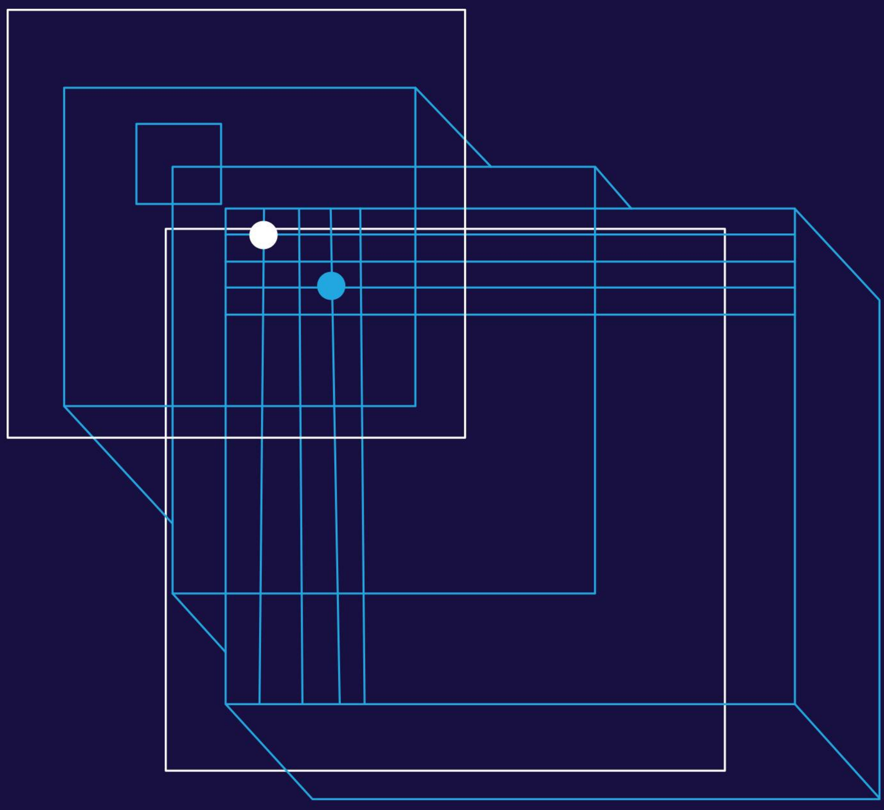
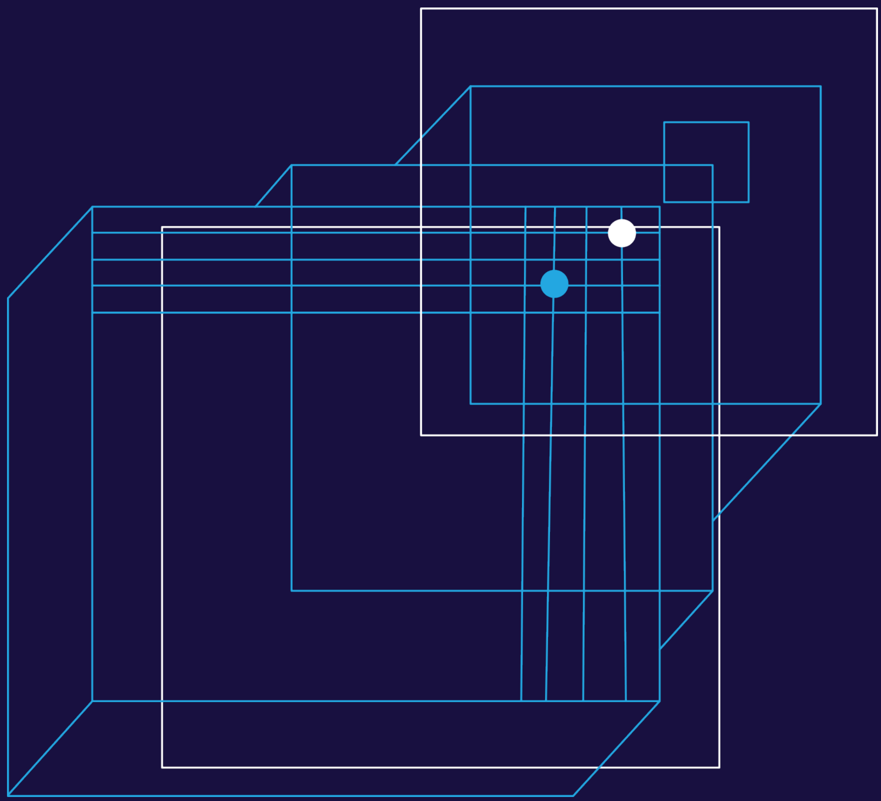
Games, Boards and Play

A Logical Perspective



Lei Li



博弈与棋盘：一个逻辑的视角

申请清华大学-阿姆斯特丹大学联合授予
博士学位论文

李磊

二〇二三年六月

Games, Boards and Play:
A Logical Perspective

Dissertation Submitted to

Tsinghua University and University of Amsterdam

in partial fulfillment of the requirement for a joint doctorate degree

by

Lei Li

June, 2023

Games, Boards and Play:
A Logical Perspective

Lei Li

Games, Boards and Play:
A Logical Perspective

ILLC Dissertation Series DS-2023-08

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 107
1098 XG Amsterdam
phone: +31-20-525 6051
e-mail: illc@uva.nl
homepage: http://www.illc.uva.nl/

We acknowledge the generous support of a 1-year Chinese Scholarship Council (CSC) scholarship.

Cover design by Yichen Liu.
Printed and bound by Print Service Ede.

ISBN: 978-90-833601-0-2

Games, Boards and Play: A Logical Perspective

ACADEMISCH PROEFSCHRIFT

ter verkrijging van de graad van doctor
aan de Universiteit van Amsterdam
op gezag van de Rector Magnificus
prof. dr. ir. P.P.C.C. Verbeek
ten overstaan van een door het College voor Promoties ingestelde commissie,
in het openbaar te verdedigen in de Agnietenkapel
op maandag 11 september 2023, te 10.00 uur

door Lei Li
geboren te Henan

Promotiecommissie

Promotores:	prof. dr. S.J.L. Smets	Universiteit van Amsterdam
	prof. dr. J.F.A.K. van Benthem	Tsinghua University
Copromotores:	dr. S. Ghosh	Indian Statistical Institute
Overige leden:	prof. dr. ing. R.A.M. van Rooij	Universiteit van Amsterdam
	dr. B.R.M. Gattinger	Universiteit van Amsterdam
	dr. D. Grossi	Universiteit van Amsterdam
	prof. dr. F. Liu	Tsinghua University
	prof. dr. Q. Feng	Tsinghua University
	prof. dr. D. Zhu	Tsinghua University

Faculteit der Natuurwetenschappen, Wiskunde en Informatica

Dit proefschrift is tot stand gekomen binnen een samenwerkingsverband tussen de Universiteit van Amsterdam en Tsinghua University met als doel het behalen van een gezamenlijk doctoraat. Het proefschrift is voorbereid in de Faculteit der Natuurwetenschappen, Wiskunde en Informatica van de Universiteit van Amsterdam en in de School of Humanities van Tsinghua University.

This thesis was prepared within the partnership between the University of Amsterdam and Tsinghua University with the purpose of obtaining a joint doctorate degree. The thesis was prepared in the Faculty of Science at the University of Amsterdam and in the School of Humanities at Tsinghua University.

To my family

TABLE OF CONTENTS

CHAPTER 1 INTRODUCTION

1.1 Motivations

Games are a powerful paradigm for social interaction, but at the same time also a good model for analyzing crucial notions in logical reasoning and computation. The study of games has attracted considerable attention from researchers across a range of fields, including sociology, economics, computer science, psychology, and philosophy, among others. Logic is a powerful tool used to investigate reasoning and decision analysis in games, as well as the social impact and normative regulations underlying social activities. From a logical perspective, modeling games, analyzing game-related reasoning from both player and modeler perspectives, investigating the impact of varying information access on agents' epistemic activities, and addressing computational problems related to games all hold both practical and academic significance. This dissertation specifically delves into game graphs, game board change, and the logical analysis of game elements across various scenarios. For a start, in this thesis, we analyze two sorts of games in terms of especially designed corresponding logical systems.

Topic 1. Sabotage games: modeling competitive or cooperative scenarios

The first kind of games we consider are graph games: in particular, sabotage games.

In the game, the graph that serves as the game board can change in the course of play, something which models scenarios where simple computational tasks can get disturbed in a hostile environment (Gierasimczuk et al., 2009). More positively, these games model scenarios where agents are pushed toward some desirable goal by removing false paths (Baltag et al., 2019b).

More specifically, one player moves along edges in order to reach the goal region, while the other player sabotages by deleting edges of the graph to prevent or help the first player. There are many different variants of the sabotage game (Rohde, 2005), we give a simple example here.

Example 1.1: Traveler and Demon play in turns in Figure 1.1. Traveler is located at the vertex g at the beginning of the game, and at each round, Demon first deletes an edge, and then Traveler moves into the next vertex along an available edge. We suppose that each player can observe the whole cube, actions played by others, and locations at each round through the game. Traveler wins if he finally arrives at any point in the goal region $\{a, f\}$.

In fact, Demon will win if he removes edges in the vertical direction once a vertex of that edge appears on the trajectory of Traveler.

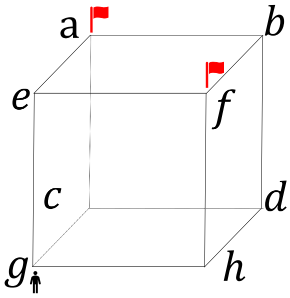


Figure 1.1 Game cube

Sabotage modal logic (SML for short) was proposed to study the reasoning in sabotage games (van Benthem, 2005). The formula $\Diamond\phi$ is used to express that Traveler can access the successor with the property ϕ, while $\blacklozenge\phi$ can be read that the current state has the property ϕ after Demon deletes some edge. Suppose the atomic proposition letter p only holds at the goal region, let's consider the players' win-loss status by formulas after each round. Since $\varphi_0 := \neg p$ holds at the starting vertex, then Traveler does not win at the beginning of the game. At the first round, Demon chooses to delete the edge (g, e) to prevent Traveler from reaching the vertex satisfying p, Traveler does not win at the end of this round, which is equivalent to say that the formula $\varphi_1 =: \blacklozenge \Box (\neg p)$ holds at the vertex g. In fact, at the end of n^{th} round,

$$\textit{Traveler does not win iff } \varphi_n =: \blacklozenge \Box ((\neg p) \wedge \varphi_{n-1}) \textit{ holds at } g$$

where $n \geq 1$. The cube has 12 edges, which means they can play 12 rounds at most. Traveler does not win this game iff the formula φ_{12} holds at g. Since the formula φ_{12} does hold at g, Demon has a strategy for winning this game.

Hence, for finite sabotage games (with finite edges), we can capture the existence of winning strategies for Demon, and by negating these assertions, of winning strategies for Traveler. However, we have difficulties in precisely depicting what the strategies are, because in SML we cannot accurately characterize players' actions. Still, SML has considerable expressive power, for instance, the property 'there are at most n successors' can be expressed in SML (Aucher et al., 2015). More results on SML for its model-theoretic and complexity aspects can be found in (Löding and Rohde, 2003b; Aucher et al., 2018; Areces et al., 2016; Li, 2020). However, one basic question left open was, perhaps surprisingly, providing an explicit axiomatization of the validities.

This problem is solved to some extent in Chapter 3 of this thesis. We slightly extend the language of sabotage modal logic with just enough expressive devices from hybrid logic, i.e., nominals and satisfiability operator @. In this way, we obtain a richer language and can give a more detailed characterization of the game. Since we assign a singleton to each nominal, we can give a concrete description of actions. For example, the formula $a \wedge \Diamond b$ can express that Traveler at the vertex named by nominal a can move into the vertex named by nominal b, while the formula $@_a \Diamond b \wedge \blacklozenge(@_a \neg \Diamond b)$ can express that Demon can delete an edge named by nominals a and b. With that, we can accurately capture players' strategies. In particular, the following operator $\langle a|b\rangle$ is useful in Chapter 3.

$$\langle a|b\rangle\varphi := (@_a \Diamond b \wedge \blacklozenge(@_a \neg \Diamond b \wedge \varphi)) \vee (@_a \neg \Diamond b \wedge \varphi)$$

Intuitively, this formula can be read that ' it is the case that φ after cutting a possibly existent edge from the world named by nominal a to the world named by nominal b'.

Based on the extended language, we develop a complete *Hilbert*-style axiomatization by drawing on some techniques from (van Benthem et al., 2020). We introduce several additional techniques that can also be used to obtain further results such as extended completeness theorems for protocol models (Hoshi, 2014) of hybrid sabotage modal logic that restrict the available sequences of link deletions. We also believe that these techniques can solve many further axiomatization problems for logics of other kinds of graph games. In addition, we clarify the connections between HSML-style logics of edge deletions and modal logics for stepwise point deletion from graphs (van Benthem et al., 2020).

Topic 2. Distributed games: a multi-perspective approach

In Example 1.1, there are two basic features. First, Traveler and Demon have access to the same information during the game. Apart from their respective decision-making processes, there is no information that one player knows and the other player is unaware of. Second, the external observer and the internal player have access to the same information during the game. External observers do not possess any additional information beyond that of the players. In other words, there is no distinction between the local internal perspective of any player and the global external perspective of the observer.

However, there are many game scenarios that do not conform to this observation.

Our next kind of game concerns interactive scenarios that have been much less studied in logical terms, where differences emerge between players' local internal view of a game and the global structure of the game as it proceeds. We employ the following example to explore this idea.

Example 1.2: Two players, Alice and Bob, are playing a card game and it is common knowledge that there are three available cards, $1, 2$ and 3, say. Suppose each of them gets a card from this pile of three cards, and one card is kept upside down so that nobody can see the value. Suppose, at each round, each of them can announce the following:

1. I have card number j
2. I accept
3. I challenge

We specify that both players only announce the card number that is equal to or higher than the actual card they have. The game starts with a round of simultaneous announcements of cards.[①] Subsequently,

- If they announce different card numbers, then
  - If the player with the announcement of the lower card announces 'I accept' in the next round, then the other player wins.
  - If the player with the announcement of the lower card announces 'I challenge' in the next round, then the other player has to show the card, and the player showing the card wins if the card matches her announcement, otherwise, the 'challenger' wins.
- If they announce the same card number, we specify that at least one player has to announce 'I challenge' in the next round. Whoever is challenged has to show the card, and this player wins if the card matches her announcement. If both players announce 'I challenge', then both have to show their cards. Both players may lose in case their cards do not match their announcements.

This is an example of a game of partial information, where each player has access to only a part of the game state and strategizes based on the local state and communication with others. In such games, players make their moves by making assumptions about other players' local states, and the actual global state decides which of the players' moves are actually enabled. This is the setting studied in Chapter 4, that of *distributed games*.

We use local models to capture the game from players' perspectives, while the global

① We could also allow them to make sequential announcements, and this would affect the players' strategies leading to a variation in the game analysis.

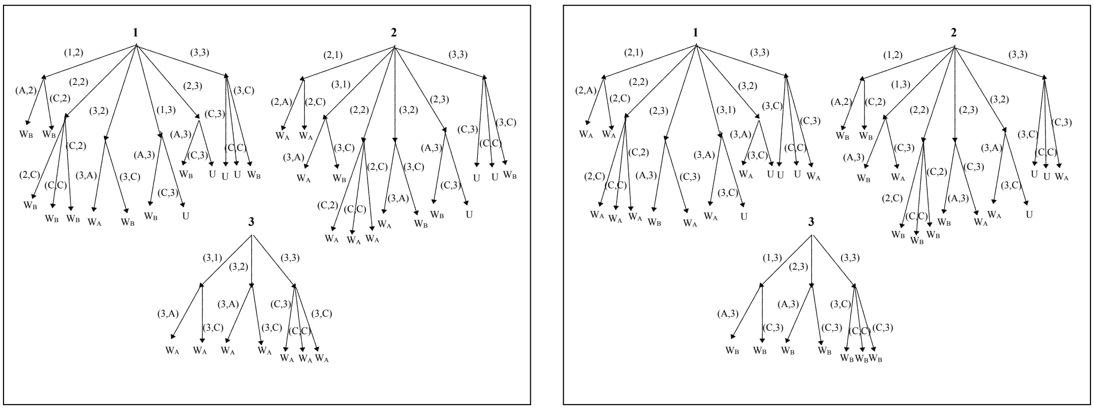


Figure 1.2 Local arenas for Alice (left) and Bob (right)

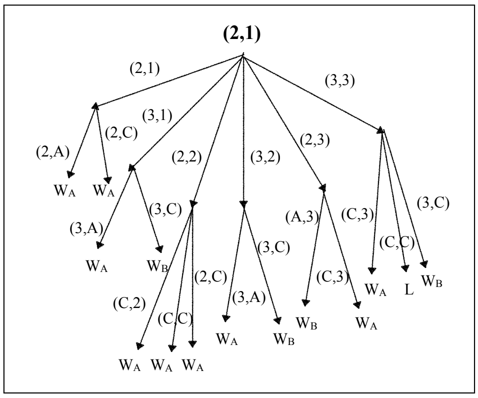


Figure 1.3 A global arena

arena generated from local models captures the whole game from the modeler's perspective. We present all the local models in Figure 1.2. In particular, let us consider the local models where Alice has received card number 2, and Bob card number 1, we have the corresponding global model in Figure 1.3.

Correspondingly, we propose two-layer language, allowing us to describe local and global perspectives precisely. For the technical results, we propose logics DGL and DGLEA, and give complete proof systems respectively. Moreover, we explore the complexity of the model-checking problems. In the end, we propose a similar framework to explore strategic reasoning. We believe that the style of analysis developed can also be applied to other game logics when we want to separate internal local views of agents from global process descriptions of what is going on as the game proceeds.

The remaining topics of the thesis explore two further directions. One direction stems from our the sabotage game concept and delves into computational problems on a broader scale, while the other explores a case study of personalized information on social platforms, which is somewhat in the spirit of the distributed game concept.

Topic 3. Complexity of Bisimulation problems: stemming out of sabotage games

The specific game logics discussed so far have a broader background. As will be demonstrated in Chapter 5, sabotage model logic, can be seen as instances of a much broader technical class of modal logics with modalities that describe the effects of various operations of model change. Such logics have been used for modeling both action and information flow, and there is a broad literature on both specific systems and general model-theoretic and proof-theoretic themes running through all of these. (e.g., (Plaza, 1989, 2007; Baltag et al., 1998; van Ditmarsch et al., 2008; Baltag et al., 2019b)) and others. Among these logics, we consider MCML($\langle sb\rangle$) and MCML($\langle gsb\rangle$) (Areces et al., 2012; Aucher et al., 2018; Fervari, 2014; van Benthem et al., 2022; Rohde, 2005)) to capture the class of models with edge deletion, MCML($\langle br\rangle$) and MCML($\langle gbr\rangle$) (Fervari, 2014) for the class of models with edge addition, and MCML($\langle sw\rangle$) and MCML($\langle gsw\rangle$) (Areces et al., 2014, 2012; Fervari, 2014) for the class of models arrow swap. In addition, we consider MCML($\langle de\rangle$) (van Benthem, 2005) and MCML($\langle ch\rangle$) (Thompson, 2020) for the class of models with point deletion and valuation change, respectively. We also know quite a bit about the computational complexity of model checking or satisfiability problems for these logics. For $\langle up\rangle \in \{\langle sb\rangle, \langle gsb\rangle, \langle sw\rangle, \langle br\rangle, \langle ch\rangle\}$, the satisfiability problem for $MCML(up)$ is *undecidable*, and for $\langle up\rangle \in \{\langle sb\rangle, \langle gsb\rangle, \langle sw\rangle, \langle gsw\rangle, \langle br\rangle, \langle gbr\rangle\}$, the model-checking problem for $MCML(up)$ is **PSPACE**-complete. However, one question that has received little attention so far is one that connects basic model-theoretic and complexity-theoretic concerns: what is the precise complexity of testing for the appropriate notions of bisimulation between given finite models.

As we know, the study of bisimulation notions can be employed to explore the expressive power of logical languages.

Example 1.3: Consider the two pointed models (M_1, w_1) and (M_2, w_2) in Figure 1.4. Please note that (M_1, w_1) and (M_2, w_2) are bisimilar, but they are not sabotage bisimilar. Thus, we cannot distinguish these two models with the language of basic modal logic, but we can differentiate them with the language of SML (i.e., MCML($\langle gsb\rangle$)), since the formula $\Diamond\blacklozenge\Diamond^2\top$ holds at (M_1, w_1), but it does not hold at (M_2, w_2).

As we have shown above, the language of MCML($\langle gsb\rangle$) has stronger expressive power than that of basic modal logic. The notion of bisimulation serves not only as a touchstone for expressive power but also, as a significant concept in game (graph) logic, enabling a novel measure of game equivalence.

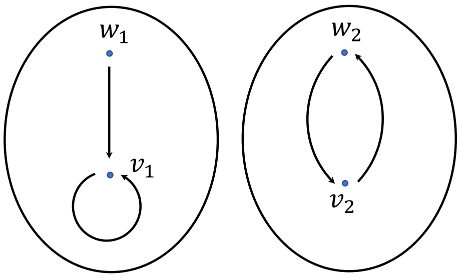


Figure 1.4 (M_1, w_1) (left) and (M_2, w_2) (right)

Example 1.4: Consider two sabotage games played on directed graphs G_1 and G_2 in Figure 1.5. In the graph G_1, the vertex s is the starting point, and the goal region is the set $\{d\}$, while in the graph G_2, the vertex s_1 is the starting point, and the goal region is the set $\{d_1, d_2\}$.

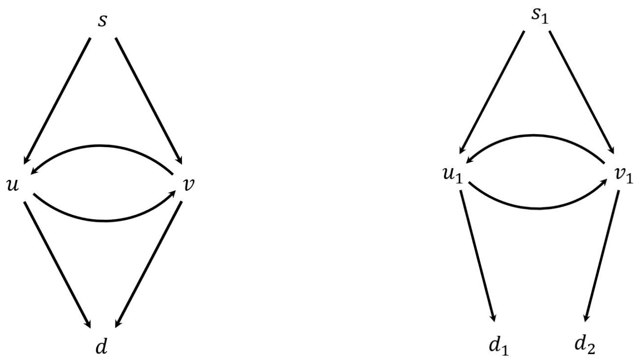


Figure 1.5 Directed graphs G_1 (left) and G_2 (right)

It is not hard to check that s and s_1 are sabotage bisimilar, and the player Demon wins in both of the games. In fact, in two sabotage games, if the starting points are bisimilar, and for any point in the goal region of one game, there is a bisimilar point in the goal region of another game, then players have winning strategies in one game iff players have winning strategies in the other game according to the definition of sabotage bisimilarity. Moreover, it's not surprising that "game equivalence" does not imply sabotage bisimulation.

Taking the approach to design the bisimulation game for standard bisimulation relation (van Benthem, 2010), and staying in line with our main topic so far, we can also view bisimulations in terms of games. In particular, we can design 'sabotage bisimulation games' as follows, giving us a finer tool for analyzing winning powers of players in logical terms.

Player S ("Spoiler") claims that two pointed models (M, s) and (N, t) are different, while player D ("Duplicator") says they are similar. They play over k rounds, starting from the match $((M,s),(N,t))$. If objects matched in a round differ in any atomic property, S wins. In each round, Spoiler starts with $((M,w),(N,v))$, and there are two cases.

- She chooses some pointed model (M,w') satisfying that w' is a successor of w in M (or some pointed model (N,v') satisfying that v' is a successor of v in N). Next, Duplicator must respond with a pointed model (N,v') satisfying that v' is a successor of v in N ((M,w') satisfying that w' is a successor of w in M, respectively), and the world match after the round is $((M,w'),(N,v'))$. If a player cannot choose a pointed model when it is her turn in a round, she loses.
- she chooses some pointed model (M',w) satisfying that M' is a model after some edge in M is deleted (or some pointed model (N',v) satisfying that N' is a model after some edge in N is deleted). Next, Duplicator must respond with a pointed model (N',v) satisfying that N' is a model after some edge in N is deleted ((M',w) satisfying that M' is a model after some edge in N is deleted, respectively), and the world match after the round is $((M',w),(N',v))$. If a player cannot choose such a pointed model when it is her turn in a round, she loses.

For a sabotage bisimulation game played with two pointed models (M_1,w_1) and (M_2,w_2), Duplicator wins iff (M_1,w_1) and (M_2,w_2) are sabotage bisimilar.

Given the entanglement between bisimulation notions and the first topic of this thesis, namely games, we incorporate the exploration of bisimulation problems into our research as a further extension of our study.

We undertake what we believe is a first in-depth study of this theme and find a number of lower and upper bounds, though we have not yet been able to settle the precise complexity of testing for basic sabotage bisimulation.

Topic 4. A case study in social platforms: in the spirit of distributed games

Our final topic in this thesis concerns another extension of the concerns in topic 2 for game scenarios. We undertake a practical case study where local and global multi-agent perspectives play in social platforms. In the spirit of distributed games, the operator of a social platform can be likened to a game designer accessing information from a global perspective. On the other hand, all users can only passively receive information distributed by the operator, which aligns with the localized perspective obtained by individual agents. The platform's filtering mechanisms, akin to game rules, influence the information available to players, directly impacting their decision-making and ultimately determining the outcome of the game. We present the following example for more details.

Example 1.5: Consider a scenario on a social platform where agents post their opinions on certain topics and receive information based on the filtering mechanisms employed by the platform. Fix a group Ag of agents and a set T of topics, where Ag consists of Alice (A) and Bob (B), and T only has two elements, cuisine (C) and sport (S). The topic of cuisines involves Chinese cuisine, Italian cuisine, and Mexican cuisine, while the topic of sports exclusively focuses on football. Their opinions on the topics mentioned are as follows.

- On topic C
  - Alice likes Chinese cuisine.
  - Alice likes Italian cuisine.
  - Bob likes Italian cuisine.
  - Bob likes Mexican cuisine.
- On topic S
  - Alice has no interest in sports at all.
  - Bob likes football.

Moreover, Alice believes that Bob likes Italian cuisine, but she has no idea whether Bob likes Mexican cuisine.

Filtering rules: Once someone posts a post on the platform, only those users who are interested in the relevant topics mentioned in the post will receive it.

Post P1 from Bob: ‘I like Mexican cuisine on topic C’.

Post P2 from Bob: ‘I like football on topic S’.

Let’s consider the belief dynamics of Alice due to the filtering rules when Bob sends Post P1 or P2 respectively.

- For the case that Bob sends Post P1, since Alice is interested in topic C. Post P1 will appear in the information flow of Alice, and then Alice will believe Bob holds the opinion m on topic C.
- For the case that Bob sends Post P2, since Alice isn’t interested in topic S. Post P2 won’t appear in the information flow of Alice, and even Alice won’t realize that Bob has already sent Post P2, and her beliefs keep the same.

Moreover, if Alice does not have enough available attention to view Post P1, then which means that even if this post appears in her flow, it will not be viewed and change her beliefs.

This rich example suggests many topics for logical analysis. In particular, there are now limitations on what agents can observe and infer due to two influences: the filtering rules performed by the recommendation system and the limited attention available to participants in social networks.

In Chapter 6, we focus on the first-mentioned topic, that of filtration performed by a system, and propose two modal logics for epistemic reasoning in social platforms, with filtration viewed in both static and dynamic settings. In the static setting, we employ 'opinion models' coming from the work of Smets and Velázquez-Quesada (2019a) as our main semantic tool, and introduce a special modality expressing filtering mechanisms from a syntactic perspective. Moreover, we present a complete axiomatization of the resulting valid principles. Next, moving to a dynamic setting, we extend the static language with a dynamic modality capturing the model transformations resulting from filtering mechanisms. Here too, we provide an axiom system for which we prove soundness and completeness. Finally, we also discuss how to extend this logic-based approach to deal with the representation and dynamics of attention span (Wickens, 2021) for the users of social media platforms.

There are numerous works closely related to this chapter. The propagation of opinions in social networks, the dynamics of the network's structure, and the entanglement of knowledge and the social relation structure have all been extensively studied. See e.g. (Seligman et al., 2011; Smets and Velázquez-Quesada, 2017, 2018, 2019a,b; Baltag et al., 2019a; Smets and Velázquez-Quesada, 2020; Liu and Liao, 2021; Liu and Li, 2022) and others. It is worth noting that we do not employ the notion of social relations in our models, and we will delve into more detailed considerations in the future.

To summarize, we start with two types of games, i.e., sabotage games and distributed games, and present the corresponding logical analysis. Next, we transition from the perspective of sabotage modal logic to a broader viewpoint, and then focus on the exploration of the complexity of bisimulation problems in some model-changing logics, which may be seen in our setting as a notion of game equivalence. In the end, we propose a case study for personalized announcements, giving rise to a natural distinction between the behavior of individual agents and that of the total system which is in the spirit of the distributed games studied in Chapter 4.

1.2 Outline of the thesis

Here is a brief summary of the main topics and results in this dissertation.

In Chapter 3 we start from sabotage games, and design a new hybrid modal logic HSML. We extend sabotage modal language with additional nominals and satisfaction operators, which enhances the ability to characterize sabotage games, at the same time, we provided a complete Hilbert-style axiomatization. Take into account the behavioral constraints that the player may have in more complicated sabotage games, we also introduce protocol models with restrictions on available edge deletions, and obtain the corresponding proof system. At last, we clarify the connections between HSML-style logics of edge deletions and recent modal logics for stepwise point deletion from graphs.

In Chapter 4 we focus on another type of game, i.e. the distributed game. we introduce the basic notions of local and global arenas for capturing the game from different perspectives, and propose distributed game logic (DGL) to characterize the reasoning about distributed games accordingly. We then provide a strong completeness result for the proposed logic. Afterwards, we propose a distributed game logic with enabled actions (DGLEA) with subtle differences from DGL to characterize more realistic interplay between local and global reasoning, and present a complete axiom system for it. In addition, we study the complexity of the model checking problem for these logics. Finally, we explore to incorporate strategic announcements on a similar framework.

In Chapter 5 we zoom in on graph games, and focus on a common feature of many graph games: graph change or model change at the model level, we extend the standard modal language with an additional operator expressing model change. This operator can be specified according to the model change we want to capture, such as edge-deleting change, arrow-swapping change and valuation change, etc. With this language, we presented the notion of bisimulation uniformly. Moreover, we investigate the model comparison problem by providing a uniform algorithmic study. Through our algorithmic analyses, we provided PSPACE upper bound results with respect to those modal logics.

In Chapter 6 we show how the filtration dynamics can be specified and analyzed completely in dynamic-epistemic logics of communication involving filtering events. More specifically, we start by proposing the static logic of personalized announcements (SLPA), followed by the dynamic logic of personalized announcements (DLPA), including axiomatizations, soundness and completeness results. The complementary process of attention dynamics can also be specified and studied in the same manner, but this extension is beyond the scope of this thesis.

In summary, the four chapters of this thesis extend the current logical study of game scenarios for multi-agent interaction with new results and techniques and also with new agenda items, some from the area of computation and some from current social media scenarios on the internet.

1.3 Sources of each chapter

- Chapter 3 is based on:

  Johan van Benthem, Lei Li, Chenwei Shi, Haoxuan Yin (2022). Hybrid sabotage modal logic. *Journal of Logic and Computation*, exac006, https://doi.org/10.1093/logcom/exac006.

  Author contributions: Johan van Benthem initiated this project, while all authors were equally involved in developing the main ideas. Lei Li developed the proof of completeness of HSML from a draft by Chenwei Shi, and organized the technical comparison results with modal logics of point removal with contributions from all the authors. Chenwei Shi and Haoxuan Yin completed the protocol version of HSML.
- Chapter 4 is based on:

  Sujata Ghosh, Lei Li, Fenrong Liu, R. Ramanujam (2023). A modal logic to reason in distributed games. *Manuscript*.

  Author contributions: Sujata Ghosh, Fenrong Liu and R. Ramanujam initiated the project and shaped the overall narrative of the paper. Lei Li developed the central logical framework, including the key results on axiomatization, soundness, and completeness.
- Chapter 5 is based on:

  Sujata Ghosh, Shreyas Gupta, Lei Li (2022). Bisimulation in model-changing modal logics: An algorithmic study. *Journal of Logic and Computation*, *Accepted*.

  Author contributions: Sujata Ghosh and Shreyas Gupta initiated the project, while all authors were equally involved in its implementation. The central algorithms and the proofs of their correctness were developed by Lei Li and Shreyas Gupta together.
- Chapter 6 is based on:

  Gaia Belardinelli, Lei Li, Sonja Smets, Anthia Solaki (2023). Logics for personalized announcements. *Manuscript*.

Author contributions: Sonja Smets and Anthia Solaki initiated the project, while all authors were equally involved in developing the main ideas. The basic setting for the logical framework was first provided by Anthia Solaki which was then further developed by Lei Li, including the formalization of key concepts, axiomatizations, and proofs of soundness, and completeness.

CHAPTER 2 PRELIMINARIES

In this chapter, we provide a comprehensive overview of the preliminary aspects of this thesis. We begin by introducing extensive games, followed by sabotage games, sabotage modal logic, and several variations of this logic, i.e., model-changing modal logics. Additionally, we present an overview of dynamic epistemic logics. Finally, in response to the requirements of complexity research, we present foundational concepts in the field of complexity theory.

2.1 Extensive games

For simplicity, we only focus on the games in which players do not take actions at the same time, all actions are made by the players and randomness is not allowed.

Definition 2.1 (Extensive game): Let N be a finite set of players, A be a set of actions, and H be a set of sequences over A. An *extensive game* has the form $(N, H, P, \succsim_i)$ satisfying the following conditions.

- $\varnothing \in H$.
- If $(a^k)_{k=1}^n \in H$, for $m < n$, we have $(a^k)_{k=1}^m \in H$.
- An infinite sequence $(a^k)_{k=1}^\infty \in H$ iff $(a^k)_{k=1}^m \in H$ for $m \geq 1$.

  (We call the element of H is a history; a history consists of actions by some players.) A history $(a^k)_{k=1}^m \in H$ is terminal if it is infinite or if there is not an additional action a^{m+1} such that $(a^k)_{k=1}^{m+1} \in H$. The set of actions available after the nonterminal history h is denoted $A(h) = \{a \mid (h, a) \in H\}$ and we use Z for the set of terminal histories.
- P is a function that assigns to each nonterminal history (each member of $H \setminus Z$) a member of N.
- $\succsim_i$ is a binary (preference) relation over Z for each player $i \in N$.

For an extensive game form $(N, H, P, \succsim_i)$, we define a partition $\mathcal{I}_i$ of $\{h \in H : P(h) = i\}$ satisfying $A(h) = A(h')$ if h and h' are in the same member of the partition. For $I_i \in \mathcal{I}_i$, player i cannot distinguish any elements of I_i, so we use $A(I_i)$ for the set $A(h)$ and by $P(I_i)$ denoting the player $P(h)$ for any $h \in I_i$. ($\mathcal{I}_i$ is the information partition of player i; a set $I_i \in \mathcal{I}_i$ is an information set of player i.)

The *extensive game with perfect information* refers to a game in which, at any stage of the game, every player knows exactly what has taken place earlier in the game, which implies every information set I_i is a singleton set. In particular, if some information set is not a singleton set, we call it an *extensive game form with imperfect information*.

Definition 2.2 (Strategy): In an extensive game with the form $(N, H, P, (\mathcal{I}_i)_{i\in N}, \succsim_i)$, a *strategy* of player i is a function that assigns an action in $A(I_i)$ to each information set $I_i \in \mathcal{I}_i$.

The definition of extensive games above is highly flexible, allowing us to further characterize intricate game scenarios. For instance, we can define an extensive game with perfect recall, which refers to players remembering everything they have known throughout the game process. Thus, we introduce $X_i(h)$ to record player i's experience along the history h. Formally, $X_i(h)$ is the ordered sequence consisting of the information sets that the player encounters in turn in the history h and the actions that he takes at them.

Definition 2.3 (Extensive game with perfect recall): An extensive game with the form $(N, H, P, (\mathcal{I}_i)_{i\in N}, \succsim_i)$ has perfect recall if for each player i we have $X_i(h) = X_i(h')$ whenever the histories h and h' are in the same information set of player i.

Let's give an example to instantiate definitions above.

Example 2.1: Consider an extensive game with the form $(N, H, P, \{\succsim_i\}_{i\in N})$ in Figure 2.1, where the set of players $N = \{T, D\}$, the set of histories H is $\{\varnothing, (a_1), (a_2), (a_2, b_1), (a_2, b_2), (a_2, b_1, c_1), (a_2, b_1, c_2), (a_2, b_2, c_1), (a_2, b_2, c_2)$, and for function P, $P(\varnothing) = P(a_2, b_1) = P(a_2, b_2) = T$, $P(a_2) = D$. In addition, the preference relations $\succsim_T$ and $\succsim_D$ for players are as follows:

- $(a_1) \succsim_T (a_2, b_1, c_1) \succsim_T (a_2, b_1, c_2) \succsim_T (a_2, b_2, c_1) \succsim_T (a_2, b_2, c_2)$
- $(a_2, b_2, c_1) \succsim_D (a_2, b_1, c_1) \succsim_D (a_2, b_1, c_2) \succsim_D (a_1) \succsim_D (a_2, b_2, c_2)$

Hence, the information partition $\mathcal{I}_T$ for player T is $\{\{\varnothing\}, \{(a_2, b_1), (a_2, b_2)\}\}$, while $\mathcal{I}_D$ for player T is $\{\{a_2\}\}$. Then there is an extensive game with imperfect information. In Figure 2.1, for player T, there is a dotted line connecting the ends of the histories (a_2, b_1) and (a_2, b_2), meaning player T cannot distinguish these two histories. Let $S_T(\{\varnothing\}) = a_2$, $S_T(\{(a_2, b_1), (a_2, b_2)\}) = c_1$, the function S_T specifies a strategy for player T. In addition, we can also list the function X_T on H, such as $X_T((a_2, b_1)) = (\{\varnothing\}, a_2, \{(a_2, b_1), (a_2, b_2)\})$ and ect, finally, we ensure the game in Figure 2.1 has perfect recall.

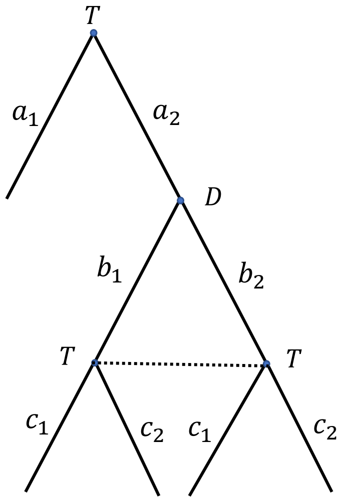


Figure 2.1 An extensive game with imperfect information

The logic analysis involving extensive games can be found in (van Benthem, 2002; Harrenstein et al., 2003; van Benthem, 2014; Grossi and Turrini, 2012; Liu et al., 2016) and others. Next, we introduce another type of game, i.e., sabotage games, and sabotage modal logic matching these games.

2.2 Sabotage game and sabotage modal logic

In the previous chapter, we have already given some examples of sabotage games, we now present a more formal definition, which is based on the proposal in (Rohde, 2005).

Definition 2.4: A graph G is a tuple $(W, R_0, \ldots, R_m)$, where

- W is a nonempty finite set of states,
- $R_i \subseteq W \times W$.

If $R_i \cap R_j = \varnothing$ for any $i \neq j$, we call G a simple graph. For any $w, u \in W$, if $(w, v) \in R_i$ implies $(v, w) \in R_j$ for some j, then we call G an undirected graph.

A sabotage game is a zero-sum game played by Traveler (T) and Demon (D) over (G, w_0), where $G = (W, R_0, \ldots, R_m)$, $w_0 \in W$ is an initial state. For convenience, we require that $\bigcup_m R_m \neq \varnothing$. A game position is a tuple $(\tau, s, R_1, \ldots, R_m)$, where

- $\tau \in \{T, D\}$ is a turn function, which specifies who will move next
- s is the current state
- $R_i \subseteq W \times W$. A pair $(w, u) \in R_i$ for some i indicates there is an available edge (w, u) in the graph.

Game rules are as follows: the initial game position is given by $(D, s_0, R_0^0, \ldots, R_m^0)$, which means the player Demon starts the game at s_0. Traveller and Demon play in turns. In each round, if $(D, s, R_0, \ldots, R_m)$ is the current position, then the player Demon deletes an edge in the graph, which means he chooses a pair (w, u) in some nonempty set R_i and $(T, s, R_0, \ldots, R_{i-1}, R_i \setminus \{(w, u)\}, R_{i+1}, \ldots, R_m)$ becomes the successor game position. In the case of undirected graphs, Demon additionally chooses the dual pair (u, w) in some nonempty set R_j, and $(T, s, R_0, \ldots, R_{i-1}, R_i \setminus \{(w, u)\}, R_{i+1}, \ldots, R_{j-1}, R_j \setminus \{(u, w)\}, R_{j+1}, \ldots, R_m)$ becomes the successor game position. If $(T, s, R_0, \ldots, R_m)$ is the current game position, Traveler chooses a state t such that $(s, t) \in R_i$ for some i, and $(D, t, R_0, \ldots, R_{i-1}, R_i \setminus \{(w, u)\}, R_{i+1}, \ldots, R_m)$ becomes the successor game position.

A player loses if he cannot make a move. Since $\bigcup_m R_m \neq \varnothing$ for the initial game position $(\tau, s, R_1, \ldots, R_m)$, then only Traveler cannot make a move in some state. Demon can always delete the edge along which Traveler moves to the successor in the previous round.

A *play* is a sequence of game positions $\pi_0, \ldots, \pi_{i-1}, \pi_i, \pi_{i+1}, \ldots$, where π_0 is the initial game position. In the i^{th} round, Demon is in the game position π_{i-1} and then moves to the game position π_i, while Traveler is in the game position π_i, and then moves to the game position π_{i+1}. Several candidates can be used for winning conditions in the following:

- Reachability. Fix a set of states $F \subseteq W$ as termination states, Traveler wins the play $\pi_0, \ldots, \pi_i, \ldots, \pi_k$ if and only if $\pi_k = (D, s, R_0, \ldots, R_m)$ with $s \in F$, which means Traveler reaches some state in F.
- Complete search. Traveller wins the play $\pi_0, \ldots, \pi_i, \ldots, \pi_k$ if and only if for any $w \in W$, there is a game position $\pi_i = (\tau_i, s_i, R_0^i, \ldots, R_m^i)$ for some i such that $w = s_i$, which means Traveler can visit each state at least once.
- Hamilton path. Traveller wins the play $\pi_0, \ldots, \pi_i, \ldots, \pi_k$ if and only if for any $w_i \in W$, there is a game position $\pi_i = (\tau_i, s_i, R_0^i, \ldots, R_m^i)$ for some i such that $w_i = s_i$, and $s_i \neq s_j$ for $w_i \neq w_j$, which mean Traveler can move along a Hamilton path, where he visits each state exactly once.

It is worth pointing out that sabotage games are games with complete information, which means that all players are fully aware of the current game position, the history of previous plays, the rules, as well as the winning conditions at all times. For the following chapters, the term ‘sabotage game’ refers to a sabotage game with a reachability condition, formally (G, w_0, F), where G is a graph $(W, R_0, \ldots, R_m)$, w_0 is the initial state in W, F

is a fixed subset of W. Sometimes F is a singleton, we use (G, w_0, t) for $(G, w_0, \{t\})$. To reason about sabotage games, van Benthem (2005) proposed the sabotage modal logic.

Definition 2.5 (Language of SML): Let Prop be a nonempty countable set of propositional variables. The language of sabotage modal logic *SML* is defined by the following grammar:

$$\varphi ::= p \mid \neg\varphi \mid \varphi \wedge \varphi \mid \Diamond\varphi \mid \blacklozenge\varphi$$

where $p \in \mathsf{Prop}$. Additionally, $\phi \vee \varphi := \neg(\neg\phi \wedge \neg\varphi)$, $\phi \to \varphi := \neg\phi \vee \varphi$, $\phi \leftrightarrow \varphi := (\phi \to \varphi) \wedge (\varphi \to \phi)$, $\Box\phi := \neg\Diamond\neg\phi$ and $\blacksquare\phi := \neg\blacklozenge\neg\phi$. Intuitively, $\blacklozenge\varphi$ can be read "after an arrow is deleted in the model, it is the case that φ". Formally, we can define it as follows.

Definition 2.6 (Model): A *model* $\mathfrak{M} = (W, R, V)$ is a standard modal relational model with worlds W, accessibility relation R and valuation function V. We call $(\mathfrak{M}, w)$ a pointed model when $\mathfrak{M}$ is a model and w is a world on it.

Definition 2.7 (Truth conditions): The semantics of SML is as follows:

$\mathfrak{M}, w \vDash p$	*iff*	$w \in V(p)$
$\mathfrak{M}, w \vDash \neg\varphi$	*iff*	not $\mathfrak{M}, w \vDash \varphi$
$\mathfrak{M}, w \vDash \varphi \wedge \phi$	*iff*	$\mathfrak{M}, w \vDash \varphi$ *and* $\mathfrak{M}, w \vDash \phi$
$\mathfrak{M}, w \vDash \Diamond\varphi$	*iff*	$\mathfrak{M}, v \vDash \varphi$ for some v with Rwv
$\mathfrak{M} = (W, R, V), w \vDash \blacklozenge\varphi$	*iff*	there is a pair $(u, v) \in R$ such that $(W, R \setminus (u, v), V), w \vDash \varphi$

For convenience, let us define relation $\mathbf{r}$: $((\mathfrak{M}_1, w_1), (\mathfrak{M}_2, w_2)) \in \mathbf{r}$ if the following holds: (i) $W_2 = W_1$, (ii) $R_2 = R_1 \backslash \{(u, v)\}$ for some $(u, v) \in R_1$, (iii) $V_2 = V_1$, and (iv) $w_2 = w_1$. With this new relation, the semantics of $\blacklozenge\varphi$ can be seen as follows:

$\mathfrak{M}, w \vDash \blacklozenge\varphi$ *iff there is* $(\mathfrak{M}', w')$ *such that* $(\mathfrak{M}, w)\mathbf{r}(\mathfrak{M}', w')$ *and* $\mathfrak{M}', w' \vDash \varphi$.

The sabotage bisimulation was introduced in Aucher et al. (2018), we give the definition as follows:

Definition 2.8 (Sabotage bisimulation): Let $\mathfrak{M}_1 = ((W_1, R_1, V_1), w_1)$ and $\mathfrak{M}_2 = ((W_2, R_2, V_2), w_2)$ be two pointed models. We say that $(\mathfrak{M}_1, w_1)$ and $(\mathfrak{M}_2, w_2)$ are *sabotage bisimilar*, denoted by $(\mathfrak{M}_1, w_1)Z(\mathfrak{M}_2, w_2)$, if the following conditions are satisfied:

- Atom: If $(\mathfrak{M}_1, w_1)Z(\mathfrak{M}_2, w_2)$, then $(\mathfrak{M}_1, w_1) \vDash p$ iff $(\mathfrak{M}_2, w_2) \vDash p$ for all atomic propositions p.

- Zig$_{\Diamond}$: If $(\mathfrak{M}_1, w_1)Z(\mathfrak{M}_2, w_2)$, and there exists $v_1 \in W_1$ such that $w_1 R_1 v_1$, then there is a $v_2 \in W_2$ such that $w_2 R_2 v_2$ and $(\mathfrak{M}_1, v_1)Z(\mathfrak{M}_2, v_2)$.
- Zag$_{\Diamond}$: Same as above in the converse direction.
- Zig$_{\blacklozenge}$: If $(\mathfrak{M}_1, w_1)Z(\mathfrak{M}_2, w_2)$, and there is $\mathfrak{M}'_1$ such that $(\mathfrak{M}_1, w_1)\mathbf{r}(\mathfrak{M}'_1, w_1)$, then there is an $\mathfrak{M}'_2$ such that $(\mathfrak{M}_2, w_2)\mathbf{r}(\mathfrak{M}'_2, w_2)$ and $(\mathfrak{M}'_1, w_1)Z(\mathfrak{M}'_2, w_2)$.
- Zag$_{\blacklozenge}$: Same as above in the converse direction.

