve *success* plays a crucial role. In game-like interaction processes, this can correspond to the ability to *win*. This then leads us to the concept of a *winning strategy*. Having a winning strategy means to have the power to ensure that the interaction process leads to success.

In the context of modern interaction, much of success of these processes relies on information and communication networks. Such networks can be unstable as connections can break down. Successfully traversing such an unstable network can then be seen as a game against an adversary opponent who causes the failure of certain connections. Analyzing this situation leads us to deciding whether there is a strategy to traverse the network to the final destination.

Considering the adversary player who cuts connections in the network, this player's objective can indeed also be seen as guiding the traversal to some particular destination by cutting all the alternative routes. In general, such a framework can be seen as a two-player game between one player who traverses the network and another one who cuts connections. In this context, our next research question arises.

> **Research Question 2** *What is the role of cooperation vs. competition in the complexity of interaction?*
>
> - *Does analyzing an interactive situation in general become easier if the participants cooperate?*

We address this question in Chapter 4 within the framework of *Sabotage Games*. Here, we will determine the complexity of deciding if a winning strategy exists, while we consider different variations of objectives of the players, distinguishing between cooperative and non-cooperative versions.

While Sabotage Games can be used as a model of information flow in networks with flaws, the information of the players themselves is not explicitly considered in this framework. However, the information of agents plays a crucial role in interaction. Intuitively, both the difficulty involved in interacting and the difficulty of reasoning about interactive situations or making predictions about them are affected by the underlying *information structure* of the interactive process. By this we mean the information that agents have about facts but also about the information that other agents have.

Similarly as for the complexity of deciding if a winning strategy exists, also for information, there are specific properties that play a special role in interaction. It is often crucial how information about facts and other agents' information is distributed among the agents. Do they have similar information about each other? In case of diverse agents, the question arises as to whether it is possible to induce information similarity of the agents by giving information to those who have less knowledge about the actual situation. The complexity of such questions could be investigated within fragments of existing epistemic modal logic frameworks. However, in order to proceed with our aim to investigate interaction itself rather than properties of logical theories of interaction, a complexity study of the algorithmic tasks that arise when analyzing information structures of agents seems more appropriate in the context of this dissertation.

Focusing on a purely semantic perspective on how information is modeled in modal epistemic logics, tasks involved in reasoning about agents' information boil down to comparing and manipulating the relational structures that represent the knowledge and beliefs that agents have. Intuitively speaking, there seems to be a difference in complexity of just reasoning about the information that one individual agent has about facts, and reasoning about higher-order information in a system with a large number of agents.

This leads us to the next research question.

Research Question 3 *Which parameters can make interaction difficult?*

- *How does the complexity of an interactive situation change when more participants enter the interaction or when we drop some simplifying assumptions on the participants themselves?*

We will address this question in Chapter 5 by focusing on what makes comparing and manipulating agents' information difficult. As we deal with concrete tasks, as opposed to the more abstract decisions problems such as satisfiability, many such tasks can be expected to be efficiently solvable. The questions that arise are the following.

What tasks about information comparison and manipulation are tractable?
When do they become intractable?

Here, particular interest should be paid to the effect of certain assumptions that are often made when formalizing knowledge, such as veridicality and positive and negative introspection. Can these assumptions make tasks which are in general intractable efficiently solvable?

These three research questions guide us from a high-level perspective on interaction using modal logics to an analysis of algorithmic tasks involved when reasoning about the information individual agents have. As our analysis is originally motivated by the need of a formal theory underlying real interaction, this leads us to the following last step in the analysis and thus back to interaction in the real world.

1.3.3 Complexity of interacting

It remains to determine whether results of a complexity theoretical analysis of interaction also have implications for real interactive processes. Intuitively, we would expect that it would be results about specific tasks which could have immediate implications for real interaction in the sense that human agents face similar tasks when interacting.

> **Research Question 4** *Finally, to what extent can we use a formal analysis of interactive processes to draw conclusions about the complexity of actual interaction?*
>
> - *Are there concrete examples of interactions in which participants actually encounter very high complexities which make it impossible for them to act?*

For addressing this question, we need a setting in which tasks arise which can be analyzed formally and which are tasks that real agents also have to face. Real (recreational) games are a natural choice here for us because recreational game playing is a very natural process, and moreover game rules control the interaction, which helps for a formal analysis.

If we want to analyze the complexity of tasks in games and be able to draw conclusions about the complexity that humans face in playing the game, it is important to focus the analysis on those tasks which players actually encounter in real play. Tasks that arise in sophisticated strategic reasoning such as computing if a winning strategy exists do not seem suitable candidates here as for recreational game playing solving such tasks is not necessary during the play. Of course such tasks do play a role also in recreational game playing but we cannot conclude that players are actually forced to complete such a task. Hence, we should focus on those tasks, which players are forced to face by the rules of the game. This leads us to the very basic task of performing a legal move. This task is at the very heart of game playing as the rules of a game actually "force" the players to face this task.

Having determined the kind of task for the complexity analysis, it remains to choose an appropriate class of recreational games to study. Here, the class of *inductive inference games* seems to be a natural candidate. In these games a designated player has to construct a rule about which moves of the other players are accepted and which are rejected. Then the other players get feedback for the moves they make and based on this inductively infer the secret rule. For our investigation, choosing an inductive inference game has the great advantage that it allows for a wide range of variations with respect to the complexity. There is a direct correspondence between the chosen secret rule and the difficulties that arise in the game. Inductive inference games exist of various kinds. For a formal analysis, a game in which both the secret rule and the moves of the players can be easily defined formally would be the best. This leads us to the card game *Eleusis* in which the secret rule is about sequences of cards and moves of players consist simply of placing cards in a sequence on the table. Additionally, Eleusis is also of interest for formal learning theory (cf. Gold (1967)), as the game illustrates learning scenarios in which general rules grammars are learned from inductively given data.

Fields to which this dissertation contributes. The results of this dissertation connect up between different areas of research, such as logic (in particular, modal logics for agency), computational complexity, game- and social choice theory, algorithmic game theory and formal learning theory.

1.4 Outline

This dissertation is structured as follows.

In Part I, we investigate the complexity of modal logics for reasoning about the abilities of groups of agents to achieve something.

Chapter 2 presents an example of a modal logic system for reasoning about interaction. More precisely, this chapter focuses on the concepts of coalitional power, actions and preferences. The logic we present is based on the cooperation logic with actions (CLA) developed by Sauro et al. (2006). We extend this framework with a fragment of the preference logic developed by van Benthem et al. (2007) in which agents have preferences over the states. Our resulting logic (cooperation logic with actions and preferences (CLA+P)) can distinguish between different ways to collectively achieve some results, not only with respect to how the results can be achieved but also with respect to whether or not it can be done in a way that leads to an improvement for individual agents. We analyze the satisfiability problem of CLA+P and show that it is decidable. We also show that it is EXPTIME-hard as so is the underlying action logic.

Then in Chapter 3 we take a broader perspective on cooperation logics, proposing a systematic analysis of how much expressive power and complexity is needed for interesting reasoning about coalitional power and preferences in different kinds of modal logics. We focus on three classes of modal logic frameworks; a simple framework based on transition systems in which transitions are labeled with coalitions, an action-based approach and a so called power-based approach. Moreover, we identify a range of notions and properties involving coalitional power and preferences and for each of these properties (ranging from the simple group ability to ensure that some fact is true, to more involved stability notions) we determine under what model theoretical operations it is invariant. This way, we can determine what kind of extended modal languages are appropriate for each of the classes of models for expressing the properties.

The languages determined by this method lead us to extended modal logics with certain complexities for model checking and satisfiability. Our methodology allows us to make precise what is the impact of design choices (e.g. whether to make coalitional power explicit in terms of actions) on how difficult it is to express certain kinds of properties. In addition to this analysis, we also study the relationship between action-based models for cooperation and power-based models. In particular, we focus on common assumptions on the powers of coalitions and discuss their interpretations on different models.

After this formal analysis of modal logics for reasoning about the interaction of groups of agents, Part II then analyzes the interaction between diverse agents.

In Chapter 4, we investigate a class of games played on a graph, called *Sabotage Games*. These games, originally introduced by van Benthem (2005), are two-player games in which one player (Runner) moves along the edges of the graph and the other player (Blocker) acts in the game by removing edges. In the standard version of this game, Runner tries to reach a goal state while Blocker tries to prevent this. Originally motivated by the interaction between Learner and Teacher in formal learning theory, we give two variations of the game: one in which Runner tries to avoid reaching a goal state (while Blocker tries to force him to move to a goal state) and a cooperative version in which both players want Runner to reach the goal. For each version, we analyze the complexity of deciding which player has a winning strategy. We show that the cooperative game is the easiest (NL-complete) while both non-cooperative games are more complex (PSPACE-complete). On the technical side, we discuss different methods for obtaining the complexity results and also point out which of them can lead to technical problems depending on which exact definition of a Sabotage Game is taken. Additionally, we consider a variation in the procedural rules of the games allowing Blocker to refrain from removing an edge. We show that this does not affect the winning abilities of the players.

Chapter 5 focuses on the concept of information, by analyzing different tasks that arise when reasoning about agents that are diverse in the sense that they have different information. This analysis takes place in the semantic structures of (epistemic) modal logic. Instead of investigating the complexity of such modal logic systems, this chapter analyzes concrete tasks such as determining whether two agents have similar information (about each other), and determining whether it is possible to give some information to one of the agents such that as a result both agents have similar information. In the complexity analysis, we pay particular attention to tracking where the border between easy (polynomially decidable) and hard (NP-hard) problems lies. We show that in general most tasks about deciding if agents have similar information (about each other) are tractable with some cases being trivial and others being among the hardest problems known to be tractable. For more dynamic tasks involving information change, the situation is different. We extend some hardness-results from graph theory in order to show that deciding if we can give some information to one agent so that his new information state will be of a certain shape is in general NP-complete. With the assumption of epistemic models being based on partitions however, in the single-agent case the problem turns out to be tractable.

Finally, Chapter 6 aims to answer the question as to what a formal complexity theoretical study can tell us about interaction in practice. This is done by investigating a particular card game: *The New Eleusis*, which is an inductive inference game. In this game, one designated player takes the role of *God* or *Nature* and constructs a secret rule about sequences of cards. Then the other players (*Scientists*) play cards in a sequence and each time get feedback from the first player about whether the card is accepted or rejected according to the secret rule. With this information, players try to inductively infer what might be the secret rule and test their hypotheses by playing certain cards. We examine the computational complexity of various tasks the players face during the game such as the task for the Scientist players of determining whether some rule might be the secret rule or the task of the first player of deciding whether a card is accepted or rejected. We show that if before the game players agree to only consider secret rules in a certain class, then the problems for the Scientists are tractable. Without these additional restrictions, the game however allows for the first player to construct a rule that is so complex that she even cannot give accurate feedback any more as to whether a card is accepted. This chapter thus shows that in the case of this game a complexity theoretical study indeed allows us to draw some conclusions for the actual play of the game. Based on this, we give some recommendations for adjusting the rules of the game.

Chapter 7 concludes this dissertation and presents some directions for further work.

Sources of the chapters. The logical system for explicit reasoning about groups of agents with preferences presented in Chapter 2 has been developed in Kurzen (2007). The complexity analysis given in this chapter is based on Kurzen (2009). Earlier version of it have been presented at the *8th Conference on Logic and the Foundations of Game and Decision Theory* (*LOFT 8*) and at the workshop *Logic and Intelligent Interaction* at the *European Summer School in Logic, Language and Information 2008* (*ESSLLI 2008*).

Chapter 3 is based on joint work with Cédric Dégremont. The analysis of expressive power and complexity of modal logics for cooperation presented in this chapter is a continuation of Dégremont and Kurzen (2009a), which has been presented at the *Workshop on Knowledge Representation for Agents and Multi-Agent Systems (KRAMAS 2008).*

Different previous versions of the work in Chapter 3 have been presented at the workshop *Logical Methods for Social Concepts* at *ESSLLI 2009*, at the *Second International Workshop on Logic, Rationality and Interaction (LORI-II)* (Dégremont and Kurzen 2009b) and at the *9th Conference on Logic and the Foundations of Game and Decision Theory (LOFT 9)* (Dégremont and Kurzen 2010).

Chapter 4 is based on joint work with Nina Gierasimzcuk and Fernando Velázquez-Quesada. The complexity analysis of Sabotage Games given in that chapter 4 originally started as a formal analysis of language learning, presented at the *10th Szklarska Poreba Workshop*. Follow-up versions of it, focusing on the interactive view on learning and teaching have been presented at the workshop *Formal Approaches to Multi-Agent Systems (FAMAS 2009)* (Gierasimczuk et al. 2009a) and at the *Second International Workshop on Logic, Rationality and Interaction (LORI-II)* (Gierasimczuk et al. 2009b). The technical results on the complexity of the different versions of Sabotage Games have also been presented at the workshop *G∀AMES 2009*.

The complexity analysis of tasks involved in reasoning about information of diverse agents as presented in Chapter 5 is based on joint work with Cédric Dégremont and Jakub Szymanik. It has been presented at the workshop *Reasoning about other minds: Logical and cognitive perspectives* at *TARK XIII* and appears in Dégremont et al. (2011).

Chapter 6 extends the complexity analysis of the recreational card game *Eleusis* given in Kurzen (2010), which has been presented at the *International Workshop on Logic and Philosophy of Knowledge, Communication and Action 2010 (LogKCA-10)*. Earlier versions of it have been presented at the student session of the *ICCL Summer School 2010* on *Cognitive Science, Computational Logic and Connectionism* and the workshop *G∀AMES 2010*. Joint work with Federico Sangati and Joel Uckelman has lead to a *wild ideas talk* on the game *Eleusis* at the first *ILLC colloquium* and to Federico Sangati's online implementation of the game (Sangati 2011).

Part I

Complexity of Reasoning about Strategic Ability using Modal Logic

Chapter 2

Reasoning about Cooperation, Actions and Preferences

We will now start our endeavor of investigating the complexity of interaction by first taking a modal logic perspective, motivated by our first research question.

> **Research Question 1** *What formal frameworks are best suited for reasoning about which concepts involved in interaction?*
>
> - *What should be the primitive notions a formal approach should be based on?*

We will give a formal model of interaction which allows us to reason about how groups can achieve results and what is the effect of actions with respect to the preferences of individual agents.

The approach presented in this chapter is conceptually motivated by situations of strategic interaction of the type mentioned in Section 1.3.1: e.g. the interaction between different competing companies.

Focusing primarily on their ability to achieve a certain profit, we can analyze the power of the competitors from an abstract perspective. However, for making predictions of what will actually happen in the interaction, it can be necessary to go into more details of how exactly a company can reach some goal. Here questions of the following form can arise:

> Can company C achieve some profit while making sure that for all of its current employees the resulting situation will be at least as good as the current one?

This chapter develops a modal logic framework for reasoning about the strategic abilities of individuals and coalitions in an explicit way. We will first conceptually motivate the design choices that are made when developing the logic and later investigate the computational consequences of the choices by analyzing the computational complexity of the resulting logical system.

This chapter contributes to the area of modal logics for reasoning about social concepts, with a particular focus on coalitional power (cf. e.g. Pauly (2002a)) and preferences (cf. e.g. Girard (2008)). On the technical side, this chapter can be seen as a case study of combining different existing logics and investigating the (computational) effects of this.

2.1 Cooperation, Actions and Preferences

Cooperation of agents plays a major role in many fields such as computer science, economics, politics, social sciences and philosophy. Agents can decide to cooperate and to form groups in order to share complementary resources or because as a group they can achieve something better than they can do individually.

When analyzing interactive situations involving multiple agents, we are interested in what results agents can achieve – individually or together as groups. There can be many ways how agents can achieve some result. They can differ significantly, e.g. with respect to their feasibility, costs or side-effects. Hence, it is not only relevant what results groups of agents can achieve but also *how* exactly they can do so. In other words, plans and actions also play a central role if we want to reason about cooperation in an explicit way. However, cooperative ability of agents expressed only in terms of results and actions that lead to these results does not tell us *why* a group of agents would actually decide to achieve a certain result. For this, we also need to take into account the preferences based on which the agents decide what to do. Summarizing these initial motivations, we can say that in interactive situations, the following three questions are of interest:

- ***What*** *results can groups of agents achieve?*
- ***How*** *can they achieve something?*
- ***Why*** *would they want to achieve a certain result?*

The above considerations show that when reasoning about the strategic abilities of agents, the concepts of coalitional power, actions/plans and preferences play a major role and are moreover tightly connected. Thus, we argue that a formal theory for reasoning about agents' cooperative abilities in an *explicit* way should also take into account actions/plans of agents and their preferences.

Modal logics have been used to develop formal models for reasoning about each of these aspects – mostly separately. Coalitional power has mainly been investigated within the frameworks of *Alternating-time Temporal Logic* (ATL) (Alur et al. 1998), *Coalition Logic* (CL) (Pauly 2002a) and their extensions.

The models of $\mathtt{CL}$ are neighborhood models defined as follows.

Definition 2.1 ($\mathtt{CL}$ Model) A $\mathtt{CL}$-model is a pair $((\mathtt{N}, W, E), \mathsf{V})$ where $\mathtt{N}$ is a set of agents, $W \neq \emptyset$ is a set of states, $E : W \to (\wp(\mathtt{N}) \to \wp(\wp(W)))$ is called an effectivity structure. It satisfies the conditions of **playability**:

- Liveness: $\forall C \subseteq \mathtt{N} : \emptyset \notin E(C)$,
- Termination: $\forall C \subseteq \mathtt{N} : W \in E(C)$,
- $\mathtt{N}$-maximality. $\forall X \subseteq W : (W \setminus X \notin E(\emptyset) \Rightarrow X \in E(\mathtt{N}))$,
- Outcome monotonicity. $\forall X \subseteq X' \subseteq W, C \subseteq \mathtt{N} : (X \in E(C) \Rightarrow X' \in E(C))$,
- Superadditivity. $\forall X_1, X_2 \subseteq W, C_1, C_2 \subseteq \mathtt{N} : ((C_1 \cap C_2 = \emptyset \ \& \ X_1 \in E(C_1) \ \& \ X_2 \in E(C_2)) \Rightarrow X_1 \cap X_2 \in E(C_1 \cup C_2))$.

$\mathsf{V} : \textsc{prop} \to \wp(W)$ is a propositional valuation function. ◀

The language $\mathcal{L}_{\mathtt{CL}}$ of $\mathtt{CL}$ is a standard modal language with a modality $\langle\!\![C\rangle\!\!]$ for each $C \subseteq \mathtt{N}$. The intended meaning of $\langle\!\![C\rangle\!\!]\varphi$ is "coalition C has the power to achieve that φ". First, we briefly recall the semantics of $\mathtt{CL}$:

$$M, w \vDash \langle\!\![C\rangle\!\!]\varphi \text{ iff } [\![\varphi]\!]_M \in E(w)(C),$$

where $[\![\varphi]\!]_M$ denotes the set of states in the model M that satisfy φ. For the details we refer the reader to Pauly (2002a).

More recently, there have been some attempts to develop logics for reasoning about coalitional power that also take into account either agents' preferences or actions. One group of such logics looks at cooperation from the perspective of cooperative games (Ågotnes et al. 2007). In a non-cooperative setting preferences and strategic abilities have been considered by van Otterloo et al. (2004). Another path that has been taken in order to make coalitional power more explicit is to combine cooperation logics with (fragments of) action logics (Sauro et al. 2006; Borgo 2007; Walther et al. 2007).

In this chapter, a sound and complete modal logic for reasoning about cooperation, actions and preferences (CLA+P) is developed, which is obtained by combining the cooperation logic with actions CLA by Sauro et al. (2006) with a preference logic (van Benthem et al. 2005, 2007). We analyze the logic's expressivity and computational complexity.

2.2 Cooperation Logic with Actions (CLA)

In this section, we briefly present the cooperation logic with actions (CLA) which will be extended in the next section by combining it with a preference logic. The idea of CLA is to make coalitional power explicit by expressing it in terms of the ability to perform actions instead of expressing it directly in terms of the ability to achieve certain outcomes. The definitions we present here in Section 2.2 are equivalent to those of Sauro et al. (2006). CLA is a modular modal logic, consisting of an environment module for reasoning about actions and their effects, and an agents module for reasoning about agents' abilities to perform actions. By combining both modules, a framework is obtained in which cooperative ability can be made more explicit as the cooperative ability to achieve results comes from the ability to ensure that actions of certain types take place which have the effect of making the result happen.

The environment is modeled as a transition system whose edges are labeled with sets of atomic actions.

Definition 2.2 (Environment model) An environment model is a set-labeled transition system

$$E = \langle W, Ac, (\rightarrow)_{A \subseteq Ac}, \mathsf{V} \rangle.$$

W is a set of states, Ac is a finite set of atomic actions, $\rightarrow_A \subseteq W \times W$ and V is a propositional valuation. Each $\rightarrow_A$ is required to be serial, i.e., for every $w \in W$ and $A \subseteq Ac$, there is at least one $v \in W$ such that $w \rightarrow_A v$. ◀

The intuition behind $w \rightarrow_A v$ is that if in w all and only the actions in A are performed concurrently, then the next state can be v (note that transition system can be nondeterministic, and there can be several states v' such that $w \rightarrow_A v'$.

Then a modal language is defined with modalities $[\alpha]$, for α being a propositional formula built from atomic actions. The intended meaning of $[\alpha]\varphi$ is that every transition $\rightarrow_A$ such that $A \models \alpha$ (using the satisfaction relation of propositional logic[1]) leads to a φ-state:

$$E, w \models [\alpha]\varphi \quad \text{iff} \quad \forall A \subseteq Ac, v \in W : \text{if } A \models \alpha \text{ and } w \rightarrow_A v \text{ then } E, v \models \varphi.$$

In this dissertation, we do not go into the underlying philosophy of actions but refer the reader to Broersen (2003) for a detailed discussion of modal action logics. The restriction to a finite set of actions is reasonable for modeling many concrete situations and also ensures that we have a finite axiomatization.

An environment logic Λ^E is developed, which is sound and complete (Sauro et al. 2006). It contains seriality axioms and the **K** axiom for each modality $[\alpha]$, for α being consistent. The environment logic can then be used for reasoning about the effects of concurrent actions.

Then an agents module is developed for reasoning about the ability of (groups of) agents to act. Each agent is assigned a set of atomic actions and each coalition is assigned the set of actions its members can perform.

[1] That is, $A \models a$ iff $a \in A$, $A \models \neg\alpha$ iff $A \not\models \alpha$, and $A \models \alpha \wedge \beta$ iff $A \models \alpha$ and $A \models \beta$.

Definition 2.3 (Agents model) An agents model is a triple $\langle \mathtt{N}, Ac, \mathsf{act}\rangle$, where $\mathtt{N}$ is a finite set of agents, Ac is a finite set of atomic actions and act is a function $\mathsf{act} : \mathtt{N} \to \wp(Ac)$ such that $\bigcup_{i\in\mathtt{N}} \mathsf{act}(i) = Ac$. For $C \subseteq \mathtt{N}$, define $\mathsf{act}(C) := \bigcup_{i\in C} \mathsf{act}(i)$. ◀

We are also interested in agents' abilities to force more complex actions. A language is developed with expressions $\langle\!\langle C \rangle\!\rangle \alpha$, meaning that the group C can force that a concurrent action is performed that satisfies α. This means that C can perform some set of atomic actions such that no matter what the other agents do, the resulting set of actions satisfies α.
$\langle \mathtt{N}, Ac, \mathsf{act}\rangle \models \langle\!\langle C \rangle\!\rangle \alpha \quad \text{iff} \quad \exists A \subseteq \mathsf{act}(C) : \forall B \subseteq \mathsf{act}(\mathtt{N} \setminus C) : A \cup B \models \alpha.$

Then a cooperation logic for actions is developed, which is very much in the style of Coalition Logic (Pauly 2002a) – the main difference being that it is concerned with the cooperative ability to force *actions* of certain types.

Definition 2.4 (Coalition Logic for Actions) The coalition logic for actions Λ^A is defined to be the logic derived from the following set of axioms, with modus ponens as rule of inference.

1. $\langle\!\langle C \rangle\!\rangle \top$, for all $C \subseteq \mathtt{N}$,
2. $\langle\!\langle C \rangle\!\rangle \alpha \to \neg\langle\!\langle \mathtt{N} \setminus C \rangle\!\rangle \neg\alpha$,
3. $\langle\!\langle C \rangle\!\rangle \alpha \to \langle\!\langle C \rangle\!\rangle \beta$ if $\vdash \alpha \to \beta$ in propositional logic,
4. $\langle\!\langle C \rangle\!\rangle a \to \bigvee_{i\in C} \langle\!\langle \{i\} \rangle\!\rangle a$ for all $C \subseteq \mathtt{N}$ and atomic $a \in Ac$,
5. $(\langle\!\langle C_1 \rangle\!\rangle \alpha \wedge \langle\!\langle C_2 \rangle\!\rangle \beta) \to \langle\!\langle C_1 \cup C_2 \rangle\!\rangle (\alpha \wedge \beta)$, for $C_1 \cap C_2 = \emptyset$,
6. $(\langle\!\langle C \rangle\!\rangle \alpha \wedge \langle\!\langle C \rangle\!\rangle \beta) \to \langle\!\langle C \rangle\!\rangle (\alpha \wedge \beta)$ if α and β have no common atomic actions,
7. $\langle\!\langle C \rangle\!\rangle \neg a \to \langle\!\langle C \rangle\!\rangle a$ for atomic $a \in Ac$,
8. $\langle\!\langle C \rangle\!\rangle \alpha \to \bigvee\{\langle\!\langle C \rangle\!\rangle \bigwedge \Psi \mid \Psi$ is a set of literals such that $\bigwedge \Psi \to \alpha\}$.

◀

Let us briefly discuss some of the axioms. Axiom 2 says that the abilities of coalitions to force actions have to be consistent. Axiom 4 says that if a coalition can perform some atomic action, this must be because one of the individuals in the coalition can perform it. Axiom 5 says how disjoint coalitions can join forces. Axiom 7 says that if a coalition can force that an atomic action won't be performed it must be the case that the coalition itself could perform this action. This has to do with the fact that for every atomic action there has to be an agent who can perform it. Axiom 8 makes explicit how complex action types can be forced.

The coalition logic for actions is sound and complete with respect to the class of agents models (Sauro et al. 2006).

Next, agents are introduced as actors into the environment. This is done by combining the environment models with the agents models. Then the agents can perform actions which have the effect of changing the current state of the environment.

Definition 2.5 (Multi-agent system) A multi-agent system (MaS) is a tuple

$$M = \langle W, Ac, (\to)_{A \subseteq Ac}, \mathsf{V}, \mathtt{N}, \mathsf{act} \rangle,$$

where $\langle W, Ac, (\to)_{A \subseteq Ac}, \mathsf{V} \rangle$ is an environment model and $\langle Ac, \mathtt{N}, \mathsf{act} \rangle$ an agents model. ◀

In these models, we can reason about what states of affairs groups can achieve by performing certain actions. The corresponding language contains all expressions of the logics previously defined in this chapter and expressions for saying that a group has the power to achieve φ, which means that the group can make the system move into a state where φ is true.

Definition 2.6 (Language for MaS) The language for multi-agent systems $\mathcal{L}_{cla}$ is generated by the following grammar:

$$\varphi ::= \quad p \mid \varphi \wedge \varphi \mid \neg\varphi \mid [\alpha]\varphi \mid \langle\!\![C]\!\!\rangle \alpha \mid \langle\!\![C]\!\!\rangle \varphi$$

for $C \subseteq \mathtt{N}$ and α being an action expression. ◀

$\langle\!\![C]\!\!\rangle \varphi$ means that C can force φ, i.e., C can perform a set of actions such that no matter what the other agents do, the system moves to a φ-state.

$$\begin{array}{lll} M, w \models \langle\!\![C]\!\!\rangle \varphi & \text{iff} & \exists A \subseteq \mathsf{act}(C) \text{ such that } \forall B \subseteq \mathsf{act}(\mathtt{N} \setminus C), v \in W : \\ & & \text{if } w \to_{A \cup B} v, \text{ then } M, v \models \varphi. \end{array}$$

A complete axiomatization is obtained by combining the environment logic and the coalition logic for actions by adding two interaction axioms.

Definition 2.7 (Cooperation Logic with Actions) The cooperation logic with actions Λ^{CLA} combines the environment logic Λ^{E} and the coalition logic for actions Λ^{A} by adding

1. $(\langle\!\![C]\!\!\rangle \alpha \wedge [\alpha]\varphi) \to \langle\!\![C]\!\!\rangle \varphi$,

2. $\langle\!\![C]\!\!\rangle \varphi \to \bigvee \{ \langle\!\![C]\!\!\rangle \alpha \wedge [\alpha]\varphi \mid \alpha$ is the conjunction of a set of atomic actions or their negations$\}$. ◀

Note how Axiom 2 makes coalitional power explicit: If a coalition can force φ, then there has to be a concrete plan as to which atomic actions the members of C perform such that the resulting action is guaranteed to lead to a state where φ holds.

CLA provides us with a formal framework for reasoning about what states of affairs groups of agents can achieve and how they can do so. A group C being able to force φ means that there has to be some set of actions $A \subseteq \mathsf{act}(C)$ such that all transitions of type

$$\bigwedge \Phi(A, C) := \bigwedge(A \cup \{\neg a | a \in (\mathsf{act}(C) \setminus A), a \notin \mathsf{act}(\mathtt{N} \setminus C)\}$$

lead to a state where φ holds. For $A \subseteq \mathsf{act}(C)$, the action expression $\bigwedge \Phi(A, C)$ describes the collective actions that can occur when coalition C performs all and only the atomic actions in A.

For a detailed discussion of CLA, the reader is referred to Sauro et al. (2006). Now, we proceed by adding preferences to CLA.

2.3 Cooperation Logic with Actions and Preferences

In this section, a logic for reasoning about cooperation, actions and preferences is developed. This is done by adding a preference logic to CLA. As the main focus of this chapter is on the *complexity* of this logical system, for further technical details of the logic, the reader is referred to Kurzen (2007).

2.3.1 Preference Logic

There are various ways how preferences of agents can be added to a logic for cooperation and actions. They could e.g. range over the actions that the agents can perform. Alternatively, we can think of each agent having preferences over the set of successor states of the current state.

In the current work, we consider preferences of individual agents ranging over the states of the environment. This is reasonable since by performing actions the agents can change the current state of the environment, and the preferences over those states can be seen as the base of how the agents decide how to act. Such a preference relation can also be lifted in several ways to one over formulas (van Benthem et al. 2005, 2007).

Definition 2.8 (Preference model (van Benthem et al. 2005)) A preference model is a tuple

$$M^P = \langle W, \mathtt{N}, \{\preceq_i\}_{i \in \mathtt{N}}, \mathsf{V} \rangle,$$

where W is a set of states, $\mathtt{N}$ is a set of agents, for each $i \in \mathtt{N}, \preceq_i \subseteq W \times W$ is reflexive and transitive, and V is a propositional valuation. ◂

We use a fragment of the preference language developed by van Benthem et al. (2007). It has strict and non-strict preference modalities.

Definition 2.9 (Preference Language) Given a set of propositional variables and a finite set of agents N, define the preference language $\mathcal{L}_p$ to be the language generated by the following syntax:

$$\varphi := \ p \mid \neg\varphi \mid \varphi \vee \varphi \mid \Diamond^{\leq_i}\varphi \mid \Diamond^{<_i}\varphi. \qquad \blacktriangleleft$$

$\Diamond^{\leq_i}\varphi$ says that there is a state satisfying φ that agent i considers to be at least as good as the current one. The semantics is defined as follows.

$$\begin{array}{lll} M^P, w \models \Diamond^{\leq_i}\varphi & \text{iff} & \exists v : w \leq_i v \text{ and } M^P, v \models \varphi. \\ M^P, w \models \Diamond^{<_i}\varphi & \text{iff} & \exists v : w \leq_i v, v \not\leq_i w \text{ and } M^P, v \models \varphi. \end{array}$$

The preference relation $\leq$ is a preorder and $<$ is its largest irreflexive subrelation. Hence, the following axiomatization.

Definition 2.10 (Preference Logic Λ^P) For a given set of agents N, let Λ^P be the logic generated by the following axioms for each $i \in$ N: For $\Diamond^{\leq_i}$ and $\Diamond^{<_i}$, we have **K** and for $\Diamond^{\leq_i}$ also reflexivity and transitivity axioms. Moreover, there are four interaction axioms specifying the relationship between strict and non-strict preferences:

1. $\Diamond^{<_i}\varphi \rightarrow \Diamond^{\leq_i}\varphi$,
2. $\Diamond^{\leq_i}\Diamond^{<}\varphi \rightarrow \Diamond^{<_i}\varphi$,
3. $\Diamond^{<_i}\Diamond^{\leq_i}\varphi \rightarrow \Diamond^{<_i}\varphi$,
4. $\varphi \wedge \Diamond^{\leq_i}\psi \rightarrow (\Diamond^{<_i}\psi \vee \Diamond^{\leq_i}(\psi \wedge \Diamond^{\leq_i}\varphi))$.

The inference rules are modus ponens, necessitation and substitution. $\blacktriangleleft$

Note that transitivity for $\Diamond^{<_i}$ follows. We can show soundness and completeness. The *bulldozing* technique (Blackburn et al. 2001) is used to deal with $<$. For details, we refer to van Benthem et al. (2007).

Theorem 2.11 *Λ^P is sound and complete with respect to the class of preference models.*

Proof. Follows from Theorem 3.9 of van Benthem et al. (2007). ■

Our main motivations for choosing this preference logic is its ability to distinguish between weak and strict preference which plays a major role in many concepts for reasoning about interaction in multi-agent systems. This makes the logic quite powerful but due to its simplicity it still has the modal character and talks about preferences from a local forwards looking perspective. A modality for talking about states being *at least as bad* would have increased the expressive power but we would have lost the local perspective since this would have resulted in a global existential modality with respect to all comparable states.

2.3.2 Environment Logic with Preferences

As an intermediate step towards a logic for reasoning about cooperation, actions and preferences, we first combine the preference logic and the environment logic. The two models are combined by identifying their sets of states. Then the preferences of the agents range over the states of the environment. In such a system, the agents cannot act in the environment, but they can rather be seen as observers that observe the environment from the outside and have preferences over its states.

Definition 2.12 (Environment with Preferences) An environment model with preferences is a tuple

$$E^{\preceq} = \langle W, Ac, (\rightarrow)_{A \subseteq Ac}, \{\preceq_i\}_{i \in \mathtt{N}}, \mathsf{V} \rangle,$$

where $\langle W, Ac, (\rightarrow)_{A \subseteq Ac}, \{\preceq_i\}_{i \in \mathtt{N}}, \mathsf{V} \rangle$ is an environment model and $\langle W, \mathtt{N}, \{\preceq_i\}_{i \in \mathtt{N}}, \mathsf{V} \rangle$ is a preference model. ◀

We combine the languages for the environment and the preferences and add expressions for saying that "every state accessible by an α transition is (strictly) preferred by agent i over the current state". The main motivation for adding such expressions is that our aim is to be able to express properties inspired by game theoretical solution concepts, which very often involve statements saying that groups can(not) achieve an outcome better for (some of) its members. Since we want to make explicit by which actions groups can(not) achieve improvements we introduce this expression saying that actions of type α are guaranteed to lead to an improvement.

Notation We will write the symbol $\lhd$ in statements that hold for both $\preceq$ and $\prec$, each uniformly substituted for $\lhd$. ◀

Definition 2.13 (Environment Language with Preferences) The language $\mathcal{L}_{ep}$ contains all expressions of the environment language and the preference language and additionally formulas of the forms $\alpha \subseteq\preceq_i$ and $\alpha \subseteq\prec_i$, for α being an action expression.

Boolean combinations and expressions of the previously defined languages are interpreted in the standard way. For the newly introduced expressions, we have:

$E^{\preceq}, w \models \alpha \subseteq\lhd_i$ iff $\forall A \subseteq Ac, v \in W$: if $w \rightarrow_A v$ and $A \models \alpha$ then $w \lhd_i v$. ◀

Expressions of the form $\alpha \subseteq\lhd_i$ cannot be defined just in terms of the preference language and the environment language. To see this, note that $\alpha \subseteq\preceq_i$ says that for every state accessible by an α-transition it holds that *this same state* is accessible by $\preceq_i$. Thus, we would have to be able to refer to particular states. Therefore, we add two inference rules for deriving the newly introduced expressions.

$$(\text{PREF-ACT})\ \frac{\Box^{\leq_i}\varphi \to [\alpha]\varphi}{\alpha \subseteq \leq_i} \qquad (\text{STRICT PREF-ACT})\ \frac{\Box^{<_i}\varphi \to [\alpha]\varphi}{\alpha \subseteq <_i}$$

In order to obtain a complete axiomatization, two axioms are added which correspond to the converse of the inference rules.

Theorem 2.14 *Let Λ^{EP} be the logic generated by all axioms of the environment logic Λ^{E}, all axioms of the preference logic Λ^{P}, and*

1. $\alpha \subseteq \leq_i \to (\Box^{\leq_i}\varphi \to [\alpha]\varphi)$,
2. $\alpha \subseteq <_i \to (\Box^{<_i}\varphi \to [\alpha]\varphi)$.

The inference rules are modus ponens, substitution, necessitation PREF-ACT and STRICT PREF-ACT. Then Λ^{EP} is sound and complete with respect to the class of environment models with preferences.

Proof. Soundness is straightforward. We sketch the proof for completeness. Details can be found in Kurzen (2007). The canonical model is constructed in the usual way, with the maximally consistent sets being closed also under the new inference rules. A truth lemma can then be shown by induction on φ. The only interesting cases are those for $\alpha \subseteq \leq_i$ and $\alpha \subseteq <_i$. The others follow from completeness of the sublogics. We give a sketch for $\alpha \subseteq \leq_i$. We have to show that for every maximally consistent set Σ it holds that $\alpha \subseteq \leq_i \in \Sigma$ iff $\Sigma \models \alpha \subseteq \leq_i$. The left-to-right direction uses Axiom 1. The other direction first uses the fact that $\alpha \subseteq \leq_i$ characterizes the property that states accessible by α-transitions are a subset of the states accessible by $\leq_i$. Then we can use the closure under the rule PREF-ACT to conclude that $\alpha \subseteq \leq_i \in \Sigma$. The case of $\alpha \subseteq <_i$ is analogous. ■

In the environment logic with preferences, the performance of concurrent actions changes the current state of the system also with respect to the agents' "happiness": A transition from one state to another can also correspond to a transition up or down in the preference orderings of the agents.

2.3.3 Cooperation Logic with Actions and Preferences

Now, agents are introduced as actors by combining the environment models with preferences with agents models. The resulting model is then called a multi-agent system with preferences (henceforth MaSP).

Definition 2.15 (Multi-agent system with preferences) A multi-agent system with preferences (MaSP) $M^{\leq}$ is a tuple

$$M^{\leq} = \langle W, Ac, (\to)_{A \subseteq Ac}, \mathtt{N}, \mathsf{act}, \{\leq_i\}_{i \in \mathtt{N}}, \mathsf{V} \rangle,$$

where $\langle W, Ac, (\to)_{A \subseteq Ac}, \mathsf{V}, \mathtt{N}, \mathsf{act} \rangle$ is a MaS, $\langle W, \mathtt{N}, \{\leq_i\}_{i \in \mathtt{N}}, \mathsf{V} \rangle$ is a preference model and $\langle W, Ac, (\to)_{A \subseteq Ac}, \{\leq_i\}_{i \in \mathtt{N}}, \mathsf{V} \rangle$ is an environment with preferences. ◄

Remark 2.16 Note that given a deterministic MaSP in which each preference relation $\preceq_i$ is total, we can consider each state w as having a strategic game (Osborne and Rubinstein 1994) $\mathcal{G}_s$ attached to it.

$$\mathcal{G}_w = \langle \mathtt{N}, (\wp(\mathsf{act}(i)))_{i \in \mathtt{N}}, (\precsim_i)_{i \in \mathtt{N}} \rangle,$$

$\times_{i=1}^{n} A_i \precsim_i \times_{i=1}^{n} A'_i$ iff $v \preceq_i v'$ for $w \rightarrow_{\bigcup_{i \in \mathtt{N}} A_i} v$ and $w \rightarrow_{\bigcup_{i \in \mathtt{N}} A'_i} v'$. ◀

For talking about the cooperative ability of agents with respect to preferences, we introduce two expressions saying that a group can force the system to move into a φ-state that some agent (strictly) prefers.

Definition 2.17 (Language $\mathcal{L}_{cla+p}$) The language $\mathcal{L}_{cla+p}$ extends $\mathcal{L}_{cla}$ by formulas of the form

$$\Diamond^{\preceq_i}\varphi \mid \Diamond^{\prec_i}\varphi \mid \alpha \subseteq \preceq_i \ \mid \alpha \subseteq \prec_i \ \mid \langle\!\![C^{\preceq_i}]\!\!\rangle \varphi \mid \langle\!\![C^{\prec_i}]\!\!\rangle \varphi.$$

The first four expressions are interpreted as in the environment logic with preferences and for the last two we have the following.

$$M^{\preceq}, w \models \langle\!\![C^{\lhd_i}]\!\!\rangle \varphi \quad \text{iff} \quad \begin{array}{l} \exists A \subseteq \mathsf{act}(C) \colon \forall B \subseteq \mathsf{act}(\mathtt{N} \setminus C), v \in W : \text{if } w \rightarrow_{A \cup B} v, \\ \text{then } M^{\preceq}, v \models \varphi \text{ and } w \lhd_i v. \end{array}$$ ◀

Let us now look at how coalitional power to achieve an improvement for an agent is made explicit in CLA+P. We can show that $\langle\!\![C^{\lhd_i}]\!\!\rangle \varphi$ is equivalent to the existence of an action expression α that C can force and that has the property that all transitions of type α are guaranteed to lead to a φ-state preferred by agent i.

Observation 2.18 *Given a MaSP $M^{\preceq}$ and a state w of its environment,*

$$M^{\preceq}, w \models \langle\!\![C^{\lhd_i}]\!\!\rangle \varphi \quad \textit{iff} \quad \begin{array}{l} \textit{there exists an action expression } \alpha \textit{ such that } M^{\preceq}, w \models \\ \langle\!\![\mathrm{C}]\!\!\rangle \alpha \wedge [\alpha]\varphi \wedge (\alpha \subseteq \lhd_i). \end{array}$$

Proof. Analogous to that of Observation 14 of Sauro et al. (2006). For the left-to-right direction, use the action expression

$$\bigwedge \Phi(A, C)$$

with $\bigwedge \Phi(A, C) := \bigwedge (A \cup \{\neg a | a \in (\mathsf{act}(C) \setminus A), a \notin \mathsf{act}(\mathtt{N} \setminus C)\})$ and A being a joint action of C that is the 'witness' of $\langle\!\![C^{\lhd_i}]\!\!\rangle \varphi$. ■

Now we need axioms establishing a relationship between the newly added formulas and the expressions of the sublogics.

Definition 2.19 (Cooperation Logic with Actions and Preferences) Λ^{CLA+P} is defined to be the smallest logic generated by the axioms of the cooperation logic with actions, the environment logic with preferences and

1. $(\langle\![C]\!\rangle\alpha \wedge [\alpha]\varphi \wedge (\alpha \subseteq\leq_i)) \rightarrow \langle\![C^{\leq_i}]\!\rangle\varphi$,
2. $(\langle\![C]\!\rangle\alpha \wedge [\alpha]\varphi \wedge (\alpha \subseteq<_i)) \rightarrow \langle\![C^{<_i}]\!\rangle\varphi$,
3. $\langle\![C^{\leq_i}]\!\rangle\varphi \rightarrow \bigvee\{\langle\![C]\!\rangle\alpha \wedge [\alpha]\varphi \wedge (\alpha \subseteq\leq_i) | \alpha$ is a conjunction of action literals$\}$,
4. $\langle\![C^{<_i}]\!\rangle\varphi \rightarrow \bigvee\{\langle\![C]\!\rangle\alpha \wedge [\alpha]\varphi \wedge (\alpha \subseteq<_i) | \alpha$ is a conjunction of action literals$\}$.

The inference rules are modus ponens, necessitation for action modalities and preference modalities ($\Box^{\leq_i}, \Box^{<_i}$), substitution of logical equivalents, PREF − ACT and STRICT PREF − ACT. ◀

The axioms specify the interaction between the coalitional power, actions and preferences. The last two axioms make explicit how a group can achieve that the system moves into a state preferred by an agent, where φ is true.

Theorem 2.20 *The logic Λ^{CLA+P} is sound and complete with respect to the class of MaSP's.*

Proof. Soundness of the axioms is straightforward. We sketch the proof of completeness. The details can be found in Kurzen (2007). The canonical model is constructed in the standard way. Then a truth lemma is shown, where the interesting cases are $\langle\![C^{\leq_i}]\!\rangle\varphi$ and $\langle\![C^{<_i}]\!\rangle\varphi$. We sketch the case of $\langle\![C^{\leq_i}]\!\rangle\varphi$. Assume that for a maximally consistent set Σ we have that $\Sigma \models \langle\![C^{\leq_i}]\!\rangle\varphi$. Then by Observation 2.18, there has to be an action expression α such that $\Sigma \models \langle\![C]\!\rangle\alpha$, $\Sigma \models [\alpha]\varphi$ and $\Sigma \models \alpha \subseteq\leq_i$. Using the induction hypothesis, the maximality of Σ and Axiom 1, it follows that $\langle\![C^{\leq_i}]\!\rangle\varphi \in \Sigma$. For the other direction, let $\langle\![C^{\leq_i}]\!\rangle\varphi \in \Sigma$. Then by maximality of Σ and Axiom 3, there is a set of action literals $\mathcal{A}$ such that $[\bigwedge \mathcal{A}]\varphi \in \Sigma \bigwedge \mathcal{A} \subseteq\leq_i \in \Sigma$ and $[\bigwedge \mathcal{A}]\varphi \in \Sigma$. Using the induction hypothesis and the previous cases, we can then apply Observation 2.18 and conclude that $\Sigma \models \langle\![C^{\leq_i}]\!\rangle\varphi$. ■

After this overview of the technical specifications of the logic, we will now look at what the combination of explicit coalitional power in terms of actions and preferences allows us to express.

2.3.4 Expressivity of CLA+P

We now show that in CLA+P, we can express some concepts relevant for reasoning about game-like interaction in multi-agent systems.

Stability. Given a MaSP, $M^{\preceq} = \langle W, Ac, (\rightarrow)_{A \subseteq Ac}, \mathtt{N}, \mathsf{act}, \{\preceq_i\}_{i \in \mathtt{N}}, \mathsf{V}\rangle$, the following formula characterizes the states that are *individually stable (group stable)*, i.e., no individual (group) has the power to achieve a strict improvement (for all its members).

$$\psi_{ind.\ stable} := \bigwedge_{i \in \mathtt{N}} \neg \langle\!\![\{i\}^{\prec_i}]\!\!\rangle \top .$$

$$\psi_{gr.\ stable} := \bigwedge_{C \subseteq \mathtt{N}} \bigwedge_{A \subseteq \mathsf{act}(C)} \left(\bigvee_{i \in C} \neg \left(\left(\bigwedge \Phi(A, C) \right) \subseteq \prec_i \right) \right).$$

Analogously, we can also express a stronger form of stability by replacing $\preceq$ by $\prec$, which then means that no (group of) agent(s) can achieve that the system moves to a state at least as good for all its members.

Dictatorship. We can express that an agent d is a (strong) dictator in the sense that coalitions can only achieve what d (strictly) prefers.

$$\psi_{d = dict.} := \bigwedge_{C \subseteq \mathtt{N}} \bigwedge_{A \subseteq \mathsf{act}(C)} \left(\left(\bigwedge \Phi(C, A) \right) \subseteq \lhd_d \right).$$

Then, we can also say that there is *no* (strong) dictator:

$$\psi_{no\ dict.} := \bigwedge_{i \in \mathtt{N}} \neg \left(\bigwedge_{C \subseteq \mathtt{N}} \bigwedge_{A \subseteq \mathsf{act}(C)} \left(\left(\bigwedge \Phi(C, A) \right) \subseteq \lhd_i \right) \right).$$

Enforcing Unanimity. In some situations we might want to impose the condition on a MaSP that groups should only be able to achieve something if they can do so by making all its members happy:

$$\bigwedge_{C \subseteq \mathtt{N}} \left(\langle\!\![C]\!\!\rangle \varphi \rightarrow \left(\bigvee_{A \subseteq \mathsf{act}(C)} \left(\bigwedge_{i \in C} \left(\left(\bigwedge \Phi(A, C) \right) \subseteq \prec_i \right) \wedge \left[\bigwedge \Phi(A, C) \right] \varphi \right) \right) \right).$$

Note that the length of the last four formulas is exponential in the number of agents (and atomic actions).

2.3.5 CLA+P and Coalition Logic

Let us now briefly discuss the relation between CLA+P and Pauly's Coalition Logic (CL) in order to illustrate how CLA+P builds upon existing frameworks for reasoning about coalitional power and how exactly the underlying actions that are only implicitly represented in the semantics of CL are made explicit.

Given a fixed set of agents $\mathtt{N}$, a coalition model $M = \langle (W, \mathsf{E}), \mathsf{V}\rangle$ with W being a set of states, $\mathsf{E} : W \to (\wp(\mathtt{N}) \to \wp(\wp(W))$ being a *truly* playable[2] effectivity function and V being a propositional valuation, we can we use Theorem 7 of Goranko et al. (2011) and obtain a corresponding *game frame*, i.e., each state has an associated strategic game form in which each outcome corresponds to some accessible state. Looking back at Remark 2.16, it is now easy to see how we can construct a corresponding MaS: We take the same set of states and add actions for each of the strategies in the attached games and define the accessibility relation in accordance with the outcome function.

If we add preferences to the game forms in the game frame we obtained from the coalition model, then we can transform it into a MaSP in an analogous way.

This shows that the framework of CLA+P is a natural way to make coalitional power as modeled in CL and its extensions more explicit.

2.4 Complexity of CLA+P

In this section, we analyze the complexity of SAT of CLA+P. We will first show decidability before studying lower bounds.

2.4.1 Decidability of CLA+P

We show that SAT of CLA+P is decidable. The first step is to show that only a restricted class of models of CLA+P needs to be considered.

We start by looking at how we can restrict the class of models with respect to the set of agents. Let $\mathtt{N}(\varphi)$ denote the set of agents occurring in φ. Now, we ask: Is every satisfiable φ also satisfiable in a MaSP with set of agents $\mathtt{N}(\varphi)$? In Coalition Logic, the answer is negative: the formula $\psi = \neg\langle\!\langle\{1\}\rangle\!\rangle p \wedge \neg\langle\!\langle\{1\}\rangle\!\rangle q \wedge \langle\!\langle\{1\}\rangle\!\rangle(p \vee q)$ is only satisfiable in models with at least two agents (Pauly 2002b). However, as in CLA+P the environment models can be nondeterministic, here ψ can indeed be satisfied in a model with only one agent.

It can be shown that every satisfiable formula $\varphi \in \mathcal{L}_{cla+p}$ is satisfiable in a MaSP with set of agents $\mathtt{N}(\varphi) \cup \{k\}$, for k being a new agent. k takes the role of all opponents of $\mathtt{N}(\varphi)$ of the model that satisfies φ collapsed into one: k gets the ability to perform exactly the actions that agents not occurring in φ can perform as a group.

[2] *True* playability adds the following condition to those of playability: For every $X \subseteq W, X \in \mathsf{E}(w)(\mathtt{N})$ implies $\{x\} \in \mathsf{E}(w)(\mathtt{N})$ for some $x \in X$. We refer to Goranko et al. (2011) for other equivalent definitions of true playability and a discussion of the relation between playability and true playability.

Theorem 2.21 *Every satisfiable formula $\varphi \in \mathcal{L}_{cla+p}$ is satisfiable in the class of MaSP's with at most $|\mathtt{N}(\varphi)| + 1$ agents.*

Proof. Assume that $M^{\preceq} = \langle W, Ac, (\rightarrow)_{A \subseteq Ac}, \mathtt{N}, \mathsf{act}, \{\preceq_i\}_{i \in \mathtt{N}}, \mathsf{V}\rangle$ satisfies φ. We only need to consider the case in which $\mathtt{N} \supset \mathtt{N}(\varphi)$ because if this is not the case we are already done.

Thus, for the case that $\mathtt{N} \supset \mathtt{N}(\varphi)$, we construct $M'^{\preceq'} = \langle W, Ac, (\rightarrow)_{A \subseteq Ac}, \mathtt{N}(\varphi) \cup \{k\}, \mathsf{act}', \{\preceq'_i\}_{i \in \mathtt{N}(\varphi) \cup \{k\}}, \mathsf{V}\rangle$, with $\mathsf{act}'(k) = \bigcup_{j \in \mathtt{N} \setminus \mathtt{N}(\varphi)} \mathsf{act}(j)$ and $\mathsf{act}'(i) = \mathsf{act}(i)$ for $i \neq k$. The preferences are defined as follows:

- $\preceq'_i = \preceq_i$ for $i \in \mathtt{N}(\varphi)$ and
- $\preceq'_k = W \times W$.

By induction, we can show that

$$M^{\preceq}, w \models \varphi \text{ iff } M'^{\preceq'}, w \models \varphi.$$

For Boolean connectives and formulas without coalition modalities, this is straightforward. The case where φ is of the form $\llparenthesis C \rrparenthesis\, \alpha$ follows from the definition of act'. Then the other cases involving coalition modalities follow. ■

Next, we want to know how many actions a model needs for satisfying some formula. Consider e.g. $\varphi = \llparenthesis C \rrparenthesis (p \wedge q) \wedge \llparenthesis C \rrparenthesis (\neg p \wedge q) \wedge \llparenthesis C \rrparenthesis (\neg p \wedge \neg q)$. It is only satisfiable in models with $|Ac| \geq 2$ because C has the power to make the system move into three disjoint sets of states and thus must be able to perform at least three different sets of actions. The main task is to find "witnesses" for formulas of the form $\llparenthesis C \rrparenthesis\, \psi$ in terms of concurrent actions. We can show that every satisfiable φ is satisfiable in a MaSP whose set of atomic actions consists of those in φ, one additional one (a dummy for ensuring that each agent can perform an action), and for every subformula $\llparenthesis C \rrparenthesis\, \psi$ or $\llparenthesis C^{\lhd_i} \rrparenthesis\, \psi$, one action for each of C's members.

The key step in transforming a model satisfying a formula φ into one whose set of actions satisfies the above condition is to appropriately define the action distribution and the accessibility relations. For every action formula α occurring in φ (in subformulas of the form $\llparenthesis C \rrparenthesis\, \alpha$ or $[\alpha]\psi$), we have to ensure that two states are related by an α-transition in the new model iff they were in the original one. Additionally, for formulas $\llparenthesis C \rrparenthesis\, \psi$ and $\llparenthesis C^{\lhd_i} \rrparenthesis\, \psi$, the set of actions introduced for them serves for making explicit how C can force φ.

Theorem 2.22 *Every satisfiable formula $\varphi \in \mathcal{L}_{cla+p}$ is satisfiable in a MaSP with at most $|Ac(\varphi)| + (\sum_{\llparenthesis C \rrparenthesis\, \psi \in Sub(\varphi)} |C|) + (\sum_{\llparenthesis C^{\preceq_i} \rrparenthesis\, \psi \in Sub(\varphi)} |C|) + (\sum_{\llparenthesis C^{\prec_i} \rrparenthesis\, \psi \in Sub(\varphi)} |C|) + 1$ actions.*

Proof. Assume that $M^{\preceq} = \langle W, Ac, (\rightarrow)_{A \subseteq Ac}, \mathtt{N}, \mathsf{act}, \{\preceq_i\}_{i \in \mathtt{N}}, \mathsf{V}\rangle$ satisfies φ. We construct a model $M'^{\preceq'} = \langle W, Ac', (\rightarrow')_{A' \subseteq Ac'}, \mathtt{N}, \mathsf{act}', \{\preceq'_i\}_{i \in \mathtt{N}}, \mathsf{V}\rangle$ as follows.

$Ac' := \quad Ac(\varphi) \quad \cup \quad \bigcup_{\langle\![C]\!\rangle\, \psi \in Sub(\varphi)} A_{C\psi} \quad \cup \quad \bigcup_{\langle\![C^{\preceq_i}]\!\rangle\, \psi \in Sub(\varphi)} A_{C^{\preceq_i}\psi} \quad \cup$
$\bigcup_{\langle\![C^{\prec_i}]\!\rangle\, \psi \in Sub(\varphi)} A_{C^{\prec_i}\psi} \cup \{\hat{a}\}.$

$A_{C\psi}$ and $A_{C^{\lhd_i}\psi}$ consist of newly introduced actions $a_{C\psi j}$, and $a_{C^{\lhd_i}\psi j}$ respectively, for each $j \in C$. Action abilities are distributed as follows:

$\mathsf{act}'(i) := \quad (\mathsf{act}(i) \cap Ac(\varphi)) \cup \{\hat{a}\} \cup \{a_{Ci} | \langle\![C]\!\rangle\, \psi \in Sub(\varphi)$ or $\langle\![C^{\lhd_i}]\!\rangle\, \psi \in$
$Sub(\varphi),$ for $i \in C\}.$

For defining the accessibility relation $\rightarrow'_{A' \subseteq Ac'}$, we first define for any state w its set of $\rightarrow'_{A'}$-successors $T^w_{A'}$.

$v \in T^w_{A'}$ iff the following conditions are satisfied:

1. $\forall [\alpha]\psi \in Sub(\varphi)$ such that $A' \models \alpha$: If $M^{\preceq}, w \models [\alpha]\psi$, then $M^{\preceq}, v \models \psi$,

2. $\forall \alpha \subseteq \lhd_i \in Sub(\varphi)$ such that $A' \models \alpha$: If $M^{\preceq}, w \models \alpha \subseteq \lhd_i$, then $w \lhd_i v$,

3. $\forall \langle\![C]\!\rangle\, \psi \in Sub(\varphi)$ such that $A' \models \bigwedge \Phi(A_{C\psi}, C)$, there is some $\bar{A} \subseteq \mathsf{act}(C)$ such that $w \rightarrow_A v$ for some $A \supseteq \bar{A}$ such that $A \models \bigwedge \Phi(\bar{A}, C)$, and if $M^{\preceq}, w \models \langle\![C]\!\rangle\, \psi$ then $M^{\preceq}, w \models [\bigwedge \Phi(\bar{A}, C)]\psi$

4. $\forall \langle\![C^{\lhd_i}]\!\rangle\, \psi \in Sub(\varphi)$ such that $A' \models \bigwedge \Phi(A_{C^{\lhd_i}\psi}, C)$, there is some $\bar{A} \subseteq \mathsf{act}(C)$ such that $w \rightarrow_A t$ for some $A \supseteq \bar{A}$ such that $A \models \bigwedge \Phi(\bar{A}, C)$, and if $M^{\preceq}, w \models \langle\![C^{\lhd_i}]\!\rangle\, \psi$ then $M^{\preceq}, w \models [\bigwedge \Phi(\bar{A}, C)]\psi$ and $M^{\preceq}, w \models (\bigwedge \Phi(\bar{A}, C) \subseteq \lhd_i)$.

For any $v \in T^w_{A'}$, we set $w \rightarrow'_{A'} v$. Then we can show by induction on $\psi \in Sub(\varphi)$ that $M^{\preceq}, w \models \psi$ iff $M'^{\preceq'}, w \models \psi$. ■

The next step is to show that every satisfiable formula φ is satisfiable in a model with a certain number of states. Such results are usually obtained by transforming a model into a smaller one using a transformation that preserves the truth of all subformulas of φ. In the case of CLA+P, the irreflexivity of the strict preferences and the fact that also $\alpha \subseteq \preceq_i$ is not modally definable in a basic modal language call for a modification of the standard techniques.

We appropriately modify the method of *filtration* (Blackburn et al. 2001) and show that any satisfiable formula $\varphi \in \mathcal{L}_{cla+p}$ is satisfiable in a model with exponentially many states. The idea of a filtration is to transform an infinite model into a finite one by identifying states that agree on the truth value of each subformula of the considered formula. So, given that we know that φ is satisfied in some MaSP $M^{\preceq}$ with states W, we construct an MaSP $\mathcal{M}^{\preceq^f}$ with set of states $W_{Sub(\varphi)} = \{|w|_{Sub(\varphi)} \mid w \in W\}$, where $|w|_{Sub(\varphi)}$ denotes the equivalence class of the states that in the model $M^{\preceq}$ agree with w on the truth values of all $\psi \in Sub(\varphi)$. The main task is to appropriately define the accessibility relations for actions and preferences in $\mathcal{M}^{\preceq^f}$ such that for $\psi \in Sub(\varphi)$, we then have that

$$M^{\preceq}, w \models \psi \text{ iff } \mathcal{M}^{\preceq^f}, |w| \models \psi.$$

Here, it is important to note that formulas of the form $\langle\![C]\!\rangle\, \psi$ and $\langle\![C^{\lhd_i}]\!\rangle\, \psi$ are equivalent to formulas of the form $\bigvee_{A \subseteq \mathsf{act}(C)} [\bigwedge \Phi(A, C)]\psi$ and $\bigvee_{A \subseteq \mathsf{act}(C)} ([\bigwedge \Phi(A, C)]\psi \wedge (\bigwedge \Phi(A, C) \subseteq \lhd_i))$, respectively – for $\bigwedge \Phi(A, C)$ as in the proof of Observation 2.18. Moreover, the transformation of the model does not change the underlying agents model. Thus, the truth of formulas $\langle\![C]\!\rangle\, \alpha$ is preserved.