In Example 1.3, we give two pointed models are bisimilar, but not sabotage bisimilar, which means the notion of sabotage bisimilar is indeed stronger than bisimilar. The expressive power can be measured by the following fact.

Fact 1: If two pointed models $(\mathfrak{M}_1, w_1)$, $(\mathfrak{M}_2, w_2)$ are sabotage bisimilar, then $\mathfrak{M}_1, w_1 \vDash \phi$ iff $\mathfrak{M}_2, w_2 \vDash \phi$ for any formula ϕ.

We explore the complexity of the model comparison problem for sabotage modal logic in Chapter 5, which concerns comparing two pointed models to determine if they are sabotage bisimilar.

2.3 Model-changing modal logics

The modality $\blacklozenge$ as a model-transforming operator characterizes model changes with edge deletions. In this subsection, we present more model-changing modal logics with some specific operators, which are proposed in Areces et al. (2012, 2014); Fervari (2014).

Definition 2.9 (Language of $MCML(up)$): Let Prop be a nonempty countable set of propositional variables. The language of model-changing modal logics $MCML(up)$ is defined by the following grammar:

$$\varphi ::= p \mid \neg\varphi \mid \varphi \wedge \varphi \mid \Diamond\varphi \mid \langle up\rangle\varphi$$

where $p \in$ Prop, $\langle up\rangle$ is the model-changing operator, which can be specified as $\langle sb\rangle$ (sabotaging edges), $\langle gsb\rangle$ (globally sabotage edges), $\langle sw\rangle$ (swapping edges), $\langle gsw\rangle$ (globally swapping edges), $\langle br\rangle$ (bridging edges) and $\langle gbr\rangle$ (globally bridging edges), which we explain below. Additionally, $\phi \vee \varphi := \neg(\neg\phi \wedge \neg\varphi)$, $\phi \to \varphi := \neg\phi \vee \varphi$, $\phi \leftrightarrow \varphi := (\phi \to \varphi) \wedge (\varphi \to \phi)$, $\Box\phi := \neg \Diamond \neg\phi$ and $[up]\phi := \neg\langle up\rangle\neg\phi$.

Intuitively, $\langle up\rangle\varphi$ can be read "it is the case that φ in some transformed model induced by the operator $\langle up\rangle$". Before we define it formally, we first introduce the following notations for transformed models. Fix a model $\mathfrak{M} = (W, R, V)$, then

- $\mathfrak{M}_S^- = (W, R_S^-, V)$, where $R_S^- = R \setminus S$, $S \subseteq R$,
- $\mathfrak{M}_S^* = (W, R_S^*, V)$, where $R_S^* = R \cup S \setminus (W \times W \setminus S)$, $S \subseteq W \times W \setminus R$,
- $\mathfrak{M}_B^+ = (W, R_B^+, V)$, where $R_B^+ = R \cup B$, $B \subseteq W \times W \setminus R$,

Now we give the truth conditions for all new operators, and the intuitive meaning is shown in Figure 2.2-2.7.

- $\mathfrak{M}, w \vDash \langle sb \rangle \phi$, if $\mathfrak{M}^-_{\{(w,v)\}}, v \vDash \phi$ for some $v \in W$ with Rwv.
- $\mathfrak{M}, w \vDash \langle gsb \rangle \phi$, if $\mathfrak{M}^-_{\{(u,v)\}}, w \vDash \phi$ for some $u, v \in W$ with Ruv.
- $\mathfrak{M}, w \vDash \langle sw \rangle \phi$, if $\mathfrak{M}^*_{\{(v,w)\}}, v \vDash \phi$ for some $v \in W$ with Rwv.
- $\mathfrak{M}, w \vDash \langle gsw \rangle \phi$, if $\mathfrak{M}^*_{\{(v,u)\}}, w \vDash \phi$ for some $v \in W$ with Ruv.
- $\mathfrak{M}, w \vDash \langle br \rangle \phi$, if $\mathfrak{M}^+_{\{(w,v)\}}, v \vDash \phi$ for some $v \in W$ with $(w, v) \notin R$.
- $\mathfrak{M}, w \vDash \langle gbr \rangle \phi$, if $\mathfrak{M}^+_{\{(u,v)\}}, w \vDash \phi$ for some $v \in W$ with $(u, v) \notin R$.

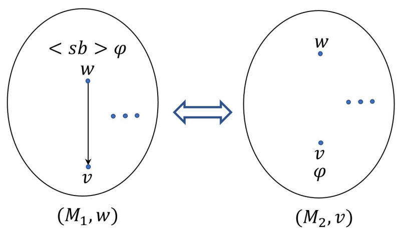


Figure 2.2 $\langle sb \rangle \varphi$

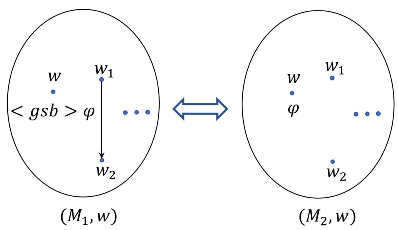


Figure 2.3 $\langle gsb \rangle \varphi$

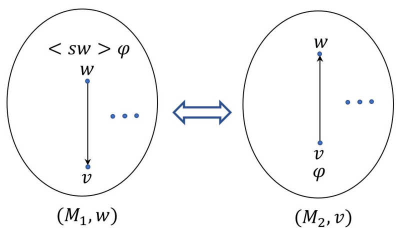


Figure 2.4 $\langle sw \rangle \varphi$

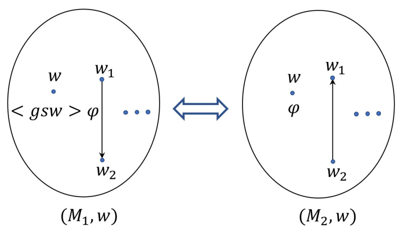


Figure 2.5 $\langle gsw \rangle \varphi$

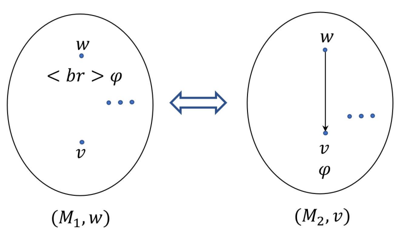


Figure 2.6 $\langle br \rangle \varphi$

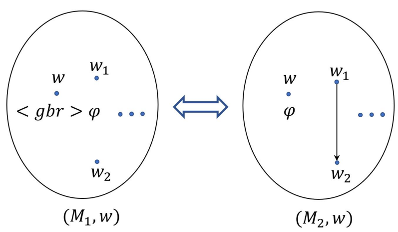


Figure 2.7 $\langle gbr \rangle \varphi$

Note that the modality $\langle gsb \rangle$ is the modality $\blacklozenge$ mentioned in Section 2.2, thus $MCML(gsb)$ is exactly sabotage modal logic. Next, we give the definition of $\langle up \rangle$-bisimulation for $\langle up \rangle \in \{\langle sb \rangle, \langle gsb \rangle, \langle sw \rangle, \langle gsw \rangle, \langle br \rangle, \langle gbr \rangle\}$.

Definition 2.10 ($\langle up \rangle$-bisimulation): Given two models $\mathfrak{M} = (W, R, V)$ and $\mathfrak{M}' = (W', R', V')$, we say that (w, S) and (w', S') are $\langle up \rangle$-bisimilar, if there is a nonempty relation $Z \subseteq (W \times \mathcal{P}(W^2)) \times (W' \times \mathcal{P}(W'^2))$ satisfying:

- $Atom$: $w \in V(p)$ iff $w' \in V'(p)$
- Zig: if $(w, v) \in S$ then for some v', $(w', v') \in S'$ and $(v, S)Z(v', S')$
- Zag: if $(w', v') \in S'$ then for some v, $(w, v) \in S$ and $(v, S)Z(v', S')$
- $Zig_{\langle up \rangle}$ and $Zag_{\langle up \rangle}$ are given for specific cases as follows:

- $Zig_{\langle sb \rangle}$: if $(w, v) \in S$, then $(v, S^-_{\{(w,v)\}})Z(v', S'^-_{\{(w',v')\}})$ for some v'
- $Zag_{\langle sb \rangle}$: if $(w', v') \in S'$ then $(v, S^-_{\{(w,v)\}})Z(v', S'^-_{\{(w',v')\}})$ for some v
- $Zig_{\langle gsb \rangle}$: if $(u, v) \in S$, then $(w, S^-_{\{(u,v)\}})Z(w', S'^-_{\{(u',v')\}})$ for some (u', v')
- $Zag_{\langle gsb \rangle}$: if $(u', v') \in S'$, then $(w, S^-_{\{(u,v)\}})Z(w', S'^-_{\{(u',v')\}})$ for some (u, v)
- $Zig_{\langle sw \rangle}$: if $(w, v) \in S$, then $(v, S^*_{\{(v,w)\}})Z(v', S'^*_{\{(v',w')\}})$ for some v'
- $Zag_{\langle sw \rangle}$: if $(w', v') \in S'$, then $(v, S^*_{\{(v,w)\}})Z(v', S'^*_{\{(v',w')\}})$ for some v
- $Zig_{\langle gsw \rangle}$: if $(u, v) \in S$, then $(w, S^*_{\{(v,u)\}})Z(w', S'^*_{\{(v',u')\}})$ for some (u', v')
- $Zag_{\langle gsw \rangle}$: if $(u', v') \in S'$, then $(w, S^*_{\{(v,u)\}})Z(w', S'^*_{\{(v',u')\}})$ for some (u, v)
- $Zig_{\langle br \rangle}$: if $(w, v) \notin S$, then $(v, S^+_{\{(w,v)\}})Z(v', S'^*_{\{(w',v')\}})$ for some v'
- $Zag_{\langle br \rangle}$: if $(w', v') \notin S'$, then $(v, S^*_{\{(w,v)\}})Z(v', S'^*_{\{(w',v')\}})$ for some v
- $Zig_{\langle gbr \rangle}$: if $(u, v) \notin S$, then $(w, S^+_{\{(u,v)\}})Z(w', S'^+_{\{(u',v')\}})$ for some (u', v')
- $Zag_{\langle gbr \rangle}$: if $(u', v') \notin S'$, then $(w, S^+_{\{(u,v)\}})Z(w', S'^+_{\{(u',v')\}})$ for some (u, v)

Fact 2: Given two models $\mathfrak{M} = (W, R, V)$ and $\mathfrak{M}' = (W', R', V')$, and let $w \in W, w' \in W', S \subseteq W^2$ and $S' \subseteq W'^2$. Suppose there is an $\langle up \rangle$-bisimulation Z such that $(w, S)Z(w', S')$, then for any formula ϕ in $MCML(up)$, $(W, S, V), w \vDash \phi$ iff $(W', S', V'), w' \vDash \phi$.

As mentioned, the modality $\langle gsb \rangle$ is identical to $\blacklozenge$. Still, even if we fix $\langle up \rangle$ as $\langle gsb \rangle$, this proposition is not the same as Proposition 1 due to the differences in corresponding definitions of bisimulation.

All model-changing modal logics above aim to capture model transforms with relational changes. We also pay attention to other modal-changing logics, which allow for changing the valuation in models, such as local fact change logic (Thompson, 2020).

Definition 2.11 (Language of LFC): Let $Prop$ denote a nonempty countable set of proposition letters. The language of Local Fact Change Logic (LFC) is defined as follows:

$$\varphi ::= p \mid \neg\varphi \mid \varphi \wedge \varphi \mid \Diamond\varphi \mid \langle ch \rangle\varphi$$

where $p \in \mathsf{Prop}$. The standard boolean connectives, $\Box$, are defined as usual. $\langle ch \rangle$ is a model-changing operator for changing valuations. The formula $\langle ch \rangle \varphi$ can be read "it is the case that φ after the valuation of current is assigned by some set of proposition letters". The corresponding satisfaction clauses are as follows, and the intuitive meaning is shown in Figure 2.8.

- $(W, R, V), w \vDash \langle ch \rangle \varphi$ if $(W, R, V_A^w), w \vDash \varphi$ for some $A \subseteq Prop$, where the function V_A^w is the same to V except that V_A^w assigns the set A to w.

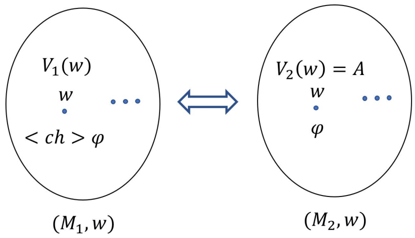


Figure 2.8 $\langle ch \rangle \varphi$

Definition 2.12 ($\langle ch \rangle$-bisimulation): Given two frames $\mathfrak{F} = (W_1, R_1)$ and $\mathfrak{F} = (W_1, R_1)$. A nonempty relation $Z \subseteq (W_1 \times Prop^{W_1}) \times (W_2 \times Prop^{W_2})$ is $\langle ch \rangle$-bisimulation if it satisfies:

- If $(s_1, V_1) Z (s_2, V_2)$ then $V_1(s_1) = V_2(s_2)$
- If $(s_1, V_1) Z (s_2, V_2)$ and $R_1 s_1 t_1$, then $(t_1, V_1) Z (t_2, V_2)$ for some t_2 with $R_2 s_2 t_2$
- If $(s_1, V_1) Z (s_2, V_2)$ and $R_2 s_2 t_2$, then $(t_1, V_1) Z (t_2, V_2)$ for some t_1 with $R_1 s_1 t_1$
- If $(s_1, V_1) Z (s_2, V_2)$ then $(s_1, V1_A^{s_1}) Z (s_2, V1_A^{s_2})$ for every $A \subseteq Prop$.

Fact 3: Given two frames $\mathfrak{F}_1 = (W_1, R_1)$ and $\mathfrak{F}_2 = (W_2, R_2)$. For $w_1 \in W_1$, $w_2 \in W_2$, if there exists a $\langle ch \rangle$-bisimulation Z such that $(w_1, V_1) Z (w_2, V_2)$, then $\mathfrak{F}_1, V_1, w_1 \vDash \phi$ iff $\mathfrak{F}_2, V_2, w_2 \vDash \phi$ for any formula ϕ in LFC.

There is also a logic capturing point deletions in models, i.e., the modal logic of stepwise removal (van Benthem et al., 2020).

Definition 2.13 (Language of MLSR): Let Prop be a nonempty countable set of propositional variables. The language of modal logic of stepwise removal (MLSR) is defined by the following grammar:

$$\varphi ::= p \mid \neg\varphi \mid \varphi \wedge \varphi \mid \Diamond\varphi \mid \langle -\varphi \rangle \varphi \mid E\varphi$$

where $p \in \mathsf{Prop}$, E is the existential modality, and $\langle -\varphi \rangle$ is the stepwise update modality. The formula $\langle -\varphi \rangle \phi$ can be read "it is the case that ϕ after some point having the property that φ (which is different from the current point) is deleted", and the corresponding satisfaction clauses are as follows.

- $\mathfrak{M}, w \models E\phi$ if there is a world v such that $\mathfrak{M}, v \models \phi$
- $\mathfrak{M}, w \models \langle -\varphi \rangle \phi$, if there is a world $v \neq w$ in $\mathfrak{M}$ with $\mathfrak{M}, v \models \varphi$ such that $\mathfrak{M}^-_v, w \models \phi$, where $\mathfrak{M}^-_v$ is the submodel of $\mathfrak{M}$ having just the world v removed from its domain.

We have used the techniques employed in (van Benthem et al., 2020) to develop a proof system in Chapter 3. Moreover, inspired by the stepwise update modality, we simplified the same as $\langle de \rangle$ given below, allowing to delete points without the formula satisfaction property, and propose the point sabotage logic below. In Chapter 5, we study the complexity of the model comparison problem for this logic.

Definition 2.14: Let Prop be a nonempty countable set of propositional variables. The language of the point sabotage logic PSL is defined by the following grammar:

$$\varphi ::= p \mid \neg\varphi \mid \varphi \wedge \varphi \mid \Diamond\varphi \mid \langle de \rangle \varphi$$

where $p \in$ Prop, and $\langle de \rangle$ is a model-changing operator for changing the domain. The formula $\langle de \rangle \varphi$ can be read "it is the case that φ after some point (which is different from the current point) is deleted". The corresponding satisfaction clauses are as follows, and the intuitive meaning is shown in Figure 2.9.

Given a model $\mathfrak{M} = (W, R, V)$ and a world $w \in W$, $\mathfrak{M}^-_w = (W', R', V')$ is induced from $\mathfrak{M}$, where $W' = W \setminus \{w\}, R' = R \setminus \{(u, v) \in R \mid u = w \text{ or } v = w\}, V'(p) = V(p) \setminus \{w\}$.

- $\mathfrak{M}, w \models \langle de \rangle \phi$, if there is a world $v \neq w$ in $\mathfrak{M}$ such that $\mathfrak{M}^-_v, w \models \phi$.

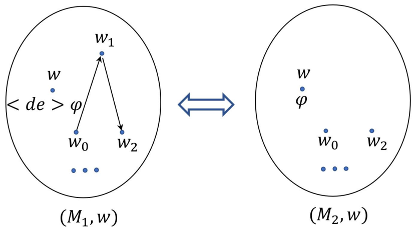


Figure 2.9 $\langle de \rangle \varphi$

Definition 2.15 ($\langle de \rangle$-bisimulation): Given two pointed models $(\mathfrak{M}_1, w_1) = ((W_1, R_1, V_1), w_1)$ and $(\mathfrak{M}_2, w_2) = ((W_2, R_2, V_2), w_2)$. A non-empty relation Z is $\langle de \rangle$-bisimulation, if it satisfies the following conditions.

- Atom: If $(\mathfrak{M}_1, w_1) Z (\mathfrak{M}_2, w_2)$, then $(\mathfrak{M}_1, w_1) \models p$ iff $(\mathfrak{M}_2, w_2) \models p$ for all atomic propositions p.

- Zig$_{\Diamond}$: If $(\mathfrak{M}_1, w_1)Z(\mathfrak{M}_2, w_2)$, and there exists $u \in W_1$ such that $w_1 R_1 v_1$, then there is a $v_2 \in W_2$ such that $w_2 R_2 v_2$ and $(\mathfrak{M}_1, w_1)Z(\mathfrak{M}_2, w_2)$.
- Zag$_{\Diamond}$: Same as above in the converse direction.
- Zig$_{\langle de \rangle}$: If $(\mathfrak{M}_1, w_1)Z(\mathfrak{M}_2, w_2)$, and $u \in W_1$ with $u \neq w_1$, then there is a world $v \in W_2$ and $v \neq w_2$ such that $(\mathfrak{M}_{1_u}^-, w_1)Z(\mathfrak{M}_{2_v}^-, w_2)$.
- Zag$_{\langle de \rangle}$: Same as above in the converse direction.

Fact 4: Given two pointed models $(\mathfrak{M}_1, w_1)$, $(\mathfrak{M}_2, w_2)$, if there is a $\langle de \rangle$-bisimulation, then for any formula ϕ in PSL, $\mathfrak{M}_1, w_1 \vDash \phi$ iff $\mathfrak{M}_2, w_2 \vDash \phi$.

2.4 Dynamic epistemic logics

In this section, we introduce Public Announcement Logic (Plaza, 1989, 2007) and then standard generalized communication operations of action models (Baltag and Moss, 2004; van Ditmarsch et al., 2008; van Benthem, 2011). We end this subsection with a variant of the action model, i.e., the edge-conditioned model (Bolander, 2018). All the results mentioned in this section come directly from the above mentioned papers.

Definition 2.16 (Language of PAL): Let G denote a finite nonempty set of agents, and $Prop$ denote a nonempty countable set of proposition letters. The language of Public Announcement Logic (PAL) is defined as follows:

$$\varphi ::= p \mid \neg\varphi \mid \varphi \wedge \varphi \mid [a]\varphi \mid [\varphi]\varphi$$

where $p \in \mathsf{Prop}, a \in G$. Other boolean operators are defined as usual, $\langle a \rangle \phi := \neg[a]\neg\phi$, and $\langle \varphi \rangle \phi := \neg[\varphi]\neg\phi$. Roughly speaking, $[a]\phi$ can be read "agent a knows that ϕ" (epistemic reading) or "agent a believes that ϕ" (doxastic reading). We adopt the latter throughout this dissertation. Additionally, $[\varphi]\phi$ can be read "after the public announcement of φ, it is still the case that ϕ". We present the formal reading as follows.

Definition 2.17 (Epistemic model): An epistemic model $\mathfrak{M}$ is a tuple $(W, \{R_a\}_{a \in G}, V)$, where

- W is a non-empty set of states,
- $R_a \subseteq W \times W$ for each $a \in G$,
- V is a valuation function from Prop to 2^W.

We call $(\mathfrak{M}, w)$ a pointed epistemic model when $\mathfrak{M}$ is an epistemic model and w is a state on it. The truth conditions for boolean formulas are as above, and we only show those for modal operators.

- $\mathfrak{M}, w \vDash [a]\phi$, if for all v such that $wR_a v : \mathfrak{M}, v \vDash \phi$
- $\mathfrak{M}, w \vDash [\varphi]\phi$, if for all $(\mathfrak{M}', w')$ such that $(\mathfrak{M}, w)[\![\varphi]\!](\mathfrak{M}', w') : \mathfrak{M}', w' \vDash \phi$, where $(\mathfrak{M}, w)[\![\varphi]\!](\mathfrak{M}', w')$ iff $w' = w$ and $\mathfrak{M}' = (W', \{R'\}_{a\in G}, V')$ is defined as follows:
  - $W' = \{w \in W \mid M, w \vDash \varphi\}$
  - $R'_a = R_a \cap W'$
  - $V'(p) = V(p) \cap W'$

Definition 2.18 (Axiomatization of PAL):

- all instantiations of propositional tautologies
- $[a](\phi \to \varphi) \to ([a]\phi \to [a]\varphi)$
- $[\varphi]p \leftrightarrow (\varphi \to p)$ for $p \in \mathsf{Prop}$
- $[\varphi](\alpha \wedge \beta) \leftrightarrow ([\varphi]\alpha \wedge [\varphi]\beta)$
- $[\varphi](\neg\phi) \leftrightarrow (\varphi \to \neg[\varphi]\phi)$
- $[\varphi][a]\phi \leftrightarrow (\varphi \to [a][\varphi]\phi)$
- modus ponens: from φ and $\varphi \to \phi$, infer ϕ
- necessitation Rule for $[a]$: from ϕ, infer $[a]\phi$
- necessitation Rule for $[\varphi]$: from ϕ, infer $[\varphi]\phi$

Note that we only show the minimal version of PAL, we shall add more axioms if we work on different class of models. Fox example, if we would like to capture the class of serial models for belief, i.e., models satisfying the condition that for any agent $a \in G$ and a state w in the model, there exists a state v with $wR_a v$, then we add the axiom $[a]\varphi \to \langle a\rangle\varphi$.

This axiom system is efficient, the reduction axioms provide a way of rewriting formulas equivalently to make them 'simpler'. Formally, we have the following theorem.

Theorem 2.1 (PAL reduction theorem): For any formula of PAL, there is an equivalent formula without public announcement operators.

The proof strategy is to design a truth-preserving translation from the language of PAL to the sublanguage without the public announcement operator, the completeness of PAL follows. This technique is developed in many logics for the completeness, for example, the epistemic action logic, we will see later.

Theorem 2.2 (Soundness and completeness of PAL): PAL is sound and complete with respect to all epistemic models.

PAL captures the scenarios with public announcements, more specifically, $[\![\varphi]\!]$ in the truth condition for $[\varphi]\phi$ describes the model-transforming relation before and after the public announcement. For more complex scenarios, 'action models' or 'event models' are introduced.

Definition 2.19 (Action/Event model): Let G denote a finite nonempty set of agents, $\mathcal{L}$ be the language of epistemic action logic, which we define later. An action model $\mathcal{A}$ is a tuple $(E, \{S_a\}_{a\in G}, Pre)$, where

- E is a nonempty finite set of events,
- $S_a \subseteq E \times E$ for each $a \in G$,
- $Pre : E \to \mathcal{L}$ assigns to each event e a precondition $Pre(e) \in \mathcal{L}$.

We call $(\mathcal{A}, e)$ a pointed action model when $\mathcal{A}$ is an action model and e is an event on it. If the precondition of event e is satisfiable at a pointed epistemic model, then we can produce a resultant situation via the following model-transforming mechanism.

Definition 2.20 (Updated model): Given a point model $(\mathfrak{M}, w)$ and a pointed action model $(\mathcal{A}, e)$. If $\mathfrak{M}, w \vDash Pre(e)$, then the updated model $\mathfrak{M}'$ is a tuple (W', R', V'), where

- $W' = \{(w, e) : M, w \vDash Pre(e)\}$,
- $(w, e)R'_a(v, f)$ iff $R_a wv$ and $S_a ef$,
- $(w, e) \in V'(p)$ iff $w \in V(p)$.

Definition 2.21 (Language of EAL): Let G denote a finite nonempty set of agents, and $Prop$ denote a nonempty countable set of proposition letters. The language of epistemic action logic (EAL) is defined as follows:

$$\varphi ::= p \mid \neg\varphi \mid \varphi \wedge \varphi \mid [a]\varphi \mid [\mathcal{A}, e]\varphi$$

where $p \in \mathsf{Prop}$, $a \in G$, $\mathcal{A}$ is an action model defined above, and e is an event in $\mathcal{A}$. Other boolean operators are defined as usual, and $\langle \mathcal{A}, e\rangle\phi := \neg[\mathcal{A}, e]\neg\phi$. $[\mathcal{A}, e]\phi$ can be read that if the event e occurs, then after that, it is the case that ϕ. We present the formal clauses as follows.

We only show the clause for $[\mathcal{A}, e]$, others are the same as the above.

- $\mathfrak{M}, w \vDash [\mathcal{A}, e]\phi$, if for all $(\mathfrak{M}', w')$ such that $(\mathfrak{M}, w)[\![\mathcal{A}, e]\!](\mathfrak{M}', w') : \mathfrak{M}', w' \vDash \phi$, where $(\mathfrak{M}, w)[\![\mathcal{A}, e]\!](\mathfrak{M}', w')$ iff $w' = (w, e)$ and $\mathfrak{M}' = (W', \{R'\}_{a\in G}, V')$ is the updated model from $(\mathfrak{M}, w)$ and $(\mathcal{A}, e)$.

Definition 2.22 (Axiomatization of EAL):

- all instantiations of propositional tautologies
- $[a](\phi \to \varphi) \to ([a]\phi \to [a]\varphi)$
- $[\mathcal{A}, e]p \leftrightarrow (Pre(e) \to p)$ for $p \in Prop$
- $[\mathcal{A}, e](\varphi \wedge \phi) \leftrightarrow ([\mathcal{A}, e]\varphi \wedge [\mathcal{A}, e]\phi)$
- $[\mathcal{A}, e](\neg\phi) \leftrightarrow (Pre(e) \to \neg[\mathcal{A}, e]\phi)$
- $[\mathcal{A}, e][a]\phi \leftrightarrow (Pre(e) \to \bigwedge_{f:S_aef}[a][\mathcal{A}, f]\phi)$
- modus ponens: from α and $\alpha \to \beta$, infer β
- necessitation Rule of $[a]$: from ϕ, infer $[a]\phi$
- necessitation Rule of $[\mathcal{A}, e]$: from ϕ, infer $[\mathcal{A}, e]\phi$

Similarly, when we would like to characterize more properties of knowledge or belief, we shall add more axioms to extend the axiom system. Moreover, we also have a reduction theorem for EAL as follows.

Theorem 2.3 (EAL reduction theorem): For any formula of EAL, there is an equivalent formula without action model modalities.

The proof strategy is similar to the case for PAL, with this theorem, the completeness of EAL follows.

Theorem 2.4 (Soundness and completeness of EAL): EAL is sound and complete with respect to all epistemic models.

There are many variants of the action model approach to epistemic model transformation, we only mention a variant involved in this dissertation.

Definition 2.23 (Edge-conditioned event model): An edge-conditioned event model ε is a tuple (E, Q, Pre), where E, Pre are defined as for standard event models, and $Q_i : E \times E \to \mathcal{L}(P, G)$ for each $i \in G$.

Definition 2.24 (Edge-conditioned updated model): Given a pointed epistemic model (M, w) and a pointed edge-conditioned event model (ε, e), and $M, w \vDash Pre(e)$. The edge-conditioned updated model M^ε is a tuple (W^ε, R', V'), where W^ε, V' are defined as for standard updated models, and for each $a \in G$,

- $W^\varepsilon = \{(w, e) : M, w \vDash Pre(e)\}$,
- $(w, e)R'_a(v, f)$ iff $R_a wv$ and $M, w \vDash Q_a(e, f)$,

- $(w, e) \in V'(p)$ iff $w \in V(p)$.

We will prove that the class of updates expressible using action models is the same as those expressible using edge-conditioned updates in Appendix 6.4.

2.5 Complexity classes

Throughout the dissertation, we use the standard definition of complexity classes, which can be found in (Arora and Barak, 2009). Intuitively, a complexity class is a collection of 'efficiently computable' functions, that is, functions on finite bit sequences that can be computed within some finite amount of resource(s).

Let $\mathbb{N}$ denote the set of natural numbers, and T, S are functions from $\mathbb{N}$ to $\mathbb{N}$, and $L \subseteq \{0,1\}^*$ denote a language. Let $\mathbf{DTIME}(T(n))$ ($\mathbf{NTIME}(T(n))$) denote the set of all Boolean functions that are computable by $c \times T(n)$-time deterministic (non-deterministic) Turing machine for some constant $c > 0$. Some complexity classes that were mentioned in the course of discussion in this dissertation are as follows.

- The class $\mathbf{P} = \bigcup_{c>0} \mathbf{DTIME}(n^c)$.
- The class $\mathbf{NP} = \bigcup_{c>0} \mathbf{NTIME}(n^c)$.
- The class $\mathbf{EXP} = \bigcup_{c>0} \mathbf{DTIME}(2^{n^c})$.
- The class $\mathbf{PSPACE} = \bigcup_{c>0} \mathbf{SPACE}(n^c)$, where $L \in \mathbf{SPACE}(S(n))$ if there is a constant c and a deterministic Turing Machine M deciding L such that on every input $x \in \{0,1\}^*$, the total number of locations that are at some point non-blank during M's execution on x is at most $c \times s(|x|)$.
- The class $\mathbf{NPSPACE} = \bigcup_{c>0} \mathbf{NSPACE}(n^c)$, where $L \in \mathbf{NSPACE}(S(n))$ if there is a constant c and a non-deterministic Turing Machine M deciding L such that on every input $x \in \{0,1\}^*$, the total number of locations that are at some point non-blank during M's execution on x is at most $c \times s(|x|)$.

The relationships among the above-mentioned complexity categories can be summarised as follows.

Fact 5: $\mathbf{P} \subseteq \mathbf{NP} \subseteq \mathbf{PSPACE} = \mathbf{NPSPACE} \subseteq \mathbf{EXP}$

Before rounding off, let us mention another important concept related to the investigations on complexity of different classes of decision problems that is often used to deal with computational hardness.

Definition 2.25: A language $\mathcal{L}_1$ is polynomial-time reducible to another language $\mathcal{L}_2$, denoted by $\mathcal{L}_1 \leq_p \mathcal{L}_2$, if there is a polynomial-time computable function $f : \{0,1\}^* \rightarrow \{0,1\}^*$ such that for every $x \in \{0,1\}^*$, $x \in \mathcal{L}_1$ if and only if $f(x) \in \mathcal{L}_2$.

Definition 2.26: Given a complexity class $\mathcal{C}$, we say that a language $\mathcal{L}$ is $\mathcal{C}$-hard if $\mathcal{L}' \leq_p \mathcal{L}$ for every $\mathcal{L}' \in \mathcal{C}$. We say $\mathcal{L}$ is $\mathcal{C}$-complete if $\mathcal{L}$ is $\mathcal{C}$-hard and $\mathcal{L} \in \mathcal{C}$.

In general, suppose that $\mathcal{L}$ is $\mathcal{C}$-hard, we say the language $\mathcal{L}$ is at least as hard as any other language in $\mathcal{C}$. Moreover, if $\mathcal{L}$ is $\mathcal{C}$-complete, we say $\mathcal{L}$ is the most difficult language in the complexity class $\mathcal{C}$. Some well-known results on complexity classes (Sipser, 1996; Halpern and Moses, 1992) are listed below.

- **P**:
  - the PATH problem
  - the model comparison problem for the minimal modal logic
- **NP**-complete:
  - the satisfiability problem (SAT) for propositional logic
  - the Hamiltonian path (HAPATH) problem
- **PSPACE**-complete:
  - the satisfiability problem (SAT) for the minimal modal logic
  - the true quantified Boolean formula (TQBF) problem

CHAPTER 3 HYBRID SABOTAGE MODAL LOGIC

Sabotage games were introduced in (van Benthem, 2005) as a model for algorithmic behavior under disturbance, a topic of increasing interest when analyzing abuses of and threats to computational systems such as the Internet. The idea is that in a task involving stepwise traversal of a graph by a player called ‘Traveler’, the disturbing influence becomes a counter-player called ‘Demon’ who starts each round by cutting some available link in the graph. The resulting sabotage game is determined, and winning conditions and winning invariants can be defined in a natural associated modal logic SML which has a standard modality for accessible nodes from the current point as well as a new ‘deletion modality’ describing what is true at the current point after some link has been deleted from the graph.

There is a strand of literature exploring applications and technical properties of sabotage games and their modal logic. Löding and Rohde (2003b) proved that model checking for SML is Pspace-complete, while satisfiability is undecidable. Aucher et al. (2018) gave a bisimulation-style characterization of SML under translation as a fragment of first-order logic, as well as a complete tableau system for validity, and similar results were obtained independently in (Areces et al., 2016) in a more general study of modal logics of graph change. More recent results include (Li, 2020) on sabotage modal logics with definable link deletions, and a Zero-One Law for SML, (Mierzewski, 2018), showing that in the long run as finite graph size increases, the sabotage game is massively in favor of Traveler, who wins at any position with probability 1. In terms of applications of the game, one interesting proposal using sabotage games for learning scenarios is found in (Gierasimczuk et al., 2009). The background to these publications is a more general investigation of the connections between modal logics and existing or newly designed graph games, advocated in the programmatic paper (van Benthem and Liu, 2020), with concrete case studies in (Zaffora Blando et al., 2020; Grossi and Rey, 2019) on ‘poison games’, and (Thompson, 2020) on modal logics of fact change.

A natural and straightforward issue left open in this literature is a *Hilbert-style axiomatization* of SML, which would be useful for actual standard reasoning about sabotage games or related dynamic scenarios. Such an axiomatization must exist by the known effective translation of SML into first-order logic, but finding a concrete workable proof system has turned out surprisingly difficult. The present paper fills this gap, at least for a mild hybrid modal extension of the original SML language called HSML, and explores some broader implications of this result. The technique used for our completeness theorem stems from a recent axiomatization of a basic modal logic MLSR for stepwise object deletion (or alternatively, of 'quantification without replacement') in (van Benthem et al., 2020), that we adapt to the sabotage setting, and simplify considerably.

Once we have the connection between the standard semantics of SML and the proof system in our completeness proof, a natural follow-up question arises. Can one modulate this relationship between semantics and proof system so as to get completeness for other natural semantics for modal logics of graph change? We show how this can be done for a new *'protocol version'*, (Hoshi, 2014), of SML that restricts the available deletions for Demon. Next, we turn to the general issue of relating modal logics for deleting edges and for deleting vertices from graphs. We embed the sabotage logic HSML faithfully into MLSR by encoding edge deletion as vertex deletion, and also provide a partial converse. We end with identifying a few further topics that seem amenable to our style of analysis, including interpolation for HSML and axiomatizing its schematic validities.

Relation to DEL For readers familiar with dynamic-epistemic logic(DEL), (van Ditmarsch et al., 2008; Baltag and Renne, 2016; van Benthem, 2011), an analogy may be helpful. A system like 'public announcement logic' (PAL) has modalities for actions $!\varphi$ of deleting all points that satisfy $\neg\varphi$ from a given graph model. PAL is decidable thanks to 'recursion axioms' that push dynamic modalities through complex postconditions. However, if we perform deletions step by step, we get the above logic MLSR which is undecidable, (van Benthem et al., 2020), since arbitrary sequences of deletions require storage in an unbounded memory, a device allowing for encoding of undecidable computational problems. The situation is similar with link deletions. There are complete and decidable dynamic-epistemic logics for uniform definable link cutting (an example of such a system occurs in the Appendices to this paper), but in contrast, SML and HSML maintain sequences of arbitrary stepwise link deletions that require memory, and thus incur higher complexity. Even so, research questions about SML show many similarities with those for PAL and MLSR. One might even think that the link deletion case is essentially the same subject as the point deletion case, but more precise information on the true connections will be found in Section 5 below.

Relation to hybrid modal logic In this paper, we employ devices from hybrid logic, (Areces and ten Cate, 2007), to boost the expressive power of the original sabotage modal logic just enough to allow for a Hilbert-style axiomatization. However, this choice of a surplus is not unique. We focus on nominals plus the @-operator as a convenient syntax, but a version of SML extended with nominals and global existential and universal modalities would also be worth investigating. Moreover, we just determine the most general logic of the above games. Sabotage logics for specific classes of graphs may well be axiomatizable using further proof-theoretic techniques from hybrid logic, such as those presented in (Blackburn and Ten Cate, 2006). Finally, one could also turn the tables, and in the spirit of (Areces et al., 2016), view our results from a hybrid perspective as exploring fragments of the full first-order language that arise as hybrid languages are enriched with modalities for various forms of graph change.

3.1 Hybrid sabotage modal logic

3.1.1 Language and semantics

We start by introducing the basic notions of the system HSML. For details of modal logic that we do not explain, we refer to (Blackburn et al., 2001).

Definition 3.1 (Language): Let $\mathsf{Prop} = \{p, q, r, \ldots\}$ be a nonempty countable set of propositional variables disjoint from a nonempty countable set of nominals $\mathsf{Nom} = \{a, b, c, d, \ldots\}$. The *hybrid modal sabotage language HSML* is defined over the set of atoms $\mathsf{Prop} \cup \mathsf{Nom}$ by the following grammar:

$$\varphi ::= a \mid p \mid \bot \mid \neg\varphi \mid \varphi \wedge \varphi \mid \Diamond\varphi \mid \blacklozenge\varphi \mid @_a\varphi$$

Definition 3.2 (Model): A *model* $\mathfrak{M} = (W, R, V)$ for HSML is a standard modal relational model with worlds W, accessibility relation R and valuation function V, subject to the condition that V assigns singleton subsets of W to nominals.

Definition 3.3 (Truth conditions): The semantics of HSML is as follows:

$\mathfrak{M}, w \vDash a$	*iff*	$w \in V(a)$
$\mathfrak{M}, w \vDash p$	*iff*	$w \in V(p)$
$\mathfrak{M}, w \vDash @_a\varphi$	*iff*	$\mathfrak{M}, v \vDash \varphi$ where $V(a) = \{v\}$
$\mathfrak{M}, w \vDash \neg\varphi$	*iff*	not $\mathfrak{M}, w \vDash \varphi$
$\mathfrak{M}, w \vDash \varphi \wedge \phi$	*iff*	$\mathfrak{M}, w \vDash \varphi$ *and* $\mathfrak{M}, w \vDash \phi$
$\mathfrak{M}, w \vDash \Diamond\varphi$	*iff*	$\mathfrak{M}, v \vDash \varphi$ for some v with Rwv

$\mathfrak{M} = (W, R, V), w \vDash \blacklozenge\varphi$ *iff* there is a pair $(u, v) \in R$ such that $(W, R \setminus (u, v), V), w \vDash \varphi$.

The deletion diamond modality of SML and its universal dual $\blacksquare = \neg\blacklozenge\neg$ describes effects of cutting arbitrary links, one at a time, allowing one to express, e.g., winning patterns for Traveler in sabotage games by modal combinations $\blacksquare\Diamond$. For more on the expressive power of this device, cf. (Aucher et al., 2018).

However, using nominals, we can define still more, in particular, the following useful operator describing the effect of cutting a specific named link:

$$\langle a|b\rangle\varphi := (@_a \Diamond b \wedge \blacklozenge(@_a\neg\Diamond b \wedge \varphi)) \vee (@_a\neg\Diamond b \wedge \varphi)$$

Informally, this formula says that after cutting a possibly existent link between the world named "a" and the world named "b", φ will hold. The first disjunct describes the effects of actually cutting such a link, the second disjunct takes care of the case that no link connected a and b. Formally, let $\mathfrak{M} = (W, R, V)$, $\mathfrak{M}^{(a|b)} = (W, R^{(a|b)}, V)$, where $R^{(a|b)} = R \setminus \{(u, v) \,|\, \mathfrak{M}, u \vDash a \text{ and } \mathfrak{M}, v \vDash b\}$. Unpacking the above truth conditions, it is easy to see that the following holds:

Fact 6: $\mathfrak{M}, w \vDash \langle a|b\rangle\varphi$ iff $\mathfrak{M}^{(a|b)}, w \vDash \varphi$.

In what follows, we will often need finite sequences of link cuts, and accordingly, we will use the notation $\mathfrak{M}^{\overline{(a|b)}_n}$ for the model $(((\mathfrak{M}^{(a_1|b_1)})^{(a_2|b_2)})^{\cdots})^{(a_n|b_n)}$ and $\overline{\langle a|b\rangle}_n\varphi$ for the formula $\langle a_1|b_1\rangle \ldots \langle a_n|b_n\rangle\varphi$ when $n \geq 1$. Moreover, in the special case of $n = 0$ we let $\mathfrak{M}^{\overline{(a|b)}_n}$ denote $\mathfrak{M}$ while $\overline{\langle a|b\rangle}_n\varphi$ denotes φ.

3.1.2 A proof system for HSML

Using our named link-cutting device, we now present the proof system **HSML** in Table 3.1. Its first module consists of standard axioms and derivation rules from the minimal modal logic with hybrid additions, (Areces and ten Cate, 2007), the second module is the usual minimal modal logic for the sabotage modality, the third module contains dynamic-epistemic style recursion axioms for definable link cutting, because of which the logic is not closed under uniform substitution, and the fourth module contains the crucial derivation rule connecting the deletion modality and the named link cutting modality.[①]

The first two modules drive standard modal completeness arguments, the third and

① We will often make tacit appeals to a proof rule of Replacement of Equivalents in what follows, but this is derivable in the system **HSML** as presented here.

Axioms and rules for basic hybrid modal logic	All tautologies of classical propositional logic, plus Modus Ponens All axioms of the minimal modal logic for $\Box$, plus the Necessitation Rule Axioms and rules of hybrid logic for $@_a$: $@_a\varphi \leftrightarrow \neg @_a \neg\varphi, \quad a \wedge \varphi \rightarrow @_a\varphi, \quad @_a a, \quad @_a b \leftrightarrow @_b a$ $@_a b \wedge @_b\varphi \rightarrow @_a\varphi, \quad @_b @_a\varphi \leftrightarrow @_a\varphi, \quad \Diamond @_a\varphi \rightarrow @_a\varphi$ $(Name): \dfrac{c \rightarrow \varphi}{\varphi}(c \notin \varphi)$ $(Paste): \dfrac{@_a \Diamond b \wedge @_b\varphi \rightarrow \delta}{@_a \Diamond \varphi \rightarrow \delta}$ ($b \notin \varphi$, δ and a are distinct from b)
Distribution Axiom for $\blacksquare$ Necessitation Rule for $\blacksquare$	$\blacksquare(\phi \rightarrow \psi) \rightarrow (\blacksquare\phi \rightarrow \blacksquare\psi)$ $\dfrac{\varphi}{\blacksquare\varphi}$
Recursion axioms for $\langle a\|b\rangle$	$\langle a\|b\rangle c \leftrightarrow c$ $\langle a\|b\rangle p \leftrightarrow p$ $\langle a\|b\rangle \neg\varphi \leftrightarrow \neg\langle a\|b\rangle\varphi$ $\langle a\|b\rangle(\varphi \wedge \psi) \leftrightarrow (\langle a\|b\rangle\varphi \wedge \langle a\|b\rangle\psi)$ $\langle a\|b\rangle @_c\varphi \leftrightarrow @_c\langle a\|b\rangle\varphi$ $\langle a\|b\rangle \Diamond \varphi \leftrightarrow ((a \wedge \Diamond(\neg b \wedge \langle a\|b\rangle\varphi)) \vee (\neg a \wedge \Diamond\langle a\|b\rangle\varphi))$
Inference rule for $\blacklozenge$, $\langle a\|b\rangle$	(B-Mix) : $\dfrac{@_c\overline{\langle a\|b\rangle}_n(@_{a_{n+1}} \Diamond b_{n+1} \wedge \langle a_{n+1}\|b_{n+1}\rangle\varphi) \rightarrow \theta}{@_c\overline{\langle a\|b\rangle}_n\blacklozenge\varphi \rightarrow \theta}$ where $n \geq 0$ and $\overline{\langle a\|b\rangle}_n = \langle a_1\|b_1\rangle \ldots \langle a_n\|b_n\rangle$; the new nominals a_{n+1}, b_{n+1} are distinct from c and other nominals in $\overline{\langle a\|b\rangle}_n$ and do not occur in φ or θ.

Table 3.1 The Hilbert-style proof system **HSML**

fourth capture the arbitrary deletion modality $\blacklozenge$. In particular, the finite prefixes of deletions in the rule schema (B-Mix) allow for reasoning about models arising from an initial one after finite histories of link cutting.①

For an illustration of how one can work with this calculus, we derive a few inference rules and theorems in the above proof system. Some of these principles will be useful in our proof for the strong completeness of **HSML** in Section 3.

Fact 7: Replacement of Equivalents: $\dfrac{\varphi \leftrightarrow \psi}{\langle a|b\rangle\varphi \leftrightarrow \langle a|b\rangle\psi}$ can be derived in HSML.

Proof First note that the monotonicity rule for $\blacklozenge$: $\dfrac{\varphi \to \psi}{\blacklozenge\varphi \to \blacklozenge\psi}$ is a derivable rule since $\blacksquare$ is a K operator. Next, we derive $\langle a|b\rangle\varphi \to \langle a|b\rangle\psi$ from $\vdash \varphi \to \psi$.

1. $\vdash \varphi \to \psi$ (assumption)
2. $\vdash (@_a\neg \Diamond b \wedge \varphi) \to (@_a\neg \Diamond b \wedge \psi)$ (from 1 by the propositional logic CPL)
3. $\vdash \blacklozenge(@_a\neg \Diamond b \wedge \varphi) \to \blacklozenge(@_a\neg \Diamond b \wedge \psi)$ (from 2 and the distribution rule for $\blacklozenge$)
4. $\vdash (@_a \Diamond b \wedge \blacklozenge(@_a\neg \Diamond b \wedge \varphi)) \to (@_a \Diamond b \wedge \blacklozenge(@_a\neg \Diamond b \wedge \psi))$ (from 3 and CPL)
5. $\vdash (@_a \Diamond b \wedge \blacklozenge(@_a\neg \Diamond b \wedge \varphi)) \vee (@_a\neg \Diamond b \wedge \varphi) \to (@_a \Diamond b \wedge \blacklozenge(@_a\neg \Diamond b \wedge \varphi)) \vee (@_a\neg \Diamond b \wedge \psi)$ (from 4 and CPL)
6. $\vdash \langle a|b\rangle\varphi \to \langle a|b\rangle\psi$ (from 5 and the definitions of $\langle a|b\rangle\varphi$ and $\langle a|b\rangle\psi$)

The derivation of the other direction of the equivalence, namely $\vdash \langle a|b\rangle\psi \to \langle a|b\rangle\varphi$ from $\vdash \psi \to \varphi$, proceeds analogously. Putting all this together, it follows that $\dfrac{\varphi \leftrightarrow \psi}{\langle a|b\rangle\varphi \leftrightarrow \langle a|b\rangle\psi}$ is an admissible inference rule in HSML. ∎

Fact 8: The formula: $\overline{\langle a|b\rangle}_n(\varphi \wedge \psi) \leftrightarrow (\overline{\langle a|b\rangle}_n\varphi \wedge \overline{\langle a|b\rangle}_n\psi)$ is provable in **HSML**.

Proof This formula generalizes the distribution of $\langle a|b\rangle$ over conjunction, which is a recursion axiom for $\langle a|b\rangle$ reflecting the fact that link cutting between named points is an operation that is a partial function on models. The proof of the Fact involves an iterated appeal to the recursion axiom for the conjunction, with successive substitutions licensed by Replacement of Equivalents. ∎

Another simple useful fact is this.

Fact 9: The formula: $@_a \Diamond b \wedge \langle a|b\rangle\psi \to \blacklozenge\psi$ is provable in **HSML**.

Proof This formula specifies the effect of cutting the link between a and b in terms of $\blacklozenge$. It follows easily from the above definition of the link-cutting modality $\langle a|b\rangle\varphi$ plus an appeal to CPL and the minimal modal logic for $\blacklozenge$. ∎

Next come two facts whose proofs are more complex than the preceding ones.

① In the logic PAL, finite sequences of announcements can be compressed to one by the Composition Axiom. However, it is easy to show that no such compression is possible in HSML, unless we define complex modalities for simultaneous link cuts.

Fact 10: $\overline{\langle a|b\rangle}_n \diamond \psi \leftrightarrow \bigvee_{S\subseteq[n]} (\bigwedge_{m\in S} a_m \wedge \bigwedge_{m\in[n]-S} \neg a_m \wedge \diamond(\bigwedge_{m\in S} \neg b_m \wedge \overline{\langle a|b\rangle}_n \psi))$ is provable in **HSML** for any natural nunber $n \in \mathbb{N}$, where $[n]$ with $n \geq 1$ denotes the set $\{1, \ldots, n\}$ while $[0]$ denotes the empty set $\varnothing$.

Proof For the case that $n = 0$, the formula reduces to $\diamond\psi \leftrightarrow \diamond\psi$, which is a tautology. For the case that $n = 1$, the formula reduces to $\langle a_1|b_1\rangle \diamond \varphi \leftrightarrow ((a_1 \wedge \diamond(\neg b_1 \wedge \langle a_1|b_1\rangle\varphi)) \vee (\neg a_1 \wedge \diamond\langle a_1|b_1\rangle\varphi))$, which is a recursion axiom for $\langle a_1|b_1\rangle$.

Next, we prove the general case, where each subset S of $[n]$ specifies a possible case. In each possible case, the left side specifies what happens to those worlds to which the current world has access to after the sequence of link cuttings.

Suppose that for all $0 \leq n \leq k$ and for all formulas ψ, we have already shown:

$$\vdash \overline{\langle a|b\rangle}_n \diamond \psi \leftrightarrow \bigvee_{S\subseteq [n]} (\bigwedge_{m\in S} a_m \wedge \bigwedge_{m\in[n]-S} \neg a_m \wedge \diamond(\bigwedge_{m\in S} \neg b_m \wedge \overline{\langle a|b\rangle}_n \psi)) \quad (I.H.)$$

We are going to to prove the assertion for $n = k + 1$.

For the sake of simplifying notation, let $\langle c_k\rangle$ denote $\langle a_k|b_k\rangle$, $\overline{\langle c\rangle}_k$ denote $\overline{\langle a|b\rangle}_k$ for $k \in \mathbb{N}$ and $\Theta^S_n \psi$ denote $\bigvee_{S\subseteq [n]} (\bigwedge_{m\in S} a_m \wedge \bigwedge_{m\in[n]-S} \neg a_m \wedge \diamond(\bigwedge_{m\in S} \neg b_m \wedge \overline{\langle a|b\rangle}_n \psi))$.

By the definition of $\overline{\langle c\rangle}_{k+1}$, we have

$$\vdash \overline{\langle c\rangle}_{k+1} \diamond \psi \leftrightarrow \overline{\langle c\rangle}_k \langle c_{k+1}\rangle \diamond \psi$$

Applying the Replacement of Equivalents rule k times to the recursion axiom $\vdash \langle c_{k+1}\rangle \diamond \psi \leftrightarrow (a_{k+1} \wedge \diamond(\neg b_{k+1} \wedge \langle c_{k+1}\rangle\psi)) \vee (\neg a_{k+1} \wedge \diamond\langle c_{k+1}\rangle\psi)$, we obtain

$$\vdash \overline{\langle c\rangle}_k \langle c_{k+1}\rangle \diamond \psi \leftrightarrow \overline{\langle c\rangle}_k ((a_{k+1} \wedge \diamond(\neg b_{k+1} \wedge \langle c_{k+1}\rangle\psi)) \vee (\neg a_{k+1} \wedge \diamond\langle c_{k+1}\rangle\psi))$$

It follows that

$$\vdash \overline{\langle c\rangle}_{k+1} \diamond \psi \leftrightarrow \overline{\langle c\rangle}_k ((a_{k+1} \wedge \diamond(\neg b_{k+1} \wedge \langle c_{k+1}\rangle\psi)) \vee (\neg a_{k+1} \wedge \diamond\langle c_{k+1}\rangle\psi))$$

Next, after applying the recursion axioms several times to the latter part of the above formula, it follows that

$$\vdash \overline{\langle c\rangle}_{k+1} \diamond \psi \leftrightarrow ((a_{k+1} \wedge \overline{\langle c\rangle}_k \diamond (\neg b_{k+1} \wedge \langle c_{k+1}\rangle\psi)) \vee (\neg a_{k+1} \wedge \overline{\langle c\rangle}_k \diamond \langle c_{k+1}\rangle\psi)) \quad (*)$$

Let α and β denote $\overline{\langle c\rangle}_k \diamond (\neg b_{k+1} \wedge \langle c_{k+1}\rangle\psi)$ and $\overline{\langle c\rangle}_k \diamond \langle c_{k+1}\rangle\psi$ respectively. Then, by applying the inductive hypothesis to α, β, we obtain the two facts

$$\vdash \alpha \leftrightarrow \Theta^S_k(\neg b_{k+1} \wedge \langle c_{k+1}\rangle\psi)$$

$$\vdash \beta \leftrightarrow \Theta^S_k(\langle c_{k+1}\rangle\psi)$$

Now replacing α, β by equivalent formulas in the formula $(*)$, we get

$$\vdash \overline{\langle c\rangle}_{k+1} \diamond \psi \leftrightarrow ((a_{k+1} \wedge \Theta^S_k(\neg b_{k+1} \wedge \langle c_{k+1}\rangle\psi)) \vee (\neg a_{k+1} \wedge \Theta^S_k(\langle c_{k+1}\rangle\psi)))$$

Focusing on the right part of this formula, we get the following equivalence:

$\vdash a_{k+1} \wedge \Theta_k^S(\neg b_{k+1} \wedge \langle c_{k+1}\rangle\psi) \leftrightarrow$

$\bigvee_{S\subseteq [k]} (\bigwedge_{m\in S\cup\{k+1\}} a_m \wedge \bigwedge_{m\in[k]-S} \neg a_m \wedge \Diamond(\bigwedge_{m\in S\cup\{k+1\}} \neg b_m \wedge \overline{\langle c\rangle}_{k+1}\psi))$

$\vdash \neg a_{k+1} \wedge \Theta_k^S \Diamond \langle c_{k+1}\rangle\psi \leftrightarrow \bigvee_{S\subseteq [k]} (\bigwedge_{m\in S} a_m \wedge \bigwedge_{m\in[k+1]-S} \neg a_m \wedge \Diamond(\bigwedge_{m\in S} \neg b_m \wedge \overline{\langle c\rangle}_{k+1}\psi))$

Notice how a_{k+1} and $\neg a_{k+1}$ distribute over the big disjunctions and how the $\neg b_{k+1}$ gets out of $\overline{\langle c\rangle}_k$ by the recursion axiom for nominals and merged into the big conjunction. Furthermore, by some combinatoric inference, we have $2^{[k+1]} = 2^{[k]} \cup \{S \cup \{k+1\} \mid S \in 2^{[k]}\}$. It thus follows that

$\vdash (a_{k+1} \wedge \Theta_k^S(\neg b_{k+1} \wedge \langle c_{k+1}\rangle\psi)) \wedge (\neg a_{k+1} \wedge \Theta_k^S(\langle c_{k+1}\rangle\psi)) \leftrightarrow \Theta_{k+1}^S\psi$

That is,

$\vdash \overline{\langle c\rangle}_{k+1} \Diamond \psi \leftrightarrow \bigvee_{S\subseteq[k+1]} (\bigwedge_{m\in S} a_m \wedge \bigwedge_{m\in[k+1]-S} \neg a_m \wedge \Diamond(\bigwedge_{m\in S} \neg b_m \wedge \overline{\langle c\rangle}_{k+1}\psi))$

which is what we needed to prove. ∎

Finally, we show how the B-Mix rule can be used to prove a basic principle about the interaction between $\Diamond$ and $\blacklozenge$.

Fact 11: $\vdash_{\mathrm{HSML}} \blacklozenge \Diamond \varphi \to \Diamond\blacklozenge\varphi$.

Proof
1. $\vdash \langle a|b\rangle\varphi \leftrightarrow (@_a \Diamond b \wedge \blacklozenge(@_a\neg \Diamond b \wedge \varphi)) \vee (@_a\neg \Diamond b \wedge \varphi)$ (by definition)
2. $\vdash @_a \Diamond b \wedge \langle a|b\rangle\varphi \to \blacklozenge\varphi$ (from 1)
3. $\vdash \Diamond(@_a \Diamond b \wedge \langle a|b\rangle\varphi) \to \Diamond\blacklozenge\varphi$ (from 2 in the minimal modal logic K)
4. $\vdash \Box@_a \Diamond b \wedge \Diamond\langle a|b\rangle\varphi \to \Diamond(@_a \Diamond b \wedge \langle a|b\rangle\varphi)$ (theorem of the logic K)
5. $\vdash \Box@_a \Diamond b \wedge \Diamond\langle a|b\rangle\varphi \to \Diamond\blacklozenge\varphi$ (from 3 and 4)
6. $\vdash \langle a|b\rangle \Diamond \varphi \leftrightarrow ((a \wedge \Diamond(\neg b \wedge \langle a|b\rangle\varphi)) \vee (\neg a \wedge \Diamond\langle a|b\rangle\varphi))$ (axiom for $\langle a|b\rangle$)
7. $\vdash \langle a|b\rangle \Diamond \varphi \to \Diamond\langle a|b\rangle\varphi$ (from 6)
8. $\vdash \Box@_a \Diamond b \wedge \langle a|b\rangle \Diamond \varphi \to \Diamond\blacklozenge\varphi$ (from 5 and 7)
9. $\vdash @_a \Diamond b \to \Box@_a \Diamond b$ (theorem of hybrid logic)
10. $\vdash @_a \Diamond b \wedge \langle a|b\rangle \Diamond \varphi \to \Diamond\blacklozenge\varphi$ (from 8 and 9)
11. $\vdash @_c(@_a \Diamond b \wedge \langle a|b\rangle \Diamond \varphi \to \Diamond\blacklozenge\varphi)$, for c not occurring in φ (**Nec** rule for $@_c$)
12. $\vdash @_c(@_a \Diamond b \wedge \langle a|b\rangle \Diamond \varphi) \to @_c \Diamond \blacklozenge\varphi$ (from 11)
13. $\vdash @_c\blacklozenge \Diamond \varphi \to @_c \Diamond \blacklozenge\varphi$ (from 12 using the B-Mix rule)
14. $\vdash @_c(\blacklozenge \Diamond \varphi \to \Diamond\blacklozenge\varphi)$ (from 13 in hybrid logic)
15. $\vdash c \to (\blacklozenge \Diamond \varphi \to \Diamond\blacklozenge\varphi)$ (from 14 in hybrid logic)

16. $\vdash \blacklozenge \Diamond \varphi \to \Diamond \blacklozenge \varphi$ (from 15 by the Name rule)

∎

It may be of interest to note that the converse implication $\Diamond\blacklozenge\varphi \to \blacklozenge \Diamond \varphi$ is not valid in HSML, as can be seen by giving a simple countermodel.

Readers who want to get still more familiar with the proof system **HSML** may find the implication $\langle a|b\rangle\langle c|d\rangle\varphi \to \langle c|d\rangle\langle a|b\rangle\varphi$ a useful further exercise.

3.2 Soundness and strong completeness for HSML

We now turn to the meta-properties of the proof system **HSML**.

Theorem 3.1 (Soundness): All provable formulas **HSML** are valid.

The soundness of most principles in the above proof system is immediate, we only concentrate on those that deserve special attention.

Fact 12: The axiom $\langle a|b\rangle \Diamond \varphi \leftrightarrow ((a \wedge \Diamond(\neg b \wedge \langle a|b\rangle\varphi)) \vee (\neg a \wedge \Diamond\langle a|b\rangle\varphi))$ is valid.

Proof Let $\mathfrak{M} = (W, R, V)$ and $\mathfrak{M}' = (W, R', V)$, where $R' = R \setminus \{(u, v) \,|\, \mathfrak{M}, u \vDash a$ and $\mathfrak{M}, v \vDash b\}$, i.e., the pair named by (a, b) has been deleted.

From left to right, if $\mathfrak{M}, w \vDash \langle a|b\rangle\Diamond\varphi$, then $\mathfrak{M}', w \vDash \Diamond\varphi$, so $\mathfrak{M}', v \vDash \varphi$ for some v with $R'wv$, and $\mathfrak{M}, v \vDash \langle a|b\rangle\varphi$. Case 1: $\mathfrak{M}, w \vDash a$. Then $\mathfrak{M}', w \vDash a$, and so $\mathfrak{M}', v \vDash \neg b$, whence $\mathfrak{M}, v \vDash \neg b$, and taking together, $\mathfrak{M}, v \vDash \neg b \wedge \langle a|b\rangle\varphi$ and $\mathfrak{M}, w \vDash (a \wedge \Diamond(\neg b \wedge \langle a|b\rangle\varphi)$: the first disjunct on the right. Case 2: $\mathfrak{M}, w \vDash \neg a$. Then, since $\mathfrak{M}, v \vDash \langle a|b\rangle\varphi$, we get the second disjunct: $\mathfrak{M}, w \vDash \neg a \wedge \Diamond\langle a|b\rangle\varphi$.

From right to left, a similar semantic argument will work, essentially reversing the preceding steps, including the case distinction. ∎

Fact 13: The B-Mix rule is sound.

Proof Assume that $@_c\overline{\langle a|b\rangle}_n(@_{a_{n+1}}\Diamond b_n \wedge \langle a_{n+1}|b_{n+1}\rangle\varphi) \to \theta$ is valid, where the nominals a_{n+1} and b_{n+1} are different from c and any nominals in the sequence $(a|b)_n$ and do not occur in φ and θ. Consider any HSML model $\mathcal{M}$ and point w such that $\mathcal{M}, w \vDash @_c\overline{\langle a|b\rangle}_n\blacklozenge\varphi$. According to the truth conditions for the link deletion modalities, there must be a still available link deletion $(d|d')$ after the links defined in the sequence $\overline{\langle a|b\rangle}_n$ have been cut such that φ is true after the deletion. Now take two fresh nominals a_{n+1} and b_{n+1} not occurring in the formulas so far, such that $V(a_{n+1}) = V(d)$ and $V(b_{n+1}) = V(d')$. Then the antecedent of the assumed validity is satisfied, and we get $\mathcal{M}, w \vDash \theta$. ∎

We have seen how the B-Mix rule is used in the proof system to prove significant theorems. In the following completeness proof, we will see it is also essential for constructing a special type of maximally consistent sets.

Theorem 3.2: The proof system **HSML** is strongly complete.

The proof to follow uses the technique introduced for the modal logic MLSR in (van Benthem et al., 2020). In the present setting, this involves combining basic modal logic, hybrid logic, the key HSML modality $\blacklozenge$ for arbitrary link deletion in a graph, and its interaction with the above defined modality for deletion of named links. A noteworthy difference with the cited reference is our simplification in defining the latter modality, cf. Fact 6 in Section 2.1, so we can do without DEL-style link cutting modalities as additional primitives.

As is standard in completeness proofs, it suffices to show that any **HSML**-consistent set of formulas is satisfiable in a HSML model.

The first step is to prove that any **HSML**-consistent set can be extended to a maximally consistent set ('**HSML**-MCS') satisfying the following properties.

Definition 3.4 (Named, pasted, mixed, B-mixed): A set of formulas Γ is (a) *named* if it contains a nominal, (b) *pasted* if $@_a\Diamond\varphi \in \Gamma$ implies that there is some nominal b such that the formula $@_a\Diamond b \wedge @_b\varphi \in \Gamma$, (c) *mixed* if $\overline{\langle a|b\rangle}_n\blacklozenge\varphi \in \Gamma$ implies that $\overline{\langle a|b\rangle}_n(@_{a_{n+1}}\Diamond b_{n+1} \wedge \langle a_{n+1}|b_{n+1}\rangle\varphi) \in \Gamma$ for some nominals a_{n+1}, b_{n+1}, and finally, (d) Γ is *B-mixed* if $@_c\overline{\langle a|b\rangle}_n\blacklozenge\varphi \in \Gamma$ implies that $@_c\overline{\langle a|b\rangle}_n(@_{a_{n+1}}\Diamond b_{n+1} \wedge \langle a_{n+1}|b_{n+1}\rangle\varphi) \in \Gamma$ for some nominals a_{n+1}, b_{n+1}.

The properties *named* and *pasted* are needed to deal with the hybrid component of the logic while *mixed* and *B-mixed* are for the link-cutting part. The property *mixed* will become relevant later in Lemma 3.2.

Lemma 3.1 (Lindenbaum Lemma): Let Nom' be a countably infinite set of nominals disjoint from Nom, and let $\mathcal{L}'$ be the language obtained by adding these new nominals to $\mathcal{L}$. Every **HSML**-consistent set of formulas in language $\mathcal{L}$ can be extended to a named, pasted and B-mixed **HSML**-MCS in the language $\mathcal{L}'$.

Proof Given a consistent set of $\mathcal{L}$-formulas Σ, let Σ_d to be $\Sigma \cup \{d\}$, where d is an arbitrary new nominal in Nom'. Σ_d is consistent. For suppose not. Then for some conjunction of formulas θ from Σ, $\vdash d \to \neg\theta$. But the new nominal d does not occur in θ, and so, by the Name rule, $\vdash \neg\theta$. This contradicts the consistency of Σ: so Σ_d must be consistent.

Next, enumerate all the formulas of $\mathcal{L}'$ (this includes the nominals in Nom'). We define a sequence of consistent sets as follows. Let Σ^0 be the set Σ_d just constructed. Now, working inductively, suppose we have defined Σ^m, where $m \geq 0$. Let φ_{m+1} be the $m+1$-th formula in our enumeration of $\mathcal{L}'$. We define Σ^{m+1} as follows. If $\Sigma^{m+1} \cup \{\varphi_{m+1}\}$ is inconsistent, then $\Sigma^{m+1} = \Sigma^m$. Otherwise:

- $\Sigma^{m+1} = \Sigma^m \cup \{\varphi_{m+1}\} \cup \{@_a\Diamond b \wedge @_b\varphi\}$, if φ_{m+1} is of the form $@_a\Diamond\varphi$.
  Here b is the first nominal in the enumeration not occurring in Σ^m or $@_a\Diamond\varphi$.
- $\Sigma^{m+1} = \Sigma^m \cup \{\varphi_{m+1}\} \cup \{@_c\overline{\langle a|b\rangle}_n(@_{a_{n+1}} \Diamond b_{n+1} \wedge \langle a_{n+1}|b_{n+1}\rangle\varphi)\}$, if φ_{m+1} is of the form $@_c\overline{\langle a|b\rangle}_n\blacklozenge\varphi \in \Gamma$. Here a_{n+1}, b_{n+1} are the first two nominals in the enumeration that do not occur in Σ^m or $@_c\overline{\langle a|b\rangle}_n\blacklozenge\varphi$.
- $\Sigma^{m+1} = \Sigma^m \cup \{\varphi_{m+1}\}$ if φ_{m+1} is not of the form $@_a\Diamond\varphi$ or $@_c\overline{\langle a|b\rangle}_n\blacklozenge\varphi$.

Let $\Sigma^+ = \bigcup_{n\geq 0}\Sigma^n$. Clearly this set is named, maximal, pasted and B-mixed. It is also consistent. For expansions of the first kind, consistency preservation is guaranteed by the Paste rule. For expansions of the second kind, if the set obtained is not consistent, then for some conjunction of formulas θ from the set $\Sigma^m \cup \{\varphi_{m+1}\}$,

$$\vdash @_c\overline{\langle a|b\rangle}_n(@_{a_{n+1}} \Diamond b_{n+1} \wedge \langle a_{n+1}|b_{n+1}\rangle\varphi \to \neg\theta.$$

By the B-mix rule, $\vdash \varphi_{m+1} \to \neg\theta$, contradicting the consistency of $\Sigma^m \cup \{\varphi_{m+1}\}$. ∎

Next, each **HSML**-MCS Γ induces a family of maximally consistent sets.

Definition 3.5: The *named set* Δ_a *yielded by* Γ is $\{\varphi \mid @_a\varphi \in \Gamma\}$.

Now we can define the modal model that will satisfy our given consistent set.

Definition 3.6 (Named model): The *named model generated by* Γ is the tuple $\mathfrak{M}^\Gamma = (W^\Gamma, R^\Gamma, V^\Gamma)$ where (a) W^Γ consists of all named sets yielded by Γ, (b) $R^\Gamma uv$ iff for all formulas φ with $\varphi \in v$, we have $\Diamond\varphi \in u$, and finally (c) $V^\Gamma(o) = \{w \in W^\Gamma \mid o \in w, o \in \mathsf{Prop} \cup \mathsf{Nom}\}$.

This model has the following basic properties that can be shown just as in standard completeness proofs for hybrid logic, (Areces and ten Cate, 2007).

Lemma 3.2 (Existence Lemma): Let Γ be a named, pasted and B-mixed **HSML**-MCS and let $\mathfrak{M}^\Gamma = (W^\Gamma, R^\Gamma, V^\Gamma)$ be the named model yielded by Γ.

(a) All named sets Δ_a yielded by Γ are **HSML**-MCSs.

(b) If $u \in W^\Gamma$ and $\Diamond\varphi \in u$, then there is some $v \in W^\Gamma$ with $R^\Gamma uv$ and $\varphi \in v$.

(c) All named sets Δ_a yielded by Γ are mixed.

Proof We only prove the least standard third item. Assume that $\overline{\langle a|b\rangle}_n\blacklozenge\varphi \in \Delta_c$, i.e., $@_c\overline{\langle a|b\rangle}_n\blacklozenge\varphi \in \Gamma$. Since the set Γ is B-mixed, we have $@_c\overline{\langle a|b\rangle}_n(@_{a_{n+1}} \Diamond b_{n+1} \wedge \langle a_{n+1}|b_{n+1}\rangle\varphi) \in \Gamma$ for some nominals a_{n+1}, b_{n+1}, and so we have immediately that $\overline{\langle a|b\rangle}_n(@_{a_{n+1}} \Diamond b_{n+1} \wedge \langle a_{n+1}|b_{n+1}\rangle\varphi) \in \Delta_c$. This means that Δ_c is mixed. ∎

Now comes the part of the proof where we need to consider models arising after link deletions, in order to deal with the modality $\blacklozenge$. In addition to the preceding named model, we introduce the following new models.

Definition 3.7 (Derived Henkin model): Let $\overline{\langle a|b\rangle}_n = \langle a_1|b_1\rangle \dots \langle a_n|b_n\rangle$. The *derived Henkin model* from a named model $\mathfrak{M}^\Gamma$ generated by Γ is the tuple

$$\mathfrak{M}^\Gamma : \overline{\langle a|b\rangle}_n = (W^{\overline{\langle a|b\rangle}_n}, R^{\overline{\langle a|b\rangle}_n}, V^{\overline{\langle a|b\rangle}_n})$$

with worlds, accessibilty and valuations defined as follows:

- $W^{\overline{\langle a|b\rangle}_n} = \{(w, \overline{\langle a|b\rangle}_n) \mid w \in W^\Gamma\}$
- $R^{\overline{\langle a|b\rangle}_n}((u, \overline{\langle a|b\rangle}_n), (v, \overline{\langle a|b\rangle}_n))$ if

  (a) $R^\Gamma uv$ and (b) $a_i \notin u$ or $b_i \notin v$ for all $i \leq n$
- $V^{\overline{\langle a|b\rangle}_n}(o) = \{(w, \overline{\langle a|b\rangle}_n) \mid w \in V^\Gamma(o), o \in \mathsf{Prop} \cup \mathsf{Nom}\}$.

We stipulate that $\mathfrak{M}^\Gamma : \overline{\langle a|b\rangle}_0 = \mathfrak{M}^\Gamma$.

Each point in the derived Henkin Model induces the following set of formulas:

$$\Phi(\mathfrak{M}, \overline{\langle a|b\rangle}_n, w) = \{\varphi \mid \overline{\langle a|b\rangle}_n\varphi \in w\}$$

We now prove the crucial Truth Lemma: for derived Henkin models, membership in these induced sets and truth in the corresponding worlds coincide.

Lemma 3.3 (Truth Lemma): For all formulas φ, finite sequences $\overline{\langle a|b\rangle}_n$ and points w in a named model $\mathfrak{M}$ yielded by Γ, we have that, for any $n \geq 0$:

$$\mathfrak{M} : \overline{\langle a|b\rangle}_n, (w, \overline{\langle a|b\rangle}_n) \vDash \varphi \text{ iff } \varphi \in \Phi(\mathfrak{M}, \overline{\langle a|b\rangle}_n, w)$$

Proof The proof is by induction on the formulas φ. For brevity, we will write $\mathfrak{M} : \overline{\langle a|b\rangle}_n, w \vDash \varphi$, leaving out the sequence notation $\overline{\langle a|b\rangle}_n$.

(a) *Atomic formulas*. We only prove the case for p, the one for nominals is similar. $\mathfrak{M} : \overline{\langle a|b\rangle}_n, w \vDash p$ iff $w \in V(p)$ iff $p \in w$ (by the definition of V in derived Henkin models) iff $\overline{\langle a|b\rangle}_n p \in w$ (by the recursion axiom for p) iff $p \in \Phi(\mathfrak{M}, \overline{\langle a|b\rangle}_n, w)$.

(b) *Negations.* $\mathfrak{M} : \overline{\langle a|b\rangle}_n, w \vDash \neg\psi$ if $\mathfrak{M} : \overline{\langle a|b\rangle}_n, w \nvDash \psi$ iff (by the inductive hypothesis) $\psi \notin \Phi(\mathfrak{M}, \overline{\langle a|b\rangle}_n, w)$ iff $\overline{\langle a|b\rangle}_n\psi \notin w$ iff $\neg\overline{\langle a|b\rangle}_n\psi \in w$ iff (by the recursion axiom for $\neg\psi$) $\overline{\langle a|b\rangle}_n\neg\psi \in w$ iff $\neg\psi \in \Phi(\mathfrak{M}, \overline{\langle a|b\rangle}_n, w)$.

(c) *Conjunction.* The proof is like the preceding one, using the inductive hypothesis and the recursion axiom for conjunctions under link cutting modalities.

(d) @ *Operators.* $\mathfrak{M} : \overline{\langle a|b\rangle}_n, w \vDash @_c\psi$ iff $\mathfrak{M} : \overline{\langle a|b\rangle}_n, \Delta_c \vDash \psi$ iff (by the inductive hypothesis) $\psi \in \Phi(\mathfrak{M}, \overline{\langle a|b\rangle}_n, \Delta_c)$ iff (by definition) $\overline{\langle a|b\rangle}_n\psi \in \Delta_c$ iff (noting that $\alpha \in \Delta_c$ iff $@_c\alpha \in \Delta_a$ for any nominal a) $@_c\overline{\langle a|b\rangle}_n\psi \in w$ iff (by the recursion axiom for $@_c$) $\overline{\langle a|b\rangle}_n @_c\psi \in w$ iff $@_c\psi \in \Phi(\mathfrak{M}, \overline{\langle a|b\rangle}_n, w)$.

(e) $\Diamond$ *modality.* In the case of $n = 0$, the assertion reduces to the standard modal case, whose proof is well-known, (Blackburn et al., 2001). So let us focus on the case $n \neq 0$. From left to right, let $\mathfrak{M} : \overline{\langle a|b\rangle}_n, w \vDash \Diamond\psi$. Then there is a v with $R^{\overline{\langle a|b\rangle}_n}((w, \overline{\langle a|b\rangle}_n), (v, \overline{\langle a|b\rangle}_n))$ and $\mathfrak{M} : \overline{\langle a|b\rangle}_n, v \vDash \psi$. By the inductive hypothesis, $\psi \in \Phi(\mathfrak{M}, \overline{\langle a|b\rangle}_n, v)$, i.e., $\overline{\langle a|b\rangle}_n\psi \in v$. Since $R^{\overline{\langle a|b\rangle}_n}(w, \overline{\langle a|b\rangle}_n)(v, \overline{\langle a|b\rangle}_n)$, it follows that Rwv. Thus by the definition of R in a named model, $\Diamond\overline{\langle a|b\rangle}_n\psi \in w$. Now, by the definition of the relations $R^{\overline{\langle a|b\rangle}_n}$, $a_x \notin w$ or $b_x \notin v$ for any $x \in [1, ..., n]$. In particular, for any $x \in [1, ..., n]$, if $a_x \in w$, $b_x \notin v$. Starting from a_1, either $a_1 \wedge \Diamond(\neg b_1 \wedge \overline{\langle a|b\rangle}_n\psi) \in w$ or $\neg a_1 \wedge \Diamond\overline{\langle a|b\rangle}_n\psi \in w$. By the recursion axiom for $\Diamond$, we then get that $\langle a_1|b_1\rangle \Diamond \langle a_2|b_2\rangle \dots \langle a_n|b_n\rangle\psi \in w$. Repeating this argument, we can push $\Diamond$ to the innermost position, which gives us the desired result $\overline{\langle a|b\rangle}_n \Diamond \psi \in w$. That is, $\Diamond\psi \in \varphi(\mathfrak{M}, \overline{\langle a|b\rangle}_n, w)$.

From right to left: let $\Diamond\psi \in \Phi(\mathfrak{M}, \overline{\langle a|b\rangle}_n, w)$, i.e., $\overline{\langle a|b\rangle}_n \Diamond \psi \in w$. By Fact 10, we have the set $S = \{x \in [1, ..., n] \mid a_x \in w\}$ such that $\Diamond(\bigwedge_{x\in S} \neg b_x \wedge \overline{\langle a|b\rangle}_n\psi) \in w$. By the Existence Lemma for $\Diamond$, there is a v with $\bigwedge_{x\in S} \neg b_x \wedge \overline{\langle a|b\rangle}_n\psi \in v$, which implies that $\psi \in \Phi(\mathfrak{M}, \overline{\langle a|b\rangle}_n, v)$. By the inductive hypothesis, $\mathfrak{M} : \overline{\langle a|b\rangle}_n, v \vDash \psi$. Also, by the definition of S and $\bigwedge_{x\in S} \neg b_x \in v$, we have for any $x \in [1, ..., n]$, $a_x \notin w$ or $b_x \notin v$. By the definition of $R^{\overline{\langle a|b\rangle}_n}$ and Rwv, then $R^{\overline{\langle a|b\rangle}_n}((w, \overline{\langle a|b\rangle}_n), (v, \overline{\langle a|b\rangle}_n))$. Therefore, $\mathfrak{M} : \overline{\langle a|b\rangle}_n, w \vDash \Diamond\psi$.

(f) *The deletion modality* $\blacklozenge$. From left to right, let $\mathfrak{M} : \overline{\langle a|b\rangle}_n, w \vDash \blacklozenge\psi$. Then there is a link in $\mathfrak{M} : \overline{\langle a|b\rangle}_n$, say $((\Delta_{a_{n+1}}, \overline{\langle a|b\rangle}_n), (\Delta_{b_{n+1}}, \overline{\langle a|b\rangle}_n))$ (the naming of the link is guaranteed by our model construction) such that $\mathfrak{M} : \overline{\langle a|b\rangle}_{n+1}, w \vDash \psi$. Then by the inductive hypothesis $\psi \in \Phi(\mathfrak{M}, \overline{\langle a|b\rangle}_{n+1}, w)$, i.e., $\overline{\langle a|b\rangle}_{n+1}\psi \in w$. Moreover, our model construction even yields that $\mathfrak{M} : \overline{\langle a|b\rangle}_n, w \models @_{a_{n+1}} \diamond b_{n+1}$. But then, by cases already proved, it follows that $@_{a_{n+1}} \diamond b_{n+1} \in \Phi(\mathfrak{M}, \overline{\langle a|b\rangle}_n, w)$, i.e., $\overline{\langle a|b\rangle}_n @_{a_{n+1}} \diamond b_{n+1} \in w$. Now recall the definition of named link cutting $\langle a_{n+1}|b_{n+1}\rangle\psi$ in the language of HSML. We noted earlier that $@_{a_{n+1}} \diamond b_{n+1} \wedge \langle a_{n+1}|b_{n+1}\rangle\psi \to \blacklozenge\psi$ is a theorem of **HSML**, and using the principles of the minimal logic K for $\langle a|b\rangle$, we get $\overline{\langle a|b\rangle}_n \blacklozenge\psi \in w$. Thus $\blacklozenge\psi \in \Phi(\mathfrak{M}, \overline{\langle a|b\rangle}_n, w)$.

Finally, from right to left, let $\blacklozenge\psi \in \Phi(\mathfrak{M}, \overline{\langle a|b\rangle}_n, w)$. By the Existence Lemma, w is mixed, and so $\psi \in \Phi(\mathfrak{M}, \overline{\langle a|b\rangle}_{n+1}, w)$ and $@_{a_{n+1}} \diamond b_{n+1} \in \Phi(\mathfrak{M}, \overline{\langle a|b\rangle}_n, w)$ for new nominals a_{n+1} and b_{n+1} that do not occur in ψ. Thus, by the inductive hypothesis, $\mathfrak{M}$: $\overline{\langle a|b\rangle}_{n+1}, w \models \psi$. Also, by inductive cases already proved, $\mathfrak{M} : \overline{\langle a|b\rangle}_n, w \models @_{a_{n+1}} \diamond b_{n+1}$, and hence $((\Delta_{a_{n+1}}, \overline{\langle a|b\rangle}_n), (\Delta_{b_{n+1}}, \overline{\langle a|b\rangle}_n)) \in R^{\overline{\langle a|b\rangle}_n}$. Now $R^{\overline{\langle a|b\rangle}_{n+1}}$ equals the relation $R^{\overline{\langle a|b\rangle}_n} \setminus ((\Delta_{a_{n+1}}, \overline{\langle a|b\rangle}_n), (\Delta_{b_{n+1}}, \overline{\langle a|b\rangle}_n))$ while the valuation functions in all derived Henkin models are the same modulo the indexical sequences, we have $\mathfrak{M} : \overline{\langle a|b\rangle}_n, w \models \blacklozenge\psi$.

As usual, this finalizes the proof of the completeness theorem, since all formulas in the initially given set Γ will be true at the initial world of the named model induced by some arbitrary maximally consistent extension of Γ. ∎

3.3 Protocol HSML

Having analyzed HSML on standard models, we now consider a natural generalization, also known from dynamic-epistemic logic, (Hoshi, 2014). Suppose that not all link deletions are available, for instance, to Demon in a sabotage game. This gives more general 'protocol models' for scenarios where agents operate under various constraints. There are several types of protocols, less or more general, cf. (van Benthem et al., 2009), but we will only analyze one particular case here.

Definition 3.8 (Protocols): Let $\Sigma = \{(a|b) \mid a, b \in \mathsf{NOM}\}$. Members of the set Σ^* of all finite sequences of elements in Σ are called *histories*. A subset S of Σ^* is *closed under initial segments* if for any $h \in S$, its initial segments $h' \sqsubseteq h$ are also in S. A *protocol* is a set of histories closed under taking initial segments. Any HSML model $\mathfrak{M} = (W, R, V)$ has an associated set $\mathsf{Prtcl}(\mathfrak{M})$ of *feasible protocols* f satisfying the following condition: if $(a_1|b_1) \dots (a_i|b_i) \in f$, then (a) $(V(a_i), V(b_i)) \in R$, and (b) for each $j < i, V(a_i) \neq V(a_j)$ or $V(b_i) \neq V(b_j)$.

Here a history represents a sequence of successive link deletions in the given model, and a protocol defines which such sequences are allowed, for various reasons that may depend on the precise application. Condition (a) on protocols states that all links to be deleted actually exist, condition (b) states that no link is deleted twice, clearly minimal conditions on executable protocols.

Definition 3.9 (Protocol model): Given a HSML model $\mathfrak{M} = (W, R_0, V_0)$ and one of its feasible protocols f, the *protocol model* $\mathfrak{F} = \mathsf{Forest}(\mathfrak{M}, f) = (H, R, V)$ is defined from initial worlds and link cut histories as follows:

(a) $H = \{w\sigma \mid w \in W, \sigma \in f\}$.

(b) Rhh' iff $h = w\sigma$ and $h' = v\sigma$ for some $\sigma \in f$ and $w, v \in W$ satisfying R_0wv while $V_0(a) \neq w$ or $V_0(b) \neq v$ for any $(a|b) \in \sigma$.

(c) $V(o) = \{w\sigma \in H \mid V_0(a) = w\}$ where $o \in \mathsf{Prop} \cup \mathsf{Nom}$.

The semantics of HSML is easily lifted to protocol models:

Definition 3.10 (Truth conditions): Given a protocol model $\mathfrak{F} = \langle H, R, U\rangle$ and a world $h = w\sigma \in H$, truth is defined by the following conditions:

$\mathfrak{F}, w\sigma \vDash o$	iff	$w\sigma \in V(o)$, where $o \in \mathsf{Prop} \cup \mathsf{Nom}$
$\mathfrak{F}, h \vDash \neg\varphi$	iff	not $\mathfrak{F}, h \vDash \varphi$
$\mathfrak{F}, h \vDash \varphi_1 \wedge \varphi_2$	iff	$\mathfrak{F}, h \vDash \varphi_1$ and $\mathfrak{F}, h \vDash \varphi_2$
$\mathfrak{F}, w\sigma \vDash @_a\varphi$	iff	there is $v\sigma \in V(a)$ such that $\mathfrak{F}, v\sigma \vDash \varphi$
$\mathfrak{F}, h \vDash \Diamond\varphi$	iff	there is $h' \in H$ such that Rhh' and $\mathfrak{F}, h \vDash \varphi$
$\mathfrak{F}, w\sigma \vDash \blacklozenge\varphi$	iff	there is $\sigma' = \sigma(a|b) \in f$ s.t. $\mathfrak{F}, w\sigma' \vDash \varphi$.

The syntactic definition of $\langle a|b\rangle$ is the same as that in HSML:

$$\langle a|b\rangle\varphi := (@_a \Diamond b \wedge \blacklozenge(@_a\neg \Diamond b \wedge \varphi)) \vee (@_a\neg \Diamond b \wedge \varphi)$$

The following proposition describing its effect can easily be verified.

Fact 14: $\mathfrak{F}, w\sigma \vDash \langle a|b\rangle\varphi$ iff

(a) $\sigma(a|b) \in f$ and $\mathfrak{F}, w\sigma(a|b) \vDash \varphi$, or (b) $\mathfrak{F}, w\sigma \vDash @_a\neg \Diamond b \wedge \varphi$

A Hilbert-style proof system for Protocol HSML is presented in Table 2. The difference with the axiom system **HSML** is that deletions are no longer freely available, so we need to modify some of the recursion axioms for named link cuts. For instance, the earlier axiom $\langle a|b\rangle p \leftrightarrow p$ now becomes

$$\langle a|b\rangle p \leftrightarrow \langle a|b\rangle\top \wedge p$$

Axioms and rules for basic hybrid modal logic	See Table 1
K axiom for $\blacksquare$ Necessitation Rule	See Table 1
Invariance axiom for $\langle a\|b\rangle$	$@_c\langle a\|b\rangle\top \leftrightarrow \langle a\|b\rangle\top$
Recursion axioms for $\langle a\|b\rangle$	$\langle a\|b\rangle c \leftrightarrow \langle a\|b\rangle\top \wedge c$ $\langle a\|b\rangle p \leftrightarrow \langle a\|b\rangle\top \wedge p$ $\langle a\|b\rangle\neg\varphi \leftrightarrow \neg\langle a\|b\rangle\varphi$ $\langle a\|b\rangle\neg\phi \leftrightarrow \langle a\|b\rangle\top \wedge \neg\langle a\|b\rangle\phi$ $\langle a\|b\rangle @_c\varphi \leftrightarrow @_c\langle a\|b\rangle\varphi$ $\langle a\|b\rangle \Diamond \varphi \leftrightarrow ((a \wedge \Diamond(\neg b \wedge \langle a\|b\rangle\varphi)) \vee (\neg a \wedge \Diamond\langle a\|b\rangle\varphi))$
Inference rule for $\blacklozenge$	(B-Mix) : $\dfrac{@_c\overline{\langle a\|b\rangle}_n(@_{a_{n+1}} \Diamond b_{n+1} \wedge \langle a_{n+1}\|b_{n+1}\rangle\varphi) \to \theta}{@_c\overline{\langle a\|b\rangle}_n\blacklozenge\varphi \to \theta}$ where $n \geq 0$ and $\overline{\langle a\|b\rangle}_n = \langle a_1\|b_1\rangle \ldots \langle a_n\|b_n\rangle$; the new nominals a_{n+1}, b_{n+1} are distinct from c and other nominals in $\overline{\langle a\|b\rangle}_n$ and do not occur in φ or θ.

Table 3.2 The Hilbert-style proof system for protocol HSML

which also contains irreducible protocol information about available deletions.[1] In addition, the system contains a new principle expressing that the protocol is 'uniform': the available deletions are the same at each world:

$$@_c\langle a|b\rangle\top \leftrightarrow \langle a|b\rangle\top$$

Theorem 3.3: Protocol HSML is strongly complete.

Proof The completeness proof follows the same pattern as our earlier one for **HSML**. We merely sketch some salient steps that require attention.

For a start, the Lindenbaum Lemma can be proved just as before. With a little more care, we can also still have the earlier named models:

Definition 3.11: We say that $\Delta_a = \{\varphi \mid @_a\varphi \in \Gamma\}$ is the *protocol named set yielded by* Γ. A named model is a tuple $(\mathfrak{M}^\Gamma, f^\Gamma) = ((W^\Gamma, R^\Gamma, V^\Gamma), f^\Gamma)$ where

[1] The modified recursion axioms allow new situations. E.g., $\neg\langle a|b\rangle p \wedge \neg\langle a|b\rangle\neg p$ is not satisfiable in HSML, but in Protocol HSML it is true in a model where $(a, b) \notin f$.

(a) W^Γ is the set of all named set yielded by Γ

(b) $R^\Gamma uv$ iff for all formulas φ, $\varphi \in v$ implies $\Diamond\varphi \in u$

(c) $V^\Gamma(o) = \{w \in W^\Gamma \mid o \in w, o \in \mathsf{Prop} \cup \mathsf{Nom}\}$

(d) $f^\Gamma = \{\overline{(a|b)_n} : \overline{\langle a|b\rangle_n}\top \wedge \bigwedge_{i=0}^{n-1} \overline{\langle a|b\rangle_i}@_{a_{i+1}} \Diamond b_{i+1} \in \Gamma\}$.

Here the condition $\bigwedge_{i=0}^{n-1} \overline{\langle a|b\rangle_i}@_{a_{i+1}} \Diamond b_{i+1} \in \Delta_c$ picks out all those sequences of link cuts admissible according to Γ which do not include any vacuous cuts.

Lemma 3.4: If $\overline{\langle a|b\rangle_n}\top \in \Gamma$, then there is $\sigma = \overline{(c|d)_m} \in f^\Gamma$ s.t. for all φ, $m \leq n$:

(a) $\overline{\langle c|d\rangle_m}\varphi \in \Gamma$ iff $\overline{\langle a|b\rangle_n}\varphi \in \Gamma$, (b) $\mathfrak{F}, w \vDash \overline{\langle c|d\rangle_m}\varphi$ iff $\mathfrak{F}, w \vDash \overline{\langle a|b\rangle_n}\varphi$

Proof We can get the sequence $\overline{(c|d)_m}$ from $\overline{(a|b)_n}$ by deleting all those pairs $(a_i|b_i)$ for which $\overline{\langle a|b\rangle_i}@_{a_{i+1}} \Diamond b_{i+1} \notin \Gamma$. ∎

Next comes a slightly different route from the completeness proof for **HSML**.

Lemma 3.5: For all formulas φ, finite sequences $\sigma = \overline{(a|b)_n} \in f^\Gamma$ and points w in the generated protocol named model $\mathfrak{F} = \mathsf{Forest}(\mathfrak{M}, f)$ yielded by Γ,

$$\mathfrak{F}, w\sigma \vDash \varphi \text{ iff } \overline{\langle a|b\rangle_n}\varphi \in w$$

Proof The proof is by induction on the formulas φ.

(a) *Atomic propositions and nominals.* Given that $\sigma \in f$, which implies that $\overline{\langle a|b\rangle_n}\top \in w$, we have $\mathfrak{F}, w\sigma \vDash p$ iff $w\sigma \in V(p)$ iff $p \in w$ iff $\overline{\langle a|b\rangle_n}p \in w$.

The case of nominals is similar.

(b) *Negations.* Since $\overline{\langle a|b\rangle_n}\top \in w$, we can use the modified recursion axiom to get $\mathfrak{F}, w\sigma \vDash \neg\psi$ iff $\mathfrak{F}, w \nvDash \psi$ iff $\overline{\langle a|b\rangle_n}\psi \notin w$ iff $\neg\overline{\langle a|b\rangle_n}\psi \in w$ iff $\overline{\langle a|b\rangle_n}\neg\psi \in w$.

As for further inductive steps, the cases for the operators $\wedge$, $@_c$, $\Diamond$ and $\blacklozenge$ are similar to those in the proof of the Truth Lemma for HSML in Section 3, using the derived Henkin Model, since the recursion axioms for these operators have not changed. Here we treat @ and $\blacklozenge$ as examples.

(c) We have the following equivalences: $\mathfrak{F}, w\sigma \vDash @_c\psi$ iff $\mathfrak{F}, \Delta_c\sigma \vDash \psi$ iff $\overline{\langle a|b\rangle_n}\psi \in \Delta_c$ iff $@_c\overline{\langle a|b\rangle_n}\psi \in w$ iff $\overline{\langle a|b\rangle_n}@_c\psi \in w$.

(d) First, assume that $\mathfrak{F}, w\sigma \vDash \blacklozenge\psi$. Then there is $\sigma' = \sigma(a|b)_{n+1} \in f$ such that $\mathfrak{F}, w\sigma' \vDash \psi$. Thus by the inductive hypothesis $\overline{\langle a|b\rangle_{n+1}}\psi \in w$. Since $\mathfrak{F}, w\sigma \vDash @_{a_{n+1}} \Diamond b_{n+1}$, by the cases we have proved, it follows that $\overline{\langle a|b\rangle_n}@_{a_{n+1}} \Diamond b_{n+1} \in w$. Therefore, $\overline{\langle a|b\rangle_n}\blacklozenge\psi \in w$.

Next, assume that $\overline{\langle a|b\rangle_n}\blacklozenge\psi \in w$. By the Existence Lemma, w is mixed, and so we have $\overline{\langle a|b\rangle}_{n+1}\psi \in w$ for some a_{n+1} and b_{n+1}. By definition of f, $\sigma' = \sigma(a|b)_{n+1} \in f$. By the inductive hypothesis, $\mathfrak{F}, w\sigma' \vDash \psi$. Therefore $\mathfrak{F}, w\sigma \vDash \blacklozenge\psi$. ∎

The key Truth Lemma follows immediately from Lemma 3.4 and Lemma 3.5.

Lemma 3.6 (Truth Lemma): For all formulas φ, finite sequences $\sigma = \overline{(a|b)}_n$ and points w in the named protocol model $\mathfrak{F} = \mathsf{Forest}(\mathfrak{M}, f)$ yielded by Γ,

$$\mathfrak{F}, w \vDash \overline{\langle a|b\rangle}_n\varphi \text{ iff } \overline{\langle a|b\rangle}_n\varphi \in w$$

This finalizes the proof of the completeness theorem for Protocol HSML. ∎

Remark: Comparing the two completeness proofs. The 'full protocol' $\mathit{full}(\mathfrak{M})$ for a model $\mathfrak{M}$ consists of all possible histories of link cuts. The derived Henkin model of Definition 3.7 in the completeness proof for HSML is in essence the full protocol model of the named model of Definition 3.6. The difference is only notational: a history $w\sigma$ in the full protocol model of $\mathcal{M}$ is attached to the model, becoming one of its pointed derived Henkin models $\mathcal{M} : \sigma, w$. Also, the truth conditions of $\Diamond$, $\blacklozenge$ in the full protocol model $\mathsf{Forest}(\mathfrak{M}, \mathit{full}(\mathfrak{M})), w\sigma$ are as in the derived Henkin model $\mathcal{M} : \sigma, w$. Thus one could also start with a completeness proof in the format that we have given here for protocol models, and then derive one for standard models as a special case.