Theorem 2.23 *Every satisfiable $\varphi \in \mathcal{L}_{cla+p}$ is also satisfiable in a MaSP with $\leq 2^{|\varphi|}$ many states.*

Proof. Given that $M^{\preceq}, w \models \varphi$ for some $M^{\preceq} = \langle W, Ac, (\rightarrow)_{A \subseteq Ac}, \mathtt{N}, \mathsf{act}, \{\preceq_i\}_{i \in \mathtt{N}}, \mathsf{V}\rangle$ and $w \in W$, we obtain $\mathcal{M}^{\preceq^f} = \langle W_{Sub(\varphi)}, Ac, (\rightarrow^f)_{A \subseteq Ac}, \mathtt{N}, \mathsf{act}^f, \{\preceq_i^f\}_{i \in \mathtt{N}}, \mathsf{V}^f\rangle$ by filtrating $M^{\preceq}$ through $Sub(\varphi)$, where the accessibility relations for actions and preferences are defined as follows:

$|w| \rightarrow_A^f |v|$ iff the following conditions are satisfied:

1. $\forall [\alpha]\psi \in Sub(\varphi)$ such that $A \models \alpha$: if $M^{\preceq}, w \models [\alpha]\psi$, then $M^{\preceq}, v \models \psi$,

2. (a) $\forall \alpha \subseteq \preceq_i \in Sub(\varphi)$ such that $A \models \alpha$: if $M^{\preceq}, w \models \alpha \subseteq \preceq_i$, then $w \preceq_i v$,

   (b) $\forall \alpha \subseteq \prec_i \in Sub(\varphi)$ such that $A \models \alpha$: if $M^{\preceq}, w \models \alpha \subseteq \prec_i$, then $w \prec_i v$,

3. $\forall \langle\![C]\!\rangle\, \psi \in Sub(\varphi)$ such that $A \models \bigwedge \Phi(A', C)$ for some $A' \subseteq \mathsf{act}(C)$: if $M^{\preceq}, w \models [\bigwedge \Phi(A', C)]\psi$, then $M^{\preceq}, v \models \psi$,

4. (a) $\forall \langle\![C^{\preceq_i}]\!\rangle\, \psi \in Sub(\varphi)$ such that $A \models \bigwedge \Phi(A', C)$ for some $A' \subseteq \mathsf{act}(C)$: if $M^{\preceq}, w \models [\bigwedge \Phi(A', C)]\psi$ and $M^{\preceq}, w \models (\bigwedge \Phi(A', C) \subseteq \preceq_i)$, then $M^{\preceq}, v \models \psi$ and $w \preceq_i v$.

   (b) $\forall \langle\![C^{\prec_i}]\!\rangle\, \psi \in Sub(\varphi)$ such that $A \models \bigwedge \Phi(A', C)$ for some $A' \subseteq \mathsf{act}(C)$: if $M^{\preceq}, w \models [\bigwedge \Phi(A', C)]\psi$ and $M^{\preceq}, w \models (\bigwedge \Phi(A', C) \subseteq \prec_i)$, then $M^{\preceq}, v \models \psi$ and $w \prec_i v$.

$|w| \preceq_i^f |v|$ iff the following conditions hold:

1. (a) $\forall \Diamond^{\preceq_i}\psi \in Sub(\varphi)$: if $M^{\preceq}, v \models \psi \vee \Diamond^{\preceq_i}\psi$ then $M^{\preceq}, w \models \Diamond^{\preceq_i}\psi$,

   (b) If there is some $\Diamond^{\prec_i}\psi \in Sub(\varphi)$, then $w \preceq_i v$,

2. If there is some $\alpha \subseteq \preceq_i \in Sub(\varphi)$ or some $\alpha \subseteq \prec_i \in Sub(\varphi)$, then $w \preceq_i v$,

3. If there is some $\llparenthesis C^{\preceq_i} \rrparenthesis \psi \in Sub(\varphi)$ or some $\llparenthesis C^{\prec_i} \rrparenthesis \psi \in Sub(\varphi)$, then $w \preceq_i v$.

$\mathsf{V}^f(p) := \{|w| \mid M^{\preceq}, w \models p\}$, for all propositional letters $p \in Sub(\varphi)$. We can show by induction that for all $\psi \in Sub(\varphi)$ and $w \in W$ it holds that $M^{\preceq}, w \models \psi$ iff $\mathcal{M}^{\preceq^f}, |w| \models \psi$. This follows from the definitions of $(\rightarrow^f)_{A \subseteq Ac}$ and $\preceq^f$, and the fact that we do not change the underlying agents model. The interesting cases are those involving strict preferences and those with formulas $\llparenthesis C^{\preceq_i} \rrparenthesis \psi$ and $\alpha \subseteq \preceq_i$. Here, what makes the proof go through is that by conditions 1 b), 2 and 3 of $\preceq^f$, $|w| \preceq_i |v|$ implies $w \preceq_i v$. Similarly, due to conditions 2 and 4 of $\rightarrow_A^f$, the truth values of subformulas $\alpha \subseteq \lhd_i$ and $\llparenthesis C^{\lhd_i} \rrparenthesis \psi$ is as in the original model. Moreover, $\mathcal{M}^{\preceq^f}$ is a proper MaSP since each $\preceq_i^f$ is reflexive and transitive, and each $\rightarrow_A^f$ is serial. By definition of $W_{Sub(\varphi)}$, $|W_{Sub(\varphi)}| \leq 2^{|\varphi|}$. ■

Now, we apply the constructions of the last three proofs successively.

Corollary 2.24 *Every satisfiable formula $\varphi \in \mathcal{L}_{cla+p}$ is satisfiable in a MaSP of size exponential in $|\varphi|$ satisfying the conditions $|\mathtt{N}| \leq |\mathtt{N}(\varphi)| + 1$ and $|Ac| \leq |Ac(\varphi)| + \sum_{\llparenthesis C \rrparenthesis \psi \in Sub(\varphi)} |C| + (\sum_{\llparenthesis C^{\preceq_i} \rrparenthesis \psi \in Sub(\varphi)} |C|) + (\sum_{\llparenthesis C^{\prec_i} \rrparenthesis \psi \in Sub(\varphi)} |C|) + 1$.* ◀

Having non-deterministically guessed a model of size exponential in $|\varphi|$, we can check in time exponential in $|\varphi|$ whether this model satisfies φ.

Theorem 2.25 *The satisfiability problem of CLA+P is in* NEXPTIME.

Proof. Given φ, we non-deterministically choose a model $M^{\preceq}$ of size exponential in $|\varphi|$ satisfying the conditions $|\mathtt{N}| \leq |\mathtt{N}(\varphi)| + 1$ and $|Ac| \leq |Ac(\varphi)| + \sum_{\llparenthesis C \rrparenthesis \psi \in Sub(\varphi)} |C| + (\sum_{\llparenthesis C^{\preceq_i} \rrparenthesis \psi \in Sub(\varphi)} |C|) + (\sum_{\llparenthesis C^{\prec_i} \rrparenthesis \psi \in Sub(\varphi)} |C|) + 1$. Then, given this model, we can check in time $\mathcal{O}(|\varphi| \|M^{\preceq}\|)$, for $\|M^{\preceq}\|$ being the size of $M^{\preceq}$, whether $M^{\preceq}$ satisfies φ. Thus, given a model of size exponential in $|\varphi|$ (the length of φ) that also satisfies the conditions on its sets of agents and actions explained earlier, it can be computed in time exponential in the length of φ whether it satisfies φ. Since it can be checked in time linear in the size of the model whether it is a proper MaSP, we conclude that SAT of CLA+P is in NEXPTIME. ■

Adapting standard techniques for modal logic to the special properties of CLA+P, we could thus show that SAT of CLA+P is decidable in NEXPTIME. Now we show that the environment logic itself is already EXPTIME-hard.

2.4.2 Lower Bound

In order to show a lower bound for the complexity of SAT of CLA+P, we show that SAT of the environment logic is EXPTIME-hard. This is done by reduction from the Boolean modal logic $\mathbf{K}_m^{\neg\cup}$ (Lutz and Sattler 2001; Lutz et al. 2001).

Formulas of $\mathbf{K}_m^{\neg\cup}$ are interpreted in models $M = \langle W, R_1, \dots R_m, \mathsf{V}\rangle$, where W is a set of states, $R_i \subseteq W \times W$ and V is a valuation. $\mathbb{M}_m^{\neg\cup}$ denotes the class of all such models. Intuitively, $\mathbf{K}_m^{\neg\cup}$ is a modal logic that can also talk about Boolean combinations of accessibility relations.

Definition 2.26 Let $\mathcal{R}_1, \dots \mathcal{R}_m$ be atomic modal parameters. Then the set of modal parameters of $\mathbf{K}_m^{\neg\cup}$ is the smallest set containing $\mathcal{R}_1, \dots \mathcal{R}_m$ that is closed under $\neg$ and $\cup$. The language $\mathcal{L}_m^{\neg\cup}$ is generated by the following grammar:

$$\varphi ::= \; p \mid \varphi \wedge \varphi \mid \neg\varphi \mid \langle \mathcal{S}\rangle\varphi \qquad\qquad \mathcal{S} ::= \; \mathcal{R}_i \mid \neg\mathcal{S} \mid \mathcal{S}_1 \cup \mathcal{S}_2. \qquad ◀$$

The extension $\mathcal{E}x(\mathcal{S}) \subseteq W \times W$ of a parameter $\mathcal{S}$ in a model is as follows.

$$\begin{array}{lcl} \mathcal{E}x(\mathcal{R}_i) & = & R_i \\ \mathcal{E}x(\neg\mathcal{S}) & = & (W \times W) \setminus \mathcal{E}x(\mathcal{S}) \\ \mathcal{E}x(\mathcal{S}_1 \cup \mathcal{S}_2) & = & \mathcal{E}x(\mathcal{S}_1) \cup \mathcal{E}x(\mathcal{S}_2) \end{array}$$

Formulas of $\mathcal{L}_m^{\neg\cup}$ are interpreted in a model $M = \langle W, R_1, \dots R_m, \mathsf{V}\rangle$ as follows: Propositional letters and Boolean combinations are interpreted in the standard way and for modal formulas we have

$$M, w \models \langle \mathcal{S}\rangle\varphi \quad \text{iff} \quad \exists w' \in W : (w, w') \in \mathcal{E}x(\mathcal{S}) \text{ and } M, w' \models \varphi.$$

We define a translation τ consisting of two components τ_1 for formulas and τ_2 for models. Let us extend the environment language $\mathcal{L}_e$ by a propositional letter $q \notin \mathcal{L}_m^{\neg\cup}$. Then the translation τ_1 for formulas is defined as follows using the translation τ^S inside the modalities:

$$\begin{array}{rcl@{\qquad}rcl} \tau_1(p) & = & p & \tau^S(\mathcal{R}_i) & = & a_i \\ \tau_1(\varphi_1 \wedge \varphi_2) & = & \tau_1(\varphi_1) \wedge \tau_1(\varphi_2) & \tau^S(\mathcal{S}_1 \cup \mathcal{S}_2) & = & \tau^S(\mathcal{S}_1) \vee \tau^S(\mathcal{S}_2) \\ \tau_1(\neg\varphi) & = & \neg\tau_1(\varphi) & \tau^S(\neg\mathcal{S}) & = & \neg\tau^S(\mathcal{S}) \\ \tau_1(\langle\mathcal{S}\rangle\varphi) & = & \neg[\tau^S(\mathcal{S})](q \vee \neg\tau_1(\varphi)) & & & \end{array}$$

τ_2 translates a model $M = \langle W, R_1, \dots R_m, \mathsf{V}\rangle$ of $\mathbf{K}_m^{\neg\cup}$ into an environment model $\tau_2(M) = \langle W \cup \{u\}, Ac, (\rightarrow)_{A \subseteq Ac}, \mathsf{V}'\rangle$ with u being a newly introduced state that will serve for making the accessibility relations for the actions serial, and $Ac = \{a_1, \dots a_m\}$. The accessibility relations $\rightarrow_A$ relates two states if the same states were in the relation for each R_i for $a_i \in A$. As we have to make it serial, we define $\rightarrow_A$ as follows.

$$w \rightarrow_A w' \text{ iff } A = \{a_i \mid (w, w') \in R_i \text{ or } w' = u\}.$$

Thus, each $\rightarrow_A$ is serial and $\tau_2(M)$ is an environment model. $\mathsf{V}'(q) = \{u\}$, and for all $p \neq q$, $\mathsf{V}'(p) = \mathsf{V}(p)$. Before showing that for any $\varphi \in \mathcal{L}_m^{\neg\cup}$ and $M \in \mathcal{M}_m^{\neg\cup}$ for any state $w \in W : M, w \models \varphi$ iff $\tau_2(M), w \models \tau_1(\varphi)$, we prove a lemma saying that if in M the state w' is $\mathcal{S}$-accessible from w, then in the model $\tau_2(M)$, w' is accessible from w by a transition of type $\tau^S(\mathcal{S})$.

Notation For $M = \langle W, R_1, \dots R_m, \mathsf{V}\rangle \in \mathbb{M}_m^{\neg\cup}$ and $\tau_2(M) = \langle W \cup \{u\}, Ac, (\rightarrow)_{A\subseteq Ac}, \mathsf{V}'\rangle$, define $A_{w,w'} := \{a_i \in Ac \mid (w, w') \in R_i\}$. ◀

Lemma 2.27 *Let $M = \langle W, R_1, \dots R_m, \mathsf{V}\rangle$ be a model of $\mathbf{K}_m^{\neg\cup}$. Then for any modal parameter $\mathcal{S}$ and for any states $w, w' \in W$ it holds that*

$$(w, w') \in \mathcal{E}x(\mathcal{S}) \textit{ iff in } \tau_2(M) : \exists A \subseteq Ac : w \rightarrow_A w' \textit{ and } A \models \tau^S(\mathcal{S}).$$

Proof. Note that by definition of $(\rightarrow_A)_{A\subseteq Ac}$, the righthand side is equivalent to $A_{w,w'} \models \tau^S(\mathcal{S})$. Then the proof goes by induction on $\mathcal{S}$. ■

Theorem 2.28 *For any formula $\varphi \in \mathcal{L}_m^{\neg\cup}$ and any model M of $\mathbf{K}_m^{\neg\cup}$, it holds that for any state w in M:*

$$M, w \models \varphi \textit{ iff } \tau_2(M), w \models \tau_1(\varphi).$$

Proof. By induction. Base case and Boolean cases are straightforward. Let $\varphi = \langle\mathcal{S}\rangle\psi$.

(⇒) If $M, w \models \langle\mathcal{S}\rangle\psi$, this means that $\exists w' : (w, w') \in \mathcal{E}x(\mathcal{S})$ and $M, w' \models \psi$. By the previous lemma and induction hypothesis, $\tau_2(M), w \models \tau_1(\langle\mathcal{S}\rangle\psi)$.

(⇐) Assume that $\tau_2(M), w \models \tau_1(\langle\mathcal{S}\rangle\psi)$. This is equivalent to $\tau_2(M), w \models \neg[\tau^S(\mathcal{S})](q \vee \neg\tau_1(\psi))$. Then there is some state $\exists w' \in W \cup \{u\}$ and a set of actions$\exists A \in Ac$ such that $A \models \tau^S(\mathcal{S}), w \rightarrow_A w'$ and $\tau_2(M), w' \models \neg q \wedge \tau_1(\psi)$. Thus, $w' \neq u$. By induction hypothesis and the previous lemma, $M, w' \models \psi$ and $(w, w') \in \mathcal{E}x(\mathcal{S})$. Hence, $M, w \models \langle\mathcal{S}\rangle\psi$. ■

Theorem 2.29 SAT *of* Λ^E *is* EXPTIME-*hard.*

Proof. Follows from the fact that SAT of $\mathbf{K}_m^{\neg\cup}$ is EXPTIME-hard (Lutz and Sattler 2001; Lutz et al. 2001) and Theorem 2.28, which says that SAT of $\mathbf{K}_m^{\neg\cup}$ is polynomially reducible to SAT of Λ^E. ■

As the environment logic itself is already EXPTIME-hard, this thus also holds for the full CLA+P.

Corollary 2.30 SAT *of CLA+P is* EXPTIME-*hard.* ◀

This section has shown that the satisfiability problem of CLA+P is EXPTIME-hard but still decidable. This rather high complexity is due to the environment logic which itself is already EXPTIME-hard.

2.5 Conclusions and Further Questions

We will now summarize the main results of this chapter and then give conclusions and further questions.

2.5.1 Summary

We developed a modular modal logic that allows for reasoning about the coalitional power of agents, actions and their effects, and agents' preferences. The current approach is based on the logic CLA (Sauro et al. 2006) which is combined with a preference logic (van Benthem et al. 2007). The resulting sound and complete logic CLA+P allows us to make explicit how groups can achieve certain results. We can also express how a group can achieve that a transition takes place that is an improvement for some agent.

Conceptual benefits In the framework of CLA+P, it can be expressed how the abilities to perform certain actions are distributed among the agents, what are the effects of the concurrent performance of these actions and what are the agents' preferences over those effects. Moreover, in CLA+P, we can distinguish between different ways how groups can achieve some result – not only with respect to the actions that lead to some result, but also with respect to the preferences. We can for instance express that a group can achieve some result in a way that is 'good' for all its members in the sense that after the achievement all of them are better off. Coming back to the strategic interaction between rivaling companies explained on page 47, we could thus formalize and answer the following.

> Can company *C* achieve some profit while making sure that for all of its current employees the resulting situation will be at least as good as the current one?

Note that 'C making a certain profit' can be represented by a propositional letter and the preferences of the employees can be represented by preference relations. Then in our formalization, this means that *C* can make the system move into a state that is at least as good for all its current employees and at which it holds that *C* makes a certain profit.

Thus, our framework provides a fine-grained model of cooperative ability as we can distinguish between "good" and "bad" ways to achieve something.

This then also allows us to axiomatize properties that one might want to impose onto a multi-agent system, e.g. the restriction that groups can only achieve the truth of a certain formula if this can be done without making anybody worse off. Thus, CLA+P provides a framework for reasoning about interactive situations in an explicit way that gives us more insights into the cooperative abilities of agents. Comparing CLA+P to CL shows that CLA+P naturally builds on game frames underlying the semantics of CL and makes both the agents' actions and the preferences explicit that are only implicitly represented in the semantics of CL.

Computational complexity The satisfiability problem of CLA+P is shown to be decidable and EXPTIME-hard. Keeping in mind that using CLA+P we can talk about strict preferences, intersections of accessibility relations as well as the property of one relation being a subset of another, EXPTIME-hardness is not a surprising result. Even though the modular models of CLA+P are rather special, its complexity is in accordance with general results concerning the connection between expressive power and complexity of modal logics for reasoning about coalitional power and preferences, as we will see in the next chapter.

We showed that the satisfiability problem of the underlying environment logic is by itself already EXPTIME-hard. Thus, we identified a source of the high complexity of CLA+P. It is mostly due to the fact that the accessibility relation of the models can be arbitrary: e.g. there does not need to be any relation between $\rightarrow_A$, $\rightarrow_B$ and $\rightarrow_{A\cap B}$. Whereas this generality allows us to model a lot of dynamic processes, from a computational viewpoint, it seems to be appealing to change the environment logic in order to decrease computational complexity. Also, when comparing our models to the game frames of CL, we can see that restricting ourselves to deterministic environment models can be reasonable. The same holds for assuming preference orders to be total preorders, which is an assumption we will make in the next chapter. This assumption would also increase the expressive power as e.g. if a coalition can perform an action that will lead to some result φ, but the action does not have the effect to lead to a state at least as good for some agent, then by the totality of the preferences we can conclude that the action can lead to a state strictly worse for the agent (cf. the discussion of different concepts combining coalitional power and preferences (Dégremont and Kurzen 2009a)).

This chapter illustrated how a modal logic for actions, cooperation and preferences is developed, based on the conceptual motivation to make the ability of individuals and groups more explicit. Our complexity analysis showed the computational consequences of the design choices made.

- Basing CLA+P on an action logic which contains the Boolean negation on relations has the effect of making the logic EXPTIME-hard.
- The crucial steps in showing decidability of CLA+P are the following:
  - to give an upper bound on the number of actions that are needed for making implicit coalition modalities explicit.

   - to adapt the technique of filtration to show that CLA+P has the finite model property.

2.5.2 Conclusion

Coming back to Research Question 1, we can conclude the following.

1. Combining an explicit representation of actions and preferences can have conceptual benefits for cooperation logics if we want to reason about the effect different joint actions have on the 'happiness' of individuals.
2. Formalizing coalitional power in terms of the effects of concurrent actions can be computationally expensive.

What the results of this chapter mean for the complexity of interaction in general depends on how much of the complexity results are due to the chosen way to combine logics for coalitional ability, effects of actions and individual preferences. One way to determine this would be to consider other methods for combining existing logical systems, e.g. with a fibring approach (Gabbay 1998). Alternatively, we could zoom in more into interaction itself, trying to get a clearer view on what is the intrinsic (and thus inevitable) complexity of certain social phenomena. In the remainder of this dissertation we will try to follow this path.

2.5.3 Further Questions

This chapter gives rise to some questions to be further investigated. We start with some immediate technical questions that follow from our results.

- Is the satisfiability problem of the logic developed in this chapter in EXPTIME or NEXPTIME-hard?

  The complexity bounds we have given are not tight. We would conjecture that the procedure we gave for decidability could be made more efficient, so that then EXPTIME-completeness could be shown.

- Is EXPTIME-hardness solely due to the underlying environment logic?

  In our analysis, we have shown hardness by showing hardness of one of the sublogics. It would be interesting to see whether hardness is only caused by the environment logic or whether it can also be shown using only some of the other sublogics.

The design choices we made resulted from the fact that first of all we wanted to make coalitional power explicit and be able to distinguish between different ways how results can be achieved, not only with respect to how the results can be achieved but also with respect to how good a certain choice of collective action would be for individual agents. Additionally, we aimed for a very general system, making only the assumptions that every action can be performed by some agent, all actions can be performed in every state and agents' preferences are reflexive and transitive. The advantage of this is that a wide range of different interactive scenarios can be modeled. However, when we are interested in specific game-like interactions, a different methodology might be more appropriate. Instead of starting with general considerations concerning the conceptual motivations for choosing a certain cooperation logic, we can start with the game theoretical concepts that we would like to be able to reason about using the logic and then evaluate different choices w.r.t. their computational properties.

This gives rise to the following question:

- Given that we want to develop a modal logic that can capture certain aspects of cooperation, how much expressive power would we need for this, and what will be the complexity of such a logic?

Clarifying and answering this question will help to make design choices when developing logics for reasoning about interaction.

Chapter 3

Complexity and Descriptive Difficulty of Reasoning about Cooperation in Modal Logics

In the previous chapter, we have seen an example of a formal framework for reasoning about strategic interaction of groups of agents in an explicit way. Investigating the complexity of that logic has led us to the question of how we can get a clear grip on the computational properties of logical systems designed for reasoning about certain aspects of strategic interaction.

In this chapter, we analyze the complexity of reasoning about the strategic abilities of groups and individuals in different modal logic frameworks. In general, the frameworks we consider in this chapter are all designed to capture situations as the one of the rivaling companies explained in the previous chapter.

The analysis given in this chapter is motivated by the following question which plays a crucial role in the process of designing formal frameworks for reasoning about interaction between agents.

- Given that we want to develop a modal logic that can capture certain aspects of cooperation, how much expressive power would we need for this, and what will be the complexity of such a logic?

We aim to answer this question focusing on the following three general types of approaches to modal logics for cooperation and preferences:

1. very simple models that directly represent coalitional power,
2. action-based coalitional models that explicitly represent the actions by which (groups of) agents can achieve something,
3. power-based coalitional models that focus on how the ability of groups arises from that of their subgroups.

For each class of models, we evaluate the following: how much expressive power is needed for expressing certain concepts inspired from game theory and social choice theory? And what will be the complexity of the resulting logical systems?

Similarly to the previous chapter, this chapter contributes to the area of modal logics for agency (cf. e.g. Horty and Belnap (1995)) and reasoning about social concepts, with a particular focus on coalitional power (cf. e.g. Pauly (2002a)) and preferences (cf. e.g. Girard (2008)). On the technical side, this chapter also provides a semantic comparison of different logics by giving transformations between different classes of models.

3.1 Reasoning about efficiency and stability in modal logics

We can think of cooperative and non-cooperative game theory as theories of *stability* of states of interactive systems, and social choice theory as a theory of *fairness* and *efficiency* of such states.

This chapter contributes to the general project of bringing the perspective of descriptive complexity to the analysis of problems raised by the analysis of multi-agent systems in terms of *stability*, *efficiency* and connected concepts. We are concerned with studying the expressive power required for logical languages to reason about interactive systems in terms of such notions. Consequences in computational complexity can then be drawn, paving the way for a descriptive perspective on the complexity of certain types of game- and social choice theoretical reasoning. In this chapter, we take an abstract perspective (as e.g. Roux et al. (2008)) on interactive systems and represent the structures for cooperative ability together with the preferences of individuals as simple relational structures, as it is done in modal logics for reasoning about cooperation in multi-agent systems. We aim towards a unified perspective on modal logics for coalitional power of agents with preferences, both on a model-theoretical and syntactic level. It is important to note that contrary to the previous chapter, the objective of this chapter is *not* to propose a new modal logic for interaction but to develop a unifying perspective on different classes of existing ones. Our work is similar in spirit to e.g. Goranko (2001), Broersen et al. (2009) and Goranko and Jamroga (2004), aiming towards a unified perspective on different logics for multi-agent systems modeling similar concepts. We distinguish logics that explicitly represent the actions (such as e.g. CLA+P of Chapter 2) and those that take coalitional power as a primitive.

Our main aim is to determine the expressive power and complexity needed for modal logics to express concepts from game theory and social choice theory. For qualitative coalitional games (Wooldridge and Dunne 2004) a similar analysis has been done by Dunne et al. (2007). Our work differs in the sense that we represent agents' individual preferences by binary relations over the state space instead of considering *goals* whose achievement leads to the satisfaction of agents.

How much expressive power and complexity is needed for expressing certain concepts depends of course on the associated semantic structures of the logics. We analyze three classes of models for cooperation and determine how demanding different concepts from game theory and social choice theory are on each class of models. Our results help to make design choices when developing modal logics for cooperation since we clarify the impact of certain choices on the complexity and expressive power required to express different concepts.

Let us briefly outline the methodology followed in this chapter. First of all, we focus on classes of logics for cooperation with natural and well studied model-theoretical properties. We consider three different normal modal logic frameworks for reasoning about the cooperative ability of agents[1] and extend them with agents' preferences as total preorders over the states. Then, we analyze both the relation between these logics and also consider the relation to CL. Next, we focus on some notions of interest for reasoning about cooperation, and give their natural interpretations in each of the models. We then determine the expressive power required by these notions; we do this by checking under which operations these properties are invariant. Using characterization results for extended modal logics, we then obtain extended modal languages that can express the notions. Among these, we choose the ones with the lowest expressive power and give explicit definability results for the notions in these languages. Using known complexity results for extended modal logics, we then also obtain upper bounds (UB) on the complexity of the satisfiability problem (SAT) and on the combined complexity of the model checking problem (MC) of modal logics expressing each notion.

3.2 Three ways of modeling cooperation

We consider three classes of models that are simplifications or generalizations of models used in the literature. We choose simple and general models in order to avoid additional complexity resulting from particular constraints on the models. Our simple models then allow us to distinguish clearly how expressing the notions is demanding by itself and also to evaluate from a high-level perspective how appropriate the models are for reasoning about which aspects of cooperation.

[1] Thus, our approach is similar to that of Broersen et al. (2009) who also investigate different normal modal logics for cooperation, with the difference that we consider generalizations of existing approaches, dropping several assumptions on the models.

The first class of models we consider is the class of *coalition-labeled transition systems* (Dégremont and Kurzen 2009a; Dégremont 2010). These models focus on the interaction between preferences and cooperation, simplifying the computation of coalitional powers itself as they are directly represented as accessibility relations for each coalition. The second class of models, *action-based coalitional models*, represents coalitional power in terms of actions. The third class, *power-based coalitional models*, focuses on reasoning about and computing coalitional power itself, representing groups' choices as partitions of the state space. Unless explicitly stated otherwise, preferences on all the models are represented by total preorders (TPO) (i.e., total, reflexive and transitive binary relations) over the states.

At this point, we note that our choice of three different approaches to modeling coalitional power by no means completely covers the whole range of approaches existing in the literature[2]. The three approaches we consider present three examples of how coalitional power can be modeled from different perspectives. There are existing approaches to coalitional power which in a sense combine different aspects of the models we consider (e.g. Herzig and Lorini (2010); Lorini et al. (2009)).

Our main aim is to explain how coalitional power is represented in each of the different approaches. We also briefly give extended modal languages which can be interpreted on the models, and also axiomatizations of the corresponding logical systems.

The reader mainly interested in the conceptual differences of coalitional ability in the three approaches can skip the sections with the languages and axiomatizations and refer back to it later in Section 3.3.4, where we use the extended modal languages to express interesting properties.

We will use the following notation in this chapter.

Notation Our models are again based on a finite set of agents N. j ranges over N. PROP is still the set of propositional letters and NOM a set of *nominals*, which is disjoint from PROP. A nominal is true in exactly one state. We let $p \in$ PROP and $i \in$ NOM. For R being a binary relation on a set W, i.e., $R \subseteq W \times W$, we write $R[w] := \{v \in W \mid wRv\}$.

3.2.1 Coalition-labeled transition systems

Sequential or *turn-based* systems – Kripke models with an accessibility relation for each coalition – can be used for reasoning about coalitional power: in each state a group (if it is its turn) has the power to move the system into exactly

[2] See e.g. Troquard et al. (2009) for an approach in which coalitional power arises from individuals' abilities to control the truth value of propositional letters.

the states accessible by that group's relation. These models generalize conversion/preference games (Roux et al. 2008) and the models of agency introduced by Segerberg (1989). The former take an abstract view on game theoretical models, based on the idea that game theory is a theory of stable vs. unstable states in interactive systems. Here, the focus is not on how coalitional power arises from the powers of individuals. Rather coalitional power itself is taken as a primitive.

Definition 3.1 (LTS model) A labeled transition system (LTS) indexed by coalitions in $\wp(\mathtt{N})$ is a 5-tuple of the form $\langle W, \mathtt{N}, (\xrightarrow{C})_{C \subseteq \mathtt{N}}, (\leq_j)_{j \in \mathtt{N}}, \mathsf{V} \rangle$, where

- $W \neq \emptyset$ is a set of states,

- $\mathtt{N}$ is a finite set of agents,

- $\xrightarrow{C} \subseteq W \times W$ for each $C \subseteq \mathtt{N}$ represents the coalitional power of C: $w \xrightarrow{C} v$ means that it is in the power of coalition C to change the system's state from w into v.

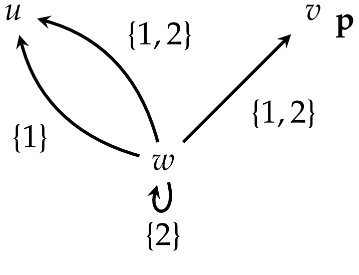
- $\leq_j \subseteq W \times W$ is a $\mathtt{TPO}$ for each $j \in \mathtt{N}$, which represents agent j's preferences over the states: $w \leq_j v$ means that j finds v at least as good as w,

- $\mathsf{V} : \textsc{prop} \cup \textsc{nom} \to \wp(W)$ is a valuation function with $|\mathsf{V}(i)| = 1$ for each $i \in \textsc{nom}$.

We also refer to labelled transition systems indexed by coalitions in $\wp(\mathtt{N})$ by $\wp(\mathtt{N}) - \mathtt{LTS}$ models or simply as $\mathtt{LTS}$ models (if the set of agents is clear). ◀

Thus, an LTS models a system of agents by representing the ability of groups of agents in terms of transitions between states of the system, over which the individuals have preferences.

Example 3.2 Consider the following example of an LTS labelled by coalitions of the set of agents $\mathtt{N} = \{1, 2\}$. We let all agents be indifferent between all states, and do not explicitly represent the preferences here.



Then the grand coalition has the power to change the state of the system from w into u and also into v. The singleton coalition consisting of agent 1 (agent 2) only has the power to change the state of the system from w into u (keep the system in state w). ◀

As an example of an interactive situation represented in an LTS model, we will consider a multi-agent resource allocation setting (cf. Endriss et al. (2006)). The idea is that the state space represents the possible allocations, and the relations for the coalitions represent the possible changes of allocations that groups of agents can achieve by exchanging their resources.

Example 3.3 Consider a resource allocation setting as studied by Endriss et al. (2006). Given a finite set of agents $\mathtt{N}$ and a finite set of resources $\mathcal{R}$, an *allocation* divides all resources in $\mathcal{R}$ amongst agents in $\mathtt{N}$: An allocation is a function $A : \mathtt{N} \to \wp(\mathcal{R})$ such that for all $j, j' \in \mathtt{N}$ it holds that $A(j) \cap A(j') = \varnothing$ and $\bigcup_{j \in \mathtt{N}} A(j) = \mathcal{R}$.

Then a *deal* among a subset of agents is an exchange of resources among them. Thus a deal can be seen as a transition from one allocation to a different one. A deal is then represented as a pair $\delta = (A, A')$ such that $A \neq A'$. Each agent $j \in \mathtt{N}$ has a utility function $u_j : \wp(\mathcal{R}) \to \mathbb{R}$ that represents how much the agent likes each set of resources.

Given a set of agents $\mathtt{N}$ a set of resources $\mathcal{R}$, and the set $\mathbb{A}$ of all allocations of $\mathcal{R}$ to $\mathtt{N}$, we can model this scenario in a $\wp(\mathtt{N}) - \mathtt{LTS}$ model as follows. We can define $\mathcal{M} = \langle W, \mathtt{N}, (\xrightarrow{C})_{C \subseteq \mathtt{N}}, (\leq_j)_{j \in \mathtt{N}}, V \rangle$ with

- $W = \mathbb{A}$,
- $A \xrightarrow{C} A'$ iff there is deal $\delta = (A, A')$ that involves only agents in C,
- $A \leq_j A'$ iff $u_j(A) \leq u_j(A')$.

The valuation function can be chosen to fit the context of the allocation setting; e.g. we might be interested in who is assigned a particular resource. This can be represented by a corresponding propositional letter for each agent, which is then made true in exactly those states in which that agent is assigned that resource.

With our formalization, we can then study resource allocation settings from an abstract perspective as graphs, and find logical characterizations of relevant properties, e.g. the existence of loops of rational deals. A deal can be called rational if it does not decrease the utility of any of the agents involved in it. In our model, a rational deal for a set of agents would then be a transition along the intersection of the coalition relation for that set of agents and each of the agents' preference relations. ◀

To summarize, the class of coalition-labeled transition systems models cooperative ability of agents in a simple way. In each state, a coalition (if it is its turn) has the power to make the system move into exactly the states accessible by the relation labeled with that coalition.

An interesting feature of the class of $\wp(\mathtt{N})-\mathtt{LTS}$ is that except for the assumption of agents' preferences being total preorders, preferences of individuals and coalitional ability are modeled in the same way. Both for checking if a coalition can force that some formula is true and also for checking if an agent prefers a state in which a formula is true, we need to check the states accessible by the corresponding accessibility relation and check if the formula is true there. This is interesting from a conceptual perspective, and, as we will see later, also has interesting technical consequences concerning the expressive power and complexity needed to reason about properties involving both coalitional power and preferences.

Language interpreted on $\wp(\mathtt{N}) - \mathtt{LTS}$

Now we can define our basic language $\mathcal{L}_{\text{LTS}}$ and some of its extensions. Note that here and later for the other systems we will only define those fragments of extended modal languages that we need later for expressing some interesting properties; thus we do not give the definitions for a full hybrid extension or a full extension with *PDL* modalities (cf. e.g. Fischer and Ladner (1979a); Harel (1984)).

We start by defining a set of programs.

$$\alpha ::= \leq_j \ | C$$

Then, $\mathcal{L}_{\text{LTS}}$ is defined as follows.

$$\varphi ::= \ p \,|\, \neg\varphi \,|\, \varphi \ \wedge \ \varphi \,|\, \langle\alpha\rangle\varphi,$$

where $j \in \mathtt{N}, C \in \wp(\mathtt{N}), C \neq \emptyset, p \in \textsc{prop}$.

Extending the language by allowing intersections of the modality, we get the language $\mathcal{L}_{\text{LTS}}(\cap)$, which is defined just as the basic language with

$$\alpha ::= \leq_j \ | \mathrm{C} \,|\, \alpha \cap \alpha.$$

Another extension that we will need later is the hybrid extension of $\mathcal{L}_{\text{LTS}}$, which we denote by $\mathcal{H}_{\mathcal{L}_{\text{LTS}}}(\downarrow)$.

$$\alpha ::= \leq_j \ | C$$

$$\varphi ::= \ p \,|\, i \,|\, x \,|\, \neg\varphi \,|\, \varphi \ \wedge \ \varphi \,|\, \langle\alpha\rangle\varphi \,| \ \downarrow x.\varphi$$

where $j \in \mathtt{N}$, $C \in \wp(\mathtt{N}), C \neq \emptyset$, $p \in \textsc{prop}$, $i \in \textsc{nom}$, $x \in \textsc{svar}$. $\textsc{svar}$ is a countable set of variables.

For all these languages, programs α are interpreted as binary relations $R_\alpha \subseteq W \times W$.

$$\begin{array}{lcl} R_{\leq_j} & = & \leq_j \\ R_C & = & \xrightarrow{C} \\ R_{\alpha\cap\beta} & = & R_\alpha \cap R_\beta \end{array}$$

Formulas are interpreted with an assignment $g : \text{SVAR} \to W$. Boolean combinations are interpreted in the standard way.

$$\begin{array}{lll} \mathcal{M}, w, g \models p & \text{iff} & w \in \mathsf{V}(p) \\ \mathcal{M}, w, g \models i & \text{iff} & \{w\} = \mathsf{V}(i) \\ \mathcal{M}, w, g \models x & \text{iff} & w = g(x) \\ \mathcal{M}, w, g \models \langle\alpha\rangle\varphi & \text{iff} & \exists v : wR_\alpha v \text{ and } \mathcal{M}, v, g \models \varphi \\ \mathcal{M}, w, g \models \downarrow x.\varphi & \text{iff} & \mathcal{M}, w, g[x := w] \models \varphi \end{array}$$

Thus, the languages we consider for the class of $\wp(\mathtt{N}) - \mathtt{LTS}$ models include a hybrid extension of the basic language and also extensions that allow for intersections inside the modalities, which then gives us modalities that run along the intersections of the basic coalition relations and the preference relations of individuals.

Axiomatization

We briefly give the axioms for the different fragments we will use later. For the basic system, we have a multi-modal logic with the **K** axiom for each coalition modality. The preference fragment is axiomatized by **S4.3**.

Definition 3.4 The axioms of the basic logic for the class of LTS contain the axiom schemes of propositional logic and additionally the following axiom schemes.

$$\begin{array}{ll} \mathbf{K}([C]) & [C](\varphi \to \psi) \to ([C]\varphi \to [C]\psi) \\ \mathbf{K}([\leq_j]) & [\leq_j](\varphi \to \psi) \to ([\leq_j]\varphi \to [\leq_j]\psi) \\ \mathbf{4}(\langle\leq_j\rangle) & \langle\leq_j\rangle\langle\leq_j\rangle p \to \langle\leq_j\rangle p \\ \mathbf{T}(\langle\leq_j\rangle) & p \to \langle\leq_j\rangle p \\ \mathbf{.3}(\langle\leq_j\rangle) & \langle\leq_j\rangle p \wedge \langle\leq_j\rangle q \to \langle\leq_j\rangle(p \wedge \langle\leq_j\rangle q) \vee \langle\leq_j\rangle(p \wedge q) \vee \langle\leq_j\rangle(q \wedge \langle\leq_j\rangle p) \end{array}$$

The rules of inference are modus ponens, uniform substitution and necessitation. ◄

The hybrid and Boolean extensions of the basic system are axiomatized in the standard way.

From the class of LTS which models coalitional power from a high-level perspective, focusing on the distribution of power among coalitions in a social system, we will now move on to classes of models focusing on agents making simultaneous and independent decisions.

3.2.2 Action-based coalitional models

In the second class of models that we consider – action-based models – coalitional power arises from the agents' abilities to perform actions, just as in the models considered in Chapter 2 and e.g. those developed by Borgo (2007) and Walther et al. (2007).

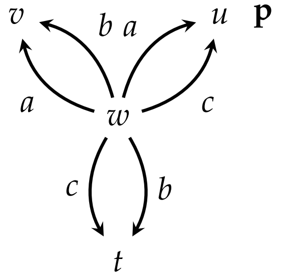
Definition 3.5 (ABC) An action-based coalitional model (ABC model) indexed by a finite set of agents $\mathtt{N}$ and a collection of finite sets of actions $(A_j)_{j\in\mathtt{N}}$ is a 5-tuple of the form $\langle W, \mathtt{N}, (\xrightarrow{j,a})_{j\in\mathtt{N}, a\in A_j}, (\leq_j)_{j\in\mathtt{N}}, \mathsf{V}\rangle$, where

- $W \neq \emptyset$ is a set of states,
- $\mathtt{N}$ is a finite set of agents,
- $\xrightarrow{j,a} \subseteq W \times W$ for each $j \in \mathtt{N}, a \in A_j$ represents the effects of agents performing actions: $w \xrightarrow{j,a} v$ means that if in w agent j performs action a then the system might move into state v,
- $\leq_j \subseteq W \times W$ is a TPO for each $j \in \mathtt{N}$, which represents agent j's preferences over the states: $w \leq_j v$ means that j finds v at least as good as w,
- $\mathsf{V} : \textsc{prop} \cup \textsc{nom} \to \wp(W)$ is a valuation function with $|\mathsf{V}(i)| = 1$ for each $i \in \textsc{nom}$. ◀

Let us now look at how the ability of agents is modeled in action-based coalitional models. At a state w, agent j can guarantee by doing a that the next state is one of the states in $\xrightarrow{j,a} [w]$. In general, at w agent j can guarantee that the next state is in $X \subseteq W$ if and only if for some $a \in A_j$, we have that $\xrightarrow{j,a} [w] \subseteq X$. We say that X is in the *exact power* of j at w if for some $a \in A_j$, it holds that $\xrightarrow{j,a} [w] = X$.

Power of individuals extends to power of coalitions as follows. Let $C = \{j_1, \ldots, j_{|C|}\}$. Then, at w, $C \subseteq \mathtt{N}$ can force the next state to be in $\{\bigcap_{j\in C} \xrightarrow{j,a_j} [w] \mid (a_1, \ldots, a_{|C|}) \in \times_{j\in C} A_j\}$. Again, $X \subseteq W$ is said to be in the exact power of coalition C at w if $X \in \{\bigcap_{j\in C} \xrightarrow{j,a_j} [w] \mid (a_1, \ldots, a_{|C|}) \in \times_{j\in C} A_j\}$. Note that as opposed to the previous class of models ($\wp(\mathtt{N})-\mathtt{LTS}$), with the above definitions, in action-based coalitional models powers are additive in the sense that powers of coalitions arise from the powers of individuals.

Example 3.6 Consider the following ABC model with $\mathtt{N} = \{1, 2\}$, actions $A = \{a, b, c\}$ and $A_1 = \{a, b\}, A_2 = \{c\}$.



In the model, we can see what are the effects of actions being performed in state w. E.g., if action a is executed, the system will move either to u or to v. At state w the coalition $\{1, 2\}$ can make the system move into u, by 1 doing a and 2 doing c. ◀

The definition of ABC models is very general and allows e.g. for states in which there are no outgoing transitions for any action. We will now consider some reasonable assumptions on ABC models

Definition 3.7 We say that an ABC model is *reactive* if the following condition is fulfilled:

- For any $(a_j)_{j\in\mathtt{N}} \in \times_{j\in N}(A_j)$, and for all w, $\bigcap_{j\in\mathtt{N}} \xrightarrow{j,a_j} [w] \neq \emptyset$, i.e., for every collective choice of actions there is some next state.

We say that an ABC model is N-*determined* if for all $w \in W$ and all action profiles $(a_j)_{j\in\mathtt{N}} \in \times_{j\in\mathtt{N}}(A_j)_j$, we have that $| \bigcap_{j\in\mathtt{N}} \xrightarrow{j,a_j} [w] \,| = 1$.
ABC^{NR} denotes the class of N-*determined* reactive ABC models.

Thus, in reactive ABC models, in every state agents have available actions and there is always a successor state. ABC models are N-determined if the next state is completely determined by the choice of the grand coalition.

To summarize, in ABC^{NR} models, agents each have a set of actions from which they can choose. The choices of all the agents then completely determine the next state of the system. Thus, as opposed to the models of CLA+P discussed in the previous chapter, the grand coalition here has complete control of the system.

Language interpreted on ABC models.

We now give the language for reasoning about ABC models. More precisely, we have a family of languages indexed by collections $(A_j)_{j\in\mathtt{N}}$. We start with the basic language $\mathcal{L}_{\mathtt{ABC}}$, which is defined as follows.

$$\alpha ::= \leq_j \ | \, a_j$$

$$\varphi ::= \; p \mid \neg\varphi \mid \varphi \wedge \varphi \mid \langle\alpha\rangle\varphi$$

Then, the first extension we consider allows to talk about the intersection of actions and preferences. This language $\mathcal{L}_{\text{ABC}}(\cap)$ is defined just as the basic language $\mathcal{L}_{\text{ABC}}$ with also allowing $\alpha \cap \alpha$ inside the modalities.

The first hybrid extension we need later is the extension of the basic language with the binder $\downarrow$. This language $\mathcal{H}_{\mathcal{L}_{\text{ABC}}}(\downarrow)$ is defined as follows.

$$\alpha ::= \leq_j \; \mid a_j$$

$$\varphi ::= \; p \mid i \mid x \mid \neg\varphi \mid \varphi \wedge \varphi \mid \langle\alpha\rangle\varphi \mid \; \downarrow x.\varphi.$$

Extending it with an intersection modality, we then get $\mathcal{H}_{\mathcal{L}_{\text{ABC}}}(\downarrow, \cap)$ with

$$\alpha ::= \leq_j \; \mid a_j \mid \alpha \cap \alpha$$

$$\varphi ::= \; p \mid i \mid x \mid \neg\varphi \mid \varphi \wedge \varphi \mid \langle\alpha\rangle\varphi \mid \; \downarrow x.\varphi.$$

Analogously, we have $\mathcal{H}_{\mathcal{L}_{\text{ABC}}}(\downarrow, {}^{-1})$:

$$\alpha ::= \leq_j \; \mid a_j \mid \alpha^{-1}$$

$$\varphi ::= \; p \mid i \mid x \mid \neg\varphi \mid \varphi \wedge \varphi \mid \langle\alpha\rangle\varphi \mid \; \downarrow x.\varphi$$

We will also use a full hybrid extension with intersection.
$\mathcal{H}_{\mathcal{L}_{\text{ABC}}}(@, \downarrow, \cap)$ is defined as follows

$$\alpha ::= \leq_j \; \mid a_j \mid \alpha \cap \alpha$$

$$\varphi ::= \; p \mid i \mid x \mid \neg\varphi \mid \varphi \wedge \varphi \mid \langle\alpha\rangle\varphi \mid @_i\varphi \mid @_x\varphi \mid \; \downarrow x.\varphi$$

where $j \in \mathtt{N}$, $a_j \in A_j$ (the set of actions available to j) and $i \in \textsc{nom}, p \in \textsc{prop}$.

The modality $\langle a_j\rangle$ runs along the relation $\xrightarrow{j,a}$. The other relations are defined as for LTS.

$$\begin{array}{lcl}
R_{a_j} & = & \xrightarrow{j,a} \\
R_{\leq_j} & = & \leq_j \\
R_{\alpha\cap\beta} & = & R_\alpha \cap R_\beta \\
R_{\alpha^{-1}} & = & \{(v, w) \mid wR_\alpha v\}
\end{array}$$

The semantics is now defined in the standard way.

$$\begin{array}{lll}
\mathcal{M}, w, g \models p & \text{iff} & w \in \mathsf{V}(p) \\
\mathcal{M}, w, g \models i & \text{iff} & \{w\} = \mathsf{V}(i) \\
\mathcal{M}, w, g \models x & \text{iff} & w = g(x) \\
\mathcal{M}, w, g \models \langle\alpha\rangle\varphi & \text{iff} & \exists v : wR_\alpha v \text{ and } \mathcal{M}, v, g \models \varphi \\
\mathcal{M}, w, g \models @_i\varphi & \text{iff} & \mathcal{M}, v, g \models \varphi \text{ for } \mathsf{V}(i) = \{v\} \\
\mathcal{M}, w, g \models @_x\varphi & \text{iff} & \mathcal{M}, g(x), g \models \varphi \\
\mathcal{M}, w, g \models \downarrow x.\varphi & \text{iff} & \mathcal{M}, w, g[x := w] \models \varphi
\end{array}$$

We will make use of some shortcuts when writing big conjunctions/disjunctions or intersections/unions. For $C \subseteq \mathtt{N}$, we let $\vec{C} := \times_{j \in C} A_j$. For an action profile $\vec{a_j} = (a_j)_{j \in C} \in \vec{C}$ we often write $\bigcap \vec{a_j}$ to stand for $\bigcap_{j \in C} a_j$. As an example, for the language indexed by $A_1 = \{T_1, M_1, B_1\}$ and $A_2 = \{L_2, R_2\}$ instead of writing $[T_1 \cap L_2]p \vee [M_1 \cap L_2]p \vee [B_1 \cap L_2]p \vee [T_1 \cap R_2]p \vee [M_1 \cap R_2]p \vee [B_1 \cap R_2]p$, we often write $\bigvee_{\vec{a_j} \in \vec{\{1,2\}}} [\cap \vec{a_j}]p$.

Thus, the basic modalities of the hybrid Boolean modal language for ABC run along the accessibility relations for individual actions and preferences.

Axiomatization

The axiomatization of the basic modal logic on ABC models is just as the one for the class of LTS. We have **K** and **Dual** for each coalition modality and the preference fragment is axiomatized by **S4.3**. The rules of inference are again modus ponens, uniform substitution and necessitation.

The extensions are axiomatized in the standard way.

3.2.3 Power-based coalitional models

We will now focus on approaches that are taking coalitional power itself as a primitive and use formal systems specifically designed to model coalitional power. The best known of such modal systems is CL. As CL uses neighborhood semantics, we will not work with CL itself in this chapter but rather consider its normal simulation NCL, which uses Kripke models. This then makes a systematic comparison with the two previously discussed approaches (LTS and ABC) easier. We will now first give an overview of NCL (Broersen et al. 2007), briefly discuss the computational properties of the logic and then present a generalization of this approach, which we will then work with in this chapter.

Normal Coalition Logic

We give a brief overview of *Normal Coalition Logic* (NCL). Broersen et al. (2007) show that CL can be simulated by a normal modal logic which is based on a combination of STIT with a temporal modality. The definitions we present are equivalent to those given in Broersen et al. (2007), adapted to our notation.

Definition 3.8 (NCL model) An NCL model is defined to be a 5-tuple of the form $\langle W, \mathtt{N}, (\sim_C)_{C \subseteq \mathtt{N}}, F_{\mathsf{X}}, \mathsf{V} \rangle$, where

- W is a set of states,
- N is a finite set of agents,

- for each $C \subseteq \mathtt{N}$, $\sim_C \subseteq W \times W$ is an equivalence relation, satisfying the following conditions.
  1. for all $C, D \subseteq \mathtt{N}$, $\sim_{C \cup D} \subseteq \sim_C$,
  2. $\sim_{\mathtt{N}} = Id = \{(w, w) \mid w \in W\}$,
  3. NCL-Independence: for all $C \subseteq \mathtt{N}$, $\sim_\emptyset \subseteq (\sim_C \circ \sim_{\overline{C}})$; for $\circ$ being the composition of relations.
- $F_{\mathsf{X}} : W \to W$ is a total function,
- $\mathsf{V} : \textsc{prop} \to \wp(W)$ is a valuation function. ◀

Let us take a closer look at how the ability of groups is modeled in NCL. The equivalence classes of $\sim_C$ represent different choices of coalition C. Each equivalence class of $\sim_\emptyset$ represents one situation. The equivalence classes of $\sim_C$ inside such a class are the choices C has in that situation.

The function F_{X} gives the outcomes (the next state) resulting from the choices of the agents. The three additional assumptions on the equivalence relations correspond to natural properties about coalitional power of different coalitions. We will now give the intuition behind each of the three conditions.

1. for all $C, D \subseteq \mathtt{N}$, $\sim_{C \cup D} \subseteq \sim_C$.

   This condition says that the choices of a coalition are at least as refined as the choices of its subcoalitions. The power of a coalition does not decrease as the coalition gets bigger. This corresponds to the property of *Coalition Monotonicity*.

2. $\sim_{\mathtt{N}} = Id = \{(w, w) \mid w \in W\}$.

   The equivalence relation of the grand coalition being the identity relation means that the grand coalition completely determines what will be the next state (which is then given by F_{X}). Put differently, once all agents have made their choices, the outcome is completely determined. This corresponds to N-maximality. Remember that also N-determined ABC models have such a property.

3. NCL-Independence: for all $C \subseteq \mathtt{N}$, $\sim_\emptyset \subseteq (\sim_C \circ \sim_{\overline{C}})$.

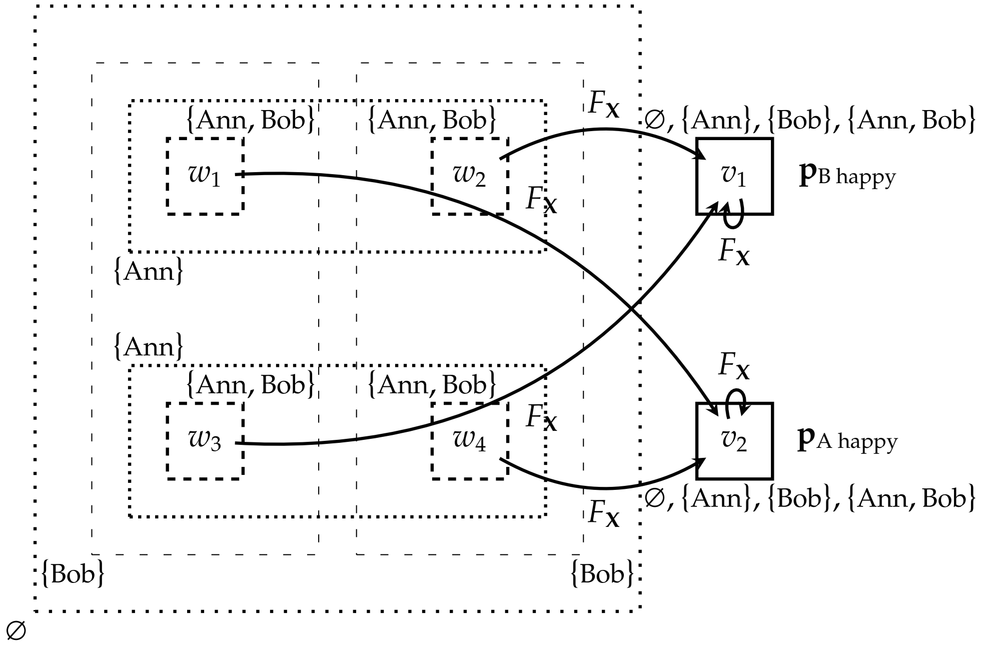
   The states in an equivalence class $[w]_{\sim_\emptyset}$ each represent a possible collective choices of all agents together. The condition then says that each of these collective choices can be achieved by *independent* choices of a coalition and its complement.

We will now give an example in which we construct an NCL model from a simple strategic game.

		Bob	
		Heads	*Tails*
Ann	*Heads*	1, −1	−1, 1
	Tails	−1, 1	1, −1

Example 3.9 Consider the *matching pennies* game between Ann and Bob.

We can take two propositional letters $p_{\mathrm{A\,happy}}$ and $p_{\mathrm{B\,happy}}$. Now. we can construct an NCL model representing the situation in which Ann and Bob play the games once and after stay in a state in which the winner is happy and the loser is not. The NCL model is depicted below.



◀

Definition 3.10 The language $\mathcal{L}_{\mathtt{NCL}}$ of NCL is given by

$$\varphi ::= p \mid \neg\varphi \mid \varphi \wedge \varphi \mid [C]\varphi \mid \mathbf{X}\varphi \ .$$

where $j \in \mathtt{N}$, $C \in \wp(\mathtt{N})$ and $p \in \textsc{prop}$. ◀

The modalities $[C]$ run along $\sim_C$ and $\mathbf{X}$ runs along $F_{\mathbf{X}}$.

Definition 3.11 Propositional formulas and Boolean connectives are interpreted in the standard way and for the the modalities we have:

$$\begin{array}{lll} \mathcal{M}, w \models [C]\varphi & \text{iff} & \forall v: \text{ if } w \sim_C v \text{ then } \mathcal{M}, v \models \varphi \\ \mathcal{M}, w \models \mathbf{X}\varphi & \text{iff} & \mathcal{M}, F_{\mathbf{X}}(w) \models \varphi. \end{array}$$

◀

Theorem 3.12 (Broersen et al. (2007)) *The logic* NCL *based on the axiom schemes for propositional logic, S5 schemes for every* $[C]$*, additional axioms listed below and rules of inference modus ponens and necessitation is sound and complete with respect to the class of* NCL *models.*

$$\begin{array}{ll}
C-\textbf{Mon}. & [C]\varphi \to [C\cup D]\varphi \\
\textbf{Elim}([\emptyset]) & \langle\emptyset\rangle\varphi \to \langle C\rangle\langle\overline{C}\rangle\varphi \\
\textbf{Triv}([\mathtt{N}]) & \varphi \to [\mathtt{N}]\varphi \\
\textbf{K}(\mathbf{X}) & \mathbf{X}(\varphi\to\psi)\to(\mathbf{X}\varphi\to\mathbf{X}\psi) \\
\textbf{D}(\mathbf{X}) & \mathbf{X}\varphi\to\neg\mathbf{X}\neg\varphi \\
\textbf{Det}(\mathbf{X}) & \neg\mathbf{X}\neg\varphi\to\mathbf{X}\varphi.
\end{array}$$ ◀

In Schwarzentruber (2007) and Balbiani et al. (2008b) it is shown that the satisfiability problem of NCL is NEXPTIME-complete. This high complexity results from the conditions on the equivalence relations which force the models to be grid-like. As in this chapter we want to determine how much complexity is required for being able to express different properties on power-based coalitional models, we will drop the assumption that forces the grid-like models and consider a more general class of power-based coalitional models whose basic logic is of lower complexity. The class of models we consider is basically just as NCL models but only requires the relations $\sim_C$ to be equivalence relations, without any further requirements. We also add preference relations for the individual agents just as we have done for for the coalition-labeled transition systems and action-based coalitional models.