Discussion: Reducing HSML and Protocol HSML. Given the analogy in completeness proofs, it is a natural question how HSML and Protocol HSML are related. For instance, can one find a formula φ' for every formula φ such that φ is satisfiable in HSML iff φ' is satisfiable in protocol HSML? One might think of such a formula φ' as a conjunction of the form

$$\bigwedge_{a,b\in Q} \overline{\langle a|b\rangle}_n\top \wedge \varphi$$

The first conjunct says that all link cuts explicitly involved in φ are admissible, where Q is the set of all nominals that occur in φ, possibly plus some new ones. The problem, however, is that not all relevant link cuts need be explicitly stated in a given formula φ, as illustrated in the following example.

$$\varphi := \Diamond\top \wedge \neg\blacklozenge p \wedge \neg\blacklozenge\neg p$$

is not satisfiable in HSML. Since there are no nominals in the formula φ, φ' would equal φ by the above method. However, φ' is satisfiable in protocol HSML.

Are there better reductions? And what about the opposite direction, from Protocol

HSML to HSML? While we believe that mutual reductions indeed exist for dynamic-epistemic PAL and Protocol PAL, we are not sure that they extend to sabotage logics, and hence leave these matters as open problems.

3.4 Comparing link deletion and point deletion

A natural companion to link or edge deletion in graphs is deletion of vertices or points. The modal logic MLSR for stepwise point deletion of (van Benthem et al., 2020), mentioned in the introduction as the inspiration for our completeness proof, adds a modality $\langle -\varphi\rangle\psi$ for stepwise world removal to the basic hybrid modal logic:

$$\varphi ::= a \mid p \mid \bot \mid \neg\varphi \mid \varphi\wedge\varphi \mid \Diamond\varphi \mid @_a\varphi \mid \langle -\varphi\rangle\varphi$$

Formulas $\langle -\varphi\rangle\psi$ have the following truth condition in models $\mathfrak{M} = (W, R, V)$:

$$\mathfrak{M}, s \vDash \langle -\varphi\rangle\psi \;\text{ iff }\; \text{there is a } t \neq s \text{ with } \mathfrak{M}, t \vDash \varphi \text{ and } \mathfrak{M} - \{t\}, s \vDash \psi$$

In MLSR, the universal modality is definable as follows: $U\varphi := \varphi\wedge\neg\langle -\neg\varphi\rangle\top$, so it is freely available in our later proofs. It follows that the hybrid notion $@_a\varphi$ is also definable, although this notation is used as primitive in MLSR for greater perspicuity of its proof system.

3.4.1 From link deletion to point deletion

Intuitively, deleting links can be simulated by deleting points in models of the right kind. We will make this precise by embedding the logic HMSL into MLSR.

Consider the following fragment of the language of MLSR, with atomic propositions from $\mathsf{Prop}\cup\{i\}$ including a distinguished proposition letter i, and Nom[①]:

$$\varphi ::= a \mid p \mid i \mid \neg\varphi \mid \varphi\wedge\varphi \mid \Diamond(i\wedge\Diamond\varphi) \mid @_a\varphi \mid \langle -(i\wedge\Diamond\neg i)\rangle\varphi$$

We can translate the language of HSML into this fragment of MLSR.

Definition 3.12 (**Translation I**): Here is the *HSML-to-MLSR translation*:

(a) $\mathsf{Tr}(a) = a$, $\quad\mathsf{Tr}(p) = p$, $\quad\mathsf{Tr}(\neg\varphi) = \neg\mathsf{Tr}(\varphi)$, $\quad\mathsf{Tr}(\varphi\wedge\psi) = \mathsf{Tr}(\varphi)\wedge\mathsf{Tr}(\psi)$

(b) $\mathsf{Tr}(@_a\varphi) = @_a\mathsf{Tr}(\varphi)$, $\mathsf{Tr}(\Diamond\varphi) = \Diamond(i\wedge\Diamond\mathsf{Tr}(\varphi))$, $\mathsf{Tr}(\blacklozenge\varphi) = \langle -(i\wedge\Diamond\neg i)\rangle\mathsf{Tr}(\varphi)$.

Next, we define models for MLSR where this translation makes sense.

① For the purpose of this section, we can use $\langle -i\rangle\varphi$ rather than $\langle -(i\wedge\Diamond\neg i)\rangle\varphi$ in the language. But the result in Appendix 3.7 needs the latter form which can specify more information about the models.

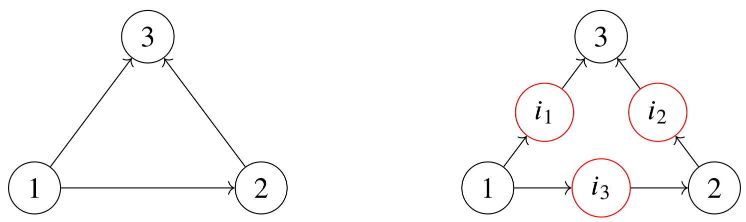
Definition 3.13 (Transformed models I): Given a model $\mathfrak{M}_0 = (W_0, R_0, V_0)$ for HSML, the model $\mathcal{F}(\mathfrak{M}_0) = (W, R, V)$ for MLSR is defined as follows:

(a) $W = W_0 \cup W_i$ where $W_i = \{(w, v, i) \mid (w, v) \in R_0 \text{ and } w, v \in W_0\}$

(b) $R = \{(w, (w, v, i)), ((w, v, i), v) \mid (w, v) \in R_0\}$

(c) $V : \mathsf{Nom} \cup \mathsf{Prop} \cup \{i\} \to W$ is a valuation function such that $V(o) = V_0(o)$ for $o \in \mathsf{Prop} \cup \mathsf{Nom}$ and $V(i) = W_i$.

Example 3.1: In Figure 3.1, $\mathcal{F}(\mathfrak{M}_0)$ is the transformed model of $\mathfrak{M}_0$. The link $(1, 3)$ is represented by i_1 in the transformed model. The sentence 'I can travel from 1 to 3' can be faithfully translated as 'I can first travel from 1 to i_1, and then to 3', while deleting the link $(1, 3)$ can be faithfully represented as deleting a point in the model $\mathcal{F}(\mathfrak{M}_0)$, namely, the node i_1.



$\mathfrak{M}_0$ $\qquad\qquad$ $\mathcal{F}(\mathfrak{M}_0)$

Figure 3.1 From $\mathfrak{M}_0$ in HSML to $\mathcal{F}(\mathfrak{M}_0)$ in MLSR

Now we have the following result connecting the two languages.

Fact 15: For any formula φ in the language of HSML and any model $\mathfrak{M}_0$,

$$\mathfrak{M}_0, w \vDash \varphi \text{ iff } \mathcal{F}(\mathfrak{M}_0), w \vDash \mathsf{Tr}(\varphi),$$

where $w \in W_0$ and $\mathcal{F}(\mathfrak{M}_0)$ is constructed from $\mathfrak{M}_0$ as in the preceding definition.

A result on equivalence of validities in the two logics follows immediately.

Corollary 3.1: We have the following equivalence:

$$\vDash_{\mathcal{C}} \varphi \text{ iff } \vDash_{\mathcal{F}(\mathcal{C})} \neg i \to \mathsf{Tr}(\varphi)$$

where $\mathcal{C}$ denotes the class of all models for HMSL, while $\mathcal{F}(\mathcal{C})$ denotes the class of all MLSR models constructed from these.

However, this result does not tell us that we can embed HMSL into the logic MLSR as it stands over arbitrary models: for that, we need to define the special class $\mathcal{F}(\mathcal{C})$ in the language of MLSR.

This can be done, with the caveat that we need extend the language of MLSR by adding the reverse operator $\Diamond^{-1}$ as in temporal logic:

$\mathfrak{M}, s \vDash \Diamond^{-1}\psi$ iff $\mathfrak{M}, t \vDash \psi$ for some t with Rts.

In this extended language for MLSR, that we will denote by MLSR$^+$, we can define the special class $\mathcal{F}(\mathcal{C})$ using the formulas listed in Table 3.3.

a. $i \to (\Diamond\neg i \wedge \langle -\top\rangle \neg \Diamond \top)$	1. an i point has exactly one successor, which is a $\neg i$ point
b. $i \to (\Diamond^{-1}\neg i \wedge \langle -\top\rangle \neg \Diamond^{-1} \top)$	2. an i point has exactly one predecessor, which is a $\neg i$ point
c. $\neg i \to \Box i$	3. if a $\neg i$ point has successors, then they are i points
d. $\neg i \wedge \Diamond\top \to [-\neg i]\langle -\top\rangle \Box \Diamond\top$	4. if a $\neg i$ point has two or more different i successors, then these cannot have the same successor.

Table 3.3 Defining $\mathcal{F}(\mathcal{C})$ in MLSR$^+$

Proposition 3.1:

$$\mathfrak{M} \in \mathcal{F}(\mathcal{C}) \text{ iff } \mathfrak{M} \vDash_{\mathrm{MLSR}^+} A$$

where A is the conjunction of the four MLSR formulas in Table 3.3.

Proof For a start, note that all the listed properties a,b,c,d in Table 3.3 hold for all models in $\mathcal{F}(\mathcal{C})$. Next, it is easy to see that, for any model $\mathfrak{M}$, $\mathfrak{M} \vDash$ a iff it satisfies property 1, and likewise for b and 2, and c and 3. Given this, it suffices to focus on establishing the following claim:

If $\mathfrak{M}$ satisfies properties 1, 2 and 3, $\mathfrak{M} \in \mathcal{F}(\mathcal{C})$ iff $\mathfrak{M} \vDash$ d.

From left to right, assume that $\mathfrak{M} \in \mathcal{F}(\mathcal{C})$. Given any point w in $\mathfrak{M}$ with $\mathfrak{M}, w \vDash \neg i \wedge \Diamond\top$, we prove that $\mathfrak{M}, w \vDash [-\neg i]\langle -\top\rangle \Box\Diamond\top$. When deleting any $\neg i$ point $v \neq w$, there are two cases. Case 1: The deleted $\neg i$ point $v \neq w$ is the successor of some i successor of w. In this case, by deleting this i-predecessor of v, since $\mathfrak{M} \in \mathcal{F}(\mathcal{C})$, all other i-successors of w must have a $\neg i$ successor. Case 2: Otherwise, deleting any i point suffices to keep $\Box \Diamond \top$ true at w.

From right to left, given a model $\mathfrak{M}$ which satisfies properties 1, 2 and 3, but lacks 4, assume that some $\neg i$ world w has two i successors s and t which share the same successor v. By the assumption then $\mathfrak{M}, w \vDash \neg i \wedge \Diamond\top$. Moreover, v is a $\neg i$ world and the unique successor of both s and t because $\mathfrak{M}$ has properties 1, 2. Next we prove that $\mathfrak{M}, w \vDash \langle -\neg i\rangle[-\top] \Diamond \Box\bot$. After deleting the world v, w has two successors s and t without a successor. Thus no matter which world we choose to delete($[-\top]$), $\Diamond \Box \bot$ is satisfied at

w. It follows that $\mathfrak{M}, w \nvDash d$. ∎

Now we have the following result connecting the two languages.

Corollary 3.2: For every formula φ in the language of HSML,

$$\vDash_{\text{HSML}} \varphi \text{ iff } \vDash_{\text{MLSR}^+} UA \to (\neg i \to \mathsf{Tr}(\varphi))$$

Note that here we add the universal operator U in front of A, because UA rather than A can make sure the model of MLSR$^+$ which refutes $UA \to (\neg i \to \mathsf{Tr}(\varphi))$ *at a certain world*, according to Proposition 3.1, is a model in $\mathcal{F}(\mathcal{C})$.

In Appendix 3.7, we present a slightly more complex method for obtaining an analogue to Corollary 3.2 which needs no extension of the language of MLSR.

Digression: sabotage games. The above model transformation also implies the equivalence of the multi-link version of the sabotage game with single-point destinations, (van Benthem, 2005), and the single-link version with multiple destinations. This result first appeared in Lemma 1 of (Löding and Rohde, 2003b). We flesh out the details of its proof to show how it relates to the above embedding result.

Fact 16: Let Ind be an arbitrary set of individuals, and let the map of a multi-link version sabotage game be $\mathfrak{M}_0 = (W_0, R_0^i, V_0)$. Then Traveler has a winning strategy starting at $w \in W_0$ in $\mathfrak{M}_0$ iff Traveler has a winning strategy on w in $\mathcal{F}(\mathfrak{M}_0) = (W, R, V)$ where $W = \{g\} \cup W_0 \cup \bigcup_{i \in \mathsf{Ind}} \{(w, v, i) \mid (w, v) \in R_0^i \text{ and } w, v \in W_0\}$, $R = \{(w, (w, v, i)), ((w, v, i), v), ((w, v, i), g) \mid (w, v) \in R_0^i\}$ and the valuation function V is the same as the old V_0 except that it makes the new node g one of the goals.

Before giving the proof, we first illustrate the definition of $\mathcal{F}(\mathfrak{M}_0)$ in the fact and how it works in the proof by the following example.

Example 3.2: In Figure 3.2, Traveler has a winning strategy on a node in $\mathfrak{M}_0$ iff Traveler has a winning strategy on the corresponding node in $\mathcal{F}(\mathfrak{M}_0)$. The added goal point g_1 plays a crucial role here. The upper $(1, 2)$ link is represented by the three links $(1, i_3)$, $(i_3, 2)$, (i_3, g) in the transformed model in the sense that Traveler can still move from 1 to 2 as long as the three links all remain untouched, while such a travel is no longer possible once at least one of the three links has been deleted.

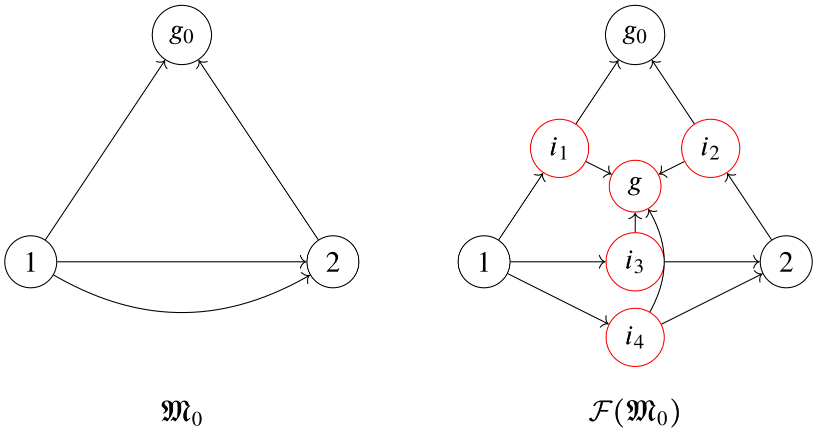


Figure 3.2 From the multi-link model $\mathfrak{M}_0$ to a single-link model $\mathcal{F}(\mathfrak{M}_0)$

Proof To start our proof, for $(a, b) \in R$, we define this mapping f into $\bigcup_i R_0^i$:

$$f((a,b)) = \begin{cases} (u,v)_i \text{ if } & (a,b) = (u,(u,v,i)) \\ (u,v)_i \text{ if } & (a,b) = ((u,v,i),v) \\ (u,v)_i \text{ if } & (a,b) = ((u,v,i),g) \end{cases}$$

We show that deleting a link (a, b) in $\mathcal{F}(\mathfrak{M}_0)$ is equivalent to deleting $f((a, b))$ in $\mathfrak{M}$, while deleting $(u, v)_i$ in $\mathfrak{M}$ is equivalent to deleting any (a, b) such that $f((a, b)) = (u, v)_i$ in $\mathcal{F}(\mathfrak{M}_0)$. More precisely, Traveler can go from u to v through R_i in $\mathfrak{M}$ iff Traveler can go from u to v via (u, v, i) in $\mathcal{F}(\mathfrak{M}_0)$. Based on the mapping f defined above, both Traveler and Demon can derive winning strategies in one model from winning strategies in the other. We have two cases.

Case 1. If Traveler can move from u to v by R_i in $\mathfrak{M}_0$, then $(u, v)_i$ was not deleted. So in $\mathcal{F}(\mathfrak{M}_0)$, $(u, (u, v, i))$, $((u, v, i), v)$, $((u, v, i), g)$ are all not deleted. On u, Traveler first goes to (u, v, i) through the link $(u, (u, v, i))$. Then Demon has to delete the link $((u, v, i), g)$, or Traveler will win immediately. Traveler now moves to v through the link $((u, v, i), v)$.

Case 2. If Traveler cannot go from u to v by R_i in $\mathfrak{M}$, then $(u, v)_i$ was deleted. So in $\mathcal{F}(\mathfrak{M}_0)$, at least one of $(u, (u, v, i))$, $((u, v, i), v)$, $((u, v, i), g)$ was deleted. If $(u, (u, v, i))$ or $((u, v, i), v)$ was deleted, then there is no path from u to (u, v, i) to v. If $((u, v, i), g)$ was deleted, then Demon can cut $((u, v, i), v)$ when Traveler reaches (u, v, i). Therefore, Traveler can no longer go to v through (u, v, i). ∎

3.4.2 From point deletion to link deletion

At this point, it is natural to seek a converse system embedding, adding a converse modality $\Diamond^{-1}$ to HSML to match the extension we made for MLSR. However, there is a mismatch here, since point removal in MLSR refers to a formula, while link deletion is arbitrary in HSML. Still, an embedding may be obtained by generalizing the link deletion operator $\blacklozenge$ of HSML to a conditional version.

More precisely, we embed MLSR$^+$ into an extended logic HSML$^+$, whose language is as follows with $a \in \mathsf{Nom}$, $p \in \mathsf{Prop}$ and e a special nominal:

$$\varphi ::= a \mid e \mid p \mid \neg\varphi \mid \varphi \wedge \varphi \mid \Diamond\varphi \mid \Diamond^{-1}\varphi \mid \blacklozenge^{\varphi}_{\varphi}\varphi$$

The truth condition for $\blacklozenge^{\varphi}_{\psi}\chi$ reads: ①

$\mathfrak{M} = (W, R, V), w \vDash \blacklozenge^{\varphi}_{\psi}\chi$ iff there is a pair $(s,t) \in R$

such that $\mathfrak{M}, s \vDash \varphi$, $\mathfrak{M}, t \vDash \psi$ and $(W, R \setminus (s,t), V), w \vDash \chi$.

Note that this covers the original sabotage modality as the special case $\blacklozenge^{\top}_{\top}\chi$. However, even using its named deletions, HSML$^+$ cannot define the universal modality $U\varphi$. But we do have the following restricted version

$$\forall\varphi := \neg\blacklozenge^{\neg\varphi}_{\top}\top \wedge \neg\blacklozenge^{\top}_{\neg\varphi}\top$$

This says that no links can be cut to or from $\neg\varphi$-worlds (such worlds will be called *isolated*), or in other words, all worlds which are in a R-relation with some world satisfy φ. This notion will suffice for our later purposes.

Next, the special nominal e is key to making point deletions in an MLSR$^+$ model become link deletions in a matching HSML$^+$ model.

Definition 3.14 (Transformed models II): Given a model $\mathfrak{M}_0 = (W_0, R_0, V_0)$ for MLSR$^+$, the model $\mathcal{G}(\mathfrak{M}_0) = (W, R, V)$ for HSML$^+$ is defined as follows:

(a) $W = W_0 \cup \{w_e\}$

(b) $R = R_0 \cup \{(w, w_e)) \mid w \in W_0\}$

(c) $V : \mathsf{Nom} \cup \mathsf{Prop} \cup \{e\} \to W$ is a valuation function such that $V(o) = V_0(o)$ for $o \in \mathsf{Prop} \cup \mathsf{Nom}$ and $V(e) = \{w_e\}$.

① $\blacklozenge^{\varphi}_{\phi}\psi$ cannot be defined in the language of HSML. Let $M_1 = (W_1, R_1, V_1)$ with $W_1 = \{w_1, w_2\}, R_1 = \{(w_1, w_1), (w_2, w_2)\}, V(p) = \{w_2\}, V(a) = \{w_1\}$ for any $a \in \mathsf{Nom}$. $M_2 = (W_2, R_2, V_2)$ with $W_2 = \{v_1, v_2\}, R_2 = \{(v_1, v_1), (v_2, v_2)\}, V(p) = \varnothing, V(a) = \{v_1\}$ for any $a \in \mathsf{Nom}$. It is easy to see that there is an HSML-style bisimulation (cf. Appendix 3.6) between (M_1, w_1) and (M_2, v_1), and so each formula α is true at w_1 iff α is true at v_1. However, $M_1, w_1 \vDash \blacklozenge^{a}_{\top}\blacklozenge^{p}_{\top}a$, $M_2, v_1 \nvDash \blacklozenge^{a}_{\top}\blacklozenge^{p}_{\top}a$.

The following translation makes the effect of deleting a point v in $\mathfrak{M}_0$ equivalent to the effect of deleting the corresponding link (v, w_e) in $\mathcal{F}(\mathfrak{M}_0)$.

Definition 3.15 (**Translation II**): The MLSR$^+$-*to*-HSML$^+$ *translation* is this:

(a) $\mathbf{Tr}(o) = o, \quad \mathbf{Tr}(\neg\varphi) = \neg\mathbf{Tr}(\varphi), \quad \mathbf{Tr}(\varphi \wedge \psi) = \mathbf{Tr}(\varphi) \wedge \mathbf{Tr}(\psi),$

(b) $\mathbf{Tr}(\Diamond\varphi) = \Diamond(\Diamond e \wedge \mathbf{Tr}(\varphi)), \qquad \mathbf{Tr}(\Diamond^{-1}\varphi) = \Diamond^{-1}(\Diamond e \wedge \mathbf{Tr}(\varphi)),$

(c) $\mathbf{Tr}(@_a\varphi) = @_a(\Diamond e \wedge \mathbf{Tr}(\varphi)), \quad \mathbf{Tr}(\langle -\varphi\rangle\psi) = \blacklozenge_e^{\mathbf{Tr}(\varphi)\wedge\Diamond e}(\Diamond e \wedge \mathbf{Tr}(\psi))$

This translation *relativizes* the operators in MLSR$^+$-formulas φ syntactically to refer only to those worlds in $\mathcal{G}(\mathfrak{M}_0)$ that satisfy either $\Diamond e$ or e. An easy induction on formulas implies the following semantic invariance property:

Proposition 3.2: For all HSML-models $\mathfrak{N}$, and all worlds w satisfying $\Diamond e$, we have that $\mathfrak{N}, w \vDash \mathbf{Tr}(\varphi)$ iff $\mathfrak{N} \mid (\Diamond e \vee e), w \vDash \mathbf{Tr}(\varphi)$, where $\mathfrak{N} \mid (\Diamond e \vee e)$ is the submodel of $\mathfrak{N}$ consisting of all worlds that satisfy $\Diamond e \vee e$.

In particular, once a link R from a world v to w_e has been deleted in a model $\mathcal{G}(\mathfrak{M}_0)$, v falls outside of the relativized model $\mathcal{G}(\mathfrak{M}_0) \mid (\Diamond e \vee e)$ and plays no role any more in the evaluation of translated formulas. Thus, the effect on such formulas is the same as if the world v had been deleted.

These observations are the key to the following result.

Fact 17: For any formula φ of MLSR$^+$, any model $\mathfrak{M}_0$ and world $w \in W_0$,

$$\mathfrak{M}_0, w \vDash \varphi \text{ iff } \mathcal{G}(\mathfrak{M}_0), w \vDash \mathbf{Tr}(\varphi),$$

where $\mathcal{G}(\mathfrak{M}_0)$ is constructed from $\mathfrak{M}_0$ as in Definition 3.14.

Proof The proof is by induction, and we only sketch the crucial case of the point-deletion modality. Recall that $\mathfrak{M}_0, s \vDash \langle -\varphi\rangle\psi$ iff there exists a world $t \neq s$ such that $\mathfrak{M}_0, t \vDash \varphi$ and $\mathfrak{M}_0 - \{t\}, s \vDash \psi$, where $\mathfrak{M}_0 - \{t\}$ is the submodel of $\mathfrak{M}_0$ in which the world t has been deleted. By the inductve hypothesis, we have that (a) $\mathcal{G}(\mathfrak{M}_0), t \vDash \mathbf{Tr}(\varphi)$, and (b) $\mathcal{G}(\mathfrak{M}_0 - \{t\}), s \vDash \mathbf{Tr}(\psi)$. To see that the formula $\mathbf{Tr}(\langle -\varphi\rangle\psi)$ as defined above is true at s in $\mathcal{G}(\mathfrak{M}_0)$, we cut the link from t to e, and need to have $\mathbf{Tr}(\psi)$ true at s. However, this follows from (b) above plus Proposition 2, since $\mathcal{G}(\mathfrak{M}_0 - \{t\})$ equals the relativization of the model $\mathcal{G}(\mathfrak{M}_0)$ after the link cut from t to e to only those worlds that satisfy $\Diamond e \vee e$. ∎

Our remaining task is to suitably define the class of models $\mathcal{G}(\mathcal{C}) = \{\mathcal{G}(\mathfrak{M}_0) \mid \mathfrak{M}_0 \in \mathcal{C}\}$, where $\mathcal{C}$ is the class of all models for MLSR$^+$. We start with two simple auxiliary observations about the defining HSML$^+$-formula (where we recall that our special universal modality $\forall$ ranges only over non-isolated points).

Proposition 3.3: For any HMSL$^+$ model $\mathfrak{N}$,

(a) If $\mathfrak{N} \in \mathcal{G}(\mathcal{C})$, then $\mathfrak{N}, w \vDash \forall((\neg e \to \Diamond e) \wedge (e \to \Box\bot))$

(b) Let $\mathfrak{N} - \mathsf{ISO}$ be the model obtained by removing all isolated worlds in $\mathfrak{N}$.
If, for some world w, $\mathfrak{N}, w \vDash \forall((\neg e \to \Diamond e) \wedge (e \to \Box\bot))$, then $\mathfrak{N} - \mathsf{ISO} \in \mathcal{G}(\mathcal{C})$.

We now obtain the following reduction from MLSR$^+$ to HMSL$^+$.

Fact 18: For each formula φ in the language of MLSR$^+$,

$$\vDash_{\mathrm{MLSR}^+} \varphi \text{ iff } \vDash_{\mathrm{HSML}^+} \forall((\neg e \to \Diamond e) \wedge (e \to \Box\bot)) \to (\neg e \wedge \Diamond e \to \mathsf{Tr}(\varphi))$$

Proof From right to left, this is straightforward. Suppose that the stated formula is valid in HSML$^+$, and let $\mathfrak{M}, s$ be any pointed model for MLSR$^+$. By Proposition 3.(a), we have that the HSML$^+$-model $\mathcal{G}(\mathfrak{M}) \vDash \forall((\neg e \to \Diamond e) \wedge (e \to \Box\bot))$. By the definition of the mapping $\mathcal{G}$, we have that s satisfies $\neg e \wedge \Diamond e$. It follows from the assumption that $\mathcal{G}(\mathfrak{M}) \vDash \mathbf{Tr}(\varphi)$, and so by Fact 12, $\mathcal{M}, s \vDash \varphi$.

From left to right, we argue by contraposition. Suppose that some HSML$^+$-model $\mathfrak{N}$ and world s make the following formulas true: (a) $\forall((\neg e \to \Diamond e) \wedge (e \to \Box\bot))$, (b) $\neg e$, (c) $\Diamond e$, and (d) $\neg\mathbf{Tr}(\varphi)$. Now remove all isolated points from $\mathfrak{N}$ to obtain the model $\mathfrak{N}$-ISO. It is easy to verify that, in this model, (a), (b) and (c) above still hold at s. [In particular, given the assumptions, neither s nor e are isolated points, so they stay in.] But (d) remains true as well, by an appeal to Proposition 2. The reason is that, given the truth of (a), $\mathfrak{N}$-ISO equals the relativized model $\mathfrak{N} \mid (\Diamond e \vee e)$. But then, finally, Proposition 3.(b) gives us an MLSR$^+$-model $\mathfrak{M}$ with $\mathcal{G}(\mathfrak{M}) = \mathfrak{N}$-ISO where φ is false at s. ∎

Finally, we close the circle of our two system embeddings so far by showing that the extended language HSML$^+$ can also be embedded into MLSR$^+$. One just extends the translation function in Definition 3.12 by adding the two clauses

$$\mathsf{Tr}(\Diamond^{-1}\varphi) = \Diamond^{-1}(i \wedge \Diamond^{-1}\mathsf{Tr}(\varphi))$$
$$\mathsf{Tr}(\blacklozenge^{\varphi}_{\psi}\chi) = \langle -(i \wedge \Diamond^{-1}(\neg i \wedge \mathsf{Tr}(\varphi)) \wedge \Diamond(\neg i \wedge \mathsf{Tr}(\psi)) \rangle \mathsf{Tr}(\chi)$$

It is not hard to verify that we have the following new result:

Proposition 3.4: For any formula φ in the language of HSML$^+$,

$$\vDash_{\mathrm{HSML}^+} \varphi \text{ iff } \vDash_{\mathrm{MLSR}^+} UA \to (\neg i \to \mathsf{Tr}(\varphi))$$

Thus, we can embed slight extensions of HSML into matching extensions of MLSR and vice versa. This gives substance to the intuition that point deletion and link deletion are closely related in a logical perspective.① We leave obtaining sharper and more parsimonious reductions as an open problem.②

3.5 Conclusion

We have axiomatized the logic HSML of arbitrary link deletion in a Hilbert-style format using modest hybrid additions to the original language of sabotage modal logic which allow for defining an auxiliary companion modality of named link deletion simplifying the proof system. In addition, we have used our setting to provide mutual reductions between existing modal logics of point deletion and link deletion that suggest more unity to logics of graph-changing games than might have been apparent at first sight.

We believe that the technique of axiomatization via a companion modality definable in the logic, which simplifies the one in (van Benthem et al., 2020) to which our general treatment remains indebted, can be applied to many further logics of graph change in the literature, and also to further kinds of semantics beyond the protocol models whose logic we have axiomatized.

In our view, two major open problems remain for judging the virtues of working with HSML. A first concern are the *schematically valid* formulas of HSML that remain valid under substitution of arbitrary formulas for atomic formulas. Most, but not all of the principles in our axiomatizations were schematically valid: in particular, the recursion axiom for proposition letters was not. (Holliday et al., 2011) axiomatizes the schematically valid formulas of public announcement logic using an abstract poly-modal semantics with modal and dynamic accessibility relations which can be seen as a generalization of our protocol models, cf. also (Wang, 2010). We believe that our approach lifts to such a setting, but this needs to be verified.

Another major question concerning HSML (also open for its parent logic SML) is an *interpolation theorem*. It is not hard to see that the proof techniques for hybrid logic in

① Point deletion and arrow deletion are also close in *Arrow Logic*, (van Benthem, 1996), which treats arrows as objects representing transitions, but we have not been able to establish a precise connection between this research line and our logics of graph change.

② We have encountered quite a few forms of deletion by now. PAL deletes all points satisfying a certain property, and the counterpart for this is the DEL-style logic of uniform definable link cutting in Appendix 3.6. One can also delete definable objects or definable links stepwise, as we have analyzed here. As a specialization of this, there is deletion of arbitrary points or links, or just individual named objects or links. We leave a comparison of the latter variants to further study.

(Areces et al., 2001) extend to HSML, but they require adding downarrow binders to our language which lack motivation in the setting of graph games. Whether we can do without them is an open problem at the present stage.

Finally, as already suggested earlier, our results for HSML can also be seen as a case study for broader issues in dynamic-epistemic and hybrid logics. We leave these to further investigation.

3.6 Appendix A: HSML with general link cutting

The nominal link cutting operator $\langle a|b\rangle$ in HSML sufficed for proving completeness. But dynamic-epistemic logic has complete systems for general link cutting modalities $\langle\varphi|\psi\rangle$ that describe the new model after cutting all the links $\{(w,v)\mid \mathfrak{M},w\vDash\varphi$ and $\mathfrak{M},v\vDash\psi\}$ simultaneously from a current model, (van Benthem and Liu, 2007).

Fact 19: General definable link cutting is not definable in HSML.

Proof We first extend the SML-bisimulations of (Aucher et al., 2018) as follows.

Definition 3.16 (HS-bisimulation): Let $\mathcal{M}_1=(W_1,R_1,V_1)$, $\mathcal{M}_2=(W_2,R_2,V_2)$ be models for the language $\mathcal{L}$. A relation $Z\subseteq W_1\times W_2$ is a *HS-bisimulation* between M_1 and M_2 if the following conditions are satisfied:

(a) for atoms: if w_1Zw_2, then $w_1\in V_1(a)$ iff $w_2\in V_2(a)$ for $a\in\mathsf{Prop}\cup\mathsf{Nom}$.

(b) forth and back conditions for $\Diamond$ are as usual.

(c) forth condition for $\blacklozenge$: if w_1Zw_2, $\mathcal{M}_1'$ is a new model obtained from $\mathcal{M}_1$ by cutting a link, then there exists a new model $\mathcal{M}_2'$ obtained from $\mathcal{M}_2$ by cutting a link such that w_1Zw_2, where Z is an HS-bisimulation between $\mathcal{M}_1'$ and $\mathcal{M}_2'$. The back condition for $\blacklozenge$ is the obvious converse.

(d) All points named by the same nominal are related by Z.

The following is easy to prove by induction on formulas.

Fact 20: HSML-formulas are invariant for HS-bisimulations.

Now we can give a concrete example to prove that $\langle\varphi|\psi\rangle$ cannot be defined in HSML. Let $M_1=(W_1,R_1,V_1)$ with $W_1=\{w_1,w_2,w_3\},R_1=\{(w_2,w_3)\},V(p)=\{w_2\},V(a)=\{w_1\}$ for any $a\in\mathsf{Nom}$. $M_2=(W_2,R_2,V_2)$ with $W_2=\{v_1,v_2,v_3\},R_2=\{(v_2,v_3)\},V(p)=\varnothing,V(a)=\{v_1\}$ for any $a\in\mathsf{Nom}$. It is easy to see that $(M_1,w_1)Z(M_2,v_1)$, which means that any formula α is true at w_1 iff α is true at v_1.

However, $M_1, w_1 \nvDash \langle p|\top\rangle\blacklozenge\top$, $M_2, v_1 \vDash \langle p|\top\rangle\blacklozenge\top$, which leads to a contradiction. ∎

Adding a general link cutting operator $\langle\varphi|\psi\rangle$ to HSML yields a logic GHSML whose syntax is given by

$$\varphi ::= a \mid p \mid \neg\varphi \mid \varphi\wedge\varphi \mid \Diamond\varphi \mid \blacklozenge\varphi \mid @_i\varphi \mid \langle\varphi|\psi\rangle\alpha$$

with $p \in \mathsf{Prop}$, $a \in \mathsf{Nom}$. Dual modal operators $\Box, \blacksquare, [\varphi|\psi]$ are defined as usual. The truth condition for $\langle\varphi|\psi\rangle$ is as follows:

$(W, R, V), w \vDash \langle\varphi_1|\varphi_2\rangle\psi$ iff $(W, R', V), w \vDash \psi$, where

$R' = R\backslash\{(w_1, w_2) \mid (W, R, V), w_i \vDash \varphi_i \text{ for } i = 1, 2\}$

The logic GHSML can be axiomatized in the same style as HSML, with the one difference that the link cutting modality is now a primitive of the system, for which we have recursion axioms in standard dynamic-epistemic style:

(a) $\langle\varphi|\psi\rangle a \leftrightarrow a$

(b) $\langle\varphi|\psi\rangle p \leftrightarrow p$

(c) $\langle\varphi|\psi\rangle\neg\alpha \leftrightarrow \neg\langle\varphi|\psi\rangle\alpha$

(d) $\langle\varphi|\psi\rangle(\alpha\wedge\beta) \leftrightarrow \langle\varphi|\psi\rangle\alpha \wedge \langle\varphi|\psi\rangle\beta$

(e) $\langle\varphi|\psi\rangle @_a\alpha \leftrightarrow @_a\langle\varphi|\psi\rangle\alpha$

(f) $\langle\varphi|\psi\rangle\Diamond\alpha \leftrightarrow ((\varphi\wedge\Diamond(\neg\psi\wedge\langle\varphi|\psi\rangle\alpha)) \vee (\neg\varphi\wedge\Diamond\langle\varphi|\psi\rangle\alpha))$

A completeness proof can be given for this extended proof system in the same style as the one we gave for HSML, though its details will now be closer to the completeness proof for MLSR in (van Benthem et al., 2020).

3.7 Appendix B: Another approach to embedding HSML into MLSR

We assume the setting of Section 5, but now introduce the following notions in order to tighten up the translation provided there.

Definition 3.17 (Named Pseudo Transformed models): A *named pseudo transformed model* $\mathfrak{M}_P$ is a named MLSR model in which for any $a, b \in \mathsf{Nom}$, the following formulas are true globally: $i \to (\Diamond a \to \Box(a\wedge\neg i))$, $\neg i \to \Box i$, $\Diamond(b\wedge i\wedge\Diamond a) \to \neg\Diamond(\neg b\wedge i\wedge\Diamond a)$ and $b\wedge\Diamond(i\wedge a) \to @_b\neg i\wedge(\neg b \to \neg\Diamond a)$.

A named pseudo transformed model ('nptm', for short) is close to a transformed model, but there are some differences. In an nptm, i-worlds may have neither successors nor predecessors. But in a transformed model, an i-world must have both a $\neg i$-successor and a $\neg i$-predecessor. However, in an nptm, if an i-world has a successor or a predecessor, it must be a $\neg i$-world and unique.

Let $\mathcal{S}$ denote the class of all named pseudo transformed models.

Proposition 3.5: $\neg i \wedge \mathsf{Tr}(\varphi)$ is satisfiable in a transformed model iff $\neg i \wedge \mathsf{Tr}(\varphi)$ is satisfiable in a named pseudo transformed model.

Proof First, given a transformed model $\mathcal{F}(\mathfrak{M}_0)$ and a translated formula $\mathsf{Tr}(\varphi)$ true at a $\neg i$-point w in it, by adding nominals to the language and naming all points in $\mathcal{F}(\mathfrak{M}_0)$, we get an nptm with $\mathsf{Tr}(\varphi)$ still true at w.

Next, given an nptm $\mathfrak{M}_P$, we delete all i-points without successors and then add for each i-point without predecessors a $\neg i$-point linking to it (with no restriction on how atomic propositions except for i are assigned to these new points). Let $\mathfrak{M}'_P$ be the resulting transformed model. Then we prove by formula induction that, for any formula $\neg i \wedge \mathsf{Tr}(\varphi)$ and $\neg i$-point in both $\mathfrak{M}_P$ and $\mathfrak{M}'_P$, the formula is true at w in $\mathfrak{M}_P - B$ iff it is true at w in $\mathfrak{M}'_P - B$, where B is a finite subset of i-points which are in both $\mathfrak{M}_P$ and $\mathfrak{M}'_P$.[1] ∎

Corollary 3.3:

$$\vDash_{\mathcal{C}} \varphi \text{ iff } \vDash_{\mathcal{S}} \neg i \to \mathsf{Tr}(\varphi)$$

Adding the four formulas in Definition 3.17 as axioms to obtain a proof system $\mathbf{MLSR}(\mathcal{C})$, soundness and completeness go through – noting that deleting points from an nptm still yields a named pseudo transformed model. Putting things together, one then obtains the desired

Corollary 3.4:

$$\vdash_{\mathbf{HSML}} \varphi \text{ iff } \vdash_{\mathbf{MLSR}(\mathcal{C})} \neg i \to \mathsf{Tr}(\varphi)$$

[1] Here are the key cases. (a) For $\Diamond(i \wedge \Diamond\varphi)$, note that neither adding $\neg i$-points to initial i-points nor deleting dead end i-points affects the links that make $\Diamond(i \wedge \Diamond\varphi)$ true at w in $\mathfrak{M}_P$. Therefore, by the inductive hypothesis, $\Diamond(i \wedge \Diamond\varphi)$ is also true at w in $\mathfrak{M}'_P$. From $\mathfrak{M}'_P$ to $\mathfrak{M}_P$, the same argument applies. (b) For $\langle -i \wedge \Diamond\neg i\rangle\varphi$, note that neither the deleted points nor the added points satisfy $i \wedge \Diamond\neg i$, so when evaluating the formula $\langle -i \wedge \Diamond\neg i\rangle\varphi$, we can always delete the same points in $\mathfrak{M}_P$ and $\mathfrak{M}'_P$ that satisfy $i \wedge \Diamond\neg i$. The equivalence follows by the inductive hypothesis.

3.8 Appendix C: On sabotage games with imperfect information

HSML has strong expressive power for giving an accurate representation of game actions at the syntax level. However, like SML, HSML has just a language for describing game boards, making both graph logics that focus on changes in the graph that represents the game board during the game. In addition to this basic structure, exploring how players access and process information, form and update knowledge and beliefs, and make decisions based on them is crucial in game theory. In this appendix, we briefly explore how some of these themes might be introduced in our setting.

In terms of players' information, there can be two limitations that deviate from the perfect information in our graph games studied so far. First, players can have bounded computational resources. For instance, in the game of Go, players cannot realistically calculate the optimal response strategy during play of the game and they may not have perfect recall of the history of the game so far. The second limitation arises from restricted ability to access information by observation. For example, in realistic sabotage games, it makes sense to assume that players cannot observe the entire game board, but just a part of it. We only consider the second source of imperfect information in what follows.

The literature has some approaches that tend in this direction. One recent approach is that of (Li et al., 2023) on so-called Cops and Robbers games which draws on the work of Grossi and Turrini (2012) on 'short sight games' to produce a sophisticated epistemic-action logic which describes the changes in player's knowledge as they make new observations in the course of the game. We believe that this epistemized approach will also work for sabotage games, but in what follows, we describe a simpler (for our present modest purposes, that is) and more classical perspective using classical extensive games with imperfect information (Osborne and Rubinstein, 1994).

For a start, we can describe a sabotage game as an extensive game where each node in the game tree is labelled with a graph and a specified position for Traveler (Demon can be assumed to be globally 'omnipresent').

Example 3.3: In the sabotage game of Figure 3.3, we suppose that Traveler can only see the successor nodes from her current node, while Demon can see the whole graph all the time. Play starts from the node s, and the goal region is $\{v\}$.

The matching extensive form game is depicted in Figure 3.4, where each node in the game tree is labelled with a graph in Figure 3.5. On such game trees, we can have imperfect information in the standard game-theoretic sense of nodes linked by epistemic

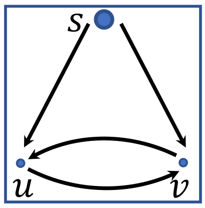


Figure 3.3 A sabotage game

indistinguishability for the relevant player, or nodes forming 'information sets' for that player. In our figure, we use dashed lines to indicate epistemic indistinguishability relations for the Traveler. For instance, at nodes b_3 and b_4, Traveler cannot observe Demon's previous action (i.e., the precise link that was cut), since this was out of Traveler's sight, while the available actions for Traveler in both situations are the same. Hence Traveler cannot distinguish these two game board states even if she has perfect recall.

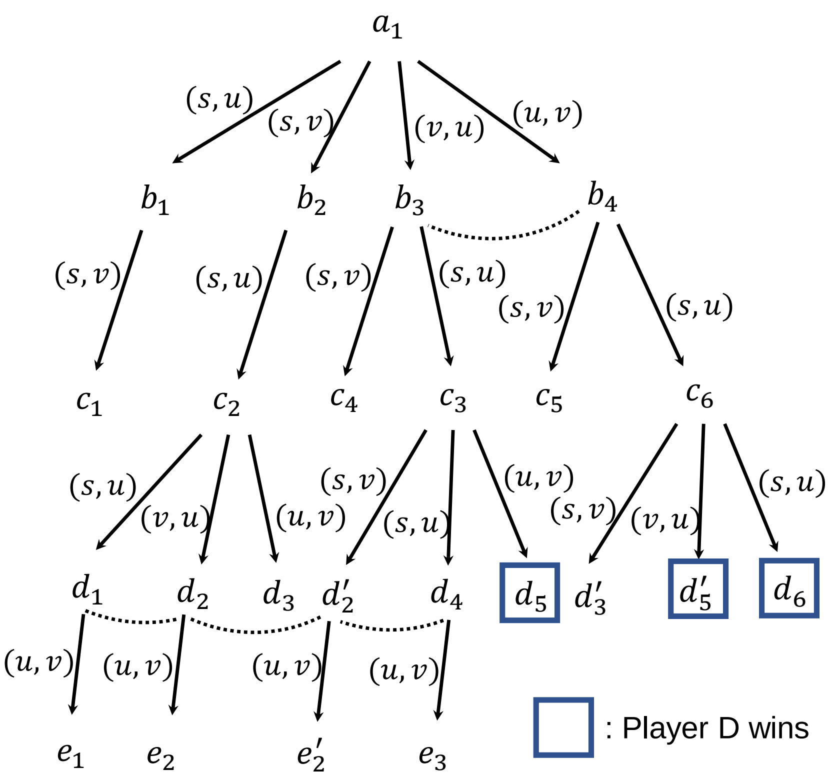


Figure 3.4 An extensive sabotage game

For a logical analysis of these game trees, we can use the game logics introduced in (van Benthem, 2014). First, we have action modalities stepping from one game node to another that represent Traveler's moves, much like in the graph logics SML and HSML. Next, the imperfect information structure where Traveler cannot distinguish between some different nodes (states of the game so far) immediately suggests introducing knowledge modalities for what she knows. Finally, the update mechanism that takes us from a layer

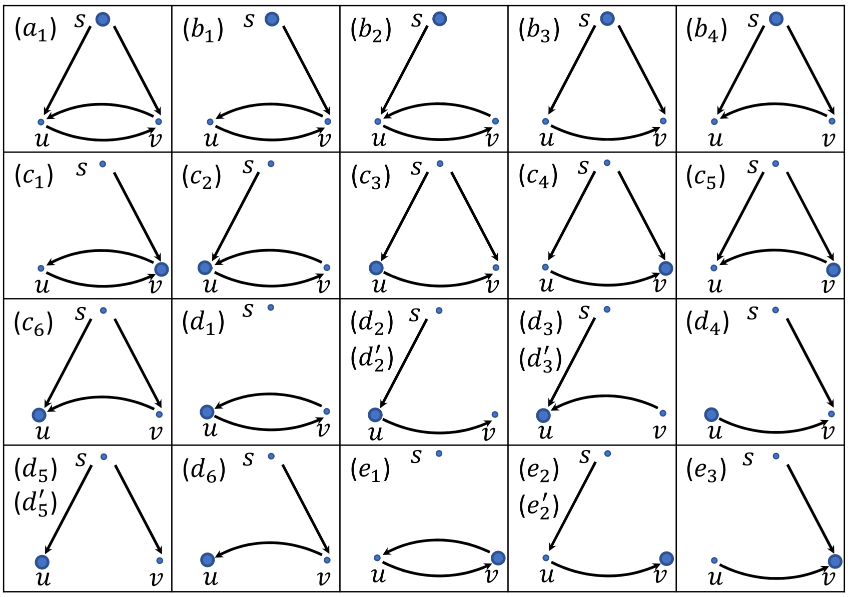


Figure 3.5 Game states

of the game tree (representing players' views of the game as played so far) to the next can be described using suitable event models for partial observation (of Demon's actions) and applying DEL-style product update, as explained in detail in (van Benthem, 2014, Chapter 9). Taken together, all this represents a dynamic-epistemic extension of our graph logics for sabotage games that seems worth studying.

Finally, in addition to what was introduced so far, these extensive game models also allow us to say more about the strategies of players, i.e., the choice rules for behavior throughout the game that ensure certain goals, such as Traveler reaching the distinguished goal region in the graph. This would require a richer game logic where we could now also talk about strategic Nash equilibria and other basic game-theoretic notions in the setting of graph games with limited observation.

CHAPTER 4 MODAL LOGICS FOR REASONING IN DISTRIBUTED GAMES

Over the years, distributed systems have been widely studied in distinct areas of applications in computer science and AI, e.g., concurrent processes, database management systems, mobile and ubiquitous computing, ad hoc networks, biological systems, intelligent interactive systems, to name a few. According to van Steen and Tanenbaum (2017), a distributed system is a collection of autonomous computing elements that appears to its users as a single coherent system. These computing elements are independent of each other but can communicate or share messages depending on various patterns. In this context, we should first note that the term ‘distributed system’ has been widely used in the literature, and its meaning has evolved accordingly. For instance, Barwise and Seligman (1997) referred to it as a structure describing the flow of information. This structure is divided into several parts, with information channels connecting different parts. For the purpose of this chapter, we consider distributed systems as concurrent systems of processes, with some amount of communication between them. Operational models of such systems are provided by labelled transition systems.

Mohalik and Walukiewicz (2003) brought a game-theoretical perspective to distributed systems. From such a game point of view, we treat a component process as a local arena for an individual player, whereas the distributed system consisting of all these processes forms a global arena. A similar formulation on local and global arenas was provided by Ramanujam and Simon (2010). In these games, a player makes choices locally, based on her game state in the local arena and information received from other players in form of public announcements that allow the player to make assumptions about other players’ local game states. The global or the product arena resolves conflicts and ambiguities arising from such partial knowledge of individual process or player. Overall, the main premise of such game analyses is that local moves of players affect their outcomes in the global arena.

Evidently, the literature on distributed systems and distributed games is quite extensive. For instance, Coulouris et al. (2012) systematically explored the design, algorithms, and challenges involved in distributed systems. From the game perspective, Berwanger et al. (2018); Beutner et al. (2019); Muscholl and Schewe (2013) focus on the synthesis of distributed strategies in games of partial information. In terms of general studies on logical characterization of game-theoretic notions one can consider the work on describing Nash equilibrium and subgame-perfect equilibrium in normal form and extensive form games (Harrenstein et al., 2003; Liu et al., 2016), backward induction (Bonanno, 2001), players' powers in extensive form games (van Benthem, 2002), players' actions in graph games (van Benthem, 2005, 2001), among others. Ghosh et al. (2017) studied game arenas with simultaneous moves, such as repeated normal-form games, with a focus on reasoning about strategies. Das and Ramanujam (2021) proposed a modal logic in two layers for reasoning about strategies in social network games.

Coming back to the studies on distributed games, much of the literature either concentrates on algorithmic questions relating to the existence of winning strategies or equilibria, or on reasoning in the global arena on players' choices. Our point of departure in this work is to provide a formal study on how the structure of local arenas determines the structure of the global arena. In some sense, the reasoning here is structural rather than outcome based. In this sense, the work is closer in spirit to that of local temporal logics (Ramanujam, 1996; Thiagarajan and Walukiewicz, 2002), but again the difference with them lies in game-theoretic reasoning.

On the whole, this chapter dwells on reasoning underlying games of partial information from the local perspective and how this reasoning evolves when viewed under the global lens. To give a few examples of formal studies on local and global reasoning, Barwise and Seligman (1997) employed local logics to characterize the components of a system and use the concept of logic infomorphisms to capture the interactions between these components, whereas Aucher (2005) discussed the difference between local and global aspects of assignments in probabilistic and plausibility approaches towards belief revision. In contrast, the discussion here stems from providing a parsimonious syntax to describe the interplay between local and global reasoning from a structural viewpoint, yielding an essential base for bringing about more sophisticated reasoning aspects with their extensions. In technical terms, this chapter proposes a couple of two-layered propositional modal logics for reasoning in distributed games and presents complete axiomatizations of their valid formulas. It finishes off with a flavour of local and global strategic reasoning phenomena that extensions and modifications of these logic frameworks can express.

Before going any further, we note that even though the main idea of local and global reasoning phenomena in distributed games comes from the notion of distributed systems

(Baier and Katoen, 2008), that is, concurrent systems of processes with some levels of communication, many real games, e.g., board games or card games can also be modeled as distributed games. Here, the occurrences of local and global reasoning correspond to the reasoning of the individual players with the information available to them vis-à-vis the reasoning from the top or high level reasoning about the games, the way we reason *about* games in game theory (Osborne and Rubinstein, 1994). Our running example for this chapter, a very simple card game described below, demonstrates this viewpoint.

Example 4.1: Two players, Alice and Bob, are playing a card game and it is common knowledge that there are three available cards, $1, 2$ and 3, say. Suppose each of them gets a card from this pile of three cards, and one card is kept upside down so that nobody can see the value. Suppose, at each round, each of them can announce the following:

1. I have card number j
2. I accept
3. I challenge

We specify that both players only announce the card number that is equal to or higher than the actual card they have. The game starts with a round of simultaneous announcements of cards.[1] Subsequently,

- if they announce different card numbers, then:
  - If the player with the announcement of the lower card announces 'I accept' in the next round, then the other player wins.
  - If the player with the announcement of the lower card announces 'I challenge' in the next round, then the other player has to show the card, and the player showing the card wins if the card matches her announcement, otherwise, the 'challenger' wins.
- if they announce the same card number, we specify that at least one player has to announce 'I challenge' in the next round. Whoever is challenged has to show the card, and this player wins if the card matches her announcement. If both players announce 'I challenge', then both have to show their cards. Both players may lose in case their cards do not match their announcements.

This is an example of a game of partial information, where each player has access to only a part of the game state and strategizes based on the local state and communication

[1] We could also allow them to make sequential announcements, and this would affect the players' strategies leading to a variation in the game analysis.

with others. In such games, players make their moves by making assumptions about other players' local states, and the actual global state decides which of the players' moves are actually enabled. This is the setting studied in this chapter, that of *distributed games*.

This simple game already illustrates a lot of salient features that have been studied in formal frameworks dealing with the reasoning processes of the participating players in different game settings. For example, from the information perspective, games of partial and/or imperfect information have been studied in the framework of dynamic epistemic logic (Ågotnes and Ditmarsch, 2011), that of epistemic situation calculus (Belle and Lakemeyer, 2010), among many others. With regard to communication, the setting builds on public announcements (Baltag et al., 1998), where the announcements may not be truthful (van Ditmarsch et al., 2012). In fact, Ågotnes and Ditmarsch (2011) introduce a notion of public announcement games. So, a natural question emerges regarding the fundamental distinguishing factor of the current study in terms of the previous studies mentioned above. As discussed earlier, we reiterate the fact that the main focus of the study presented in this chapter is to analyze and express the interplay of local and global reasoning in these games from a structural point of view. We study the simplest possible interface involving the actions (announcements) of the players, and keep the information or the epistemic aspect as well as the communication or the explicit announcement aspect for future work.

The remainder of this chapter is organized as follows: In the following section, we introduce the basic notions of local and global arenas and illustrate the distinct reasoning phenomena that we explore further. The next one provides the syntax and semantics of distributed game logic (DGL), using the language to show the interplay between the local and global views. We then provide a strong completeness result for the proposed logic. Afterwards, we propose a distributed game logic with enabled actions (DGLEA) with subtle differences from DGL to characterize more realistic interplay between local and global reasoning, and present a complete axiom system for it. In addition, we study the complexity of the model checking problem for these logics. Before concluding the chapter, we propose a similar framework that naturally incorporates strategic announcements, thereby attesting to the usefulness of the proposed frameworks for studying various reasoning phenomena in distributed games.

4.1 Preliminaries

Let $N = \{1, 2, \ldots, n\}$ denote the set of players. We assume that each player is making her moves locally based on (i) her own information about the game, which takes some form of (public) announcements, and, (ii) some possible/actual information exchange with the other players. For each player $i \in N$, we associate a finite set Γ^i, the set of symbols that player i employs for announcements/actions. Let $\tilde{\Gamma} = \prod_{i \in N} \Gamma^i$. Throughout the text, we will denote the elements of $\tilde{\Gamma}$ by γ, and use $\gamma[i]$ for i-th projection of γ, where $i \in N$. We now provide a simple setting of local and global arenas facilitating the description of the distributed games.

Definition 4.1 (Local arena): A *local arena* for player i is a tuple $\mathrm{G}^i = (W^i, \rightarrow_i, w_0^i)$ such that W^i is the set of local game states, w_0^i is the initial game state, and $\rightarrow_i$ is a partial move function given by, $\rightarrow_i : W^i \times \tilde{\Gamma} \rightarrow W^i$.

Thus, in her local arena, each player is making a move based on her own information set and the information she is receiving from the other players. The idea is that a player may not know where another player is in his local arena, but, based on the information received from the other player, she may infer some details about the other player and move accordingly in her own local arena. Let us consider the game described in Example 4.1, where the local arenas of Alice and Bob can be modeled as follows. We note that the announcements of Alice and Bob based on the cards they have can be modeled in the same way, and are given in Figure 4.1.

Example 4.2 (Local arenas: Modeling games from an individual perspective): If a player has card j, then her local arena is given by the tree starting with the number j. A move is labelled by a pair of announcements, the first coordinate represents Alice's announcement and the second one represents Bob's. There are some auxiliary symbols in the diagram explained as follows:

- Number n represents 'I have card number n'
- Action A represents 'I accept'
- Action C represents 'I challenge'
- The symbol W_i denotes that 'player i wins the game'
- The symbol U denotes that 'the corresponding player is undecided regarding the outcome of the game'

We note here that a local arena for a player is providing us with a possible deal for the

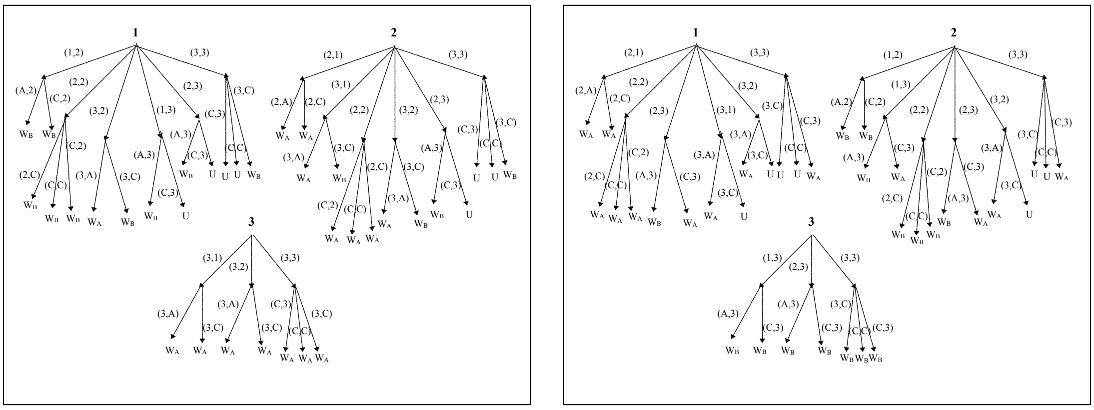


Figure 4.1 Local arenas for Alice (left) and Bob (right)

player in the game presented in Example 4.1. Let us consider the local arenas where Alice has received card number 2, and Bob card number 1.

If both the players announce 'I have card number 3', then according to the local arena of Alice, Bob will win if he challenges in the next round. However, if Alice challenges in the next round, she would be undecided regarding the outcome of the game until Bob shows his card number. The same indecisiveness would follow from Alice's perspective if they both challenge simultaneously. That is depicted in the local arena of Alice when she is having card number 2.

To reason about all these players together, and the corresponding global outcomes, that is, winning or losing of the concerned players after each possible play of the game, we introduce the notion of global arenas as follows.

Definition 4.2 (Global arena): Given a set of local arenas $\{\mathrm{G}^i\}_{i\in N}$, one for each player, the *global arena* $\mathrm{G} = (W, \rightarrow, w_0)$ is defined as follows: $W = W^1 \times \ldots \times W^n$, $w_0 = (w_0^1, \ldots, w_0^n)$, and $\rightarrow: W \times \tilde{\Gamma} \rightarrow W$ is a partial function satisfying the following condition: for all $w, v \in W$, we have $w \xrightarrow{\gamma} v$ iff

- for all $i \in enabled(w, \gamma)$, $w^i \xrightarrow{\gamma}_i v^i$,
- for all $i \in N \setminus enabled(w, \gamma)$, $v^i = w^i$,

with, $enabled(w, \gamma) = \{i \in N \mid \exists v^i \in W^i \text{ with } w^i \xrightarrow{\gamma}_i v^i\} \neq \varnothing$.

We note that such a simple notion of 'enabled moves' in this product based global arena may not always serve our purpose (cf. Section 4.3), but for now we keep our concepts simple and focus on what can be expressed and modeled by them. With respect to Example 4.1, a deal for both the players, and the possibilities therein can be modeled using this global arena. We note that the global arena gives us a view from top, that is, all the

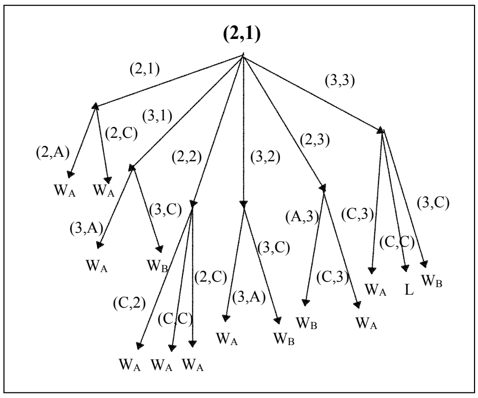


Figure 4.2 A global arena

indecisiveness among the players, where possible actions are taken by both the players, get resolved, and the outcomes are evaluated accordingly.

Example 4.3 (Global arena: Modeling games from a global joint perspective): The global arena where Alice has been given card 2 and Bob card 1 is depicted in Figure 4.2. We only show the relevant nodes in the game and leave out the isolated, irrelevant nodes. We note that the proposition U has been replaced by relevant global propositions in the context of the global arenas, where the top view tells us the cards that each player has, and the announcements are considered accordingly. When both of them say 'I have card number 3', suppose that Alice challenges in the next round. Even though in her local arena Alice would be unsure of her win, in the global arena, the global outcome is the 'win for Alice'. And, in case both Alice and Bob challenge, the global outcome is 'loss' for both the players, as they were both bluffing.

In what follows, we introduce a simple modal language to model reasoning in these games from both local and global perspectives, as illustrated by the examples given above. A main aim is to express Alice's inability to conclude about her win in her local arena when she has the card 2 and Bob has card 1, and they both bluff saying that they have card 3, whereas, under global reasoning, such a conclusion gets elucidated quite naturally.

4.2 Distributed game logic

In this section, we propose a modal logic, viz. Distributed Game Logic (DGL), to reason about such a combination of local and global reasoning in these arenas. We now present the syntax and semantics of this logic.

4.2.1 Syntax and semantics

The formulas of the logic are presented in two layers: *local formulas* describing the reasoning of the game from an individual perspective and *global outcome formulas* describing the outcomes of the game depending on the initial states. The local languages are indexed by players, one for each player in N. We have a modality referring to local moves, which are indexed by tuples of announcements. Let us fix P_i, a set of atomic propositions for each player i. The syntax of i-local formulas is given by:

Definition 4.3: The local language for player i, $\mathcal{L}_i$, is given as follows:

$$\alpha \in \mathcal{L}_i ::= p \mid \neg\alpha \mid \alpha \vee \alpha' \mid \langle\gamma\rangle\, \alpha,$$

where, $p \in \mathsf{P}_i$ and $\gamma \in \tilde{\Gamma}$.

Thus, the local formulas constitute the basic modal logic formulas for each player i. The model and truth definition of the local formulas are given as follows:

Definition 4.4: Given a player i, a model $M^i = (\mathrm{G}^i, V^i)$, where, G^i is a local arena for the player i and $V^i : W^i \to 2^{P_i}$ is the valuation function in the local arena. The truth definition of the formulas at a world in a model are defined inductively as follows:

- $M^i, w^i \models_i p$ iff $p \in V^i(w^i)$.
- $M^i, w^i \models_i \neg\alpha$ iff $M^i, w^i \not\models_i \alpha$
- $M^i, w^i \models_i \alpha \vee \beta$ iff $M^i, w^i \models_i \alpha$ or $M^i, w^i \models_i \beta$
- $M^i, w^i \models_i \langle\gamma\rangle\, \alpha$ if there exists $v^i \in W^i$ such that $w^i \xrightarrow{\gamma}_i v^i$ and $M^i, v^i \models_i \alpha$.

We say that an i-local formula α in $\mathcal{L}_i$ is i-satisfiable if there is a model M^i and a state w_i such that $M^i, w_i \models_i \alpha$. In what follows, all the formulas are evaluated at the root of the corresponding modes.

Example 4.4: In the game between Alice and Bob, as considered in Example 4.2, suppose that Alice gets card 2 and Bob gets card 1. Consider the local arena of Alice where she has card 2. Now, when they both say that they have card 3, they are both bluffing. But Alice cannot see Bob's card and so in Alice's local arena, using Alice's local language $\mathcal{L}_A$, we have that $[(3,3)]\langle(3,C)\rangle W_B$, and also, $[(3,3)]\langle(C,3)\rangle(W_A \vee W_B)$. Similarly, consider the local arena of Bob where he has card 1. Bob cannot see Alice's card and so in Bob's local arena, using Bob's local language $\mathcal{L}_B$, we have that $[(3,3)]\langle(C,3)\rangle W_A$, and also, $[(3,3)]\langle(3,C)\rangle(W_A \vee W_B)$.

On the other hand, suppose Alice announces truthfully that she has card 2, and Bob bluffs that he has card 3. Then, in Alice's local arena, she can either accept, thereby losing the game, or she can challenge in which case she will be undecided regarding the outcome of the game. We can express this reasoning of Alice in her local language as follows: $[(2,3)](\langle(A,3)\rangle W_B \wedge \langle(C,3)\rangle(W_A \vee W_B))$.

With local reasoning taken care of by the local languages, which are basically representing static modal environments with action tuples abstracting out the moves or announcements of the players, we are now all set to describe global reasoning. To this end, we define the global formulas below based on the local ones.

Definition 4.5: The global language $\mathcal{L}$ is given as follows:

$$\varphi \in \mathcal{L} ::= \alpha@i \mid \neg\varphi \mid \varphi \vee \varphi' \mid \langle\gamma\rangle\, \varphi$$

where $\alpha \in \mathcal{L}_i, \gamma \in \tilde{\Gamma}$.

As in the case of local formulas, the global formulas are also quite simple as basic modal logic, the only difference being that we use annotated local formulas as atomic global formulas. Intuitively, information from the local level can be directly perceived from a global perspective. To a certain extent, the structure of the game is analogous from both local and global perspectives, which makes the languages we use for local and global reasoning quite alike as well. As we notice in Example 4.5 below, to describe global reasoning at the outcome level, it is convenient to consider global outcome propositions at the terminal/leaf nodes. To keep things simple from a technical viewpoint (with respect to finding normal forms for global formulas), we do not include them in the current syntax. They can be evaluated at the leaf nodes without any constraint, as described in Section 4.5 which deals with frameworks we subsequently propose for modeling local and global strategic reasoning. The valuation function for these global propositions can be restricted to the leaf nodes which can be easily described both syntactically and semantically. For now, the semantics for our current global formulas is presented as follows:

Definition 4.6: A global model M is given by a global arena G, together with the local valuation functions. The truth definition of the global formulas at a world in a global model is given inductively as follows:

- $M, w \vDash \alpha@i$ if $M^i, w^i \vDash_i \alpha$
- $M, w \vDash \neg\varphi$ iff $M, w \nvDash \varphi$
- $M, w \vDash \varphi \vee \psi$ iff $M, w \vDash \varphi$ or $M, w \vDash \psi$
- $M, w \vDash \langle\gamma\rangle\, \varphi$ if there exists $v \in W$ such that $w \xrightarrow{\gamma} v$ and $M, v \vDash \varphi$.

We say that a formula φ is satisfiable if there is a global model M and a state $(w_1, ..., w_n)$ such that $M, (w_1, ..., w_n) \vDash \varphi$. A formula φ is said to be valid if $M, (w_1, ..., w_n) \vDash \varphi$ for any global model M and any state $(w_1, ..., w_n)$ in that model.

Example 4.5: Consider the global arena with Alice having card 2 and Bob having card 1. When they both bluff by saying that 'I have card 3', and a subsequent challenge by Alice, we would have $[(3,3)](\langle(C,3)\rangle(W_A \vee W_B)@A \wedge \langle(C,3)\rangle\mathbf{W@A})$ which represents Alice's local information and also, globally available information, with $\mathbf{W@A}$ denoting a 'win for Alice' being a global proposition. Similarly, $\mathbf{W@B}$ denoting a 'win for Bob', is a global proposition. Also, when both Alice and Bob challenge after the bluff, we would have $[(3,3)]\langle(C,C)\rangle\neg(\mathbf{W@A} \vee \mathbf{W@B})$, that is both would lose from the global point of view. Alternatively, when Alice announces truthfully that she has card 2 and Bob bluffs that he has card 3, globally we express Alice's and Bob's local information in a similar manner: $[(2,3)](\langle(A,3)\rangle(W_B@A \wedge W_B@B) \wedge \langle(C,3)\rangle((W_A \vee W_B)@A \wedge W_A@B))$. If Alice accepts, then both Alice's and Bob's local information include a win for Bob, and if Alice challenges, while Bob would have the information about a win for Alice, whereas Alice herself would not be sure of her win.

Above, we have talked about local and global information available to the players, solely depending on the basic modal logic environment where the labelled modal operators only correspond to the moves in a game. In particular, we did not even introduce any epistemic modal operator to deal with players' information levels. In process, we have proposed a rather descriptive logic talking about moves and outcomes of the games. So, what is essentially new about the logic we are proposing here? The answer to this question lies in what we are about to present below with the global axiom system highlighting a minimal interplay of local and global reasoning that happens in terms of modal reasoning in distributed games. We believe that this study would provide a base for richer studies on reasoning in such games, especially involving strategizing of players and information available to them. We provide an example in Section 4.5.

4.2.2 On axiomatization

We have proposed a modal logic to reason about distributed games in the section above. Given a new logical system, we generally formulate various model-theoretic and proof-theoretic queries regarding the logic, and focus on providing answers to those queries. In this section, we provide a strong completeness result.

4.2.2.1 Validities in DGL

We are now all set to provide a complete axiomatization for the validities of the logic. Let us first mention some validities of the logic, explicating the interplay between local and global reasoning in terms of, say, conjunction and modality.

- $(\alpha \wedge \beta)@i \equiv (\alpha@i \wedge \beta@i)$
- $[\gamma](\alpha@i) \supset ((\langle\gamma\rangle\top)@i \supset ([\gamma]\alpha)@i)$
- $[\gamma](\alpha@i) \supset (([\gamma]\bot)@i \supset \alpha@i)$

The proofs of these validities follow directly from the truth definition. We now provide a complete axiom system for the logic and follow it up by providing proofs for soundness and completeness of the axiom system. Before proceeding further, we should mention here that the first validity above with respect to conjunction and the related axioms $(B2)$ and $(B3)$, that we will see below with respect to Boolean connectives, show that the interplay is well-behaved in terms of propositional connectives. The final two validities, and the corresponding axioms $(B4)$ and $(B5)$ given below show the interplay in terms of modal operators characterizing the *enabled* property of the move relation in the global arena. Whenever there is a local move enabled ($(\langle\gamma\rangle\top)@i$), the local and global modalities basically tell the same story ($([\gamma]\alpha)@i \equiv [\gamma](\alpha@i)$), and for the players whose local moves are not enabled ($([\gamma]\bot)@i$), the local states remain the same ($[\gamma](\alpha@i) \equiv (\alpha@i)$).

4.2.2.2 Axiom system

We have an axiom system Ax_i for each player i in the system, and in addition a global axiom system AX. We use the notation $\vdash_i \alpha$ to mean that the formula $\alpha \in \mathcal{L}_i$ is a theorem of the system Ax_i. Similarly, $\vdash \phi$ means that ϕ is a theorem of the global system. We say a set of global formulas Δ is consistent if $\nvdash \bigwedge_{i\in S} \supset \bot$ for any finite $S \subseteq \Delta$, similarly, we mean a set of i-local formulas Δ_i is i-consistent if $\nvdash_i \bigwedge_{i\in S_i} \supset \bot$ for any finite set $S_i \subseteq \Delta_i$, where $i \in N$.

Ax_i, the axiom schemes for agent i

$(A0_i)$ Substitutional instances of propositional tautologies

$(A1_i)$ $[\gamma](\alpha \supset \beta) \supset ([\gamma]\alpha \supset [\gamma]\beta)$

Local inference rules

$$(MP_i)\ \frac{\alpha,\ \ \alpha \supset \beta}{\beta} \qquad (LG_i)\ \frac{\alpha}{[\gamma]\alpha}$$

Ax, the global axiom schemes

$(B0)$ Substitutional instances of propositional tautologies

$(B1)$ $[\gamma](\phi_1 \supset \phi_2) \supset ([\gamma]\phi_1 \supset [\gamma]\phi_2)$

$(B2)$ $(\neg\alpha)@i \equiv \neg(\alpha@i)$

$(B3)$ $(\alpha \vee \beta)@i \equiv (\alpha@i \vee \beta@i)$

$(B4)$ $(\langle\gamma\rangle\top)@i \supset (([\gamma]\alpha)@i \equiv [\gamma](\alpha@i))$

$(B5)$ $\alpha@i \supset (([\gamma]\bot)@i \supset [\gamma](\alpha@i))$

$(B6)$ $$\langle\gamma\rangle(\bigwedge_{i\in N}\alpha_i@i) \equiv \bigvee_{\varnothing\neq S\subseteq N}\Big(\bigwedge_{i\in S}((\langle\gamma\rangle\top)@i \wedge (\langle\gamma\rangle\alpha_i)@i) \wedge \bigwedge_{i\in N\setminus S}(([\gamma]\bot)@i \wedge \alpha_i@i)\Big)$$

$(B7)$ $(\bigvee_{i\in N}(\langle\gamma\rangle\top)@i \wedge [\gamma]\phi) \supset \langle\gamma\rangle\phi$

Global inference rules

$$(MP)\ \frac{\phi,\ \ \phi\supset\psi}{\psi} \qquad (G_i)\ \frac{\vdash_i \alpha}{\alpha@i} \qquad (GG)\ \frac{\phi}{[\gamma]\phi}$$

Let us first discuss the axioms in the list. The local axioms and the global axioms (B0) - (B3) are self-explanatory. The global axioms $(B4)$ - $(B7)$ spell out the interaction between local and global modalities. Axioms $(B4)$ and $(B5)$ have already been explained above. Axiom $(B7)$ deals with enabled local moves in the global arena, and finally, axiom $(B6)$ binds the local and global moves together, which in turn, helps us to get a normal form for the global formulas. The inference rules are also self-explanatory.

That the axioms and the rules are sound can be proved in the standard way, and we give a few of those proofs below, for axioms $(B2)$ – $(B6)$ and the rule (G_i). Let M be a global model with respect to the local models $\{M^i \mid i \in N\}$.

Proposition 4.1: Axiom $(B2)$ is valid.

Proof For any global model M, and any $w = (w^0, \ldots, w^n)$ in it. According to the local and global semantics, we have $M, w \vDash (\neg\alpha)@i$ iff $M^i, w^i \vDash_i \neg\alpha$ iff $M^i, w^i \nvDash_i \alpha$ iff $M, w \nvDash \alpha@i$ iff $M, w \vDash \neg(\alpha@i)$. hence, $M, w \vDash (\neg\alpha)@i \equiv \neg(\alpha@i)$. ∎

Proposition 4.2: Axiom $(B3)$ is valid.

Proof For any global model M, and any $w = (w^0, \ldots, w^n)$ in it. According to the local and global semantics, we have $M, w \vDash (\alpha \vee \beta)@i$ iff $M^i, w^i \vDash_i \alpha \vee \beta$ iff $M^i, w^i \vDash_i \alpha$ or $M^i, w^i \vDash_i \beta$ iff $M, w \vDash \alpha@i$ or $M, w \vDash \beta@i$ iff $M, w \vDash \alpha@i \vee \beta@i$ ∎

Proposition 4.3: Axiom $(B4)$ is valid.

Proof We prove the dual form of the proposition which reads $(\langle\gamma\rangle\top)@i \supset (\langle\gamma\rangle(\alpha@i) \equiv (\langle\gamma\rangle\alpha)@i)$. Suppose $M, w \vDash (\langle\gamma\rangle\top)@i \wedge \langle\gamma\rangle(\alpha@i)$. Since $M, w \vDash \langle\gamma\rangle(\alpha@i)$, then there exists v with $w \xrightarrow{\gamma} v$ such that $M, v \vDash \alpha@i$, it follows that $M^i, v^i \vDash_i \alpha$. Since $M, w \vDash (\langle\gamma\rangle\top)@i$, then $M^i, w^i \vDash_i \langle\gamma\rangle\top$, which means $i \in enabled(w,\gamma)$. According to $w \xrightarrow{\gamma} v$, we have $w^i \xrightarrow{\gamma}_i v^i$, then $M^i, w^i \vDash_i \langle\gamma\rangle\alpha$. Thus, $M, w \vDash (\langle\gamma\rangle\alpha)@i$.

Suppose $M, w \vDash (\langle\gamma\rangle\top)@i \wedge (\langle\gamma\rangle\alpha)@i$, then we have $M^i, w^i \vDash_i \langle\gamma\rangle\alpha$, thus there exists v_i with $w^i \xrightarrow{\gamma}_i v_i$ such that $M^i, v_i \vDash_i \alpha$. For $j \in enabled(w,\gamma) \setminus \{i\}$, there exits v_j with $w^j \xrightarrow{\gamma}_j v_j$. We construct $v = (v^1, \ldots, v^n)$ as follows:

$$v^j = \begin{cases} v_j & \textit{for } j \in enabled(w,\gamma), \\ w^j & \textit{for } j \in N \setminus enabled(w,\gamma). \end{cases}$$

It follows that $w \xrightarrow{\gamma} v$ by definition. Since $i \in enabled(w,\gamma)$ and $M^i, v_i \vDash_i \alpha$, then $M^i, v^i \vDash_i \alpha$, thus $M, v \vDash \alpha@i$, it follows $M, w \vDash \langle\gamma\rangle(\alpha@i)$. ∎

Proposition 4.4: Axiom $(B5)$ is valid.

Proof We also prove the dual form, i.e., $([\gamma]\bot)@i \supset (\langle\gamma\rangle(\alpha@i) \supset \alpha@i)$. Suppose $M, w \vDash ([\gamma]\bot)@i \wedge \langle\gamma\rangle(\alpha@i)$. Since $M, w \vDash \langle\gamma\rangle(\alpha@i)$, then there exists v with $w \xrightarrow{\gamma} v$ such that $M, v \vDash \alpha@i$, it follows that $M^i, v^i \vDash_i \alpha$. Since $M, w \vDash ([\gamma]\bot)@i$, then $M^i, w^i \vDash_i [\gamma]\bot$, which means $i \in N \setminus enabled(w,\gamma)$. According to $w \xrightarrow{\gamma} v$, we have $v^i = w^i$, then $M^i, w^i \vDash_i \alpha$. Thus, $M, w \vDash \alpha@i$. ∎

Proposition 4.5: Axiom $(B6)$ is valid.

Proof From left to right, suppose $M, w \vDash \langle\gamma\rangle(\bigwedge_{i\in N} \alpha_i@i)$, then there exists v with $w \xrightarrow{\gamma} v$ such that $M, v \vDash \bigwedge_{i\in N} \alpha_i@i$, thus $M^i, v^i \vDash_i \alpha_i$ for $i \in N$. According to $w \xrightarrow{\gamma} v$, there exists $S = \{i \in N \mid \exists v^i \in W^i \text{ with } w^i \xrightarrow{\gamma}_i v^i\} \neq \varnothing$ such that

- for all $i \in S$, $w^i \xrightarrow{\gamma}_i v^i$. Then $M^i, w^i \vDash_i \langle\gamma\rangle\alpha_i$, thus $M, w \vDash (\langle\gamma\rangle\alpha_i)@i$.
- for all $i \in N \setminus S$, $v^i = w^i$. Then $M^i, w^i \vDash_i \alpha_i$, thus $M, w \vDash \alpha_i@i$.

Moreover, since $S \neq \varnothing$, then

- for all $i \in S$, $M, w \vDash (\langle\gamma\rangle\top)@i$.
- for all $i \in N \setminus S$, $M, w \vDash ([\gamma]\bot)@i$.

It follows that $M, w \vDash \bigvee_{\varnothing\neq S\subseteq N} \bigwedge_{i\in S}(((\langle\gamma\rangle\top)@i \wedge (\langle\gamma\rangle\alpha_i)@i) \wedge \bigwedge_{i\in N\setminus S}(([\gamma]\bot)@i \wedge \alpha_i@i)$.

From right to left, suppose that we have $M, w \vDash \bigvee_{\varnothing \neq S \subseteq N} \bigwedge_{i \in S} (((\langle\gamma\rangle\top)@i \wedge (\langle\gamma\rangle\alpha_i)@i) \wedge \bigwedge_{i \in N \setminus S} (([\gamma]\bot)@i \wedge \alpha_i@i)$, then exists $S \neq \varnothing$ such that

- $M^i, w^i \vDash_i \langle\gamma\rangle\top \wedge \langle\gamma\rangle\alpha_i$ for $i \in S$,
- $M^i, w^i \vDash_i [\gamma]\bot \wedge \alpha_i$ for $i \in N \setminus S$.

Since $M^i, w^i \vDash_i \langle\gamma\rangle\top$ for $i \in S$ and $M^i, w^i \vDash_i [\gamma]\bot$ for $i \in N \setminus S$, we have that $S = enabled(w, \gamma) \neq \varnothing$. For $i \in S$, since $M^i, w^i \vDash_i \langle\gamma\rangle\alpha_i$, then there exits v_i with $w^i \xrightarrow{\gamma} v_i$ such that $M^i, v_i \vDash_i \alpha_i$ for $i \in S$. We construct $v = (v^1, \dots, v^n)$ as follows:

$$v^i = \begin{cases} v_i & for\ i \in enabled(w, \gamma), \\ w^i & for\ i \in N \setminus enabled(w, \gamma). \end{cases}$$

It follows that $w \xrightarrow{\gamma} v$ by definition. Since $M^i, v_i \vDash_i \alpha_i$ for $i \in S$, then $M^i, v^i \vDash_i \alpha_i$ for $i \in S$. Moreover, since $M^i, w^i \vDash_i [\gamma]\bot \wedge \alpha_i$ for $i \in N \setminus S$, then $M^i, v^i \vDash_i \alpha_i$ for $i \in N \setminus S$. Thus, $M^i, v^i \vDash_i \alpha_i$ for $i \in N$, then $M, v \vDash \alpha_i@i$ for $i \in N$, it follows $M, w \vDash \langle\gamma\rangle(\bigwedge_{i \in N} \alpha_i@i)$. ∎

Proposition 4.6: Axiom $(B7)$ is valid.

Proof Suppose that $M, w \vDash \bigvee_{i \in N} (\langle\gamma\rangle\top)@i \wedge [\gamma]\phi$, then we have that $M^i, w^i \vDash_i \langle\gamma\rangle\top$ for some $i \in N$, it follows that $w^i \xrightarrow{\gamma}_i v_i$ for some v_i in M^i. Additionally, for all $j \in enabled\{w\}$, without loss of generality, we assume that $w^i \xrightarrow{\gamma}_j v_j$ for v_j in M^j. Then we construct $v = (v^1, \dots, v^n)$ as follows.

$$v^i = \begin{cases} v_i & for\ i \in enabled(w, \gamma), \\ w^i & for\ i \in N \setminus enabled(w, \gamma). \end{cases}$$

It follows that $w \xrightarrow{\gamma} v$ by definition. Since $M, w \vDash [\gamma]\phi$, then $M, u \vDash \phi$ for any u with $w \xrightarrow{\gamma} u$, which means $M, v \vDash \phi$. Hence, we have that $M, w \vDash \langle\gamma\rangle\phi$. ∎

Proposition 4.7: Rule (G_i) is valid.

Proof Suppose that $M, w \nvDash \alpha@i$ for some global model M and w in this model, then $M, w \vDash \neg(\alpha@i)$, since we have proved that Axiom $(B2)$ is valid in Proposition 4.1, then $M, w \vDash (\neg\alpha)@i$, it follows that $M^i, w^i \vDash_i \neg\alpha$, contradiction. ∎

We finish this discussion by proving a theorem in DGL.

Proposition 4.8: $\bigwedge_i ((((\langle\gamma\rangle\top)@i \supset ([\gamma]\alpha)@i) \wedge (([\gamma]\bot)@i \supset \alpha@i)) \supset [\gamma] \bigwedge_i \alpha@i$ is a theorem of DGL.

Proof We prove this by applying the global axioms and propositional reasoning.

1. $\vdash ((\langle\gamma\rangle\top)@i \wedge ([\gamma]\alpha)@i) \supset [\gamma](\alpha@i)$ by axiom $(B4)$
2. $\vdash (([\gamma]\bot)@i \wedge (\alpha@i)) \supset [\gamma](\alpha@i)$ by axiom $(B5)$
3. $\vdash (((\langle\gamma\rangle\top)@i \wedge ([\gamma]\alpha)@i) \vee (([\gamma]\bot)@i \wedge \alpha@i)) \supset [\gamma](\alpha@i)$ by steps 1 and 2
4. $\vdash (((\langle\gamma\rangle\top)@i \supset ([\gamma]\alpha)@i) \wedge (([\gamma]\bot)@i \supset \alpha@i)) \supset [\gamma](\alpha@i)$, by step 3
5. $\vdash \bigwedge_i (((\langle\gamma\rangle\top)@i \supset ([\gamma]\alpha)@i) \wedge (([\gamma]\bot)@i \supset \alpha@i)) \supset \bigwedge_i [\gamma](\alpha@i)$ by step 4
6. $\vdash [\gamma] \bigwedge_i \alpha@i \equiv \bigwedge_i [\gamma](\alpha@i)$, from basic modal logic
7. $\vdash \bigwedge_i (((\langle\gamma\rangle\top)@i \supset ([\gamma]\alpha)@i) \wedge (([\gamma]\bot)@i \supset \alpha@i)) \supset [\gamma] \bigwedge_i \alpha@i$ by steps 5, 6

This completes the proof. ∎

4.2.2.3 Completeness proof

The local axiom system Ax_i for each i is the basic modal logic system corresponding to local arenas which can be represented as Kripke models. As a result, we can safely assume that Ax_i is sound and complete with respect to the i-local arenas for each i. To prove the completeness of the given global axiom system AX, we follow the usual procedure, that is, we show that any global consistent formula is satisfiable. However, while doing so, we provide a to and fro movement between global maximal consistent sets and tuples of i-local maximal consistent sets (cf. proposition 4.12), that is worth mentioning. We will always need these kinds of journeys back and forth while doing axiomatic studies of such local and global reasoning and proving the coherence of such journeys is the main novel aspect of this work. Our proof below yields a strong completeness result.

To begin the proof, let us consider φ to be a global consistent formula. Let A, B denote *global maximal consistent sets*, and L, L' denote *local maximal consistent sets* (for some i, determined by the context).

Definition 4.7 (i-local set): Let A be a set of global formulas. Define $(A)_i = \{\alpha \mid \alpha@i \in A\}$, the set of i-local formulas appearing in A.

We have the following result describing the interplay between local and global maximal consistent sets.

Proposition 4.9: If A is a global maximal consistent set then for each i, $(A)_i$ is a local (i-local) maximal consistent set.

Proof Let A be a global maximal consistent set. Take any $i \in N$. To show that $(A)_i$ is a local maximal consistent set. We suppose not, then $(A)_i$ is either inconsistent or not maximally consistent.

CASE 1: Suppose $(A)_i$ is inconsistent. Then there exist $\alpha_1, \alpha_2, \ldots, \alpha_m \in (A)_i$ such that:

1. $\vdash_i \alpha_1 \wedge \alpha_2 \wedge \ldots \wedge \alpha_m \supset \bot$
2. $\vdash_i \neg\alpha_1 \vee \neg\alpha_2 \vee \ldots \vee \neg\alpha_m$
3. $\vdash (\neg\alpha_1 \vee \neg\alpha_2 \vee \ldots \vee \neg\alpha_m)@i$ by rule (G_i)
4. $\vdash (\neg\alpha_1)@i \vee (\neg\alpha_2@i) \vee \ldots \vee (\neg\alpha_m)@i$ by axiom $(B3)$
5. $\vdash \neg(\alpha_1@i) \vee \neg(\alpha_2@i) \vee \ldots \vee \neg(\alpha_m@i)$ by axiom $(B2)$
6. $\vdash \alpha_1@i \wedge \alpha_2@i \wedge \ldots \wedge \alpha_m@i \supset \bot$

Thus A becomes inconsistent, a contradiction. So, $(A)_i$ is a local maximal consistent set.

CASE 2: Suppose $(A)_i$ is not a local maximal consistent. Then there is some α such that $\alpha \notin (A)_i$ and $\neg\alpha \notin (A)_i$. Then, $\alpha@i, \neg(\alpha@i) \notin A$, a contradiction.

This completes the proof. ∎

Let us now consider the move relations on the local and global maximal consistent sets. Let L, L' be i-local MCS's. $L \xrightarrow{\gamma}_i L'$ iff $\{\alpha \mid [\gamma]\alpha \in L\} \subseteq L'$. Consider the global MCS's A, B, with $A \xrightarrow{\gamma} B$ iff $\{\phi \mid [\gamma]\phi \in A\} \subseteq B$. We have the following result.

Proposition 4.10: Let A, B be global MCS's. If $A \xrightarrow{\gamma} B$ then the following holds:

- for all i such that $(\langle\gamma\rangle\top)@i \in A$, $(A)_i \xrightarrow{\gamma}_i (B)_i$, and
- for all j such that $([\gamma]\bot)@j \in A$, $(A)_j = (B)_j$.

Proof Let A, B be global MCS's. Suppose first that $A \xrightarrow{\gamma} B$. Consider $(A)_i$ for some i. Since, $(A)_i$ is an i-local MCS, either $\langle\gamma\rangle\top \in (A)_i$ or $[\gamma]\bot \in (A)_i$. So, either $(\langle\gamma\rangle\top)@i \in A$ or $([\gamma]\bot)@i \in A$. Let us now consider these two cases separately.

- Let $(\langle\gamma\rangle\top)@i \in A$. To show that $(A)_i \xrightarrow{\gamma}_i (B)_i$. Let $[\gamma]\alpha \in (A)_i$. To show $\alpha \in (B)_i$. Now, $(\langle\gamma\rangle\top)@i \in A$. Also, since $[\gamma]\alpha \in (A)_i$, $([\gamma]\alpha)@i \in A$. Then by $(B4)$, $[\gamma](\alpha@i) \in A$. Hence, $\alpha@i \in B$, and so, $\alpha \in (B)_i$.
- Let $([\gamma]\bot)@i \in A$. To show that $(A)_i = (B)_i$. Take any $\alpha \in (A)_i$. Now, since $([\gamma]\bot)@i \in A$, we have by $(B5)$, $[\gamma](\alpha@i) \in A$. Hence, $\alpha@i \in B$, and so, $\alpha \in (B)_i$. Thus, $(A)_i \subseteq (B)_i$, and since $(A)_i$ and $(B)_i$ are i-local MCS's, thus we have that $(A)_i = (B)_i$.

This completes the proof. ∎

To prove a converse result, that is, to have this interplay between local and global communication/actions in a better way, we would need to find a normal form of a global formula in terms of the constituent local formulas. To this end, we have the following result.

Proposition 4.11: Every global formula ϕ has an equivalent formula ϕ' of the form $\bigvee_{i\in K}\bigwedge_{j\in N}(\alpha^i_j@j)$, where K is a finite set of natural numbers, and N is the set of all players.[1]

Proof We prove this by induction on the size of the global formulas.

- ϕ is of the form $\alpha_k@k$: Then, $\phi \equiv \alpha_k@k \wedge \bigwedge_{i\neq k}\top@i$.
- ϕ is of the Boolean form ($\neg\psi$ or $\psi\vee\chi$): The result follows from the axioms $(B2)$, $(B3)$ and classical propositional logic reasoning.
- ϕ is of the form $\langle\gamma\rangle\psi$: By I.H. we can assume that ψ has an equivalent formula ψ' of the form $\bigvee_{i\in K'}\bigwedge_{j\in N}(\alpha^i_j@j)$. Now, $\psi_1\equiv\psi_2$ implies $\langle\gamma\rangle\psi_1\equiv\langle\gamma\rangle\psi_2$ (from axiom $(B1)$ and rule (GG)), it follows that

$$\langle\gamma\rangle\psi \equiv \langle\gamma\rangle\bigvee_{i\in K'}\bigwedge_{j\in N}(\alpha^i_j@j)$$

Since $\langle\gamma\rangle(\psi\vee\chi)\equiv\langle\gamma\rangle\psi\vee\langle\gamma\rangle\chi$, then we have

$$\langle\gamma\rangle\bigvee_{i\in K'}\bigwedge_{j\in N}(\alpha^i_j@j) \equiv \bigvee_{i\in K'}\langle\gamma\rangle\bigwedge_{j\in N}(\alpha^i_j@j)$$

Thus we have

$$\phi \equiv \bigvee_{i\in K'}\langle\gamma\rangle\bigwedge_{j\in N}(\alpha^i_j@j)$$

Hence, considering $\langle\gamma\rangle\bigwedge_{j\in N}(\alpha^i_j@j)$, by axiom $(B6)$, we have,

$$\phi \equiv \bigvee_{i\in K''}\bigwedge_{j\in N}(\beta^i_j@j)$$

This completes the proof. ∎

Before delving into more details of the completeness proof, let us first outline the overall proof strategy. We aim to show the consistent formula ϕ is satisfiable. We start by constructing i-local model with all i-local MCSs for all $i\in N$. In particular, we extend ϕ to a global MCS A and specify the induced i-local MCS $(A)_i$ as the initial state in i-

[1] Following the work of Fine (1975) and the technique developed here, we can get a similar result for the global language including global propositions.

local model for $i \in N$. Then we construct the global model with all local models we have using Definition 4.6. To demonstrate that such a model meets our requirements, we need to show the feasibility from a syntactic perspective, which involves proving Proposition 4.12. This proposition essentially encompasses the *existence lemma* we currently require. Finally, by utilizing the *truth lemma* at the local level, we can establish the *truth lemma* at the global level, i.e., Proposition 4.13. Thus we complete all the proofs. Let us now move on to define the local and global models in the following.

Definition 4.8: An i-local model M^i is a tuple $(W^i, \rightarrow_i, w_0^i, V^i)$, where,

- $W^i = \{L \mid L \text{ is an } i\text{-local MCS }\}$,
- for any $L, L' \in W^i$, $L \xrightarrow{\gamma}_i L'$ if $\{\alpha \mid [\gamma]\alpha \in L\} \subseteq L'$,
- $w_0^i = (A)_i$, such that $\phi \in A$, where A is a fixed MCS,
- $V^i(p) = \{L \mid p \in L\}$ for $p \in \mathsf{P}_i$.

Definition 4.9: Given a set of local models, $\{M^i\}_{i \in N}$, one for each player, the global model M is a tuple $(W, \rightarrow, w_0)$, where,

- $W = W^1 \times \ldots \times W^n$,
- $\rightarrow: W \times \tilde{\Gamma} \rightarrow W$ is defined by: for all $w, v \in W$, we have $w \xrightarrow{\gamma} v$ iff
  - for all $i \in enabled(w, \gamma)$, $w^i \xrightarrow{\gamma}_i v^i$,
  - for all $i \in N \setminus enabled(w, \gamma), v^i = w^i$,

  where, $enabled(w, \gamma) = \{i \in N \mid \exists v^i \in W^i \text{ with } w^i \xrightarrow{\gamma}_i v^i\}$,
- $w_0 = (w_0^1, \ldots, w_0^n)$, where w_0^i is an initial state in W^i,

With these canonical local and global models, we now prove the following result.