Power-based coalitional models

Definition 3.13 (PBC model) A power-based coalitional model (PBC model) indexed by a finite set of coalitions $\wp(\mathtt{N})$) is a 6-tuple of the form $\langle W, \mathtt{N}, (\sim_C)_{C\subseteq\mathtt{N}}, F_{\mathtt{X}}, (\leq_j)_{j\in\mathtt{N}}, \mathsf{V}\rangle$, where

- $W \neq \emptyset$ is a set of states,
- N is a finite set of agents,
- for each $C \subseteq \mathtt{N}$, $\sim_C\subseteq W\times W$ is an equivalence relation,
- $F_{\mathtt{X}} : W \to W$ is a total function,
- $\leq_j\subseteq W\times W$ is a TPO for each $j\in\mathtt{N}$, which represents agent j's preferences over the states: $w\leq_j v$ means that j finds v at least as good as w,
- V : PROP ∪ NOM → $\wp(W)$ is a valuation function with $|\mathsf{V}(i)| = 1$ for each $i\in$ NOM. ◀

Language interpreted on PBC models

We extend $\mathcal{L}_{\texttt{NCL}}$ with preference modalities, one for each agent, and obtain our basic language $\mathcal{L}_{\texttt{PBC}}$. Note that the fact that we use $\langle C \rangle$ here as a primitive instead of the box is for technical convenience only.

$$\varphi ::= \; p \mid \neg\varphi \mid \varphi \wedge \varphi \mid \langle C \rangle\varphi \mid \mathbf{X}\varphi \mid \langle \leq_j \rangle\varphi.$$

The first hybrid extension we will use is $\mathcal{H}_{\mathcal{L}_{\texttt{PBC}}}(\downarrow)$, which is defined as follows.

$$\varphi ::= \; p \mid i \mid x \mid \neg\varphi \mid \varphi \wedge \varphi \mid \langle C \rangle\varphi \mid \mathbf{X}\varphi \mid \downarrow x.\varphi$$

We also use the full hybrid extension $\mathcal{H}_{\mathcal{L}_{\texttt{PBC}}}(@, \downarrow)$:

$$\varphi ::= \; p \mid i \mid x \mid \neg\varphi \mid \varphi \wedge \varphi \mid \langle C \rangle\varphi \mid \mathbf{X}\varphi \mid @_i\varphi \mid @_x\varphi \mid \downarrow x.\varphi$$

and the binder-extension with a converse modality for preferences $\mathcal{H}_{\mathcal{L}_{\texttt{PBC}}}(\;^{-1}, \downarrow)$.

$$\alpha ::= \leq_j \mid \alpha^{-1}$$

$$\varphi ::= \; p \mid i \mid x \mid \neg\varphi \mid \varphi \wedge \varphi \mid \langle C \rangle\varphi \mid \mathbf{X}\varphi \mid \langle \alpha \rangle\varphi \mid \downarrow x.\varphi$$

It is important to note that, as opposed to the languages we defined for LTS and ABC, we now only take the Boolean modal extension with respect to the preferences.

$$\begin{array}{rcl} R_{\leq_{\mathrm{j}}} & = & \leq_j \\ R_{\alpha^{-1}} & = & \{(v, w) \mid wR_\alpha v\} \end{array}$$

Now, the semantics is defined as follows. $\langle C \rangle$ runs along $\sim_C$ and $\mathbf{X}$ runs along F_{X}.

$$\begin{array}{lll} \mathcal{M}, w, g \models \langle C \rangle\varphi & \text{iff} & \exists v : w \sim_C v \text{ and } \mathcal{M}, v, g \models \varphi \\ \mathcal{M}, w, g \models \mathbf{X}\varphi & \text{iff} & \mathcal{M}, F_{\mathsf{X}}(w), g \models \varphi \\ \mathcal{M}, w, g \models \langle \alpha \rangle\varphi & \text{iff} & \exists v : wR_\alpha v \text{ and } \mathcal{M}, v, g \models \varphi \end{array}$$

In NCL, the power of a coalition to force the system into some set of states involves a combination of the equivalence relations for the empty set, the equivalence relation for the coalition itself, and also the outcome function F_{X}. In principle, we could thus also consider Boolean extensions that allow e.g. for taking intersections of preferences and coalition relations, or of preferences and the outcome function. These are however conceptually harder to motivate and to understand. Moreover, we will see that Boolean combinations of preferences alone are sufficient for expressing interesting concepts on power-based coalitional models.

Axiomatization

Definition 3.14 The axioms of the basic system for PBC models contain axiom schemes for propositional logic, $S5$ schemes for every $[C]$ and the axioms listed below.

$$\begin{array}{ll}
\mathbf{K(X)} & \mathbf{X}(\varphi \to \psi) \to (\mathbf{X}\varphi \to \mathbf{X}\psi) \\
\mathbf{D(X)} & \mathbf{X}\varphi \to \neg\mathbf{X}\neg\varphi \\
\mathbf{Det(X)} & \neg\mathbf{X}\neg\varphi \to \mathbf{X}\varphi.
\end{array}$$

and additionally **S4.3** for the preference fragment. The rules of inference are modus ponens and necessitation. ◀

In this section, we have introduced three classes of normal modal logic systems for reasoning about the abilities of agents. In Section 3.3, we will then evaluate the models and determine how much expressive power and complexity is required for reasoning about cooperative ability on each of the classes. We start by clarifying the relationship between the different models.

3.3 Comparing modal logics for cooperation

This section gives the main results of this chapter. First we clarify the relation between the classes of models we introduced; then we analyze how demanding different concepts from game theory and social choice theory are on them. We start by analyzing coalitional power as modeled in the frameworks of PBC, NCL and CL and also determine the precise relationship between different standard assumptions on coalitional power. For this, we take a purely semantic perspective. We note that determining the relationship between different assumptions about coalitional power could also have been done using (extended) modal languages that can express these assumptions and then use the logics themselves to prove certain dependencies between them.

3.3.1 Coalitional power in power-based coalitional models

We now take a closer look at properties of coalitional ability as it is modeled in power-based coalitional models. We investigate the relationship between some cooperation-specific assumptions that can be made on the models. On PBC models, we say that a coalition C can *force* a set X at w iff at w it is the case that C can guarantee that the next state is in X. On these models, this means that there is a state $v \in [w]_{\sim_\emptyset}$, such that for any $v' \in [v]_{\sim_C}$, $F_\mathsf{X}(v') \in X$. So basically, this means that at w, C can choose a $\sim_C$-equivalence class that has a nonempty intersection with the $\sim_\emptyset$-equivalence class of the current state and whose image under F_X is a subset of X.

We say that Y is in the exact power of C at w, which we denote by $Y \in P_C(w)$, if for some v with $w \sim_\emptyset v$, we have that $Y = \{F(v') \mid v \sim_C v'\}$. Thus, C can force X at w iff there is some $Y \in P_C(w)$ with $Y \subseteq X$.

Definition 3.15 For a PBC model $\mathcal{M} = \langle W, \mathtt{N}, (\sim_C)_{C\subseteq \mathtt{N}}, F_{\mathsf{X}}, (\leq_j)_{j\in\mathtt{N}}, \mathsf{V}\rangle$, the set of exact powers of a coalition $C \subseteq \mathtt{N}$ at a state $w \in W$ is defined as follows.

$$P_C(w) = \Big\{ \bigcup_{v'\in[v]_{\sim_C}} F_{\mathsf{X}}(v') \mid w \sim_\emptyset v \Big\}$$

As the definition does not use preferences, we can define the set of exact powers in an NCL model in the same way. ◀

Thus, $P_C(w)$ contains the smallest sets of states coalition C can force the system to move into at w.

Some reasonable assumptions about the coalitional powers reflect the independence of agents and are generally assumed in the literature (e.g. Pauly (2002a); Broersen et al. (2007); Belnap et al. (2001)). We consider three assumptions and show their relation. Independence of coalitions then says that two disjoint coalitions cannot force the system to move into disjoint sets of states.

Definition 3.16 (Independence of coalitions (IC)) For all w, if $C \cap D = \emptyset$ then for all $X \in P_C(w)$ and all $Y \in P_D(w)$ we have that $X \cap Y \neq \emptyset$. ◀

The next condition says that the powers of a coalition and its complement have to be consistent.

Definition 3.17 (Condition about complementary coalitions (CCC)) For all $w \in W$ and all $X \subseteq W$, if there is some X' with $X \supseteq X' \in P_C(w)$, then there is no $Y \subseteq W$ such that $\overline{X} \supseteq Y \in P_{\overline{C}}(w)$. ◀

Coalition monotonicity says that if a coalition can achieve something then so can all supersets of this coalition.

Definition 3.18 (Coalition monotonicity (CM)) For all $w \in W$ and $X \subseteq W$, if $C \subseteq D$ and there is some $Y \subseteq W$ such that $X \supseteq Y \in P_C(w)$, then there is some $Z \subseteq W$ such that $X \supseteq Z \in P_D(w)$. ◀

We now show some results about the connection between the different conditions. The first result says that if for all choices of any two disjoint coalitions, there is a next state then the powers of coalitions and their complements have to be consistent.

Fact 3.19 *IC implies CCC.*

Proof. Let $w \in W, X \subseteq W$. Assume that for some $X' \in P_C(w)$, $X' \subseteq X$. Now suppose that there is some $Y \in P_{\overline{C}}(w)$ such that $Y \subseteq \overline{X}$. But then by IC, since $C \cap \overline{C} = \emptyset$, it follows that $X' \cap Y = \emptyset$ and thus $Y \not\subseteq \overline{X'}$, which contradicts $Y \subseteq \overline{X}$ because $\overline{X} \subseteq \overline{X'}$. ■

Now, we can show that under the assumption of coalition monotonicity, also the converse holds.

Fact 3.20 *CCC and CM together imply IC.*

Proof. Let $C, D \subseteq \mathtt{N}$ such that $C \cap D = \emptyset$, and let $X \in P_C(w)$ for some $w \in W$. Now suppose towards contradiction that there is some $Y \in P_D(w)$ such that $X \cap Y = \emptyset$, i.e., $Y \subseteq \overline{X}$. Since $C \cap D = \emptyset, D \subseteq \overline{C}$. Then by CM, there is some $Z \in P_{\overline{C}}(w)$ with $Z \subseteq \overline{X}$. But this then contradicts CCC. ■

Note that on PBC models, CCC actually says the following: For all $w \in W$ and $X \subseteq W$, if for some $v \in [w]_{\sim_\emptyset}$ we have that $\{F_X(v') \mid v' \in [v]_{\sim_C}\} \subseteq X$, then there is no $\dot{v} \in [w]_{\sim_\emptyset}$ such that $\{F_X(\dot{v}') \mid \dot{v}' \in [\dot{v}]_{\sim_{\overline{C}}}\} \subseteq \overline{X}$.

NCL models have the property of NCL-Independence (Definition 3.8), which says that every collective choice of the grand coalition can also be achieved by independent choices of a coalition and its complement.

We now show that on PBC models with F_X being injective, CCC and NCL-Independence turn out to be equivalent. Injectivity of F_X is needed here in order to get the correspondence between CCC, which is about properties of the sets of states coalitions can force the system to move into (after the application of F_X) and the property of NCL-Independence which is about the partitions from which complementary coalitions can choose and thereby determine a set whose F_X-image are the possible next states.

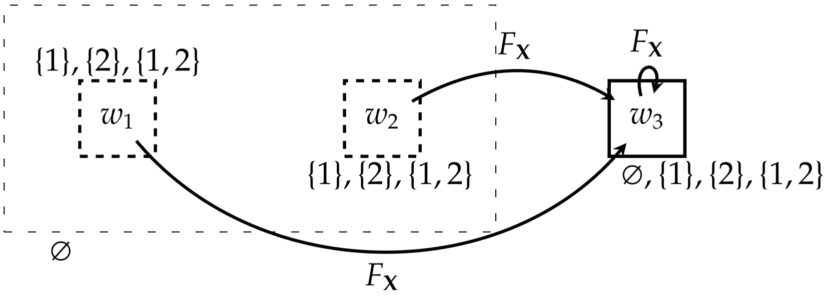
Proposition 3.21 *On* PBC *models with the function* F_X *being injective, CCC is equivalent to* NCL*-Independence.*

Proof. From left to right, assume that NCL-Independence does not hold. Then there is some model such that for two states w, v we have that $w \sim_\emptyset v$ and there is no v' such that $w \sim_C v'$ and $v' \sim_{\overline{C}} v$. Thus, $[w]_{\sim_C} \cap [v]_{\sim_{\overline{C}}} = \emptyset$. Now, since F_X is injective, $\{F(w') \mid w \sim_C w'\} \cap \{F(v') \mid v \sim_{\overline{C}} v'\} = \emptyset$. This then means that $\{F(v') \mid v \sim_{\overline{C}} v'\} \subseteq \overline{\{F(w') \mid w \sim_C w'\}}$, which means that CCC does not hold.

From right to left, assume that CCC does not hold. Then there is a model with some state w and some set of states X such that for some $v \in [w]_{\sim_\emptyset}$ it holds that $\{F_X(v') \mid v \sim_C v'\} \subseteq X$ and there is some $\dot{v} \in [w]_\emptyset$ such that $\{F_X(\dot{v}') \mid \dot{v} \sim_{\overline{C}} \dot{v}'\} \subseteq \overline{X}$. Now, as F_X is injective,it follows from $\{F_X(v') \mid v \sim_C v'\} \cap \{F_X(\dot{v}') \mid \dot{v} \sim_C \dot{v}'\} = \emptyset$ that $[v]_{\sim_C} \cap [\dot{v}]_{\sim_{\overline{C}}} = \emptyset$. Therefore, we have that $v \sim_\emptyset \dot{v}$ holds but it is not the case that $v \sim_C \circ \sim_{\overline{C}} \dot{v}$, which thus means that NCL-Independence does not hold. ■

To illustrate the role of injectivity of F_X in the preceding proposition, we give an example of a PBC model in which F_X is not injective and CCC holds but NCL-Independence does not.

Example 3.22 Consider the PBC model illustrated below. We omit the representation of the agents' preferences as they are irrelevant for this example.



NCL-Independence is violated in w_1 and w_2, as $w_1 \sim_{\emptyset} w_2$ but it is not the case that $w_1 \sim_{\{1\}} \circ \sim_{\{2\}} w_2$, and neither that $w_2 \sim_{\{1\}} \circ \sim_{\{2\}} w_1$. CCC on the other hand is satisfied as in fact for each state it holds that the powers of all coalitions are the same (they all can force $\{w_3\}$). ◄

Let us sum up the preceding results.

- If the *exact* powers of disjoint coalitions are consistent, then so are the powers of complementary coalitions (Fact 3.19).
- Under the assumption that coalitions can at least achieve what their subcoalitions can achieve, the converse of the previous result also holds: If the powers of complementary coalitions are consistent then so are the exact powers of disjoint coalitions (Fact 3.20).
- When F_X is injective, complementary coalitions having consistent powers then means that every possible next state can be the result of complementary coalitions making their independent choices (Proposition 3.21).

Figure 3.1 illustrates the relations between these properties of PBC models. Moreover, CCC and NCL-Independence are actually equivalent if the function F_X is injective.

3.3.2 On the relation between NCL and CL

In order to clarify the relationship between power-based coalitional models and non-normal modal frameworks for cooperation such as CL, we now analyze the relation between CL and the subclass of power-based coalitional models without preferences NCL. In Broersen et al. (2007), a translation τ from $\mathcal{L}_{\mathrm{CL}}$ to

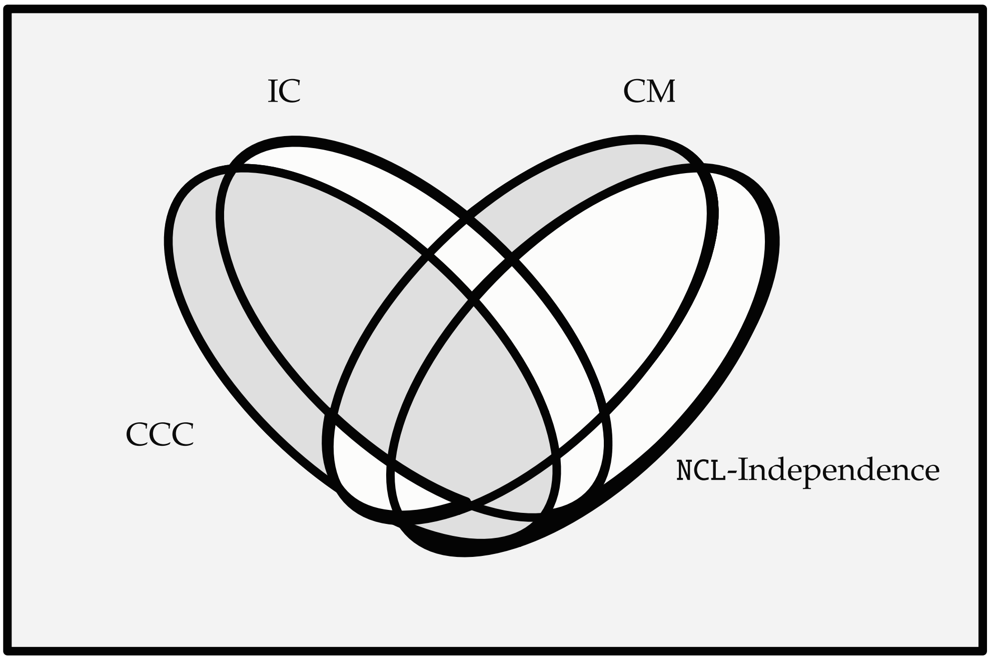


Figure 3.1: Different properties of PBC models. White areas are empty.

$\mathcal{L}_{\texttt{NCL}}$ is given such that for all $\varphi \in \mathcal{L}_{\texttt{CL}}$, φ is satisfiable in a CL model iff $\tau(\varphi)$ is satisfiable in a NCL model. The crucial clauses of the definition of the translation τ are the following.

Definition 3.23 The translation τ from $\mathcal{L}_{\texttt{CL}}$ to $\mathcal{L}_{\texttt{NCL}}$ is defined as follows

$$\tau(p) = p, \qquad \tau(\langle\!\![C]\!\!\rangle\varphi) = \langle\emptyset\rangle[C]\mathbf{X}\tau(\varphi).$$

For Boolean combinations, the translation is defined in the standard way. ◀

The main result is then the following.

$$\varphi \text{ is a theorem of } \texttt{CL} \text{ iff } \tau(\varphi) \text{ is one of } \texttt{NCL}.$$

The right-to-left direction of their proof is constructive, while the left-to-right direction uses completeness of CL and soundness of NCL to show that whenever $\tau(\varphi)$ is satisfied in an NCL model, there is also some CL model that satisfies φ.

We give a *constructive* proof of the right-to-left direction of this result in order to get a clear view of how the two frameworks are related. We give a procedure of how to translate pointed NCL models $(\mathcal{M}, w)$ into CL models $f(\mathcal{M}, w)$ such that for all $\varphi \in \mathcal{L}_{\texttt{CL}}$,

$$(\mathcal{M}, w) \models \tau(\varphi) \text{ iff } f(\mathcal{M}, w) \models \varphi.$$

We insist on the fact that the proposition itself was already proven; the novelty is that we provide a method to construct a $\mathtt{CL}$ model from an $\mathtt{NCL}$-Kripke structure, and thus give a constructive proof of this proposition. This clarifies the relationship between the two logics in purely syntactic terms. We belief that this can have conceptual benefits for understanding the normal simulation of $\mathtt{CL}$. The idea of our proof is as follows. The effectivity functions of the $\mathtt{CL}$ model are constructed such that $E_w(C)$ contains the exact powers of C at w in the $\mathtt{NCL}$ model and also their supersets.

Definition 3.24 We transform $\mathtt{NCL}$ models into $\mathtt{CL}$ models as follows. For $\mathcal{M} = \langle W, \mathtt{N}, (\sim_C)_{C\subseteq \mathtt{N}}, F_{\mathbf{X}}, (\leq_j)_{j\in\mathtt{N}}, \mathsf{V}\rangle$, we define $f(\mathcal{M}) := \langle \mathtt{N}, (W, E), \mathsf{V}\rangle$, where

$$E_w(C) := \{\{Y|Y \supseteq F_{\mathbf{X}}[[w']_{\sim_C}]\}|w' \in [w]_{\sim_\emptyset}\},$$

with $F_{\mathbf{X}}[[w']_{\sim_C}] := \{F_{\mathbf{X}}(w'') \mid w'' \in [w']_{\sim_C}\}$. ◀

We now use the above transformation to give a *constructive* proof of the following proposition by Broersen et al. (2007).

Proposition 3.25 *For all $\varphi \in \mathcal{L}_{\mathrm{CL}}$, if $\tau(\varphi)$ is satisfiable in a pointed model $(\mathcal{M}, w)$ of* $\mathtt{NCL}$*, then φ is satisfiable in a model $f(\mathcal{M}, w)$ of* $\mathtt{CL}$*.*

Proof. We define f as in Definition 3.24. First, we show that E is playable and thus $f(\mathcal{M})$ is a $\mathtt{CL}$ model. Liveness follows from the totality of $F_{\mathbf{X}}$. Termination follows from the closure of $E_w(C)$ under supersets. For $\mathtt{N}$-maximality, let $X \subseteq W$ such that $W \setminus X \notin E_w(\emptyset)$. Then there is some $w' \in [w]_{\sim_\emptyset}$ such that $F_{\mathbf{X}}(w') \in X$. Since $X \supseteq F_{\mathbf{X}}[\{w'\}] = \{F_{\mathbf{X}}(w')\}$, $X \in E_w(\mathtt{N})$. Outcome-monotonicity follows from the closure of $E_w(C)$ under supersets. For Superadditivity, let $X_1, X_2 \subseteq W, C_1, C_2 \subseteq \mathtt{N}$, such that $C_1 \cap C_2 = \emptyset$. Assume that $X_1 \in E_w(C_1)$ and $X_2 \in E_w(C_2)$. Then for all $i \in \{1, 2\}$, $\exists w_i \in [w]_{\sim_\emptyset}$ such that $X_i \supseteq F_{\mathbf{X}}[[w_i]_{\sim_{C_i}}]$. We have that $E_w(C_1 \cup C_2) = \{\{Y|Y \supseteq F_{\mathbf{X}}[[w']_{\sim_{C_1\cup C_2}}]\}|w' \in [w]_{\sim_\emptyset}\}$. Thus, we have to show that $\exists w^+ \in [w]_{\sim_\emptyset} : X_1 \cap X_2 \supseteq F_{\mathbf{X}}[[w^+]_{\sim_{C_1\cup C_2}}]$. We have that $w_1 \sim_\emptyset w_2$. Thus, $w_1 \sim_{C_1} \circ \sim_{\overline{C_1}} w_2$ and since $C_1 \cap C_2 = \emptyset$ and thus $C_2 \subseteq \overline{C_1}$, $\sim_{\overline{C_1}} \subseteq \sim_{C_2}$. Then $w_1 \sim_{C_1} \circ \sim_{C_2} w_2$. Thus, $\exists w^+ : w_1 \sim_{C_1} w^+$ and $w^+ \sim_{C_2} w_2$. Thus, $w^+ \in [w_1]_{\sim_{C_1}} \cap [w_2]_{\sim_{C_2}}$ and therefore $[w^+]_{\sim_{C_1}} = [w_1]_{\sim_{C_1}}$ and $[w^+]_{\sim_{C_2}} = [w_2]_{\sim_{C_2}}$. Since $\sim_{C_1\cup C_2} \subseteq (\sim_{C_1} \cap \sim_{C_2})$, $[w^+]_{\sim_{C_1\cup C_2}} \subseteq [w^+]_{\sim_{C_1}} \cap [w^+]_{\sim_{C_2}}$. Hence, $F_{\mathbf{X}}[[w^+]_{\sim_{C_1\cup C_2}}] \subseteq X_1 \cap X_2$, and thus $X_1 \cap X_2 \subseteq E_w(C_1 \cup C_2)$.

This shows that $f(\mathcal{M})$ is a $\mathtt{CL}$ model. Now, we show by induction that for all $\varphi \in \mathcal{L}_{\mathrm{CL}}$, for an $\mathtt{NCL}$ model $\mathcal{M}$, $\mathcal{M}, w \models \tau(\varphi)$ iff $f(\mathcal{M}, w) \models \varphi$. The interesting case is $\varphi := \langle\!\![C]\!\!\rangle\, \psi$. Let $\mathcal{M}, w \models \langle\emptyset\rangle[C]\mathbf{X}\tau(\psi)$. Then there is some $w' \in [w]_{\sim_\emptyset}$ such that for all $w'' \in [w']_{\sim_C}$, $\mathcal{M}, F_{\mathbf{X}}(w'') \models \tau(\psi)$. By induction hypothesis, $f(\mathcal{M}, F_{\mathbf{X}}(w'')) \models \psi$. Now, $[\![\psi]\!]_{f(\mathcal{M},w)} \in E_w(C)$ follows from the fact that for all $w'' \in [w']_{\sim_C}, f(\mathcal{M}, F_{\mathbf{X}}(w'')) \models \psi$. For the other direction, let $f(\mathcal{M}, w) \models \langle\!\![C]\!\!\rangle\, \psi$. Then, there is some $X \in E_w(C)$ such that $X \subseteq [\![\psi]\!]_{f(\mathcal{M},w)}$. By definition of $f(\mathcal{M}, w)$, there is some $w' \in [w]_{\sim_\emptyset}$ such that $X \supseteq F_{\mathbf{X}}[[w']_{\sim_C}]$. Since by inductive hypothesis, $[\![\tau(\psi)]\!]_{M,w} = [\![\psi]\!]_{f(\mathcal{M},w)}$, $X \subseteq [\![\tau(\psi)]\!]_{M,w}$. Hence, $\mathcal{M}, w \models \langle\emptyset\rangle[C]\mathbf{X}\tau(\psi)$. ■

So, we have shown how to transform `NCL` models into corresponding `CL` models by transforming the partitions of the equivalence relations $\sim_C$ and the function $F_{\mathtt{X}}$ together into corresponding effectivity functions. The key of the transformation is to construct the effectivity functions from the exact powers and their supersets. Our proof thus sheds some light on the relation between `CL` and its normal simulation by clarifying the relationship between the semantic structures of both logics.

3.3.3 Coalitional power in action-based coalitional models

Let us now take a closer look at coalitional power as it is modeled by action-based coalitional models. In this section, we will try to position action-based coalitional models with respect to power-based approaches. This way we can clarify how coalitional power is made explicit in `ABC`, and also determine the role of some natural assumptions on `ABC` models such as being `N`-determined or reactive.

In order to determine the relationship between power– and action-based coalitional models, we will show how to construct a power-based coalitional model from a given action-based coalitional model in such a way that coalitions have the same powers in the models.

Definition 3.26 For every pointed $\mathtt{ABC}^{\mathtt{N}R}$ model $(\mathcal{M}, w)$ which is given by $\mathcal{M} = \langle W, \mathtt{N}, (\xrightarrow{j,a})_{j\in\mathtt{N}, a\in A_j}, (\leq_j)_{j\in\mathtt{N}}, \mathsf{V}\rangle$, and $w \in W$, we construct the following `PBC` model $\mathcal{M}'$. $\mathcal{M}' = \langle W', \mathtt{N}, (\sim_C)_{C\subseteq\mathtt{N}}, F_{\mathtt{X}}, (\leq'_j)_{j\in\mathtt{N}}, \mathsf{V}'\rangle$, where

- $W' = W \times \vec{\mathtt{N}}$,
- $(w, (a_j)_{j\in\mathtt{N}}) \sim_C (v, (a'_j)_{j\in\mathtt{N}})$ iff $w = v$ and for all $j' \in C$, $((a_j)_{j\in\mathtt{N}})_{j'} = ((a'_j)_{j\in\mathtt{N}})_{j'}$,
- $F_{\mathtt{X}}((w, (a_j)_{j\in\mathtt{N}})) = (v, (a_j)_{j\in\mathtt{N}}))$ for $\{v\} = \bigcap_{j\in\mathtt{N}} \xrightarrow{j,a_j} [w]$,
- $(w, (a_{j'})_{j'\in\mathtt{N}}) \leq_j (v, (a'_{j'})_{j'\in\mathtt{N}})$ iff $w \leq_j v$,
- for each $p \in \textsc{prop}$, $\mathsf{V}'(p) = \mathsf{V}(p) \times \vec{\mathtt{N}}$. ◀

The reason why we only transform reactive `N`-determined `ABC` models instead of arbitrary `ABC` models is that the two conditions help to define the total function $F_{\mathtt{X}}$.

Fact 3.27 *If $\mathcal{M}$ is reactive and `N`-determined, $\mathcal{M}'$ is an `NCL` model extended with preferences.*

Proof. We first show that $\mathcal{M}'$ is indeed a proper `PBC` model. It is easy to see that $\sim_C$ is an equivalence relation for each C. The fact that $F_{\mathtt{X}}$ is a function follows from $\mathcal{M}$ being $\mathtt{N}$-determined. F_{X} being total follows from $\mathcal{M}$ being reactive. As for every $j \in \mathtt{N}$ the preference relation $\leq_j$ is a total preorder, this also holds for each $\leq'_j$. Lastly, we note that V' is a proper valuation function but it will have to be adapted for nominals, to ensure that they cannot be true in more than one state. This could be done by adding for each nominal i a new nominal $i_{(a_j)_{j\in\mathtt{N}}}$ for each $(a_j)_{j\in\mathtt{N}} \in \vec{\mathtt{N}}$ and setting $\mathsf{V}'(i_{(a_j)_{j\in\mathtt{N}}}) = \{(w,(a_j)_{j\in\mathtt{N}})\}$, for $\mathsf{V}(i) = \{w\}$.

We now show that indeed $\mathcal{M}'$ is an $\mathtt{NCL}$ model (extended with preferences). We show that $\mathcal{M}'$ satisfies the three additional conditions on the powers of coalitions.

1. for all $C, D \subseteq \mathtt{N}$, $\sim_{C\cup D} \subseteq \sim_C$
2. $\sim_{\mathtt{N}} = Id = \{(w,w) \mid w \in W'\}$.
3. $\mathtt{NCL}$-Independence: for all $C \subseteq \mathtt{N}$, $\sim_\emptyset \subseteq (\sim_C \circ \sim_{\overline{C}})$,

The first two follow immediately from the definition of $\sim_C$ for each $C \subseteq \mathtt{N}$. For $\mathtt{NCL}$-Independence let us first look at how $\sim_\emptyset$ is defined in $\mathcal{M}'$. From our definition it follows that $(w,(a_j)_{j\in\mathtt{N}}) \sim_\emptyset (v,(a'_j)_{j\in\mathtt{N}})$ iff $w = v$. So, take two states $(w,(a_j)_{j\in\mathtt{N}}),(w,(a'_j)_{j\in\mathtt{N}}) \in W'$. Then it follows that these are also related by $\sim_C \circ \sim_{\overline{C}}$ because $(w,(a_j)_{j\in\mathtt{N}}) \sim_C (w,(a''_j)_{j\in\mathtt{N}})$ with $a''_j = a_j$ for all $j \in C$ and $a''_j = a'_j$ for all $j \in \overline{C}$. Analogously, $(w,(a''_j)_{j\in\mathtt{N}}) \sim_{\overline{C}} (w,(a'_j)_{j\in\mathtt{N}})$. Hence, $(w,(a_j)_{j\in\mathtt{N}}) \sim_C \circ \sim_{\overline{C}} (w,(a'_j)_{j\in\mathtt{N}})$. Thus, $\mathtt{NCL}$-Independence holds, which concludes the proof that $\mathcal{M}'$ is an $\mathtt{NCL}$ model with preferences. ■

Now, by definition of $\sim_C$ and F_{X} we immediately get the following fact.

Fact 3.28 *Let $C \neq \emptyset$. Then X is in the exact power of C at w if and only if in $\mathcal{M}'$ it holds that $X \times \vec{\mathtt{N}}$ is in the exact power of C at every state in $\{w\} \times \vec{\mathtt{N}}$.* ◀

Note that we focused purely on the semantic relationship between $\mathtt{NCL}$ and $\mathtt{ABC}$ with respect to the powers of coalitions. On a more syntactic level, we thus have a correspondence between formulas of the following form

Fact 3.29

$$\mathcal{M}, w \models \bigvee_{\vec{a_j}\in\vec{C}} \Big[\bigcap \vec{a_j}\Big]p \ \textit{ iff for all } (a_j)_{j\in\mathtt{N}} \in \vec{\mathtt{N}},\ \ \mathcal{M}', (w,(a_j)_{j\in\mathtt{N}}) \models \langle\emptyset\rangle[C]\mathbf{X}p.$$

◀

After we have clarified the relationship between different approaches to modeling cooperative ability of agents, we will now evaluate how much complexity and expressive power is needed for reasoning about interesting concepts.

3.3.4 What game– and social choice theoretical notions demand: complexity and expressivity.

This section analyzes the complexity of describing and reasoning about interactive systems from an abstract perspective. We summarize the main results that we obtained when investigating how much expressive power and complexity is required for reasoning about coalitional power. As mentioned before, we obtain our results by determining under which operations on models certain properties are invariant.

All the properties that we will discuss are definable in first-order logic with one free variable. Table 3.1 gives an overview of the characterization results that we use.

For the definitions of these operations and a detailed discussion of the underlying characterization results, the reader is referred to Blackburn et al. (2001) and ten Cate (2005). Figure 3.2 illustrates the expressive power hierarchy of some extended modal logics.

Invariance	**Modal Language**
Bisimulation	basic modal language $\mathcal{L}_{\mathbf{ML}}$, (van Benthem 1976)
$\cap$-bisimulation	$\mathcal{L}_{\mathbf{ML}}(\cap)$
generated submodels	$\mathcal{H}_{\mathcal{L}_{\mathrm{ML}}}(\downarrow, @)$, (Feferman 1969; Areces et al. 2001)

Table 3.1: Characterization Results.

Table 3.2 summarizes the results we use for upper bounds on the complexity of modal logics with different expressive powers. $\mathcal{H}_{\mathcal{L}_{\mathrm{ML}}}(\downarrow) - \Box \downarrow \Box$ denotes the fragment of $\mathcal{H}_{\mathcal{L}_{\mathrm{ML}}}(\downarrow)$ without occurrences of alternations of the form $\Box \downarrow \Box$, where $\Box$ stands for a box-modality. Note that these complexity results (the upper bounds) also transfer to the classes of LTS, ABC and PBC. For NCL however, this is not the case as already its basic logic has a higher complexity. As already mentioned, all properties that we discuss are definable in first-order logic, and therefore the data complexity of checking whether the property holds in a given model is in LOGSPACE.

We start our analysis with the simplest notions of coalitional power and preferences.

Simple coalitional power and preference

The property of a coalition C having the power to ensure that in the next state some proposition p will be the case turns out to be invariant under bisimulation

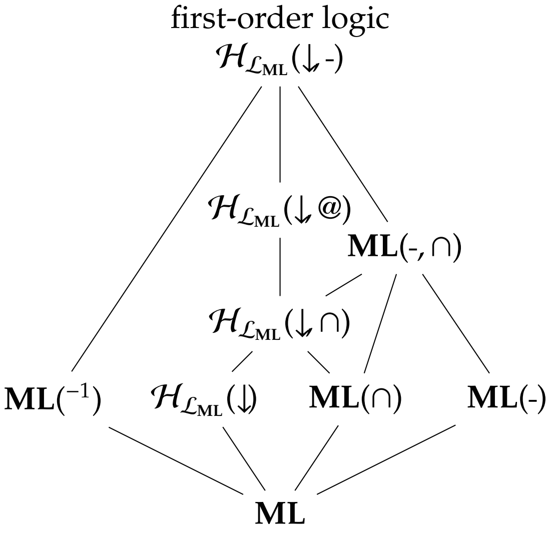

first-order logic
$\mathcal{H}_{\mathcal{L}_{\text{ML}}}(\downarrow, \text{-})$
$\mathcal{H}_{\mathcal{L}_{\text{ML}}}(\downarrow, @)$
$\mathbf{ML}(\text{-}, \cap)$
$\mathcal{H}_{\mathcal{L}_{\text{ML}}}(\downarrow, \cap)$
$\mathbf{ML}(^{-1})$
$\mathcal{H}_{\mathcal{L}_{\text{ML}}}(\downarrow)$
$\mathbf{ML}(\cap)$
$\mathbf{ML}(\text{-})$
$\mathbf{ML}$

Figure 3.2: Expressive power hierarchy of some extended modal logics.

on LTS, PBC and NCL. Let us briefly argue why this is the case. On LTS, the property simply says that there is a $\xrightarrow{C}$-successor which satisfies the proposition p. On PBC it means the following. From the current state there is a $\sim_{\emptyset}$-accessible state such that in all states $\sim_C$-accessible from that one it holds that the $F_{\mathbf{X}}$-image is a state that satisfies p. Even though this might look more complicated than for LTS, note that we just needed existential and universal quantification over accessible states.

Thus, we can conclude that for LTS, PBC and NCL the property of a coalition having the power to force that in the next state p holds can be expressed using the respective basic multi-modal languages. To be more precise, we can express this property by $\langle C \rangle p$ on LTS and by $\langle \emptyset \rangle [C] \mathbf{X} p$ on PBC and NCL. For LTS and PBC, we thus get PSPACE and P as upper bounds on SAT and MC of logics expressing the property of C being able to achieve that p. For NCL, the upper bound for SAT are much higher: NEXPTIME (Schwarzentruber 2007). For MC on the other hand, there is no difference as this can be done using a model checker for the basic multi-modal logic as the models of NCL are just a special kind of multi-agent Kripke structures. Hence, we have a P upper bound for MC of NCL.

On ABC models on the other hand, saying that a coalition can achieve something requires the intersection of the relations for the actions for the agents. This is because C being able to force the system into a p-state means that there are actions for each member of C such that all states accessible by the intersection of these action relations satisfy p. It is not invariant under bisimulation (see Fact 3.30) but it is under $\cap$-bisimulation, a bisimulation that

	Complexity of the logic	
Language	MC	SAT
$\mathcal{L}_{\mathbf{ML}}$	P (Fischer and Ladner 1979b)	PSPACE (Ladner 1977)
$\mathcal{L}_{\mathbf{ML}}(\cap)$	P (Lange 2006)	PSPACE (Donini et al. 1991)
$\mathcal{H}_{\mathcal{L}_{\mathbf{ML}}}(\downarrow) - \Box \downarrow \Box$	PSPACE (Franceschet and de Rijke 2003)	EXPTIME (ten Cate and Franceschet 2005)
$\mathcal{H}_{\mathcal{L}_{\mathbf{ML}}}(\downarrow, @)$	PSPACE (Franceschet and de Rijke 2003)	Π^0_1 (ten Cate and Franceschet 2005)
$\mathcal{H}_{\mathcal{L}_{\mathbf{ML}}}(\downarrow, @, ^{-1})$	PSPACE	Π^0_1 (ten Cate and Franceschet 2005)

Table 3.2: Complexity for different modal logics.

also checks for the intersection of the relations. Thus it can be expressed in the basic language with intersection. This can be done using the formula $\bigvee_{\vec{a_j}\in\vec{C}}[\bigcap \vec{a_j}]p$. Even though we need more expressive power here (as we need the intersection modality) than on the other models, the upper bounds on SAT and MC that we obtain are still PSPACE for SAT and P for MC, respectively. Thus, we can add intersection to the language without increasing the complexity. To summarize, the ability of a group to force the system into a state where some proposition holds is invariant under bisimulation on the models of LTS and PBC. This implies that the property can thus be expressed in the basic language on these models. On ABC, the property is invariant under $\cap$-bisimulation but not under bisimulation, which then means that it cannot be expressed in the basic language but in the language extended with an intersection modality for modalities of basic individual actions. The results are listed in Table 3.3.

Fact 3.30 *On* ABC *models "C can ensure that in the next state p is true." is not invariant under bisimulations.*

Proof. As a counter example consider the models depicted in Figure 3.3, in which the dashed line represents the relation $Z = \{(w_0, v_0), (w_1, v_1)\} \cup (\{w_2, w_3\} \times \{v_2, v_3\})$.

We claim that Z is a bisimulation. For $(w_1, v_1), (w_2, v_2), (w_2, v_3), (w_3, v_2)$ and

	Invariance	**Formula**	**Upper bound for** MC, SAT
LTS	Bisimulation	$\langle C\rangle p$	P, PSPACE
ABC	$\cap$-Bisimulation	$\bigvee_{\vec{a_j}\in\vec{C}}[\bigcap \vec{a_j}]p$	P, PSPACE
PBC	Bisimulation	$\langle\varnothing\rangle[C]\mathbf{X}p$	P, PSPACE
NCL	Bisimulation	$\langle\varnothing\rangle[C]\mathbf{X}p$	P, NEXPTIME

Table 3.3: "C can ensure that in the next state p is true."

(w_3, v_3) this is easy to see. Thus, it remains to show that for (w_0, v_0), the back and forth conditions are satisfied for each relation in our similarity type. To increase readability, we give the witnesses in a table, on the left the transitions in $\mathcal{M}_1$ and on the right the corresponding transitions in $\mathcal{M}_2$.

Forth for $\xrightarrow{1,a_1}$	Zig	Zag
	(w_0, w_1)	(v_0, v_1)
	(w_0, w_3)	(v_0, v_2)
Forth for $\xrightarrow{1,b_1}$	Zig	Zag
	(w_0, w_2)	(v_0, v_3)
Forth for $\xrightarrow{2,a_2}$	Zig	Zag
	(w_0, w_1)	(v_0, v_1)
	(w_0, w_2)	(v_0, v_2)
Forth for $\xrightarrow{2,b_2}$	Zig	Zag
	(w_0, w_3)	(v_0, v_3)

The witnesses for the back condition can also be read off the preceding table, by switching Zigs and Zags. Thus, Z is a bisimulation. However, we have that in $\mathcal{M}_1, w_0$ the coalition $\{1, 2\}$ can force p by agent 1 doing a_1 and agent 2 doing a_2, whereas in $\mathcal{M}_2, v_0$ the coalition cannot achieve that p because agent 1 and 2 together cannot make sure that the system moves into state v_1. ■

Now, let us move to basic concepts involving the other primitive in our models: preferences. The simplest preference notion we consider here is that of an agent finding some state at least as good in which some proposition p is true. Since in all our models preferences are represented in the same way and the preference fragments of the different languages we consider are the same, we get the same results for this notion on all three classes of models.

Therefore, an agent finding a state at least as good where p is true can thus be expressed on all models using the corresponding basic modal language

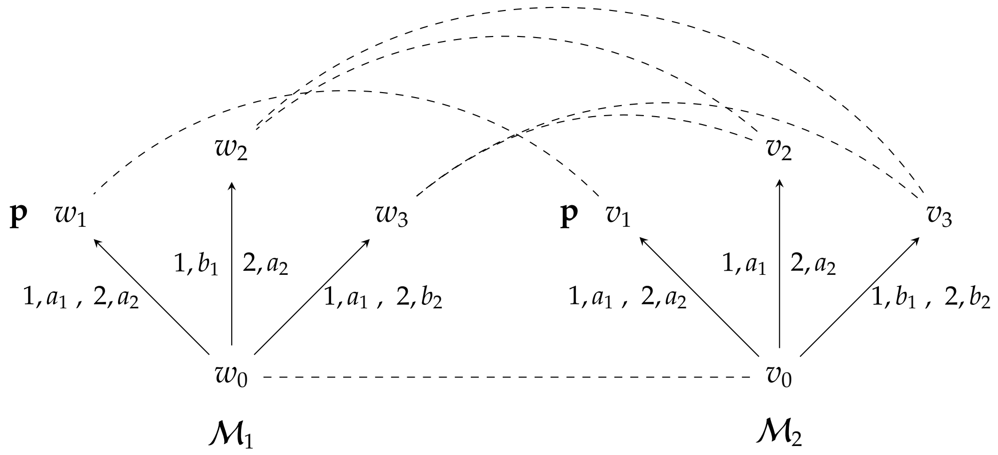


Figure 3.3: Coalitional power is not invariant under bisimulation on ABC. Note that preference relations and the reflexive loops for all actions at w_1, w_2, w_3, v_1, v_2 and v_3 are omitted.

	Invariance	**Formula**	**Upper bound for** MC, SAT
LTS, ABC, PBC	Bisimulation	$\langle \leq_j \rangle p$	P, PSPACE

Table 3.4: "j finds a state at least as good where p is true."

with MC in P and SAT in PSPACE (see Table 3.4). Let us now move on and look at more interesting properties combining coalitional power and preferences. For a similar study of notions involving preferences, we refer the reader to Dégremont and Kurzen (2009a) and to Chapter 7 of Dégremont (2010).

Coalition C **can make agent** j **happy.** A very basic combination of coalitional power and individual preference – which we were also already concerned with in the previous chapter – is the ability of a coalition to guarantee that the next state will be one that is at least as good for some agent. Our invariance results show that this property turns out to be easiest to express on LTS. Here, it is invariant under ∩-bisimulation, as the ability of a group to make the system move into a state at least as good for some agent just says that there is some state that is accessible both by the coalition relation and by the preference relation of the agent.

For ABC and PBC on the other hand, the property turns out to be more complicated. For ABC, a coalition being able to force the system into a state at least as good for some agent means that there are actions for all members of the coalition such that the set of states that is accessible by the intersection of the relations for the individual actions is a subset of the set of states accessible by the preference relation for the agent. On PBC, the property means that in the current state, the coalition can choose a cell within the $\sim_\emptyset$-cell of the current system such that no matter what the other agent chooses, the F_X image of the resulting state will be at least as good for the agent as the current state. This also involves reasoning about the states accessible by one relation being a subset of the states accessible by another relation. Both for ABC and PBC, the property is not invariant under any natural kind of bisimulation. For ABC, this is shown in Fact 3.31, and for PBC the idea is very similar. Nevertheless, the property of a coalition being able to force the system into a state at least as good for some agent is indeed invariant under taking generated submodels.

Fact 3.31 *On* ABC *models, "C can move the system into a state which is as least as good for j as the current state" is not invariant under* $\cap$*-bisimulation.*

Proof. Consider the countable infinite models $\mathcal{M}_1$ and $\mathcal{M}_2$, both with one agent (1) and one action (a). The accessibility relation for the action is defined as depicted in Figure 3.4, with the preference relation for the agent running vertically: In $\mathcal{M}_1$, 1 finds w_i at least as good as w_j iff $i \geq j$, and analogously in $\mathcal{M}_2$ 1 finds v_i at least as good as v_j iff $i \geq j$. Now, in the state w_0 of $\mathcal{M}_1$, by performing action a, agent 1 can force the system into a state as least as good for her, namely into w_1. In $\mathcal{M}_2$, on the other hand, in v_0 agent 1 cannot force the system into a state at least as good for her, as for any action she can perform (which is only a), there is the possibility that the system moves into v_{-1}, which is strictly worse for her than v_0. ■

Both for ABC and PBC, showing that the property is invariant under taking generated submodels is straightforward as the property only involves reasoning about states being accessible from the current state by different relations. All these states will still be accessible from the current state in the generated submodel, and moreover no states will be accessible that were not already accessible in the original model.

Fact 3.32 *On* ABC *and* PBC *the property of a coalition having the power to force that the next state is at least as good as the current one for some agent is invariant under generated submodels.* ◄

Table 3.5 summarizes the results for the ability to make an individual happy. These results nicely illustrate that the choice of models has a great impact on how difficult it is to express certain concepts. Saying that a coalition can ensure that an agent will be at least as happy as before can be done quite easily (by only adding intersection to the basic language) in coalition-labeled transition systems, while it seems to require undecidable logics on the other two classes.

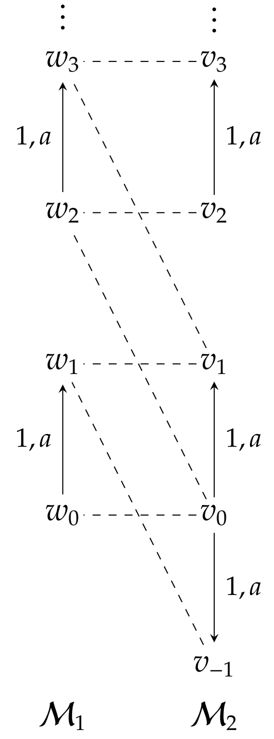


Figure 3.4: Coalitional ability to force the system into a state at least as good for some agent is not invariant under $\cap$-bisimulation. The preference relation $\leq_1$ in both models runs from bottom to top.

Next, we investigate *stability* notions. To be precise, we consider two versions of Nash-stability.

Nash-stability

Nash-stability says that no single agent has the power to make the system move into a state that is *strictly* better for him. In Nash-stable states, individuals thus do not have an incentive to act in order to change the current state.

On LTS, a state being Nash-stable means that for every agent all the states accessible by the relation for the coalition consisting of that agent have to be at least as bad as the current one for that agent; i.e., for all of them it holds that the current state is at least as good.

On ABC, Nash-stability means that no agent can choose an action that is guaranteed to lead to a strictly better state for that agent. Thus, for every action of every agent, there has to be at least one accessible state by that relation which is at least as bad as the current state for that agent.

For PBC models, a Nash-stable state has the following property. For any

	Invariance	Formula	MC, SAT
LTS	∩-Bisimulation	$\langle C\cap \leq_j\rangle\top$	P,PSPACE
ABC	GSM	$\bigvee_{\vec{a_j}\in\vec{C}}(\downarrow x.[\bigcap \vec{a_j}](\downarrow y.@_x\langle\leq_j\rangle y))$	PSPACE, Π^0_1
PBC	GSM	$\downarrow x.\langle\varnothing\rangle[C]\mathbf{X}\downarrow y.@_x\langle\leq_j\rangle y$	PSPACE, Π^0_1

Table 3.5: "*C* can move the system into a state at least as good for *j*."

	Invariance	Formula	MC, SAT
LTS	GSM	$\bigwedge_{j\in\mathtt{N}}\downarrow x.[j]\langle\leq_j\rangle x$	PSPACE, EXPTIME
ABC	GSM	$\bigwedge_{j\in\mathtt{N}}\bigwedge_{a_j\in A_j}\downarrow x.\langle a_j\rangle\langle\leq\rangle x$	PSPACE, EXPTIME
PBC	GSM	$\bigwedge_{j\in\mathtt{N}}\downarrow x.[\varnothing]\langle\{j\}\rangle\mathbf{X}\langle\leq\rangle x$	PSPACE, EXPTIME

Table 3.6: "The current state is Nash-stable."

cell that a singleton coalition (a coalition consisting of one agent only) chooses, there is a possible next state (a F_{X}-successor) that is not strictly better for that agent than the current state.

On all these models, Nash-stability is invariant under taking generated submodels. We now give the proof for ABC.

Fact 3.33 *On* ABC *models, Nash-stability is invariant under generated submodels.*

Proof. We show that if a state is not Nash-stable, then so is its image in a generated submodel (and conversely). Assume that $\mathcal{M}, w$ is not Nash-stable and without loss of generality assume that agent i is the witness. Then there exists some action a_i for i, which by definition has at least one successor, such that for every $v \in a_i[w]$, $w \leq_i v$ and $v \nleq_i w$. By definition of a generated submodel, w will have the same a_i-successors and the preferences relations between w and all the $v \in a_i[w]$ will be the same, making w not Nash-stable either. The other direction is similar. ■

The proofs for the other two classes of models are similar. The key idea is once again that the property is only about states accessible from the current state.

On all three classes of models, Nash-stability can be expressed in a modal logic with the combined complexity of MC in PSPACE and with SAT in EXPTIME. Our results for Nash-stability are summarized in Table 3.6.

Strong Nash-stability

Strong Nash-stability says that no single agent has the power to make the system move into a state that is *at least as good* for him. Thus, if a state is strongly Nash-stable then it is also Nash-stable.

	Invariance	**Formula**	MC, SAT
LTS	∩-bisimulation	$\bigwedge_{j\in\mathtt{N}}[\{i\}\cap \leq_j]\bot$	P, PSPACE
ABC	GSM	$\neg \bigvee_{j\in\mathtt{N}} \bigvee_{a_j\in A_j} \downarrow x.[a_j]\langle\leq^{-1}\rangle x$	PSPACE, Π^0_1
PBC	GSM	$\neg\bigvee_{j\in\mathtt{N}} \downarrow x.\langle\emptyset\rangle[\{j\}]\mathbf{X}\langle\leq^{-1}\rangle x$	PSPACE, Π^0_1

Table 3.7: "The current state is *strongly* Nash-stable."

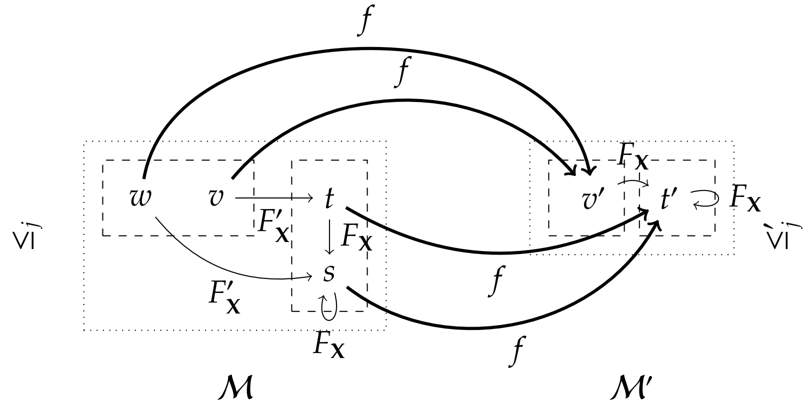
On LTS, strong Nash-stability is invariant under ∩-bisimulation. This is easy to see as strong Nash-stability on LTS just says that for every agent it holds that there is no state that is accessible by the intersection of the preference relation and the ability relation of some agent. On ABC and PBC, strong Nash-stability is invariant under taking generated submodels, but not under bounded morphisms (see e.g. Blackburn et al. (2001)for a definition) or any natural bisimulations. Comparing this to the results for Nash-stability, we can see that on LTS strong Nash-stability is easier to express than Nash-stability while on ABC and PBC we get opposite results.

Fact 3.34 *On* NCL *models, strong Nash-stability is not invariant under bounded morphisms, and thus it is neither invariant under bounded morphisms on* PBC.

Proof. Take two single-agent models with $\mathtt{N} = \{j\}$: $\mathcal{M} = \langle W, \mathtt{N}, (\sim_C)_{C\subseteq\mathtt{N}}, F_\mathsf{X}, (\leq_j)_{j\in\mathtt{N}}, \mathsf{V}\rangle$, with $W = \{w, v, s, t\}$. and $\mathcal{M}' = \langle W', \mathtt{N}, (\sim')_{C\subseteq\mathtt{N}}, F_\mathsf{X}', (\leq'_j)_{j\in\mathtt{N}}, \mathsf{V}'\rangle$ with $W' = \{v', t'\}$. Let $F_\mathsf{X}(s) = F_\mathsf{X}(w) = s$ and $F_\mathsf{X}(t) = F_\mathsf{X}(v) = t$. Let $\sim_{\{j\}}$ be the smallest equivalence containing $(w, v), (t, s)$ and let the TPO $\leq_j$ be defined such that $w \leq_j v \leq_j t \leq_j w \not\leq_j s$. Let the valuation functions V and V' be such that every propositional letter is false everywhere. Let $F'_\mathsf{X}(t') = F'_\mathsf{X}(v') = t'$, $\sim'_{\{j\}} = \{(t', t'), (v', v')\}$ and $\leq'_i = (W' \times W')$. Consider the function $f : W \to W'$ with $f(w) = f(v) = v'$ and $f(s) = f(t) = t'$.

The following figure illustrates the models and the function between their domains. The preference relation runs from bottom to top. The dashed boxed represent the cells of the equivalence relations for $\{j\}$, and the dotted boxes those of the empty coalition.

The forth condition is trivial for $\leq_j$ because $\leq'_i = W' \times W'$. For $\sim_{\{j\}}$ the forth condition is also easily checked. For F_X it holds because $f(w) = f(v) = v'$ and $f(s) = f(t) = t'$ and that we have $F'_\mathsf{X}(t') = F'_\mathsf{X}(v') = t'$.



For the back condition, $\sim'_j$ and F'_{X} are easy. For $\leq'_j$, note that for $f(s) \leq'_j v'$, we have the witnesses w and v as $s \leq_j w, s \leq_j v$ and $f(w) = f(v) = v'$.

Now it should be clear that in $\mathcal{M}$, at w and v agent j cannot exclude that the next will be a state that she finds strictly worse, while in v' in W' she can (trivially) guarantee that the next state will be at least as good. ∎

Table 3.7 gives our results for strong Nash-stability. At this point we have to mention that the notions of Nash-stability used in `LTS` and `ABC`/`PBC`/`NCL` models are strongly related but mathematically slightly different. Note also that our logical definition of Nash-stability with respect to `ABC` models crucially draws on the fact that the preference relation is a total preorder.

Let us summarize the main results concerning the expressive power and complexity. First of all, we have seen that on `LTS` and `ABC` expressing the intersection of two relations is often needed for expressing interesting concepts. We have seen that for `LTS` strong Nash-stability is easier to express than weak Nash-stability. Both for `ABC` and `PBC`, it is exactly the other way around.

Lower bounds

In the end, we would like to be able to say that if on some class of models one wants to be able to reason about some property, then – under the assumption that one chooses a normal modal logic with some natural property – this logic (MC and SAT) will have at least a certain complexity. It would be very useful for developers of modal logical frameworks for multi-agent systems to have some guidelines that give both upper and lower bounds that can be expected for the complexity of a modal logic system that can reason about certain interesting concepts on a given class of models.

Such an upper bound then tells the system designer that it is possible to build a logic that can reason about the concepts under consideration in such a way that the logic has at most a certain complexity. A lower bound on the other hand would say what will be the lowest possible complexity that we can expect from a normal modal logic that can reason about certain concepts. In other words, such a lower bound can be interpreted as follows: Being able to express the concept under consideration forces that the logic is of a certain complexity (or higher).

In our analysis in this chapter, we have only given upper bounds on the complexity of normal modal logics being able to express certain concepts that are interesting for reasoning about the cooperative ability of groups.

Our invariance results indicate that our definability results are tight to some extent. Indeed, they show that within a large family of known extended modal languages with a natural model-theoretical characterization we could not improve on them. Upper bounds on the complexity are thus accurate to some extent. However, it is of course always possible to design ad hoc logics to express exactly the notion of interest. This leads us to the question of whether we can find a strategy to identify tight lower bounds. We would have to show that for *every* logic interpreted on some class of structures, if the logic can express a particular property, then its satisfiability (model-checking) problem has at least a certain complexity. A first idea could be to use results from the computational social choice literature to obtain lower bounds on the *data* complexity of model-checking a logic that can express some notion. In general, the difficulty is that the results from this literature often take the number of resources (and/or number of agents) as primitives, while the data complexity of a modal logic is usually taken relatively to the size of the model whose domain is in general exponentially bigger than the number of resources. Recall Example 3.3, in which the size of the LTS constructed from a resource allocation setting is exponential in the number of resources.

3.4 Conclusions and Further Questions

We will now summarize the main results of this chapter and then give conclusions and further questions.

3.4.1 Summary

In this chapter, we made a step towards a unified perspective on modal logics for cooperation, focusing on the complexity and expressive power required to reason about interesting concepts from social choice and game theory in different logics. We have seen that action- and power-based models, together with coalition-labeled transition systems, constitute three natural families of cooperation logics with different primitives.

We now summarize the results we obtained for each class of systems.

Coalition-labeled transition systems

The results for coalition-labeled transition systems can be found in Table 3.8. We can see that only Nash-stability cannot be expressed in the basic language extended with an intersection modality. The fact that coalitional power is simplified a lot in this class of models pays off in terms of computational complexity of the logics we determined for reasoning about various concepts. As opposed to both action- and power-based coalitional models, on LTS all the properties we considered could be expressed in modal logics that have a complexity of at most EXPTIME (for SAT).

Property	expressible in	Upper bounds MC, SAT
Simple coalitional power	$\mathcal{L}_{\text{LTS}}$	P, PSPACE
Simple preference	$\mathcal{L}_{\text{LTS}}$	P, PSPACE
C can make j happy	$\mathcal{L}_{\text{LTS}}(\cap)$	P, PSPACE
Nash-stability	$\mathcal{H}_{\mathcal{L}_{\text{LTS}}}(\downarrow)$	PSPACE, EXPTIME
strong Nash-stability	$\mathcal{L}_{\text{LTS}}(\cap)$	P, PSPACE

Table 3.8: Results for LTS.

Action-based coalitional models

The results for action-based coalitional models can be found in Table 3.9. We notice that already for expressing that a coalition can force that a proposition holds, the basic language is not sufficient as we need to be able to take the intersection of action relations. Expressing the ability of a coalition to make the system move into a state at least as good for some agent does not seem to be possible in a decidable normal modal logic with natural model theoretical properties. The same holds for strong Nash-stability. For regular Nash-stability on the other hand, we identified a hybrid extension of the basic logic that can express this property and whose satisfiability problem is in EXPTIME. The key for getting a decidable hybrid extension here is that we can express Nash-stability in the fragment without the alternation $\Box \downarrow \Box$.

Power-based coalitional models

Table 3.10 summarizes the results for power-based coalitional models. The basic property of a coalition being able to force that a proposition holds can be

Property	**expressible in**	**Upper bounds** MC, SAT
Simple coalitional power	$\mathcal{L}_{\mathsf{ABC}}(\cap)$	P, PSPACE
Simple preference	$\mathcal{L}_{\mathsf{ABC}}$	P, PSPACE
C can make j happy	$\mathcal{H}_{\mathcal{L}_{\mathsf{ABC}}}(@, \downarrow, \cap)$	PSPACE, Π^0_1
Nash-stability	$\mathcal{H}_{\mathcal{L}_{\mathsf{ABC}}}(\downarrow)$	PSPACE, EXPTIME
strong Nash-stability	$\mathcal{H}_{\mathcal{L}_{\mathsf{ABC}}}(\downarrow, {}^{-1})$	PSPACE, Π^0_1

Table 3.9: Results for ABC.

expressed in the basic language using three modalities (one for the equivalence relation of the empty set, one for that of the coalition itself, and one for the outcome function). Again, the fact that for Nash-stability the bounded fragment without alternations $\Box\downarrow\Box$ is sufficient makes this stability notion expressible in a decidable logic. For the ability of a coalition to force the system into a state at least as good for some agent we could only identify an undecidable hybrid extension. The same holds for strong Nash-stability, for which we also need the converse modality.

Property	**expressible in**	**Upper bounds** MC, SAT
Simple coalitional power	$\mathcal{L}_{\mathsf{PBC}}$	P, PSPACE
Simple preference	$\mathcal{L}_{\mathsf{PBC}}$	P, PSPACE
C can make j happy	$\mathcal{H}_{\mathcal{L}_{\mathsf{PBC}}}(@, \downarrow)$	PSPACE, Π^0_1
Nash-stability	$\mathcal{H}_{\mathcal{L}_{\mathsf{PBC}}}(\downarrow)$	PSPACE, EXPTIME
strong Nash-stability	$\mathcal{H}_{\mathcal{L}_{\mathsf{PBC}}}(\downarrow, {}^{-1})$	PSPACE, Π^0_1

Table 3.10: Results for PBC.

3.4.2 Conclusions

We now present the conclusions that we can draw from the above results.

Intersection is crucial for reasoning about cooperation. In general, our invariance results show that interesting notions about coalitional power and preferences are often not invariant under bisimulation. However, in many cases it is only a matter of allowing the underlying logics to reason about the intersection of two relations. From this, we can draw the conclusion that being able to express intersection is crucial when reasoning about cooperation of agents in normal modal logics.

On simple coalition-labeled transition systems, intersection is needed as soon as we want to reason about coalitions having the ability to make an agent happy (by forcing the system into a state the agent prefers). On action-based coalitional models, the intersection of action relations is already needed for expressing even the simplest notion involving coalitional power: the ability to make some proposition true. The good news is that we can add an intersection modality to the basic language without (significantly) increasing the complexity of the logic. In general, the observation that the intersection of agents' ability-modalities is needed for many interesting properties of coalitional ability is not surprising: talking e.g. about a particular state being in a certain relation for more than one agent seems to be at the heart of reasoning about coalitions. The same holds for the intersection of ability modalities and preferences: the combination of ability and preference plays a central role as soon as we want to reason about the ability to achieve an improvement of the situation for an agent.

Choice of primitive has a great impact on whether strong or weak stability notions are easier to express. One of our main findings is that the choice of primitives of the modal frameworks for cooperation has a great impact on whether strong or weak stability notions are easier to express.

- In action- and power-based models, weak stability notions require less expressive power and complexity than strong stability notions.
- In coalition-labeled transition systems, the situation is just the opposite.

This has to do with whether coalitional power to achieve an improvement for an agent can be expressed in a simple way such that the intersection of relations is sufficient to express the stability notion or whether we need to express something like a subset relation.

Let us come back to our first research question.

> **Research Question 1** *What formal frameworks are best suited for reasoning about which concepts involved in interaction?*
>
> - *What should be the primitive notions a formal approach should be based on?*

With our above results, we can thus draw the following conclusions for answering the question.

1. For a framework that makes explicit how coalitional power arises from that of individuals which have preferences, being able to express the intersection of relations is crucial.

2. Making coalitional power explicit in terms of the power of subcoalitions or in terms of actions in general leads to a higher complexity than choosing a very general approach, abstracting away from the intrinsic structure of coalitional power.

3. In the design of modal logic frameworks for reasoning about game theoretical concepts, special attention needs to be paid to whether the system should be able to reason about weak or strong stability notions.

Remarks

It is important to note that in our definability results we made use of very big conjunctions and disjunctions. When taking conjunctions/disjunctions over all coalitions, they will be exponentially related to the number of agents. The consequences we draw about the upper bounds on the complexity of satisfiability or of combined complexity of model checking is thus to be balanced by the fact that we generally use very big conjunctions or disjunctions that might well be exponential if we take the number of agents as a parameter for the complexity results.

The fact that both for action- and power-based models for relatively basic notions such as the ability of a group to make an agent happy we identified very expressive logics which are undecidable could be understood as bad news for reasoning about cooperation using modal logic. However, we note that the models we considered are very general and for modeling specific scenarios at hand it will often be possible to take restricted classes of models for which less expressive power is needed.

Our invariance results indicate that our definability results are tight as far as we are concerned with expressibility in modal logics with natural model-theoretic characterization.

3.4.3 Further Questions

The work in this chapter gives rise to some interesting questions for further research.

- What are natural extensions of our analysis?

  The stability notions that we considered in this work express that agents do not have an incentive to change the current state within one step. In order to express more sophisticated stability notions for interactive systems fixed point logics such as the modal μ-calculus are needed. A central open problem arising from our work is to extend action-based models to reason about transitive closure in order to simulate more powerful logics such as ATL*.

- How can our complexity results be interpreted in a way that takes into account how succinctly the different properties can be expressed in the different logical frameworks?

  Here, a detailed analysis using methods from parametrized complexity could lead to more insights, showing which are the exact parameters that contribute to the complexity.

- How can tight lower bounds be obtained for the complexity of logics that can express some concept?

  As we have discussed on page 84, we could focus purely on the algorithmic tasks of checking certain properties on the different semantic structures. Precisely determining this complexity will then give us lower bounds on the data complexity of model checking these properties on the different classes of models.

3.4.4 Concluding Part I

The work in this part has clarified the impact of certain design choices for the complexity of modal logics for cooperation. We have seen how the choice of primitives influences the complexity required for expressing certain concepts inspired by game theory. While we could pinpoint specific sources of the complexity, we also note that the kind of complexity analysis that we were concerned with in Part I is very general as we considered satisfiability and model checking problems. These problems are concerned with any formula of a given language. Hence, they tell us something about the complexity of deciding whether any property expressible in some language holds in some/any model. However, it might just be the case that the properties that are relevant for reasoning about interaction all lie at one end of the complexity spectrum. Thus, the complexity results given so far can rather be seen as describing the complexity of *theories of interaction* than describing the complexity of interaction itself.

In order to focus more on the complexity of interaction itself, Part II will analyze the computational complexity of very interaction-specific decision problems, such as

- deciding whether a player has a winning strategy (Chapter 4),
- deciding whether the information states of two agents are in a certain relation (Chapter 5),

- deciding what are the legal moves in a game (Chapter 6).

While we will still use modal logic frameworks in Chapter 4 and 5, our complexity analysis will be about the complexity of decision problems specifically about the interaction of individual agents.

Part II

Complexity in the Interaction between Diverse Agents

Chapter 4

Variations on Sabotage – Obstruction and Cooperation

While in the two previous chapters, our complexity analysis of interactive processes has focused on various forms of strategic ability in very general models of interaction, we will now move on to investigating more concrete interactive processes. We shift the perspective from general frameworks for group ability to concrete settings of interaction between diverse individual agents.

Instead of analyzing the computational complexity of logical theories for social action, we now investigate the complexity of deciding if in a particular interactive setting some agent has the ability to achieve success.

For the choice of interactive setting to be investigated, let us come back to our original motivation to develop formal theories that lead to a deeper understanding of modern interaction. Analyzing communication networks as given in Example 1.4, an important question is how robust such a system is against flaws in the network: do messages still reach the intended recipient when connections can break down?

In a sense, a strategy for success can here be seen as a strategy to traverse the network in a way that even if connections break down the destination can be reached.

The breakdowns of the connections can be seen as the work of an adversary agent, and the resulting interaction can be seen as a two-player game between one player traversing the network and the other one cutting edges of the network. This can be modeled in the framework of *Sabotage Games*, which are two-player games played on a graph in which one player travels through the graph while the other one cuts connections.

Sabotage Games have received attention in modal logic as they have corresponding modal logic extended with dynamic modalities for the removal of edges in the model. Their complexity has been well studied and they can be used to model various situations, including the following.

- Travel through a transportation networks with trains breaking down or flights being canceled (van Benthem 2005)
- Fault-tolerant computation and routing problems in dynamically changing networks (Klein et al. 2010)
- Process of language learning in which a learner moves from one grammar to another by changing his mind in response to observations, and a teacher acts by providing information which makes certain mind changes impossible (Gierasimczuk et al. 2009b; Gierasimczuk 2010)

Especially in the last of the above interpretations, it also seems to be very intuitive to think of the "adversary" player as actually having the objective of guiding the other player to a certain destination. This consideration leads us to new variations of Sabotage Games and thus to our second research question.

> **Research Question 2** *What is the role of cooperation vs. competition in the complexity of interaction?*
>
> - *Does analyzing an interactive situation in general become easier if the participants cooperate?*

In Part I, we have already seen that logical theories of cooperating agents can be rather complex if they are designed in a way that makes explicit the internal structure of cooperation. In this chapter we investigate in how far the complexity of Sabotage Games is sensitive to changes in the objectives of the players.

This chapter contributes to the complexity theoretical study of games on dynamically changing structures (cf. e.g. Rohde (2006)), while paying particular attention to how sensitive complexity results of such games are to variations in players' objectives and to variations in the procedural rules. With respect to logic, our contributions lie in the field of modal logics that have the property that evaluating a formula can change the model (cf. e.g. Gabbay (2008)).

4.1 Sabotage Games

The framework of Sabotage Games has originally been introduced by van Benthem (2005). It can be used to model the dynamics of networks in which transitions can break down. Examples of such systems include transportation networks with flights being canceled or trains breaking down, and communication networks with unstable connections. The underlying idea is that the breakdowns can be seen as the result of the interference of an adversary player who is trying to obstruct the travel through the network.

A Sabotage Game is a two-player game played on a multi-graph, i.e., a graph that can have more than one edge between two vertices. The two players, *Runner* and *Blocker*, move in alternation with Runner being the first to move. Runner's moves consist of making a transition along an edge. Blocker moves by deleting an edge from the graph. The game ends when a player cannot move or Runner has reached one of the designated goal vertices. To be more precise, as Runner is the first to move, it cannot happen that the game ends because Blocker cannot remove any edge, as the game would have already ended when it was Runner's turn and he could not move. Runner wins if he has reached a goal vertex; otherwise Blocker wins.

4.1.1 Representing Sabotage Games

To define the game formally, let us first introduce the structure in which Sabotage Games take place.

Definition 4.1 (Directed multi-graph) A *directed multi-graph* is a tuple (V, E) where V is a finite set of vertices and $\mathsf{E} : V \times V \to \mathbb{N}$ is a function indicating the number of edges between any two vertices. For a given vertex $v \in V$, we let $\mathsf{E}(v)$ denote the number of outgoing edges from v, i.e., $\mathsf{E}(v) := \Sigma_{u \in V} \mathsf{E}(v, u)$.◀

The Sabotage Game is then defined as follows.

Definition 4.2 (Sabotage Game (Rohde 2006)) A *Sabotage Game* $SG = \langle V, \mathsf{E}_0, v_0, \mathbf{F} \rangle$ is given by a directed multi-graph (V, E_0) with $V \neq \emptyset$, a vertex $v_0 \in V$, and $\mathbf{F} \subseteq V, \mathbf{F} \neq \emptyset$. v_0 is Runner's starting point and $\mathbf{F}$ is the set of *goal* vertices. We let $\mathbf{SG}$ denote the class of all Sabotage Games.

A *position* of the game is then given by $\langle \tau, \mathsf{E}, v \rangle$, where $\tau \in \{0, 1\}$ determines whose turn it is (Runner's turn if $\tau = 0$ and Blocker's turn if $\tau = 1$), $\mathsf{E} \subseteq \mathsf{E}_0$ and $v \in V$.

Each match is played as follows. The initial position is given by $\langle 0, \mathsf{E}_0, v_0 \rangle$, and each round consists of two moves, as indicated in Table 4.1. A round from position $\langle 0, \mathsf{E}, v \rangle$ consists of the following moves.

1. First Runner chooses some vertex $v' \in V$ such that $\mathsf{E}(v, v') > 0$.

2. In the resulting position $\langle 1, \mathsf{E}, v' \rangle$, Blocker picks some edge (u, u') such that $\mathsf{E}(u, u') > 0$.

The game continues in position $\langle 0, \mathsf{E}^{-(u,u')}, v' \rangle$ where $\mathsf{E}^{-(u,u')}$ is defined as follows for each $(w, w') \in V \times V$:

$$\mathsf{E}^{-(u,u')}(w, w') := \begin{cases} \mathsf{E}(w, w') - 1 & \text{if } (w, w') = (u, u') \\ \mathsf{E}(w, w') & \text{otherwise.} \end{cases}$$

	Position	Player	Admissible moves
1.	$\langle 0, \mathsf{E}, v\rangle$	Runner	$\left\{\langle 1, \mathsf{E}, v'\rangle \mid \mathsf{E}(v, v') > 0\right\}$
2.	$\langle 1, \mathsf{E}, v'\rangle$	Blocker	$\left\{\langle 0, \mathsf{E}^{-(u,u')}, v'\rangle \mid \mathsf{E}(u, u') > 0\right\}$

Table 4.1: A round in the Sabotage Game

Final Position	Winner
$\langle 0, \mathsf{E}, v\rangle$, with $\mathsf{E}(v) = 0$	Blocker
$\langle 1, \mathsf{E}, v\rangle$, with $v \in \mathbf{F}$	Runner

Table 4.2: Final Positions in the Sabotage Game

The match ends as soon as Runner has reached a vertex in **F** (in which case he wins) or Runner cannot make a move (in which case Blocker wins). The precise conditions are given in Table 4.2.

As in every round one edge is removed, every Sabotage Game $\langle V, \mathsf{E}_0, v_0, \mathbf{F}\rangle$ ends after at most $\sum_{(v,v')\in V\times V} \mathsf{E}_0(v, v')$ rounds. ◄

In the above definition, Blocker's moves of deleting an edge are represented by subtracting 1 from the value of $\mathsf{E}(u, u')$ for the chosen pair of vertices (u, u'). As we will see later, this definition can lead to some technical problems when we want to interpret a modal logic over these structures. Therefore, we will now present an alternative definition of a Sabotage Game, which we subsequently show to be equivalent with respect to the existence of winning strategies.

Definition 4.3 (Directed labeled multi-graph) Let $\Sigma = \{a_1, \ldots a_m\}$ be a finite set of labels. A *directed labeled multi-graph* based on Σ is a tuple $(V, \mathcal{E})$, where V is a finite set of vertices and $\mathcal{E} = (\mathcal{E}^{a_1}, \ldots, \mathcal{E}^{a_m})$ is a sequence of binary relations over V, i.e., $\mathcal{E}^{a_i} \subseteq V \times V$ for each $a_i \in \Sigma$. For each vertex $v \in V$, we define $\mathcal{E}(v) := \bigcup_{a\in\Sigma}\{u \in V \mid (v, u) \in \mathcal{E}^a\}$. Thus, $\mathcal{E}(v)$ denotes the set of all vertices u such that there is an edge from v to u, labeled by a label in Σ. ◄

In this definition, labels from Σ are used to represent multiple edges between two vertices. The sequence $\mathcal{E}$ is simply an ordered collection of binary relations on V with labels from Σ. Accordingly, the modified definition of Sabotage Games is as follows.

Definition 4.4 (Labeled Sabotage Game) A *Labeled Sabotage Game* $\mathcal{SG} = \langle V, \mathcal{E}_0, v_0, \mathbf{F}\rangle$ is given by a directed labeled multi-graph $(V, \mathcal{E}_0)$ (based on $\Sigma = \{a_1, \ldots a_m\}$) with $V \neq \emptyset$, a vertex $v_0 \in V$ and subset of vertices $\mathbf{F} \subseteq V, \mathbf{F} \neq \emptyset$. Vertex v_0 is Runner's starting point and $\mathbf{F}$ is the set of *goal* vertices. We let $\mathfrak{SG}$ denote the class of all Labeled Sabotage Games.

Each match is played as follows. The initial position is given by $\langle 0, \mathcal{E}_0, v_0\rangle$, and each round consists of two moves, as indicated in Table 4.3. In words, a round from position $\langle 0, \mathcal{E}, v\rangle$ with $\mathcal{E} = (\mathcal{E}^{a_1}, \ldots, \mathcal{E}^{a_m})$ consists of the following moves.

1. First, Runner chooses a vertex v' such that $(v, v') \in \mathcal{E}^{a_i}$ for some $a_i \in \Sigma$.

2. Then the game continues in position $\langle 1, \mathcal{E}, v'\rangle$ with Blocker picking some edge (u, u') and a label $a_j \in \Sigma$ such that $(u, u') \in \mathcal{E}^{a_j}$.

The game then continues in position $\langle 0, \mathcal{E}^{-(u,u'),a_j}, v'\rangle$, where $\mathcal{E}^{-(u,u'),a_j}$ is given by

$$\mathcal{E}^{-(u,u'),a_j} = (\mathcal{E}^{a_1}, \ldots, \mathcal{E}^{a_j} \setminus \{(u,u')\}, \ldots, \mathcal{E}^{a_m}).$$

	Position	Player	Admissible moves
1.	$\langle 0, \mathcal{E}, v\rangle$	Runner	$\left\{\langle 1, \mathcal{E}, v'\rangle \mid (v, v') \in \mathcal{E}^{a_i} \text{ for some } a_i \in \Sigma\right\}$
2.	$\langle 1, \mathcal{E}, v'\rangle$	Blocker	$\left\{\langle 0, \mathcal{E}^{-(u,u'),a_j}, v'\rangle \mid (u, u') \in \mathcal{E}^{a_j} \text{ for some } a_j \in \Sigma\right\}$

Table 4.3: A round in the Labeled Sabotage Game

The match ends when Runner has reached a vertex in $\mathbf{F}$ or cannot make a move. In the first case, Runner wins and in the second case Blocker wins. The precise conditions are given in Table 4.4.

Final Position	Winner
$\langle 0, \mathcal{E}, v\rangle$, with $\mathcal{E}(v) = \emptyset$	Blocker
$\langle 1, \mathcal{E}, v\rangle$, with $v \in \mathbf{F}$	Runner

Table 4.4: Final Positions in the Labeled Sabotage Game

As in every round an edge is removed and the set of labels is finite, every Labeled Sabotage Game $\langle V, \mathcal{E}_0, v_0, \mathbf{F}\rangle$ ends after at most $\sum_{a \in \Sigma} |\mathcal{E}_0^a|$ rounds. ◄

Note that due to the order in which the players make their moves, the positions given in Table 4.4 are the only final positions.

The only difference between the two definitions of a Sabotage Game is the way in which the multiple edges are represented: while the standard Sabotage Game uses a function that tells us how many edges are between any given pair of vertices, the *labeled* version uses relations with different labels.

Note that both games have the *history-free determinacy property*:

> if a player has a winning strategy, (s)he has a *positional* winning strategy, i.e., a winning strategy that depends only on the current position of the game, and not on which moves have led to it.

Then, in particular, in a *Labeled* Sabotage Game $\langle V, \mathcal{E}_0, v_0, \mathbf{F}\rangle$, a round from position $\langle 0, \mathcal{E}, v\rangle$ with Runner choosing to move to v' and Blocker choosing to let the game continue in position $\langle 0, \mathcal{E}^{-(u,u'),a_j}, v'\rangle$ can be seen as a transition from the game $\langle V, \mathcal{E}, v, \mathbf{F}\rangle$ to the game $\langle V, \mathcal{E}^{-(u,u'),a_j}, v', \mathbf{F}\rangle$ because all previous moves become irrelevant. We will use this fact throughout this chapter. Also, by edges and vertices of a game $\langle V, \mathcal{E}, v, \mathbf{F}\rangle$, we will mean edges and vertices of its underlying directed (labeled) multi-graph $(V, \mathcal{E})$.

In Labeled Sabotage Games, the labels of the edges are irrelevant for the existence of a winning strategy: Observation 4.5 shows that only the number of edges between each pair of vertices matters.

Observation 4.5 *Let $\mathcal{SG} = \langle V, \mathcal{E}, v_0, \mathbf{F}\rangle$ and $\mathcal{SG}' = \langle V, \mathcal{F}, v_0, \mathbf{F}\rangle$ be two Labeled Sabotage Games, both based on the set of labels Σ, such that the games differ only in the labels of the edges, i.e., given $\mathcal{E} = (\mathcal{E}^{a_1}, \ldots, \mathcal{E}^{a_m})$ and $\mathcal{F} = (\mathcal{F}^{a_1}, \ldots, \mathcal{F}^{a_m})$, we have that for all $(v, v') \in V \times V$ it holds that*

$$\Big|\{a \in \Sigma \mid (v, v') \in \mathcal{E}^a\}\Big| = \Big|\{a \in \Sigma \mid (v, v') \in \mathcal{F}^a\}\Big|.$$

Then Runner has a winning strategy in $\mathcal{SG}$ if and only if he has a wining strategy in $\mathcal{SG}'$.

Proof. Follows from the fact that between any two vertices $v, v' \in V$ there are the same number of edges in $\mathcal{E}$ as there are in $\mathcal{F}$, and thus there is a bijection

$$f_{(v,v')} : \{a \in \Sigma \mid (v, v') \in \mathcal{E}^a\} \to \{a \in \Sigma \mid (v, v') \in \mathcal{F}^a\}.$$

Assume that Runner has a winning strategy in $\mathcal{SG}$. Now, if the strategy tells him that in the first round, he should choose position $\langle 1, \mathcal{E}_0, v'\rangle$, then in the first round of $\mathcal{SG}'$ he can choose $\langle 1, \mathcal{F}_0, v'\rangle$ as this is a legal move. Then, when Blocker replies in $\mathcal{SG}'$ by choosing position $\langle 0, {\mathcal{F}_0}^{-(u,u'),a_i}, v'\rangle$, Runner can now use the winning strategy he has for position $\langle 0, {\mathcal{E}_0}^{-(u,u'),f^{-1}_{(u,u')}(a_i)}, v'\rangle$ in $\mathcal{SG}$. If this strategy tells him to choose position $\langle 1, {\mathcal{E}_0}^{-(u,u'),f^{-1}_{(u,u')}(a_i)}, v''\rangle$ then in $\mathcal{SG}'$ he can choose position $\langle 1, \mathcal{F}_0{}^{-(u,u'),a_i}, v'' \rangle$. It is clear that proceeding this way, Runner will also be able to win $\mathcal{SG}'$.

The other direction is analogous: If in $\mathcal{SG}$ Blocker removes an edge between two vertices u, u' Runner can always respond by doing the move he would have done in $\mathcal{SG}'$ if Blocker had removed an edge between the same two vertices with the corresponding label (now using the function $f_{(u,u')}$). ■

Using Observation 4.5, we can now prove the following theorem, which states that Sabotage Games and Labeled Sabotage Games are equivalent w.r.t. the existence of winning strategies. More precisely, we show the following. First of all, for every Sabotage Game, there is a Labeled Sabotage Game that has the same set of vertices, the same set of goal vertices and also the same number of edges between any two vertices, and moreover the players have the same abilities to win. Second, for every Labeled Sabotage Game, there is also such a corresponding Sabotage game satisfying the same conditions. Thus, we show that the games can be transformed into each other while the winning abilities stay the same and the games are of the same size if we measure the size as the sum of the vertices and the edges between them[1].

Theorem 4.6 *1. We can define a function $f : \mathbf{SG} \to \mathfrak{S}\mathfrak{G}$ such that, for every Sabotage Game $SG = \langle V, \mathsf{E}, v, \mathbf{F} \rangle$ in* **SG**,

(a) $f(SG)$ is based on the set of labels $\Sigma = \{1, \ldots, m\}$, where m is the largest number of edges between any two vertices in SG, that is, $m := \max\{\mathsf{E}(u,u') \mid (u,u') \in (V \times V)\}$;

(b) SG and $f(SG)$ have the same vertices, initial vertex and set of goal vertices;

(c) the number of edges between any two vertices is the same in SG and $f(SG)$;

(d) Runner has a winning strategy in SG iff he has one in $f(SG)$.

*2. We can define a function $g : \mathfrak{S}\mathfrak{G} \to$ **SG** such that, for every Labeled Sabotage Game $\mathcal{SG} = \langle V, \mathcal{E}, v, \mathbf{F} \rangle$ in $\mathfrak{S}\mathfrak{G}$ with Σ its set of labels,*

(a) $\mathcal{SG}$ and $g(\mathcal{SG})$ have the same vertices, initial vertex and set of goal vertices;

(b) the number of edges between any two vertices is the same in $\mathcal{SG}$ and $g(\mathcal{SG})$;

(c) Runner has a winning strategy in $\mathcal{SG}$ iff he has one in $g(\mathcal{SG})$.

Proof.

[1] At this point, it is possible to argue that encoding the multiplicity of the edges as (binary) numbers is more succinct than using different relations, which results in a unary coding of the multiplicity of the edges.

1. For any Sabotage Game $SG = \langle V, \mathsf{E}, v, \mathbf{F}\rangle$ in **SG**, define $\Sigma := \{1, \ldots, m\}$ with m as in the theorem statement. Then, define the Σ-based Labeled Sabotage Game $f(\langle V, \mathsf{E}, v, \mathbf{F}\rangle)$ as

$$f(\langle V, \mathsf{E}, v, \mathbf{F}\rangle) := \langle V, \mathcal{E}, v, \mathbf{F}\rangle$$

where $\mathcal{E} := (\mathcal{E}^1, \ldots, \mathcal{E}^m)$ and every $\mathcal{E}^i$ is given by

$$\mathcal{E}^i := \{(u, u') \in V \times V \mid \mathsf{E}(u, u') \geq i\}$$

Points (a) and (b) follow immediately from f's definition. For (c), the number of edges in SG between any two vertices u, u' is given by $\mathsf{E}(u, u')$, but then the pair (u, u') will be in every $\mathcal{E}^i$ such that $i \leq \mathsf{E}(u, u')$, i.e., in exactly $\mathsf{E}(u, u')$ of the relations in $\mathcal{E}$. It is just left to show that Runner has a winning strategy in SG iff he has one in $f(SG)$.

The proof is by induction on $n := \sum_{(v,v') \in V \times V} \mathsf{E}(v, v')$, the total number of edges of SG which, by point (c), is also the total number of edges in $f(SG)$. Moreover, since f only changes the way the vertices are represented, we will be more precise and denote $f(\langle V, \mathsf{E}, v, \mathbf{F}\rangle)$ as $\langle V, f(\mathsf{E}), v, \mathbf{F}\rangle$.

The base case. Straightforward, since when there are no edges, in both games Runner has a winning strategy iff $v \in \mathbf{F}$.

The inductive step. From left to right, suppose that Runner has a winning strategy in the game $\langle V, \mathsf{E}, v, \mathbf{F}\rangle$ with $n + 1$ edges. If Runner wins immediately, $v \in \mathbf{F}$ and he also wins $f(\langle V, \mathsf{E}, v, \mathbf{F}\rangle)$ immediately. Otherwise, there is some $v' \in V$ such that $\mathsf{E}(v, v') > 0$ and Runner has a winning strategy in all games $\langle V, \mathsf{E}^{-(u,u')}, v', \mathbf{F}\rangle$ that result from Blocker choosing to remove an edge between u and u' with $\mathsf{E}(u, u') > 0$. Since all games $\langle V, \mathsf{E}^{-(u,u')}, v', \mathbf{F}\rangle$ have n edges, by inductive hypothesis Runner has a winning strategy in $\langle V, f(\mathsf{E}^{-(u,u')}), v', \mathbf{F}\rangle$.

But then, by Observation 4.5, Runner has also a winning strategy in all games $\langle V, f(\mathsf{E})^{-(u,u'),i}, v', \mathbf{F}\rangle$ that result from removing an edge from u to u' with label i (with $1 \leq i \leq m$ and $(u, u') \in \mathcal{E}^i$) because the only possible difference between $\langle V, f(\mathsf{E}^{-(u,u')}), v', \mathbf{F}\rangle$ and $\langle V, f(\mathsf{E})^{-(u,u'),i}, v', \mathbf{F}\rangle$ is in the labels of the edges between u and u': in the former, the removed label is the largest; in the latter, the removed label is any. Since moving to v' is also an admissible move in $\langle V, f(\mathsf{E}), v, \mathbf{F}\rangle$, Runner also has a winning strategy in this latter game.

From right to left. If Runner has a winning strategy in $\langle V, f(\mathsf{E}), v, \mathbf{F}\rangle$, he can choose some v' with $(v, v') \in \mathcal{E}^i$ for some i with $1 \leq i \leq m$ such that he has a winning strategy in all games $\langle V, f(\mathsf{E})^{-(u,u'),i}, v', \mathbf{F}\rangle$ that result from Blocker removing an edge with label i between a pair of vertices (u, u').

Now, by Observation 4.5, if Runner has a winning strategy in every such $\langle V, f(\mathsf{E})^{-(u,u'),i}, v', \mathbf{F}\rangle$ then he has a winning strategy in every $\langle V, f(\mathsf{E}^{-(u,u')}), v', \mathbf{F}\rangle$ because, again, the only difference between the two games is in the labels of the edges. But then, by the inductive hypothesis, he also has a winning strategy in every $\langle V, \mathsf{E}^{-(u,u')}, v', \mathbf{F}\rangle$. Hence, since v' is also an admissible move in $\langle V, \mathsf{E}, v, \mathbf{F}\rangle$, Runner has a winning strategy in this latter game.

2. For any Labeled Sabotage Game $\mathcal{SG} = \langle V, \mathcal{E}, v, \mathbf{F}\rangle \in \mathfrak{SG}$ with $\Sigma := \{a_1, \ldots, a_m\}$, define

$$g(\langle V, \mathcal{E}, v, \mathbf{F}\rangle) := \langle V, \mathsf{E}, v, \mathbf{F}\rangle$$

where E is defined as follows, for every $(u, u') \in V \times V$,

$$\mathsf{E}(u, u') := \Big|\{a \in \Sigma \mid (u, u') \in \mathcal{E}^a\}\Big|.$$

Point (a) is given by g's definition. For (b), note that for every pair of vertices (u, u'), $\mathsf{E}(u, u')$ is given by the number of relations $\mathcal{E}^a$ that contain (u, u'). For (c), showing that Runner has a winning strategy in $\mathcal{SG}$ iff he has one in $g(\mathcal{SG})$ is straightforward and can be done by induction on the number of edges of $\mathcal{SG}$, given by $\sum_{a \in \Sigma} |\mathcal{E}^a|$. In the inductive step, for the left-to-right direction, the idea is that Runner can respond to Blocker removing an edge in the same way as he would have in $\mathcal{SG}$ if Blocker had removed the edge with the highest label. The other direction uses Observation 4.5.

This concludes the proof. ■

We have thus shown that Labeled Sabotage Games and Sabotage Games as defined originally are very similar. For each game in one of the classes, there is a corresponding one with the same vertices, same number of edges and same winning abilities in the other class. In the remainder of this chapter, we will work with Labeled Sabotage Games. We now continue with Sabotage Modal Logic, a framework in which we can reason about the abilities of the players in Sabotage Games.

4.1.2 Sabotage Modal Logic

Sabotage Modal Logic (SML) (van Benthem 2005) has been introduced to reason about reachability-type problems in dynamic structures, such as the graph of our Sabotage Games. Besides the standard modalities, its language contains "transition-deleting" modalities for reasoning about model change that occurs when an edge is removed. More precisely, we have formulas of the form $\diamondminus\varphi$, expressing that it is possible to delete a pair of states from the accessibility relation such that φ holds in the resulting model at the current state.

Definition 4.7 (Sabotage Modal Language (van Benthem 2005)) Let PROP be a countable set of propositional letters and let Σ be a finite set. Formulas φ of the language of Sabotage Modal Logic are given by

$$\varphi ::= p \mid \neg\varphi \mid \varphi \vee \varphi \mid \Diamond_a\varphi \mid \blacklozenge_a\varphi$$

with $p \in \texttt{PROP}$ and $a \in \Sigma$. The formula $\boxminus_a\varphi$ is defined as $\neg\blacklozenge_a\neg\varphi$, and we also define the following abbreviations:

$$\Diamond\varphi := \bigvee_{a\in\Sigma} \Diamond_a\varphi \qquad\qquad \blacklozenge\varphi := \bigvee_{a\in\Sigma} \blacklozenge_a\varphi$$ ◀

The language of Sabotage Modal Logic is interpreted over Kripke models that here will be called *Sabotage Models*.

Definition 4.8 (Sabotage Model (Löding and Rohde 2003b)) Given a countable set of propositional letters PROP and a finite set $\Sigma = \{a_1, \ldots, a_m\}$, a Sabotage Model is a tuple $M = \langle W, (R_{a_i})_{a_i\in\Sigma}, \mathsf{V}\rangle$ where W is a non-empty set of worlds, each $R_{a_i} \subseteq W \times W$ is an accessibility relation and $\mathsf{V} : \texttt{PROP} \to \wp(W)$ is a propositional valuation function. The pair (M, w) with $w \in W$ is called *Pointed Sabotage Model*. ◀

For the semantics, we define the model resulting from removing an edge.

Definition 4.9 Let $M = \langle W, R_{a_1}, \ldots R_{a_m}, \mathsf{V}\rangle$ be a Sabotage Model. The model $M^{-(u,u'),a_i}$ that results from removing the edge $(u, u') \in R_{a_i}$ is defined as

$$M^{-(u,u'),a_i} := \langle W, R_{a_1}, \ldots, R_{a_i} \setminus \{(u, u')\}, \ldots R_{a_m}, \mathsf{V}\rangle.$$ ◀

Definition 4.10 (Sabotage Modal Logic: Semantics (van Benthem 2005)) Given a Sabotage Model $M = \langle W, (R_a)_{a\in\Sigma}, \mathsf{V}\rangle$ and a world $w \in W$, atomic propositions, negations, disjunctions and standard modal formulas are interpreted as usual. For "transition-deleting" formulas, we have

$$(M, w) \models \blacklozenge_a\varphi \quad \text{iff} \quad \exists\, u, u' \in W \ :\ (u, u') \in R_a \ \text{ and } \ (M^{-(u,u'),a}, w) \models \varphi.$$ ◀

Complexity of SML. The computational complexity of SML has been analyzed by Löding and Rohde (2003a). The satisfiability problem of SML is undecidable; the proof is by reduction from Post's Correspondence Problem. Model checking is PSPACE-complete.

We will see in Section 4.2.2 how the logic can be used to express the existence of winning conditions in Sabotage Games and some variations, which we will now define and analyze.

4.2 From Obstruction to Cooperation

In this section, we look at variations of the standard Sabotage Game that differ with respect to the attitude of the players. In the standard version, Runner wins if and only if he reaches a goal state, and Blocker wins if and only if she can prevent this.

However, in many game like interactions on graphs it also makes sense to think of one player trying to *avoid* reaching certain *bad* states, and another player trying to force him to reach such a state. This motivation is related to the situations modeled in *Cops and Robbers* games (cf. e.g. Fomin et al. (2008); Kreutzer (2011)) when we look at it from the perspective of the robbers, or in a variation of the recreational board game *Scotland Yard* (Sevenster 2006), in which we can think of the cops trying to block escape routes of the fugitive and this way forcing him to move to the prison. We will consider a variation on Sabotage Games with a *Safety* winning condition. The moves of the players stay the same, as do the final positions of the game, but the winning conditions change: Runner looses as soon as he reaches a vertex in **F**. He wins otherwise (i.e., when the game stops in a position in which he is not at a goal state, which is the case when he reaches a dead-end which is not in **F**.

Learning theoretical interpretation of Sabotage Games. In previous work (Gierasimczuk et al. 2009b), we have shown that Sabotage Games and their variations can also model processes which at first sight might not be so closely related to strategic game-like interactions, namely the processes of learning and teaching.

Here the idea is that the underlying graph of the Sabotage Game represents the set of hypotheses or grammars, and the transitions between them represent the ways how the learner can change his mind when he gets new information. The set of goal states represents the correct grammars. Then in the standard Sabotage Game, Learner (taking the role of Runner) tries to reach such a goal state while the evil Teacher (Blocker) is trying to obstruct him by giving information that makes certain transitions impossible (represented by cutting the edges). Then with the safety winning condition, we can model the situation in which Teacher wants Learner to reach the learning goal, but the learner however is not willing to learn and is trying to stay away from the goal.

Under the interpretation of the learning scenario, also a third kind of Sabotage Games makes sense because it can be seen to represent the ideal learning situation in which both Teacher and Learner have the aim of the learner reaching the goal, and both cooperate in order to make this happen. Of course, with the learning interpretation the question arises as to what is the most natural sabotage framework to model the interaction of Teacher and Learner. This could lead us to new variations of Sabotage Games with a mix of competition and cooperation of the players, e.g. variations in which Teacher's aim is that Learner reaches the goal within a certain number of rounds and Learner starts with a safety objective but switches to reachability after some rounds.

We will very briefly come back to the learning theoretical interpretation of Sabotage Games at certain points throughout the remainder of this chapter. For a more detailed discussion of this interpretation of Sabotage Games, the reader is referred to Chapter 7 of Gierasimczuk (2010).

4.2.1 Complexity of three Sabotage Games

In this section, we will describe the variations of the Sabotage Games and investigate their complexity, i.e., the complexity of deciding whether a player has a winning strategy.

Definition 4.11 (Variations on Sabotage Games) We distinguish three different versions of Sabotage Games, $\mathcal{SG}^{Reach}$ (the standard Sabotage Game), $\mathcal{SG}^{Safety}$ and $\mathcal{SG}^{Coop}$. The structure on which the games are played and the moves allowed for both players remain the same as in Definition 4.4. The winning conditions are as given in Table 4.5. ◀

Game	Final Position	Winner
$\mathcal{SG}^{Reach}$	$\langle 1, \mathcal{E}, v\rangle$, with $v \in \mathbf{F}$	Runner
	$\langle 0, \mathcal{E}, v\rangle$, with $\mathcal{E}(v) = \emptyset$	Blocker
$\mathcal{SG}^{Safety}$	$\langle 1, \mathcal{E}, v\rangle$, with $v \in \mathbf{F}$	Blocker
	$\langle 0, \mathcal{E}, v\rangle$, with $\mathcal{E}(v) = \emptyset$	Runner
$\mathcal{SG}^{Coop}$	$\langle 1, \mathcal{E}, v\rangle$, with $v \in \mathbf{F}$	both
	$\langle 0, \mathcal{E}, v\rangle$, with $\mathcal{E}(v) = \emptyset$	none

Table 4.5: Three Sabotage Games

The different winning conditions correspond to different levels of Blocker's helpfulness and Runner's attitude towards reaching the set of vertices **F**. Having defined games representing various types of Blocker-Runner interaction, for each version of the game we now determine the complexity of deciding whether a player has a winning strategy.

In previous works (e.g. (Gierasimczuk et al. 2009b; Löding and Rohde 2003b)) an upper bound on the complexity of the games has been obtained by transforming the games into corresponding pointed Kripke models and then using a PSPACE model checking algorithm to check if a formula characterizing the existence of a winning strategy is true.

In the current work, we also consider different approaches, following ideas from Rohde (2006) and Klein et al. (2010), in order to show that tight complexity bounds for all three versions of the game can be given, even without using Sabotage Modal Logic. The decision problems that we investigate are the following.

Decision Problem 4.12 (Winning strategy for $\mathcal{SG}^{Reach}$ ($\mathcal{SG}^{Safety}$))
Input: *Labeled Sabotage Game* $\mathcal{SG} = \langle V, \mathcal{E}, v, \mathbf{F}\rangle$.

Question: *Does Runner have a winning strategy in* $\mathcal{SG}^{Reach}$ ($\mathcal{SG}^{Safety}$)*?* ◄

Decision Problem 4.13 (Winning strategy for $\mathcal{SG}^{Coop}$))
Input: *Labeled Sabotage Game* $\mathcal{SG} = \langle V, \mathcal{E}, v, \mathbf{F}\rangle$.

Question: *Do Runner and Blocker have a joint winning strategy in* $\mathcal{SG}^{Coop}$*?* ◄

We now investigate the complexity of this problem for the three different winning conditions.

Sabotage Game with reachability winning condition ($\mathcal{SG}^{Reach}$). For the standard Sabotage Game with reachability objective in which numbers are used to represent the multiplicity of the edges, Rohde (2006) has shown a PSPACE upper bound for deciding which player has a winning strategy. This is done by showing that each game can be transformed into an equivalent one in which both the multiplicity of the edges and the size of the set of goal vertices is bound by a constant. Then an alternating PSPACE algorithm is given to solve the game. Exactly the same ideas can be used for the labeled version of the game. Note that using these techniques then allows us to transform labeled games into equivalent games in which the multiplicity of edges is bound by a constant. This will take care of the potential problem arising from the way the multiplicity of edges is coded in labeled games (cf. Footnote 1 on page 101). PSPACE-hardness is shown by reduction from *Quantified Boolean Formula* (Löding and Rohde 2003b; Rohde 2006).

We can use these results and immediately obtain the same complexity bounds for Labeled Sabotage Game with reachability winning condition.

Theorem 4.14 $\mathcal{SG}^{Reach}$ *is* PSPACE*-complete.* ◄

Sabotage Game with safety winning condition ($\mathcal{SG}^{Safety}$). Whereas at first sight, $\mathcal{SG}^{Safety}$ and $\mathcal{SG}^{Reach}$ might seem to be duals of each other, the relationship between them is more complex due to the different nature of the players' moves:

Runner moves locally by choosing a state accessible *from the current one*, while Blocker moves globally by removing *an arbitrary edge*.

This is one of the reasons why determining the complexity of the safety game (especially the upper bound) turns out to be quite interesting in the sense that it does not immediately follow from the complexity of the reachability game.

We will start with discussing different options we have for obtaining an upper bound, thus also shedding some light on the different methods which have in general been suggested for determining the complexity of Sabotage Games in the literature. We will discuss the following three options.

1. Reduction to Sabotage Games with reachability winning condition.

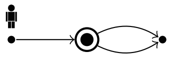
2. Using model checking of SML, as done in Gierasimczuk et al. (2009b).

3. Explicitly giving a PSPACE algorithm for solving the game.

Option 1 is a method that would fit best into the focus of this dissertation on the complexity of different interactive processes. Being able to establish a relationship between the games with reachability and safety winning condition would be conceptually nice as it would clarify the effect of changing the objectives of the players. In order to compute such a reduction, we need to find a way to transform a Sabotage Game with safety winning condition into one with reachability winning condition such that Runner has a winning strategy in the former if and only if he has one in the latter. The first intuition would be to use the idea that Runner wins the safety game if and only if he can at some point *reach* a vertex from which there is no path to **F**.

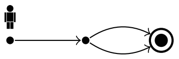
Following this line of thought would result in the following transformation. Given a safety Sabotage Game, transform it into a reachability Sabotage Game on exactly the same graph with the only difference that the new set of goal vertices **F**′ is the set of vertices from which there is no path to goal set **F** of the original game.

However, this is not a correct reduction as there can be situations in which Runner can reach a goal vertex in the reachability game and can only do it by moving through a vertex that is in **F**. We illustrate this with an example. Assume that we have the following graph for the safety game with Runner starting in the leftmost vertex and the vertex in the middle being the goal.



Then Runner will loose the safety game once he has moved.

Using the transformation explained above we will obtain the following graph for the reachability game, with Runner having the same starting position and the goal vertex now being the rightmost vertex.



As Runner is the first to move, he can indeed win this game. How could we repair the transformation in order to make it a proper reduction? One idea would be to remove the goal states of the original game from the graph of the reachability game.

For our example, this would give us the a reachability game on the following graph without any edges.

Now, Runner also looses this game as he cannot move from the initial vertex. Nevertheless, we note that the above example is very particular in the sense that the games end very quickly. In general, the above procedure does not work because in general the operation of transforming a safety game into a reachability game and the operation of one round of the game being played do not commute. This is because of the following: once Runner has made a move in the safety game, Blocker could decide to remove an edge that has the effect that now there are no paths from some vertex v to $\mathbf{F}$. Then transforming the game into a reachability game will then add v to the goal vertices $\mathbf{F}'$ of the reachability game. If on the other hand, we first transform the game and Runner and Blocker make the same moves, then no vertex is added to $\mathbf{F}'$. This shows that a reduction from safety Sabotage Games to reachability Sabotage Games cannot be done with the transformation just explained. The challenge in constructing such a reduction lies in the dynamic nature of the game as the set of vertices from which there is no path to the goal vertices eventually grows.

Let us discuss Option 2: reducing the problem of deciding if a player can win the safety Sabotage Game to the model checking problem of SML. This will indeed be discussed in detail in Section 4.2.2 where we will show how Labeled Sabotage Games can be transformed into Kripke models such that SML can be used to reason about the existence of winning strategies.

For Option 3 – directly designing a PSPACE procedure, we suggest methods developed by Klein et al. (2010) who showed that for any winning condition for the randomized Sabotage Game which can be expressed by a formula of linear temporal logic (LTL) (interpreted over the path of vertices visited by Runner through the game) deciding whether Runner has a winning strategy can be done in PSPACE. In the randomized Sabotage Game, which is similar to the *games against nature* introduced by Papadimitriou (1985), edges are removed with a certain probability. The same methods can indeed also be applied for our safety Sabotage Game. Expressing the safety winning condition as a formula of LTL is straightforward as it boils down to saying that the path has to satisfy that at every state $\neg goal$ has to be true and moreover at some point a vertex is reached that does not have a successor.

To summarize our discussion of how a PSPACE upper bound can be shown for the safety Sabotage Game: both SML and LTL model checkers can be used on a formula that expresses the existence of a winning strategy for Runner. A direct reduction from safety Sabotage Games to reachability Sabotage Games still needs to be developed and would have the great conceptual benefit of clarifying the (non-trivial) relationship between both games.

We also show PSPACE-hardness of $\mathcal{SG}^{Safety}$. This can be done by a reduction from *Quantified Boolean Formula*. The proof follows similar ideas as the one of the reachability Sabotage Game being PSPACE-hard, as shown in Theorem 2.20 of Rohde (2006). For every formula that can be the input for QBF, we construct a labeled multi-graph with an initial vertex and a set of goal vertices such that the formula is true if and only if Runner has a winning strategy in the game with safety winning condition starting in the initial vertex. The idea of the multi-graph is that for each quantifier occurring in the formula we have a component in the graph such that in the components for the existential quantifiers Runner chooses a value for the respective variable and in the components corresponding to the universal quantifiers, Blocker chooses the value. The graph is constructed in a way that forces the players to go through the components in the order as given by the formula. The final component of the graph corresponds to the Boolean formula containing the variables bound by the quantifiers. It contains subcomponents for each clause in the formula and Blocker chooses which of the components Runner has to move to. Once Runner is in such a clause component, he has the choice of moving to different vertices that each represent a literal occurring in the clause. The graph is constructed in a way that Runner can win if and only if one of the literals is true under the assignment given earlier in the game.

Theorem 4.15 *$\mathcal{SG}^{Safety}$ is* PSPACE*-complete.*

Proof. $\mathcal{SG}^{Safety}$ being in PSPACE follows from Theorem 4.20 of the next section and the fact that model-checking of SML is in PSPACE.

PSPACE-hardness is proved by showing that the *Quantified Boolean Formula* (QBF) problem, known to be PSPACE-complete, can be polynomially reduced to $\mathcal{SG}^{Safety}$. Let φ be an instance of QBF, i.e., a formula:

$$\varphi := \exists x_1 \forall x_2 \exists x_3 \dots Q x_n \psi$$

where Q is the quantifier $\exists$ for n odd, and $\forall$ for n even, and ψ is a quantifier-free formula in conjunctive normal form.

We will now construct a directed labeled multi-graph on which $\mathcal{SG}^{Safety}$ is played, such that Runner has a winning strategy in the game iff φ is true. We first describe the essential components of the graph (a $\forall$-gadget for each universally quantified variable in the formula, a $\exists$-gadget for each existential quantifier in the formula, and a verification gadget for the quantifier free formula ψ), and the key properties they have with respect to how they force the course of the game. After that we show that with this construction it holds that Runner can win the safety game on that graph if and only if the formula is true.

The $\exists$-gadget. Figure 4.1 shows the subgraph corresponding to the existential quantifier for x_i, with i odd. Curly edges represent edges with a multiplicity of five and the symbol # represents a dead end. Edges drawn as $\leftrightarrow$ represent two edges, each going in one direction. The numbers labeling edges stand for the multiplicity of the edges, thus an edge labeled with n represents n edges each with a different label from $1, \ldots, n$. The vertices represented by circles are the vertices in **F**.

When Runner enters the gadget at A, he can choose between going towards X_i or going to $\bar{X}_i$. In our interpretation, Runner moves to X_i if he wants to make x_i false and to $\bar{X}_i$ if he wants to make x_i true. Once Runner has chosen which path to take, there is no choice for Blocker other than removing the four edges from the variable vertex that Runner is approaching to the dead end #. If Blocker removes an edge somewhere else in the graph, Runner will win since when he arrives at the variable vertex (i.e., X_i or $\bar{X}_i$), there is still an edge left to #. So, we can assume that when Runner reaches the variable vertex, there is only one edge to # left, which Blocker will then remove. Then in the next round, Runner will move to B because the only other choice would be to move to the goal vertex in which case he would loose immediately. When Runner is at vertex B, Blocker is forced to remove the edge from B to the other variable vertex as that one still has four edges to #, and thus if Blocker doesn't prevent Runner from going there, he will win. Then in the next round Runner is forced to exit the gadget at B, because moving back to the variable vertex he came from is of no use for him as Blocker can then remove the edge back to B, thus forcing Runner to move to the goal.

The $\forall$-gadget. Figure 4.2 shows the subgraph corresponding to the universal quantifier for x_i, with i even. In this component, Blocker can choose the value of the variable x_i. Runner enters the gadget at A, and in the next three rounds, he moves towards C.

If Blocker wants Runner to pass through X_i (corresponding to Blocker choosing to make x_i false) she does the following: While Runner is on his way from A to C, she removes the three edges from C to $\bar{X}_i$. Then when it is Runner's turn to move from C, no edge to $\bar{X}_i$ is left and he is forced to move via D to X_i. During these four rounds, Blocker has to remove the four edges from X_i to #, because otherwise Runner can win by moving to # once he has arrived at X_i.

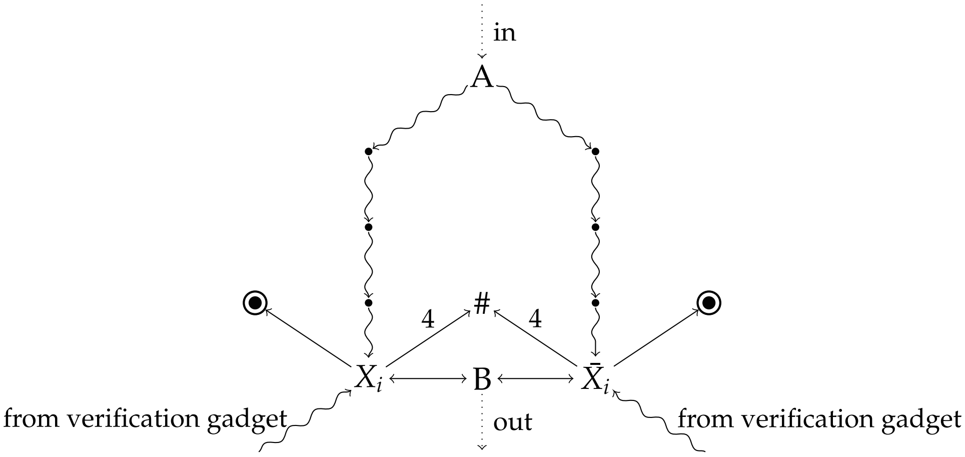


Figure 4.1: The ∃-gadget

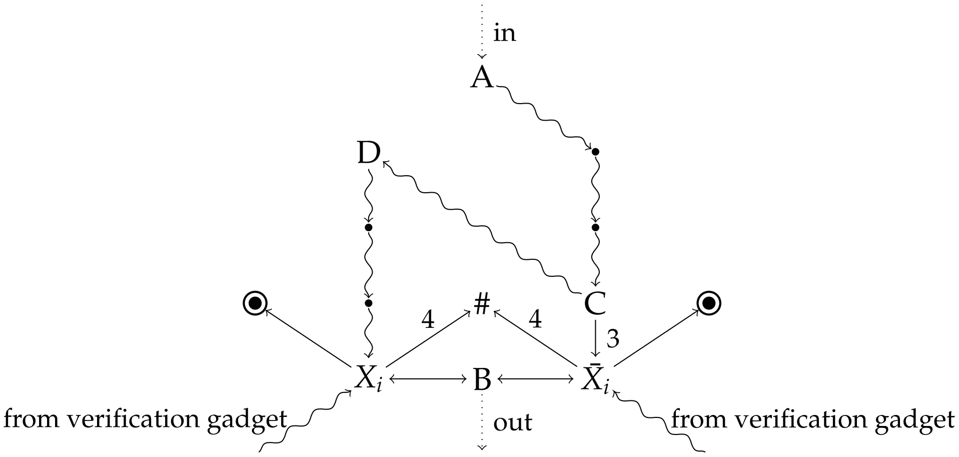


Figure 4.2: The ∀-gadget

So, when it is Runner's turn again, he will move to B because his only other option is moving to **F**. Now, Blocker has to remove the edge from B to $\bar{X}_i$ in order to prevent Runner from moving to $\bar{X}_i$ from where he could win, as there are still four edges to #. So, next Runner will exit the gadget.

If Blocker wants Runner to pass through $\bar{X}_i$ (to make x_i true), the analysis is more complicated as Runner can not actually be forced to traverse the gadget via $\bar{X}_i$. In what follows, we first analyze what will happen if Runner indeed moves as intended and then we consider the case when he does not. With Runner starting to move along the path from A, Blocker can now remove three of the four edges from $\bar{X}_i$ to #. Then when it is Runner's turn to move from C, there is one edge left from $\bar{X}_i$ to #. Now, there are in principle two options for Runner: first, to move to $\bar{X}_i$ as intended. Second, to move to D and from there to X_i. In the first case, if Runner moves to $\bar{X}_i$, then Blocker has to remove the last edge to #. Then Runner will move to B as the only other option is to move to **F**. When Runner is in B, Blocker has to remove the edge leading to X_i, preventing Runner from reaching # via X_i. Then Runner will exit the gadget.

Now, consider the case that Runner decides to move from C to D instead of moving to $\bar{X}_i$. If this happens, Blocker has to make sure that all the four edges are removed before it is Runner's turn again to move from X_i. Then Runner will move from X_i to B because the only other possibility would be to move to **F**. Once Runner reaches B, the gadget looks like in Figure 4.3 and it is Blocker's turn. Then there are three possible scenarios.

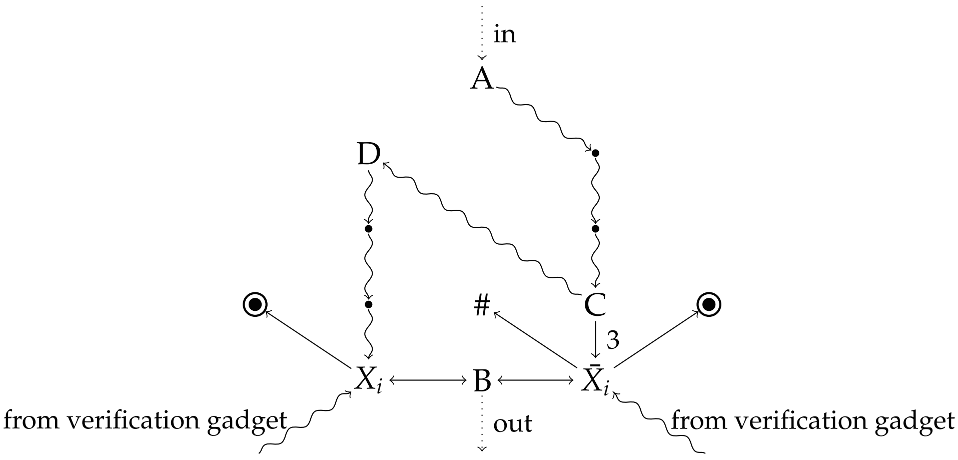


Figure 4.3: The ∀-gadget when Runner does not move to $\bar{X}_i$ as intended, with B being the current position of Runner, and Blocker being the one to move.

1. Blocker removes the remaining edge from $\bar{X}_i$ to #.

2. Blocker removes the edge from B to $\bar{X}_i$.

3. Blocker removes an edge somewhere else in the graph.

In Case 1, Runner will leave the gadget because he would not want to go to $\bar{X}_i$ or X_i as Blocker could force him to move to **F** from there.

In Case 2, Runner will also leave the gadget since as in the first case he would not want to move back to X_i.

In Case 3, if Runner moves to $\bar{X}_i$, Blocker can still remove the last edge to #, in which case Runner will move back to B. Then it is Blocker's turn again. No matter what she does, if Runner does not want to loose, he will exit the gadget, as returning to one of the variable vertices leads him to a situation in which Blocker can force him to move to **F**.

Note that in Case 1, if Runner returns to one of the variable vertices later via an edge from the verification gadget then Blocker can win by removing the edge to B thus forcing Runner to move to **F**. In the two other cases, there is one edge left from $\bar{X}_i$ to # and when Runner ever comes back to $\bar{X}_i$ from the verification gadget, Blocker can still make sure that Runner does not reach #, and neither any other dead end.

Thus, we can conclude that in the case that Blocker chooses to make x_i true and acts as described above (i.e., she removes three of the four edges from $\bar{X}_i$ to #), there is no benefit for Runner in moving from C to D. Thus, we can assume that Runner will then indeed move to $\bar{X}_i$ as intended.

The verification gadget. Figure 4.4 shows the verification gadget for the case that ψ consists of k clauses, i.e., $\psi = c_1 \wedge c_2 \wedge \ldots \wedge c_k$. Each subgraph with root vertex C_j represents the clause c_j in ψ. Accordingly, each vertex L_{jh} represents the literal l_{jh} occurring in clause c_j, i.e., $c_j = l_{j1} \vee l_{j2} \vee \ldots \vee l_{jm_j}$. If the literal is of the form x_i then there is an edge from L_{jh} to X_i in the corresponding quantifier gadget (∃-, if i is odd and ∀- otherwise). If the literal is of the form $\neg x_i$, then there is an edge from L_{jh} to $\bar{X}_i$ in that corresponding quantifier gadget.

Runner enters the gadget at the top, and Blocker can choose which clause vertex C_j, Runner has to move to. He can do this by removing the edges from $A_{j'}$ to $C_{j'}$ for all $j' < j$, and the edge from A_j to A_{j+1} (or to #, if $j = k$). Note that this way Runner has no choice as to where to move until he reaches C_j. Once Runner reaches C_j, Blocker has to remove the edge from there to #. Then Runner can choose to move to one of the literal vertices L_{jh}. At this point, Blocker can remove an edge somewhere in the graph. The crucial point here is that if l_{jh} is true under the assignment chosen through the traversal of the quantifier gadget for the variable in l_{jh}, then Runner can win because there are still at least three edges left from X_i to # in case $l_{jh} = x_i$, and analogously at least three edges from $\bar{X}_i$ to # if $l_{jh} = \neg x_i$. This is because in the intended play, in the traversal of the quantifier gadgets, if x_i was chosen to be true, $\bar{X}_i$ was visited and the edges between X_i and # remained intact, and analogously for the case when x_i was chosen to be true. For the converse, if l_{jh} is false under the assignment chosen through the traversal of the quantifier gadget for the variable in l_{jh} then Blocker can win because the variable vertex that Runner will move to has been visited before and thus all the edges from it to # have been removed and Blocker can force Runner to move to **F**.

Now, to summarize the essential properties of the construction: if φ is true then in each ∃-gadget Runner can choose a truth value for x_i for i odd such that once Blocker has forced him to move to some clause vertex C_j, there is some literal vertex that Runner can move to that leads to a variable vertex back in one of the quantifier gadgets that has enough edges left to #, allowing Runner to move there and win. For the other direction, if φ is false then Blocker can choose truth values for the x_i with i even such that when the verification gadget is reached, she can force Runner to move to some C_j such that under the assignment made through the traversal of the quantifier gadgets all the literals

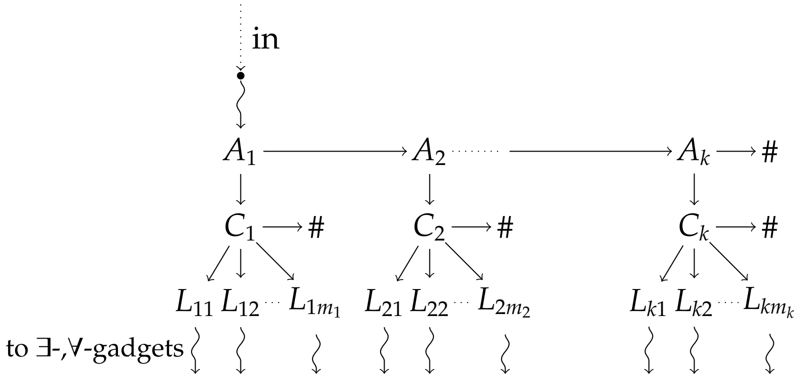

in
A_1
A_2
A_k
#
C_1
C_2
C_k
L_{11} L_{12} L_{1m_1}
L_{21} L_{22} L_{2m_2}
L_{k1} L_{k2} L_{km_k}
to ∃-,∀-gadgets

Figure 4.4: The verification gadget

occurring in the clause c_j are false, and thus the corresponding variable vertices have been visited already. So, no matter which literal vertex Runner chooses, there are not enough edges left from the variable vertex that he could arrive to #. Thus, Blocker can force him to move to **F** and Blocker wins.