Proposition 4.12: The following statements hold:

1. For any global MCS A, $((A)_1, (A)_2, \ldots, (A)_n) \in W$.
2. For any global MCS's A, B, if $A \xrightarrow{\gamma} B$, then $((A)_1, \ldots (A)_n) \xrightarrow{\gamma} ((B)_1, \ldots (B)_n)$ in the global arena.
3. For any global MCS A, if $((A)_1, \ldots (A)_n) \xrightarrow{\gamma} (X_1, \ldots X_n)$ in the global arena, then there exists a global MCS B such that $A \xrightarrow{\gamma} B$ and for all i, $X_i = (B)_i$.

Proof The proofs are given in the following.

1. Follows from the definitions of local and global models and Proposition 4.9.
2. Let A, B be global MCS's such that $A \xrightarrow{\gamma} B$. Then Proposition 4.10 gives us that:
   - for all i such that $(\langle\gamma\rangle\top)@i \in A$, $(A)_i \xrightarrow{\gamma}_i (B)_i$, and
   - for all j such that $([\gamma]\bot)@j \in A$, $(A)_j = (B)_j$.

These exactly correspond to the following conditions of the move relation in the global model:

- for all $i \in enabled(((A)_1, \dots (A)_n), \gamma)$, $(A)_i \xrightarrow{\gamma}_i (B)_i$,
- for all $i \in N \setminus enabled(((A)_1, \dots (A)_n), \gamma), (A)_i = (B)_i$,

where, $enabled(((A)_1, \dots (A)_n), \gamma) = \{i \in N \mid \exists (B)_i \in W_i \text{ with } (A)_i \xrightarrow{\gamma}_i (B)_i\}$. Correspondingly, $enabled(((A)_1, \dots (A)_n), \gamma) = \{i \in N \mid (\langle\gamma\rangle\top)@i \in A\}$. So we have, $((A)_1, \dots (A)_n) \xrightarrow{\gamma} ((B)_1, \dots (B)_n)$ in the global arena. This completes the proof.

3. Take a global MCS A. Suppose that $((A)_1, \dots (A)_n) \xrightarrow{\gamma} (X_1, \dots X_n)$ in the global arena. To show that there exists a global MCS B such that $A \xrightarrow{\gamma} B$ and for all i, $X_i = (B)_i$. Let us first define the following: $\Delta_i := \{\alpha@i \mid [\gamma](\alpha@i) \in A\}$ for all i. We now show that:

$\Delta = \bigcup_i \{\Delta_i\}$ is a globally consistent set.

Take any finite subset Δ' of Δ. Let $\delta' = \bigwedge_i \delta@i$, where $\delta@i \in \Delta'$. Now, for these i's, either $\langle\gamma\rangle\top \in (A)_i$ or, $[\gamma]\bot \in (A)_i$. Let us consider the cases separately.

- Suppose $\langle\gamma\rangle\top \in (A)_i$. So, $(\langle\gamma\rangle\top)@i \in A$ and also, $[\gamma](\delta@i) \in A$. Then, by axiom $(B4)$, $([\gamma]\delta)@i \in A$.
- Suppose $[\gamma]\bot \in (A)_i$. So, $([\gamma]\bot)@i \in A$ and also, $[\gamma](\delta@i) \in A$. Now, we have $\bigvee_i (\langle\gamma\rangle\top)@i \in A$, since $((A)_1, \dots (A)_n) \xrightarrow{\gamma} (X_1, \dots X_n)$, $enabled(((A)_1, \dots (A)_n), \gamma)$ is non-empty. Then, by axioms $(B5)$, and $(B7)$, $\delta@i \in A$.

Then, by Proposition 4.8, $[\gamma]\delta' \in A$. Once again, by $(B7)$, $\langle\gamma\rangle\delta' \in A$. So, $\langle\gamma\rangle\delta'$ is consistent. Then, by GG, δ' is consistent. Since every finite subset of Δ is consistent, so is the set. Then, Δ can be extended to a global maximal consistent set, B, say. We now have to prove that $A \xrightarrow{\gamma} B$.

Case I. $[\gamma]\bot \in (A)_i$. Suppose $\alpha \in (A)_i$. Then $\alpha@i \in A$. Also, $([\gamma]\bot)@i \in A$. Then, by $(B5)$, $[\gamma](\alpha@i) \in A$. So, $\alpha@i \in B$ and hence, $\alpha \in (B)_i$. So, $(A)_i \subseteq (B)_i$, and hence, $(A)_i = (B)_i$.

Case II. $\langle\gamma\rangle\top \in (A)_i$. Suppose $[\gamma]\alpha \in (A)_i$. Then $([\gamma]\alpha)@i \in A$. Also, $(\langle\gamma\rangle\top)@i \in A$. Then, by $(B4)$, $[\gamma](\alpha@i) \in A$. So, $\alpha@i \in B$ and hence, $\alpha \in (B)_i$. So, $(A)_i \xrightarrow{\gamma} (B)_i$.

Thus, for each i, we have either $(A)_i = (B)_i$, or, $(A)_i \xrightarrow{\gamma} (B)_i$. Hence, for all i,

$X_i = (B)_i$. Finally, to show $A \xrightarrow{\gamma} B$, we have to show that $\{\langle\gamma\rangle\phi \mid \phi \in B\} \subseteq A$. By Proposition 4.11, it is enough to show that $\langle\gamma\rangle(\bigwedge_i \alpha_i@i) \in A$, whenever $\bigwedge_i(\alpha_i@i) \in B$. Suppose, $\bigwedge_i(\alpha_i@i) \in B$. Then, for all such i, $\alpha_i@i \in B$, and so $\alpha_i \in (B)_i$. Once again, we have two cases. For those i's, such that $[\gamma]\bot \in (A)_i$, $([\gamma]\bot)@i \in A$. Since $\alpha_i \in (B)_i$, and in this case, $(A)_i = (B)_i$, $\alpha \in (A)_i$. Then $\alpha@i \in A$. So, $([\gamma]\bot)@i \wedge \alpha_i@i \in A$. Then, for those i's, such that $\langle\gamma\rangle\top \in (A)_i$, $(\langle\gamma\rangle\top)@i \in A$. Since $\alpha_i \in B_i$, and $A_i \xrightarrow{\gamma}_i B_i$, then $\langle\gamma\rangle\alpha_i \in A_i$, thus $(\langle\gamma\rangle\alpha_i)@i \in A$. According to $(B4)$, we have $\langle\gamma\rangle(\alpha_i@i) \in A$. Then, for all these i's, using axiom $(B6)$, $\langle\gamma\rangle(\bigwedge_i \alpha_i@i) \in A$. This completes the proof.

All the cases of the proof are now complete. ∎

Now, we are ready to prove the *truth lemma*, which gives us the completeness result:

Proposition 4.13: For all MCS's A, for all global formulas ϕ,

$$\phi \in A \text{ iff } M, ((A)_1, \ldots, (A)_n) \vDash \phi$$

Proof This is by induction on ϕ, involving a subsidiary induction to show, for all i-local MCS's L and i-local α,

$$\alpha \in L \text{ iff } M^i, L \vDash_i \alpha \tag{4.1}$$

where M^i is the standard canonical model for multi-agent modal logic. The proof of (4.1) follows from basic modal logic (Blackburn et al., 2001). The crucial aspect is to establish a lemma that for any MCS's L and a formula in the form of $\langle\gamma\rangle\phi$ in L, there always exists a successor L' in M^i such that $\phi \in L'$.

Let us return to the main line of the proof. The induction steps are as follows.

1. $\phi = \alpha@i$: We have, $\phi \in A$ iff $\alpha@i \in A$ iff $\alpha \in (A)_i$ iff $M^i, (A)_i \vDash_i \alpha$ iff $M, ((A)_1, \ldots, (A)_n) \vDash \alpha@i$ iff $M, ((A)_1, \ldots, (A)_n) \vDash \phi$.
2. The boolean cases are as usual.
3. $\phi = \langle\gamma\rangle\psi$: We have, $\phi \in A$ iff $\langle\gamma\rangle\psi \in A$ iff there is an MCS B such that $A \xrightarrow{\gamma} B$ and $\psi \in B$ iff $((A)_1, \ldots (A)_n) \xrightarrow{\gamma} ((B)_1, \ldots (B)_n)$ and $M, ((B)_1, \ldots (B)_n) \vDash \psi$ (by Proposition 4.12 and induction hypothesis) iff $M, ((A)_1, \ldots, (A)_n) \vDash \langle\gamma\rangle\psi$ iff $M, ((A)_1, \ldots, (A)_n) \vDash \phi$.

This completes the proof. ∎

Above, we have proposed a two-layered modal logic, viz. DGL, for reasoning in distributed games from both local and global perspectives, and provided a complete Hilbert-style axiomatic proof system. With this base analysis settled, we now present an analogous, somewhat enhanced exploration of the global system towards a more realistic global model for distributed games, with the underlying logic framework remaining the same. As before, we provide a complete axiom system.

4.3 Distributed game logic with enabled actions

In the previous section, we presented a descriptive logic DGL, modeling local and global interaction. We expressed what a player can achieve and what actions lead to those outcomes. The actions considered are intuitively described as making announcements. Continuing our modeling at this level, we now propose a revised global arena that reflects a more sophisticated interaction from the local to the global levels, and conduct our logical analysis that brings out this interaction. The whole point is to show how the same two-layered language can be used to reason about richer global structures, where the changes in the interplay of local and global reasoning get reflected in the global axiom system, as earlier. To motivate our current study, we start with a more involved scenario of our running card game example as follows.

Example 4.6: As earlier, Alice and Bob play a card game with five cards this time, say, 1,2,3,4 and 5. They are dealt two cards each, and we suppose that Alice gets cards 1 and 4, and Bob gets cards 2 and 3, and they can only see their own cards. The rules of the game are as follows:

- Alice and Bob play in turns.
- They can choose to throw one card or jump (Q).
- If one player throws a card, then the other player has to throw a bigger card or jump.
- If one player chooses to jump, then the other player can only choose to throw a card.

The *winning condition* is given as follows:

- The player holding no card first wins the game.

Note that ‘throwing a card’ can be identified with ‘making an announcement about a card’ in terms of abstract moves. The game starts with one of the players, Alice say, throwing a card. We present the corresponding local arenas in figure 4.3. We note that these figures do not show all the moves explicitly in the players’ local arenas, they are basically used to illustrate certain instances of the game we are talking about.

We reiterate that the local arenas are game processes from the perspective of players playing the game, while the global arena is the game process from the perspective of

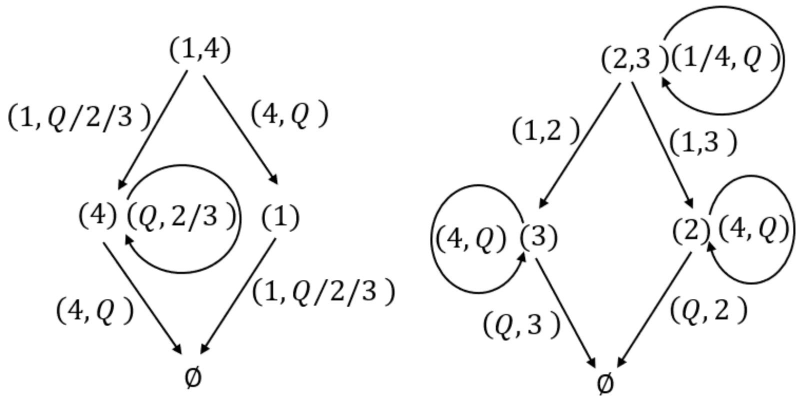


Figure 4.3 A local arena for Alice (left) and a local arena for Bob (right)

those watching the game from top, that is, those who can reason *about* the game. They can clearly identify the enabled moves at each game state and the outcomes of the game. In the following, we explain the symbols and notations used in figure 4.3:

- Each node in a local arena for player i is a state depicting the cards that player i holds at a certain stage of the game.
- The symbol $\varnothing$ is used to denote that player i holds no card.
- The action symbol Q is used to denote that player i jumps.
- Each move in the local arena represents a pair of actual moves in the game as follows: (Alice's move, Bob's subsequent move).

At the beginning of the game described above, Alice holds cards 1 and 4, while Bob holds cards 2 and 3. Suppose Alice plays card 4, Bob then chooses to jump, after that Alice only holds card 1 and Bob still holds cards 2 and 3. From the perspective of Alice, there is a game state transition from $(1,4)$ to (1) through the move $(4,Q)$, while for Bob, there is a state transition from $(2,3)$ to $(2,3)$ through the $(4,Q)$.

If we consider the game as a whole without looking into the individual moves of the players, we can consider a state transition from $((1,4),(2,3))$ to $((1),(2,3))$ through the pair of moves, $(4,Q)$, and that is what we envisage our global arena to deal with. However, if we focus on the game state $((1),(2,3))$ in the global arena, according to Definition 4.2, then we have a transition from the game state $((1),(2,3))$ to $((1),(2,3))$ through the action pair $(4,Q)$, which does not make sense. For Alice, at her local state (1), she cannot play the card 4 again.

In what follows, we propose a method for eliminating such redundant state transitions. Consider the set of enabled actions for Alice at her local state (1), $enabled_A((1)) = \{1\}$, and that for Bob at his local state (2,3), $enabled_B((2,3)) = \{2,3,Q\}$, and we ask that at the global game state $((1),(2,3))$, for any enabled pair of moves of players, say γ, the first component of γ, say $\gamma[Alice] \in enabled_A((1))$ and the second component of γ, say, $\gamma[Bob] \in enabled_B((2,3))$. The global arena is now defined accordingly by taking the sets of *enabled* moves under consideration. In the global arena depicted by figure 4.4, we only collect the game states starting from $((1,4),(2,3))$ for the sake of brevity.

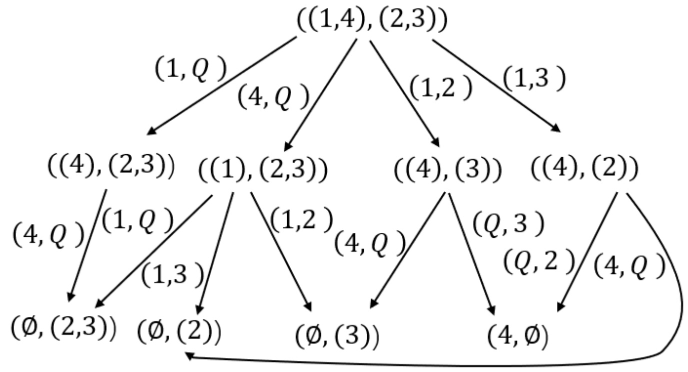


Figure 4.4 A global arena

As exemplified in the aforementioned instance, the local arenas exhibit precision in depicting the game from an individual perspective, whereas the existing concept of global arena manifests inadequacy in articulating the game above. To reflect the interplay between local and global arenas, as shown in Example 4.6, we hereby introduce a new description of global arenas. For the sake of completeness, we define both local arena (as earlier) and global arena, in this case, with a more involved notion of enabled actions.

Definition 4.10 (Local arena): A *local arena* for player i is a tuple $\mathrm{G}^i = (W^i, \to_i, w_0^i)$ such that W^i is the set of local game states, w_0^i is the initial game state, and $\to_i$ is a partial move function given by, $\to_i$: $W^i \times \tilde{\Gamma} \to W^i$.

Definition 4.11 (Global arena with enabled actions): Given a set of local arenas $\{\mathrm{G}^i\}_{i\in N}$, one for each player, *the global arena with enabled actions* G with respect to $\{\mathrm{G}^i\}_{i\in N}$ is a tuple $(W, \to, w_0)$, where

- W is the smallest subset of $\prod_{i=0}^{n} W^i$ under the relation $\to$.
- $w_0 = (w_0^1, \dots, w_0^n)$, and w_0 belongs to W, which is called the initial game state.
- $\to \subseteq W \times \tilde{\Gamma} \times W$, and $w \xrightarrow{\gamma} v$ (*denoting* $(w,\gamma,v) \in \to$) iff
  - $(b1)$ $enabled(w,\gamma) \neq \varnothing$.
  - $(b2)$ for all $i \in enabled(w,\gamma)$, $w^i \xrightarrow{\gamma}_i v^i$.
  - $(b3)$ for all $i \in N \setminus enabled(w,\gamma), v^i = w^i$ and $\gamma[i] \in enabled_i(w^i)$.

  where, $enabled(w,\gamma) = \{i \in N \mid \exists u \in W^i$ with $w^i \xrightarrow{\gamma}_i u\}$,

$$enabled_i(w^i) = \{\gamma[i] \mid \exists u \in W^i \text{ and } \gamma \in \tilde{\Gamma} \text{ with } w^i \xrightarrow{\gamma}_i u\}.$$

We emphasize that the description of the local arena remains unaltered from that presented in Section 4.1. However, we have updated the definition of the global arena from that in Section 4.1 to incorporate a more elaborate notion of *enabled actions*. Specifically, our approach involves focusing solely on those game states that are expected to materialize during the game, while excluding those that are deemed irrelevant. Additionally, we have introduced the condition '$\gamma[i] \in enabled_i(w^i)$' to ensure that game state transitions are executed more realistically in the given class of games.

Drawing on the aforementioned local arenas and global arenas with enabled actions, we propose a framework, namely, distributed game logic with enabled actions (DGLEA) to describe local and global reasoning in the current setting of distributed games. As previously discussed, we have updated the global arena with available actions to capture certain intricacies of the notion of distributed games. With respect to the logical framework, the syntax and semantics remain unaltered at the local level. This is because of the fact that at the local level, we only need the basic modal language to characterize the structure of the game. The main modification lies in the way to use the local descriptions of the game to show that a more reasonable model of the game is reflected at the global level. Note however, that our game graph from a global perspective is still a labelled transition system, and the original syntactic definitions at the global level remain effective for our current purposes. Hence at the global level, despite the shift in the class of models we intend to depict, the language remains the same. As we have been mentioning all through this chapter, our main target is not to introduce a sophisticated or rich language framework, but to focus on a simple basic modal logic syntax, and show the extent of interaction between local and global reasoning we can express at this level. Once again, for the sake of completion, we provide below the syntax and semantics of DGLEA at both the local and the global levels.

4.3.1 Syntax and semantics

The syntax and semantics of DGLEA in the local layer is the same as those of DGL. Let P_i be a set of atomic propositions for each player i:

Definition 4.12: The local language for player i, $\mathcal{L}_i$, is given as follows:

$$\alpha \in \mathcal{L}_i ::= p \mid \neg\alpha \mid \alpha \vee \alpha' \mid \langle\gamma\rangle\, \alpha,$$

where, $p \in \mathsf{P}_i$ and $\gamma \in \tilde{\Gamma}$.

Definition 4.13: Given a player i, a model $M^i = (\mathrm{G}^i, V^i)$, where, G^i is a local arena for the player i and $V^i : W^i \to 2^{P_i}$ is the valuation function in the local arena. The truth definition of the formulas at a world in a model are defined inductively as follows:

- $M^i, w^i \models_i p$ iff $p \in V^i(w^i)$.
- $M^i, w^i \models_i \neg\alpha$ iff $M^i, w^i \not\models_i \alpha$
- $M^i, w^i \models_i \alpha \vee \beta$ iff $M^i, w^i \models_i \alpha$ or $M^i, w^i \models_i \beta$
- $M^i, w^i \models_i \langle\gamma\rangle\, \alpha$ if there exists $v^i \in W^i$ such that $w^i \xrightarrow{\gamma}_i v^i$ and $M^i, v^i \models_i \alpha$.

Similarly, we say that an i-local formula α is i-satisfiable if there is an i-local model M^i and a state w^i such that $M^i, w^i \models_i \alpha$. For the global layer, we still use the same syntax as that of DGL.

Definition 4.14: The global language $\mathcal{L}$ is given as follows:

$$\varphi \in \mathcal{L} ::= \alpha@i \mid \neg\varphi \mid \varphi \vee \varphi' \mid \langle\gamma\rangle\, \varphi$$

where $\alpha \in \mathcal{L}_i, \gamma \in \tilde{\Gamma}$.

The semantics of global formulas is presented as follows:

Definition 4.15: A *global model with enabled actions* M is given by a global arena with enabled actions G and a collection of local valuation functions. The truth definition of the global formulas at a world in a global model is given inductively as follows:

- $M, w \models \alpha@i$ if $M^i, w^i \models_i \alpha$
- $M, w \models \neg\varphi$ iff $M, w \not\models \varphi$
- $M, w \models \varphi \vee \psi$ iff $M, w \models \varphi$ or $M, w \models \psi$
- $M, w \models \langle\gamma\rangle\, \varphi$ if there exists $v \in W$ such that $w \xrightarrow{\gamma} v$ and $M, v \models \varphi$.

Similarly, we say that a formula φ is said to be D-satisfiable if there is a global model with enabled actions M and a state w such that $M, w \models \varphi$. A formula φ is said to be valid if $M, w \models \varphi$ for any M and any state w in this model.

4.3.2 On axiomatization

DGLEA is a two-level logic akin to DGL, sharing similar characteristics at the local level while differing at the global level, and we present an axiom system below which will explicate these differences in terms of the available features on the reasoning involved. We have an axiom system Ax_i for each player i in the system, and in addition a global axiom system Ax^*. We use the notation $\vdash_i \alpha$ to mean that the formula $\alpha \in \mathcal{L}_i$ is a theorem of the system Ax_i. Similarly, $\vdash_D \phi$ means that ϕ is a theorem of the global system Ax^*. We say that a set of global formulas Δ is consistent if $\nvdash_D \bigwedge_{i \in S} \supset \bot$ for any finite $S \subseteq \Delta$, and similarly, we mean that a set of i-local formulas Δ_i is i-consistent if $\nvdash_i \bigwedge_{i \in S_i} \supset \bot$ for any finite set $S_i \subseteq \Delta_i$, where $i \in N$.

As a first point of distinction regarding what we intend to do with the framework of DGLEA, we introduce the notation $j \rightsquigarrow_i \delta$ to denote $\bigvee\limits_{\gamma:\gamma[j]=\delta} \langle\gamma\rangle\top$, where $\langle\gamma\rangle$ is a modality in the local language $\mathcal{L}_i$. It can be read as 'player i observes that there is an available announcement δ for player j in his local arena'. Intuitively, $\bigwedge\limits_{j \in N} (j \rightsquigarrow_i \delta_j)$ means 'player i observes that there is an available announcement δ_j for each player j', where $i \in N$. This formula plays an important role in the axiomatization below bringing out the modification we made in defining global arenas for DGLEA.

Ax_i, the axiom schemes for agent i

$(A0_i)$ Substitutional instances of propositional tautologies

$(A1_i)$ $[\gamma](\alpha \supset \beta) \supset ([\gamma]\alpha \supset [\gamma]\beta)$

Local inference rules

$$(MP_i)\ \frac{\alpha,\ \ \alpha \supset \beta}{\beta} \qquad (LG_i)\ \frac{\alpha}{[\gamma]\alpha}$$

Ax*, the global axiom schemes

$(B0)$ Substitutional instances of propositional tautologies

$(B1)$ $[\gamma](\phi_1 \supset \phi_2) \supset ([\gamma]\phi_1 \supset [\gamma]\phi_2)$

$(B2)$ $(\neg\alpha)@i \equiv \neg\alpha@i$

$(B3)$ $(\alpha \vee \beta)@i \equiv (\alpha@i \vee \beta@i)$

$(B5)$ $([\gamma]\bot)@i \wedge \langle\gamma\rangle\,(\alpha@i) \supset \alpha@i$

$(C0)$ $(\langle\gamma\rangle\,\alpha)@i \wedge \bigwedge_{j \neq i}(j \rightsquigarrow_j \gamma[j])@j \supset \langle\gamma\rangle\,(\alpha@i)$

$(C1)$ $\langle\gamma\rangle\top \equiv \bigvee_{i \in N}((\langle\gamma\rangle\top)@i) \wedge \bigwedge_{j \in N}(j \rightsquigarrow_j \gamma[j])@j$

$(C2)$ $(\langle\gamma\rangle\top)@i \wedge \langle\gamma\rangle\,(\alpha@i) \supset (\langle\gamma\rangle\,\alpha)@i$

$(C3)$ $\langle\gamma\rangle \bigwedge_{i \in S}(\alpha_i@i) \equiv \bigwedge_{i \in S}\langle\gamma\rangle\,(\alpha_i@i)$, *where* $S \subseteq N$, *and* $S \neq \varnothing$

Global inference rules

$$(MP)\ \frac{\phi,\ \ \phi \supset \psi}{\psi} \qquad (GG)\ \frac{\phi}{[\gamma]\phi} \qquad (G_i)\ \frac{\vdash_i \alpha}{\alpha@i}$$

In the above axiom system, the global axioms $(B5)$ and $(C0)$ – $(C3)$ spell out the interaction between local and global modalities. The axioms $(B5)$ and $(C2)$ can be explained as in the case for DGL. Axiom $(C0)$, the main distinctive feature of this axiom system, captures the condition $(b3)$ in Definition 4.11, while axiom $(C3)$ reflects a particular principle of our models, the order of the operators $\wedge$ and $\langle\gamma\rangle$ can be exchanged, from a technical point of view, which does not hold in multi-agent modal logic. Axiom $(C1)$ also corresponds to the (new) conditions provided in the global arena, and is used towards getting the normal form for global formulas in DGLEA, along with the others.

As an illustration of how one can work with this axiom system, we derive a theorem in the above proof system. This theorem also helps us in getting the normal form, and consequently, is used in our proof for the completeness result later.

Proposition 4.14: $\langle\gamma\rangle(\alpha@i) \equiv (((\langle\gamma\rangle\top)@i \supset ((\langle\gamma\rangle\alpha)@i \wedge \bigwedge_{j\neq i}(j \rightsquigarrow_j \gamma[j])@j))$ $\wedge\ (([\gamma]\bot)@i \supset \alpha@i) \wedge \langle\gamma\rangle\top$ is a theorem of DGLEA, denoted by $(D0)$.

Proof From left to right,

1. $\vdash_D \langle\gamma\rangle(\alpha@i) \supset ((\langle\gamma\rangle\top)@i \supset (\langle\gamma\rangle\alpha)@i)$ (from axiom $(C2)$)
2. $\vdash_D \langle\gamma\rangle(\alpha@i) \supset \langle\gamma\rangle\top$ (from axiom $(B1)$ and rule (MP))
3. $\vdash_D \langle\gamma\rangle\top \supset \bigwedge_{j\in N}(j \rightsquigarrow_j \gamma[j])@j$ (from axiom $(C1)$)
4. $\vdash_D \langle\gamma\rangle(\alpha@i) \supset \bigwedge_{j\in N}(j \rightsquigarrow_j \gamma[j])@j$ (from $2,3$ and rule (MP))
5. $\vdash_D \langle\gamma\rangle(\alpha@i) \supset ((\langle\gamma\rangle\top)@i \supset ((\langle\gamma\rangle\alpha)@i \wedge \bigwedge_{j\in N}(j \rightsquigarrow_j \gamma[j])@j))$(from $1,4$)
6. $\vdash_D \langle\gamma\rangle(\alpha@i) \supset (([\gamma]\bot)i \supset \alpha@i)$ (from axiom $(B5)$)
7. $\vdash_D \langle\gamma\rangle(\alpha@i) \supset ((\langle\gamma\rangle\top)@i \supset ((\langle\gamma\rangle\alpha)@i \wedge \bigwedge_{j\neq i}(j \rightsquigarrow_j \gamma[j])@j))$ (from $2,5,6$)

From right to left, we let ϕ be $(\langle\gamma\rangle\top)@i$, η be $(\langle\gamma\rangle\alpha)@i \wedge \bigwedge_{j\neq i}(j \rightsquigarrow_j \gamma[j])@j)$, φ be $\alpha@i$, ξ be $\langle\gamma\rangle\top$, β be $\langle\gamma\rangle(\alpha@i)$, and θ be $[\gamma](\alpha@i)$ for convenience.

1. $\vdash_D ((\phi \supset \eta) \wedge (\neg\phi \supset \varphi) \wedge \xi) \supset ((\eta \wedge \xi) \vee (\neg\phi \wedge \varphi \wedge \xi))$ (Propositional logic)
2. $\vdash_D \eta \supset \beta$ (from axiom $(C0)$)
3. $\vdash_D \neg\phi \wedge \varphi \supset \theta$ (from axiom $(B5)$)
4. $\vdash_D \theta \wedge \xi \supset \beta$ (using theorems in modal logic)
5. $\vdash_D \neg\phi \wedge \varphi \wedge \xi \supset \beta$ (from 3 and 4)
6. $\vdash_D ((\eta \wedge \xi) \vee (\neg\phi \wedge \varphi \wedge \xi)) \supset \beta$ (from 2 and 5)
7. $\vdash_D ((\phi \supset \eta) \wedge (\neg\phi \supset \varphi) \wedge \xi) \supset \beta$ (from $1,6$ and rule (MP))

This completes the proof. ∎

That the axioms and the rules are sound can be proved in the standard way, and we only concentrate on the proofs for the axioms $(C0)-(C3)$. Let M be a global model with enabled actions with respect to local models $\{M^i\}_{i\in N}$.

Proposition 4.15: Axiom $(C0)$ is valid.

Proof Suppose $M, w \vDash (\langle\gamma\rangle\,\alpha)@i$, which means $M^i, w^i \vDash_i \langle\gamma\rangle\,\alpha$. It follows that there exists v_i with $w^i \xrightarrow{\gamma} v_i$ such that $M^i, v_i \vDash_i \alpha$. Since $w^i \xrightarrow{\gamma}_i v_i$, then $i \in enabled(w,\gamma) \neq \emptyset$. We construct $v = (v^1, \dots v^n)$ as follows:

$$v^j = \begin{cases} v_j & for\ j \in enabled(w,\gamma), \\ w^j & for\ j \in N \setminus enabled(w,\gamma). \end{cases}$$

Since $M, w \vDash \bigwedge_{j\neq i}(j \rightsquigarrow_j \gamma[j])@j$, then we have $M^j, w^j \vDash_j j \rightsquigarrow_j \gamma[j]$ for $j \neq i$, which means $\gamma[j] \in enabled_j(w^j)$ for $j \in N \setminus enabled(w,\gamma)$. It follows that $w \xrightarrow{\gamma} v$, and $M, v \vDash \alpha@i$. Thus, $M, w \vDash \langle\gamma\rangle\,(\alpha@i)$. ∎

Proposition 4.16: Axiom $(C1)$ is valid.

Proof This axiom corresponds to the relation part in Definition 4.11. The left side signifies the existence of state transition relations in the model, while the right side corresponds to conditions $(b1)-(b3)$. In formal, for any w in M,

From left to right, Suppose $M, w \vDash \langle\gamma\rangle\top$, then there is v such that $w \xrightarrow{\gamma} v$. By Definition 4.11, we have $(b1) : enabled(w,\gamma) \neq \emptyset$, which means $M, w \vDash \bigvee_{i\in N}((\langle\gamma\rangle\top)@i)$. To show $M, w \vDash (j \rightsquigarrow_j \gamma[j])@j$ for $j \in N$, we consider two cases.

Case 1: $j \in enabled(w,\gamma)$, which means $M^j, w^j \vDash_j \langle\gamma\rangle\top$ by (b2), it follows $M, w \vDash (j \rightsquigarrow_j \gamma[j])@j$ for $j \in enabled(w,\gamma)$.

Case 2: $j \in N \setminus enabled(w,\gamma)$, by (b3), there exists $\gamma' \in \tilde{\Gamma}$ and u in M^j such that $\gamma'[j] = \gamma[j]$ and $w^j \xrightarrow{\gamma'}_j u$, then it follows that $M, w \vDash (j \rightsquigarrow_j \gamma[j])@j$ for $j \in N \setminus enabled(w,\gamma)$.

Hence, we have that $M, w \vDash (j \rightsquigarrow_j \gamma[j])@j$ for $j \in N$.

From right to left, suppose that $M, w \vDash \bigvee_{i\in N}((\langle\gamma\rangle\top)@i) \wedge \bigwedge_{j\in N}(j \rightsquigarrow_j \gamma[j])@j$, we have to construct v such that $w \xrightarrow{\gamma} v$.

Since $M, w \vDash \bigvee_{i\in N}((\langle\gamma\rangle\top)@i)$, then we have $(b1) : enabled(w,\gamma) \neq \emptyset$. We consider two cases.

Case 1: $j \in enabled(w,\gamma)$, there is v_j in M^j such that $w^j \xrightarrow{\gamma}_j v_j$.

Case 2: $j \in N \setminus enabled(w, \gamma)$, since $M, w \models (j \rightsquigarrow_j \gamma[j])@j$ for $j \in N$, then exists $\gamma' \in \tilde{\Gamma}$ and v_j in M^j such that $\gamma'[j] = \gamma[j]$ and $w^j \xrightarrow{\gamma'}_j v_j$. It follows that $\gamma[j] \in enabled_j(w^j)$.

We construct $v = (v^1, \dots v^n)$ as follows:

$$v^j = \begin{cases} v_j & for\ j \in enabled(w, \gamma), \\ w^j & for\ j \in N \setminus enabled(w, \gamma). \end{cases}$$

According to Definition 4.11, we have $w \xrightarrow{\gamma} v$. ∎

Proposition 4.17: Axiom $(C2)$ is valid.

Proof Suppose $M, w \models (\langle\gamma\rangle \top)@i \wedge \langle\gamma\rangle (\alpha@i)$. Since $M, w \models \langle\gamma\rangle (\alpha@i)$, then there exists v with $w \xrightarrow{\gamma} v$ such that $M, v \models \alpha@i$, it follows that $M^i, v^i \models_i \alpha$. Since $M, w \models (\langle\gamma\rangle \top)@i$, then $M^i, w^i \models_i \langle\gamma\rangle \top$, which means $i \in enabled(w, \gamma)$. According to Definition 4.11, we have $w^i \xrightarrow{\gamma}_i v^i$, then $M^i, w^i \models_i \langle\gamma\rangle\alpha$. Thus, $M, w \models (\langle\gamma\rangle\alpha)@i$. ∎

Proposition 4.18: Axiom $(C3)$ is valid.

Proof From left to right, we can prove it in basic modal logic, then we focus on the other direction. Suppose $M, w \models \bigwedge_{i \in S} \langle\gamma\rangle (\alpha_i@i)$, then $M, w \models \langle\gamma\rangle (\alpha_i@i)$ for $i \in S \subseteq N$. It follows that there is (vi) with $w \xrightarrow{\gamma} (vi)$ such that $M, (vi) \models \alpha_i@i$ for $i \in S$, then $M^i, (vi)^i \models_i \alpha_i$ for $i \in S$. We construct v as follows:

$$v^i = \begin{cases} (vi)^i & for\ i \in S \subseteq N, \\ (vk)^i & for\ i \in N - S,\ where\ k\ is\ a\ fixed\ number\ in\ S. \end{cases}$$

Then we have $M^i, v^i \models_i \alpha_i$ for $i \in S$, thus $M, v \models \bigwedge_{i \in S}(\alpha_i@i)$. We claim $w \xrightarrow{\gamma} v$ as follows:

- $enabled(w, \gamma) \neq \emptyset$ by $w \xrightarrow{\gamma} (vi)$ for $i \in S$,
- for all $i \in enabled(w, \gamma)$,
  - for $i \in S$, since $v^i = (vi)^i$ and $w^i \xrightarrow{\gamma}_i (vi)^i$ by $w \xrightarrow{\gamma} (vi)$, then $w^i \xrightarrow{\gamma}_i v^i$,
  - for $i \notin S$, since $v^i = (vk)^i$ and $w^i \xrightarrow{\gamma}_i (vk)^i$ by $w \xrightarrow{\gamma} (vk)$, then $w^i \xrightarrow{\gamma}_i v^i$,
- for all $i \in N \setminus enabled(w, \gamma)$,
  - for $i \in S$, since $v^i = (vi)^i$ and $(vi)^i = w^i$ by $w \xrightarrow{\gamma} (vi)$, then $v^i = w^i$, and $\gamma[i] \in enabled_i(w^i)$ by $w \xrightarrow{\gamma} (vi)$,
  - for $i \notin S$, since $v^i = (vk)^i$ and $(vk)^i = w^i$ by $w \xrightarrow{\gamma} (vk)$, then $v^i = w^i$, and $\gamma[i] \in enabled_i(w^i)$ by $w \xrightarrow{\gamma} (vk)$.

It follows that $M, w \models \langle\gamma\rangle \bigwedge_{i \in S}(\alpha_i@i)$. ∎

Similar to DGL, we have the following lemma.

Lemma 4.1: Every global formula α has an equivalent formula α' of the form $\bigvee_{i\in K}\bigwedge_{j\in S_i}(\alpha^i_j@j)$, where K is a finite set of natural numbers and $S_i \subseteq N$.

Proof We prove by induction on the structure of α.

- α is of the form $\beta_i@i$, it is trivial.
- α is of the form $\neg\beta$, or $\phi\vee\psi$, then easily we can find an equivalent formula α' of the form $\bigvee_{i\in K}\bigwedge_{j\in S_i}(\alpha^i_j@j)$ for α, as we obtain an equivalent one in DNF for any boolean formula in Propositional logic.
- α is of the form $\langle\gamma\rangle\beta$, suppose β has an equivalent formula β' of the form $\bigvee_{i\in K'}\bigwedge_{j\in S'_i}(\alpha^i_j@j)$ by I.H. without loss of generality. Since $\psi_1\equiv\psi_2$ implies $\langle\gamma\rangle\psi_1\equiv\langle\gamma\rangle\psi_2$ (from axiom $(B1)$ and rule (MP)), it follows that

  $$\langle\gamma\rangle\beta\equiv\langle\gamma\rangle\bigvee_{i\in K'}\bigwedge_{j\in S'_i}(\alpha^i_j@j)$$

  Since $\langle\gamma\rangle(\alpha\vee\beta)\equiv\langle\gamma\rangle\alpha\vee\langle\gamma\rangle\beta$, then we have

  $$\langle\gamma\rangle\bigvee_{i\in K'}\bigwedge_{j\in S'_i}(\alpha^i_j@j)\equiv\bigvee_{i\in K'}\langle\gamma\rangle\bigwedge_{j\in S'_i}(\alpha^i_j@j)$$

  Thus we have

  $$\alpha\equiv\bigvee_{i\in K'}\langle\gamma\rangle\bigwedge_{j\in S'_i}(\alpha^i_j@j)$$

  According to axiom $(C3)$, we have

  $$\alpha\equiv\bigvee_{i\in K'}\bigwedge_{j\in S'_i}\langle\gamma\rangle(\alpha^i_j@j)$$

  According to axiom $(C1)$ and theorem $(D0)$, we have

  $\vdash\langle\gamma\rangle(\alpha^i_j@i)\equiv(((\langle\gamma\rangle_j\top_j)@j\supset((\langle\gamma\rangle_j\alpha^i_j)@j\wedge\bigwedge_{m\neq j}(m\rightsquigarrow_m\gamma[m])@m))$
  $\wedge(([\gamma]_j\bot_j)@j\supset\alpha^i_j@j)\wedge(\bigvee_{j\in N}((\langle\gamma\rangle_j\top_j)@j)\wedge\bigwedge_{j\in N}(m\rightsquigarrow_m\gamma[m])@m)$

  Note that the right side of the theorem above is boolean combinations of global proposition letters. It follows that we can obtain an equivalent formula to $\bigvee_{i\in K'}\bigwedge_{j\in S'_i}\langle\gamma\rangle(\alpha^i_j@j)$, and it has no global modality. After the reorganization as we deal with boolean formula in Propositional logic to obtain an equvialent formula in DNF , we can give an equvialent formula to α', which is of the form $\bigvee_{i\in K}\bigwedge_{j\in S_i}(\alpha^i_j@j)$.

This completes the proof. ∎

The above theorem plays a crucial role in our completeness proof. To prove the strong completeness result for DGLEA, we can follow the same method used for proving the strong completeness of DGL. In the following, we take an alternative approach to present a weak completeness result.

Theorem 4.1: Ax^* is complete, i.e., every consistent global formula is D-satisfiable.

Proof Given any consistent global formula α, according to Lemma 4.1, there is a global formula $\bigvee_{i\in K}\bigwedge_{j\in S_i}(\alpha^i_j@j)$ equivalent to α, where K is a finite set of natural numbers and $S_i \subseteq N$. Now it is sufficient to show $\bigvee_{i\in K}\bigwedge_{j\in S_i}(\alpha^i_j@j)$ is D-satisfiable.

Since α is consistent, then there exists $i \in K$ such that α^i_j is j−consistent for all $j \in S_i$. Suppose not, for all $i \in K$, there is $j \in S_i$ such that α^i_j is not j−consistent, which means

- $\vdash_j \neg\alpha^i_j$.
- $\vdash_D (\neg\alpha^i_j)@j$ by the rule (G_j).
- $\vdash_D \neg(\alpha^i_j@j)$ by the axiom $(B2)$.
- $\vdash_D \neg\bigwedge_{j\in S_i}(\alpha^i_j@j)$.
- $\vdash_D \neg\bigvee_{i\in K}\bigwedge_{j\in S_i}(\alpha^i_j@j)$.

which means $\bigvee_{i\in K}\bigwedge_{j\in S_i}(\alpha^i_j@j)$ is not consistent, it follows that α is not consistent, contradiction.

Without loss of generality, suppose $\bigwedge_{j\in S_0}\alpha^{i_0}_j$ is j-consistent for $i_0 \in K$. Since for each $j \in N$, any j-consistent j-local formula is j-satisfiable. Suppose $\alpha^{i_0}_j$ is j-satisfiable at o_j in local model $M^j = (W^j, \rightarrow_j, o_j, V^j)$ for $j \in S_{i_0}$, and if $S_{i_0} \neq N$, we fix any rooted model $M^j = (W^j, \rightarrow_j, o_j, V^j)$ for $j \in N \setminus S_{i_0}$. Let M be the global model with enabled actions generated from $\{M^j\}_{j\in N}$. Since $M^j, o_j \vDash_j \alpha^{i_0}_j$ for $j \in S_{i_0}$, then $M, (o_1, \dots, o_n) \vdash_D \bigwedge_{j\in S_{i_0}}(\alpha^{i_0}_j@j)$ for $i_0 \in K$, it follows that $M, (o_1, \dots, o_n) \vdash_D \bigvee_{i\in K}\bigwedge_{j\in S_i}(\alpha^i_j@j)$, thus we have $M, (o^1, \dots, o^n) \vdash_D \alpha$, which means α is D-satisfiable. ∎

Basically, we 'reduce' all the concepts at the global level to those at the local levels by means of our axioms, and finally prove the global completeness by local completeness. Note that we cannot obtain the strong completeness result directly through this way, and thus the whole point of the to and fro movements between the global maximal consistent sets and local maximal consistent sets through product construction were taken up in Section 4.2.2.3, explicating the interplay.

4.4 On model checking problems

After considering the completeness results for the logics above, natural follow-up questions that come up include the satisfiability problem and the model checking problem of the concerned systems. We keep studying the satisfiability problem for future work, and focus on the model checking problem here. Such a study has particular relevance in the context of games, for example, we can investigate whether the games underlying the local and global models satisfy some relevant properties. Especially, checking whether a game position is a winning position and finding paths of such winning possibilities starting from certain game positions are some of the main queries one needs to deal with while modeling such frameworks for distributed games. Moreover, the study of complexity of such decision problems is a necessary step towards potential applications. In what follows, we explore the complexity of the model checking problems for DGL and DGLEA. Before doing so, we first define the length of a local formula and a global formula.

Definition 4.16: The length of a local formula in $\mathcal{L}_i$ is defined as follows.

$$|\top| = 1$$

$$|p_i| = 1$$

$$|\neg\alpha_i| = |\alpha_i| + 1$$

$$|\alpha_i \vee \beta_i| = |\alpha_i| + |\beta_i| + 1$$

$$|\langle\gamma\rangle\alpha_i| = |\alpha_i| + 1$$

Definition 4.17: The length of a global formula in $\mathcal{L}$ is defined as follows.

$$|\top| = 1$$

$$|\alpha_i @ i| = |\alpha_i|$$

$$|\neg\alpha| = |\alpha| + 1$$

$$|\alpha \vee \beta| = |\alpha| + |\beta| + 1$$

$$|\langle\gamma\rangle\alpha| = |\alpha| + 1$$

Fact 21: Given a formula ϕ in $\mathcal{L}$ ($\mathcal{L}_i$), let $SF(\phi)$ ($SF_i(\phi)$) be the set of all subformulas of ϕ, then $|SF(\phi)|$ ($|SF_i(\phi)|$) $\leq |\phi|$, where $|S|$ is the size of the set S of formulas in $\mathcal{L}_i$ ($\mathcal{L}$).

Proof This proposition can be proved by induction on the structure of ϕ. ∎

Lemma 4.2: Given a local formula α_i and a local model M^i for agent i, deciding whether α_i is i-satisfiable on M^i takes time $\mathcal{O}(|\alpha_i| \times |M^i|^2)$, where $|M^i|$ is the size of M^i.

Proof This lemma has been proved in (van Benthem, 2010). The approach is as follows: We calculate the truth values of the subformulas that appear in order, considering the time needed in the worst-case scenario. By applying this approach, we are able to derive the desired result. ∎

Based on the results of the aforementioned lemma and the proof technique used, we can conclude the following.

Theorem 4.2: The model checking problem for DGL, i.e., for any global formula ϕ in $\mathcal{L}$ and finite global model M, deciding whether ϕ is satisfiable on M is in **P**.

Proof Given any global formula ϕ and finite global model M generated from local models $\{M^i\}_{i\in N}$, we can check whether ϕ is D-satisfiable on M following the procedure below:

Firstly, enumerate all subformulas of ϕ as $\phi^1, \dots, \phi^m$, in increasing length, such that ϕ^m is ϕ and if ϕ^i is a subformula of ϕ^j then $i < j$. Next, compute the truth value of ϕ^k at each state w of M for $1 \leq k \leq m$ in orders. At each state, we have following cases:

- Case 1: If ϕ^k is a global proposition letter, which is of the form $\alpha_i@i$, then it is sufficient to compute the truth value of α_i at w^i in M^i. According to Lemma 4.2, this step can be carried out in time $\mathcal{O}(|\alpha_i| \times |M^i|^2)$. Since $|\alpha_i| < |\phi|$ and $|M^i| < |M|$ for $i \in N$. Thus this case can be checked in time $\mathcal{O}(|\phi| \times |M|^2)$.
- Case 2: If ϕ^k is of the form $\alpha \vee \beta$, it takes constant time.
- Case 3: If ϕ^k is of the form $\langle\gamma\rangle\,\alpha$, it can be done in time $\mathcal{O}(|M|)$,

Finally, in the worst case, ϕ^m can be completed in time $\mathcal{O}(m \times |M| \times (|\phi| \times |M|^2))$. According to Fact 21, we have $m \leq |\phi|$. Hence, ϕ^m can be completed in time $\mathcal{O}(|\phi|^2 \times |M|^3)$, which means the model checking problem for DGL is in **P**. ∎

For the model checking problem of DGLEA, we can apply a similar method, and obtain the following result.

Theorem 4.3: The model checking problem for DGLEA, i.e., for any global formula ϕ in $\mathcal{L}$ and finite global model with enabled actions M, deciding whether ϕ is D-satisfiable on M is in **P**.

Before concluding the chapter we provide a glimpse of how a more involved logical system with two layers of reasoning, as discussed earlier using the basic modal logic syntax, can be used to describe strategic reasoning, viz. strategic announcements, in the form of best responses. This in turn would lead to corresponding strategic equilibria studies, among others. The basic interaction of local and global reasoning has already been studied in the frameworks DGL and DGLEA, which provide the necessary base for any such study incorporating richer reasoning structures.

4.5 On strategic reasoning

Till now we have focused on the interplay of local and global reasoning, where there were no role of the actual announcements per se. The announcements have been modeled as usual actions/moves in a game, and the investigations on reasoning took shape accordingly. In the process, we presented a couple of descriptive logics, modeling local and global interaction. We expressed what a player can achieve and what actions lead to those outcomes. The actions considered are intuitively described as making announcements. In what follows, we incorporate richer structures in both models and languages to deal with strategic reasoning in distributed games, where we bring announcements to the fore. To motivate, we once again consider a card game combining the ideas from the previous two card games mentioned in this chapter.

Example 4.7: Suppose now, for Alice and Bob, there are 5 available cards, $1, 2, 3, 4$ and 5, say, and each of them gets two cards from this pile of five cards, and one card is kept upside down, so that nobody can see the value. Once again, we specify that both players only announce the card number that is equal to or higher than the actual card they have. The game starts with two rounds of simultaneous announcements about the card numbers that the players have. The first round of simultaneous announcements are dealt with as earlier. Below, we focus on the second round of announcements about the card numbers.