Moreover, the construction of the graph for the game can clearly be done in polynomial time with respect to the size of φ. This concludes the proof of $\mathcal{SG}^{Safety}$ being PSPACE-hard. ■

Corollary 4.16 *$\mathcal{SG}^{Safety}$ is* PSPACE*-complete.* ◄

We have thus shown that the games with reachability winning condition ($\mathcal{SG}^{Reach}$) and safety winning condition ($\mathcal{SG}^{Safety}$) for Runner have the same complexity.

Cooperative Sabotage Game ($\mathcal{SG}^{Coop}$) Finally, let us look at $\mathcal{SG}^{Coop}$. This game is different from the two previous ones: Runner and Blocker win or lose together. Thus, a joint winning strategy does not need to take into account all possible moves of an opponent. This suggests that this version should be less complex than $\mathcal{SG}^{Reach}$ and $\mathcal{SG}^{Safety}$.

Note that Runner and Blocker have a joint winning strategy if and only if the goal vertex is reachable from Runner's position. Thus, determining whether they can win $\mathcal{SG}^{Coop}$ is equivalent to solving the *Reachability* (*st*-Connectivity) problem, which is known to be non-deterministic logarithmic space complete (NL-complete) (Papadimitriou 1994).

Theorem 4.17 *$\mathcal{SG}^{Coop}$ is* NL*-complete.*

Proof. The equivalence of $\mathcal{SG}^{Coop}$ and Reachability follows from the fact that Runner and Blocker have a joint winning strategy if and only if there is a path from v to an element of **F**. From left to right this is obvious. From right to left, if there is such path, then there is also one without cycles, and a joint winning strategy is e.g. one in which Runner follows this acyclic path and at each step Blocker removes the edge that has just been used by Runner. ■

Note that Theorem 4.17 relies on the fact that Runner is the first to move. If Blocker was the first to move, it is easy to construct a graph in which **F** is reachable in one step from Runner's initial position, and moreover this edge is the only edge of the graph so that Blocker is actually forced to make **F** unreachable.

Table 4.6 summarizes the complexity results for the different Sabotage Games.

Game	Winning Condition	Complexity
$\mathcal{SG}^{Reach}$	Runner wins iff he reaches **F**, Blocker wins otherwise	PSPACE-complete.
$\mathcal{SG}^{Safety}$	Runner wins iff he can avoid **F** throughout the game, Blocker wins otherwise.	PSPACE-complete.
$\mathcal{SG}^{Coop}$	Both players win iff Runner reaches the goal state. Both lose otherwise.	NL-complete.

Table 4.6: Complexity Results for Sabotage Games

This section has shown that Sabotage Games with a safety objective for Runner have the same complexity as the standard version of the game, whereas the cooperative version of the game is easier. It is important to note here that the results seem to crucially rely on the fact that Runner is the first to move. Moreover, some of our constructions used directed graphs, which leads to the question of whether the complexity for safety games changes in the case of undirected graphs.

We will now make the connection between Sabotage Games and Sabotage Modal Logic more precise.

4.2.2 Sabotage Games in Sabotage Modal Logic

Sabotage Modal Logic is useful for reasoning about graph-like structures where edges can be removed; in particular, it is useful for reasoning about Sabotage Games. In order to do that, we need to transform the structure on which the Labeled Sabotage Game is played into a Pointed Sabotage Model where formulas of the logic can be interpreted. The straightforward construction is as follows.

Definition 4.18 Let $\mathcal{SG} = \langle V, \mathcal{E}, v, \mathbf{F}\rangle$ be a Labeled Sabotage Game with $\mathcal{E} = (\mathcal{E}^a)_{a\in\Sigma}$. The Pointed Sabotage Model $PSM(\mathcal{SG})$ over the atomic propositions $\mathtt{PROP} := \{goal\}$ is given by

$$PSM(\mathcal{SG}) := \Big(\langle V, \mathcal{E}, \mathsf{V}\rangle, v\Big)$$

where $\mathsf{V}(goal) := \mathbf{F}$. ◀

In the light of this construction, Sabotage Modal Logic becomes useful for reasoning about players' strategic power in Sabotage Games. Each winning condition in Table 4.5 can be expressed by a formula of SML that characterizes the existence of a winning strategy, that is, the formula is true in a given pointed Sabotage Model if and only if the corresponding player has a winning strategy in the game represented by the model.

Reachability Sabotage Game ($\mathcal{SG}^{Reach}$) Consider $\mathcal{SG}^{Reach}$, the original Sabotage Game (van Benthem 2005). Define the formula γ_n^{Reach} inductively as follows:

$$\gamma_0^{Reach} := goal, \qquad \gamma_{n+1}^{Reach} := goal \vee \Diamond \boxminus \gamma_n^{Reach}.$$

The following result is Theorem 7 of Löding and Rohde (2003b), now for Labeled Sabotage Games. We provide a detailed proof to show how our *labeled* definition avoids a technical issue present in the original proof.

Theorem 4.19 *Runner has a winning strategy in the* $\mathcal{SG}^{Reach}$ $\mathcal{SG} = \langle V, \mathcal{E}, v, \mathbf{F}\rangle$ *if and only if* $PSM(\mathcal{SG}) \models \gamma_n^{Reach}$, *where n is the number of edges of* $\mathcal{SG}$, *i.e.,* $n = \sum_{a\in\Sigma} \mid \mathcal{E}_a \mid$.

Proof. The proof is by induction on n.

Base case (n = 0).

(⇒) If Runner has a winning strategy in a game $\mathcal{SG}$ with no edges, then he should be already in $\mathbf{F}$, i.e., $v \in \mathbf{F}$. Thus, $v \in \mathsf{V}(goal)$ so $PSM(\mathcal{SG}) \models goal$ and hence, $PSM(\mathcal{SG}) \models \gamma_0^{Reach}$.

(⇐) If $PSM(\mathcal{SG}) \models \gamma_0^{Reach}$ then $PSM(\mathcal{SG}) \models goal$. But then $v \in \mathsf{V}(goal)$ so $v \in \mathbf{F}$ and therefore Runner wins $\mathcal{SG}$ immediately.

Inductive case.

(⇒) Suppose $\mathcal{SG}$ has $n+1$ edges, and assume Runner has a winning strategy. There are two possibilities: Runner's current position is in $\mathbf{F}$ (i.e., $v \in \mathbf{F}$), or it is not.

In the first case, we get $v \in \mathsf{V}(goal)$, hence $PSM(\mathcal{SG}) \models goal$ and then $PSM(\mathcal{SG}) \models \gamma_{n+1}^{Reach}$. In the second case, since Runner has a winning strategy in $\mathcal{SG} = \langle V, \mathcal{E}, v_0, \mathbf{F}\rangle$, there is some state $v' \in V$ with $v' \in \mathcal{E}(v)$ such that in all games $\mathcal{SG}^{-(u,u'),a_j} = \langle V, \mathcal{E}^{-(u,u'),a_j}, v', \mathbf{F}\rangle$ that result from Blocker removing the edge (u, u') from a relation labeled a_j, Runner as a winning strategy.

All such games have n edges, so by inductive hypothesis we have

$$PSM(\mathcal{SG}^{-(u,u'),a_j}) \models \gamma_n^{Reach}$$

for every edge $(u,u') \in V \times V$ and label $a_j \in \Sigma$ such that $(u,u') \in \mathcal{E}_{a_j}$. Now, the key observation is that each such $PSM(\mathcal{SG}^{-(u,u'),a_j})$ is exactly the model that results from removing edge (u,u') with a_j from the *pointed model* $\big(\langle V, \mathcal{E}, \mathsf{V}\rangle, v'\big)$.[2] Then, for all such (u,u') and a_j, we have

$$\big(\langle V, \mathcal{E}, \mathsf{V}\rangle^{-(u,u'),a_j}, v'\big) \models \gamma_n^{Reach}.$$

It follows that $\big(\langle V, \mathcal{E}, \mathsf{V}\rangle, v'\big) \models \boxminus\gamma_n^{Reach}$ and therefore $\big(\langle V, \mathcal{E}, \mathsf{V}\rangle, v\big) \models \Diamond\boxminus\gamma_n^{Reach}$, that is, $PSM(\mathcal{SG}) \models \gamma_{n+1}^{Reach}$.

(⇐) Suppose $PSM(\mathcal{SG}) \models goal \vee \Diamond\boxminus\gamma_n^{Reach}$, and recall that $PSM(\mathcal{SG})$ is given by $\big(\langle V, \mathcal{E}, \mathsf{V}\rangle, v\big)$. Then, $v \in \mathbf{F}$ or else there is a state $v' \in \mathcal{E}(v)$ such that $\big(\langle V, \mathcal{E}, \mathsf{V}\rangle, v'\big) \models \boxminus\gamma_n^{Reach}$, i.e.,

$$\big(\langle V, \mathcal{E}, \mathsf{V}\rangle^{-(u,u'),a_j}, v'\big) \models \gamma_n^{Reach}$$

for all edges (u,u') and labels $a_j \in \Sigma$ with $(u,u') \in \mathcal{E}_{a_j}$. By inductive hypothesis, Runner has a winning strategy in each game that corresponds to each pointed model $\big(\langle V, \mathcal{E}, \mathsf{V}\rangle^{-(u,u'),a_j}, v'\big)$, but these games are exactly those that result from removing any edge from the game $\langle V, \mathcal{E}, v, \mathbf{F}\rangle$ after Runner moves from v to v'. Hence, Runner has a winning strategy at $\langle V, \mathcal{E}, v, \mathbf{F}\rangle$, the game that corresponds to the pointed model $PSM(\mathcal{SG})$, as required. ■

We have thus seen that Sabotage Modal Logic can express the existence of a winning strategy for Runner in $\mathcal{SG}^{Reach}$; the crucial point in the proof being the fact that with Labeled Sabotage Games the operations of removing an edge and building the corresponding Kripke model commute. We now continue by expressing the existence of a winning strategy in the safety game using Sabotage Modal Logic.

[2] In the original definition of a Sabotage Game in which the edges are given by a function from a pair of vertices to a natural number (Definition 4.2) this is not the case. With that definition, transforming such a model into a Kripke model does not commute with the operation of removing an edge. First transforming a game into a model and then removing an edge does not always give the same result as first removing an edge and then transforming the game into a model. This is because removing and edge from the model does not mean that we have to remove the edge with the label corresponding to the highest number.

Safety Sabotage Game ($\mathcal{SG}^{Safety}$) Now consider $\mathcal{SG}^{Safety}$, the game in which Blocker tries to force Runner to go to **F** and Runner tries to avoid reaching **F**. Inductively, we define γ_n^{Safety} as

$$\gamma_0^{Safety} := \neg goal \wedge \Box\bot, \qquad \gamma_{n+1}^{Safety} := \neg goal \wedge (\Box\bot \vee \Diamond\boxminus\gamma_n^{Safety}).$$

We show that this formula corresponds to the existence of a winning strategy for Runner. The idea of the proof is the same as for the reachability game.

Theorem 4.20 *Runner has a winning strategy in the $\mathcal{SG}^{Safety}$ $\mathcal{SG} = \langle V, \mathcal{E}, v, \mathbf{F}\rangle$ if and only if $PSM(\mathcal{SG}) \models \gamma_n^{Safety}$, where $n = \sum_{a\in\Sigma} \mid \mathcal{E}_a \mid$.*

Proof. The proof is by induction on n.

Base case (n = 0).

(⇒) If Runner has a winning strategy in a game $\mathcal{SG}$ with no edges, then this means that he cannot be in **F** and moreover he cannot move anywhere as there are no edges, i.e., $v \notin \mathbf{F}$ and $\mathcal{E}(v) = \emptyset$. Thus, $PSM(\mathcal{SG}) \models \neg goal \wedge \Box\bot$.

(⇐) If $PSM(\mathcal{SG}) \models \neg goal \wedge \Box\bot$ then $v \notin \mathbf{F}$ and $\mathcal{E}(v) = \emptyset$. Therefore, Runner wins $\mathcal{SG}$ immediately.

Inductive case.

(⇒) Assume that $\sum_{a\in\Sigma} \mid \mathcal{E}_a \mid= n+1$ and Runner has a winning strategy. There are two possibilities: Runner wins immediately, or he doesn't.

In the first case, we get $v \notin \mathsf{V}(goal)$ and $\mathcal{E}(v) = \emptyset$, hence $PSM(\mathcal{SG}) \models \neg goal \wedge \Box\bot$ and thus $PSM(\mathcal{SG}) \models \gamma_{n+1}^{Safety}$. In the second case, since Runner has a winning strategy in $\mathcal{SG} = \langle V, \mathcal{E}, v, \mathbf{F}\rangle$, he cannot have lost immediately and thus $v \notin \mathbf{F}$ (which means that $PSM(\mathcal{SG}) \models \neg goal$) and there is some state $v' \in V$ with $v' \in \mathcal{E}(v)$ such that in all games $\mathcal{SG}^{-(u,u'),a_j} = \langle V, \mathcal{E}^{-(u,u'),a_j}, v', \mathbf{F}\rangle$ that result from Blocker removing edge (u,u') from the relation labeled a_j, Runner has a winning strategy.

All such games have n edges, so by inductive hypothesis we have

$$PSM(\mathcal{SG}^{-(u,u'),a_j}) \models \gamma_n^{Safety}$$

for every edge $(u,u') \in V \times V$ and label $a_j \in \Sigma$ such that $(u,u') \in \mathcal{E}_{a_j}$. Now, each such $PSM(\mathcal{SG}^{-(u,u'),a_j})$ is exactly the model that results from removing edge (u,u') with label a_j from the *pointed model* $\big(\langle V, \mathcal{E}, \mathsf{V}\rangle, v'\big)$. Then, for all such (u,u') and a_j, we have

$$\big(\langle V, \mathcal{E}, \mathsf{V}\rangle^{-(u,u'),a_j}, v'\big) \models \gamma_n^{Safety}.$$

It follows that $\big(\langle V, \mathcal{E}, \mathsf{V}\rangle, v'\big) \models \boxminus\gamma_n^{Safety}$ and therefore $\big(\langle V, \mathcal{E}, \mathsf{V}\rangle, v\big) \models \Diamond\boxminus\gamma_n^{Safety}$. Hence, $\big(\langle V, \mathcal{E}, \mathsf{V}\rangle, v\big) \models \neg goal \wedge (\Box\bot \vee \Diamond\boxminus\gamma_n^{Safety})$, i.e $PSM(\mathcal{SG}) \models \gamma_{n+1}^{Safety}$.

(⇐) Assume that $PSM(\mathcal{SG}) \models \neg goal \wedge (\Box\bot \vee \Diamond\boxminus\gamma_n^{Safety}$, and recall that $PSM(\mathcal{SG})$ is given by $\big(\langle V, \mathcal{E}, \mathsf{V}\rangle, v\big)$. Then, $v \notin \mathbf{F}$ and either $\mathcal{E}(v) = \emptyset$ (and Runner wins $\mathcal{SG}$ immediately) or there is a state $v' \in \mathcal{E}(v)$ such that $\big(\langle V, \mathcal{E}, \mathsf{V}\rangle, v'\big) \models \boxminus\gamma_n^{Safety}$, i.e.,

$$\big(\langle V, \mathcal{E}, \mathsf{V}\rangle^{-(u,u'),a_j}, v'\big) \models \gamma_n^{Safety}$$

for all edges (u, u') and labels $a_j \in \Sigma$ with $(u, u') \in \mathcal{E}_{a_j}$. By inductive hypothesis, Runner has a winning strategy in each game that corresponds to each pointed model $\big(\langle V, \mathcal{E}, \mathsf{V}\rangle^{-(u,u'),a_j}, v'\big)$, but these games are exactly those that result from removing any edge from the game $\langle V, \mathcal{E}, v, \mathbf{F}\rangle$ after Runner moves from v to v'. Hence, Runner has a winning strategy in $\langle V, \mathcal{E}, v, \mathbf{F}\rangle$, the game that corresponds to the pointed model $PSM(\mathcal{SG})$, as required. ■

Cooperative Sabotage Game ($\mathcal{SG}^{Coop}$) Finally, for $\mathcal{SG}^{Coop}$, the corresponding formula is defined as

$$\gamma_0^{Coop} := goal, \qquad \gamma_{n+1}^{Coop} := goal \vee \Diamond\mathbin{\Diamond\!\!\!\!\!-}\gamma_n^{Coop}.$$

Theorem 4.21 *Blocker and Runner have a joint winning strategy in the $\mathcal{SG}^{Coop}$ $\mathcal{SG} = \langle V, \mathcal{E}, v, \mathbf{F}\rangle$ if and only if $PSM(\mathcal{SG}) \models \gamma_n^{Coop}$, where n is the number of edges of $\mathcal{SG}$.*

Proof. As argued in the proof of Theorem 4.17, Runner and Blocker have a joint winning strategy if and only if there is a path from v to an element of $\mathbf{F}$. The theorem follows by observing that γ_n^{Coop} expresses the existence of such path, keeping in mind that Blocker can remove edges used by Runner. ■

The above results are summarized in Table 4.7.

Game	Winning Condition in SML	Winner
$\mathcal{SG}^{Reach}$	$\gamma_0^{Reach} := goal, \quad \gamma_{n+1}^{Reach} := goal \vee \Diamond\boxminus\gamma_n^{Reach}$	Runner
$\mathcal{SG}^{Safety}$	$\gamma_0^{Safety} := \neg goal \wedge \Box\bot, \quad \gamma_{n+1}^{Safety} := \neg goal \wedge (\Box\bot \vee \Diamond\boxminus\gamma_n^{Safety})$	Runner
$\mathcal{SG}^{Coop}$	$\gamma_0^{Coop} := goal, \quad \gamma_{n+1}^{Coop} := goal \vee \Diamond\mathbin{\Diamond\!\!\!\!\!-}\gamma_n^{Coop}$	Both

Table 4.7: Winning Conditions for $\mathcal{SG}$ in SML

Thus, the existence of winning strategies for all three winning conditions can be expressed in SML. After establishing a relationship between SML and Sabotage Games, let us briefly come back to the learning theoretical interpretation of Sabotage Games.

Learning theoretical interpretation of the results. The existence of a winning strategy in a Sabotage Game with learning interpretation corresponds to the existence of a strategy for Learner, Teacher or both (depending on the variation of the game) to ensure that the learner reaches the learning goal for the learner, or the teacher, or both, depending on the winning condition. The complexity results for Sabotage Games thus give us the complexity of PSPACE of deciding whether a there is a strategy (for Learner or Teacher, depending on whether we have a reachability or safety winning condition) that guarantees that Learner reaches the learning goal in the non-cooperative case, and NL in the cooperative case.

What does the undecidability of the satisfiability problem of SML mean with the learning interpretation? As formulas of SML express the abilities of Teacher and Learner regarding Learner's abilities to reach a state with certain properties (e.g. the state being the learning goal), this means that there are specifications about the learning and teaching abilities of Learner and Teacher, respectively such that it cannot be decided whether this is a consistent specification, i.e., we can design a learning scenario according to the specification. Concluding the learning theoretical interpretation of the complexity results, we can say that the analysis of a learning scenario with respect to the learning and teaching abilities of Learner and Teacher, respectively can be done in PSPACE and in the cooperative case it can be done in NL. Giving a procedure for designing a scenario from given specifications is in general impossible.

In this chapter, up to now we have considered different Sabotage Games that differed only in the winning positions, while the available moves for the players were exactly the same. Next, we will consider a variation in which Blocker has additional moves to choose from.

4.3 Allowing breaks

As mentioned before, the players' moves are asymmetric: Runner moves locally (moving to a vertex accessible *from the current one*) while Blocker moves globally (removing *any* edge from the graph, and thereby manipulating the space in which Runner is moving). Intuitively, both in the case that Blocker wants to stop Runner from reaching **F** and in the case that she tries to force him to reach **F**, it is not always necessary for her to react to a move of Runner immediately. This leads us to a variation of a $\mathcal{SG}$ in which Runner's move does not in principle need to be followed by Blocker's move; i.e., Blocker has the possibility of skipping a move.

Definition 4.22 A *Sabotage Game without strict alternation* (for Blocker) is a tuple $\mathcal{SG}_* = \langle V, \mathcal{E}, v_0, \mathbf{F}\rangle$. Moves of Runner are as in $\mathcal{SG}$ and, once he has chosen a vertex v', Blocker can choose between removing an edge, in which case the next

	Position	**Player**	**Admissible moves**
1.	$\langle 0, \mathcal{E}, v\rangle$	Runner	$\left\{\langle 1, \mathcal{E}, u'\rangle \mid (u, u') \in \mathcal{E}^{a_i} \text{ for some } a_i \in \Sigma\right\}$
2.	$\langle 1, \mathcal{E}, u'\rangle$	Blocker	$\left\{\langle 0, \mathcal{E}^{-(v,v'),a_j}, u'\rangle \mid (v, v') \in \mathcal{E}^{a_j} \text{ for some } a_j \in \Sigma\right\} \cup \left\{\langle 0, \mathcal{E}, u'\rangle\right\}$

Table 4.8: A round in the Labeled Sabotage Game without strict alternation

game is given as in $\mathcal{SG}$, and doing nothing, in which case the game continues as $\langle V, \mathcal{E}, v', \mathbf{F}\rangle$; Table 4.8 shows a round in this game. Again, there are three versions with different winning conditions, now called $\mathcal{SG}_*^{Reach}$, $\mathcal{SG}_*^{Safety}$ and $\mathcal{SG}_*^{Coop}$. ◄

After defining the class of games $\mathcal{SG}_*$, the natural question that arises is how the winning abilities of the players change from $\mathcal{SG}$ to $\mathcal{SG}_*$, since in the latter Blocker can choose between removing an edge or doing nothing. In the rest of this section, we show that for all three winning conditions ($\mathcal{SG}_*^{Reach}, \mathcal{SG}_*^{Safety}, \mathcal{SG}_*^{Coop}$), the winning abilities of the players remain the same as in the case in which players move in strict alternation. This is surprising in the case of $\mathcal{SG}^{Safety}$, since we might expect that with the possibility of skipping a move Blocker would not be forced to remove an edge that leads to the goal.

We start with the reachability game $\mathcal{SG}_*^{Reach}$. Note that even though in this new setting matches can be infinite, in fact if Runner can win the game, he can do so in a finite number of rounds. We now give a lemma stating that if Runner can win some $\mathcal{SG}^{Reach}$ in some number of rounds, then he can do so also if the underlying multi-graph has additional edges.

Definition 4.23 (Supergraph of a directed labeled multi-graph) Let $\Sigma = \{a_1, \ldots, a_m\}$ be a finite set of labels. For directed labeled multi-graphs, $\mathcal{H} = (V, \mathcal{E})$ and $\mathcal{H}' = (V', \mathcal{E}')$, we say that $\mathcal{H}'$ is a supergraph of $\mathcal{H}$ if $V \subseteq V'$ and $\mathcal{E}^{a_i} \subseteq \mathcal{E}'^{a_i}$ for all labels $a_i \in \Sigma$. ◄

Lemma 4.24 *If Runner has a strategy for winning the $\mathcal{SG}^{Reach} = \langle V, \mathcal{E}, v, \mathbf{F}\rangle$ in at most n rounds, then he can also win any $\mathcal{SG}^{Reach} = \langle V, \mathcal{E}', v, \mathbf{F}\rangle$ in at most n rounds, where $(V, \mathcal{E}')$ is a supergraph of $(V, \mathcal{E})$.*

Proof. The proof is by induction on the number of rounds n. In the inductive step, for the case that Blocker removes an edge which was not in the original multi-graph, note that the resulting graph is a supergraph of the original one. Then we can use the inductive hypothesis. ■

Theorem 4.25 *Consider the $\mathcal{SG} = \langle V, \mathcal{E}, v, \mathbf{F}\rangle$ with $(V, \mathcal{E})$ a directed labeled multigraph, v a vertex and* **F** *a subset of vertices. Runner has a winning strategy in the corresponding $\mathcal{SG}^{Reach}$ iff he has a wining strategy in the corresponding $\mathcal{SG}_*^{Reach}$.*

Proof. From left to right, we show by induction on n that Runner can win the $\mathcal{SG}^{Reach}$ in at most n rounds, then he can also win the $\mathcal{SG}_*^{Reach}$ in at most n rounds. In the inductive step, in the case that Blocker responds by not removing any edge, we first use the previous lemma and can then apply the inductive hypothesis.

The direction from right to left is immediate: if Runner has a winning strategy for $\mathcal{SG}_*^{Reach}$, then he can also win the corresponding $\mathcal{SG}^{Reach}$ by using the same strategy. ■

The case of Runner trying to avoid reaching **F**, i.e., the game $\mathcal{SG}_*^{Safety}$, is more interesting. One might expect that the additional possibility of skipping a move gives more power to Blocker, since she could avoid removals that would have made the goal unreachable from the current vertex. However, we can show that this is not the case. First we state two lemmas. The first one says that if from Runner's current position there is a path to **F** and no path to a vertex from which there is no path to **F**, then Blocker can win. The idea is that wherever Runner moves, he will stay on a path to **F**, so Blocker can make sure that Runner will eventually reach **F**.

Lemma 4.26 *Consider the $\mathcal{SG}_*^{Safety} = \langle V, \mathcal{E}, v, \mathbf{F}\rangle$. If there is a path from v to a vertex in* **F** *and there is no path from v to a vertex from where there is no path to* **F** *then Blocker has a winning strategy.*

Proof. Suppose that all vertices reachable from v are on paths to **F**. Then even if Blocker refrains from removing any edge, Runner will be on a path to **F**. Now, either the path to **F** does not include a loop or it does. If it does not then Blocker can simply wait until Runner is at **F**. If it does, Blocker can remove the edges that lead to the loops in such a way that **F** is still reachable from any vertex. ■

The next lemma says that if Blocker can force Runner to reach **F**, then she can do so as well on a graph that is the same as the original one except that an edge is removed which is not on a path to **F**.

Lemma 4.27 *For all $\mathcal{SG}_*^{Safety} = \langle V, \mathcal{E}, v, \mathbf{F}\rangle$, if Blocker has a winning strategy and there is an edge $(u, u') \in \mathcal{E}^a$ for some $a \in \Sigma$ such that no path from v to a vertex in* **F** *uses (u, u'), then Blocker also has a winning strategy in $\langle V, \mathcal{E}^{-(u,u'),a}, v, \mathbf{F}\rangle$.*

Proof. If u is not reachable from v, it is easy to see that the claim holds. Assume that u is reachable from v. Blocker's winning strategy should prevent Runner from moving from u to u' (otherwise Runner wins). Hence, Blocker can also win if (u, u') is not there. ■

Using the two previous lemmas, we now show that if Blocker can win $\mathcal{SG}_*^{\mathit{Safety}}$, then she can also win by not skipping any moves.

Theorem 4.28 *Blocker has a winning strategy in the* $\mathcal{SG}_*^{\mathit{Safety}} = \langle V, \mathcal{E}, v, \mathbf{F}\rangle$*, then she also has a winning strategy in which she removes an edge in each round.*

Proof. The proof proceeds by induction on the number of edges $n = \sum_{a\in\Sigma} |\mathcal{E}^a|$.

The base case is straightforward. For the inductive case, assume that Blocker has a winning strategy in $\mathcal{SG}_*^{\mathit{Safety}} = \langle V, \mathcal{E}, v, \mathbf{F}\rangle$ with $\sum_{a\in\Sigma} |\mathcal{E}^a| = n + 1$.

If $v \in \mathbf{F}$ we are done. Otherwise, since Blocker can win, there is some $v' \in V$ such that $(v, v') \in \mathcal{E}^a$ for some $a \in \Sigma$ and for all such v' we have:

1. there is a path from v' to $\mathbf{F}$ and
2. (a) Blocker can win $\langle V, \mathcal{E}, v', \mathbf{F}\rangle$, or
   (b) there are $(u, u') \in V \times V$ and $a \in \Sigma$ such that $(u, u') \in \mathcal{E}^a$ and Blocker can win $\langle V, \mathcal{E}^{-(u,u'),a}, v', \mathbf{F}\rangle$.

If 2b holds, since $\sum_{a\in\Sigma} |\mathcal{E}'^a| = n$, we are done: we use the inductive hypothesis to conclude that Blocker has a winning strategy in which she removes an edge in each round (in particular, her first choice is to remove (v, v') from $\mathcal{E}^a$. Let us show that 2b holds.

If there is some $(u, u') \in V \times V$ such that $(u, u') \in \mathcal{E}^a$ for some $a \in \Sigma$ and this edge is not part of any path from v' to $\mathbf{F}$ then by Lemma 4.27, Blocker can remove this edge and 2*b* holds, so we are done.

If all edges in $(V, \mathcal{E})$ belong to a path from v' to $\mathbf{F}$, from 1, there are two cases: either there is only one, or there is more than one path from v' to $\mathbf{F}$.

In the first case (only one path), the edge (v, v') can be chosen since it cannot be part of the *unique* path from v' to $\mathbf{F}$. Assume now that there is more than one path from v' to $\mathbf{F}$. Let $p = (w_1, \dots, w_g)$ with $v' = w_1$ be the/a shortest path from v' to a $w_g \in \mathbf{F}$. This path cannot contain any loops. Then, from this path take w_i such that i is the smallest index for which it holds that from w_i there is a path $(w_i, w'_{i+1}, \dots, w'_g)$ to a $w'_g \in \mathbf{F}$ that is at least as long as the path following p from w_i (i.e., $(w_i, w_{i+1}, \dots, w_g)$).

Intuitively, when following path p from v' to $\mathbf{F}$, w_i is the first point at which Runner can deviate from p in order to take another path to $\mathbf{F}$ (recall that we consider the case where every vertex in the graph is part of a path from v' to a $\mathbf{F}$-state). Now it is possible for Blocker to choose $((w_i, w'_{i+1}), a)$ such that $(w_i, w'_{i+1}) \in \mathcal{E}^a$. Then, after removing (w_i, w'_{i+1}) from $\mathcal{E}^a$ we are in the game $\langle V, \mathcal{E}^{-(w_i,w'_{i+1}),a}, v', \mathbf{F}\rangle$. Note that due to the way we chose the edge to be removed, in the new graph it still holds that from v there is no path to a vertex from which a $\mathbf{F}$-state is not reachable (this holds because from w_i $\mathbf{F}$ is still reachable). Then by Lemma 4.26, Blocker can win $\langle V, \mathcal{E}^{-(w_i,w'_{i+1}),a}, v, \mathbf{F}\rangle$, which then implies 2*b*.

Hence, we conclude that 2*b* is the case and thus using the inductive hypothesis, Blocker can win $\langle V, \mathcal{E}, v, \mathbf{F}\rangle$ also by removing an edge in every round. ■

Corollary 4.29 *Blocker has a $\mathcal{SG}_*^{Safety}$ winning strategy in $\langle V, \mathcal{E}, v, \mathbf{F}\rangle$ iff she has a $\mathcal{SG}^{Safety}$ winning strategy.* ◄

As the reader might have noticed, the result that if Blocker can win a $\mathcal{SG}_*^{Safety}$ then she can also win the corresponding $\mathcal{SG}^{Safety}$ relies on the fact that Runner is the first to move. For instance, in a graph with two vertices, the initial v and a single goal state v_g, and one edge leading from the first to the second, if Blocker was to move first, she can win the $\mathcal{SG}_*^{Safety}$ only by skipping the move.

Finally, let us consider the cooperative case.

Theorem 4.30 *If Runner and Blocker have a joint $\mathcal{SG}_*^{Coop}$-winning strategy in $\langle V, \mathcal{E}, v, \mathbf{F}\rangle$ then they have a joint $\mathcal{SG}^{Coop}$-winning strategy*

Proof. If the players have a joint $\mathcal{SG}_*^{Coop}$-winning strategy, then there is an acyclic path from v to $\mathbf{F}$, which Runner can follow. In each round, Blocker can remove the just used edge. ■

Let us briefly conclude this section. We have shown that in $\mathcal{SG}$, allowing Blocker to skip moves does not change the winning abilities of the players. Using these results, both the complexity and definability results for all three winning conditions from the previous section also apply to the games in which Blocker can skip a move.

Allowing *Runner* to take a break. The reader might wonder why we did not consider the variation in which Runner has the possibility of skipping a move. With reachability winning condition, it is easy to see that Runner would not want to use the option of skipping a move as this will just have the effect of the graph getting smaller and thus restricting his way to the goal states. Similarly, in the cooperative case Runner skipping a move does not have any advantages for the team of Runner and Blocker. In the game with safety winning condition on the other hand, giving Runner the possibility of skipping moves results in a trivial game which Runner can win if and only if he starts at a vertex which is not in $\mathbf{F}$. He can do so by simply skipping all the moves and waiting until eventually all goal vertices have become unreachable.

4.4 Conclusions and Further Questions

We will now summarize the main results of this chapter and then give conclusions and further questions.

4.4.1 Summary

This chapter has analyzed the complexity of deciding if a player has a winning strategy in a game on a graph in which players have asymmetric kinds of moves.

We considered three different winning conditions for Sabotage Games and showed that the complexity of the game with safety winning condition is the same as the complexity of the reachability version (PSPACE-complete).

We have discussed different options of getting upper bounds on the complexity for the safety game, thus clarifying different methods used in the literature.

The analysis in this chapter has shown that Sabotage Games are also PSPACE-complete when Runner has a safety objective and wants to avoid reaching certain states. The upper bound was obtained by giving a reduction from the standard Sabotage Games, thus clarifying the relationship between the two games, which is interesting as roles of the players are asymmetric with Runner moving locally and Blocker moving globally.

Moreover, we have shown that cooperation makes the Sabotage Game much easier (NL-complete). We also considered further variations of the game that allow Blocker to take a break and skip a move. We showed that this additional freedom for Blocker does not have any influence on the strategic abilities of the players and neither on the complexity of the games. We have not considered a version in which Runner can skip some moves, as this will not help him when he has a reachability objective and in the case of the safety game as we have defined it here, this variation will lead to a trivial game as Runner either looses immediately or does not have any incentive to move.

Throughout this chapter, we analyzed the complexity of deciding whether a player has the ability to win, no matter what the other player does. Our results of Section 4.2.2 connect this to the complexity analysis of the previous chapters in which we were concerned with the complexity of logics for reasoning about strategic ability of agents. We have seen that the existence of winning strategies in Sabotage Games can be expressed in Sabotage Modal Logic. Thus, deciding if a player has a winning strategy can be done by using a model checking algorithm to check if the corresponding formula is true in the model that corresponds to the game.

4.4.2 Conclusions

Let us come back to our research question.

> **Research Question 2** *What is the role of cooperation vs. competition in the complexity of interaction?*

- *Does analyzing an interactive situation in general become easier if the participants cooperate?*

Our main conclusions towards answering this question are the following.

1. Based on the results in this chapter, we can conclude that in the context of Sabotage Games, cooperation has the effect of making it easier to decide if a winning strategy exists.

2. Even though the precise connection between the two different non-cooperative versions (with safety and reachability objective) does not seem straightforward, they are equivalent with respect to the complexity of deciding who can win.

Complexity of logical theories of interaction vs. complexity of interaction. This chapter was motivated by the need to study problems which are more interaction-specific than model checking and satisfiability of logics for interaction, as studied in the previous two chapters. In the light of this, we note that for Sabotage Modal Logic, model checking formulas characterizing the winning strategies in non-cooperative safety and reachability games is among the hardest tasks in model checking Sabotage Modal Logic.

Learning theoretical interpretation of our results. Interpreting the results of this chapter in terms of formal learning theory, with Runner being Learner and Blocker being Teacher, we conclude the following

1. Deciding if in the non-cooperative case learning or teaching can be successful is PSPACE-complete.

2. Deciding if cooperative learning and teaching can be successful is easier (NL-complete).

3. It does not matter for successful learning and teaching if Teacher does not give feedback after each step of Learner.

4. On a more general level, considering Sabotage Modal Logic as a logic for reasoning about Learner-Teacher interaction we conclude that the analysis of learning scenarios with respect to the abilities of the two participants is PSPACE-complete. The problem of deciding whether it is possible to design a scenario according to specifications given in Sabotage Modal Logic however is undecidable.

In the other direction, the learning theoretical perspective can also lead us to new analyses of the complexity of Sabotage Games, inspired from learning theory (Gierasimczuk and de Jongh 2010).

4.4.3 Further Questions

The work in this chapter gives rise to some interesting new directions for further research.

Let us start with some technical questions regarding the variations on Sabotage Games discussed in this chapter.

Technical questions about $\mathcal{SG}^{Safety}$ and $\mathcal{SG}^{Reach}$.

- Is $\mathcal{SG}^{Safety}$ also PSPACE-hard on *undirected* graphs?

  In our proof of Theorem 4.15 we used directed multi-graphs for showing hardness. In fact, the directedness is crucial in the reduction that we give. It remains to be investigated whether a similar reduction can also be constructed for undirected multi-graphs.

- How can $\mathcal{SG}^{Safety}$ be transformed into corresponding $\mathcal{SG}^{Reach}$?

  In Section 4.2.1, we have discussed some ideas for transforming reachability games into safety games. Such a transformation still has to be worked out and would have the great conceptual benefit of clarifying the relationship between the two games.

Variations of Sabotage Modal Logic. The analysis of Sabotage Games given in this chapter gives rise to some variations of Sabotage Modal Logic which are motivated by game variations.

- From Sabotage Modal Logic to "Pacman's Logic".

  A key feature of Sabotage Games is the asymmetry between the players as Runner acts locally while Blocker acts globally. Eliminating this asymmetry by making Blocker's moves also local, then leads to a game closely related to *Pacman* (Heckel 2006), in which Blocker moves along the edges and removes them by moving along them. Based on this game variation, we can construct a variation of Sabotage Modal Logic in which the sabotage modalities act locally. We will briefly come back to this in the last chapter.

- From skipping a move to iterating moves.

  While we have discussed the variation in which Blocker is allowed to skip a move, there would also be the possibility to allow players to make a sequence of moves. With respect to the corresponding logic, this would then lead to the addition of something like a Kleene star operator.

What is the complexity involved in actually playing Sabotage Games? While the analysis given in this chapter does focus on more specific problems in interaction than the two previous chapters, the question as to what we can conclude about the complexity of actually playing Sabotage Games is mostly left open. However, we can draw some conclusions as to the complexity of evaluating a game configuration with respect to ones own abilities to win: In general, this is intractable (as it is PSPACE-complete) both for the reachability game and for the safety game.

For getting better grip on the complexity involved in actually playing the game, concrete board game implementations of Sabotage Games should be considered; or taking this even a step further also a pervasive game version can be constructed and investigated as has been done e.g. for Pacman (Magerkurth et al. 2005).

Let us briefly sum up what we have done so far in our quest for getting a better grip on the complexity of interaction. In Part I we started with logical frameworks for social action and determined the complexity of some of such theories. Our perspective on interactive processes was quite abstract and external. The concepts that we considered in that context included coalitional power, preferences and actions. In the current chapter we considered a concrete type of game, but still take an external perspective from which we reason about the abilities of players to win. For being able to draw some conclusions not only about the difficulties of reasoning about interaction but also of interacting itself, we have to switch perspective and zoom in more into the reasoning of agents. This leads us to the concept of *information*.

From strategic ability to information. The concept of information was only implicitly present in the previous two chapters and in the current one. From the way the strategic abilities were distributed among the agents in modal logics for cooperation and the way Sabotage Games are defined, we can draw some conclusions about the uncertainty that agents have: in Sabotage Games for instance, the players have perfect information about what has happened so far in the game and also how the graph looks like; in CLA+P or ABC on the other hand, at each state we can think of agents making their choices simultaneously, thus not knowing what the others' choices are.

We will now move on to an explicit investigation of the fine-structure of agents' information and concrete tasks that arise when analyzing information structures.

With respect to our analysis of complexity in interaction, we will now investigate which parameters make reasoning about agents' information difficult.

Chapter 5

Exploring the Tractability Border in Epistemic Tasks

The different complexity analyses given in the previous chapters have in common that they investigate the complexity from an external perspective, focusing on how difficult it is to describe and reason about the abilities of agents. We will now zoom in more into the actual tasks involved in *epistemic* reasoning in multi-agent scenarios.

So far, we have been concerned with the preferences and abilities of agents to perform actions or to achieve certain outcomes. There is an important aspect of interaction which we have not considered explicitly yet, which is that of *information*. This concept was only implicitly present in the frameworks previously discussed.

Information plays a central role in interactions. Agents have information about the actual situation they are in, information about the information of other agents etc. For strategic reasoning in concrete games, the level of the higher-order reasoning about information required can also influence the types of strategies employed (cf. Raijmakers et al. (2011)).

In this chapter, we analyze the complexity of interaction in terms of the complexity of reasoning about information that agents have. To be more precise, we investigate the complexity of comparing and manipulating the information of agents. We do this within the semantic structures of (epistemic) modal logics, i.e., structures as the Kripke model constructed in the example of the card playing train passengers (Example 1.5). In this setting, we will address the following research question.

Research Question 3 *Which parameters can make interaction difficult?*

- *How does the complexity of an interactive situation change when more participants enter the interaction or when we drop some simplifying assumptions on the participants themselves?*

5.1 From the Complexity of Epistemic Logics to the Complexity of Epistemic Tasks

Epistemic modal logics and their extensions are concerned with global and abstract problems in reasoning about information. They are designed to model a wide range of epistemic scenarios (cf. Fagin et al. (1995); Baltag and Moss (2004)). As logics have to be quite complex in order to be able to express various scenarios and problems in epistemic reasoning, it is not surprising that there are many intractability and even undecidability results in the literature (see e.g. Halpern and Vardi (1989) and van Benthem and Pacuit (2006) for a survey). Consequently, the issue of trade-off between expressivity and complexity plays a central role in the field of epistemic modal logics.

The existing complexity results of modal logics provide us with an overview of the difficulty of epistemic reasoning in modal logic frameworks from an abstract global perspective. In this chapter, we zoom in into epistemic reasoning and take a more agent-oriented perspective. Our main aim is to initiate the mapping of the tractability border among epistemic tasks rather than epistemic logics. As a result, we can identify a theoretical threshold in the difficulty of reasoning about information, similarly to how this has been done in the context of reasoning with quantifiers (cf. Pratt-Hartmann and Moss (2009); Szymanik (2010)). In order to do this, we shift our perspective: Instead of investigating the complexity of a given logic that can be used to describe certain tasks in epistemic reasoning, we turn towards a complexity study of the concrete tasks themselves, determining what computational resources are needed in order to perform the required reasoning.

At this point, we would like to clarify that we do not propose a new formal model for epistemic reasoning from an internal agent-oriented perspective. For two approaches to modeling epistemic scenarios such as the muddy children puzzle in a more concise way than standard epistemic logic models, we refer the reader to Gierasimczuk and Szymanik (2011) and Wang (2010). For a version of epistemic logic in which the modeler is one of the agents, we refer to Aucher (2010). In this chapter, we work with models from epistemic modal logic and investigate the complexity of various interesting specific problems that arise when reasoning about these semantic structures.

Focusing on specific problems, the complexity may be much lower since concrete problems involved in the study of multi-agent interaction are rarely as general as e.g. satisfiability. In most cases, checking whether a given property is satisfied in a given (minimal) epistemic scenario is sufficient. This may sound as if we study the model checking complexity of different properties in epistemic logic. Indeed, some of the simpler tasks and problems we consider boil down to model checking (data complexity) epistemic formulas. However, we want to point out that we study the problems in purely semantic terms and our complexity results are thus independent of how succinctly, if at all, the properties could be expressed in an (extended) epistemic modal logic. This is thus different from the complexity results for model-checking that we gave in Chapter 3, where we considered the *combined* complexity, taking both models and the formula that we want to check as input of the decision problem. The problems we consider in this chapter take epistemic models and sometimes also additional parameters as input and ask whether the models satisfy certain properties or whether the models are in a certain relation.

Many of the concrete problems we study turn out to be tractable. Still, we will see that even in this perspective there are some intractable problems. We believe that this feasibility border in epistemic tasks is an interesting new topic for a formal study, which also has the potential of leading to an empirical assessment of the cognitive plausibility of epistemic logic frameworks. The cognitive plausibility of the tractability that our study aims to identify can later be tested for its correlation with the difficulties faced by human agents solving such tasks (cf. Verbrugge (2009); Szymanik and Zajenkowski (2010)).

So in a sense, we aim to initiate a search for an appropriate perspective and complexity measures that describe in plausible ways the cognitive difficulties agents face while interacting. Certain experimental results in the economics literature explore similar directions for specific game settings (Feltovich 2000; Weber 2001).

In this chapter we investigate the computational complexity of various decision problems that are relevant for interactive reasoning in epistemic modal logic frameworks. In particular, we explore the complexity of comparing and manipulating information structures possessed by different agents.

With respect to the comparison of information structures we are interested in whether agents have similar information (about each other) or whether one of them has more information.

Information similarity and symmetry

- Is one agent's information strictly less refined than another agent's information?
- Do two agents have the same knowledge/belief about each other's knowledge/belief?

In a situation with diverse agents that have different information, the question arises as to whether it is possible that some of the agents can be provided with information so that afterward the agents have similar information.

Information manipulation

- Given two agents, is it possible to give some information to one of them such that as a result

  - both agents have similar information structures? (cf. van Ditmarsch and French (2009))
  - one of them has more refined information than the other?

Determining the complexity of the above questions will then help to analyze how the complexity of various reasoning tasks is influenced by

- the choice of similarity notion taken for similarity of information structures,
- the choice of information structures,
- the number of agents involved.

5.1.1 Preliminaries

We will briefly give some preliminaries of (epistemic) modal logic. We use relational structures from epistemic logic to model information (cf. Blackburn et al. (2001); Fagin et al. (1995)). Kripke models can compactly represent the information agents have about the world and about the information possessed by the other agents. It is frequently assumed that information structures are partition-based (Aumann 1999; Fagin et al. 1995; Osborne and Rubinstein 1994):

Definition 5.1 (Epistemic Models) An *epistemic model* is a (multi-agent) Kripke model such that for all $i \in \mathtt{N}$, R_i is an equivalence relation. (We usually write $\sim_i$ instead of R_i). ◀

The associated modal logic is S5, which adds the following axioms to the basic system $\mathbf{K}_{\mathtt{N}}$.

- T-axiom: $\Box_i p \rightarrow p$, corresponding to reflexivity

  (veridicality: what is known is true),

- 4-axiom: $\Box_i p \rightarrow \Box_i \Box_i p$, corresponding to transitivity

  (positive introspection),

- $S5$-axiom: $\Diamond_i \Box_i p \rightarrow p$, corresponding to symmetry

  (negative introspection).

Instead of $\Box_i$, sometimes K_i is used, as $\Box_i \varphi$ is supposed to mean that *i knows that* φ.

In this chapter we will only work with the semantic structures of epistemic modal logic and not with the logics itself. We refer the reader interested in epistemic modal logic to the literature (van Ditmarsch et al. 2007; van Benthem 2010, 2011).

Intuitively, with an epistemic interpretation, an accessibility relation R_i in a Kripke model encodes i's uncertainty: if wR_iv, then if the actual world was w then i would consider it possible that the actual world is v. We write $\mathcal{K}_i[w] := \{v \in W \mid wR_iv\}$ to denote i's information set at w. For epistemic models for one agent, we sometimes also write $[w]$ to denote the equivalence class of w under the relation $\sim$, i.e., $[w] = \{w' \in W \mid w \sim w'\}$. For a group of agents G we write $R_G = \cup_{i \in G} R_i$, and $R^*_G[w] := \{v \in W \mid wR^*_Gv\}$. For any non-empty set $G \subseteq \mathtt{N}$, we write R^*_G for the reflexive transitive closure of $\bigcup_{i \in G} R_i$.

The notion of horizon generalizes that of an information set:

Definition 5.2 (Horizon) The *horizon* of i at $(\mathcal{M}, w)$ (notation: $(\mathcal{M}, w)^i$) is the submodel generated by $\mathcal{K}_i[w]$. ◄

The domain of $(\mathcal{M}, w)^i$ thus contains all the states that can be reached from w by first doing one step along the relation of agent i and then doing any number of steps along the relations of any agents.

This chapter will not use syntactic notions. In terms of intuition, the important definition is that of knowledge K_i: at w, agent i knows that φ iff it is the case that φ is true in all states that i considers possible at w. In equivalent semantic terms: at w, i knows that some event $E \subseteq W$ is the case iff $\mathcal{K}_i[w] \subseteq E$. E is common knowledge in a group G at w iff $R^*_G[w] \subseteq E$.

In the technical parts of this chapter, we use complexity results from graph theory (see e.g. Garey and Johnson (1990)). Here, we use the connection between Kripke models and graphs: graphs are essentially Kripke models without valuations, i.e., *frames* (Blackburn et al. 2001).

For graphs, the notion of *induced subgraph* is just like that of submodel (Definition 1.11) without the condition for the valuations. The notion of *subgraph* is weaker than that of an induced subgraph as it allows that $R'_i \subset R_i \cap W' \times W'$.

5.2 Complexity of comparing and manipulating information

In this section, we give the results we obtained when studying the complexity of different epistemic reasoning tasks in the semantic structures of modal logics.

The tasks we investigate deal with three different aspects.

Information similarity (Section 5.2.1).

Are the information structures of two agents similar?

Information symmetry (Section 5.2.2).

Do two agents have the same (similar) information about each other?

Information manipulation (Section 5.2.3).

> Can we manipulate the information of one agent such that as a result he knows at least as much as another agent?

5.2.1 Information similarity

The first natural question we would like to address is whether an agent in a given situation has similar information to the one possessed by some other agent (in a possibly different situation). One very strict way to understand such similarity is through the use of isomorphism.

For the general problem of checking whether two Kripke models are isomorphic, we can give the following complexity bounds, in the sense that the problem is equivalent to the graph isomorphism problem. The graph isomorphism problem is neither known to be NP-complete nor to be in P (see e.g. Garey and Johnson (1990)) and the set of problems with a polynomial-time reduction to the graph isomorphism problem is called GI.

Decision Problem 5.3 (Kripke model isomorphism)
Input: *Pointed Kripke models* $(\mathcal{M}_1, w_1)$, $(\mathcal{M}_2, w_2)$.

Question: *Are* $(\mathcal{M}_1, w_1)$ *and* $(\mathcal{M}_2, w_2)$ *isomorphic, i.e., is it the case that* $(\mathcal{M}_1, w_1) \cong (\mathcal{M}_2, w_2)$*?* ◄

Fact 5.4 *Kripke model isomorphism is GI-complete.*

Proof. Kripke model isomorphism is equivalent to the variation of graph isomorphism with labeled vertices, which is polynomially equivalent to graph isomorphism (see e.g. Hoffmann (1982)), and thus GI-complete.

However, isomorphism is arguably a too restrictive notion of similarity. Bisimilarity is a weaker concept of similarity. As we take a modal logic perspective in this chapter and want to analyze the complexity of epistemic tasks on the semantic structures of epistemic modal logic, bisimilarity is a very natural choice of similarity.

Here the question arises as to whether working with S5 models – a common assumption in the epistemic logic and interactive epistemology literature – rather than arbitrary Kripke structures has an influence on the complexity of the task.

Decision Problem 5.5 (Epistemic model bisimilarity)
Input: *Two pointed multi-agent epistemic S5 models* $(\mathcal{M}_1, w_1), (\mathcal{M}_2, w_2)$.

Question: *Are the two models bisimilar, i.e.,* $(\mathcal{M}_1, w_1) \underline{\leftrightarrow} (\mathcal{M}_2, w_2)$*?* ◄

Balcázar et al. (1992) have shown that deciding bisimilarity is P-complete for finite labeled transition systems. As epistemic models are just a special kind of labeled transition systems, we can use an algorithm that solves bisimilarity for labeled transition systems also for epistemic models. It follows that epistemic model bisimilarity is also in P.

Fact 5.6 *Multi-agent epistemic S5 model bisimilarity can be done in polynomial time with respect to the size of the input* $(|\mathcal{M}_1| + |\mathcal{M}_2|)$. ◂

Thus, multi-agent epistemic S5 model bisimilarity is in P. Now, of course the question arises if it is also P-hard.

Proposition 5.7 *Multi-agent epistemic S5 model bisimilarity is* P*-complete.*

Proof. P membership follows immediately from Fact 5.6. For P-hardness, we can adapt the hardness proof of Balcázar et al. (1992). In the reduction from monotone alternating circuits, the labeled transition systems that are constructed are irreflexive. We can transform them into corresponding S5 models for two agents using the method also used in Halpern and Moses (1992) and replace every edge $w \to v$ by $w \sim_1 w' \sim_2 v$, keeping the valuation of w and v the same as before and making the valuation of w' the same as that of w. Additionally, reflexive loops have to be added. Bisimilarity of two irreflexive finite structures is invariant under this transformation. Moreover, note that for the replacement of the edges, we only need constant memory space. P-hardness follows. ■

To summarize, while deciding Kripke model isomorphism lies on the tractability border, deciding whether two Kripke models are bisimilar is among the hardest problems that are known to be in P. For S5 epistemic models with at least two agents, we get the same results.

5.2.2 Information symmetry: knowing what others know

The preceding notions of similarity are very strong as they are about the similarity of whole information structures. In the context of analyzing epistemic interactions between agents, weaker notions of similarity are of interest as often already the similarity of some relevant parts of information are sufficient for drawing some conclusions. In general, the information that agents have about each other's information state plays a crucial role. We will now analyze the problem of deciding whether two agents' views about the interactive epistemic structure, and in particular about the knowledge of other agents, are equivalent. A first reading is simply to fix some fact $E \subseteq W$ and ask whether E is common knowledge in a group G. Clearly this problem is tractable.

Fact 5.8 *Given a pointed multi-agent epistemic model $(\mathcal{M}, w)$, some $E \subseteq Dom(\mathcal{M})$ and a subset of agents $G \subseteq \mathtt{N}$, deciding whether E is common knowledge in the group G at w can be done in polynomial time.*

Proof. To decide if E is common knowledge among the agents in G, we can use a reachability algorithm for checking if any of the states which are not in E (i.e., any state in $Dom(\mathcal{M}) \setminus E$) is reachable from w by a *path* along the relation $\cup_{j \in G} \sim_j$. If this is the case, then the answer is *no*, otherwise the answer is *yes* as then $\sim_G^* [w] \subseteq E$. ■

If some fact is common knowledge between two agents, the information of the two agents about this fact can be seen as symmetric, in the sense that both agents have the same information about the fact and about the information they have about the fact. More generally, instead of fixing some specific fact of interest, an interesting question is whether an epistemic situation is symmetric with respect to two given agents, say Ann and Bob. In other words, is the interactive informational structure from Ann's perspective similar to how it is from Bob's perspective? We first introduce some notation that we will use for representing situations in which the information of two agents is exchanged, in the sense that each of the agents gets exactly the information that the other one had before.

Definition 5.9 For a Kripke model $\mathcal{M} = (W, (R_i)_{i \in \mathtt{N}}, V)$, with $j, k \in \mathtt{N}$, we write $\mathcal{M}[j/k]$ to be the model $(W, (R'_i)_{i \in \mathtt{N}}, V)$ for $R'_i = R_i$ for $i \notin \{j, k\}$, $R'_j = R_k$ and $R'_k = R_j$. ◄

So, in the model $\mathcal{M}[j/k]$ agent j gets the accessibility relation of k in $\mathcal{M}$ and vice versa.

The intuition is that in many multi-agent scenarios it can be interesting to determine if the situation is symmetric w.r.t. two given agents in the sense that those two agents have similar information about facts, other agents and also about each other. As a typical such situation consider a two-player card game. Here, it can be crucial for the strategic abilities of the players whether they both know equally less about each other's cards and whether they know the same about each other's information. From a modeling perspective, determining if the information of two agents is interchangeable can also be crucial if we want to find a succinct representation of the situation (cf. Chapter 7 of Wang (2010)), as in some situations only explicitly representing one of the agents might be sufficient.

To formalize this property of information symmetry, we introduce the notion of *flipped* bisimulation for a pair of agents. The main difference w.r.t. a standard bisimulation is that for each step along the accessibility relation for one agent in one model, there has to be a corresponding step along the relation of the *other* agent in the other model.

Definition 5.10 We say that two pointed multi-agent epistemic models $(\mathcal{M}, w)$ and $(\mathcal{M}', w')$ (with set of agents $\mathtt{N}$) are flipped bisimilar for agents $i, j \in \mathtt{N}$, $(\mathcal{M}, w)\underline{\leftrightarrow}_f{}^{(ij)}(\mathcal{M}', w')$, iff $(\mathcal{M}, w)\underline{\leftrightarrow}(\mathcal{M}'[i/j], w')$.

So, two models are flipped bisimilar for two agents if after swapping the accessibility relations of the two agents in one of the models, the resulting model is bisimilar to the other model. To help to get an intuition of this notion, we list two facts about flipped bisimilarity that follow directly from its definition.

Fact 5.11 *For any pointed multi-agent Kripke models $(\mathcal{M}, w), (\mathcal{M}', w')$ with set of agents $\mathtt{N}$ and agents $i, j \in \mathtt{N}$ the following hold.*

- $(\mathcal{M}, w)\underline{\leftrightarrow}_f^{(ii)}(\mathcal{M}', w')$ *iff* $(\mathcal{M}, w)\underline{\leftrightarrow}(\mathcal{M}', w')$,
- $(\mathcal{M}, w)\underline{\leftrightarrow}_f^{(ij)}(\mathcal{M}', w')$ *iff* $(\mathcal{M}, w)\underline{\leftrightarrow}_f^{(ji)}(\mathcal{M}', w')$.

Moreover, in general we can have that $(\mathcal{M}, w)\not\underline{\leftrightarrow}_f^{(ij)}(\mathcal{M}, w)$, i.e., the relation of flipped bisimilarity is not reflexive, and thus not an equivalence relation. ◀

Thus, flipped bisimilarity for the same agent is equivalent to regular bisimilarity. While the relation of flipped bisimilarity for a pair of agents is not reflexive, flipped bisimilarity is indeed symmetric with respect to the flipping of the agents.

Note that it is *not* the case that for all models $(\mathcal{M}, w), (\mathcal{M}', w'), (\mathcal{M}'', w'')$ with set of agents $\mathtt{N}$ and agents $i, j, k \in \mathtt{N}$, if $(\mathcal{M}, w)\underline{\leftrightarrow}_f^{(ij)}(\mathcal{M}', w')\underline{\leftrightarrow}_f^{(jk)}(\mathcal{M}'', w'')$, then $(\mathcal{M}, w)\underline{\leftrightarrow}_f^{(ik)}(\mathcal{M}'', w'')$. This is because the performing two consecutive swaps of agents is in general not equivalent to performing one swap of agents.

In the context of epistemic multi-agent models, the following question arises: How does flipped bisimilarity relate to knowledge of the individual agents and common knowledge?

The following is immediate: If on a whole model it holds that everything that two individual agents know is common knowledge among them, then every state is flipped bisimilar (for these two agents) to itself. The intuition here is that if everything that the two individuals know is commonly known among them, then the two agents have exactly the same information and can thus be swapped.

Observation 5.12 *If for a multi-agent S5 model $\mathcal{M} = (W, (\sim_i)_{i \in \mathtt{N}}, V)$, it holds that $\sim^*_{\{i,j\}} \subseteq \sim_i \cap \sim_j$ for some $i, j \in \mathtt{N}$, then for all $w \in W$, $(\mathcal{M}, w)\underline{\leftrightarrow}_f{}^{(ij)}(\mathcal{M}, w)$.* ◀

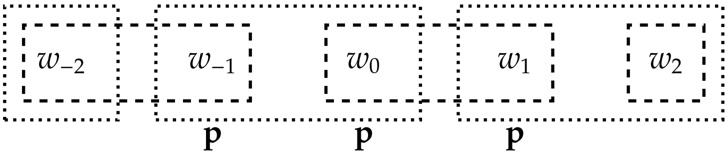
Does the other direction hold? Locally, even on S5 models, flipped self-bisimulation is much weaker than the property of individual knowledge being common knowledge: flipped self-bisimulation does not even imply that (shared) knowledge of facts is common knowledge:

Fact 5.13 *There exists a multi-agent S5 model $\mathcal{M} = (W, (\sim_i)_{i \in \mathtt{N}}, \mathsf{V})$, such that for some $i, j \in \mathtt{N}$ we have that for some $w \in W$ it holds that $(\mathcal{M}, w) \underline{\leftrightarrow}_f^{(ij)} (\mathcal{M}, w)$, and for some $p \in \textsc{prop}$ we have that $\mathcal{M}, w \models K_i p$ and $\mathcal{M}, w \models K_j p$ but p is not common knowledge among i and j at w.*

Proof. Consider the model $\mathcal{M} = (W, (\sim_i)_{i \in \mathtt{N}}, \mathsf{V})$, where

- $W = \{w_{-2}, w_{-1}, w_0, w_1, w_2\}$,
- $\mathtt{N} = \{\text{Ann}, \text{Bob}\}$,
- $\sim_{\text{Ann}}$ is the smallest equivalence relation on W containing $\{(-2, -1), (0, 1)\}$, and $\sim_{\text{Bob}}$ is the smallest equivalence relation on W containing $\{(-1, 0), (1, 2)\}$,
- $\mathsf{V}(p) = \{w_{-1}, w_0, w_1\}$.