- If they announce different card numbers in the second round, then:
  - If the player with the announcement of the lower card announces 'I accept' in the next round, then the other player wins.
  - If the player with the announcement of the lower card announces 'I challenge' in the next round, then the other player has to show all the cards, and that player wins if the cards match his/her announcement of each round.

- If they announce the same card number, we specify that at least one player has to announce 'I challenge' in the next round. Whoever is challenged has to show all the cards, and that player wins if the cards match his/her announcements of each round. They will restart this round if both of them do not want to announce 'I challenge'.

While introducing DGLEA, we kept the definition of the local arena unchanged and only modified that of the global arena as we were focusing on the actions and/or announcements that are *allowed* in the global arena. Our current focus is to model strategic announcements and accordingly, we introduce announcement maps in both local and global arenas. Correspondingly, at the syntactic level, we introduce operators that reflect the role of announcements more precisely in the games.

Definition 4.18 (Local arena with announcements): A *local arena* for player i is a tuple $\mathrm{G}^i = (W^i, \rightarrow_i, w_0^i, \chi^i)$ such that W^i is the set of local game states, w_0^i is the initial game state, $\chi^i : W^i \rightarrow 2^{\Gamma^i} \setminus \varnothing$ gives the set of possible announcements that an agent can make when she is at a certain game state, and $\rightarrow_i$ is a partial move function given by, $\rightarrow_i : W^i \times \tilde{\Gamma} \rightarrow W^i$ satisfies that for all $w^i, v^i \in W^i$, if $w^i \xrightarrow{\gamma}_i v^i$, then $\gamma[i] \in \chi^i(w^i)$.

Example 4.8: In the game between Alice and Bob, described in Example 4.7. let us assume that each player can announce all the cards, as well as 'accept' or 'challenge'. Suppose Alice has 1, 4 and Bob has 2, 3. Suppose in the first round, both the players truthfully announce their least cards, say, 1 and 2. Then, of course, in the next round, both Alice and Bob would try to strategize for their next announcements depending on what they heard from the opponent. For example, as Alice announced card 1 in the first round, and Bob has the cards 2 and 3, Bob is unsure whether Alice has the card 4 or the card 5. Accordingly, he can consider the following strategy: If Alice announces that she has card 4, then I might get away by announcing that I have card 5. Evidently, Alice is unsure whether Bob has the card 3 or the card 5, and she can strategize similarly. It might also be the case that Bob cannot announce 'challenge'. Then, whatever Bob might consider regarding Alice's announcement, he has to announce a bigger card to win. Then Alice can always announce 5 to ensure her win.

Now, we are all set to define the global arenas with announcements. We note here that the introduction of the announcement maps in the local arena, and inclusion of the corresponding component in the global arena below, have allowed us to deal with the notion of enabled actions, e.g., in DGLEA, quite naturally: The interaction between the *move* function $\xrightarrow{\gamma}$ and the announcement function χ have taken care of it.

Definition 4.19 (Global arena with announcements): Given a set of local arenas $\{G^i\}_{i\in N}$, one for each player, the global arena $G = (W, \rightarrow, w_0, \chi)$ is defined as follows: $W = W^1 \times \ldots \times W^n$, $w_0 = (w_0^1, \ldots, w_0^n)$, $\chi : W \rightarrow 2^{\tilde{\Gamma}}$ satisfying $\chi(w) = \chi^1(w^1) \times \ldots \times \chi^n(w^n)$, where $w = (w^1, \ldots, w^n)$, and $\rightarrow: W \times \tilde{\Gamma} \rightarrow W$ is defined by: for all $w, v \in W$, we have $w \xrightarrow{\gamma} v$ iff

- for all $i \in enabled(w, \gamma)$, $w^i \xrightarrow{\gamma}_i v^i$, with $\gamma \in \chi(w)$,
- for all $i \in N \setminus enabled(w, \gamma)$, $v^i = w^i$,

where, $enabled(w, \gamma) = \{i \in N \mid \exists v^i \in W^i \text{ with } w^i \xrightarrow{\gamma}_i v^i,\ where\ \gamma \in \chi(w)\} \neq \emptyset$. For each $i \in N$, define $\chi^i(w) = \chi^i(w^i)$, where $w = (w^1, \ldots, w^n)$.

Example 4.9: Suppose the cards are as discussed in Example 4.8. For the global arena, one can say that a (1,2) announcement, and subsequently a (4,5) announcement followed by an (A,5) announcement would lead to a win for Bob. On the other hand, if Bob cannot announce a card that he already has, it would reduce the possibility of his winning. As discussed earlier, if Alice announces 1 followed by 5, then Bob, in case he challenges, would not be able to conclude in his local arena whether he will win, but in the global arena one can conclude a win for Bob. Also, one could describe different kinds of strategies, for example, if Alice announces 3, then there is no point in Bob announcing 2 in the next round, or, Alice or Bob deciding to announce a certain number irrespective of what the other's announcement might be.

Given these local and global arenas with announcements, the local announcement syntax is given as follows:

Definition 4.20: The local announcement language for player i, $A\mathcal{L}_i$, is given as follows:

$$\alpha \in A\mathcal{L}_i ::= \tau \in \Gamma^i \mid \sigma@j, \sigma \in \Gamma^j \mid \neg\alpha \mid \alpha \vee \beta \mid \Box\alpha$$

Thus, the local formulas consist of two kinds of atomic formulas, one's own announcements and possible/actual announcements of others, their boolean combinations and the usual basic modal operator. For providing the truth definition of local formulas, since announcements made by the other players are also taken under consideration, we consider both local and global models. The whole point of the local semantics is to evaluate announcement formulas. The truth definition of the local formulas are given as follows:

Definition 4.21: Given local arenas with announcements, M^i for each player i, and the corresponding global model M (defined below), the truth definition of the local announcement formulas at a world in a model are defined inductively as follows:

- $M, M^i, w^i \models_i \tau$ iff $\tau \in \chi^i(w^i)$.
- $M, M^i, w^i \models_i \sigma@j$ iff there exists w in the global model M whose i-th component is w^i and the j-th component w^j satisfies the following condition: $M, M^j, w^j \models_j \sigma$
- $M, M^i, w^i \models_i \neg\alpha$ iff $M, M^i, w^i \not\models_i \alpha$
- $M, M^i, w^i \models_i \alpha \vee \beta$ iff $M, M^i, w^i \models_i \alpha$ or $M, M^i, w^i \models_i \beta$
- $M, M^i, w^i \models_i \Box\alpha$ iff for all successor states v^i of w^i in M^i, such that $M, M^i, v^i \models_i \alpha$

We note that $\sigma@j$ asserted by player i may be epistemic in nature according to the following interpretation: it is consistent for i to believe that j's local state supports σ. This requires further discussion which we leave for the future. In what follows, all the local announcement formulas are evaluated at the root node of the respective local arenas.

Example 4.10: For the game in Example 4.7, Bob's strategic announcement, as described in Example 4.8, can be expressed as follows: $(1@Alice \wedge 2) \wedge ((\Box(4@Alice) \rightarrow 5) \vee (\Box(5@Alice) \rightarrow (3 \wedge \Box C)))$.

We move on to global announcement formulas, which are given in the following. We fix a set of global propositions P_i for every $i \in N$.

Definition 4.22: The global announcement language, $A\mathcal{L}$, is given as follows:

$$\phi, \psi \in A\mathcal{L} ::= p@i \mid \alpha_i@i \mid \neg\phi \mid \phi \vee \psi \mid \langle\gamma\rangle\phi \mid \Box\phi$$

where $p \in \mathsf{P}_i, \alpha_i \in A\mathcal{L}_i, \gamma \in \tilde{\Gamma}$.

The main addition in the global syntax is the announcement modality $\langle\gamma\rangle$ as we had it in Section 4.2. The global semantics is given as follows.

Definition 4.23: A global model is given by $M = (\mathrm{G}, O)$, where G is a global arena, and $O : W_l \rightarrow 2^{\mathsf{P}_1} \times \ldots \times 2^{\mathsf{P}_n}$ is a valuation. The truth definitions of the global formulas are given as follows:

- $M, w \models \alpha@i$ if $M^i, w^i \models_i \alpha$
- $M, w \models p@i$ if w is a leaf node and p is in the i-th coordinate of $O(w)$
- $M, w \models \neg\phi$ if $M, w \not\models \phi$
- $M, w \models \phi \vee \psi$ if $M, w \models \phi$ or $M, w \models \psi$
- $M, w \models \langle\gamma\rangle\ \phi$ if there exists $v \in W$ such that $w \xrightarrow{\gamma} v$ and $M, v \models \phi$.
- $M, w \models \Box\phi$ if for all v and all γ such that $w \xrightarrow{\gamma} v$, $M, v \models \phi$

We end this section with a glimpse of what the new global language can express in terms of the example we described above.

Example 4.11: Finally, we can provide global outcome formulas corresponding to the deal that Alice (1,4) and Bob (2,3) have in Example 4.8 as follows.

- Alice and Bob announce $(3,2)$ and then $(5,5)$, followed by the challenge announcements from both of them and then both of them lose. Formally: $[(3,2)][(5,5)]\langle(C,C)\rangle\neg(W@Alice \vee W@Bob)$. As we can see, both their actions in the second round lead them to lose the game.
- Alice and Bob announce $(3,2)$ in the first round and then in the second round, no matter what Alice announces, Bob chooses to announce the card number 3. Formally, $[(3,2)]([(1,3)]\langle(C,3)\rangle(W@Bob) \wedge ([(2,3)]\langle(C,3)\rangle(W@Bob)) \wedge ([(3,3)](\langle(C,C)\rangle(W@Bob) \wedge (\langle(C,3)\rangle(W@Bob)) \wedge (\langle(3,C)\rangle\neg(W@Alice))) \wedge ([(4,3)]\langle(4,C)\rangle\neg(W@Alice)) \wedge ([(5,3)]\langle(5,C)\rangle\neg(W@Alice)))$. As long as Bob sticks to this strategy that he announces the card number 3 in the second round, then he definitely would not lose.
- Also, if Bob follows his strategic announcement as given in Example 4.10, he will end up winning: $((1@Alice \wedge 2) \wedge ((\Box(4@Alice) \rightarrow 5) \vee (\Box(5@Alice) \rightarrow (3 \wedge \Box C))))@Bob \rightarrow \langle(1,2)\rangle\langle(5,3)\rangle\langle(5,C)\rangle(W@Bob)$.

In the above examples, we formally express both local and global strategic reasoning, showing strategies that can be good or bad for the players, and also those that can be an optimal one for one of the players in a specific context. This logic is intended to show how strategic reasoning can be developed further based on the foundations laid here. We will take up the technical development of such logics in further work. For now, we conjecture the following: The satisfaction problem of this logic is decidable.

4.6 Summary and further work

In conclusion, we have proposed a two-layered propositional modal logic DGL at the outset, for reasoning in distributed games, in which every player has access only to her local game arena and makes choices based on local game state and public announcements from other players. The local and the global modality have the same syntax - the former on one hand describes synchronous choices and on the other hand considers possible assumptions made by a subset of players, whereas the latter describes global moves, that is, a simple combination of such local moves. The axiom system brings out the complexity of such reasoning, and the completeness proof, the intricacy of the product construction. Furthermore, using a similar logical framework, viz. DGLEA, we have dealt with global models for distributed games with a more realistic construction of global moves based on subtle combinations of local moves. In addition, we explored the complexity of the model checking problems for both these logics, showing them to be tractable as in the case of basic modal logic. While both these logics are descriptive in nature, based on the foundation laid there, we have proposed a third framework elucidating strategic choices and player responses. The technical development of this logic is along similar lines and will be taken up in the future.

We note that the kind of global reasoning we consider here which is not arising as product of local reasoning, involve outcomes evaluated at the terminal or leaf nodes of the games which are basically booleans formulas. An important question arises as to how much global reasoning we can accommodate within such product based reasoning. The logics proposed in this chapter are minimal, with no modalities for transitive closure, and others. While some of these extensions proceed along standard lines, products bring their own complications, which are worth pursuing.

Another important line of work that we are much interested in is epistemic reasoning in distributed games. Note that due to announcements being always public, two players i and j are equally (un)certain about a third player k. We hope to characterize such strategic reasoning influenced by the information available to the players in our next steps.

On the whole, this work can be termed as an initial logic-based study on the structural aspects of local and global arenas concerning distributed games. It would also be worthwhile to build on these frameworks for providing logical characterizations of game-theoretic concepts concerning such games, e.g., notions of equilibrium coming from the structural interplay, dominant and dominated strategies, and others. Such investigations may pave the way for further compositional studies of structural interplay in strategic reasoning in these games modeling local and global phenomena.

CHAPTER 5 BISIMULATION IN MODEL-CHANGING MODAL LOGICS: AN ALGORITHMIC STUDY

In Chapter 3, we have studied different aspects of sabotage modal logic which can be considered as one of many model-changing modal logics that have been introduced over the years to capture the model dynamics in many relevant areas, from mathematical systems to machine intelligence, from economic theories to philosophical queries as well as other important phenomena. Among the notable topics in model dynamics, these logical systems deal with dynamical systems, graph and game dynamics, information change, strategy-plan-protocol-programming updates, belief revision, social network updates, memory upgrades and many others. The current chapter broadens the focus of our semantic study on one hand, considering a varied collection of model-changing frameworks. On the other hand the study narrows down to that of a particular property of the logics discussed, viz. model invariance or bisimulation. Before moving on to the main study, let us first give a brief survey of the existing literature on model-changing logics.

5.1 Model-changing modal logics: A discussion

The study on these logics initiated with the introduction of public announcement logic (*PAL*) that deals with information updates brought about by public communication. In *PAL*, as studied in (Plaza, 1989, 2007), the evaluation of the announcement formulas $\langle\varphi\rangle\psi$ (read as 'there is an announcement of φ after which ψ holds') involves deleting all the $\neg\varphi-$worlds in the Kripke models and consequently, the relations involving those worlds. In contrast, in *PAL* as studied in (Gerbrandy and Groeneveld, 1997), the evaluation of the formula $[\varphi]_a\psi$ (read as 'after all possible a-announcement of φ, ψ holds', where a is an agent) involves deleting arrows in the Kripke models that are pointing to $\neg\varphi-$worlds, with the domain remaining the same. Dynamic epistemic logic (*DEL*) (Baltag et al., 1998; Baltag and Moss, 2004; van Ditmarsch et al., 2008; van Benthem, 2011), a generalization of *PAL*, characterizes such announcements in more subtle communications. Towards modeling belief upgrades among agents, model-transforming operators like lexicographic upgrade $[\Uparrow]$, elite change $[\uparrow]$ and suggestion $[\#]$ capture plausibility relation updates under soft information (van Benthem and Liu, 2007; van Benthem, 2007) involving relational changes without changes in the domain. A similar model-changing modal framework, arrow update logic (*AUL*) in (Kooi and Renne, 2011) describes epistemic access elimination that can be used to reason about multi-agent belief change. In addition to such structural changes discussed above, factual changes in these models are captured by updating valuations (Renardel de Lavalette, 2004; van Ditmarsch et al., 2005; van Benthem et al., 2006; Kooi, 2007).

Model-changing logics have also played a significant role in describing strategic reasoning in games on graphs, which in turn have received a lot of attention in diverse domains, e.g., computer science, logic, linguistics, economics, mathematics, philosophy, biology and others. As the name indicates, such a game is played on directed or undirected graphs, and the players' actions are assigned based on the game designer's research objectives. One can also consider different variants of such graph games where such variations can arise from different winning conditions (e.g., reachability, parity in (Grädel, 2011)), independent moves of players (e.g., cops and robber game in (Nowakowski and Winkler, 1983)), one player obstructing moves of the others (e.g., sabotage game in (van Benthem, 2005), poison game in (Duchet and Meyniel, 1993)) and others. In the interplay between game theory, logic and computer science, these graph games provide exploratory models for reactive systems that need to interact with the uncertain environment.

From the perspective of relation changes in models, in particular, link/edge deletion in graphs, sabotage games (van Benthem, 2005) are natural examples where one player is concerned with a reachability objective and the other player is involved in obstructing her opponent's moves by deleting edges from the graph. Model-changing logics related to sabotage-style graph games with edge deletions are presented in (van Benthem and Liu, 2020; Rohde, 2005; Li, 2020; Baltag et al., 2019b; van Benthem et al., 2022). For example, consider the language of definable modal sabotage logic (S_dML) in (Li, 2020), which is a direct extension of basic modal language with an additional operator $[-\psi]$, where, $[-\psi]\varphi$ expresses the condition that after local deletion/sabotage of all arrows from the current point, whose end points satisfy ψ, φ still holds. The modal logic of supervised learning (*SLL*) in (Baltag et al., 2019b) is equipped with even more advanced sabotage operators that are introduced to characterize relation changes in multi-relation models. Moreover, there is a generalized sabotage operator in (van Benthem et al., 2022), denoted by $\blacklozenge^{\alpha}_{\beta}$ that is used to capture an arrow deletion whose end-points satisfy α and β, respectively.

With regard to domain changes in models, a game that is close to the spirit of describing point/vertex deletion on graphs is the poison game in (Duchet and Meyniel, 1993): one player is concerned with moving indefinitely in the game graph, and her opponent is involved in obstructing her moves by poisoning certain vertices whose effect is analogous to that of 'point deletion' from the perspective of the former player. To reason about poison games, model-changing logics *PSL* and *PML* in (Zaffora Blando et al., 2020; Grossi and Rey, 2019) use operators for changing valuations in and/or domains of models, which are inspired by memory logics (Zaffora Blando et al., 2020; Mera, 2009; Areces et al., 2008). In addition, operators involving such changing of valuations are also mentioned in (Thompson, 2020; van Benthem and Liu, 2020) and, point-deletion style operators have been proposed in (van Benthem, 2005; Fervari, 2014; van Benthem et al., 2020; Areces et al., 2012). To give an example of what we are talking about, consider the operator $\langle -\psi \rangle$ in the language of modal logic of stepwise removal (*MLSR*) proposed in (van Benthem et al., 2020) - it involves deletion of $\psi-$worlds in the models.

Let us digress a bit and talk about the game logic connection here. In general, these logics are referred to as game (graph) logics, more intuitively, as logics of game boards. The underlying idea is that they characterize changes in the graph that serves as the game board, such as sabotage game logic (Kvasov, 2016). They are somewhat different from the logics for extensive-form games (Osborne and Rubinstein, 1994) in the sense that the latter ones are mostly proposed to deal with and reason about various game-theoretic phenomena. For example, Harrenstein et al. (2003) characterize (subgame perfect) Nash equilibrium concept for extensive-form games. To summarize, logics for graph games focus on the structural changes of the game board, while those extensive-form game logics focus on logical analyses of game-theoretic concepts. In fact, graph games can also be described using extensive-form games, where one can think of possible unfoldings at any node of the game graph, and the nodes of the game tree correspond to sequences of play through the possible graph structures during the course of the game.

Coming back to the perspective of models, the logics that we are talking about aim to capture three mechanisms of model transformation, namely, those describing domain-changes, relation-changes and valuations-changes in models and their combinations. In this work, we concentrate on various extensions of basic modal logic with the new operator $\langle up \rangle$, which we call $MCML(up)$, where $\langle up \rangle$ reflects various mechanisms of model transformation. In particular, we deal with the bisimulation / model comparison problems of such model-changing logics. In model-theoretic studies of modal logics, the notion of bisimulation plays a central role as formally, whenever a mathematical structure or a model is introduced, a notion of invariance of such models comes up by default. And, bisimulation is the notion of model-invariance with respect to modal languages. Thus it comes up quite naturally in a study featuring the landscape of model changing modal logics. In addition, from the viewpoint of the logics of game boards, this notion of bisimulation paves the way for measuring equivalence of 'game boards' that is, the graphs, for the purposes of playing the game. For example, a question to ponder upon while considering such notions of game equivalence: What simplest graph in an equivalence class still shows the essential structure of the game? Moreover, from the context of the underlying graph structure and the dynamic nature of our study, one could also think studying novel invariance notions of graoh dynamics, based on the different bisimulation notions described below. Last but not the least, across all these viewpoints, the notion of bisimulation or model comparison aids in the study of the expressive power of the corresponding modal languages.

In this work, we provide a uniform algorithmic study of the model comparison problem to shed some light on the complexity of these problems. For our purposes, we consider the operators $\langle sb \rangle$ and $\langle gsb \rangle$ (Areces et al., 2012; Aucher et al., 2018; Fervari, 2014; van Benthem et al., 2022; Rohde, 2005) to model edge deletion in models, $\langle br \rangle$ and $\langle gbr \rangle$ (Fervari, 2014) to model edge addition in models, and $\langle sw \rangle$ and $\langle gsw \rangle$ (Areces et al., 2014, 2012; Fervari, 2014) to model arrow swap in models. In addition, we consider $\langle de \rangle$ (van Benthem, 2005) and $\langle ch \rangle$ (Thompson, 2020) for point deletion and valuation change in models, respectively. Such a study provides insight into the complexity of the bisimulation problems of these modal logics which have not been studied before.

We note here that there is a strand of literature exploring technical properties of these logics. *MCML*(gsb) was first introduced in (van Benthem, 2005), and complete proof systems for *MCML*(gsb) have been discussed in (Aucher et al., 2018; Fervari, 2014). For the decidability and complexity questions, we have the following results (Löding and Rohde, 2003b; Fervari, 2014; Löding and Rohde, 2003a; Thompson, 2020): (i) for $\langle up \rangle \in \{\langle sb \rangle, \langle gsb \rangle, \langle sw \rangle, \langle br \rangle, \langle ch \rangle\}$, the satisfiability problem for *MCML*(up) is *undecidable*, and (ii) for $\langle up \rangle \in \{\langle sb \rangle, \langle gsb \rangle, \langle sw \rangle, \langle gsw \rangle, \langle br \rangle, \langle gbr \rangle\}$, the model-checking problem for *MCML*(up) is *PSPACE-complete*. A result that is missing in this picture is the complexity of bisimulation or the model comparison problem, which is worth exploring. From a game-theoretic perspective, the notion of bisimulation not only helps us to understand the expressive power of logics on game boards, but also measures the equivalence of games (graphs) for the purposes of playing the game.

We should mention here that the model comparison problem has been in the radar of researchers for a long time. In addition to the development of verification algorithms (Cleaveland et al., 1993; Garavel et al., 2013; Bunte et al., 2019), model comparison problem also holds fundamental importance in the field of concurrency theory and related areas of computer science (Aceto et al., 2011; Hopcroft et al., 2001). It is well-known that deciding bisimilarity over finite labelled transition systems is in deterministic polynomial time (Paige and Tarjan, 1987; Balcázar et al., 1992; Kanellakis and Smolka, 1983). To the best of our knowledge, complexity study of the bisimulation problems concerning these model-changing logics is still open. Solving this problem will, on one hand, provide us with a finer understanding of the practical applicabilities of these logics, and on the other hand, provide us with better insights about their expressive powers. In this work, we provide *PSPACE* upper bounds for the bisimulation problems for all the model-changing modal logics described above. Our quest for the lower bounds for these problems did not deliver any result at the current point, and we leave '*finding lower bounds for these bisimulation problems*' as open questions.

The rest of the chapter can be summarized as follows: In the following section, we introduce the relevant logic frameworks together with their respective notions of bisimulations. The next one gives us a detailed algorithmic study together with upper bounds of the complexity of the relevant problems. The final section provides some further related results and concludes the paper with a discussion on the lower bound.

5.2 A general framework

In this section, we first describe the various model-changing logics that we are going to base our study on. We also recapitulate the corresponding notions of bisimulation. The main focus will be on the logics describing relation updates where the domains remain fixed. In addition, we will also look into domain updates as well as valuation updates. To have a uniform description of these logics, we start with a general framework followed by the specific ones.

5.2.1 A uniform language

Given a countable, infinite set of propositional variables Prop, The syntax of the general model-changing modal logic $MCML(up)$ is given as follows:

$\varphi : p \mid \neg\varphi \mid \varphi \wedge \varphi \mid \Diamond\varphi \mid \langle up\rangle\psi,$

where $p \in \mathsf{Prop}$, $\langle up \rangle$ is a model-update modality. The dual $[up]\psi$ formula is defined as usual: $\neg\langle up \rangle\neg\psi$.

The models for $MCML(up)$ are given by usual relational models $\mathcal{M} = (W, R, V)$ for modal logics, where, W is a non-empty set, $R \subseteq W \times W$, and $V : W \to 2^{\mathsf{Prop}}$. A pair $(\mathcal{M}, w)$, where $w \in W$ is called a *pointed model*. Let $\mathfrak{M}$ denote the class of all pointed models, and r_{up} be a subset of $\mathfrak{M} \times \mathfrak{M}$ corresponding to the operator $\langle up \rangle$. Given a pointed model $(\mathcal{M}, w)$, the set $\{(\mathcal{M}', w') \mid ((\mathcal{M}, w), (\mathcal{M}', w')) \in r_{up}\}$ collects all the updated pointed models from $(\mathcal{M}, w)$ that we get with respect to the operator $\langle up \rangle$. The truth definition of the formulas of $MCML(up)$ in pointed models are as usual for the boolean and the modal formulas, and for the operator $\langle up \rangle$, it is given as follows:

- $(\mathcal{M}, w) \vDash \langle up \rangle\psi$ *iff there is a pointed model* $(\mathcal{M}', w')$ *with* $((\mathcal{M}, w), (\mathcal{M}', w')) \in r_{up}$ *and* $(\mathcal{M}', w') \vDash \psi$.

With the syntax and semantics out of the way, we now focus on the following question which forms the backbone of this work: when do two pointed models satisfy the same formulas under the language $MCML(up)$? The definition of the relevant bisimulation concept, that is, $\langle up \rangle$-bisimulation is given as follows.

Let $\mathcal{M}_1 = (W_1, R_1, V_1)$ and $\mathcal{M}_2 = (W_2, R_2, V_2)$ be two relational models. A non-empty relation Z over a set of pointed models is an $\langle up \rangle$-bisimulation between $(\mathcal{M}_1, w_1)$ and $(\mathcal{M}_2, w_2)$, denoted by $(\mathcal{M}_1, w_1) Z (\mathcal{M}_2, w_2)$, if the following conditions are satisfied:

(1). **Atom**: If $(\mathcal{M}_1, w_1) Z (\mathcal{M}_2, w_2)$, then $(\mathcal{M}_1, w_1) \vDash p$ iff $(\mathcal{M}_2, w_2) \vDash p$ for all atomic propositions $p \in \mathsf{Prop}$.

(2). **Zig**$_\Diamond$: If $(\mathcal{M}_1, w_1) Z (\mathcal{M}_2, w_2)$, and there exists $v_1 \in W_1$ such that $w_1 R_1 v_1$, then there is a $v_2 \in W_2$ such that $w_2 R_2 v_2$ and $(\mathcal{M}_1, v_1) Z (\mathcal{M}_2, v_2)$.

(3). **Zag**$_\Diamond$: Same as above in the converse direction.

(4). **Zig**$_{up}$: If $(\mathcal{M}_1, w_1) Z (\mathcal{M}_2, w_2)$, and there exists a pointed model $(\mathcal{M}_1', u_1)$ such that $((\mathcal{M}_1, w_1), (\mathcal{M}_1', u_1)) \in r_{up}$, then there exists a pointed model $(\mathcal{M}_2', u_2)$ such that $((\mathcal{M}_2, w_2), (\mathcal{M}_2', u_2)) \in r_{up}$ and $(\mathcal{M}_1', u_1) Z (\mathcal{M}_2', u_2)$.

(5). **Zag**$_{up}$: Same as above in the converse direction.

Note that the definition above is given in a generalized way, we shall make changes below according to the specific operators. Generally speaking, there are three cases.

- We do not need to make any adjustments, the definition may fit well for the operator $\langle up \rangle$ under consideration.
- The dynamics of the models, that the operator $\langle up \rangle$ reflects, may be quite compli-

cated. Then, an abundant amount of information may be wrapped up in the respective definitions of r_{up} that we shall process further with respect to the items (4) and (5). For example, in the item (4), '$((\mathcal{M}_1, w_1), (\mathcal{M}'_1, w'_1)) \in r_{up}$' may involve complex formulas being satisfied at certain points, which shall be translated into additional conditions for bisimulation. In such cases, we shall restate the items (4) and (5) in the terms of the specific forms of the operator $\langle up \rangle$.

- Alternatively, the operator $\langle up \rangle$ may not increase the expressivity of the logic, which means that for any formula with $\langle up \rangle$, there is an equivalent formula without it. In such cases, items (4) and (5) become redundant, and we shall not consider them.

Thus, we treat the definition of bisimulation above in a broader perspective and many specific instances will be taken up later where we will delve into the minute details. Based on this definition, we can prove that bisimulation implies modal equivalence which we claim formally in the following. For simplicity, if there is an $\langle up \rangle$-bisimulation between two pointed models $(\mathcal{M}_1, w_1)$ and $(\mathcal{M}_2, w_2)$, we call them $\langle up \rangle$-bisimilar.

Proposition 5.1: If two pointed models $(\mathcal{M}_1, w_1)$ and $(\mathcal{M}_2, w_2)$ are $\langle up \rangle$-bisimilar, then they satisfy the same formulas of the logic $MCML(up)$.

Proof We can prove this by applying induction on the structure of formulas, and we only focus on the formula of the form $\langle up \rangle \psi$. Suppose that $(\mathcal{M}_1, w_1) \vDash \langle up \rangle \psi$. Then there is $(\mathcal{M}'_1, u_1)$ such that $((\mathcal{M}_1, w_1), (\mathcal{M}'_1, u_1)) \in r_{up}$, and $(\mathcal{M}'_1, u_1) \vDash \psi$. According to the definition of $\langle up \rangle$-bisimulation, there exists $(\mathcal{M}'_2, u_2)$ such that $((\mathcal{M}_2, w_2), (\mathcal{M}'_2, u_2)) \in r_{up}$ and $(\mathcal{M}'_1, u_1) Z (\mathcal{M}'_2, u_2)$. we have $(\mathcal{M}'_2, u_2) \vDash \psi$ by I.H., it follows that $(\mathcal{M}_2, w_2) \vDash \langle up \rangle \psi$. ∎

5.2.2 On specific ones

We have proposed the language $MCML(up)$ for describing certain model-changing logics in a uniform way and the corresponding notion of bisimulation. Next we will demonstrate the specific notions of bisimulations with respect to the specific logics.

A number of model-changing operators have been proposed over the years which are basically modelling different dynamic mechanisms. We now investigate some of these modal operators characterizing basic mechanisms of model-changing. The operators $\langle sb \rangle, \langle gsb \rangle, \langle sw \rangle, \langle gsw \rangle, \langle br \rangle$ and $\langle gbr \rangle$ are proposed to capture relation-changing in models, while $\langle de \rangle$ is proposed to characterize domain-changing in models (followed by relation-changes), and $\langle ch \rangle$ for valuation-changing. We have chosen these operators as representatives for expressing the three different kinds of model-changing operations: (i) domain-changing, (ii) relation-changing (with domain remaining fixed) and (iii) valuation-changing (with domain and relation remaining fixed). The intuitive meaning of these operators are as follows.

- $\langle sb\rangle\psi$ can be read as 'it is the case that ψ, after we sabotage some arrow starting at the present point'.
- $\langle gsb\rangle\psi$ can be read as 'it is the case that ψ, after we sabotage some arrow in the model'.
- $\langle sw\rangle\psi$ can be read as 'it is the case that ψ, after we swap some arrow starting at the present point'.
- $\langle gsw\rangle\psi$ can be read as 'it is the case that ψ, after we swap some arrow in the model'.
- $\langle br\rangle\psi$ can be read as 'it is the case that ψ, after we add a new arrow at the present point'.
- $\langle gbr\rangle\psi$ can be read as 'it is the case that ψ, after we add a new arrow in the model'.
- $\langle de\rangle\psi$ can be read as 'it is the case that ψ, after some point is deleted from the model'.
- $\langle ch\rangle\psi$ can be read as 'it is the case that ψ, after the valuation at the present point is updated'.

All these operators have been studied extensively in the literature. The operators $\langle sb\rangle, \langle br\rangle$ appear in (Areces et al., 2012; Fervari, 2014), $\langle gsb\rangle$ appears in (Areces et al., 2012; Aucher et al., 2018; Fervari, 2014; van Benthem et al., 2022; Rohde, 2005), $\langle sw\rangle$ appears in Areces et al. (2014, 2012); Fervari (2014), $\langle gsw\rangle, \langle gbr\rangle$ are proposed in (Fervari, 2014), $\langle de\rangle$ occurs in (van Benthem, 2005) and $\langle ch\rangle$ is proposed in (Thompson, 2020) (with $\bigcirc$ expressing the same). We now define the corresponding r_{up}s'.

Let $\mathcal{M}_1 = (W_1, R_1, V_1)$ and $\mathcal{M}_2 = (W_2, R_2, V_2)$ be two models with $w \in W_1, v \in W_2$. We give the specific definitions of r_{up}, where $\langle up\rangle$ can be the operators we mentioned above. We have that $((\mathcal{M}_1, w), (\mathcal{M}_2, v)) \in r_{up}$ if the following holds:

- $\langle sb\rangle$: $W_2 = W_1$, $(w, v) \in R_1$, $R_2 = R_1\backslash\{(w, v)\}$ and $V_2 = V_1$.
- $\langle gsb\rangle$: $W_2 = W_1, R_2 = R_1\backslash\{(w_1, w_2)\}$ for some $(w_1, w_2) \in R_1$, $V_2 = V_1$ and $w = v$.
- $\langle sw\rangle$: $W_2 = W_1$, $(w, v) \in R_1$,$R_2 = R_1\backslash\{(w, v)\} \cup \{(v, w)\}$ and $V_2 = V_1$.
- $\langle gsw\rangle$: $W_2 = W_1, R_2 = R_1\backslash\{(w_1, w_2)\} \cup \{(w_2, w_1)\}$ for some $(w_1, w_2) \in R_1$, $V_2 = V_1$ and $w = v$.
- $\langle br\rangle$: $W_2 = W_1$, $(w, v) \notin R_1$, $R_2 = R_1 \cup \{(w, v)\}$ and $V_2 = V_1$.
- $\langle gbr\rangle$: $W_2 = W_1, R_2 = R_1 \cup \{(w_1, w_2)\}$ for some $(w_1, w_2) \notin R_1$, $V_2 = V_1$ and

$w = v$.

- $\langle de \rangle$: $W_2 = W_1 \backslash \{w_1\}$ for some $w_1 \neq w$ in W_1, $R_2 = \{(u, v) \in R_1 \mid u \neq w_1$ *and* $v \neq w_1\}$, $V_2(u) = V_1(u)$ for all $u \in W_2$ and $w = v$.
- $\langle ch \rangle$: $W_2 = W_1$, $R_2 = R_1$, $V_2(w) = A$ and $V_2(u) = V_1(u)$ for $u \neq w$, where A is a set of proposition letters, and $w = v$.

Intuitively, the truth conditions of the above operators can be displayed in Figure5.1–5.8. For example, in Figure5.1, $\langle sb \rangle \varphi$ is true at $(\mathcal{M}_1, w)$, if and only if there exists pointed model $(\mathcal{M}_2, v)$ with $((\mathcal{M}_1, w), (\mathcal{M}_2, v)) \in r_{sb}$ such that φ is true at $(\mathcal{M}_2, v)$. It is worth mentioning that when $\langle up \rangle$ is $\langle ch \rangle$, we have a single item to replace the items 4 and 5 as follows.

$(\star)$ $(W_1, R_1, V_1, w_1) Z (W_2, R_2, V_2, w_2)$ implies $(W_1, R_1, {V_1}_A^{w_1}, w_1) Z (W_2, R_2, {V_2}_A^{w_2}, w_2)$ for every $A \subseteq \mathsf{Prop}$, where for $i = 1, 2$, ${V_i}_A^{w_i}$ is almost V_i, except ${V_i}_A^{w_i} = A$.

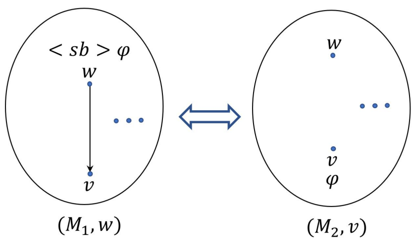


Figure 5.1 $\langle sb \rangle \varphi$

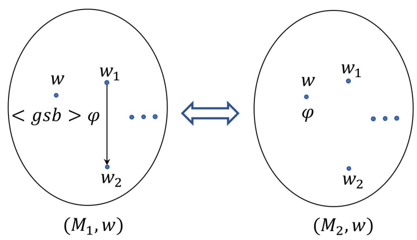


Figure 5.2 $\langle gsb \rangle \varphi$

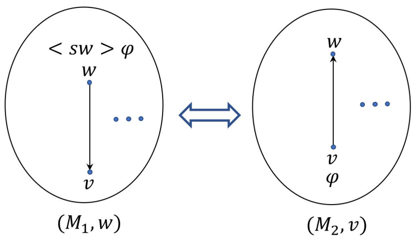


Figure 5.3 $\langle sw \rangle \varphi$

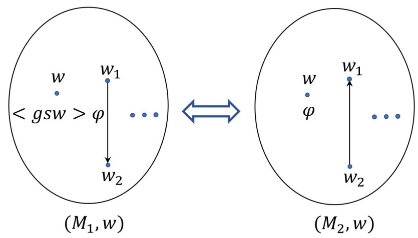


Figure 5.4 $\langle gsw \rangle \varphi$

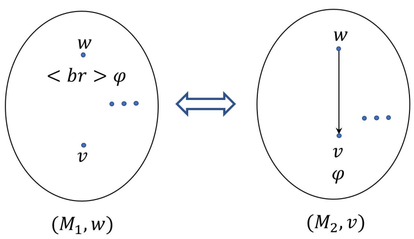


Figure 5.5 $\langle br \rangle \varphi$

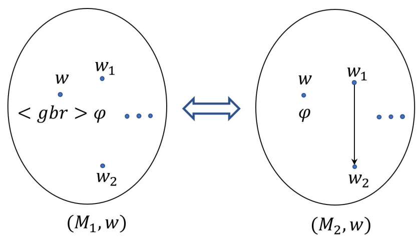


Figure 5.6 $\langle gbr \rangle \varphi$

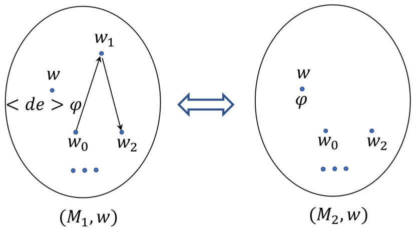


Figure 5.7 $\langle de \rangle\varphi$

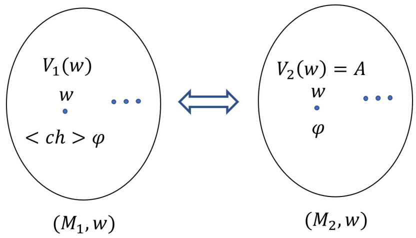


Figure 5.8 $\langle ch \rangle\varphi$

Meanwhile, Proposition 5.1 still works when we unfold the definitions of $\langle up \rangle$-bisimulation for different operators.

For basic modal logic, we can use the notion of bisimulation to reduce equivalently a given model to a smaller one. Moreover, given a finite model M, we can find a minimal model and it is bisimilar to the original model M, i.e., the bisimulation contraction (van Benthem, 2010). With these distinct notions of $\langle up \rangle$-bisimulation above, do we have the corresponding notions of $\langle up \rangle$-bisimulation contraction? For finite models, the simple answer is Yes.

Fix a finite pointed model (M_1, w_1), we aim to find a minimal model M_2, and a world w_2, such that (M_1, w_1) is $\langle up \rangle$-bisimilar to (M_2, w_2), there is a straightforward method to obtain such a result. The approach is to systematically test all models that are smaller than M_1 and check if they are $\langle up \rangle$-bisimilar. The key to this method is to have an algorithm that can determine whether two pointed models are $\langle up \rangle$-bisimilar, and it is desirable that the algorithm is efficient. Hence, we will investigate the complexity of the following decision problem: given two relational models, are they $\langle up \rangle$-bisimilar?

5.3 An algorithmic study

Let us now provide algorithms to check whether two pointed models are $\langle up \rangle$-bisimilar - we have eight distinct notions of bisimilarity based on different logics. Natural questions would be as follows: Do we need to have eight distinct algorithms or can we have a generalized one? How are these algorithms connected to each other? Can one be reduced to the other? We will first provide a general algorithm to check for bisimulation among models in all these logics and then move on to provide the same for the specific ones for a better understanding of the inherent connections/differences between various model-changing phenomena.

5.3.1 The algorithm(s)

In what follows, we provide a general algorithm (Algorithm 5.1). We define a function gen-Bisimilar that takes as input two pointed relational models, $(\mathcal{M}_1, w_1)$, $(\mathcal{M}_2, w_2)$ and two lists $L, L' \subseteq W_1 \times W_2$, where $\mathcal{M}_1 = (W_1, R_1, V_1)$ and $\mathcal{M}_2 = (W_2, R_2, V_2)$, and a state variable to specify the notion of bisimilarity that is to be checked. It outputs 'Yes' if the two models are bisimlar, in the notion specified, and the function is called with $L = L' = \emptyset$. All the notions of bisimilarity that we considered have five conditions to check. In the algorithm, we write a function to check these five conditions. Across different notions, conditions (1) – (3) remain the same. Therefore the only difference in the run of the algorithm for different notions comes in the implementation of conditions (4) and (5). Now, one of the main problems that may come in implementation is when the given models have cycles. We have to check the successors for the (gen) bisimilarity too. This process may not terminate if the given model is pointed at a node that is part of a cycle. We take care of this problem by maintaining a list of edges we have travelled. We initialize the algorithm with this list being empty, and keep adding edges that we have travelled before changing the models. We again make the list empty after the model changing step.

Another way to think about writing this algorithm might be to use the existing algorithm for modal bisimulation (which is a poly-time algorithm) and add on to it to take care of the extra conditions. This type of approach does not directly work as it is not enough to check the satisfaction of the extra conditions for the two bisimliar models. Take the following example (cf. Figure 5.9). The models $(\mathcal{M}_1, w_1)$ and $(\mathcal{M}_2, w_2)$ are bisimilar in the basic modal logic sense. Moreover, the pointed models $(\mathcal{M}_1, w_1)$ and $(\mathcal{M}_2, w_2)$ also satisfy the (4) and (5) condition of the definition of $\langle gsb \rangle$-bisimilarity. But these models are not $\langle gsb \rangle$-bisimilar. To see this, assume on the contrary that $(\mathcal{M}_1, w_1)$ and $(\mathcal{M}_2, w_2)$ are indeed $\langle gsb \rangle$-bisimilar. Then, $(\mathcal{M}_1, u_1)$ and $(\mathcal{M}_2, u_2)$ must be $\langle gsb \rangle$-bisimilar as well. But if we delete e_1 from $\mathcal{M}_1$, there is no edge in $\mathcal{M}_2$ such that $(\mathcal{M}_1, u_1)$ and $(\mathcal{M}_2, u_2)$ are even bisimilar.

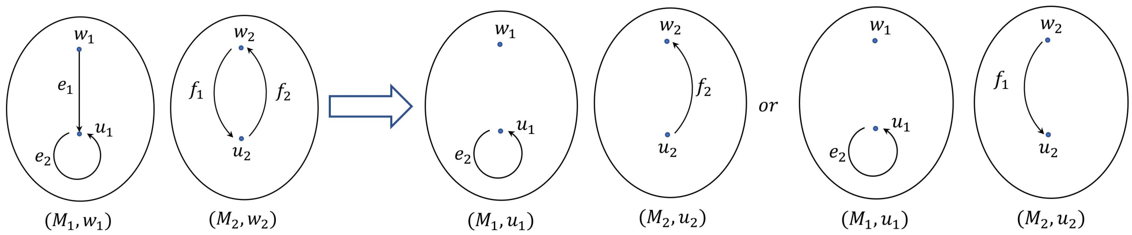


Figure 5.9 Counterexample for $\langle gsb \rangle$-bisimilar

The Algorithm 5.1 takes input and passes the information, according to the value of $state$, to different algorithms for different checks. The first three conditions in the definition of $\langle up \rangle$-bisimilarity remain the same, and hence Algorithms 5.4 and 5.14 are always called. The fourth and fifth conditions change, and accordingly, different algorithms are called.

Algorithm 5.1 Algorithm to check whether two models are bisimilar in some model changing modal logic

Require: $((W_1, R_1, V_1), w_1), ((W_2, R_2, V_2), w_2), L, L', state$

function GEN-BISIMILAR($((W_1, R_1, V_1), w_1), ((W_2, R_2, V_2), w_2), L, L', state$)
  if (state = $\langle gsb \rangle$-bisimilar OR $\langle gsw \rangle$-bisimilar) **then**
    if (checkEdges($((W_1, R_1, V_1), w_1), ((W_2, R_2, V_2), w_2)$) = No) **then**
      return No
    end if
  end if
  if (state = $\langle de \rangle$-bisimilar OR $\langle gbr \rangle$-bisimilar) **then**
    if (checkNodes($((W_1, R_1, V_1), w_1), ((W_2, R_2, V_2), w_2)$) = No) **then**
      return No
    end if
  end if
  if (checkAtomicPropositionInCurrentWorlds($(((W_1, R_1, V_1), w_1), ((W_2, R_2, V_2), w_2)$) = No) **then**
    return No
  end if
  if (state = $\langle sb \rangle$-bisimilar) **then**
    if (checkEdgeDeletion($((W_1, R_1, V_1), w_1), ((W_2, R_2, V_2), w_2)$) = No) **then**
      return No
    end if

end if
if (state = $\langle gsb \rangle$-bisimilar) **then**
if (checkGeneralizedEdgeDeletion$(((W_1, R_1, V_1), w_1), ((W_2, R_2, V_2), w_2))$ = No) **then**
return No
end if
end if
if (state = $\langle sw \rangle$-bisimilar) **then**
if (checkSwap$(((W_1, R_1, V_1), w_1), ((W_2, R_2, V_2), w_2), L')$ = No) **then**
return No
end if
end if
if (state = $\langle gsw \rangle$-bisimilar) **then**
if (checkGeneralizedSwap$(((W_1, R_1, V_1), w_1), ((W_2, R_2, V_2), w_2), L')$ = No) **then**
return No
end if
end if
if (state = $\langle br \rangle$-bisimilar) **then**
if (checkBridge$(((W_1, R_1, V_1), w_1), ((W_2, R_2, V_2), w_2))$ = No) **then**
return No
end if
end if
if (state = $\langle gbr \rangle$-bisimilar) **then**
if (checkGeneralizedBridge$(((W_1, R_1, V_1), w_1), ((W_2, R_2, V_2), w_2))$ = No) **then**
return No
end if
end if
if (state = $\langle de \rangle$-bisimilar) **then**
if (checkNodeDeletion$(((W_1, R_1, V_1), w_1), ((W_2, R_2, V_2), w_2))$ = No) **then**
return No
end if

end if
if (state = $\langle ch \rangle$-bisimilar) **then**
if (checkValuationChange$(((W_1, R_1, V_1), w_1), ((W_2, R_2, V_2), w_2), L')$ = No) **then**
return No
end if
end if
if (checkSuccessors$(((W_1, R_1, V_1), w_1), ((W_2, R_2, V_2), w_2), L, state)$ = No) **then**
return No
end if
return Yes
end function

Now we will give a brief explanation of the different functions followed by the functions themselves. The function *checkEdges* returns No if the input models do not have equal number of edges. This check makes the proof in Section 5.3.2 a little easier.

Algorithm 5.2 checkEdges

if $|R_1| \neq |R_2|$ **then**
return No
end if
return Yes

The function *checkNodes* returns No if the input models do not have equal number of nodes.

Algorithm 5.3 checkNodes

if $|W_1| \neq |W_2|$ **then**
return No
end if
return Yes

The function *checkAtomicPropositionInCurrentWorlds* returns No if the nodes at which the input models are pointed, do not satisfy same set of atomic propositions. To do this, the algorithm checks that $w_1 \in V_1(p)$ if and only if $w_2 \in V_2(p)$, for all atomic propositions p.