The following figure represents $\mathcal{M}$. The dashed rectangles are the equivalence classes for Ann and the dotted rectangles those of Bob.



It is easy to check that at state w_0 both Ann and Bob know that p: $\mathcal{K}_{\text{Ann}}[w_0] = \{w_0, w_1\} \subseteq \mathsf{V}(p)$ and $\mathcal{K}_{\text{Bob}}[w_0] = \{w_{-1}, w_0\} \subseteq \mathsf{V}(p)$. But p is not common knowledge between Ann and Bob at w_0: we have that $w_0 \sim_{\text{Ann}} w_1 \sim_{\text{Bob}} w_2$ and $w_2 \notin \mathsf{V}(p)$. Now it remains to show that $(\mathcal{M}, w_0)$ is Ann, Bob-flipped bisimilar to itself. We can define a flipped bisimulation as follows $Z = \{(w_n, w_{-n}) \mid w_n \in W\}$, i.e., $Z = \{(w_{-2}, w_2), (w_{-1}, w_1), (w_0, w_0), (w_1, w_{-1}), (w_2, w_{-2})\}$. It is easy to check that Z is indeed a flipped bisimulation for Ann and Bob. ■

But required globally of every state, we do have the following converse: If for two agents we have flipped bisimilarity of every state to itself and the accessibility relations of the agents are transitive, then every fact that is known by at least one of the agents is immediately also common knowledge among the two agents.

Fact 5.14 *For every Kripke model $\mathcal{M} = (W, (R_i)_{i \in \mathtt{N}}, \mathsf{V})$ with R_i and R_j being transitive for some $i, j \in \mathtt{N}$, it holds that for each $w \in W$ we have the following. Whenever the submodel $\mathcal{M}'$ of $\mathcal{M}$ generated by $\{w\}$ is such that for every state $w' \in Dom(\mathcal{M}')$ it holds that $(\mathcal{M}', w') \underline{\leftrightarrow}_f^{(ij)} (\mathcal{M}', w')$, then for any $p \in \textsc{prop}$, if at w at least one of the two agents i and j knows that p (i.e., $\mathsf{V}(p) \subseteq K_j[w]$ or $\mathsf{V}(p) \subseteq K_i[w]$), then p is common knowledge among i and j at w.*

Proof. Assume that for some model $\mathcal{M} = (W, (R_i)_{i\in\mathtt{N}}, \mathsf{V})$ it holds that R_i and R_j are transitive for some $i, j \in \mathtt{N}$. Now, assume that p is not common knowledge between i and j at w. It follows that we have a finite i, j-path leading to a state where p is false. Let $wR_f(1)w_1Rf(2)\dots Rf(n)w_n$ with $w_n \notin \mathsf{V}(p)$ and $f(k) \in \{i, j\}$ for all $k \leq n$ be a shortest such path. Then, by transitivity of R_i and R_j it has to be the case that for all k with $1 \leq k < n$, $f(k) \neq f(k+1)$. W.l.o.g. assume that the path is of the form $wR_iw_1R_j\dots R_iw_n$; the other cases are completely analogous. Now, as all the states in the path $wR_iw_1Rj\dots R_iw_n$ are in $\mathcal{M}'$, by assumption for each w_k in the path we have $(\mathcal{M}', w_k)\underline{\leftrightarrow}_f^{(ij)}(\mathcal{M}', w_k)$. Then, in particular $(\mathcal{M}', w)$ is flipped i, j-bisimilar to itself. Then there has to be a path $wR_jw_1^1R_iw_2^1\dots R_jw_n^1$ with $w_n^1 \notin V(p)$. Then, we can continue this argument, as also $(\mathcal{M}', w_1^1)$ has to be flipped i, j-bisimilar to itself. Thus, there has to be some path $w_1^1R_jw_2^2R_j\dots R_n^2w_n^2$. Then, by transitivity of R_j, $wR_jw_2^2$. Iterating this procedure, we will finally get that there is an R_j path from w to a state where p is false. Using the transitivity of R_j, we then conclude that $M, w \not\models K_jp$.

It remains to show that at w agent i does not know that p neither. By assumption, wR_iw_1 and thus $(\mathcal{M}', w_1)$ has to be flipped i, j-bisimilar to itself. Thus, there has to be a path a $w_1R_iw_2'R_jw_3'\dots R_jw_n'$ with $w_n' \notin V(p)$ Then, by transitivity, it follows from $wR_iw_1R_iw_2'$ that wR_iw_2'. Iterating this procedure, we get a state which is R_i-accessible from w where p is false. Hence, we conclude that at w neither i nor j knows that p. This concludes the proof. ■

Let us recall the notion of an agent's horizon (Definition 5.2). It is the submodel generated by the information set of the agent: the *horizon* of i at (M, w) (notation: $(M, w)^i$) is the submodel generated by the set $\mathcal{K}_i[w]$.

We now analyze the complexity of deciding (flipped) bisimilarity of two agents' horizons at the same point in a model. We distinguish between S5 models and the class of all Kripke structures.

Proposition 5.15 *For horizon bisimilarity of multi-agent Kripke models we have the following complexity results*

1. *For multi-agent S5 models $(\mathcal{M}, w)$ with set of agents $\mathtt{N}$,*
   (a) *deciding whether $(\mathcal{M}, w)^i\underline{\leftrightarrow}(\mathcal{M}, w)^j$ is trivial.*
   (b) *deciding whether $(\mathcal{M}, w)^i\underline{\leftrightarrow}_f^{(ij)}(\mathcal{M}, w)^j$ is in* P.
2. *For multi-agent Kripke models $(\mathcal{M}, w)$ with set of agents $\mathtt{N}$,*
   (a) *deciding whether $(\mathcal{M}, w)^i\underline{\leftrightarrow}(\mathcal{M}, w)^j$ is* P*-complete.*
   (b) *deciding whether $(\mathcal{M}, w)^i\underline{\leftrightarrow}_f^{(ij)}(\mathcal{M}, w)^j$ is* P*-complete.*

Proof. 1a follows from the fact that if the agents' accessibility relations are reflexive then the horizons of the agents are the same.

This is the case because $(\mathcal{M}, w)^i$ is the submodel generated by $\mathcal{K}_i[w]$, i.e., the submodel generated by the set of states that i considers possible at w. If at w, i considers w itself possible, the domain of this submodel will also contain the domain of the submodel generated by $\mathcal{K}_j[w]$. The argument for the other direction is analogous.

1b follows from the fact that deciding flipped bisimilarity of horizons in multi-agent S5 is polynomially equivalent to deciding (flipped) bisimilarity of multi-agent S5 models. Both decision problems of 2a and 2*b* are polynomially equivalent to deciding bisimilarity of multi-agent Kripke models because in general the horizons of two agents at a point in the model can be two completely disjoint submodels. ■

Let us summarize the results we have on the complexity of deciding information symmetry. Both, deciding whether a fact is commonly known and deciding horizon flipped bisimilarity in Kripke models are tractable, with the latter being among the hardest problems known to be tractable. Flipped bisimilarity of horizons remains P-hard even if we consider the horizons of two agents at the very same point in a model. For partition-based models, however, deciding bisimilarity of the horizons of two agents at the same point in a model is trivial, whereas for flipped bisimilarity, this is harder, but still tractable (in P).

The tasks we considered so far dealt with the comparison of agents' information states in given situations. Here, we were concerned with *static* aspects of agents' information. However, in many interactive situations *dynamic* aspects play a central role, as the information of agents can change while the agents interact. There are even interactive processes where information change can be the aim of the interaction itself, e.g. interactive deliberation processes. In such contexts the question arises as to whether it is possible to manipulate the information state of agents in a particular way.

5.2.3 Can we reshape an agent's mind into some desired informational state?

The problem that we investigate in this section is to decide whether new informational states (satisfying desired properties) can be achieved in certain ways. One immediate question is whether one can give some information to an agent (i.e., to restrict the agent's horizon) such that after the update the horizon is bisimilar to the horizon of some other agent. Concretely, we would like to know if there is any type of information that could reshape some agent's information in order to fit some desired new informational state or at least be similar to it. More precisely, we will consider information that *restricts* the horizon of an agent; we do not consider the process of changing an agent's information state by introducing more uncertainty. The processes we consider are related to those modeled by *public announcement logic* (PAL), an epistemic logic with formulas of the form $[\varphi]\psi$ saying that after the announcement of φ it is the case that ψ holds. In semantic terms this means that if the current state satisfies φ (i.e., announcing φ is a truthful announcement) then after the model has been restricted to all those states at which φ is true, it is the case that ψ holds at the current state. Deciding whether such a formula $[\varphi]\psi$ holds at a state in a model (i.e., model checking this formula) thus involves first checking if φ holds at the state and then relativizing the original model to the set of states where φ holds and finally checking if ψ then holds at the current state. In order to put the complexity results of this section into perspective, note that for PAL it holds that given a pointed model and a formula, checking if the formula holds in the model can be done in time polynomial in the length of he formula and the size of the model (cf. Kooi and van Benthem (2004) for polynomial model checking results for PAL with relativized common knowledge).

The model checking problem of PAL is about deciding whether getting a particular piece of information (i.e., the information that φ holds) has a certain effect (i.e., the effect of ψ being the case). In this section, we will investigate a more general problem which is about whether it is possible to restrict the model such that a certain effect is achieved. To be more precise, we consider the task of checking whether there is a *submodel* that has certain properties. This means that we determine if it is possible to purposely refine a model in a certain way. This question is in line with problems addressed by arbitrary public announcement logic (APAL) and arbitrary event modal logic (Balbiani et al. 2008a; van Ditmarsch and French 2009)[1]. Looking at the complexity results for such logics (see e.g. French and van Ditmarsch (2008) for a proof of undecidability of SAT of APAL), we can already see that reasoning about the existence of information whose announcement has a certain effect seems to be hard. Our analysis will show whether this is also the case for concrete tasks about deciding whether a given model can be restricted such that it will have certain properties.

We start with the problem of checking whether there is a submodel of one model that is bisimilar to another one. On graphs, this is related to the problem of deciding if one graph contains a subgraph bisimilar to another graph. Note that in the problem referred to in the literature as "subgraph bisimulation" (Dovier and Piazza 2003), the subgraph can be any graph whose vertices are a subset of the vertices of the original graph, and the edges can be any subset of the edges of the original graph restricted to the subset of vertices. To be more

[1] Note that in the current work, we focus on the semantic structures only and do not require that the submodel can be characterized by some formula in a certain epistemic modal language.

specific, the problem investigated in Dovier and Piazza (2003) is the following:

> Given two graphs $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$, is there a graph $G'_2 = (V'_2, E'_2)$ with $V'_2 \subseteq V_2$ and $E'_2 \subseteq E_2$ such that there is a total bisimulation between G'_2 and G_1?

Since we want to investigate the complexity of reasoning about epistemic interaction using modal logic, we are interested in subgraphs that correspond to *relativization* in modal logic: *induced* subgraphs. This leads us to an investigation of *induced subgraph bisimulation*.

Decision Problem 5.16 (Induced subgraph bisimulation)
Input: *Two finite graphs* $G_1 = (V_1, E_1)$, $G_2 = (V_2, E_2)$, $k \in \mathbb{N}$.
Question: *Is there an induced subgraph of* G_2 *with* $\geq k$ *vertices that is totally bisimilar to* G_1*, i.e., is there a* $V' \subseteq V_2$ *with* $|V'| \geq k$ *and* $(V', E_2 \cap (V' \times V')) \underline{\leftrightarrow}_{total} G_1$*?* ◀

Even though the above problem looks very similar to the original subgraph bisimulation problem (NP-hardness of which is shown by reduction from Hamiltonian Path), NP-hardness does not follow immediately.[2] Nevertheless, we can show NP-hardness by reduction from Independent Set.

Proposition 5.17 *Induced subgraph bisimulation is* NP*-complete.*

Proof. Showing that the problem is in NP is straightforward. Hardness is shown by reduction from Independent Set. First of all, let $I_k = (V_{I_k}, E_{I_k} = \emptyset)$ with $|V_{I_k}| = k$ denote a graph with k vertices and no edges. Given the input of Independent Set, i.e., a graph $G = (V, E)$ and some $k \in \mathbb{N}$ we transform it into $(I_k, G), k$, as input for Induced Subgraph Bisimulation.

Now, we claim that G has an independent set of size at least k iff there is some $V' \subseteq V$ with $|V'| \geq k$ and $(V', E \cap (V' \times V')) \underline{\leftrightarrow}_{total} I_k$.

From left to right, assume that there is some $S \subseteq V$ with $|S| = k$, and for all $v, v' \in S$, $(v, v') \notin E$. Now, any bijection between S and V_{I_k} is a total bisimulation between $G' = (S, E \cap (S \times S))$ and I_k, since $E \cap (S \times S) = \emptyset$ and $|S| = |V_{I_k}|$.

For the other direction, assume that there is some $V' \subseteq V$ with $|V'| = k$ such that for $G' = (V', E' = E \cap (V' \times V'))$ we have that $G' \underline{\leftrightarrow}_{total} I_k$. Thus, there is some total bisimulation Z between G' and I_k. Now, we claim that V' is an independent set of G of size k. Let $v, v' \in V'$. Suppose that $(v, v') \in E$. Then since G' is an induced subgraph, we also have that $(v, v') \in E'$. Since Z is a total bisimulation, there has to be some $w \in I_k$ with $(v, w) \in Z$ and some w' with $(w, w') \in E_{I_k}$ and $(v', w') \in Z$. But this is a contradiction with $E_{I_k} = \emptyset$. Thus, V' is an independent set of size k of G. The reduction can clearly be computed in polynomial time. This concludes the proof. ■

[2] For induced subgraph bisimulation, a reduction from Hamiltonian Path seems to be more difficult, as does a direct reduction from the original subgraph bisimulation problem.

Now, an analogous result for Kripke models follows. Here, the problem is to decide whether it is possible to 'gently' restrict one model without letting its domain get smaller than k such that afterward it is bisimilar to another model. With an epistemic/doxastic interpretation of the accessibility relation, the intuitive interpretation is that we would like the new information to change the informational state of the agent as little as possible.

Decision Problem 5.18 (Submodel bisimulation for Kripke models)
Input: *Kripke models $\mathcal{M}_1$, $\mathcal{M}_2$ with set of agents $\mathtt{N}$ and some $k \in \mathbb{N}$.*

Question: *Is there a submodel $\mathcal{M}_2'$ of $\mathcal{M}_2$ with $|Dom(\mathcal{M}_2')| \geq k$ such that $\mathcal{M}_1$ and $\mathcal{M}_2'$ are totally bisimilar i.e., $\mathcal{M}_1 \underline{\leftrightarrow}_{total} \mathcal{M}_2'$?* ◀

Corollary 5.19 *Submodel bisimulation for Kripke models is* NP*-complete.*

Proof. Checking if a proposed model is indeed a submodel and has at least k states can be done in polynomial time. As also bisimilarity can be checked in polynomial time, membership of NP is immediate. NP-hardness follows from Proposition 5.17 as the problem of deciding induced subgraph bisimilarity can be reduced to submodel bisimilarity. ■

As we are interested in the complexity of reasoning about the interaction of epistemic agents as it is modeled in (dynamic) epistemic logic, let us now see how the complexity of induced subgraph bisimulation changes when we make the assumption that models are partitional, i.e., that the relation is an equivalence relation, as it is frequently assumed in the AI or interactive epistemology literature. We will see that this assumption makes the problem significantly easier.

Proposition 5.20 *If for graphs $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$, E_1 and E_2 are reflexive, then induced subgraph bisimulation for G_1 and G_2 can be solved in linear time.*

Proof. In this proof, we will use the fact that $G_1 = (V_1, E_1) \underline{\leftrightarrow}_{total} G_2 = (V_2, E_2)$ if and only if it is the case that $V_1 = \varnothing$ iff $V_2 = \varnothing$. Let us prove this. From left to right, assume that $G_1 = (V_1, E_1) \underline{\leftrightarrow}_{total} G_2 = (V_2, E_2)$. Then since we have a total bisimulation, it must be the case that either $V_1 = V_2 = \varnothing$ or $V_1 \neq \varnothing \neq V_2$.

For the other direction, assume that $V_1 = \varnothing$ iff $V_2 = \varnothing$. Now, we show that in this case, $V_1 \times V_2$ is a total bisimulation between G_1 and G_2. If $V_1 = V_2 = \varnothing$, we are done. So, consider the case where $V_1 \neq \varnothing \neq V_2$. Let $(v_1, v_2) \in V_1 \times V_2$, and assume that $(v_1, v_1') \in E_1$ for some $v_1' \in V_1$. Since E_2 is reflexive, we know that there is some $v_2' \in V_2$ such that $(v_2, v_2') \in E_2$. Of course $(v_1', v_2') \in V_1 \times V_2$. The back condition is analogous. Since $V_1 \times V_2$ is total, we thus have $G_1 \underline{\leftrightarrow}_{total} G_2$. Hence, $G_1 = (V_1, E_1) \underline{\leftrightarrow}_{total} G_2 = (V_2, E_2)$ if and only if it is the case that $V_1 = \varnothing$ iff $V_2 = \varnothing$.

Therefore, for solving the induced subgraph bisimulation problem for input G_1 and G_2 with E_1 and E_2 being reflexive and $k \in \mathbb{N}$, all we need to do is to go through the input once and check whether $V_1 = \emptyset$ iff $V_2 = \emptyset$, and whether $|V_2| \geq k$. If the answer to both is *yes* then we know that $G_1 \underline{\leftrightarrow}_{total} G_2$ and since $|V_2| \geq k$, we answer *yes*, otherwise *no*. ■

Assuming the edge relation in a graph to be reflexive makes induced subgraph bisimulation a trivial problem because, unless its set of vertices is empty, every such graph is bisimilar to the graph $(\{v\}, \{(v, v)\})$. But for Kripke models, even for S5 models, this is of course not the case, as the bisimulation takes into account the valuation. Nevertheless, we will now show that also for single-agent S5 models, the problem of submodel bisimulation is significantly easier than in the case of arbitrary single-agent Kripke models. To be more precise, we will distinguish between two problems:

The first problem is *local* single-agent S5 submodel bisimulation. Here, we take as input two pointed S5 models. Then we ask whether there is a submodel of the second model that is bisimilar to the first one. Thus, the question is whether it is possible to restrict one of the models in such a way that there is a state in which the agent has exactly the same information as in the situation modeled in the other model. Note that in this problem we do not require the resulting model be of a certain size.

Decision Problem 5.21 (Local single-agent S5 submodel bisimulation)
Input: *A pointed S5 epistemic model* $(\mathcal{M}_1, w)$ *with* $\mathcal{M}_1 = (W_1, \sim_1, \mathsf{V}_1)$ *and* $w \in W_1$, *and an S5 epistemic model* $\mathcal{M}_2 = (W_2, \sim_2, \mathsf{V}_2)$.
Question: *Is there a submodel* $\mathcal{M}'_2 = (W'_2, \sim'_2, \mathsf{V}'_2)$ *of* $\mathcal{M}_2$ *such that* $(\mathcal{M}_1, w) \underline{\leftrightarrow} (\mathcal{M}'_2, w')$ *for some* $w' \in Dom(\mathcal{M}'_2)$? ◀

We will show that this problem is tractable. First we introduce some notation that we will use.

Notation Let $\mathcal{M} = (W, \sim, \mathsf{V})$ be a single-agent epistemic model. For the valuation function $\mathsf{V} : \textsc{prop} \to W$, we define $\hat{\mathsf{V}} : W \to 2^{\textsc{prop}}$, with $w \mapsto \{p \in \textsc{prop} \mid w \in \mathsf{V}(p)\}$. Abusing notation, for $X \subseteq W$ we sometimes write $\hat{\mathsf{V}}(X)$ to denote $\{\hat{\mathsf{V}}(w) \mid w \in X\}$. We let $W/\sim$ denote the set of all equivalence classes of W for the relation $\sim$. ◀

Proposition 5.22 *Local submodel bisimulation for single-agent pointed epistemic models is in* P.

Proof. Given the input of the problem, i.e., a pointed epistemic model $\mathcal{M}_1, w$ with $\mathcal{M}_1 = (W_1, \sim_1, \mathsf{V}_1)$, and $w \in W_1$ and an epistemic model $\mathcal{M}_2 = (W_2, \sim_2, \mathsf{V}_2)$, we run the following procedure.

1. For all $[w_2] \in W_2/\sim_2$ do the following:
   (a) Initialize the set $Z := \emptyset$.
   (b) for all $w' \in [w]$ do the following
      i. For all $w'_2 \in [w_2]$ check if it is the case that $\hat{\mathsf{V}}_1(w') = \hat{\mathsf{V}}_2(w'_2)$. If this is the case, set $Z := Z \cup \{(w', w'_2)\}$.
      ii. if there is no such w'_2, continue with 1 with the next element in $W_2/\sim_2$, otherwise we return Z and we stop.
2. In case we didn't stop at 1(b)ii, we can stop now, and return *no*.

This does not take more than $|\mathcal{M}_1| \cdot |\mathcal{M}_2|$ steps.

If the procedure has stopped at 2, there is no bisimulation with the required properties. To see this, note that if we stopped in 2, this means that there was no $[w_2] \in W_2/\sim_2$ such that for every state in $[w]$ there is one in $[w_2]$ in which exactly the same propositional letters are true. Thus, since we were looking for a bisimulation that is also defined for the state w, such a bisimulation cannot exist.

If the algorithm returned a relation Z, this is indeed a bisimulation between $\mathcal{M}_1$ and the submodel $\mathcal{M}'_2$ of $\mathcal{M}_2$ where $\mathcal{M}'_2 = (W'_2, \sim'_2, \mathsf{V}'_2)$, where

$$W'_2 = \{w_2 \in W_2 \mid \text{there is some } w_1 \in [w] \text{ such that } (w_1, w_2) \in Z\}$$

and $\sim'_2$ and V'_2 are the usual restrictions of $\sim_2$ and V_2 to W'_2. This follows from the following two facts: First, for all pairs in Z it holds that both states satisfy exactly the same proposition letters. Second, since Z is total both on $[w]$ and on W'_2 and all the states in $[w]$ are connected to each other by $\sim_1$ and all states in W'_2 are connected to each other by $\sim'_2$, both the*forth* and *back* conditions are satisfied. This concludes the proof. ■

The second problem we consider is *global* S5 submodel bisimulation, where the input are two models $\mathcal{M}_1$ and $\mathcal{M}_2$ and we ask whether there exists a submodel of $\mathcal{M}_2$ such that it is totally bisimilar to $\mathcal{M}_1$.

Decision Problem 5.23 (Global single-agent S5 submodel bisimulation)
Input: *Two S5 epistemic models* $\mathcal{M}_1 = (W_1, \sim_1, \mathsf{V}_1), \mathcal{M}_2 = (W_2, \sim_2, \mathsf{V}_2)$.
Question: *Is there a submodel* $\mathcal{M}'_2 = (W'_2, \sim'_2, \mathsf{V}'_2,)$ *of* $\mathcal{M}_2$ *such that* $\mathcal{M}_1 \underline{\leftrightarrow}_{total} \mathcal{M}'_2$*?* ◄

We can show that even though the above problem seems more complicated than Decision Problem 5.21, it can still be solved in polynomial time. The proof uses the fact that finding a maximum matching in a bipartite graph can be done in polynomial time (see e.g. Papadimitriou and Steiglitz (1982)).

Theorem 5.24 *Global submodel bisimulation for single-agent epistemic models is in* P.

Before we give the proof, we do some pre-processing.

Definition 5.25 Given a single-agent epistemic model $\mathcal{M} = (W, \sim, \mathsf{V})$, $\mathcal{M}^{min\_cells}$ denote a model obtained from $\mathcal{M}$ by the following procedure:

1. Initialize X with $X := W/\sim$.
2. Go through all the pairs in $X \times X$.
   (a) When you find $([w], [w'])$ with $[w] \neq [w']$ such that $\hat{\mathsf{V}}([w]) = \hat{\mathsf{V}}([w'])$, continue at 2 with $X := X - [w']$.
   (b) Otherwise, stop and return the model $\mathcal{M}^{min\_cells} := (\bigcup X, \sim', \mathsf{V}')$, where $\sim'$ and V' are the usual restrictions of $\sim$ and V to $\bigcup X$. ◀

Fact 5.26 *With input* $\mathcal{M} = (W, \sim, \mathsf{V})$*, the procedure in Definition 5.25 runs in time polynomial in* $|\mathcal{M}|$.

Proof. Follows from the fact that the cardinality of $W/\sim$ is bounded by $|W|$; we only enter step 2 at most $|W|$ times, and each time do at most $|W|^2$ comparisons. ■

Fact 5.27 *The answer to total submodel bisimulation for single-agent epistemic models (Decision Problem 5.23) with input* $\mathcal{M}_1 = (W_1, \sim_1, \mathsf{V}_1), \mathcal{M}_2 = (W_2, \sim_2, \mathsf{V}_2)$ *is* ***yes*** *iff it is with input* ${\mathcal{M}_1}^{min\_cells} = (W_1, \sim_1, \mathsf{V}_1), \mathcal{M}_2 = (W_2, \sim_2, \mathsf{V}_2)$.

Proof. From left to right, we just need to restrict the bisimulation to the states of ${\mathcal{M}_1}^{min\_cells}$. For the other direction, we start with the given bisimulation and then extend it as follows. The states in a cell $[w']$ which was removed during the construction of ${\mathcal{M}_1}^{min\_cells}$ can be mapped to the ones of a cell $[w]$ in ${\mathcal{M}_1}^{min\_cells}$ with the same valuation. ■

We can now prove Theorem 5.24.

Proof. By Fact 5.26 and Fact 5.27, transforming $\mathcal{M}_1$ into ${\mathcal{M}_1}^{min\_cells}$ can be done in polynomial time. Thus, without loss of generality, we can assume that $\mathcal{M}_1$ is already of the right shape; i.e., $\mathcal{M}_1 = {\mathcal{M}_1}^{min\_cells}$. Given the two models as input, we construct a bipartite graph $G = ((W_1/\sim_1, W_2/\sim_2), E)$ where E is defined as follows.

$$([w_1], [w_2]) \in E \text{ iff } \hat{\mathsf{V}}_1([w_1]) \subseteq \hat{\mathsf{V}}_2([w_2]).$$

Claim 5.28 *The following are equivalent.*

(a) There is a submodel $\mathcal{M}'_2$ of $\mathcal{M}_2$ such that $\mathcal{M}_1 \underline{\leftrightarrow}_{total} \mathcal{M}'_2$

(b) G has a matching of size $|W_1/\sim_1|$.

Proof. Assume that there is a submodel $\mathcal{M}'_2 = (W'_2, \sim'_2, V'_2)$ of $\mathcal{M}_2$ such that $\mathcal{M}_1 \underline{\leftrightarrow}_{total} \mathcal{M}'_2$. Let Z be such a total bisimulation.

Note that since we assumed that $\mathcal{M}_1 = \mathcal{M}^{min\_cells}$ the following holds:

1. For all $([w_1],[w_2]) \in W_1/\sim_1 \times W_2/\sim_2$ it is the case that whenever $Z \cap ([w_1] \times [w_2]) \neq \emptyset$, then for all $[w'_1] \in W_1/\sim_1$ such that $[w'_1] \neq [w_1]$, $Z \cap ([w'_1] \times [w_2]) = \emptyset$.

Thus, the members of different equivalence classes in $W_1/\sim_1$ are mapped by Z to into different equivalence classes of $W_2/\sim_2$.

Now, we construct $\dot{E} \subseteq E$ as follows.

$$([w_1],[w_2]) \in \dot{E} \text{ iff } ([w_1],[w_2]) \in E \text{ and } ([w_1] \times [w_2]) \cap Z \neq \emptyset.$$

Then $|\dot{E}| \geq |W_1/\sim_1|$ because of the definitions E and $\dot{E}$ and the fact that Z is a bisimulation that is total on W_1. Now, if $|\dot{E}| = |W_1/\sim_1|$ then we are done since by definition of $\dot{E}$, for each $[w_1] \in W_1/\sim_1$ there is some $[w_2] \in W_2/\sim_2$ such that $([w_1],[w_2]) \in \dot{E}$. Then it follows from 1, that $\dot{E}$ is indeed a matching.

If $|\dot{E} > |W_1/\sim_1|$ then we can transform $\dot{E}$ into a matching E' of size $W_1/\sim_1|$: For each $[w_1] \in W_1/\sim_1$, we pick *one* $[w_2] \in W_2/\sim_2$ such that $([w_1],[w_2]) \in \dot{E}$ and put it into E' (note that such a $[w_2]$ always exists because by definition of $\dot{E}$, for each $[w_1] \in W_1/\sim_1$ there is some $[w_2] \in W_2/\sim_2$ such that $([w_1],[w_2]) \in \dot{E}$; moreover because of 1 all the $[w_2] \in W_2/\sim_2$ that we pick will be different). Then the resulting $E' \subseteq \dot{E} \subseteq E \subseteq (W_1/\sim_1 \times W_2/\sim_2)$ is a matching of G of size $|W_1/\sim_1|$. Thus, we have shown that if there is a submodel $\mathcal{M}'_2$ of $\mathcal{M}_2$ such that $\mathcal{M}_1 \underline{\leftrightarrow}_{total} \mathcal{M}'_2$ then G has a matching of size $|W_1/\sim_1|$.

For the other direction, assume that G has a matching $E' \subseteq E$ with $|E'| = |W_1/\sim_1|$. Then, recalling the definition of E, it follows that for all $[w] \in W_1/\sim$ there is some $[w'] \in W_2/\sim_2$ such that $([w],[w']) \in E'$ and thus $\hat{V}_1([w]) \subseteq \hat{V}_2([w'])$.

Let us define the following submodel $\mathcal{M}'_2$ of $\mathcal{M}_2$. $\mathcal{M}'_2 = (W'_2, \sim'_2, V'_2)$, where

$$W'_2 = \{w_2 \in W_2 \mid \text{there is a } w \in W_1 \text{ with } \hat{V}_1(w) = \hat{V}_2(w_2) \text{ and } ([w],[w_2]) \in E'\}$$

and $\sim'_2$ and V'_2 are the usual restrictions of $\sim_2$ and V_2 to W'_2.

Now, we define a relation $Z \subseteq W_1 \times W'_2$, which we then show to be a total bisimulation between $\mathcal{M}_1$ and $\mathcal{M}'_2$

$$(w_1, w_2) \in Z \text{ iff } \hat{V}(w_1) = \hat{V}_2(w_2) \text{ and } ([w_1],[w_2]) \in E'.$$

Next, let us show that Z is indeed a bisimulation.

Let $(w_1, w_2) \in Z$. Then, by definition of Z, for every propositional letter p, $w_1 \in \mathsf{V}_1(p)$ iff $w_2 \in \mathsf{V}_2(p)$. Next, we check the *forth* condition. Let $w_1 \sim_1 w_1'$ for some $w_1' \in W_1$. Then since $(w_1, w_2) \in Z$, and thus $([w_1],[w_2]) \in E'$, there has to be some $w_2' \in [w_2]$ such that $\hat{\mathsf{V}}_2(w_2') = \hat{\mathsf{V}}_1(w_1')$. Then since $[w_1'] = [w_1]$ and $[w_2'] = [w_2]$, $([w_1'],[w_2']) \in E'$. Then $w_2' \in W_2'$, and $(w_1', w_2') \in Z$.

For the *back* condition, let $w_2 \sim_2 w_2'$, for some $w_2' \in W_2'$. Then by definition of W_2', there is some $w \in W_1$ such that $\hat{\mathsf{V}}_1(w) = \hat{\mathsf{V}}_2(w_2')$ and $([w],[w_2']) \in E'$. Thus, it follows that $(w, w_2') \in Z$. Now, we still have to show that $w_1 \sim_1 w$. As the following hold: $([w],[w_2']) \in E'$,$[w_2] = [w_2']$, $([w],[w_2]) \in E'$ (because $(w_1, w_2) \in Z$) and E' is a matching, it follows that $[w] = [w_1]$. Thus, $w_1 \sim_1 w$.

Hence, we conclude that Z is a bisimulation. It remains to show that Z is indeed total.

Let $w_1 \in W_1$. Since E' is a matching of size $W_1/\sim_1$, there is some $[w_2] \in W_2/\sim_2$ such that $([w_1],[w_2]) \in E'$. Thus, there is some $w_2' \in [w_2]$ such that $\hat{\mathsf{V}}_1(w_1) = \hat{\mathsf{V}}_2(w_2')$. This means that $w_2' \in W_2'$ and $(w_1, w_2') \in Z$. So Z is total on W_1.

Let $w_2 \in W_2'$. By definition of W_2', there is some $w \in W_1$ such that $\hat{\mathsf{V}}_1(w) = \hat{\mathsf{V}}_2(w_2)$ and $([w],[w_2]) \in E'$. Thus, by definition of $Z, (w, w_2) \in Z$. Therefore, Z is indeed a total bisimulation between $\mathcal{M}_1$ and $\mathcal{M}_2'$. This concludes the proof of Claim 5.28. ■

Hence, given two models, we can transform the first one using the polynomial procedure of Definition 5.25 and then we construct the graph G, which can be done in polynomial time as well. Finally, we use a polynomial algorithm to check if G has a matching of size ${M_1}^{min\_cells}$. If the answer is yes, we return *yes*, otherwise *no*. This concludes the proof of Theorem 5.24. ■

Now, the question arises whether the above results also hold for the multi-agent case.

Decision Problem 5.29 (Global multi-agent S5 submodel bisimulation)
Input: *Two epistemic models* $\mathcal{M}_1 = (W_1, (\sim_{1i})_{i \in \mathtt{N}}, \mathsf{V}_1), \mathcal{M}_2 = (W_2, (\sim_{2i})_{i \in \mathtt{N}}, \mathsf{V}_2)$, *for* $\mathtt{N}$ *being a finite set (of agents), and* $k \in \mathbb{N}$.

Question: *Is there a submodel* $\mathcal{M}_2' = (W_2', (\sim_{2i}')_{i \in \mathtt{N}}, \mathsf{V}_2')$ *of* $\mathcal{M}_2$ *such that* $\mathcal{M}_1 \underline{\leftrightarrow}_{total} \mathcal{M}_2'$? ◄

Open Problem 5.30 Is *global multi-agent S5 submodel bisimulation* NP-hard? ◄

We expect the answer to this question to be positive, as for S5, there seems to be a complexity jump between the single-agent case and the two-agent case: In case of the satisfiability problem of the logic, the one-agent logic is NP-complete, whereas as soon as we have at least two agents, we get PSPACE completeness. Similarly, in Section 5.2.1, we showed in Proposition 5.7 that also for bisimilarity, there seems to be a complexity jump for S5 models when a second agent is added: the problem becomes P-hard and thus as hard as the problem for arbitrary Kripke models.

The idea behind these results is that arbitrary accessibility relations can be simulated by a concatenation of two equivalence relations. However, these techniques, as they have been used e.g. by Halpern and Moses (1992), do not seem to work for transforming models into S5-models for two agents such that the existence of submodels bisimilar to some model is preserved. The problem is caused by the fact that the resulting model has to be reflexive, in which case several states could be collapsed whereas they could not before the transformation. Thus, a coding of the existence of a successor and the existence of reflexive loops in the original model would be required to take care of this issue[3].

Let us summarize our complexity results for problems related to deciding whether it is possible to restrict an agent's information structure so that after he will have similar information as another agent in some other situation. We started by showing that induced subgraph bisimulation is intractable (NP-complete). Using this, we could show that the same holds for submodel bisimulation of arbitrary Kripke models.

For partition-based graphs (with the edge relations being equivalence relations) however, we showed that the problem of induced subgraph bisimilarity is very easy: it is solvable in linear time if we are looking for a subgraph of a certain size with a total bisimulation. Deciding whether there is any subgraph with a total bisimulation is trivial for graphs where the edge relation is an equivalence relation, as then all such non-empty graphs are bisimilar. In fact, this already holds if the edge relation is reflexive.

Extending these results for S5 Kripke models, we could show that submodel bisimulation for single-agent models is not as trivial as for graphs, but still in P. For multi-agent S5 models, the problem remains open. We conjecture it to be polynomially equivalent to the problem for Kripke models in general. The technical challenge in showing this lies in the simulation of arbitrary accessibility relations by a combination of different equivalence relations. The idea would again be to use a method similar to that used e.g. by Halpern and Moses (1992) and replace every edge by a composition of two equivalence relations. This technique can however not be applied in its standard way as some issues arise due to the fact that the resulting model has to be reflexive while in the original arbitrary Kripke structure this does not need to be the case. This means that the existence of reflexive loops in the original model would somehow have to

[3] Note that the situation in Proposition 5.7 is different as there we started with models that were irreflexive and thus we did not need to take care of coding the information about loops and could thus apply the standard technique.

be coded using a propositional letter so as not to loose the information during the transformation.

In dynamic systems with diverse agents, an interesting question is whether it is possible to give some information to one agent such that afterward she knows at least as much as some other agent. This is captured by an asymmetric notion, that of simulation. With this difference, the question can be raised of the effect on tractability and intractability of requiring simulation versus requiring bisimulation. With this motivation, we would like to explore the problem of induced subgraph simulation.

Decision Problem 5.31 (Induced subgraph simulation)
Input: *Two finite graphs* $G_1 = (V_1, E_1)$, $G_2 = (V_2, E_2)$, $k \in \mathbb{N}$.

Question: *Is there an induced subgraph of* G_2 *with at least k vertices that is simulated by* G_1*, i.e., is there some* $V' \subseteq V_2$ *with* $|V'| \geq k$ *and* $(V', E_2 \cap (V' \times V')) \sqsubseteq_{total} G_1$*?* ◄

Proposition 5.32 *Induced subgraph simulation is* NP*-complete.*

Proof. Showing that the problem is in NP is straightforward. Hardness is shown by reduction from Independent Set. First of all, let $I_k = (V_{I_k}, E_{I_k} = \varnothing)$ with $|V_{I_k}| = k$ denote a graph with k vertices and no edges. Given the input of Independent Set, i.e., a graph $G = (V, E)$ and some $k \in \mathbb{N}$ we transform it into (I_k, G), k, as input for Induced Subgraph Simulation.

Now, we claim that G has an independent set of size at least k iff there is some $V' \subseteq V$ with $|V'| \geq k$ and $(V', E \cap (V' \times V')) \sqsubseteq_{total} I_k$.

From left to right, assume that there is some $S \subseteq V$ with $|S| = k$, and for all $v, v' \in S$, $(v, v') \notin E$. Now, any bijection between S and V_{I_k} is a total simulation (and in fact an isomorphism) between $G' = (S, E \cap (S \times S))$ and I_k, since $E \cap (S \times S) = \varnothing$ and $|S| = |V_{I_k}|$.

For the other direction, assume that there is some $V' \subseteq V$ with $|V'| = k$ such that for $G' = (V', E' = E \cap (V' \times V'))$ we have that $G' \sqsubseteq_{total} I_k$. Thus, there is some total simulation Z between G' and I_k. Now, we claim that V' is an independent set of G of size k. Let $v, v' \in V'$. Suppose that $(v, v') \in E$. Then since G' is an induced subgraph, we also have that $(v, v') \in E'$. Since Z is a total simulation, there has to be some $w \in I_k$ with $(v, w) \in Z$ and some w' with $(w, w') \in E_{I_k}$ and $(v', w') \in Z$. But this is a contradiction with $E_{I_k} = \varnothing$. Thus, V' is an independent set of size k of G. The reduction can clearly be computed in polynomial time. This concludes the proof. ■

In De Nardo et al. (2009), it has been shown that given two graphs it is also NP-complete to decide if there is a subgraph (not necessarily an induced one) of one such that it is simulation equivalent to the other graph. Here, we show that this also holds if the subgraph is required to be an induced subgraph.

Decision Problem 5.33 (Induced subgraph simulation equivalence)
Input: *Two finite graphs* $G_1 = (V_1, E_1)$, $G_2 = (V_2, E_2)$, $k \in \mathbb{N}$.

Question: *Is there an induced subgraph of* G_2 *with at least* k *vertices that is similar to* G_1*, i.e., is there some* $V' \subseteq V_2$ *with* $|V'| \geq k$ *and* $(V', E_2 \cap (V' \times V')) \sqsubseteq_{total} G_1$ *and* $G_1 \sqsubseteq_{total} (V', E_2 \cap (V' \times V'))$*?* ◀

Proposition 5.34 *Induced subgraph simulation equivalence is* NP*-complete.*

Proof. For showing that the problem is in NP, note that we can use a simulation equivalence algorithm as provided in Henzinger et al. (1995). Hardness can again be shown by reduction from Independent Set. Given the input for Independent Set, i.e., a graph $G = (V, E)$ and some $k \in \mathbb{N}$, we transform it into two graphs $I_k = (V_{I_k} = \{v_1, \dots v_k\}, E_{I_k} = \emptyset)$ and G, and we keep the $k \in \mathbb{N}$. This can be done in polynomial time.

Now, we claim that G has an independent set of size k iff there is an induced subgraph of G with k vertices that is similar to I_k. From left to right assume that G has such an independent set S with $S \subseteq V$, $|S| = k$ and $E \cap S \times S = \emptyset$. Then $(S, \emptyset)$ is isomorphic to I_k since both have k vertices and no edges. Thus, they are also simulation equivalent.

For the other direction, assume that there is an induced subgraph $G' = (V', E')$ with $V' \subseteq V$, $|V'| = k$ and $E' = (V' \times V') \cap E$ such that G' is simulation equivalent to I_k. Suppose that there are $v, v' \in V'$ such that $(v, v') \in E$. Since G' is an induced subgraph, it must be the case that $(v, v') \in E'$, but since I_k simulates G', this leads to a contradiction since I_k does not have any edges. This concludes the proof. ■

As a corollary of the two previous propositions we get that for arbitrary Kripke models both submodel simulation and submodel simulation equivalence are NP-hard. An NP upper bound follows from the fact that given a relation between a model and a submodel of some other model, it can be checked in polynomial time if this relation is indeed a simulation.

Corollary 5.35 *Deciding submodel simulation and submodel equivalence for Kripke structures is* NP*-complete.* ◀

For single-agent S5, we can use the methods as used in the proof of Theorem 5.24 in order to obtain a polynomial procedure for the single-agent case.

Proposition 5.36 *Deciding submodel simulation and submodel equivalence single-agent S5 models is in* P.

Proof. We use the procedure of the proof of Theorem 5.24. This also works for simulation and simulation equivalence because of the totality constraint and the fact that as we deal with S5 models, we only need to take care of the different valuations occurring in the equivalence classes. ■

Let us now summarize our complexity analysis of tasks that involve whether in one situation an agent knows at least as much as another agent in a possibly different situation. We have shown that we can extend graph theoretical complexity results about subgraph simulation equivalence to the case where the subgraph is required to be an induced subgraph. Via this technical result, we can then transfer the complexity bounds also for the problem of submodel simulation (equivalence) of Kripke models, which with an epistemic interpretation of the accessibility relation is the following problem: *decide whether it is possible to give information to one agent such that as a result he knows as least as much as some other agent.* In case of partition-based models (S5), for a single agent this problem can be solved in polynomial time analogously to how we have done it for submodel bisimulation. For the multi-agent case, the problem remains open, however. As for submodel bisimulation of multi-agent S5 model, the technical issue that would have to be solved for showing NP-hardness is caused by the reflexivity of the underlying relations.

5.3 Conclusions and Further Questions

We will now summarize the main results of this chapter and then give conclusions and further questions.

5.3.1 Summary

In this chapter, we have identified concrete epistemic tasks related to the comparison and manipulation of information states of agents in possibly different situations. Interestingly, our complexity analysis shows that such tasks and decision problems live on both sides of the border between tractability and intractability. We now summarize our results for the different kinds of tasks we investigated.

Information similarity. Our results for the complexity of deciding whether information structures are similar can be found in Table 5.1.

Problem	**Tractable?**	**Comments**
Kripke model isomorphism	unknown	GI-complete
Epistemic model bisimilarity	Yes	P-complete in the multi-agent case

Table 5.1: Complexity results for deciding information similarity.

If we take isomorphism as similarity notion, then in general (without any particular assumptions on the Kripke structures representing agents' information) it is open whether checking if two information structures are similar is tractable. This follows from the fact that checking if two Kripke models are isomorphic is as hard as the graph isomorphism problem which is neither known to be in P nor known to be NP-hard. Thus, we can say that given the current knowledge, for isomorphism deciding if two information structures are similar can be located on the border between tractability and intractability. We did not investigate the isomorphism problem for S5 but conjecture it to become as hard as Kripke model isomorphism (GI-complete) as soon as we have at least two agents.

Taking bisimilarity as similarity notion, deciding if two structures are similar is among the hardest problems known to be tractable. If the models are based on partitions (S5), the problem is very easy in the single-agent case but also becomes P-hard in the multi-agent case.

Information symmetry. Table 5.2 summarizes the results of our complexity analysis of tasks concerned with deciding whether the information of agents is symmetric, where symmetry can be understood in different ways.

Problem	**Tractable?**	**Comments**
Common knowledge of a fact	Yes	solvable using a reachability algorithm
Horizon bisimilarity (Kripke models)	Yes	P-complete for arbitrary models, even for horizons at the same point in the model
Flipped horizon bisimilarity (Kripke models)	Yes	P-complete, even for horizons at the same point in the model
Horizon bisimilarity (S5-models)	Yes	trivial for horizons at the same point in a model
Flipped horizon bisimilarity (S5-models)	Yes	problem does not get easier for horizons at the same point in a model

Table 5.2: Complexity results for deciding information symmetry.

We started our investigation of information symmetry with the symmetry of two agents' knowledge about a given fact being true. This kind of symmetry arises if the fact is common knowledge among the two agents. Given an information structure, deciding if this is the case can be done using a reachability algorithm that checks for every state at which the fact is not true whether there is a path to it (via the union of the two relations of the agents) from the current state. This is the case if and only if the fact is not common knowledge by the two agents.

We then introduced the notion of *(epistemic) horizon*, which represents the submodel that is relevant for an agent at a given situation (i.e., at a given point in the model). The horizon of an agent in a situation is the submodel that is generated by the set of worlds the agent considers possible in that situation. When considering the epistemic reasoning of agents, our notion of horizon plays a crucial role as an agent's horizon contains exactly the possible worlds which the agent might take into consideration during his reasoning. We have shown that in general deciding if the horizons of two agents are bisimilar is exactly as hard as deciding bisimilarity of Kripke models. Without assuming reflexivity of the accessibility relation, deciding about the similarity of two agents' horizons does not get easier in the special case in which we compare horizons at the very same point in a model. As soon as the accessibility relations of the two agents under consideration are reflexive however, the problem becomes completely trivial if we compare horizons at the same point in the model as they are always identical. Thus, if we take information structures to be arbitrary Kripke models, then in general comparing horizons of agents in one given situation is as hard as comparing information structures in general. For S5 models however, the situation is slightly different as horizon bisimilarity becomes trivial for horizons taken at the same point in a model.

For our investigation of information symmetry, we have introduced the notion of flipped bisimilarity, which captures the similarity of two models after swapping the information of two agents. For Kripke structures, in general the complexity of deciding flipped bisimilarity is just as for bisimilarity, even though for the special case in which two pointed structures are identical deciding bisimilarity is trivial but flipped bisimilarity can be as hard as it is for arbitrary pointed Kripke structures.

Our results for horizon comparison for arbitrary Kripke models show that both flipped bisimilarity and regular bisimilarity are P-complete, even if we take the horizon at the very same situation. Thus, comparing different agents' perspectives on the very same situation is as hard as comparing structures in general. Under the assumption of partition-based information structures (S5) however, we observed a significant difference between bisimilarity and flipped bisimilarity of horizons. While bisimilarity of horizons of different agents becomes trivial if they are taken at the very same situation (i.e., at the same point in the model), flipped bisimilarity stays as hard as it is for multi-agent S5 models in general.

Let us briefly summarize the technical facts that explain our complexity results as given in Table 5.2.

- Problems about information symmetry which can be solved by checking if certain states are reachable by (combinations of) agents' accessibility relations are relatively easy as they boil down to solving the Reachability problem which is NL-complete.
- Problems involving bisimilarity of arbitrary models are among the hardest tractable problems and thus believed to be slightly easier than problems involving isomorphism of Kripke models as for isomorphism no polynomial algorithms are know.
- In the single-agent case, assuming S5 relations makes bisimilarity easier because checking for bisimilarity boils down to just comparing the propositional valuations of information cells.
- While flipped bisimilarity does not seem to be more complex than regular bisimilarity, the fact that flipped bisimilarity is in general not reflexive has the effect of making it harder than bisimilarity in the special case where we ask if a pointed model is flipped bisimilar to itself.

Information manipulation. Apart from the rather static problems about the comparison of information structures we also investigated the complexity of tasks related to more dynamic aspects of information. In many interactive processes, the information of agents changes through time because agents can make observations or receive new information from other agents. Then an interesting question that arises is whether given an information state of an agent it is possible that through incoming information, the agent's information structure can change such that in the end the agent has similar information to some other agent.

Table 5.3 summarizes the results of our complexity analysis of tasks concerned with the manipulation of information structures.

For determining the complexity of deciding whether it is possible to restrict an information structure in some way such that it becomes similar to some other structure, we started by investigating the NP-complete graph theoretical problem *subgraph bisimulation*. The problem is to decide whether one of two given graphs has a subgraph which is bisimilar to the other graph. We showed that it remains NP-complete if we require the subgraph to be an *induced* subgraph. This technical result then allowed us to show that for Kripke models, it is also NP-complete to decide if one given model has a submodel which is bisimilar to another given model. We then showed that this problem does indeed get easier if we have S5 structures with one agent only: we gave a polynomial procedure that uses the fact that computing whether there is

Problem	**Tractable?**	**Comments**
Kripke submodel bisimulation	No	NP-complete. Reduction from Independent Set
Single agent S5 submodel bisimulation	Yes	Local version easier; in general an algorithm for finding matchings in bipartite graphs can be used
Multi-agent S5 submodel bisimulation	Unknown	Conjectured to be NP-complete
Kripke submodel simulation (equivalence)	No	NP-complete. Reduction from Independent Set
Single agent S5 submodel simulation (equivalence)	Yes	Similar polynomial procedure as for single-agent S5 submodel bisimulation
Multi-agent S5 submodel simulation (equivalence)	Unknown	Same technical issues as for S5 submodel bisimulation

Table 5.3: Complexity results for tasks about information manipulation.

a matching of a certain size in a bipartite graph can be done in polynomial time. This shows that deciding if an agent's information can be restricted in a certain way is easier under the assumption of S5 information structures. It remains open to show whether the problem also becomes intractable for S5 as soon as we have more than one agent. The technical issue which needs to be resolved here is to determine whether an arbitrary accessibility relation can be simulated by the composition of two equivalence relations in such a way that the existence of a submodel bisimilar to some other model is preserved. While it is relatively straightforward to make sure that in the model that results from the transformation the accessibility relations are symmetric and transitive, the requirement of reflexivity seems to cause some problems.

Instead of asking whether it is possible to give some information to an agent such that the resulting information structure is similar to some other structure, in many situations it might be sufficient to know if it is possible to manipulate the information of an agent such that he will know at least as much as some other agent. In more general terms, this leads us to the task of deciding whether it is possible to restrict some structure such that it becomes at least as refined as some other structure. Similar to the case of submodel bisimulation, we started by investigating the problem of induced subgraph simulation, which we showed to be NP-complete by reduction from *Independent Set*. Using this, we could then show that submodel simulation is NP-complete for Kripke models.

Under the assumption of S5 models, we can adapt the polynomial procedure that we had for single-agent S5 local submodel bisimulation for solving the analogous problem for simulation. This means that with S5 models, it is tractable to decide if we can restrict an information structure for one agent such that it becomes at least as refined as that of another agent in a given situation. We get an analogous result for simulation equivalence, a weaker notion of similarity than bisimulation. Whether on S5 structures these problems become intractable as soon as we have models with at least two agents is open, and depends on the same technical issues as this problem for submodel bisimulation.

Let us briefly summarize the technical facts that explain our complexity results as listed in Table 5.3.

- For single-agent S5 models, submodel bisimilarity and simulation equivalence turned out to be solvable in polynomial time. We used the fact that a model has a submodel bisimilar to some other model if and only if the bipartite graph that consists of the equivalence classes of both models and in which edges connect information cells of the two models if and only if the different valuations of the first occur also in the second.
- In general, submodel bisimilarity of Kripke models is NP-complete, as a graph having an induced subgraph bisimilar to the graph of k isolated points is equivalent to the graph having an independent set of size k.
- Whether submodel bisimilarity is NP-complete for S5 models with more than one agent depends on how arbitrary Kripke structures can be simulated using the combination of two equivalence classes in such a way that the existence of submodels bisimilar to another model is preserved.

5.3.2 Conclusions

From the above results, we can conclude the following for the three classes of tasks that we have analyzed.

Information similarity

- If information of agents is modeled by simple relational structures without any particular assumptions, then deciding if the information of agents in two different multi-agent situations is similar will in general be somewhere in between tractable but hard and the border to intractability.

- Under the assumption of S5 properties of the information structures, the complexity jump from easy to P-hard happens with the introduction of a second agent.

Information symmetry

- All problems we encountered are tractable, but nevertheless we were able to identify significant differences in their complexity.
  - Comparing the perspectives of agents in the very same situation becomes trivial as soon as we check for bisimilarity and the agents' accessibility relations are reflexive.
  - For checking if the agents have similar information *about each other* (captured by flipped bisimilarity of horizons) however, neither the assumption of reflexivity of the agents' relations nor considering horizons at the very same point in the model make the problem easier.
  - Thus, deciding if agents have similar information *about each other* can in certain cases be harder than deciding if agents have similar information.

Information manipulation

- For the problems we identified for deciding whether an information structure can be restricted in a way such that it will be in a certain relation to another model, we get the same pattern of complexity results for simulation, simulation equivalence and bisimulation.
  - Deciding whether a model can be restricted such that it is in one of those three relations to another model is tractable for single-agent S5 models and intractable in general.
  - Whether for S5 models the jump from being tractable to being intractable happens with the introduction of a second agent depends on whether we can simulate arbitrary relations by a combination of two equivalence relations while preserving the existence of submodels that are in a certain relationship to another model.

Comparing the three classes of tasks (about information similarity, symmetry and manipulation), information similarity is the easiest one in general if we stick to bisimulation as our notion of similarity. For information symmetry, all the problems we identified are tractable, with some special cases even being trivial, such as for reflexive models similarity of horizons at the same situation. Deciding if two agents have the same information about each other however does not become trivial unless the two agents are equal. For deciding whether it is possible to manipulate agents' information in a certain way, we considered problems of a wide variety of complexities ranging from very easy to NP-complete. Deciding if it is possible to restrict an information structure such that it becomes similar to or at least as refined as another is easiest if we only have one agent and assume S5 models. For arbitrary Kripke structures the problem is NP-complete. For multi-agent S5 models we conjecture it to be NP-complete as well. Locating the tractability border in epistemic tasks on modal logic frameworks, we conclude that for the static tasks concerning similarity and symmetry, most problems are tractable, whereas for the dynamic tasks involving the manipulation of information intractable tasks arise when we have multiple agents. In general, for S5 models, complexity jumps for various tasks seem to occur when a second agent is introduced.

Let us now come back to our research question.

> **Research Question 3** *Which parameters can make interaction difficult?*
>
> - *How does the complexity of an interactive situation change when more participants enter the interaction or when we drop some simplifying assumptions on the participants themselves?*

In the context of concrete tasks in reasoning about epistemic agents, we can give the following answers.

1. The complexity of comparing the information of diverse agents crucially depends on the notion of similarity used.

2. Under standard assumptions about knowledge (veridicality and full introspection) intractable tasks can become very easy

3. Moreover, under these assumptions, for various tasks a complexity jump occurs with the introduction of a second agent.

4. Without any assumptions on information structures reasoning about a single agent seems to be already as hard as reasoning about multi-agent situations.

5.3.3 Further Questions

The work in this chapter gives rise to some interesting questions for further investigation. Let us start with some technical open problems.

Does submodel bisimulation for S5 become intractable with two agents?

It remains open to show whether the problem also becomes intractable for S5 as soon as we have more than one agent. The technical issue which needs to be resolved here is to determine whether an arbitrary accessibility relation can be simulated by the composition of two equivalence relations in such a way that the existence of a submodel bisimilar to some other model is preserved[4].

While it is relatively straightforward to make sure that in the model that results from the transformation the accessibility relations are symmetric and transitive, the requirement of reflexivity seems to cause some problems.

Does submodel simulation (equivalence) for S5 become intractable with two agents? Whether on S5 structures, the problems of submodel simulation and submodel simulation equivalence become intractable as soon as we have models with at least two agents depends on the same technical issues as the problem for submodel bisimulation.

A more general problem that came up in our analysis is the following.

Is for S5 models simulation (equivalence) at least as hard as bisimulation? We did not investigate simulation and simulation equivalence of information structures. Here, an interesting general question arises as to whether also for epistemic (S5) models it holds that in general simulation (equivalence) is at least as hard as bisimulation as this holds for Kripke structures (Kučera and Mayr 2002).

Linking up to real epistemic reasoning. In addition to the technical questions above, our results also call for an empirical investigation of the tasks we identified in order to clarify the correspondence between our results and the cognitive difficulties involved in epistemic reasoning. For this, we note that the formal concepts that we used in the decision problems (e.g. bisimilarity) were mostly motivated by the fact that they come up naturally in the context of modal logics.

However, for being able to draw conclusions about the complexity that real agents face in epistemic reasoning, it needs to be investigated which are cognitively adequate notions of similarity. One possibility would be to work out the connection between the similarity notions that we considered and those underlying analogical reasoning in interactive situations (cf. Besold et al. (2011)).

[4] We stress that here we are concerned with *submodels*, i.e., in general these are *not* generated submodels.

Summing up our investigation so far, we have moved from a high-level perspective on the strategic abilities of agents to an analysis of concrete tasks about information structures which represent the uncertainties that individuals have about the situation they are in. Since our work is originally motivated by the need of a formal theory of real interaction, this leads to the next and last step of the analysis in this dissertation, and thus back to interaction in real life.

We will address the question of whether a complexity theoretical analysis as we have provided so far can actually allow us to draw some conclusions about real interactions, in particular about the complexity that real agents face in interactive situations. We will investigate this in the setting of a particular recreational game. Such a setting has the advantage that it is clearly defined and controlled by the rules of the game.

In addition to the setting, we also have to decide which kind of problems we would like to consider in the chosen setting. This choice should be motivated by the aim of establishing a connection between the formal study of the problems and the tasks that real agents face in interaction. Thus, for being able to draw conclusions about the complexity real agents face, we should choose tasks/problems that players will actually face when playing the game. In particular, this means that we should not focus on problems which are only involved in sophisticated strategic reasoning as this would first require a study of strategic reasoning of human reasoners in the chosen game. Appropriate tasks to be analyzed seem to be those that players cannot avoid during game play. An example of this is the task to perform a legal move.

As for the particular game to be investigated, we will choose the class of inductive inference games, i.e., games in which one player constructs a secret rule and the others have to inductively infer what might be the rule based on feedback they get for the moves they make. For our purpose, inductive inference games have the advantage of covering a wide range of complexity, which can be easily adapted as the complexity depends on the chosen secret rule.

Chapter 6

The Complexity of Playing Eleusis

Throughout this dissertation, our complexity analysis of interactive processes has moved from the study of formal logical systems to the complexity analysis of actual tasks relevant for the interaction of agents. The next and last step in our work is now to determine what a complexity theoretical analysis of formal frameworks for interaction can actually tell us about the difficulties that real agents face in interactive situations.

> **Research Question 4** *Finally, to what extent can we use a formal analysis of interactive processes to draw conclusions about the complexity of actual interaction?*
>
> - *Are there concrete examples of interactions in which participants actually encounter very high complexities which make it impossible for them to act?*

In this chapter, we present a concrete case study in which we investigate the computational complexity involved in actually playing a particular recreational game. It is important to mention that as opposed to e.g. Chapter 4 we focus on the complexity which is already involved when just performing legal moves in the game, and we are thus not so much concerned with strategic considerations such as deciding whether a winning strategy exists or computing such a winning strategy. The advantage of this is that the complexity involved in performing a legal move in a sense cannot be avoided by the players. Hence, a study of those tasks will allow us to draw conclusions for the complexity of actually playing the game.

One might expect that for concrete recreational games, the tasks of determining legal moves should be tractable as this task is at the very heart of actually playing a game. However, we will show that in the game we consider here, which belongs to the class of inductive inference games, this is not the case and even without strategic considerations, players can face intractable and even undecidable problems during the play.

6.1 Eleusis: an inductive inference game

In this chapter, we want to put forward the complexity theoretical analysis of a particular class of games, called *inductive inference games*, which are not only interesting from a game theoretical and computational perspective but also from a philosophical and learning theoretical point of view as they provide a simulation of scientific discovery. Thus, this chapter also links up with Chapter 4 where we investigated Sabotage Games, which can also be seen as a model of learning theoretical interaction. The general idea of inductive inference games is that players try to infer a general rule from the feedback they get to their moves. One designated player has to come up with a rule about which moves of the other players are accepted and which are rejected. The goal of the other players is then to discover the rule. They make their moves (which can e.g. be of the form of playing cards (Abbott 1977; Gardner 1977; Golden 2011) or building configurations with objects (Looney et al. 1997)) and the first player gives feedback as to whether a move was accepted or rejected. Then the players use this information to inductively infer the rule.

In the card game Eleusis, Player 1 – who in the game is referred to as *God* or *Nature* – comes up with a rule about sequences of cards. Then the other players – called *Scientists* – take turn in each playing a card in a sequence. After each move, Player 1 announces whether the card was accepted. Rejected cards are moved out of the sequence and stay below the position for which they were played. This way during the whole game, all players can see which cards have been accepted or rejected at which positions.

Eleusis has received attention within the philosophy of science literature, since it nicely illustrates scientific inquiry (Romesburg 1978): Playing the cards can be seen as performing experiments, and the feedback given by Player 1 (i.e., the acceptance or rejection of the cards played) can be thought of as the outcomes of the experiments. The players form hypotheses about the rule and choose to perform experiments accordingly, after each move updating their information state with the outcome of the experiment, and then revising their hypotheses. The game Eleusis can thus be seen as a nice simulation of scientific inquiry in which players employ two kinds of strategies: *selection strategies*, which determine what experiment to perform (i.e., what cards to play), and *reception strategies* for using the results of the experiments (i.e., the acceptance and rejection of the cards) for constructing and choosing hypotheses about the rule (Romesburg 1978). Eleusis has also been investigated within the computer science and artificial intelligence literature since there is a close relationship to pattern recognition as discovering a rule essentially means to discover a pattern in the sequence of accepted cards. Several algorithms have been developed taking the role of the scientist in Eleusis (Berry 1981; Diettrich and Michalski 1989; Michalski et al. 1985). Some sample secret rules have been classified informally with respect to the difficulty for the scientist players to discover them (cf. Golden (2011)). However, to the best of our knowledge, there has not been done any complexity theoretical analysis of Eleusis. In this chapter, we show that computational complexity plays a crucial role in Eleusis and give complexity results with a practical relevance for the actual play of the game. Player 1's choice of rule not only determines the difficulty of the tasks of the other players during the game but also has an impact for herself since as we show there are secret rules that Player 1 can choose that make it impossible for herself to give feedback to the other players since she is faced with undecidable problems during the play.

6.2 Eleusis: The rules

In this section, we will describe the rules of the card game Eleusis. There are several versions of the rules (Abbott 1977; Gardner 1977; Golden 2011). In this chapter, we will focus on *The New Eleusis* (Matuszek 1995). We first briefly give the rules in order to give the reader an idea of the actual game, as it is played in practice, and then proceed by pointing out some connections to similar games that the reader might be familiar with.

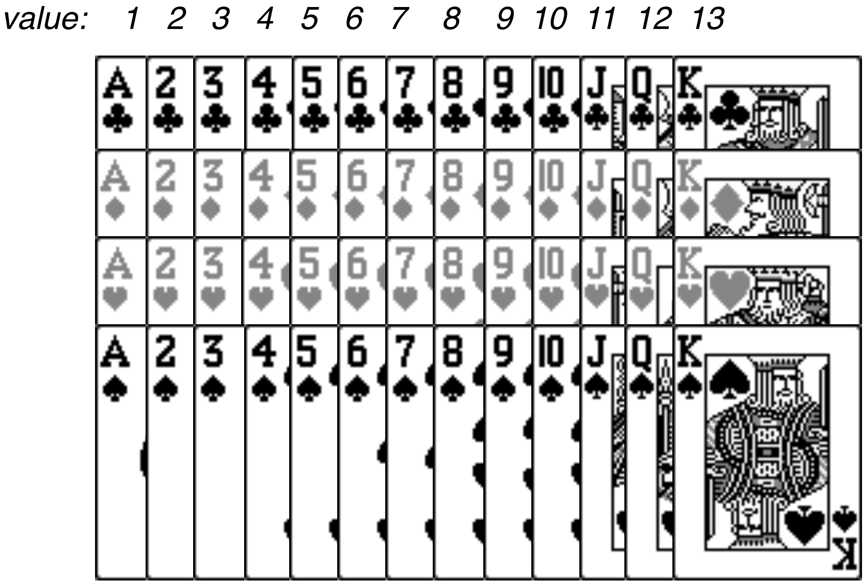

value: 1 2 3 4 5 6 7 8 9 10 11 12 13
A 2 3 4 5 6 7 8 9 10 J Q K

Figure 6.1: Deck of cards and their associated values.

6.2.1 The New Eleusis

The New Eleusis is a card game played with decks of cards as depicted in Figure 6.1. The number of decks needed depends on the number of players but a (small) finite number of decks is sufficient. We now briefly go through the rules of the game.

Beginning of the Game. One player (we call her *Player 1*) has the designated role of *God* or *Nature*. She starts the game by constructing a secret rule determining which sequences of cards are accepted. An example of such a rule is the following: *"every black card has to be followed by a card with an even value"*. Player 1 writes down the rule on a piece of paper without any other player seeing it.

Secret Rule. The only constraints on the secret rule are that it can only take into account the sequence of cards previously accepted and the card currently played. Thus, whether a particular card is accepted can only depend on the cards previously accepted and the card itself. External factors, such as who played the card or whether the player uses his left or right hand to play the card, have to be irrelevant.

Playing Procedure. Then each of the other players receives a number of cards (usually 14). Player 1 draws cards from the deck until she draws one that is accepted according to the rule; this card is called the *starter card* and will be the first card of what is called the *mainline*. Cards that have been rejected as starter card are placed below the starter card position, in the *sideline*. Then the other players take turns in each playing one of their cards by appending it on the right to the mainline. After each move, Player 1 announces whether this card is accepted according to the secret rule. If it is rejected, it is moved from the mainline to the *sideline*, directly below the position at which it was played in the mainline, and the player who played the card has to draw an additional card from the deck. In case the card played is accepted, it stays in the mainline and the player does not need to draw a card. If a player thinks that none of the cards on his hand would be accepted, he can declare *"no play"*. In this case, his hand of cards has to be shown to everyone, and Player 1 has to check whether indeed none of the cards would have been accepted. If this is the case, Player 1 gives him a new hand of cards, which should be one card less than the hand he had before, and the old hand of cards is placed above the mainline. If Player 1 finds a card that could have been played, he plays it and the player has to draw five cards from the deck. Figure 6.2 shows an example of a configuration of the game.

Further Tasks of Player 1 (God). Player 1 has to keep track of the number of accepted cards. This can be done by putting a marker every ten cards. After 40 accepted cards, a sudden death period starts in which the other players are expelled as soon as their card is rejected. If there is a prophet, Player 1 has to approve/disprove of each of the prophet's decisions concerning the acceptance of cards.

Becoming a Prophet. After playing a card, a player can declare himself Prophet, in case the following three conditions hold.

1. There is not already a prophet.
2. The player has not been a prophet before.
3. There are still at least two other scientist players in the game.

Being a Prophet. A prophet has to take over the job of the god player and has to announce whether played cards are accepted or rejected. After having done this correctly 30 times, a sudden death period starts in which the other players are expelled as soon as their card is rejected. If the prophet makes a mistake, he is overthrown by the god player and returns to normal play with his hand of cards and additionally five more cards.

End of the Game. The game ends in any of the following cases:

1. A player does not have any cards any more.
2. All players have been expelled during a sudden death period.

Scoring. The player with the highest score wins, where the score is calculated as follows.

- For the scientist players:
  - Everyone gets as many points as the highest number of cards held by any player minus the cards in his/her own hand.
  - Players without cards get four points bonus.
  - A prophet who survived until the end gets one point for each correctly accepted card and two points for each correctly rejected card.
- For the god player:
  - The god player's score is the minimum of the highest score of the other players and twice the number of cards played before the prophet who survived until the end started being a prophet.

Note that the idea is to play repeatedly until every player has been the god player once and add up the points of the plays.

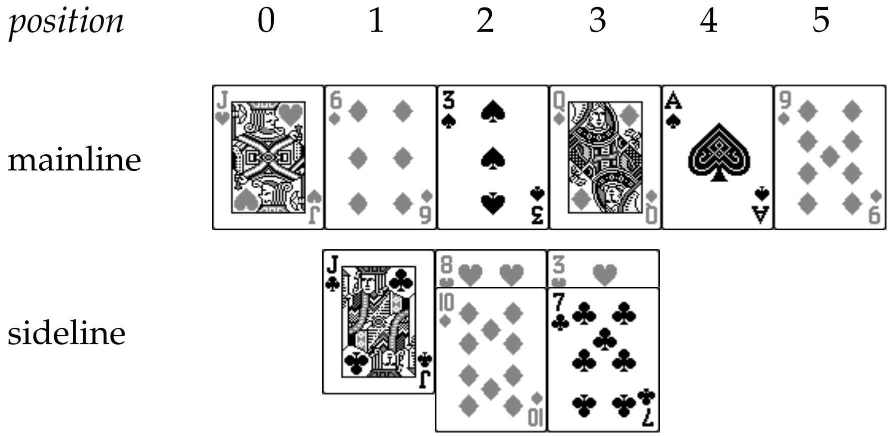


Figure 6.2: Example configuration of the game with e.g. the following secret rule *"Alternate cards of odd and even value, with the following exception: if an ace has been accepted, accept any card at the next position"*.

Discussion of the rules. The winning conditions of the game as given above seem rather complicated. The scoring rules reflect the aim of designing the game in such a way as to achieve that Player 1 has an incentive to choose a secret rule which is of a level of difficulty that makes the game entertaining to play[1]. Player 1 gets the most points if she chooses a rule that is difficult enough to not be discovered too quickly but still easy enough so that there is a player who will eventually become a prophet and survive as a prophet until the end of the game. Looking more closely into the scoring for the prophet, we can see that both for a successful prophet and for Player 1 it is best if the other players are still far from discovering the rule as the prophet (and eventually also Player 1 (depending on when the successful prophet became a prophet)) get more points for cards being correctly rejected by the prophet.

From the perspective of the scientist players, discovering the rule can surely be seen as profitable, as this allows them both to play the correct cards and also to be come a successful prophet. In the process of discovery a scientist player might face the problem of trade-off between playing cards which he expects to be accepted and cards he expects to have the highest *eliminative power* (Gierasimczuk and de Jongh 2010) with respect to the set of rules the player still considers possible candidates for being the secret rule.

[1] This is different from Eleusis Express (Golden 2011) in which the player taking the role of Player 1 does not get any points in that round.

6.2.2 Connection to other games

As discovering the rule plays a crucial role in Eleusis, it is closely related to the game *Zendo* (Looney et al. 1997), a game in which the goal is to inductively infer a secret rule about the configurations of pyramid shaped pieces.

Even though in Eleusis, discovering the rule is not the main objective of the scientist players, it still gives bonus points, and the reader familiar with the game *Mastermind* might see some similarities between Eleusis for two players and Mastermind. Mastermind is a code breaking game which has received a lot of attention within computer science (Knuth 1976; Kooi 2005; Stuckman and Zhang 2006) and also psychology (Best 2000; Verbrugge and Mol 2008). In this game, one player constructs a code consisting of four pegs that can each have one of six different colors. The other player starts by guessing the code and gets feedback from the first player saying how many colors were at the correct position, and how many were at wrong positions. The game continues until Player 2 has inferred the code. Whereas the roles of the players seem similar in Mastermind and Eleusis, there are some substantial differences. For Mastermind, there are strategies that allow a player to infer the secret code with certainty within a small number of rounds (e.g. five, cf. Kooi (2005)). In Eleusis, in general this is not possible as there are rules that cannot be identified with certainty at a finite stage of the game. Speaking in terms of formal learning theory, there are thus rules which are not finitely identifiable (Mukouchi 1992). To illustrate this, consider the situation in which the play seems to suggest that the secret rule is to alternate black and red cards. Then no matter how long the game has been going on already, it will still be possible that the rule says that only for the first n cards (for n being some number greater than the number of cards already accepted at the current position) black and red cards have to alternate, and from position $n + 1$ only red cards will be accepted.

Another difference between Eleusis and Mastermind is the impact of the chosen code or rule on the difficulty of the subsequent play. In Mastermind, the difficulty for Player 2 to infer the code and for Player 1 to check the guesses of Player 2 are similar for all the codes that Player 1 could choose. As we illustrate in Section 6.3, in Eleusis on the other hand, the choice of secret rule has a great influence on the difficulty of the game for both players.

6.3 Complexities in Eleusis

In this section, we will give a complexity analysis of different decision problems and tasks involved in Eleusis. It is important to note that we do not propose a game theoretical analysis of the game but give a complexity theoretical analysis of some decision problems involved in actually playing the game. One motivation for our study is to investigate the complexity involved in scientific inquiry, trying to determine what features of rules contribute to the difficulty of their inductive discovery. We are interested in the complexity that agents face in interactive processes involving inductive inference. Thus, we examine the complexity of the game Eleusis from an agent-oriented perspective focusing on different tasks the players face during the game rather than taking an external perspective examining the complexity of determining which player has a winning strategy. There are several levels of complexity in the game of Eleusis. On the one hand, there is the complexity or difficulty of playing the game itself, as there is the challenge for Player 1 to choose a rule of adequate complexity. Note that there is a close relationship between the complexity/difficulty of playing the game and the complexity of the secret rule.

One way to determine the complexity of the secret rules would of course be empirically, by determining how difficult it is for human subjects to discover them. This would lead to an interesting study identifying the features of rules about (finite) sequences that make their discovery easy or difficult. For the moment, we leave such an analysis to future work, and in this chapter we focus on a theoretical analysis of the complexity involved in Eleusis.

Another perspective from which we can investigate the complexity in Eleusis is to capture the complexity of the secret rules using methods from descriptive complexity by specifying the formal languages in which the rules can be expressed. This way, the complexity of a rule is captured by the expressive power required to express it in a formal language.

Example 6.1 As examples of rules of different descriptive complexity, consider the following two rules

1. *"At even positions, accept a card iff it is red, and at odd positions accept a card iff it is black."*
2. *"First accept two black cards, then three red, then five black, . . . then p_{2k} red, then p_{2k+1} black cards, etc."*, where p_n is the n-th prime number. ◄

Then, it is easy to see that Rule 1 can be expressed by a regular expression (cf. e.g. Chapter 2 of Sudkamp (1988)) while Rule 2 cannot. Rule 1 can be expressed by the following regular expression over the alphabet $\Sigma = \{(1,\clubsuit),(1,\blacklozenge),(1,\spadesuit),(1,\heartsuit),\dots,(13,\clubsuit),(13,\blacklozenge),(13,\spadesuit),(13,\heartsuit)\}$ representing a set of cards. Note that | stands for a Boolean "or".

$$((h \mid d)(s \mid d))^{*}(\epsilon \mid (h \mid d)),$$

with $h = ((1,\heartsuit) \mid \dots \mid (13,\heartsuit)), d = ((1,\blacklozenge) \mid \dots \mid (13,\blacklozenge)), s = ((1,\spadesuit) \mid \dots \mid (13,\spadesuit))$ and $c = ((1,\clubsuit) \mid \dots \mid (13,\clubsuit))$. For showing that Rule 2 cannot be expressed by a regular expression, we can use the pumping lemma (cf. e.g. pages 175–179 of Sudkamp (1988)).

With the only restriction of the acceptance of a card only depending on previously accepted cards, it is clear that rules from a wide range of the Chomsky hierarchy can be taken.

The complexity of the secret rules can also be analyzed by investigating the computational complexity of different decision problems arising from the secret rules. We will now present some of these decision problems informally and explain their motivation, before we will investigate them in more detail. Consider the following decision problems related to Eleusis.

1. Given a class of rules, a configuration of the game (i.e., a finite sequence of cards (accepted/rejected)), is there a rule in the class such that the play so far has been consistent with the rule (i.e., a rule that could have been the secret rule)?

2. Given a rule and a configuration of the game, is the play consistent with the rule?

3. Given a rule, a finite sequence of previously accepted cards, and a card c, is c accepted by the rule?

Problem 1 is the Eleusis analogue of the Mastermind-Satisfiability problem, which has been shown to be NP-complete (Stuckman and Zhang 2006). However, an important difference of the problem for Eleusis is that we restrict the class of secret rules. The reason for this is the following. Suppose we are given a sequence of cards that have been accepted/rejected in the game so far. Now, if we ask whether there is some rule that could be the secret rule that Player 1 constructed, we only need to check if no card has been both rejected and accepted at the same position. If there is no such card, then the answer to the question is *yes* because there are rules that are consistent with the play so far (e.g. the rule that explicitly says for each position to accept the card that actually has been accepted and to reject the cards that have been rejected).

Problem 2 is a problem that the scientist players encounter when analyzing the current situation in the game and deliberating whether a certain rule might be the secret rule. Problem 3 is relevant in the game because it describes the very task that Player 1 has to solve in each round. This problem is of course very relevant in practice and should be kept in mind by Player 1 when constructing the rule.

A closer investigation of these decision problems requires that we first formalize some aspects of Eleusis. Let us start by fixing some notation.

Notation

- We let *Card* be a finite set, representing the set of cards; alternatively we could also represent cards as a pair consisting of its value and its suit.

- $Card^*$ is the set of finite sequences of elements of *Card*.

- For $s \in Card^*$, $|s|$ denotes the length of the sequence s, defined in the standard way.

- s_i denotes the i-th element of the sequence of cards s, and $s_{<i}$ denotes the initial subsequence of s of length i, i.e., if $s = s_0 s_1 \dots s_i \dots s_n$, then $s_{<i} = s_0 s_1 \dots s_{i-1}$

- For $s, t \in Card^*$, st is the sequence of cards resulting from the concatenation of s and t.

- By $\overline{C}_i$, we denote the set of cards that have been rejected at position i. ◄

Next, we want to formalize the secret rules. Considering Eleusis in practice, human players mostly define rules in terms of certain properties or attributes that the cards have, such as *color, suit* and *value* but also properties of having a face (of some gender) and certain numerical properties of the value, such as being even/odd, greater/smaller than some number or being prime. Analyzing the reasoning involved in humans playing Eleusis requires a cognitively adequate representation of the rules in terms of the attributes and properties of the cards. Technically speaking however, all of the rules can of course also be expressed in terms of the cards itself. This is what we will do in this chapter.

An Eleusis rule says which sequences of cards are accepted and which are not. The way in which the rules are used in the game is that in each round Player 1 has to check whether it is accepted to extend the current sequence with a certain card. Thus, we represent rules as functions that tell us for every pair consisting of a sequence of cards and a single card whether appending the sequence with the card is allowed.

Definition 6.2 Eleusis rules ρ are functions $\rho : Card^* \times Card \to \{0, 1\}$. ◄

Note that with this definition, whether a card is accepted is fully determined by the sequence of cards that have been accepted so far; the previously rejected cards are irrelevant here. In practice, it can probably be observed that most rules chosen by human players have the property that accepted sequences are closed under taking prefixes, i.e., for any $s \in Card^*$, if $\rho(s, c) = 1$, then also for every $0 \leq i < |s|$, $\rho(s_{<i}, s_i) = 1$. However, in the rules of the game, this is not required (Matuszek 1995), and therefore neither will we do here. Note however that all rules we work with in this chapter are actually prefix-closed and our complexity results also hold for a definition of Eleusis rules that requires the rules to be closed under taking prefixes. To give the reader an intuition of what it means for Eleusis rules to be prefix-closed, we give an example of a rule that satisfies this constraint and one example of a rule that does not.

Example 6.3 As an example of a prefix-closed secret Eleusis rule consider the following rule.

- *Accept a card iff accepting it implies that all accepted cards are red.*

This rule on the other hand is not prefix-closed.

- *Accept a card iff after accepting it the sum of the values of all accepted cards is odd* ◄

We will later see that the property of being closed under prefixes plays an important role when we want to interpret complexity results about Eleusis rules with respect to their impact on the difficulties for actual play.

6.3.1 Is a play consistent with a given class of rules?

In the following we will focus on several restricted classes of rules.

Definition 6.4 (Periodic Rules) We call a secret Eleusis rule ρ **periodic** if it satisfies the following condition: There is some $p \in \mathbb{N}$ such that for all $s, s' \in Card^*, c \in Card$, if $|s| = |s'| = n$ and for all $0 \leq l < |s|$, it holds that if $l \mod p = n \mod p$ then $s_l = s'_l$, then it holds that $\rho(s, c) = \rho(s', c)$. We call the greatest such p the **number of phases** of ρ. A periodic rule ρ with p phases can then be written as a sequence of rules $(\rho_0, \ldots \rho_{p-1})$, where $\rho(s, c) = \rho_i(s, c)$ if $|s| \mod p = i$. ◄

Periodic rules are thus rules that can be split into different phases, each following some rule which is independent of the other phases. Let us give some examples of periodic rules with different numbers of phases.

Example 6.5 (Periodic Rules)

1. 1 Phase: *"At every position, accept all the red cards and the black ones with a male face. The other black cards are only accepted if they are preceded by two red cards."*

2. 2 Phases: *"On even positions only accept cards that have a face or a value greater than or equal to the one of the card at the previous even position. At odd positions, accept any card."*

3. 3 Phases: *"Two cards of even value, then one with an odd value, then two even ones again, etc."* ◄

Comparing these rules, we see that in Rule 3 we only need to look at the current position in order to determine whether a card is accepted. In Rule 1, on the other hand, if a black card without a male face is played, then Player 1 has to look at the two previously accepted cards in order to determine if the card is accepted. In Rule 2, on even positions we also have to look at the card that is placed at the previous even position, in order to check if a card is accepted.

This leads us to the notion of *lookback*, which is the length of the sequence of previously accepted cards that are relevant when deciding whether a card should be accepted.

Definition 6.6 (Lookback) Let ρ be an Eleusis rule ρ. Now if

$$\min\{l \in \mathbb{N} \mid \text{for all } c \in Card, s, s', s'' \in Card^* \text{ with } |s''| \leq l, \rho(ss'', c) = \rho(s's'', c)\}$$

is defined, we call it the **lookback** of ρ. ◄

Example 6.7 The following are example rules with lookback.

- Lookback 0: *"Accept all black cards and all red cards that have a face; reject all the others."*
- Lookback 1: *"If the previous card had a female face, accept only aces."*
- Lookback 2: *"Accept a card if and only if at least one of the following conditions is satisfied*
  1. *It is not the case that immediately before two cards with prime values have been accepted,*
  2. *The card is red."* ◄

Definition 6.8 (Periodic Rules with Lookback) We define $\mathbf{P}_l^p$ to be the class of periodic rules ρ of p phases, such that the maximum lookback of $\rho_0, \ldots, \rho_{p-1}$ is l. ◄

Intuitively speaking, the simplest secret rules in Eleusis are those that accept a card only on the basis of the card itself, and neither take into account previously played cards nor the position at which a card is played. These are the rules in the class $\mathbf{P}_0^1$.

Fact 6.9 *For every $\rho \in \mathbf{P}_0^1$, the following condition is satisfied: For all $s, s' \in Card^*$, and $c \in Card$, $\rho(s, c) = \rho(s', c)$. Every rule $\rho \in \mathbf{P}_0^1$ can thus be expressed as a function $\rho' : Card \to \{0, 1\}$.* ◄

Example 6.10 The following are examples of rules in $\mathbf{P}_0^1$.

1. *"Accept all red cards, reject all black cards."*
2. *"Accept all cards with a value ≤ 7, reject all the others."*
3. *"Accept all cards of clubs and all the ones of hearts that have an even value, reject all the others."* ◄

After we have introduced some formal notation and defined some classes of Eleusis rules, we will now start investigating the complexity of decision problems related to Eleusis. We start with the Eleusis satisfiability problem ESAT, which can be seen as an analogue to the problem investigated for Mastermind in Stuckman and Zhang (2006). For Mastermind, the problem asks given a configuration of the game, whether there is any secret code that is consistent with the play so far. For Eleusis, the problem ESAT is to determine whether, given a configuration of the game, there is some rule which is consistent with the play so far. If we do not make any restrictions onto the class of rules under consideration, this problem becomes easy as it boils down to just checking whether the same card has been both rejected and accepted at the same position[2].

Definition 6.11 For R being a class of Eleusis rules, the decision problem $\mathsf{ESAT}(R)$ is defined as follows.

Decision Problem 6.12 $\mathsf{ESAT}(R)$
Input: *A sequence of cards $s \in Card^*$, and for each $i, 0 \leq i \leq |s|$ a set $\overline{C}_i \subseteq Card$ (representing the cards rejected at position i).*
Question: *Is there some $\rho \in R$ such that for all i with $0 \leq i < |s|$, $\rho(s_{<i}, s_i) = 1$ and for all $c' \in \overline{C}_i$, $\rho(s_{<i}, c') = 0$?* ◄

The first class of rules for which we investigate this problem is the class of very simple rules $\mathbf{P}^1_0$. Given a configuration of the game Eleusis, it is quite easy to check whether it is possible that the secret rule that Player 1 has in mind is in $\mathbf{P}^1_0$. These rules are so simple because whether a card is accepted does not depend on the current position of the sequence on the table, and neither on the cards played so far. If during the play one card has ever once been accepted and once been rejected, then the secret rule cannot be in $\mathbf{P}^1_0$. On the other hand, if no card has been both accepted and rejected, then it is indeed possible that the secret rule is in $\mathbf{P}^1_0$. Any rule that accepts all the cards that have previously been accepted and rejects those who have not is a candidate.

Proposition 6.13 *The problem $\mathsf{ESAT}(\mathbf{P}^1_0)$ can be solved in polynomial time.*

Proof. Going through the sequence of cards, for each position, we check whether the card accepted at the current position is rejected at the same position or at a further position, and then for each card rejected at the current position, we check if this card is accepted at any future position. As soon as we find a card where any of this is the case, we can stop and the answer is *no*. If we reach the end of the sequence, the answer is *yes*. Since in this procedure each card in

[2] This is the case because whenever there is no card accepted and rejected at the same position, any function $\rho : Card^* \times Card \to \{0, 1\}$ that extends the current play (i.e., that accepts the accepted cards at the correct positions and rejects the rejected ones) is an Eleusis rule consistent with the game so far.

the input is compared to at most all the other cards, this takes time at most n^2 in the worst case, for n being the size of the input. ■

We now look at ESAT for periodic rules without lookback.

Proposition 6.14 *For any* $p \in \mathbb{N}$*, the problem* ESAT($\mathbf{P}_0^p$) *can be solved in polynomial time.*

Proof. First of all, if $p \geq |s|$, then we only need to check if there is some position $i \leq |s|$ such that $s_i \in \overline{C}_i$. If this is the case, the answer is *no*, otherwise the answer is *yes*. If $p < |s|$, then for all i, j such that $i \leq j < |s|$ and $j \mod p = i \mod p$, we check if $s_i \in C_j$ or $s_j \in C_i$. If this is the case for any such i, j, then we can stop, and the answer is *yes*. If there are no such i, j, then the answer is *no*. ■

We will now move to rules that do take into account previously accepted cards. Let us first consider ESAT($\mathbf{P}_1^1$). Rules in $\mathbf{P}_1^1$ have the property that whether a card is accepted is completely determined by the card accepted at the previous position. So, we have to look for an instance where the same card has been accepted at two positions, and at the immediate respective successors of these positions the same card has been rejected in one case and accepted in the other. If we find such an instance, we know that the secret rule cannot be in $\mathbf{P}_1^1$.

Proposition 6.15 ESAT($\mathbf{P}_1^1$) *can be solved in polynomial time.*

Proof. In order to solve this problem, we can go through the sequence of cards, and for all $0 \leq i < |s|$, we check if there are $i, j, i \leq j < |s| - 1$ such that $s_i = s_j$ and $s_{i+1} \in \overline{C}_{j+1}$ or $s_{j+1} \in \overline{C}_{i+1}$. If we find such i, j, then the answer is no. Otherwise, the answer is yes. ■

Note that given a sequence of cards, deciding whether there is some p such that there is a rule in $\mathbf{P}_0^p$ that could be the secret rule is trivial, as the answer is 'yes' if and only if no card has been accepted and rejected at the same position.

A similar problem to investigate would be the problem of given a sequence of cards and some $k \in \mathbb{N}$, decide whether there is some $p \leq k$ such that there is a rule in $\mathbf{P}_0^p$ that could be the secret rule. For this problem, it is sufficient to check if is the case that the secret rule could consist of k independent rules.

Solving ESAT($\mathbf{P}_k^1$), and in general ESAT($\mathbf{P}_k^p$) can be done analogously to Proposition 6.15; instead of looking for positions where the same card has been accepted, for each phase we have to look for two sequences of positions where the same k cards have been accepted, and then check if it is the case that at the next positions one card has been once accepted and once rejected.

Corollary 6.16 ESAT($\mathbf{P}_k^p$) *can be solved in polynomial time.* ◄

Thus, we have seen that for several classes of rules, it can be decided in polynomial time whether there is a rule in the class that is consistent with the play so far. In actual play, we can think of Player 2 solving this problem for various classes of rules, trying to restrict the set of rules that are still possible. In other words, coming back to Eleusis as a simulation of scientific inquiry, this is the problem describing the scientist checking whether there is some hypothesis in a certain class that is consistent with the experimental results so far.

6.3.2 A hard task for Player 1: accept or reject?

After having discussed various tractable decision problems in Eleusis, we will now show that Eleusis also gives rise to hard problems. We give a secret Eleusis rule that has the property that checking whether the sequence of cards on the table is consistent with the rule is NP-complete.

We use the *Collision-Aware String Partition Problem* (CA-SP) which Condon et al. (2008) have shown to be NP-complete. CA-SP is the problem of deciding whether a string can be partitioned into substrings of at most length k such that no two substrings are equal.

Decision Problem 6.17 (Collision-Aware String Partition) Input: *A string $s \subseteq \Sigma^*$, for Σ being a finite alphabet, and a natural number $k \in \mathbb{N}$.*

Question: *Is there a collision-free k-partition of s? That is, are there strings $p_1, \dots, p_l \subseteq \Sigma^*$ such that*

- $p_1 \dots p_l = s$,
- $|p_i| \leq k$ *for all i with* $1 \leq i \leq l$ *and*
- *for all i, j such that* $i \neq j$ *and* $1 \leq i \neq j \leq l$ *it holds that* $p_i \neq p_j$? ◀

Condon et al. (2008) investigate this problem also for strings over a four-letter alphabet (i.e., $|\Sigma| = 4$) and show that it stays NP-complete. The motivation behind studying the problem for an alphabet of size four is that it relates to problems involved in the synthesis of long strands of DNA (sequences over the alphabet $\{A, C, G, T\}$). We will show that CA-SP can also occur in a game of Eleusis.

The following is the task Player 1 has to solve in each round of Eleusis when giving feedback to Player 2. In the strategic considerations of Player 1, the complexity of this task plays of course a crucial role since in practice she should be able to solve it in reasonable time.

Decision Problem 6.18 $\mathsf{E-Check}(\rho)$
Input: *A sequence of cards $s \in Card^*$, and a card $c \in Card$.*
Question: *Is it the case that $\rho(s, c) = 1$?* ◀

Note that here we don't take ρ to be part of the input but keep it fixed. This way, we measure the complexity independently of the representation of the rule.

We now show that there are Eleusis rules that make it NP-hard for Player 1 to check if a card should be accepted. One such example is a rule that forces Player 1 to solve the problem CA-SP, because she has to check if the sequence of suits of the cards accepted so far can be partitioned into k substrings such that no two of them are equal. The parameter k will be given by the sequence of accepted cards, or more precisely, by the position of the first King that has been accepted. The rule we define does not force Player 1 to solve the CA-SP problem in every round, but only whenever a so called *trigger* card is played. This trigger card, $(7,\clubsuit)$ is accepted if and only if the sequence of previously accepted cards represents a positive instance of CA-SP.

$$\rho_{CA-SP}(s,c) := \begin{cases} 1 & \text{if } c \neq (7,\clubsuit) \text{ or there is no } i \text{ such that } 0 \leq i < |s| \text{ and } Values_i = 13 \\ & \\ & \text{or } c = (7,\clubsuit) \text{ and } k := \min\{i \mid Value(s_i) = 13\} \text{ is defined and it holds that } suit(s_0)suit(s_1)\ldots suit(s_{|s|-1}) \text{ can be } k\text{-partitioned into strings of length at most } k \text{ such that no two strings are equal.} \\ 0 & \text{otherwise.} \end{cases}$$

Now, the following proposition follows immediately.

Proposition 6.19 $\mathsf{E}-\mathsf{Check}(\rho_{CA-SP})$ *is* NP*-complete.*

Proof. NP membership is straightforward as it can be checked in polynomial time if a proposed k-partitioning of the sequence of cards is indeed correct in the sense that no substring is longer than k and no two are equal. NP-hardness follows by reduction from CA-SP, as we will show now.

Given an instance of CA-SP over the alphabet $\Sigma = Suit = \{\heartsuit, \diamondsuit, \spadesuit, \clubsuit\}$ i.e., a sequence $s \subseteq \Sigma^*$ and some $k \in \mathbb{N}$, we transform it into the following instance $f(s,k)$ of $\mathsf{E}-\mathsf{Check}(\rho_{CA-SP})$.

$$(s_0, n_0)\ldots(s_{|s|-1}, n_{|s|-1})(7,\clubsuit),$$

where $n_k = 13$ and for all the other n_i with $i \neq k$ we let $n_i \in \{1,\ldots,12\}$.

Now, if s,k is a positive instance of CA-SP, then s can be k-partitioned into different substrings, which then implies that $(7,\clubsuit)$ is accepted as the sequence of the suits of the cards accepted before can be k-partitioned in the same way.

The other direction is also immediate: If $(7,\clubsuit)$ is accepted, then as there is a King in the sequence it means that the sequence of the suits of the of cards played before can be k-partitioned into different substrings, for k being the position of the first King. Hence, (s,k) is also a positive instance of CA-SP. ■

The property that we used in the above proof is that the cards can be used to code hard problems. We chose a reduction from the collision-aware string partition problem because the transformation from a string of a four letter alphabet to a sequence of cards of four different suits is particularly straightforward, showing that the coding of hard problems is not only theoretically possible but can even be practically feasible to do in the actual play of the game. This leads us to the question of what our analysis means for Player 1's choices in the game. Our hardness result shows that when Player 1 is constructing a secret rule, she should be aware of its complexity to ensure that she won't be faced with intractable problems when she has to give feedback to the other players, as it happens with the rule ρ_{CA-SP}.

6.3.3 An impossible task for Player 1: accept or reject?

We can now show that Eleusis allows for even harder rules: we give an Eleusis rule such that it can get undecidable to compute whether a card should be accepted. We will first introduce some notation for this.

Notation • We now use standard decks of cards, and let $\mathit{Card} = \mathit{Value} \times \mathit{Suit}$, for $\mathit{Value} = \{1, \ldots, 13\}$ and $\mathit{Suit} = \{\heartsuit, \diamondsuit, \spadesuit, \clubsuit\}$.

- For a sequence of cards $s \in \mathit{Card}^*$, $\mathit{value}(s)$ denotes the sequence of the values, i.e., $\mathit{value}(s) = \mathit{value}(s_0) \ldots \mathit{value}(s_{|s|-1})$.
- We define a function $\mathit{color} : \mathit{Card} \to \{b, r\}$, assigning to each card its color (black or red), defined as follows.

$$\mathit{color}(c) = \begin{cases} b & \text{if } \mathit{suit}(c) \in \{\spadesuit, \clubsuit\} \\ r & \text{if } \mathit{suit}(c) \in \{\heartsuit, \diamondsuit\} \end{cases}$$ ◀

We now define the set of *black (red) words* in a sequence of cards. The set of black (red) words in a sequence of cards contains all the maximal subsequences of black (red) cards in the sequence, i.e., the subsequences of black (red) cards that are separated by red (black) cards. Let us illustrate this with an example. Given the sequence $s = (4, \spadesuit)(3, \clubsuit)(9, \diamondsuit)(8, \spadesuit)$, its set of red words is the singleton $\{(9, \diamondsuit)\}$, and its set of black words is $\{(4, \spadesuit)(3, \clubsuit), (8, \spadesuit)\}$.

Definition 6.20 For a sequence $s \in \mathit{Card}^*$, we define the *set of black words* of s $BW(s)$ to be the set of all those $w \in \mathit{Card}^*$ with $|s| \geq 1$ that satisfy the following conditions.

1. $\forall i$ such that $0 \leq i < |w|$ it holds that $\mathit{color}(w_i) = b$ and
2. $\exists i$ such that $0 \leq i < |s|$ and $\forall j$ with $0 \leq j < |w|$ it holds that $s_{i+j} = w_j$ and
   (i) if $i > 0$, then $\mathit{color}(s_{i-1}) = r$ and

(ii) if $i + |w| < |s|$, then $color(s_{(i+j)+1}) = r$.

The *set of red words* of a sequence of cards s, $RW(s)$, is defined analogously by swapping r and b in the above definition. Then, the *multiset of black words* $\mathcal{BW}(s)$ of a sequence of cards s is defined as $\mathcal{BW}(s) = (BW(s), m)$, where m gives the multiplicity of how many occurrences of each black word in $BW(s)$ there are in s. $\mathcal{RW}(s)$ is defined analogously. ◀

For constructing a rule that gives rise to an undecidable problem, we will use the above definition and view a sequence of cards as a sequence of black and red words. Our proof of undecidability is by reduction from Post's Correspondence Problem (Post 1946).

Decision Problem 6.21 Post's Correspondence Problem
Input: *A finite set of pairs of non-empty strings over a finite alphabet* Σ, $P = \{(x_1, y_1), \dots (x_n, y_n)\}$.
Question: *Is there a sequence* $(i_1, \dots i_m)$ *for some* $m \in \mathbb{N}$, *with* $1 \le i_j \le n$ *such that for all* $1 \le j \le m$

$$x_{i_1} \dots x_{i_m} = y_{i_1} \dots y_{i_m}? \qquad ◀$$

Even if Σ is small ($|\Sigma| = 2$), the problem is undecidable (Ruohonen 1983).

We define an Eleusis rule that has the property that the problem of deciding whether a card should be accepted is in general at least as hard as solving Post's Correspondence Problem. Before giving the formal definition, let us explain the intuition. The idea of the rule is the following. Every card which is not the trigger card $(7, ♧)$ is accepted. The trigger card is accepted if and only if the following holds: If we view the sequence of previously accepted cards as a sequence of pairs, each consisting of a red word and a black word, then it is possible to rearrange the order of these pairs (possibly using a pair more than once or not at all) such that the resulting string of red values is the same as the resulting sequence of black values. Let us illustrate this with an example showing a positive instance. Consider the sequence of accepted cards $(9, ♤)(3, ♥)(9, ♦)(10, ♧)(3, ♧)(3, ♦)(10, ♦)(3, ♤)(3, ♧)(3, ♥)$, and assume that the next card being played is the trigger card $(7, ♧)$.Reading the sequence of previously accepted cards as a sequence of red and black words, gives

$$\underbrace{(9, ♤)}_{w_1^b}\ \underbrace{(3, ♥)(9, ♦)}_{w_1^r}\ \underbrace{(10, ♧)(3, ♧)}_{w_2^b}\ \underbrace{(3, ♦)(10, ♦)}_{w_2^r}\ \underbrace{(3, ♤)(3, ♧)}_{w_3^b}\ \underbrace{(3, ♥)}_{w_3^r}.$$

Now, (3,2,2,1) is a solution since $(value(w_3^b)\ value(w_2^b)\ value(w_2^b)\ value(w_1^b))$ = (3 3 10 3 10 3 9) = $(value(w_3^r)\ value(w_2^r)\ value(w_2^r)\ value(w_1^r))$. Similarly, $(3, 2, 1)$ is a solution. Thus, $(7, ♧)$ is accepted.

Formally, we define the rule ρ_{Post} as follows.

$$\rho_{Post}(s,c) := \begin{cases} 1 & \text{if } c \neq (7,\varclubsuit) \text{ or} \\ & |\mathcal{BW}(s)| \neq |\mathcal{RW}(s)| \text{ or} \\ & |\mathcal{BW}(s)| = |\mathcal{RW}(s)| \text{ and } s = w^r_1 w^b_1 w^r_2 w^b_2 \dots w^r_k w^b_k \text{ with} \\ & w^b_l \in BW(s) \text{ and } w^r_l \in RW(s) \text{ then } \exists(i_1 \dots i_m) \text{ with } 1 \leq i_j \leq k \\ & \text{ and } (value(w^r_{i_1}) \dots value(w^r_{i_m})) = (value(w^b_{i_1}) \dots value(w^b_{i_m})) \text{ or} \\ & |\mathcal{BW}(s)| = |\mathcal{RW}(s)| \text{ and } s = w^b_1 w^r_1 w^b_2 w^r_2 \dots w^b_k w^r_k \text{ with} \\ & w^b_l \in BW(s) \text{ and } w^r_l \in RW(s) \text{ then } \exists(i_1 \dots i_m) \text{ with } 1 \leq i_j \leq k \\ & \text{ and } (value(w^r_{i_1}) \dots value(w^r_{i_m})) = (value(w^b_{i_1}) \dots value(w^b_{i_m})); \\ 0 & \text{otherwise.} \end{cases}$$

Even though, this rule looks more complicated than the rules we have previously discussed, it is easy to see that it is still a proper Eleusis rule as it can be written on a small piece of paper and moreover the acceptance of a card only depends on the card itself and previously accepted cards. We now show undecidability of $\mathsf{E} - \mathsf{Check}(\rho_{Post})$, the problem of deciding whether for a given sequence $s \in Card^+$, $\rho_{Post}(s) = 1$.

Theorem 6.22 $\mathsf{E} - \mathsf{Check}(\rho_{Post})$ *is undecidable.*

Proof. By reduction from Post's Correspondence Problem with alphabet $\Sigma = Value = \{1, \dots 13\}$. Given $P = \{(x_1, y_1), \dots (x_n, y_n)\}$ with $x_j, y_j \in Value^*$, we transform it into a sequence of cards. We define a (partial) function $g : Value^* \to (Value \times Suit)^*$ as follows: For each $(x_i, y_i) \in P$, we define $g(x_i) = (x_{i0}, \varheartsuit)(x_{i1}, \varheartsuit) \dots (x_{i|x_i|-1}, \varheartsuit)$ and $g(y_i) = (y_{i0}, \varspadesuit)(x_{i1}, \varspadesuit) \dots (y_{i|y_i|-1}, \varspadesuit)$. Then, let $g'(P) = g(x_1)g(y_1) \dots g(x_n)g(y_n)$ and finally define the reduction function f as follows: $f(P) = (g'(P), (7, \varclubsuit))$. First of all, note that f can be computed in polynomial time since g' and g can be computed in polynomial time. Now, we have to show that f is indeed a proper reduction.

Assume that $P = \{(x_1, y_1), \dots (x_n, y_n)\}$ is a positive instance of Post's Correspondence Problem. Then there is a sequence $(i_1 \dots i_m)$, with $1 \leq i_j \leq n$ such that $x_{i_1} \dots x_{i_m} = y_{i_1} \dots y_{i_m}$. Now, we have to show that $\rho_{Post}(f(P)) = \rho_{Post}(g'(P), (7, \varclubsuit)) = 1$. First of all, note that by construction $|\mathcal{BW}(g'(P))| = |\mathcal{RW}(g(P))|$. Moreover, $(value({w^r}_{i_1}) \dots value({w^r}_{i_m})) = (value({w^b}_{i_1}) \dots value({w^b}_{i_m}))$. Thus $\rho_{Post}(f(P)) = 1$.

For the other direction, assume that $\rho_{Post}(f(P)) = 1$, for $f(P) = (g'(P), (7, \varclubsuit))$. By construction of $g'(P)$, it has to be the case that $|\mathcal{BW}(g'(P))| = |\mathcal{RW}(g'(P))|$, and $g'(P)$ has to start with a red card. Thus, $g'(P) = {w^r}_1 {w^b}_1 {w^r}_2 {w^b}_2 \dots {w^r}_k {w^b}_k$ with ${w^b}_l \in BW(g'(P)), {w^r}_l \in RW(g'(P))$ and there is a sequence$(i_1 \dots i_m)$ with $1 \leq i_j \leq k$ such that $(value({w^r}_{i_1}) \dots value({w^r}_{i_m})) = (value({w^b}_{i_1}) \dots value({w^b}_{i_m}))$. But then it must also be the case that $x_{i_1} \dots x_{i_m} = y_{i_1} \dots y_{i_m}$. This concludes the proof. ■

We have thus shown that whereas there are various tractable problems in Eleusis, the game also gives rise to NP-complete problems and problems that are undecidable even when played with a standard deck of cards.

This section has shown that the inductive inference game Eleusis is interesting from a complexity theoretical point of view as it gives rise to decision problems of various complexities. We also showed that there are hard decision problems that are relevant for the actual play of the game, as they are not about deciding which player has a winning strategy as the usual complexity results about games, but instead describe the tasks the players face during the game. Considering the problem $\mathsf{E} - \mathsf{Check}(\rho)$, we have seen that as opposed to Mastermind, in Eleusis the complexity for Player 1 crucially depends on her choice at the beginning of the game, as some choices can make it impossible for her to make a move, i.e., to give feedback to Player 2. Therefore, we have shown that the first player has a very active role in Eleusis, and as opposed to the literature where the difficulty of Eleusis is only discussed with respect to the difficulty to discover certain rules, our results show that Player 1's first move has crucial complexity implications also for herself. Coming back to Eleusis as a simulation of scientific inquiry, our work thus fits with approaches putting forward an interactive view on learning, with the environment or teacher having an active role (Gierasimczuk 2010).

Linking up to previous chapters. As the current chapter presents a very concrete setting, the reader might wonder how this relates to some of the abstract concepts discussed in previous chapters.

Preferences In this concrete game, preference of the players can be seen as given by the points they get at the end of the game.

Coalitional power Cooperation plays a crucial role in the game as in many situations players can have an incentive to cooperate, e.g. in order to stop some other player from winning or to be able to test some hypotheses about the secret rule which they could not have tested individually because they do not have the cards needed for that.

Information Players have perfect information about the course of the game played so far but do not know what cards the other players have and of course the Scientists do not know the secret rule. Playing a card and getting feedback reduces the uncertainty as some hypotheses might be discarded based on the feedback given by Player 1.

6.4 Conclusions and Further Questions

We start by summarizing the main results of this chapter which we obtained by giving a complexity theoretical analysis of different problems arising in the game Eleusis.

As technical methods for showing hardness and undecidability, we used reductions from a variation of the Partition problem which has relevance for the synthesis of long strands of DNA, and a reduction from Post's Correspondence Problem, respectively.

6.4.1 Summary

This chapter brought together complexity theory, game theory and learning theory. We investigated the inductive inference game Eleusis and gave a computational complexity analysis of different tasks that players face during the play of the game, ranging from polynomial time solvable to undecidable.

Tasks for scientists.

First of all, we have shown that for the natural class of periodic secret rules with a fixed number of phases and lookback it can be decided in polynomial time whether there is such a rule that is consistent with the current state of the game. For the actual play of the game this means that if before the game players agree to only use rules from one of these classes, it will be tractable for the scientist players to check during the game whether a rule they have in mind might be the secret rule.

Tasks for Player 1.

Moreover, our results also show that Eleusis can give rise to intractable problems. We have constructed a rule that requires the players to solve the NP-complete Collision-Aware String Partition Problem in order to decide if a card should be accepted.

Finally, using Post's Correspondence Problem, we showed that – even when played with standard decks of cards – Eleusis allows for rules that make it undecidable for Player 1 to check whether cards are accepted. This result implies unplayability of the game in practice: Following the official rules of the game, Player 1 can get into a situation in which she cannot decide anymore whether a card should be accepted, and thus cannot perform a legal move any more.

6.4.2 Conclusions

Let us come back to our research question.

Research Question 4 *Finally, to what extent can we use a formal analysis of interactive processes to draw conclusions about the complexity of actual interaction?*

- *Are there concrete examples of interactions in which participants actually encounter very high complexities which make it impossible for them to act?*

Based on our results in this chapter, we can give the following answers.

1. A complexity theoretical analysis of algorithmic tasks in recreational games allows us to draw conclusions about the complexity that players face during play.

   The main challenge for a formal complexity theoretical study to be able to have some impact on real interaction seems to be carefully choose appropriate problems to be analyzed. While complexity results for logical theories or problems arising in sophisticated strategical reasoning do not seem to have immediate practical implications[3], the study of tasks in recreational games which players cannot avoid seems to be promising.

2. In the game Eleusis we could identify undecidable problems which a player can be forced to face after some move she has made in the first round (i.e., constructing a secret rule which involves undecidable problems).

All our complexity results also extend to other versions of Eleusis such as Eleusis Express (cf. Golden (2011)), but not immediately to other inductive inference game such as Zendo (Looney et al. 1997).

Recommended rule adjustments. Based on our analysis, we give two recommendations for adjusting the rules of Eleusis, restricting the set of secret rules that Player 1 can choose from.

1. For the sake of actual playability, having in mind the limited computational power of actual players we recommend to restrict the set of Player 1′s possible choices of secret rules with respect to the complexity of checking whether a card is accepted (Decision Problem 6.18).

2. For keeping the game entertaining, we suggest to explicitly require that a secret rule should satisfy the following condition.

At every position, there is at least one card that is accepted.

[3] At least not without having carefully determined the precise connection between such theories and forms of reasoning and interaction of real agents.

The first suggestion could of course be made precise by formulating it in terms of the computational complexity of the decision problem $\mathsf{E}-\mathsf{Check}(\rho)$ (Decision Problem 6.18). For the actual play however, simply adding an appropriate time-limit for Player 1 for deciding about the acceptance of a card (and a penalty in terms of point deduction or immediate loss in case the time limit is exceeded) would solve the potential problem, as this would make it more apparent – especially for beginning players – to keep in mind the complexity of the secret rule.

The second adaptation of the rules of the game ensures that it cannot happen that at some point in the game all cards will be rejected. The following secret rule would e.g. be forbidden by our second suggestion.

> *"Accept any card as the first card. Then accept a card if and only if its value is greater than that of the previously accepted card"*

This rule could be easily adapted by adding the exception *"After a King has been accepted, accept any card at the next position"*. While in principle it is not a problem to have a secret rule that at some point rejects all cards, we have noticed that especially beginning players construct such rules without being aware that at some point no cards will be accepted any more.

It is important to note that our recommended adjustments of the rules are not aimed towards changing the actual game but rather to make explicit some particularities of the game in order to make Player 1 aware of the consequences of her choice of secret rule so that she can avoid rules that lead to undesired pathological plays. These adjustments are probably unnecessary for experienced players but certainly helpful for beginners.

6.4.3 Further Questions

From the complexity theoretical analysis of Eleusis given in this chapter, a number of directions for further research arise.

Are there other examples of recreational games in which players can be forced to encounter undecidable problems? The game we chose seems to be quite special in the sense that the secret rules about sequences could in principle be used to encode problems of arbitrary complexity. For other inductive inference games such as Zendo, constructing very hard rules seems to be more difficult than in Eleusis. It remains to be investigated whether there are also other classes of recreational games which are actually being played and in which very hard problems can arise already for just making a legal move.

Game theoretical analysis of Eleusis. A precise game theoretical analysis of Eleusis still has to be given. A first suggestion would be to look at versions in which the set of rules to be chosen from is restricted. When considering the strategic abilities of the players, the role of cooperation also needs to be taken into account, as there could be an incentive for coalitions consisting of Player 1 and a Scientist to form, as collusion of such a form could lead to high payoff for those two players.

Challenges for AI in games Eleusis as such presents some challenges for AI methods for game playing due to high complexities that can arise. Considering different simpler two-player variations in which Player 1 can only choose rules of a certain (manageable) complexity would then allow an analysis of these variations with respect to the existence of winning strategies for the players, e.g. using methods from Schadd (2011).

Eleusis, complexity and formal learning theory. The work in this chapter also promotes a categorization of secret Eleusis rules not only with respect to their difficulty of being discovered but also with respect to how difficult it is for the first player to give feedback to the other players. Our work thus fits with approaches to formal learning theory that consider the learning process as an interaction between a learner and a teacher (Gierasimczuk 2010; Gierasimczuk et al. 2009b). Considering variations of Eleusis in which Player 1's feedback is more refined than simply accepting or rejecting a card, Eleusis can also be used as a concrete setting in which different levels of helpfulness of a teacher can be illustrated.

More generally, the current work promotes the computational complexity analysis of inductive inference games, showing that a variety of interesting problems arise, ranging from very easy to undecidable. This complexity theoretical perspective gives us new insights into the strategic abilities of agents engaged in interactive processes that involve inductive inference and also highlights the special role complexity plays in inductive inference games, distinguishing them from other inference games such as Mastermind.

Empirical investigation of inductive inference games A natural follow-up of our analysis would be an empirical investigation of the game being played by human players. As a first step towards this, we refer to the webpage of Sangati (2011) which is used to collect data of plays of the game. We will come back to this in Section 7.3.4.

Chapter 7

Conclusion

This dissertation analyzed the (computational) complexity of interaction from different perspectives. We started the investigation from an external perspective, analyzing the complexities of modal logical systems designed for reasoning about the strategic abilities of individuals and groups of agents involved in interactive processes. We then zoomed in more into precisely defined game-like interactions, focusing on the complexity of deciding whether a player has a winning strategy. We then moved on by analyzing a different concept involved in interactions between agents, namely that of information. We determined the complexity involved in various tasks about comparing the information that agents have about facts and about other agents. Finally, we gave complexity results for tasks that are involved in actually playing a particular recreational game in which the concept of information plays a crucial role.

We now summarize the results of each individual chapter before we will give some general conclusions.

7.1 Summary of the chapters

Chapter 2 presented an extended modal logic framework for reasoning about the strategic ability of groups of agents. The cooperation logic with actions and preferences (CLA+P) was designed by extending the cooperation logic with actions of Sauro et al. (2006) with a modal preference logic, which is a fragment of the preference logic developed by van Benthem et al. (2007). The particular contribution of our framework to the field of modal logics for multi-agent systems is its combination of explicit actions and preferences, which allows for explicitly distinguishing between different ways to achieve results, w.r.t. whether these ways are good for the agents. We showed decidability (in NEXPTIME) of this logic by determining an upper bound on how many actions are needed to make coalitional power in implicit coalition modalities explicit and by adapting the technique of filtration to handle the non-standard modalities of the logic such as the strict preference modality and the modality of saying that performing a certain action leads to a (strictly) preferred state.

For a lower bound, we showed that already the fragment of CLA+P that only deals with actions and their effects is EXPTIME-hard as it basically is a full Boolean modal logic.

The design choices made for CLA+P were based on the conceptual motivation to keep the models very general, allowing for a wide range of situations being modeled.

The work in Chapter 2 raised the question as to whether we can give some guidelines for making design choices for developing modal logics that can express certain concepts inspired by game theory or social choice theory. In particular, Chapter 2 illustrated the need for a systematic study of the impact that the choice of primitives of a modal logic for strategic ability has on the complexity required for expressing interesting properties involving strategic ability and preferences.

Chapter 3 picked up this question and focused on the computational complexity required for reasoning involving game theoretical concepts. In this chapter, we investigated three different approaches to modeling the ability of groups in modal logic: simple coalition-labeled transition systems, action-based coalitional models and power-based coalitional models, which are a generalization of the simulation of Coalition Logic (Pauly 2002a) on Kripke models. We have clarified the framework of power-based coalitional models by showing how standard assumptions on coalitional power such as coalition monotonicity, independence of coalitions and a consistency condition for complementary coalitions relate to each other on these models. Moreover, we gave a transformation from power-based coalitional models to corresponding models of Coalition Logic. The existence of such a transformation follows from earlier results by Broersen et al. (2007). The contribution of Chapter 3 is to give an explicit constructive transformation showing how the powers of coalitions in power-based coalitional models can be transformed into effectivity functions. Clarifying the relationship between implicit and explicit coalitional power, we showed how a power-based coalitional model can be constructed from an action-based coalitional model. Additionally, we showed how the properties of an action-based model being reactive and its transitions being determined by the choice of the grand coalition translate into a power-based coalitional model being the normal simulation of a Coalition Logic model.

For each of the three classes of models, we determined under what model theoretical operations certain properties about cooperation and preferences are invariant. The properties range from the simplest notions about coalitional power or preferences (e.g. *a coalition having the ability to make a fact true*, or *an individual preferring a state in which some fact is true*, respectively) to more complex combinations such as the ability of a group to make the system move into a state preferred by an agent, or the concept of (strict) Nash-stability.

Using the invariance results and underlying characterization results for extended modal logics, we determined how much expressive power is needed for expressing the different concepts on each of the three classes of models. This way we identified extended modal logics in which the concepts can be expressed. Then we explicitly gave formulas in the languages determined this way that express the concepts. Finally, using complexity results for (extended) modal logics, we could then for each concept and each class of models specify an upper bound on the complexity (model checking and satisfiability) of modal logics being able to express this concept.

We showed that whether the aim of the designer is to develop a formal system for reasoning about stability notions involving strict preferences or stability notions with weak preferences can make a crucial difference w.r.t. which of the three classes of formal systems leads to the lowest complexity. We showed that on simple coalition-labeled transition systems strong Nash-stability is easier to express than Nash-stability, while on action- and power-based models we see the opposite effect: on these classes of modes strong Nash-stability turns out to be more demanding in terms of complexity and expressive power.

Chapter 4 focused on the complexity analysis of the problem of deciding which player has a winning strategy in different versions of Sabotage Games, two-player games played on graphs. A key feature of Sabotage Games are the asymmetric roles of the two players, Runner and Blocker: Runner moves locally along the edges of the graph, while Blocker's moves are of a more global nature: she manipulates the graph by removing edges and thereby restricts possible choices of moves by Runner. In the standard game, the goal of Runner is to reach one of the goal vertices, while Blocker tries to prevent this from happening. In our work, we examined the effects of different winning conditions on the complexity. We showed that with opposite objectives (i.e., Runner trying to avoid reaching the goal vertices, and Blocker trying to force him to move to a goal vertex) the complexity of deciding if a player has a winning strategy remains unchanged (PSPACE-complete). In a cooperative setting in which both players' aim is that Runner reaches a goal vertex however, the game becomes easier (NL-complete).

For each of the three versions of winning conditions, we also determined the complexity of the game in which Blocker is allowed to skip moves. Our results show that the complexity stays the same because the winning abilities in this game are as in the version in which Blocker has to remove an edge in every round.

Chapter 5 is devoted to the concept of information and more specifically to the complexity of tasks about comparing the information of different agents. Our study took place in the semantic structures of (epistemic) modal logics. We focused on three different classes of decision problems: determining if agents have similar information, if they have symmetric information in the sense that they have similar knowledge about each other and deciding if the information of an agent can be manipulated in a certain way.

We used the semantic structures of (epistemic) modal logics but our results are independent of how certain properties can be expressed in the syntax of such logics, as we purely focus on the tasks involved in reasoning about the information of agents. Our results show that deciding information similarity and information symmetry are in general tractable if we take similarity notions based on the notion of bisimilarity. We introduced the notion of flipped bisimilarity, which can be used to capture that two agents have similar information about each other. We also used the notion of epistemic horizon of an agent, which is the submodel with exactly that part of a Kripke model that is relevant for the reasoning of an agent in a given situation. We showed that in reflexive models (i.e., models in which it holds that whatever is known by an agent has to be true) horizon bisimilarity becomes trivial for the horizons at the same point in a model, while flipped horizon bisimilarity does not.

Moreover, we showed that deciding about whether it is possible to manipulate the information structure of agents in a certain way is in general more difficult than deciding information similarity or symmetry. However, under the assumption of information being modeled by S5 structures, deciding whether an information structure can be restricted such that it is similar to (at least as refined as) another structure is indeed tractable in the single-agent case. We gave a polynomial procedure that uses a polynomial algorithm for finding matchings in a bipartite graph. Whether for the multi-agent S5 case the problem becomes NP-complete is still open and depends on whether we can simulate arbitrary accessibility relations by combinations of equivalence relations in a way that preserves the existence of submodels bisimilar to some other model.

Concerning the location of the border between tractability and intractability, our results show that static tasks about similarity and symmetry are tractable, with some being among the hardest tractable problems (e.g. bisimilarity of Kripke models or horizons of Kripke models) and others being trivial (e.g. under the assumption of S5 models to decide whether the epistemic horizons of two agents in the same situation are similar), while for the dynamic tasks about information manipulation NP-hardness can arise quite quickly unless we consider single-agent S5 structures.

Chapter 6 presented a case study of complexity in interaction by focusing on the complexity involved in playing a concrete recreational game: the inductive inference game *The New Eleusis*. This chapter served as an example of a formal complexity analysis with implications for the actual play of a real game. Eleusis is a card game in which one player (Player 1) constructs a rule about sequences of cards and the other players try to find out the rule by inductive reasoning based on feedback they get as to whether cards they played are accepted or rejected according to the secret rule. The game is interesting from a learning theoretical perspective as it illustrates a form of learning with membership queries.

We formalized the secret rules as functions that for every sequence of cards and any card say whether it is accepted to extend the sequence with the card. We identified different tasks that players face during the play and investigated their complexity. We showed that for some natural classes of rules the problem of deciding whether the secret rule might be in this class can be done in polynomial time. For the task of Player 1 to say whether a card is accepted or rejected, we have shown that Player 1 can choose secret rules that will make it extremely hard – if not impossible – for her to perform this task and give feedback to the other players. More precisely, our analysis shows that Player 1 can construct a rule based on the NP-complete *collision-aware string partition problem* which can force her to eventually solve this problem. Moreover, we showed that indeed even undecidable problems can arise in the game, as the rules of the game allow e.g. Player 1 to construct a secret rule that requires her to solve *Post's correspondence problem* (which is undecidable) in order to say if a card is accepted. Based on this complexity analysis, we gave the suggestion of restricting the set of secret rules Player 1 can choose from as to explicitly avoid rules that make it impossible for Player 1 to perform a legal move (i.e., to give accurate feedback). In practice, Player 1's awareness of the impact of her choice of secret rule could be sufficiently increased by introducing a time limit within which she has to accept/reject cards in each round. A further constraint on the secret rules can be to always accept at least one card. As opposed to the first adjustment of the rules, this one is not aimed at making the game more playable for players with restricted computational resources, but rather to ensure that the game stays entertaining and to avoid that Player 1 chooses a rule which at some point rejects all cards, which – as we observed during play of this game – actually happens quite frequently with inexperienced players.

7.2 Conclusion

Let us now conclude what we have achieved with respect to answering our four research questions.

> **Research Question 1** *What formal frameworks are best suited for reasoning about which concepts involved in interaction?*
>
> - *What should be the primitive notions a formal approach should be based on?*

Part I of this dissertation has addressed this question by focusing on modal logic frameworks for reasoning about strategic abilities of individuals and groups. We have shown that an explicit representation of agents' preferences and actions by which results can be achieved can have conceptual benefits but can also lead to high complexity of the resulting logical system, depending on the chosen underlying logic of actions.

In Chapter 3, we presented a systematic study of different modal logic frameworks for coalitional interaction. A model theoretical study of the invariance of different game theoretical properties on modal logic models allowed us to draw conclusions as to what kind of approaches are best suited for reasoning about which kind of notions.

In particular, we have seen that attention has to be payed to the difference between strict and weak preferences in stability notions that one wants to reason about. We have seen that e.g. for Nash-stability, on action- and power-based models its weak version is easier to express than the strong one while for coalition-labeled transition systems the situation is just the opposite.

> **Research Question 2** *What is the role of cooperation vs. competition in the complexity of interaction?*
>
> - *Does analyzing an interactive situation in general become easier if the participants cooperate?*

Chapter 4 has addressed this question for a class of games which represent the travel through a network with connection failures. We have shown that non-cooperative versions of this game are of much higher complexity than a cooperative version. Moreover, changing the objectives of the players in the non-cooperative case does not have any influence on the complexity as long as the situation stays non-cooperative. Thus, while an analysis of interactive situations with respect to strategic abilities seems to be easier with cooperation, we also note that for modal logical frameworks for reasoning about coalitions, sometimes the opposite effect can be observed. To be more precise, our work in Chapters 2 and 3 has shown that if such systems are based on models in which individuals rather than coalitions are taken as primitive notions, then an exponential blow-up can occur when formalizing the ability of groups.

> **Research Question 3** *Which parameters can make interaction difficult?*
>
> - *How does the complexity of an interactive situation change when more participants enter the interaction or when we drop some simplifying assumptions on the participants themselves?*

We have addressed this question by focusing on structures representing the information that agents have. In particular we analyzed the complexity involved in *comparing* and *manipulating* information structures as modeled by modal logic. For the complexity of comparing information of agents, which in general is tractable, we can conclude that under the assumptions of knowledge being truthful and fully introspective, a complexity jump occurs with the introduction of a second agent. Without any particular assumptions on knowledge, more agents entering the situation does not significantly increase the complexity.

For the complexity of manipulating the information of agents in a certain way, we have shown that as long as we only consider one individual whose knowledge is truthful and fully introspective, this is easy. Dropping the assumptions however makes this an intractable task.

> **Research Question 4** *Finally, to what extent can we use a formal analysis of interactive processes to draw conclusions about the complexity of actual interaction?*
>
> - *Are there concrete examples of interactions in which participants actually encounter very high complexities which make it impossible for them to act?*

While a complexity theoretical study of whole logical systems does not seem to necessarily have implications for real interaction, a complexity theoretical study of the *tasks* involved in interaction is more promising with respect to implications for real interaction. Chapter 6 gave a case study of the card game Eleusis and has shown that in general it cannot be taken for granted that recreational games are playable in the sense that players should always be able to find a legal move without facing any unsolvable problems. This leads us to the conclusion that in the design of recreational games careful attention has to be paid to the problem of deciding what are legal moves in a game, as this is a problem that players face during the play, even without more sophisticated strategical considerations.

Additionally, the complexity study of concrete tasks that interacting individuals face also has the benefit that such tasks can also be investigated empirically, which can then lead to the development of new measures for the *cognitive* complexity of such tasks.

In general, the work in this dissertation shows that tasks and problems about and involved in interaction cover the whole range of the complexity hierarchy. Moving from satisfiability of extended modal logic frameworks to concrete tasks in playing actual games does not necessarily imply a decrease in computational complexity. In general, we have seen that there is a need for more game-specific characterizations of different kinds of interactive processes according to their complexity.

7.3 Further Work

Our work gave rise to some interesting further questions to investigate.

7.3.1 New questions for modal logics for reasoning about interaction

Our analysis of different modal logic frameworks for reasoning about interaction opened some interesting questions for further research of modal logic with particular focus on complexity and game-like interaction.

Complexity of logics for multi-agent systems. The methodology we used in Chapter 3 of systematically checking the invariance of interesting properties on different kinds of models for determining how much expressive power is needed to express them could be applied also to modal logic approaches for reasoning about other kinds of concepts in multi-agent systems. This could shed some light onto the landscape of modal logics developed for multi-agent systems.

Additionally, our work raised the question of how lower bounds could be obtained. To be more precise, this calls for a method of showing that for a given property and a given class of models *every* modal logic (with some reasonable properties) that is able to express this property on the class of models will have at least a certain complexity.

Extensions of sabotage-style logics. In Chapter 4, we gave complexity results for some variations of Sabotage Games. The variations were originally conceptually motivated by an interactive view on learning scenarios, focusing on the interaction between Learner and Teacher. One of the key properties of the games are the different roles of the players; one acting locally and the other acting globally. We could also look at interesting game variations in which the roles of the players are different and then investigate what will be the effect on the corresponding logic. A particularly interesting variation would be the game in which Blocker also moves locally and removes edges by moving along them. This version of a Sabotage Game could then be compared to the game *Pacman* (cf. Heckel (2006)). A corresponding modal logic could then be developed with a modality with the following semantics.

$$\mathcal{M}, w \models \Diamond_{Pacman}\varphi \quad \text{iff } \exists v \text{ with } (w,v) \in R_a \text{ for some } a \in \Sigma \text{ and } \mathcal{M}^{-(w,v),a}, v \models \varphi.$$

Thus, the Pacman-modality is a local dynamic modality, saying that it is possible to move to a successor state while erasing that transition such that at that state φ is true.

Similarly, another variation on the game inspired by other games would be to introduce imperfect information so that e.g. Blocker does not always know the exact location of Runner. This would lead to a game closer connected to the game of Scotland Yard, and accordingly to the question as to whether for Sabotage Games it is also the case that both the version with perfect information and that with imperfect information are of the same complexity (PSPACE), as it is the case for Scotland Yard (Sevenster 2006).

We have shown that allowing Blocker to refrain from removing an edge does not change the abilities of the players with respect to whether they can win. Additionally, we could consider a version in which players are allowed to make several moves in a row. In the logic, this would then lead to adding a Kleene star operation for the diamond for Runner and/or the sabotage diamond for Blocker. This then leads to the question as to what is the effect on the complexity of the associated games and the extended Sabotage Modal Logic.

Modal logic frameworks for epistemic interaction. Taking a semantic and agent-oriented perspective in Chapter 5 has led us to the investigation of various tasks about comparing information structures of agents. This has led us to an investigation of the similarity notion of flipped-bisimulation. Taking this a step further would lead to other similarity notions that could be motivated by a more internal agent-oriented perspective. One way to go into this direction would be to explore both the model theoretical and the complexity theoretical properties of weaker notions of similarity such as those underlying analogical reasoning, on the domain of epistemic reasoning.

We have seen that both for static and dynamic tasks on information structures, increasing the number of agents involved or dropping particular assumptions on the epistemic accessibility relations (such as reflexivity) can cause a complexity jump of the task under consideration. This leads us to the question as to how far we can characterize the epistemic modal logics in which these tasks are of certain complexity classes. As a first step, we suggest a careful investigation of the problems for logics between K and S5.

7.3.2 New questions for complexity theory

The complexity analyses in this dissertation specifically focused on the complexity that arises in interactive processes. The problems and tasks we investigated were motivated by their role in the interaction between (groups of) agents.

More generally, our work also leads to some new paths to be explored in the complexity analysis of graph theoretical problems. While for graph isomorphism many variations have been investigated, much less work has been done for problems involving graph bisimilarity. In particular, our work gave rise to the question as to which special cases of the NP-complete problem of induced subgraph bisimulation can be shown to be tractable.

7.3.3 New questions for artificial intelligence in games

Our analysis of tasks in reasoning about agents' information structures in Chapter 5 is relevant for game AI for games that simulate social interaction (cf. Chapter 4 of Witzel (2009)).

Our analysis of the game Eleusis leads to new challenges for AI for tackling difficult tasks in determining what are legal moves, a task that is usually straightforward.

7.3.4 New questions for cognitive science

Switching from an external perspective on interaction to a more internal perspective in which we investigated reasoning about the information of agents, our work in Chapter 5 naturally calls for an empirical investigation as to whether the borders of certain difficulty levels in actual reasoning about information correspond to our complexity findings.

In Chapter 6, we showed that a complexity theoretical study of interaction can have some implications for real interaction. This was done for the game Eleusis. In order to determine the precise impact of the complexity for actual play of humans, some more work has to be done. As a first step for gathering more insight into actual play of the game, data has to be gathered as e.g. in Sangati (2011) in order to get a better idea of cognitive difficulties involved in the game. Moreover, different strategies of inductive inference in practice can be investigated this way.

Throughout this dissertation we have moved from a complexity analysis of abstract general frameworks to a complexity analysis of tasks that players face during play of a game. A natural next step would be to take this further and focus on subtasks involved here. A complexity theoretical analysis of reasoning tasks involved in game playing can then also contribute to a theoretical foundation underlying the design of games for teaching and training certain skills such as performing arithmetic operations (Klinkenberg et al. 2011). Based on a computational complexity analysis such tasks can be classified at a high level.

Bibliography

R. Abbott. The new Eleusis. Unpublished manuscript. Available from Box 1175, General Post Office, New York, N.Y. 10001, 1977. Cited on pages **166 and 167**.

T. Ågotnes, P. Dunne, W. van der Hoek, and M. Wooldridge. Logics for coalitional games. In J. van Benthem, S. Ju, and F. Veltman, editors, *A Meeting of the Minds*, number 8 in Texts in Computer Science, pages 3–20, London, UK, 2007. College Publications. Cited on page **29**.

R. Alur, T. A. Henzinger, and O. Kupferman. Alternating-time temporal logic. *Lecture Notes in Computer Science*, 1536:23–60, 1998. Cited on page **28**.

C. Areces, P. Blackburn, and M. Marx. Hybrid logics: characterization, interpolation and complexity. *The Journal of Symbolic Logic*, 66(3):977–1010, 2001. Cited on page **75**.

G. Aucher. An internal version of epistemic logic. *Studia Logica*, 94(1):1–22, 2010. Cited on page **132**.

R. J. Aumann. Interactive epistemology I: Knowledge. *International Journal of Game Theory*, 28(3):263–300, 1999. Cited on page **134**.

P. Balbiani, A. Baltag, H. van Ditmarsch, A. Herzig, T. Hoshi, and T. de Lima. 'knowable' as 'known after an announcement'. Technical Report IRIT/RR-2008-2-FR, IRIT, University of Toulouse 3, 2008a. Cited on page **143**.

P. Balbiani, O. Gasquet, A. Herzig, F. Schwarzentruber, and N. Troquard. Coalition games over kripke semantics: expressiveness and complexity. In C. Dégremont, L. Keiff, and H. Rückert, editors, *Essays in Honour of Shahid Rahman*. College Publications, London, 2008b. Cited on page **65**.

J. L. Balcázar, J. Gabarró, and M. Santha. Deciding bisimilarity is P-complete. *Formal Aspects of Computing*, 4(6A):638–648, 1992. Cited on pages **13 and 137**.

A. Baltag and L. S. Moss. Logics for epistemic programs. *Synthese*, 139(2): 165–224, 2004. Cited on page **132**.

N. Belnap, M. Perloff, and M. Xu. *Facing the future: Agents and choices in our indeterminist world*. Oxford University Press, Oxford, 2001. Cited on page **68**.

J. van Benthem. *Modal Correspondence Theory*. PhD thesis, Mathematisch Instituut & Instituut voor Grondslagenonderzoek, Universiteit van Amsterdam, 1976. Cited on pages **9 and 75**.

J. van Benthem. An essay on sabotage and obstruction. In *Mechanizing Mathematical Reasoning, Essays in Honor of Jörg H. Siekmann on the Occasion of His 60th Birthday*, pages 268–276, 2005. Cited on pages **21, 96, 103, 104, and 117**.

J. van Benthem. *Modal Logic for Open Minds*. Number 199 in CSLI lecture notes. Center for the Study of Language and Information, 2010. Cited on pages **2 and 135**.

J. van Benthem. *Logical Dynamics of Information Flow*. Cambridge University Press, 2011. Cited on page **135**.

J. van Benthem and E. Pacuit. The tree of knowledge in action: towards a common perspective. In I. H. G. Governatori and Y. Venema, editors, *Advances in Modal Logic*, volume 6. College Publications, 2006. Cited on page **132**.

J. van Benthem, S. van Otterloo, and O. Roy. Preference logic, conditionals and solution concepts in games. In *Festschrift for Krister Segerberg*. University of Uppsala, 2005. Cited on pages **29 and 33**.

J. van Benthem, O. Roy, and P. Girard. Everything else being equal: A modal logic approach to ceteris paribus preferences, 2007. Cited on pages **20, 29, 33, 34, 47, and 189**.

M. J. Berry. APL and the search for truth: A set of functions to play New Eleusis. In *APL '81: Proceedings of the international conference on APL*, pages 47–53, New York, NY, USA, 1981. ACM. Cited on page **166**.

T. R. Besold, H. Gust, U. Krumnack, A. Abdel-Fattah, M. Schmidt, and K.-U. Kühnberger. An argument for an analogical perspective on rationality and decision-making. In R. Verbrugge and J. van Eijck, editors, *Proceedings of the Workshop on Reasoning About Other Minds: Logical and Cognitive Perspectives*

(RAOM-2011), Groningen, The Netherlands, July 11th, 2011, volume 751 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2011. Cited on page **162**.

J. B. Best. The role of context on strategic actions in Mastermind. *Journal of General Psychology*, 127(2):165–77, 2000. Cited on page **171**.

P. Blackburn, M. de Rijke, and Y. Venema. *Modal Logic*. Number 53 in Cambridge Tracts in Theoretical Computer Science. Cambridge University Press, UK, 2001. Cited on pages **2, 34, 42, 75, 83, 134, and 135**.

S. Borgo. Coalitions in action logic. In *IJCAI '07: Proceedings of the twentieth international joint conference on Artificial Intelligence*, pages 1822–1827, Hyderabad, India, 2007. Cited on pages **29 and 59**.

J. Broersen. *Modal Action Logics for Reasoning about Reactive Systems*. PhD thesis, Faculteit der Exacte Wetenschappen, Vrije Universiteit Amsterdam, 2003. Cited on page **30**.

J. Broersen, A. Herzig, and N. Troquard. Normal Coalition Logic and its conformant extension. In D. Samet, editor, *TARK'07*, pages 91–101. PUL, 2007. Cited on pages **62, 65, 68, 70, 72, and 190**.

J. Broersen, A. Herzig, and N. Troquard. What groups do, can do, and know they can do: an analysis in normal modal logics. *Journal of Applied Non-classical Logics*, 19:261–290, 2009. Cited on pages **52 and 53**.

B. ten Cate. *Model theory for extended modal languages*. PhD thesis, University of Amsterdam, 2005. ILLC Dissertation Series DS-2005-01. Cited on page **75**.

B. ten Cate and M. Franceschet. On the complexity of hybrid logics with binders. In L. Ong, editor, *Proc. of CSL 2005*, volume 3634 of *LNCS*, pages 339–354. Springer, 2005. Cited on page **77**.

A. Church. An unsolvable problem of elementary number theory. *American Journal of Mathematics*, 58(2):345–363, 1936. Cited on page **12**.

A. Condon, J. Maňuch, and C. Thachuk. Complexity of a collision-aware string partition problem and its relation to oligo design for gene synthesis. In *Proceedings of the 14th annual international conference on Computing and Combinatorics*, COCOON '08, pages 265–275. Springer-Verlag, 2008. Cited on page **179**.

L. De Nardo, F. Ranzato, and F. Tapparo. The subgraph similarity problem. *IEEE Transactions on Knowledge and Data Engineering*, 21(5):748–749, 2009. ISSN 1041-4347. Cited on page **152**.

C. Dégremont. *The Temporal Mind. Observations on the logic of belief change in interactive systems*. PhD thesis, ILLC, Universiteit van Amsterdam, 2010. Cited on pages **54 and 79**.

C. Dégremont and L. Kurzen. Modal logics for preferences and cooperation: Expressivity and complexity. In J.-J. Meyer and J. Broersen, editors, *Knowledge Representation for Agents and Multi-Agent Systems*, volume 5605 of *Lecture Notes in Computer Science*, pages 32–50. Springer, 2009a. Cited on pages **22, 48, 54, and 79**.

C. Dégremont and L. Kurzen. Getting together: A unified perspective on modal logics for coalitional interaction. In X. He, J. F. Horty, and E. Pacuit, editors, *Logic, Rationality, and Interaction, Second International Workshop, LORI 2009, Proceedings*, volume 5834 of *Lecture Notes in Computer Science*, pages 317–318. Springer, 2009b. Cited on page **23**.

C. Dégremont and L. Kurzen. Cooperation and stability in modal logics: Comparing frameworks and determining descriptive difficulty. In *9th Conference on Logic and the Foundations of Game and Decision Theory (LOFT)*, 2010. `http://loft2010.csc.liv.ac.uk/papers/60.pdf`. Cited on page **23**.

C. Dégremont, L. Kurzen, and J. Szymanik. On the tractability of comparing informational structures. In R. Verbrugge and J. van Eijck, editors, *Proceedings of the Workshop on Reasoning About Other Minds: Logical and Cognitive Perspectives (RAOM-2011), Groningen, The Netherlands, July 11th, 2011*, volume 751 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2011. Cited on page **23**.

T. G. Diettrich and R. S. Michalski. Learning to predict sequences. In *Machine Learning: An Artificial Intelligence Approach*, volume II. Morgan Kaufmann, 1989. Cited on page **166**.

H. van Ditmarsch and T. French. Simulation and information: Quantifying over epistemic events. In *Knowledge Representation for Agents and Multi-Agent Systems: First International Workshop, KRAMAS 2008, Sydney, Australia, September 17, 2008, Revised Selected Papers*, pages 51–65, Berlin, Heidelberg, 2009. Springer-Verlag. Cited on pages **134 and 143**.

H. van Ditmarsch, W. van der Hoek, and B. Kooi. *Dynamic Epistemic Logic*. Springer Netherlands, 2007. Cited on page **135**.

F. M. Donini, M. Lenzerini, D. Nardi, and W. Nutt. The complexity of concept languages. In *KR*, pages 151–162, 1991. Cited on page **77**.

A. Dovier and C. Piazza. The subgraph bisimulation problem. *IEEE Transactions on Knowledge and Data Engineering*, 15(4):1055–1056, 2003. Cited on pages **143 and 144**.

P. E. Dunne, W. van der Hoek, and M. Wooldridge. A logical characterisation of qualitative coalitional games. *Journal of Applied Non-classical Logics*, 17: 477–509, 2007. Cited on page **53**.

T. Egawa, Y. Kiriha, and A. Arutaki. Tackling the complexity of future networks. In *Proceedings of the 6th IFIP TC6 international working conference on Active networks*, IWAN'04, pages 78–87, Berlin, Heidelberg, 2007. Springer-Verlag. Cited on page **1**.

P. van Emde Boas. Machine models and simulation. In *Handbook of Theoretical Computer Science, Volume A: Algorithms and Complexity*, pages 1–66. MIT Press, 1990. Cited on page **12**.

U. Endriss, N. Maudet, F. Sadri, and F. Toni. Negotiating socially optimal allocations of resources. *Journal of Artificial Intelligence Research*, 25:315–348, 2006. Cited on page **56**.

R. Fagin, J. Y. Halpern, Y. Moses, and M. Y. Vardi. *Reasoning About Knowledge*. MIT Press, Cambridge, 1995. ISBN 0262061627. Cited on pages **132 and 134**.

S. Feferman. Persistent and invariant formulas for outer extensions. *Compositio Mathematica*, 20:29–52, 1969. Cited on page **75**.

N. Feltovich. Reinforcement-based vs. beliefs-based learning in experimental asymmetric-information games. *Econometrica*, 68:605–641, 2000. Cited on page **133**.

M. J. Fischer and R. E. Ladner. Propositional Dynamic Logic of Regular Programs. *Journal of Computer and System Sciences*, 18(2):194–211, 1979a. Cited on page **57**.

M. J. Fischer and R. E. Ladner. Propositional dynamic logic of regular programs. *J. Comput. Syst. Sci.*, 1979b. Cited on page **77**.

F. Fomin, P. Golovach, and J. Kratochvíl. On tractability of cops and robbers game. In G. Ausiello, J. Karhumäki, G. Mauri, and L. Ong, editors, *Fifth Ifip International Conference On Theoretical Computer Science – Tcs 2008*, volume 273 of *IFIP International Federation for Information Processing*, pages 171–185. Springer, 2008. Cited on page **105**.

M. Franceschet and M. de Rijke. Model checking for hybrid logics. In *Proceedings of the Workshop Methods for Modalities*, 2003. Cited on page **77**.

T. French and H. van Ditmarsch. Undecidability for arbitrary public announcement logic. In C. Areces and R. Goldblatt, editors, *Advances in Modal Logic*, pages 23–42. College Publications, 2008. Cited on page **143**.

M. Friedewald and O. Raabe. Ubiquitous computing: An overview of technology impacts. *Telematics and Informatics*, 28(2):55–65, 2011. Cited on page **1**.

D. Gabbay. *Fibring Logics*, volume 38 of *Oxford Logic Guides*. Oxford Univesity Press, 1998. Cited on page **49**.

D. Gabbay. Introducing reactive kripke semantics and arc accessibility. In A. Avron, N. Dershowitz, and A. Rabinovich, editors, *Pillars of computer science*, volume 4800 of *Lecture Notes in Computer Science*, pages 292–341. Springer-Verlag, Berlin, Heidelberg, 2008. Cited on page **96**.

M. Gardner. On playing New Eleusis, the game that simulates the search for truth. *Scientific American*, 237:18–25, 1977. Cited on pages **166 and 167**.

M. R. Garey and D. S. Johnson. *Computers and Intractability; A Guide to the Theory of NP-Completeness*. W. H. Freeman & Co., New York, NY, USA, 1990. Cited on pages **13, 135, and 136**.

N. Gierasimczuk. *Knowing One's Limits. Logical Analysis of Inductive Inference*. PhD thesis, ILLC, Universiteit van Amsterdam, 2010. Cited on pages **96, 106, 184, and 188**.

N. Gierasimczuk and D. de Jongh. On the minimality of definite tell-tale sets in finite identification of languages, 2010. `http://staff.science.uva.nl/~dickdj/submissiom_COLT10_NGDdJ.pdf`. Cited on pages **127 and 170**.

N. Gierasimczuk and J. Szymanik. A note on a generalization of the muddy children puzzle. In K. R. Apt, editor, *Proceedings of the 13th Conference on Theoretical Aspects of Rationality and Knowledge (TARK-2011), Groningen, The Netherlands, July 12–14, 2011*, pages 257–264. ACM, 2011. Cited on page **132**.

N. Gierasimczuk, L. Kurzen, and F. R. Velázquez-Quesada. Games for learning: A sabotage approach. In M. Baldoni, C. Baroglio, J. Bentahar, G. Boella, M. Cossentino, M. Dastani, B. Dunin-Keplicz, G. Fortino, M. P. Gleizes, J. Leite, V. Mascardi, J. A. Padget, J. Pavón, A. Polleres, A. E. Fallah-Seghrouchni, P. Torroni, and R. Verbrugge, editors, *Proceedings of the Second Multi-Agent Logics, Languages, and Organisations Federated Workshops (MALLOW), Turin, Italy, September 7-10, 2009*, volume 494 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2009a. Cited on page **23**.

N. Gierasimczuk, L. Kurzen, and F. R. Velázquez-Quesada. Learning and teaching as a game: A sabotage approach. In X. He, J. F. Horty, and E. Pacuit, editors, *Logic, Rationality, and Interaction, Second International Workshop, LORI 2009, Proceedings*, volume 5834 of *Lecture Notes in Computer Science*, pages 119–132. Springer, 2009b. Cited on pages **23, 96, 105, 106, 108, and 188**.

P. Girard. *Modal Logic for Preference Change*. PhD thesis, Stanford University, 2008. Cited on pages **28 and 52**.

E. Gold. Language identification in the limit. *Information and Control*, 10:447–474, 1967. Cited on page **20**.

J. Golden. Eleusis Express, September 2011. `http://www.logicmazes.com/games/eleusis/express.html`. Cited on pages **166, 167, 170, and 186**.

V. Goranko. Coalition games and alternating temporal logics. *TARK: Theoretical Aspects of Reasoning about Knowledge*, 8, 2001. Cited on page **52**.

V. Goranko and W. Jamroga. Comparing semantics of logics for multi-agent systems. *Synthese*, 139:241–280, 2004. Cited on page **52**.

V. Goranko, W. Jamroga, and P. Turrini:. Strategic games and truly playable effectivity functions. In *Proceedings of the 10th International Conference on Autonomous Agents and Multi-Agent Systems (AAMAS 2011), Taipei, Taiwan, May 2–6, 2011*, pages 727–734, 2011. Cited on page **40**.

J. Y. Halpern and Y. Moses. A guide to completeness and complexity for modal logics of knowledge and belief. *Artificial Intelligence*, 54:319–379, April 1992. Cited on pages **137 and 151**.

J. Y. Halpern and M. Y. Vardi. The complexity of reasoning about knowledge and time. I. Lower bounds. *Journal of Computer and Systems Science*, 38(1): 195–237, 1989. Cited on page **132**.

H. H. Hansen, C. Kupke, and E. Pacuit. Neighbourhood structures: Bisimilarity and basic model theory. *Logical Methods in Computer Science*, 5(2), 2009. Cited on page **7**.

D. Harel. Dynamic logic. In D. Gabbay and F. Guenther, editors, *Handbook of Philosophical Logic Volume II – Extensions of Classical Logic*, pages 497–604. D. Reidel Publishing Company, Dordrecht, The Netherlands, 1984. Cited on page **57**.

R. Heckel. Graph transformation in a nutshell. *Electronic Notes in Theoretical Computer Science*, 148(1):187 – 198, 2006. Proceedings of the School of SegraVis Research Training Network on Foundations of Visual Modelling Techniques (FoVMT 2004). Cited on pages **128 and 196**.

M. R. Henzinger, T. A. Henzinger, and P. W. Kopke. Computing simulations on finite and infinite graphs. In *FOCS '95: Proceedings of the 36th Annual Symposium on Foundations of Computer Science*, pages 453–462. IEEE Computer Society Press, 1995. Cited on page **153**.

A. Herzig and E. Lorini. A dynamic logic of agency i: Stit, capabilities and powers. *Journal of Logic, Language and Information*, 19(1):89–121, 2010. Cited on page **54**.

C. M. Hoffmann. *Group-Theoretic Algorithms and Graph Isomorphism*, volume 136 of *Lecture Notes in Computer Science*. Springer-Verlag, 1982. Cited on page **136**.

J. F. Horty and N. Belnap. The deliberative stit: A study of action, omission, ability, and obligation. *Journal of Philosophical Logic*, 24(6):583 – 644, 1995. Cited on page **52**.

D. Klein, F. G. Radmacher, and W. Thomas. Moving in a network under random failures: A complexity analysis. *Science of Computer Programming*, 2010. Cited on pages **96, 107, and 109**.

S. Klinkenberg, M. Straatemeier, and H. van der Maas. Computer adaptive practice of maths ability using a new item response model for on the fly ability and difficulty estimation. *Computers & Education*, 57(2):1813 – 1824, 2011. Cited on page **198**.

D. E. Knuth. The computer as mastermind. *Journal of Recreational Mathematics*, 9:1–6, 1976. Cited on page **171**.

B. Kooi. Yet another mastermind strategy. *ICGA Journal*, 28(1):13–20, 2005. Cited on page **171**.

B. Kooi and J. van Benthem. Reduction axioms for epistemic actions. In R. Schmidt, I. Pratt-Hartmann, M. Reynolds, and H. Wansing, editors, *Advances in Modal Logic 2004*, pages 197–211. Department of Computer Science, University of Manchester, 2004. Cited on page **143**.

S. Kreutzer. Graph searching games. In K. R. Apt and E. Grädel, editors, *Lectures in Game Theory for Computer Scientists*, pages 213–263. Springer, 2011. Cited on page **105**.

A. Kučera and R. Mayr. Why is simulation harder than bisimulation? In *CONCUR '02: Proceedings of the 13th International Conference on Concurrency Theory*, pages 594–610, London, UK, 2002. Springer-Verlag. Cited on page **162**.

L. Kurzen. Logics for Cooperation, Actions and Preferences. Master's thesis, Universiteit van Amsterdam, the Netherlands, 2007. Cited on pages **22, 33, 36, and 38**.

L. Kurzen. Reasoning about cooperation, actions and preferences. *Synthese*, 169(2):223 – 240, 2009. Cited on page **22**.

L. Kurzen. Eleusis: Complexity and interaction in inductive inference. In *Proceedings of the Second ILCLI International Workshop on Logic and Philosophy of Knowledge, Communication and Action, LogKCA-10, Donostia - San Sebastián, Spain, November 3-5, 2010*, pages 287 – 303. The University of the Basque Country Press, 2010. Cited on page **23**.

R. E. Ladner. The computational complexity of provability in systems of modal propositional logic. *SIAM J. Comput.*, 6(3):467–480, 1977. Cited on pages **13 and 77**.

M. Lange. Model checking pdl with all extras. *J. Applied Logic*, 4(1):39–49, 2006. Cited on page **77**.

C. Löding and P. Rohde. Solving the sabotage game is PSPACE-hard. In *Proceedings of the 28th International Symposium on Mathematical Foundations of Computer Science, MFCS '03*, volume 2474 of *LNCS*, pages 531–540. Springer, 2003a. Cited on page **104**.

C. Löding and P. Rohde. Solving the sabotage game is PSPACE-hard. Technical report, Aachener Informatik Berichte, Rwth Aachen, 2003b. Cited on pages **104, 106, 107, and 117**.

A. Looney, J. Cooper, K. Heath, J. Davenport, and K. Looney. *Playing with Pyramides*. Looneylabs, 1997. Cited on pages **166, 171, and 186**.

E. Lorini, F. Schwarzentruber, and A. Herzig. Epistemic games in modal logic: Joint actions, knowledge and preferences all together. In X. He, J. Horty, and E. Pacuit, editors, *Logic, Rationality, and Interaction*, volume 5834 of *Lecture Notes in Computer Science*, pages 212–226. Springer, 2009. Cited on page **54**.

C. Lutz and U. Sattler. The complexity of reasoning with boolean modal logics. In F. Wolter, H. Wansing, M. de Rijke, and M. Zakharyaschev, editors, *Advances in Modal Logics Volume 3*. CSLI Publications, Stanford, 2001. Cited on pages **45 and 46**.

C. Lutz, U. Sattler, and F. Wolter. Modal logics and the two-variable fragment. In *Annual Conference of the European Association for Computer Science Logic CSL'01*, LNCS, Paris, France, 2001. Springer Verlag. Cited on pages **45 and 46**.

C. Magerkurth, A. D. Cheok, R. L. Mandryk, and T. Nilsen. Pervasive games: bringing computer entertainment back to the real world. *Comput. Entertain.*, 3(3):4, July 2005. Cited on page **129**.

D. Matuszek. New eleusis, 1995. `http://matuszek.org/eleusis1.html`. Cited on pages **167 and 174**.

R. S. Michalski, H. Ko, and K. Chen. SPARC/E(V.2): An Eleusis rule generator and player. Report ISG85-11, UIUCDC-F-85941, Department of Computer Science, University of Illinois, 1985. Cited on page **166**.

Ministry of Public Management, Home Affairs, Posts and Telecommunications, Japan. Information and Communications in Japan, 2004. 2004 White Paper. Cited on page **1**.

Y. Mukouchi. Characterization of finite identification. In *AII '92: Proceedings of the International Workshop on Analogical and Inductive Inference*, pages 260–267, London, UK, 1992. Springer-Verlag. Cited on page **171**.

M. J. Osborne and A. Rubinstein. *A Course in Game Theory*. MIT Press, 1994. Cited on pages **11, 15, 37, and 134**.

S. van Otterloo, W. van der Hoek, and M. Wooldridge. Preferences in game logics. In *AAMAS '04: Proceedings of the Third International Joint Conference on Autonomous Agents and Multiagent Systems*, pages 152–159. IEEE Computer Society, 2004. Cited on page **29**.

C. H. Papadimitriou. Games against nature. *Journal of Computer and System Sciences*, 31(2):288–301, 1985. Cited on page **109**.

C. H. Papadimitriou. *Computational Complexity*. Addison Wesley, 1994. Cited on pages **10 and 115**.

C. H. Papadimitriou and K. Steiglitz. *Combinatorial optimization: algorithms and complexity*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1982. Cited on page **147**.

M. Pauly. A modal logic for coalitional power in games. *Journal of Logic and Computation*, 12(1):149–166, 2002a. Cited on pages **28, 29, 31, 52, 68, and 190**.

M. Pauly. On the complexity of coalitional reasoning. *International Game Theory Review*, 4:237–254, 2002b. Cited on page **40**.

E. L. Post. A variant of a recursively unsolvable problem. *Bull. Amer. Math. Soc.*, 52:264–268, 1946. Cited on page **182**.

I. Pratt-Hartmann and L. S. Moss. Logics for the relational syllogistic. *The Review of Symbolic Logic*, 2(04):647–683, 2009. Cited on page **132**.

M. E. J. Raijmakers, S. van Es, and M. Counihan. Children's strategy use in playing strategic games. In R. Verbrugge and J. van Eijck, editors, *Proceedings of the Workshop on Reasoning About Other Minds: Logical and Cognitive Perspectives (RAOM-2011), Groningen, The Netherlands, July 11th, 2011*, volume 751 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2011. Cited on page **131**.

P. Rohde. *On games and logics over dynamically changing structures*. PhD thesis, RWTH Aachen, 2006. Cited on pages **96, 97, 107, and 110**.

C. H. Romesburg. Simulating scientific inquiry with the card game eleusis. *Science Edulation*, 5(63):599–608, 1978. Cited on page **166**.

S. L. Roux, P. Lescanne, and R. Vestergaard. Conversion/preference games. *CoRR*, 2008. Cited on pages **52 and 55**.

K. Ruohonen. On some variants of post's correspondence problem. *Acta Informatica*, 19:357–367, 1983. Cited on page **182**.

F. Sangati. Eleusis game, 2011. URL `http://www.eleusisgame.org`. Cited on pages **23, 188, and 198**.

L. Sauro, J. Gerbrandy, W. van der Hoek, and M. Wooldridge. Reasoning about action and cooperation. In *AAMAS '06: Proceedings of the fifth International Joint Conference on Autonomous Agents and Multi-agent Systems*, pages 185–192, Hakodate, Japan, 2006. Cited on pages **20, 29, 30, 32, 33, 37, 47, and 189**.

W. J. Savitch. Relationships between nondeterministic and deterministic tape complexities. *Journal of Computer and System Sciences*, 4(2):177 – 192, 1970. Cited on page **13**.

M. P. D. Schadd. *Selective Search in Games of Different Complexity*. PhD thesis, Universiteit Maastricht, 2011. Cited on page **188**.

F. Schwarzentruber. Décidabilité et complexité de la logique normale des coalitions. Master's thesis, IRIT, Université Paul Sabatier, Toulouse, 2007. Cited on pages **65 and 76**.

K. Segerberg. Bringing it about. *Journal of Philosophical Logic*, 18(4):327–347, 1989. Cited on page **55**.

M. Sevenster. *Branches of imperfect information: logic, games, and computation*. PhD thesis, ILLC, Universiteit van Amsterdam, 2006. Cited on pages **105 and 197**.

J. Stuckman and G. Zhang. Mastermind is NP-complete. *INFOCOMP Journal of Computer Science*, 5:25–28, 2006. Cited on pages **171, 173, and 177**.

T. Sudkamp. *Languages and machines - an introduction to the theory of computer science*. Addison-Wesley series in computer science. Addison-Wesley, 1988. Cited on page **172**.

J. Szymanik. Computational complexity of polyadic lifts of generalized quantifiers in natural language. *Linguistics and Philosophy*, 33:215–250, 2010. ISSN 0165-0157. Cited on page **132**.

J. Szymanik and M. Zajenkowski. Comprehension of simple quantifiers. Empirical evaluation of a computational model. *Cognitive Science: A Multidisciplinary Journal*, 34(3):521–532, 2010. Cited on page **133**.

N. Troquard, W. van der Hoek, and M. Wooldridge. A logic of games and propositional control. In *Proceedings of The 8th International Conference on Autonomous Agents and Multiagent Systems - Volume 2*, AAMAS '09, pages 961–968, Richland, SC, 2009. International Foundation for Autonomous Agents and Multiagent Systems. Cited on page **54**.

A. M. Turing. On Computable Numbers, with an application to the Entscheidungsproblem. *Proc. London Math. Soc.*, 2(42):230–265, 1936. Cited on page **12**.

R. Verbrugge. Logic and social cognition. The facts matter, and so do computational models. *Journal of Philosophical Logic*, 38(6):649–680, 2009. Cited on page **133**.

R. Verbrugge and L. Mol. Learning to apply theory of mind. *Journal of Logic, Language and Information*, 17(4):489–511, 2008. Cited on page **171**.

D. Walther, W. van der Hoek, and M. Wooldridge. Alternating-time temporal logic with explicit strategies. In *TARK '07: Proceedings of the 11th conference on Theoretical aspects of rationality and knowledge*, pages 269–278. ACM, 2007. Cited on pages **29 and 59**.

Y. Wang. *Epistemic Modelling and Protocol Dynamics*. PhD thesis, Universiteit van Amsterdam, 2010. Cited on pages **132 and 138**.

R. Weber. Behavior and learning the "dirty faces" game. *Experimental Economics*, 4:229–242, 2001. Cited on page **133**.

A. Witzel. *Knowledge and Games: Theory and Implementation*. PhD thesis, Universiteit van Amsterdam, 2009. Cited on page **198**.

M. Wooldridge and P. E. Dunne. On the computational complexity of qualitative coalitional games. *Artificial Intelligence*, 158:2004, 2004. Cited on page **52**.

Index

Symbols

$\bigwedge \Phi(A, C)$ 33

$\mathcal{H}_{\mathcal{L}_{\mathtt{ABC}}}(\downarrow)$ 61

$\mathcal{H}_{\mathcal{L}_{\mathtt{ABC}}}(\downarrow, \cap)$ 61

$\mathcal{H}_{\mathcal{L}_{\mathtt{ABC}}}(\downarrow, {}^{-1})$ 61

$\mathcal{H}_{\mathcal{L}_{\mathtt{ABC}}}(\mathbf{@}, \downarrow, \cap)$ 61

$\mathcal{H}_{\mathcal{L}_{\mathtt{LTS}}}(\downarrow)$ 57

$\mathcal{H}_{\mathcal{L}_{\mathtt{PBC}}}({}^{-1}, \downarrow)$ 66

$\mathcal{H}_{\mathcal{L}_{\mathtt{PBC}}}(\downarrow)$ 66

$\mathcal{H}_{\mathcal{L}_{\mathtt{PBC}}}(\mathbf{@}, \downarrow)$ 66

$\mathbf{K}_m^{\neg\cup}$ 45

$\mathcal{SG}_*^{Reach}$ 122

$\mathcal{SG}_*$ 121

Λ^{CLA+P} 38

$\mathcal{L}_{\mathtt{ABC}}$ 60

$\mathcal{L}_{\mathtt{ABC}}(\cap)$ 61

$\mathcal{L}_{\mathtt{LTS}}$ 57

$\mathcal{L}_{\mathtt{LTS}}(\cap)$ 57

$\mathcal{L}_{\mathtt{PBC}}$ 66

MC *see* model checking problem

$\mathcal{M}^{min\_cells}$ 148

SAT *see* satisfiability problem

$\alpha \subseteq \preceq_i$ 35

$\alpha \subseteq \prec_i$ 35

$\underline{\leftrightarrow}$ *see* bisimulation

$\underline{\leftrightarrow}_{total}$ *see* total bisimulation

$\xrightarrow{C}$ 55

$Dom(\mathcal{M})$ *see* domain of a model

$|\mathcal{M}|$ *see* size of a model

$|s|$ 174

$\mathcal{L}_{cla+p}$ *see* language of CLA+P

ρ *see* secret Eleusis rules

ρ_{Post} 183

$\sqsubseteq$ *see* simulation

$\sqsubseteq_{total}$ *see* total simulation

$\lhd$ 35

$\mathcal{SG}^{Reach}$ 106

$\#$ 111

$\mathcal{SG}^{Coop}$ 106

$\mathcal{SG}_*^{Coop}$ 122

$\mathcal{SG}^{Safety}$ 106

$\mathcal{SG}_*^{Safety}$ 122

A

`ABC`
- axiomatization 62
- language 60
- models 59
  - $\mathtt{ABC}^{\mathtt{NR}}$ 60
  - `N`-determined 60
  - reactive 60

agents model 31

APAL *see* arbitrary public announcement logic

arbitrary public announcement logic 143

B

bisimilarity
- flipped 139
  - horizon. 141
- horizon 141

bisimilarity problems
- epistemic models. 136
- global multi-agent S5 submodel . 150
- global single-agent S5 submodel. 147
- induced subgraph 144
- local single-agent S5 submodel . . 146
- submodel for Kripke models 145

bisimulation 9
- flipped 139

breaks. 121

BW(*s*) . . *see* set of black (red) words

$\mathcal{BW}(s)$ . . . *see* multiset of black (red) words

C

CA-SP. . . *see* Collision-Aware String Partition Problem

Card. 173, 181

CCC . . . . *see* condition about complementary coalitions

Church-Turing Thesis 12

CL model 29

CM . . . . . *see* coalition monotonicity

coalition logic for actions 31

coalition monotonicity. 63, 68

coalitional ability to make an agent happy 79

Collision-Aware String Partition Problem 179

color. 181

common knowledge 138

Condition about complementary coalitions 68

conversion/preference games. . . . 55

cooperation logic with actions (CLA) 32

cooperation logic with actions and preferences (CLA+P). . . . 38

cops and robbers 105

D

decision problems 10

deck of cards. 167

dictator. 39

directed multi-graph 97
- labeled. 98

domain of a model. 2

E

Eleusis
- rules of the game 167

Eleusis rules 174

environment language with preferences 35

environment logic with preferences . 36

environment model. 30

environment model with preferences 35

ESAT() 177

exact power
- in ABC. 59
- in PBC. 68

EXPTIME. 14

F

F 97, 99

flipped horizon bisimilarity 141

force 67

G

GI 136

Global multi-agent S5 submodel bisimulation 150

graph isomorphism problem . . . 136

H

happy *see* coalitional ability to make an agent happy
history-free determinacy property . . 100
horizon 135
  bisimilarity 141

I

IC . . . *see* independence of coalitions
independence of coalitions 68
Induced subgraph bisimulation . 144
Induced subgraph simulation . . 152
  equivalence 152
inductive inference games 166
introspection
  negative 134
  positive 134
invariance thesis 12

K

K_i 134
Kripke model isomorphism 136
Kripke models
  basic Kripke model 2
  multi-agent Kripke model 3
  pointed model 8

L

L (LOGSPACE) 12
Labeled Sabotage Game 99
  complexity
    $\mathcal{SG}^{Coop}$ 115
    $\mathcal{SG}^{Reach}$ 107
    $\mathcal{SG}^{Safety}$ 107
  decision problem for $\mathcal{SG}^{Coop}$. 107
  decision problem for $\mathcal{SG}^{Reach}$ 107
  decision problem for $\mathcal{SG}^{Safety}$ 107
  variations 106
  without strict alternation . . . 121
language of CLA 32
language of CLA+P 37

Local single-agent S5 submodel bisimulation 146
lookback 176
LTS
  axiomatization 58
  language 57
  models 55

M

mainline 168
Mastermind 171
matching pennies game 64
mechanism design 11
model checking problem 11
  combined complexity 11
  data complexity 11
model of Coalition Logic 29
multi-agent modal language 7
multi-agent modal logic $\mathbf{K}_{\mathtt{N}}$ 7
multi-agent system model 32
multi-agent system model
  with preferences 36
multiset of black words 182

N

NCL
  axiomatization 65
  complexity 65
  language 64
  models 62
NCL-Independence 63
NEXPTIME 14
NL 12
"no play" 168
NP 13
number of phases 175

P

P (PTIME) 13
Pacman 128, 129
PAL. *see* public announcement logic

PBC
- axiomatization 67
- language 66
- models 65

periodic Eleusis rules 175
- with lookback 176

phases of a secret Eleusis rule . . 175
playability 29
- true playability 40

positional winning strategy 100
Post's Correspondence Problem 182
PREF-ACT 36
preference language 34
preference logic 34
preference model 33
prefix-closed 174
Prophet 169
PSM 117
PSPACE 13
public announcement logic 143

Q

Quantified Boolean Formula (QBF) . . 107, 110

R

rational deal 56
Reachability problem 115
reception strategies 166
relational structures 2
resource allocation 56
$RW(s)$ . . *see* set of black (red) words
$\mathcal{RW}(s)$ . . . *see* multiset of black (red) words

S

Sabotage Game 97
- position 97

sabotage modal logic
- existence of a winning strategy . . 120
- $\mathcal{SG}^{Coop}$ 120
- $\mathcal{SG}^{Reach}$ 117
- $\mathcal{SG}^{Safety}$ 118
- language 104
- models 104
  - pointed 104
- semantics 104

safety winning condition 105
satisfiability problem 10
Scotland Yard 105
secret Eleusis rules ρ 174
selection strategies 166
set of black (red) words 181
set of black words 181
set of red words 181
$\mathfrak{SG}$ 99
SG 97
sideline 168
simulation 8
- total 8

size of a model 8
SML . . . . . . *see* sabotage modal logic
stability
- group 39
- individual 39
- Nash 81
- strong Nash 83

STRICT PREF-ACT 36
subgraph 135
- induced 135

submodel 9
- generated 9

Submodel bisimulation for Kripke models 145
supergraph of a directed labeled multi-graph 122

T

The New Eleusis 167
- rules about the secret rule . . 168

total bisimulation 9
total preorder 54

TPO . . . . *see* total preorder
trigger card 180
Turing machine. 12

U

unanimity 39

V

value(*s*) 181
veridicality 134

Z

Zendo. 171, 186

Samenvatting

Dit proefschrift geeft een formele analyse van de computationele complexiteit van interactie van agenten.

Omdat veel interactieve scenario's als netwerken of relationele structuren kunnen worden gerepresenteerd, kiezen we in dit proefschrift voor modale logica als gereedschap voor formalisatie.

Dit perspectief op interactie leidt ons tot een onderzoek naar de complexiteit van modale logica's voor multi-agentensystemen. Het bepalen van de complexiteit van interactie door middel van de complexiteit van het vervulbaarheidsprobleem (SAT) of het model checking probleem is een te ruime aanpak omdat het iets zegt over hoe moeilijk die beslissingsproblemen zijn voor willekeurige formules van een bepaalde logica en niet alleen voor die formules die relevant zijn voor interactie.

Een meer interactie-specifiek beeld van de complexiteit in strategische interacties zouden we kunnen krijgen door de complexiteit te bestuderen van het berekenen of een speler een manier van spelen heeft die winst garandeert (winnende strategie). Om dan de bron van deze complexiteit te bepalen kunnen we bestuderen welk invloed het op de complexiteit heeft als spelers andere doelen hebben of als we de regels van het spelverloop wijzigen.

De hier genoemde complexiteitsanalyses beschouwen de complexiteit van interactie vanuit een extern perspectief. Voor een meer interne visie op complexiteit van interactie kunnen we structuren bekijken die de kennis en informatie van de deelnemers van een interactief proces weergeven.

Uiteindelijk zou ook moeten worden bepaalt wat theoretische complexiteitsresultaten betekenen voor de moeilijkheid van interactieve processen in het echte leven.

In hoofdstuk 2 van dit proefschrift presenteren we een modale logica voor het redeneren over het strategische vermogen van agenten en coalities. We doen dit op een expliciete manier door de acties en voorkeuren van de agenten expliciet weer te geven. We bespreken de conceptuele voordelen van onze aanpak en bestuderen de complexiteit van het vervulbaarheidsprobleem van de ontwikkelde logica (oplosbaar in NEXPTIME en EXPTIME-moeilijk).

In hoofdstuk 3 bestuderen we logica's voor het redeneren over strategische kracht op een meer algemeen niveau. We kijken naar drie soorten Kripke modellen: eenvoudige modellen met een relatie voor elke coalitie, modellen die acties expliciet weergeven en modellen die de interne structuur van het strategisch vermogen van coalities weergeven. We laten zien dat de keuze van het systeem bepaalt of sterke of zwakke evenwichtsbegrippen makkelijker zijn om uit te drukken in de logica's.

Hoofstuk 4 kijkt naar meer specifieke strategische interactie en de complexiteit van het beslissingsprobleem of een bepaalde speler een winnende strategie heeft voor een *sabotage spel*. In dit tweespeler spel wandelt een speler door een graaf met het doel om een bepaald knoop te bereiken. De tegenstander probeert dit te voorkomen door kanten van de graaf te verwijderen. Onze resultaten tonen dat dit spel het makkelijkst is als de spelers samenwerken (NL-volledig) en dat de niet-coöperatieve versies moeilijker zijn (PSPACE-volledig). Dit geldt voor versies met een bereikbaarheidsdoelstelling maar ook voor een versie waarin de eerste speler het doel juist niet wil bereiken.

Hoofdstuk 5 bestudeert interactie van agenten in detail. We bestuderen de complexiteit van het vergelijken van de structuren die de informatie van agenten weergeven zoals de semantische structuren van kennislogica's. Onder de aanname dat kennis altijd waar is en introspectief (positief en negatief) gaat de complexiteit omhoog zodra we een tweede agent introduceren. Zonder die aannames gebeurt dit niet. Voor de manipulatie van de informatie van agenten laten we zien dat de aannames over de kennis sommige beslissingsproblemen die in het algemeen NP-moeilijk zijn, makkelijker kunnen maken.

In hoofdstuk 6 wordt de complexiteit van het gezelschapsspel *Eleusis* – een spel waarin inductief geredeneerd moet worden – bestudeert. Door middel van het Post Correspondentieprobleem laten we zien dat spelers in het spel Eleusis onbeslisbare problemen tegen kunnen komen bij het zoeken naar een legale zet. Dit maakt het spel dus in feite onspeelbaar. We geven aanbevelingen hoe de regels zouden kunnen worden aangepast om dit te voorkomen.

Abstract

This dissertation presents a formal analysis of the computational complexity aspects of interaction of agents.

As many interactive scenarios can be represented as relational structures, we use modal logic for formalizing them. This leads us to an investigation of the complexity of modal logics for reasoning about interaction. Capturing the complexity of interaction in terms of the complexity of satisfiability or model checking however might not be very accurate as these problems are concerned with arbitrary formulas of the associated language and not only with those expressing relevant concepts for interaction.

In order to capture more interaction-specific complexities in game-like interactive processes, we can investigate the complexity of deciding if a player has a winning strategy in a given game. Testing the robustness of this complexity with respect to small changes in the game rules or changing objectives of the players then helps to determine the sources of the complexity.

The complexity notions discussed above capture the complexity of interaction from an external perspective by focussing on how difficult it is to reason about interaction. Taking a more agent-oriented perspective leads us to the following question. What is the complexity of comparing agents' information structures and in how far is it influenced by underlying simplifying assumptions on the agents?

The final task arising for a formal analysis of the complexity of interaction is then to determine to what extent the analysis has implications for real-life interactive processes.

In this dissertation, it is shown how strategic ability of groups and individuals can be made explicit in modal logic in terms of actions and a representation of agents' preferences (Chapter 2). We discuss the conceptual benefits of such an approach and also determine the computational consequences for the satisfiability problem of the resulting logic, which turns out to be decidable in NEXPTIME and EXPTIME-hard.

More generally, considering simple coalition-labeled transition systems and action- and power based normal modal logics for reasoning about cooperative ability, Chapter 3 shows that the choice of primitive influences whether weak or strong stability notions are easier to express.

Focussing on more specific game-like interaction and the complexity of determining if a player has a wining strategy, Chapter 4 analyzes different versions of *Sabotage Games*, two player games played on a graph with one player moving through the graph trying to reach a goal vertex and the other player obstructing him by removing edges. It is shown that cooperative versions of this game are easiest (NL-complete), while the non-cooperative versions with reachability and safety objectives for the first player are both PSPACE-complete. The complexities are robust with respect to small changes in the procedural rules of the games.

Zooming in onto the agents involved in interaction, Chapter 5 analyzes the complexity of determining the relationship between different information structures of agents. It is shown that when assuming knowledge to be truthful and fully introspective a complexity jump occurs with the introduction of a second agent, while in the general case more agents do not increase the complexity. For problems related to the manipulation of agents' information it is shown that the assumptions on agents' knowledge can make intractable problems tractable.

In Chapter 6, the computational complexity of actually playing a recreational game is investigated for the inductive inference game *Eleusis*. Using Post's Correspondence Problem, it is shown that players can be forced to face undecidable problems during the game, which makes the game in principle unplayable. Recommendations are given for adjusting the rules of the game.

Titles in the ILLC Dissertation Series:

ILLC DS-2006-01: **Troy Lee**
Kolmogorov complexity and formula size lower bounds

ILLC DS-2006-02: **Nick Bezhanishvili**
Lattices of intermediate and cylindric modal logics

ILLC DS-2006-03: **Clemens Kupke**
Finitary coalgebraic logics

ILLC DS-2006-04: **Robert Špalek**
Quantum Algorithms, Lower Bounds, and Time-Space Tradeoffs

ILLC DS-2006-05: **Aline Honingh**
The Origin and Well-Formedness of Tonal Pitch Structures

ILLC DS-2006-06: **Merlijn Sevenster**
Branches of imperfect information: logic, games, and computation

ILLC DS-2006-07: **Marie Nilsenova**
Rises and Falls. Studies in the Semantics and Pragmatics of Intonation

ILLC DS-2006-08: **Darko Sarenac**
Products of Topological Modal Logics

ILLC DS-2007-01: **Rudi Cilibrasi**
Statistical Inference Through Data Compression

ILLC DS-2007-02: **Neta Spiro**
What contributes to the perception of musical phrases in western classical music?

ILLC DS-2007-03: **Darrin Hindsill**
It's a Process and an Event: Perspectives in Event Semantics

ILLC DS-2007-04: **Katrin Schulz**
Minimal Models in Semantics and Pragmatics: Free Choice, Exhaustivity, and Conditionals

ILLC DS-2007-05: **Yoav Seginer**
Learning Syntactic Structure

ILLC DS-2008-01: **Stephanie Wehner**
Cryptography in a Quantum World

ILLC DS-2008-02: **Fenrong Liu**
Changing for the Better: Preference Dynamics and Agent Diversity

ILLC DS-2008-03: **Olivier Roy**
Thinking before Acting: Intentions, Logic, Rational Choice

ILLC DS-2008-04: **Patrick Girard**
Modal Logic for Belief and Preference Change

ILLC DS-2008-05: **Erik Rietveld**
Unreflective Action: A Philosophical Contribution to Integrative Neuroscience

ILLC DS-2008-06: **Falk Unger**
Noise in Quantum and Classical Computation and Non-locality

ILLC DS-2008-07: **Steven de Rooij**
Minimum Description Length Model Selection: Problems and Extensions

ILLC DS-2008-08: **Fabrice Nauze**
Modality in Typological Perspective

ILLC DS-2008-09: **Floris Roelofsen**
Anaphora Resolved

ILLC DS-2008-10: **Marian Counihan**
Looking for logic in all the wrong places: an investigation of language, literacy and logic in reasoning

ILLC DS-2009-01: **Jakub Szymanik**
Quantifiers in TIME and SPACE. Computational Complexity of Generalized Quantifiers in Natural Language

ILLC DS-2009-02: **Hartmut Fitz**
Neural Syntax

ILLC DS-2009-03: **Brian Thomas Semmes**
A Game for the Borel Functions

ILLC DS-2009-04: **Sara L. Uckelman**
Modalities in Medieval Logic

ILLC DS-2009-05: **Andreas Witzel**
Knowledge and Games: Theory and Implementation

ILLC DS-2009-06: **Chantal Bax**
Subjectivity after Wittgenstein. Wittgenstein's embodied and embedded subject and the debate about the death of man.

ILLC DS-2009-07: **Kata Balogh**
Theme with Variations. A Context-based Analysis of Focus

ILLC DS-2009-08: **Tomohiro Hoshi**
Epistemic Dynamics and Protocol Information

ILLC DS-2009-09: **Olivia Ladinig**
Temporal expectations and their violations

ILLC DS-2009-10: **Tikitu de Jager**
"Now that you mention it, I wonder. . . ": Awareness, Attention, Assumption

ILLC DS-2009-11: **Michael Franke**
Signal to Act: Game Theory in Pragmatics

ILLC DS-2009-12: **Joel Uckelman**
More Than the Sum of Its Parts: Compact Preference Representation Over Combinatorial Domains

ILLC DS-2009-13: **Stefan Bold**
Cardinals as Ultrapowers. A Canonical Measure Analysis under the Axiom of Determinacy.

ILLC DS-2010-01: **Reut Tsarfaty**
Relational-Realizational Parsing

ILLC DS-2010-02: **Jonathan Zvesper**
Playing with Information

ILLC DS-2010-03: **Cédric Dégremont**
The Temporal Mind. Observations on the logic of belief change in interactive systems

ILLC DS-2010-04: **Daisuke Ikegami**
Games in Set Theory and Logic

ILLC DS-2010-05: **Jarmo Kontinen**
Coherence and Complexity in Fragments of Dependence Logic

ILLC DS-2010-06: **Yanjing Wang**
Epistemic Modelling and Protocol Dynamics

ILLC DS-2010-07: **Marc Staudacher**
Use theories of meaning between conventions and social norms

ILLC DS-2010-08: **Amélie Gheerbrant**
Fixed-Point Logics on Trees

ILLC DS-2010-09: **Gaëlle Fontaine**
Modal Fixpoint Logic: Some Model Theoretic Questions

ILLC DS-2010-10: **Jacob Vosmaer**
Logic, Algebra and Topology. Investigations into canonical extensions, duality theory and point-free topology.

ILLC DS-2010-11: **Nina Gierasimczuk**
Knowing One's Limits. Logical Analysis of Inductive Inference

ILLC DS-2011-01: **Wouter M. Koolen**
Combining Strategies Efficiently: High-Quality Decisions from Conflicting Advice

ILLC DS-2011-02: **Fernando Raymundo Velazquez-Quesada**
Small steps in dynamics of information

ILLC DS-2011-03: **Marijn Koolen**
The Meaning of Structure: the Value of Link Evidence for Information Retrieval

ILLC DS-2011-04: **Junte Zhang**
System Evaluation of Archival Description and Access

ILLC DS-2011-05: **Lauri Keskinen**
Characterizing All Models in Infinite Cardinalities

ILLC DS-2011-06: **Rianne Kaptein**
Effective Focused Retrieval by Exploiting Query Context and Document Structure

ILLC DS-2011-07: **Jop Briët**
Grothendieck Inequalities, Nonlocal Games and Optimization

ILLC DS-2011-08: **Stefan Minica**
Dynamic Logic of Questions

ILLC DS-2011-09: **Raul Andres Leal**
Modalities Through the Looking Glass: A study on coalgebraic modal logic and their applications

ILLC DS-2011-10: **Lena Kurzen**
Complexity in Interaction