Algorithm 5.4 checkAtomicPropositionInCurrentWorlds

for atomic propositions p **do**
  if $(((w_1 \in V_1(p))$ & $(w_2 \notin V_2(p)))$ OR $((w_1 \notin V_1(p))$ & $(w_2 \in V_2(p))))$ **then**
    return No
  end if
end for
return Yes

The function *checkEdgeDeletion* returns No, if there is no pair of models related to input models by relation r_{sb} that are $\langle sb \rangle$-bisimilar. To do this, the algorithm recursively calls Algorithm 5.1 on new models after deleting one edge from each (pointing from w_1). If all such instances of Algorithm 5.1 return No, then by recursion, there is no pair of edges that can be deleted from given models that satisfies the conditions (4) and (5) in $\langle sb \rangle$-bisimilar definition.

Algorithm 5.5 checkEdgeDeletion

for $u_1 \in W_1$ **do**
Found = 0
  if $(w_1, u_1) \in R_1$ **then**
    for $u_2 \in W_2$ **do**
      if $(w_2, u_2) \in R_2$ **then**
        if gen-Bisimilar($((W_1, R_1 \backslash \{(w_1, u_1)\}, V_1), u_1)$,$((W_2, R_2 \backslash \{(w_2, u_2)\}$,
$V_2), u_2), \varnothing, \varnothing$, $\langle sb \rangle$-bisimilar) = Yes **then**
          Found++
          break
        end if
      end if
    end for
    if Found = 0 **then**
      return No
    end if
  end if
end for
for $u_2 \in W_2$ **do**
Found = 0;

    if $(w_2, u_2) \in R_2$ **then**
        for $u_1 \in W_1$ **do**
            if $(w_2, u_2) \in R_2$ **then**
                if gen-Bisimilar$(((W_1, R_1\backslash\{(w_1,u_1)\}, V_1), u_1), ((W_2, R_2\backslash\{(w_2,u_2)\},$
$V_2), u_2), \varnothing, \varnothing,$ $\langle sb \rangle$-bisimilar) = Yes **then**
                    Found++
                    break
                end if
            end if
        end for
        if Found = 0 **then**
            return No
        end if
    end if
end for
return Yes

The function *checkGeneralizedEdgeDeletion* returns No, if there is no pair of models related to the input models by relation r_{gsb} that are $\langle gsb \rangle$-bisimilar. This algorithm works very similar to the previous one with the only difference being as follows - instead of deleting edges pointed from w_1, it runs over all edges. This is in accordance with the difference in definitions of $\langle sb \rangle$-bisimilarity and $\langle gsb \rangle$-bisimilarity.

Algorithm 5.6 checkGeneralizedEdgeDeletion

for $e_1 \in R_1$ **do**
Found = 0
    for $e_2 \in R_2$ **do**
        if gen-Bisimilar$(((W_1, R_1\backslash\{e_1\}, V_1), w_1), ((W_2, R_2\backslash\{e_2\}, V_2), w_2), \varnothing, \varnothing,$
$\langle gsb \rangle$-bisimilar) = Yes **then**
            Found++
            break
        end if
    end for
    if Found = 0 **then**
        return No

    end if
end for
for $e_2 \in R_2$ **do**
Found = 0
    for $e_1 \in R_1$ **do**
        if gen-Bisimilar$(((W_1, R_1 \backslash \{e_1\}, V_1), w_1), ((W_2, R_2 \backslash \{e_2\}, V_2), w_2), \varnothing, \varnothing,$ $\langle gsb \rangle$-bisimilar) = Yes **then**
            Found++
            break
        end if
    end for
    if Found = 0 **then**
        return No
    end if
end for
return Yes

The function *checkSwap* returns No, if there is no pair of models related to the input models by relation r_{sw} that are $\langle sw \rangle$-bisimilar. To do this, it runs over all the edges from w_1 and w_2, swaps their directions, and calls Algorithm 5.1 recursively.

Algorithm 5.7 checkSwap

for $u_1 \in W_1$ **do**
Found = 0
    if $(w_1, u_1) \in R_1 \backslash L'$ **then**
        for $u_2 \in W_2$ **do**
            if $(w_2, u_2) \in R_2 \backslash L'$ **then**
                if gen-Bisimilar$(((W_1, R_1 \cup \{(u_1, w_1)\} \backslash \{(w_1, u_1)\}, V_1), u_1)$,$((W_2,$ $R_2 \cup \{(u_2, w_2)\} \backslash \{(w_2, u_2)\}, V_2), u_2), \varnothing, L' \cup \{(w_1, u_1), (w_2, u_2)\}, \langle sw \rangle$-bisimilar) = Yes **then**
                    Found++
                    break
                end if
            end if
        end for

if Found = 0 **then**
return No
end if
end if
end for
for $u_2 \in W_2$ **do**
Found = 0
if $(w_2, u_2) \in R_2 \backslash L'$ **then**
for $u_1 \in W_1$ **do**
if $(w_1, u_1) \in R_1 \backslash L'$ **then**
if gen-Bisimilar$(((W_1, R_1 \cup \{(u_1, w_1)\} \backslash \{(w_1, u_1)\}, V_1), u_1), ((W_2, R_2 \cup \{(u_2, w_2)\} \backslash \{(w_2, u_2)\}, V_2), u_2), \varnothing, L' \cup \{(w_1, u_1), (w_2, u_2)\}, \langle sw \rangle$-bisimilar) = Yes **then**
Found++
break
end if
end if
end for
if Found = 0 **then**
return No
end if
end if
end for
return Yes

The function *checkGeneralizedSwap* returns No, if there is no pair of models related to the input models by relation r_{gsw} that are $\langle gsw \rangle$-bisimilar. Again, this is very similar to the previous algorithm with the only difference being that it now runs over all edges.

Algorithm 5.8 checkGeneralizedSwap

for $(u_1, u_2) \in R_1 \backslash L'$ **do**
Found = 0;
for $(v_1, v_2) \in R_2 \backslash L'$ **do**
if gen-Bisimilar$(((W_1, R_1 \cup \{(u_2, u_1)\} \backslash \{(u_1, u_2)\}, V_1), w_1), ((W_2, R_2 \cup \{(v_2, v_1)\} \backslash \{(v_1, v_2)\}, V_2), w_2), \varnothing, L' \cup \{(u_1, u_2), (v_1, v_2)\}, \langle gsw \rangle$-bisimilar) = Yes

then
Found++
break
end if
end for
if Found = 0 **then**
return No
end if
end for
for $(v_1, v_2) \in R_2 \backslash L'$ **do**
Found = 0;
for $(u_1, u_2) \in R_1 \backslash L'$ **do**
if gen-Bisimilar$(((W_1, R_1 \cup \{(u_2, u_1)\} \backslash \{(u_1, u_2)\}, V_1), w_1), ((W_2, R_2 \cup \{(v_2, v_1)\} \backslash \{(v_1, v_2)\}, V_2), w_2), \varnothing, L' \cup \{(u_1, u_2), (v_1, v_2)\}, \langle gsw \rangle$-bisimilar) = Yes **then**
Found++
break
end if
end for
if Found = 0 **then**
return No
end if
end for
return Yes

The function *checkBridge* returns No, if there is no pair of models related to the input models by relation r_{br} that are $\langle br \rangle$-bisimilar. To do this, the algorithm adds new edges from w_1 and w_2 and calls Algorithm 5.1 recursively.

Algorithm 5.9 checkBridge

for $u_1 \in W_1$ **do**
Found = 0
for $u_2 \in W_2$ **do**
if $(((w_1, u_1) \notin R_1)$ & $((w_2, u_2) \notin R_2)$ **then**
if gen-Bisimilar$(((W_1, R_1 \cup \{(w_1, u_1)\}, V_1), u_1), ((W_2, R_2 \cup \{(w_2, u_2)\},$

$V_2), u_2), \varnothing, \varnothing, \langle br\rangle$-bisimilar) = Yes **then**
            Found++
            break
        end if
    end if
  end for
  if Found = 0 **then**
    return No
  end if
end for
for $u_2 \in W_2$ **do**
Found = 0
  for $u_1 \in W_1$ **do**
    if $(((w_1, u_1) \notin R_1)$ & $((w_2, u_2) \notin R_2)$ **then**
        if gen-Bisimilar$(((W_1, R_1 \cup \{(w_1, u_1)\}, V_1), u_1)$,$((W_2, R_2 \cup \{(w_2, u_2)\},$
$V_2), u_2), \varnothing, \varnothing, \langle br\rangle$-bisimilar) = Yes **then**
            Found++
            break
        end if
    end if
  end for
  if Found = 0 **then**
    return No
  end if
end for
return Yes

The function *checkGeneralizedBridge* returns No, if there is no pair of models related to the input models by relation r_{gbr} that are $\langle gbr\rangle$-bisimilar. This algorithm adds one new edge to both the models and calls Algorithm 5.1 recursively.

Algorithm 5.10 checkGeneralizedBridge

for $(u_1, u_2) \in W_1 \times W_1$ **do**
Found = 0
  for $(v_1, v_2) \in W_2 \times W_2$ **do**

if $(((u_1,u_2) \notin R_1)$ & $((v_1,v_2) \notin R_2)$ **then**
if gen-Bisimilar$(((W_1, R_1 \cup \{(u_1,u_2)\}, V_1), w_1)$,$((W_2, R_2 \cup \{(v_1,v_2)\},$
$V_2), w_2), \varnothing, \varnothing, \langle gbr \rangle$-bisimilar) = Yes **then**
Found++
break
end if
end if
end for
if Found = 0 **then**
return No
end if
end for
for $(v_1,v_2) \in W_2 \times W_2$ **do**
Found = 0
for $(u_1,u_2) \in W_1 \times W_1$ **do**
if $(((u_1,u_2) \notin R_1)$ & $((v_1,v_2) \notin R_2)$ **then**
if gen-Bisimilar$(((W_1, R_1 \cup \{(u_1,u_2)\}, V_1), w_1)$,$((W_2, R_2 \cup \{(v_1,v_2)\},$
$V_2), w_2), \varnothing, \varnothing, \langle gbr \rangle$-bisimilar) = Yes **then**
Found++
break
end if
end if
end for
if Found = 0 **then**
return No
end if
end for
return Yes

The previous algorithms were the implementations of the model-changing step for the six relation-changing logics that we have described. We now move on to implement the model-changing step for domain-changing logic and valuation-changing logic. The next algorithm is a precursor to the case of domain-changing logic. Specifically, it computes the new model after a node has been deleted from it.

Algorithm 5.11 algorithm to compute new relational model after point deletion

Require: $((W, R, V), w), u$

function SUCCESSOR$(((W, R, V), w), u)$
$W' = W \backslash \{u\}$
$R' = R \backslash (\{(u, v) \in W \mid v \in R\} \cup \{(v, u) \in R \mid v \in W\})$
    for $p \in \mathsf{Prop}$ **do**
        $V'(p) = V(p) \cap W'$
    end for
    return $((W', R', V'), w)$
end function

The function *checkNodeDeletion* returns No, if there is no pair of models related to the input models by relation r_{de} that are $\langle de \rangle$-bisimilar. To do this, the algorithm makes a recursive call to $gen\text{-}Bisimilar$ with new pair of models that have one node deleted in each.

Algorithm 5.12 checkNodeDeletion

for $u_1 \in W_1$ **do**
Found = 0
    for $u_2 \in W_2$ **do**
        if $(u_1 \neq w_1)$ & $(u_2 \neq w_2)$ **then**
            if gen-Bisimilar$((successor((W_1, R_1, V_1), u_1), w_1)$,$(successor((W_2,$
$R_2, V_2), u_2), w_2), \varnothing, \varnothing, \langle de \rangle$-bisimilar) = Yes **then**
                Found++
                break
            end if
        end if
    end for
    if Found = 0 & $(u_1 \neq w_1)$ **then**
        return No
    end if
end for
for $u_2 \in W_2$ **do**
Found = 0
    for $u_1 \in W_1$ **do**
        if $(u_1 \neq w_1)$ & $(u_2 \neq w_2)$ **then**

if gen-Bisimilar$((successor((W_1, R_1, V_1), u_1), w_1)$,$(successor((W_2, R_2, V_2), u_2), w_2), \varnothing, \varnothing, \langle de \rangle$-bisimilar) = Yes **then**
Found++
break
end if
end if
end for
if Found = 0 & $(u_2 \neq w_2)$ **then**
return No
end if
end for
return Yes

The function *checkValuationChange* returns No, if there is no pair of models related to the input models by relation r_{ch} that are $\langle ch \rangle$-bisimilar. To do this, the function changes valuation of the current node and then calls Algorithm 5.14 recursively.

Algorithm 5.13 checkValuationChange

if $(w_1, w_2) \notin L'$ **then**
for $A \subset \mathcal{P}$ **do**
$\tilde{V}_1(w_1) = A$
$\tilde{V}_2(w_2) = A$
for $w \in W_1 \backslash \{w_1\}$ **do**
$\tilde{V}_1(w) = V_1(w)$
end for
for $w \in W_2 \backslash \{w_2\}$ **do**
$\tilde{V}_2(w) = V_2(w)$
end for
if gen-Bisimilar$(((W_1, R_1, \tilde{V}_1), w_1)$,$((W_2, R_2, \tilde{V}_2), w_2), \varnothing, L' \cup \{(w_1, w_2)\}, \langle ch \rangle$-bisimilar) = No **then**
return No
end if
end for
end if
return Yes

The following algorithm checks for the existence of successors to the node, at which the input models are pointed, for the specific bisimulation according to the state. Specifically, it checks if conditions (2) and (3) in the definition of $\langle up \rangle$-bisimilarity are true. To do this, it changes the node where the models are pointed to one of the successors of initial nodes at which the models were pointed, and then makes a recursive call to $gen\text{-}Bisimilar$.

Algorithm 5.14 checkSuccessors

if $(w_1, w_2) \notin L$ **then**
 for $u_1 \in W_1$ **do**
  Found = 0
  for $u_2 \in W_2$ **do**
   if $((w_1 R_1 u_1)$ & $(w_2 R_2 u_2))$ **then**
    if $((u_1, u_2) \notin L)$ **then**
     if gen-Bisimilar$(((W_1, R_1, V_1), u_1), ((W_2, R_2, V_2), u_2), L \cup \{(w_1, w_2)\}, \varnothing, state)$ = Yes **then**
      Found++
     end if
    else
     Found++
    end if
   end if
  end for
  if (Found = 0) & $(w_1 R_1 u_1)$ **then**
   return No
  end if
 end for
 for $u_2 \in W_2$ **do**
  Found = 0
  for $u_1 \in W_1$ **do**
   if $((w_1 R_1 u_1)$ & $(w_2 R_2 u_2))$ **then**
    if $((u_1, u_2) \notin L)$ **then**
     if gen-Bisimilar$(((W_1, R_1, V_1), u_1), ((W_2, R_2, V_2), u_2), L \cup \{(w_1, w_2)\}, \varnothing, state)$ = Yes **then**
      Found++

```
                    end if
                else
                    Found++
                end if
            end if
        end for
        if (Found = 0) & (w2 R2 u2) then
            return No
        end if
    end for
end if
return Yes
```

With these algorithms out of the way, we are now ready to show the correctness of these algorithms. We will show a couple of cases in all details. The rest follow similarly.

5.3.2 On $\langle gsb\rangle$-bisimulation

We will now give a detailed proof that the algorithm works when $state = \langle gsb\rangle$-$bisimilar$. Before going into the main proof, we have the following lemma.

Lemma 5.1: If $((W_1, R_1, V_1), w_1)\underline{\leftrightarrow}_s((W_2, R_2, V_2), w_2)$ and $|R_1|$ and $|R_2|$ are finite, then $|R_1| = |R_2|$.

Proof Suppose on the contrary, $|R_1| \neq |R_2|$. Without loss of generality, we assume $|R_1| < |R_2|$, and prove by induction on $n = |R_1|$.

Base case: $n = 0$. By assumption $|R_1| = 0$ and $|R_2| > 0$. So $\exists e \in R_2$. Now, since $((W_1, R_1, V_1), w_1)\underline{\leftrightarrow}_s((W_2, R_2, V_2), w_2)$, they satisfy condition (5) of the definition of $\langle gsb\rangle$−bisimilarity. Therefore, there must exist an edge $f \in R_1$ such that $((W_1, R_1\backslash\{f\}, V_1), w_1)\underline{\leftrightarrow}_s((W_2, R_2\backslash\{e\}, V_2), w_2)$. But, since $|R_1| = 0$, no such f can exist. Contradiction.

Induction hypothesis: Suppose the claim holds good for $n \leq k$, *i.e.*, $|R_1| = |R_2|$, whenever $|R_1| \leq k$.

Induction step: $n = k + 1$ Suppose $\min(|R_1|, |R_2|) = |R_1| = k + 1$. Let $e_1 \in R_1$ be any edge. Since, $((W_1, R_1, V_1), w_1)\underline{\leftrightarrow}_s((W_2, R_2, V_2), w_2)$, they satisfy condition (4) in the definition of $\langle gsb\rangle$−bisimilarity, so $\exists e_2 \in R_2$ such that $((W_1, R_1\backslash\{e_1\}, V_1), w_1)$

$\underline{\leftrightarrow}_s((W_2, R_2\backslash\{e_2\}, V_2), w_2)$. But then by induction hypothesis, we have $|R_1\backslash\{e_1\}| = R_2\backslash\{e_2\}| \implies |R_1| - 1 = |R_2| - 1 \implies |R_1| = |R_2|$.

This completes the proof. ∎

We will now prove the main theorem below which basically shows the correctness of the algorithm corresponding to the $\langle gsb\rangle$-bisimulation.

Theorem 5.1: Given two models $(\mathcal{M}_1, w_1)$ and $(\mathcal{M}_2, w_2)$, where $\mathcal{M}_1 = (W_1, R_1, V_1)$, $\mathcal{M}_2 = (W_2, R_2, V_2)$, $w_1 \in W_1$ and $w_2 \in W_2$; $(\mathcal{M}_1, w_1)\underline{\leftrightarrow}_s(\mathcal{M}_2, w_2)$ iff the function gen-Bisimilar($(\mathcal{M}_1, w_1), (\mathcal{M}_2, w_2), \varnothing, \varnothing, \langle gsb\rangle\text{-}bisimilar$) returns yes. Here, by $(\mathcal{M}_1, w_1)\underline{\leftrightarrow}_s(\mathcal{M}_2, w_2)$ we will denote that $(\mathcal{M}_1, w_1)$ and $(\mathcal{M}_2, w_2)$ are $\langle gsb\rangle\text{-}bisimilar$.

Proof Suppose $\mathcal{M}_1$ and $\mathcal{M}_2$ have different number of edges, then the function returns No at line 2 in algorithm 2, and $(\mathcal{M}_1, w_1)\underline{\not\leftrightarrow}_s(\mathcal{M}_2, w_2)$. So, let us consider that both models have equal number of edges (say n). We prove by induction on n:

> **Base case:** $n = 0$.
To prove $(\mathcal{M}_1, w_1)\underline{\leftrightarrow}_s(\mathcal{M}_2, w_2)$ iff the function returns yes when $R_1 = \varnothing = R_2$. We will first prove, by contrapositivity, that if the function returns yes, then $(\mathcal{M}_1, w_1)\underline{\leftrightarrow}_s(\mathcal{M}_2, w_2)$.

> > Suppose $(\mathcal{M}_1, w_1)\underline{\not\leftrightarrow}_s(\mathcal{M}_2, w_2)$. Then they violate one of the five conditions in the definition of $\langle gsb\rangle$-bisimilarity (in Section 5.2.2).

> > > Suppose they violate condition (1). Then there is some atomic proposition p such that either $(\mathcal{M}_1, w_1) \vDash p$ and $(\mathcal{M}_2, w_2) \nvDash p$; or $(\mathcal{M}_1, w_1) \nvDash p$ and $(\mathcal{M}_2, w_2) \vDash p$. From truth definitions, we have $w_1 \in V_1(p)$ but $w_2 \notin V_2(p)$; or $w_1 \notin V_1(p)$ but $w_2 \in V_2(p)$. In this case, the function returns No at line 3 in Algorithm 5.4.

> > > Suppose they violate condition (2). Then, there is a successor v_1 of w_1, i.e. $\exists v_1 \in W_1$ such that $w_1 R_1 v_1$, but $\forall v_2$ such that $w_2 R_2 v_2$, we do not have $(\mathcal{M}_1, v_1)\underline{\leftrightarrow}_s(\mathcal{M}_2, v_2)$. But since $n = 0$, $w_1 R_1 v_1$ does not hold for any v_1 as $R_1 = \varnothing$. Therefore, condition (2) in the definition of $\langle gsb\rangle$-bisimilarity cannot be violated in this case.

> > > Suppose that they violate condition (3). Again by similar argument as last point, we can not have $v_2 R_2 w_2$ and hence condition (3) can not be violated when $n = 0$.

> > > Suppose they violate condition (4). Then there is an edge $e_1 \in R_1$ such that for any edge $e_2 \in R_2$, it is not the case that $(\mathcal{M}_1\backslash\{e_1\}, w_1)\underline{\leftrightarrow}_s(\mathcal{M}_2\backslash\{e_2\}, w_2)$. But again since $n = 0$, $R_1 = \varnothing$, hence no such e_1 exists. So this case cannot arise.

> > > By similar argument as in previous point, the models cannot violate condition (5).

> > Now we will prove the other side. Therefore, suppose that the function returns No, Then one of the following cases occur:

> > > The function returns No at line number 3 in Algorithm 5.4. This can only happen when the If condition at line 2 in Algorithm 5.4 is true. Therefore, there exists an atomic proposition p such that, $w_1 \in V_1(p)$ but $w_2 \notin V_2(p)$; or $w_1 \notin V_1(p)$ but $w_2 \in V_2(p)$. From truth definitions, we have either $(\mathcal{M}_1, w_1) \vDash p$ and $(\mathcal{M}_2, w_2) \nvDash p$; or $(\mathcal{M}_1, w_1) \nvDash p$ and $(\mathcal{M}_2, w_2) \vDash p$. But then $(\mathcal{M}_1, w_1) \not\underline{\leftrightarrow}_s (\mathcal{M}_2, w_2)$ as they violate condition (1) of the definition of $\langle gsb \rangle$-bisimilarity.

> > > The function returns No at line number 10 or 22 in Algorithm 5.6. But since $R_1 = \emptyset$ and $R_2 = \emptyset$, This can not happen as the function *checkGeneralizedEdgeDeletion* will not execute any for loop.

> > > Suppose the function returns No from line 16 or 33 in Algorithm 5.14. Again, this can not happen because w_1 and w_2 do not have any successors.
This completes both sides of the base case.

> **Induction Hypothesis 1:** Suppose the theorem holds for $n \leq k$. That is, $(\mathcal{M}_1, w_1) \underline{\leftrightarrow}_s (\mathcal{M}_2, w_2)$ iff gen-Bisimilar($(\mathcal{M}_1, w_1), (\mathcal{M}_2, w_2), \emptyset, \emptyset, \langle gsb \rangle$-$bisimilar$) returns yes when $|R_1| = |R_2| \leq k$.

> **Induction Step:** Let $n = k + 1$
We will first prove that if $(\mathcal{M}_1, w_1) \underline{\leftrightarrow}_s (\mathcal{M}_2, w_2)$ then the function returns yes. Again we will prove this by contrapositivity.

> > Suppose the function returns No in Algorithm 5.1. Then it executes one of the 4 return No statements, that are reachable when $state = \langle gsb \rangle - bisimilar$. But it can not return No at line 2 in Algorithm 5.12, as we have assumed $|R_1| = |R_2|$. So the following cases can occur:

> > > The function returns No at line number 13 in Algorithm 5.1. This can only happen when the If condition at line 12 in algorithm 5.1 is true. But then, by argument similar to that in base case, $(\mathcal{M}_1, w_1) \not\underline{\leftrightarrow}_s (\mathcal{M}_2, w_2)$ as they violate condition (1) of the definition of $\langle gsb \rangle$-bisimilarity.

> > > The function returns No at line number 22 in Algorithm 5.1. Then condition in line 21 is true. This happens if the function returns No at line 10 or 22. If the function returns No at line 10 in function *checkGeneralized-EdgeDeletion*, there is an $e_1 \in R_1$ such that for all $e_2 \in R_2$, we have gen-bisimilar($(\mathcal{M}_1 \setminus \{e_1\}, w_1), (\mathcal{M}_2 \setminus \{e_2\}, w_2), \emptyset, \emptyset, \langle gsb \rangle$-$bisimilar$) returns No. But

the model $\mathcal{M}_1' = \mathcal{M}_1 \backslash \{e_1\}$ and $\mathcal{M}_2' = \mathcal{M}_2 \backslash \{e_2\}$ have k edges. Therefore, by induction hypothesis 1, $(\mathcal{M}_1 \backslash \{e_1\}, w_1) \not\rightleftarrows_s (\mathcal{M}_2 \backslash \{e_2\}, w_2)$ for all $e_2 \in R_2$. This is a violation to condition (4) in the definition of $\langle gsb \rangle$-bisimilarity. Therefore, $(\mathcal{M}_1, w_1) \not\rightleftarrows_s (\mathcal{M}_2, w_2)$. The case is similar if No is returned at line 22 in function *checkGeneralizedEdgeDeletion*.

> > > The function returns No at line 56 in Algorithm 5.1. Then the condition(s) at line 15 or 32 in Algorithm 5.14 in the function *checkSuccessors* is true for some $u_1 \in W_1$ or $u_2 \in W_2$ respectively. Suppose the function returns No at line 16. Therefore, the following cases arise (the following line numbers are in function *checkSuccessors*):

> > > > For a successor u_1 of w_1, condition at line 5 is false for all $u_2 \in W_2$, *i.e.*, $w_2 R_2 u_2$ is not true for any $u_2 \in W_2$. This is a violation of condition (2) in the definition of $\langle gsb \rangle$-bisimilarity and hence $(\mathcal{M}_1, w_1) \not\rightleftarrows_s (\mathcal{M}_2, w_2)$

> > > > Conditions at line 5 and 6 are true, but condition at line 7 is false, *i.e.*, $\exists u_1 \in W_1$ such that $w_1 R_1 u_1$, $\forall u_2 \in R_2$ such that $w_2 R_2 u_2$ and L is such that $(u_1, u_2) \notin L$ (and $(w_1, w_2) \notin L$ because line 7 can be executed only if condition in line 1 is true); we get gen-Bisimilar$((\mathcal{M}_1, u_1), (\mathcal{M}_2, u_2), L \cup \{(w_1, w_2)\}, \emptyset, \langle gsb \rangle\text{-}bisimilar)$ returns No.

To prove:$(\mathcal{M}_1, w_1) \not\rightleftarrows_s (\mathcal{M}_2, w_2)$.

Proof by induction on $m = |W_1 \times W_2| - |L \cup \{(w_1, w_2)\}|$

> > > > > **Base case:** $|W_1 \times W_2| = |L \cup \{(w_1, w_2)\}|$

We need to prove that if $\exists u_1 \in W_1$ such that $w_1 R_1 u_1$, $\forall u_2 \in R_2$ such that $w_2 R_2 u_2$ and L is such that $(u_1, u_2) \notin L$ (and $(w_1, w_2) \notin L$ because line 5 can be executed only if the condition in line 1 is true) and $|W_1 \times W_2| - |L \cup \{(w_1, w_2)\}| = 0$; and gen-Bisimilar$((\mathcal{M}_1, u_1), (\mathcal{M}_2, u_2), L \cup \{(w_1, w_2)\}, \emptyset, \langle gsb \rangle\text{-}bisimilar)$ returns No, then $(\mathcal{M}_1, w_1) \not\rightleftarrows_s (\mathcal{M}_2, w_2)$.

But since $|W_1 \times W_2| - |L \cup \{(w_1, w_2)\}| = 0$, we have $(u_1, u_2) \in L \cup \{(w_1, w_2)\}$. This is in contradiction with the condition in line 6 being true. So the antecedent is false, and hence the base case is true vacuously.

> > > > > **Induction Hypothesis 2:** Suppose the claim holds for $m \leq l$, *i.e.*,

Suppose whenever $\exists u_1 \in W_1$ such that $w_1 R_1 u_1$, $\forall u_2 \in R_2$ such that $w_2 R_2 u_2$ and L is such that $(u_1, u_2) \notin L$ (and $(w_1, w_2) \notin L$ because line 7 can be executed only if the condition in line 1 is true) and $|W_1 \times W_2| - |L \cup \{(w_1, w_2)\}| \leq l$; and gen-Bisimilar$((\mathcal{M}_1, u_1), (\mathcal{M}_2, u_2), L \cup \{(w_1, w_2)\}, \emptyset, \langle gsb \rangle\text{-}bisimilar)$ returns No, then $(\mathcal{M}_1, w_1) \underline{\not\leftrightarrow}_s (\mathcal{M}_2, w_2)$

> > > > > **Induction step:** Suppose $m = l + 1$.

In this case, suppose the condition in line 6 is true, and the condition in line 7 is false. Therefore, we have $\exists u_1 \in W_1$ such that $w_1 R_1 u_1$, $\forall u_2 \in R_2$ such that $w_2 R_2 u_2$ and L is such that $(u_1, u_2) \notin L$ (and $(w_1, w_2) \notin L$ because line 6 can be executed only if condition in line 1 is true) and $|W_1 \times W_2| - |L \cup \{(w_1, w_2)\}| = l + 1$; and gen-Bisimilar$((\mathcal{M}_1, u_1), (\mathcal{M}_2, u_2), L \cup \{(w_1, w_2)\}, \varnothing, \langle gsb\rangle\text{-}bisimilar)$ returns No. Now, gen-Bisimilar$((\mathcal{M}_1, u_1), (\mathcal{M}_2, u_2), L \cup \{(w_1, w_2)\}, \varnothing, \langle gsb\rangle\text{-}bisimilar)$ can return No either at one of 6 , reachable return No statements in gen-bisimilar. If it returns No at first 4, then by above cases, we have already proved that $(\mathcal{M}_1, u_1) \underline{\not\leftrightarrow}_s (\mathcal{M}_2, u_2)$ because they violate conditions (1) or (4) or (5) in the definition of $\langle gsb\rangle$-bisimilarity. Suppose it returns No at line 56 in gen-bisimilar, then the function *checkSuccessors* returns No at line 16 or 33. Assume it returns No at line 16. Then if condition at line 5 is always false, then $(\mathcal{M}_1, u_1) \underline{\not\leftrightarrow}_s (\mathcal{M}_2, u_2)$ because they violate condition (2) of definition of $\langle gsb\rangle$-bisimilarity. So suppose condition at line 6 is true but at line 7 is false. Therefore, $\exists v_1 \in W_1$ such that $u_1 R_1 v_1$ and $\forall v_2 \in W_2$ such that $u_2 R_2 v_2$, L is such that $(v_1, v_2) \notin L \cup \{(w_1, w_2)\}$, we have gen-Bisimilar$((\mathcal{M}_1, v_1), (\mathcal{M}_2, v_2), L \cup \{(w_1, w_2), (u_1, u_2)\}, \varnothing, \langle gsb\rangle\text{-}bisimilar)$ returns No. Now by induction hypothesis 2, $(\mathcal{M}_1, v_1) \underline{\not\leftrightarrow}_s (\mathcal{M}_2, v_2)$ which implies $(\mathcal{M}_1, u_1) \underline{\not\leftrightarrow}_s (\mathcal{M}_2, u_2)$ and hence $(\mathcal{M}_1, w_1) \underline{\not\leftrightarrow}_s (\mathcal{M}_2, w_2)$.

> > > The function returns No at line 33 in Algorithm 5.14, then by argument similar to last case, $(\mathcal{M}_1, w_1) \underline{\not\leftrightarrow}_s (\mathcal{M}_2, w_2)$. We will now prove the remaining side by contrapositivity.

> Suppose $(\mathcal{M}_1, w_1) \underline{\not\leftrightarrow}_s (\mathcal{M}_2, w_2)$. Then these models must violate one of the five conditions in the definition of $\langle gsb\rangle$-bisimilarity.

> > Suppose they violate condition (1). There is some atomic proposition p such that either $(\mathcal{M}_1, w_1) \vDash p$ and $(\mathcal{M}_2, w_2) \nvDash p$; or $(\mathcal{M}_1, w_1) \nvDash p$ and $(\mathcal{M}_2, w_2) \vDash p$. From truth definitions, we have $w_1 \in V_1(p)$ but $w_2 \notin V_2(p)$; or $w_1 \notin V_1(p)$ but $w_2 \in V_2(p)$. In this case the function returns No in line 13.

> > Suppose they violate condition (4). Then there is an edge $e_1 \in R_1$ such that for any edge $e_2 \in R_2$, $(\mathcal{M}_1 \backslash \{e_1\}, w_1) \underline{\not\leftrightarrow}_s (\mathcal{M}_2 \backslash \{e_2\}, w_2)$. In this case for e_1, condition in line 4 in algorithm 6 is never true. By induction hypothesis 1, $(\mathcal{M}_1 \backslash \{e_1\}$ and

$\mathcal{M}_2\backslash\{e_2\}$ have k edges, hence we can use induction hypothesis). Therefore, return No is executed in line 10 in function *checkGeneralizedEdgeDeletion*.

> > Suppose they violate condition (5), by a similar argument as the previous case, by induction hypothesis, the function returns No.

> > Suppose they violate condition (2) and/ or (3). We prove if $(\mathcal{M}_1, w_1)\not\leftrightarrows_s(\mathcal{M}_2, w_2)$ because they violate condition (2) and/or (3), but not (1), (4) or (5) in the definition of $\langle gsb\rangle$-bisimilarity, then gen-Bisimilar((($\mathcal{M}_1, w_1$), $(\mathcal{M}_2, w_2), \emptyset, \emptyset, \langle gsb\rangle$-$bisimilar$) returns No.

Since $(\mathcal{M}_1, w_1)\not\leftrightarrows_s(\mathcal{M}_2, w_2)$ because they violate condition (2) and/or (3), therefore $\exists u_{11} \in W_1$, $w_1R_1u_{11}$, such that $\forall u_{21} \in W_2$, $w_2R_2u_{21}$, $(\mathcal{M}_1, u_{11})\not\leftrightarrows_s(\mathcal{M}_2, u_{21})$ (if condition (2) is violated); or $\exists u_{12} \in W_2$, $w_2R_2u_{12}$, such that $\forall u_{11} \in W_1$, $w_1R_1u_{11}$, $(\mathcal{M}_1, u_{11})\not\leftrightarrows_s(\mathcal{M}_2, u_{21})$. Now if $(\mathcal{M}_1, u_{11})\not\leftrightarrows_s(\mathcal{M}_2, u_{21})$ because they violate conditions (1), (4) or (5), then by previous cases, the function returns No, and we will be done. Let us pick a general such pair (v_{11}, v_{21}). So, assume $(\mathcal{M}_1, v_{11})\not\leftrightarrows_s(\mathcal{M}_2, v_{21})$ because they violate condition(s) (2) and/or (3). Therefore, again, $\exists u_{12} \in W_1$, $v_{11}R_1u_{12}$, such that $\forall u_{22} \in W_2$, $v_{12}R_2u_{22}$, $(\mathcal{M}_1, u_{12})\not\leftrightarrows_s(\mathcal{M}_2, u_{22})$ (if they violate (2)); or $\exists u_{22} \in W_2$, $v_{12}R_2u_{22}$, such that $\forall u_{12} \in W_1$, $v_{11}R_1u_{12}$, $(\mathcal{M}_1, u_{12})\not\leftrightarrows_s(\mathcal{M}_2, u_{22})$ (if they violate condition (3). Again, choose a general such pair (v_{12}, v_{22}) from above such that $v_{11}R_1v_{12}$ and $v_{21}R_2v_{22}$ and $(\mathcal{M}_1, v_{12})\not\leftrightarrows_s(\mathcal{M}_2, v_{22})$. Again, we are done if $(\mathcal{M}_1, v_{12})\not\leftrightarrows_s(\mathcal{M}_2, v_{22})$ because they violate condition (1), (4) or (5). So, again, we can assume that they violate condition (2) and /or (3). This can go on until we reach a leaf node, *i.e.*, there is some k such that exactly one of the following is true: $v_{1k}R_1v_{1(k+1)}$ for some $v_{1(k+1)} \in W_1$ or $v_{2k}R_2v_{2(k+1)}$ for some $v_{2(k+1)} \in W_2$. Again the function returns No, from function *checkGeneralizedEdgeDeletion* in both cases. The only case that remains is when there is no leaf nodes and there is some k such that $v_{1k} = v_{1l}$ or w_1 and $v_{2k} = v_{2l}$ or w_2 for some $l < k$. In this case, since v_{1i} and v_{2i} were some general node in the reachable part from w_1 and w_2, such that they do not violate condition (1), (4) or (5) in the definition of $\langle gsb\rangle$-bisimilarity, we have the following:

> > > $(\mathcal{M}_1, w_1)$ and $(\mathcal{M}_2, w_2)$ satisfy conditions (1), (4) and (5) in the definition of $\langle gsb\rangle$-bisimilarity.

> > > For every n, $\exists v_1 \in W_1$ such that $w_1R_1^nv_1$ iff $\exists v_2 \in W_2$ such that $w_2R_2^nv_2$; and $(\mathcal{M}_1, v_1)$ and $(\mathcal{M}_2, v_2)$ satisfy condition (1), (4) and (5) from the definition of

$\langle gsb \rangle$-bisimilarity. But these conditions are same as the conditions in definition of $\langle gsb \rangle$-bisimilarity. Hence, $(\mathcal{M}_1, w_1) \underline{\leftrightarrow}_s (\mathcal{M}_2, w_2)$ and the function does not return No in this case.

This completes the proof. ∎

5.3.2.1 An example

In what follows, we present an example run of the algorithm for gen-Bisimilar for $state = \langle gsb \rangle\text{-}bisimilar$. Proposition p is true in all the worlds of both models. Figure 5.10 shows all the important nodes and calls to function *checkGeneralizedEdgeDeletion*.

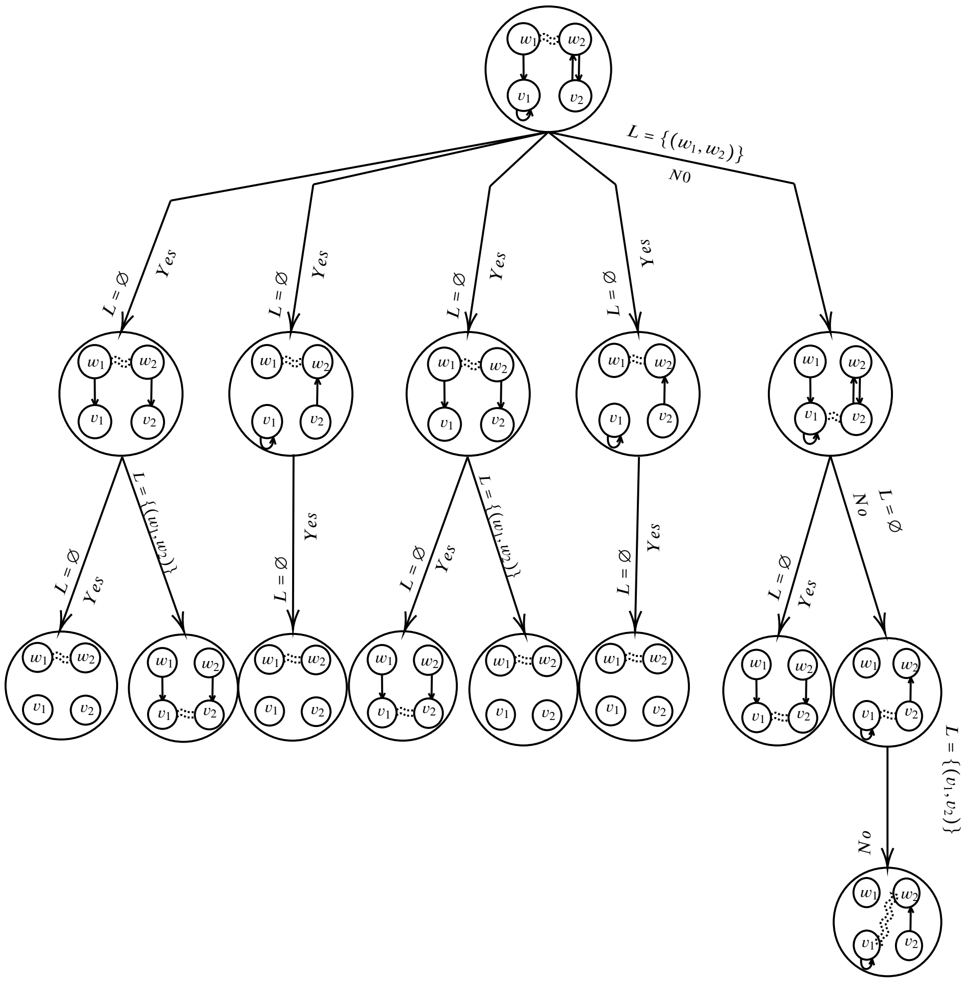


Figure 5.10 Recursive graph

In the above run of the algorithm, not all children of the root return 'Yes', and hence the input models are not $\langle gsb \rangle\text{-}bisimilar$.

5.3.2.2 On complexity

We now show that the complexity of the $\langle gsb \rangle$-bisimulation problem is in *PSPACE*.

Theorem 5.2: Function gen-Bisimilar terminates and is in **PSPACE** for $state = \langle gsb \rangle$-$bisimilar$.

Proof We will form a recursion tree in Figure 5.11 to see whether the function gen-Bisimilar terminates for $state = \langle gsb \rangle$-$bisimilar$ and analyze the space complexity of the function.

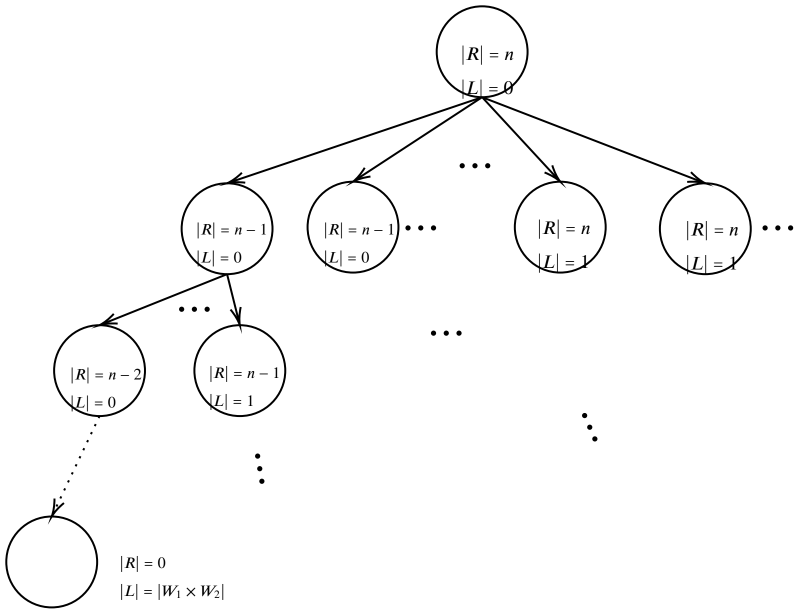


Figure 5.11 A recursive tree

- When the input models have different number of edges, the algorithm terminates without any recursion. The algorithm takes the space required in one instance of the function. The function defines constant number of variables that need to be accounted for in terms of space in addition to the input. So, an instance of the function takes $\mathcal{O}(1)$ space.
- When the two models have same number of edges, algorithm *checkGeneralizedEdgeDeletion* is called. The number of edges in the models for each successive call to *checkGeneralizedEdgeDeletion* is strictly less than n (namely, $n-1$). Another algorithm that is called is *checkSuccessors*. In this algorithm, $|L|$ strictly increases. Next it should be noted that the function call is not made if $|L| = |W_1 \times W_2|$. With these observations, we can bound the depth of recursion tree by $|R_1| \times |W_1 \times W_2|$. This shows that the algorithm terminates.

With the above observations, we see that the depth of the recursion tree is bounded by $|R_1| \times |W_1 \times W_2|$. Therefore, the space used by the algorithm is $s \times |R_1| \times |W_1 \times W_2|$, where s is the space used by one instance of the algorithm gen-Bisimilar. The algorithm defines constant number of variables, which take space other than the input. So, once again, one instance of the function takes $\mathcal{O}(1)$ space. Therefore, space used by whole run of Algorithm 5.1 is $|R_1| \times |W_1 \times W_2|$ which is a polynomial function in the size of the input. ∎

5.3.3 On $\langle de \rangle$-bisimulation

In what follows, we prove that the algorithm works when $state = \langle de \rangle\text{-}bisimilar$. The other cases can be proved in a similar manner.

Theorem 5.3: Given two models $(\mathcal{M}_1, w_1)$ and $(\mathcal{M}_2, w_2)$, $(\mathcal{M}_1, w_1) \underline{\leftrightarrow}_d (\mathcal{M}_2, w_2)$ iff the function gen-Bisimilar$((\mathcal{M}_1, w_1), (\mathcal{M}_2, w_2), \varnothing, \varnothing, \langle de \rangle\text{-}bisimilar)$ returns yes. Here, by $(\mathcal{M}_1, w_1) \underline{\leftrightarrow}_d (\mathcal{M}_2, w_2)$ we denote that $(\mathcal{M}_1, w_1)$ and $(\mathcal{M}_2, w_2)$ are $\langle de \rangle$-*bisimilar*.

Proof The proof is very similar to the correctness of Algorithm 5.1.

Suppose $\mathcal{M}_1$ and $\mathcal{M}_2$ have a different number of points, then the function returns No at line 2 in Algorithm 5.3, and $(\mathcal{M}_1, w_1) \not\underline{\leftrightarrow}_d (\mathcal{M}_2, w_2)$. So, let us consider that both models have an equal number of points (say n). We prove by induction on n:

> **Base case:** $n = 1$.
To prove $(\mathcal{M}_1, w_1) \underline{\leftrightarrow}_d (\mathcal{M}_2, w_2)$ iff the function returns yes when $|W_1| = |W_2| = 1$. We will first prove, by contrapositivity, that if the function returns yes, then $(\mathcal{M}_1, w_1) \underline{\leftrightarrow}_d (\mathcal{M}_2, w_2)$.

> > Suppose $(\mathcal{M}_1, w_1) \not\underline{\leftrightarrow}_d (\mathcal{M}_2, w_2)$. Then they violate one of the five conditions in the definition of $\langle de \rangle$-bisimilarity (in Section 5.2.1).

> > > Suppose they violate condition (1). Then there is some atomic proposition p such that either $(\mathcal{M}_1, w_1) \vDash p$ and $(\mathcal{M}_2, w_2) \nvDash p$; or $(\mathcal{M}_1, w_1) \nvDash p$ and $(\mathcal{M}_2, w_2) \vDash p$. From truth definitions, we have $w_1 \in V_1(p)$ but $w_2 \notin V_2(p)$; or $w_1 \notin V_1(p)$ but $w_2 \in V_2(p)$. In this case, the function returns No in line 3 in Algorithm 5.4.

> > > Suppose they violate condition (2). Then, there is a successor $v_1 \in W_1$ such that $w_1 R_1 v_1$, but $\forall v_2$ such that $w_2 R_2 v_2$, we do not have $(\mathcal{M}_1, v_1) \underline{\leftrightarrow}_d (\mathcal{M}_2, v_2)$. But since $n = 1$, $w_1 R_1 v_1 \implies v_1 = w_1$ as $|W_1| = 1$ and w_2 has no success as $|W_2| = 1$. In this case, the function returns No at line 16 in Algorithm 5.14.

> > > Suppose they violate condition (3), according to a similar argument as the last case, the function returns No at line 33 in Algorithm 5.14.

> > > Suppose they violate condition (4). Then there is a point $u_1 \in W_1$ and $u_1 \neq w_1$ such that for any point $u_2 \in W_2$ and $u_2 \neq w_2$, it is not the case that $(\mathcal{M}_1 \backslash \{u_1\}, w_1) \underline{\leftrightarrow}_d (\mathcal{M}_2 \backslash \{u_2\}, w_2)$. But again s,ince $n = 1$, $u_1 \neq w_1$ and $u_2 \neq w_2$

can not hold, this case cannot happen.

> > > By a similar argument as in the previous case, the condition (5) cannot be violated.

> > Now we will prove the other side, again by contrapositivity. Suppose that the function returns No, Then one of the following cases occur:

> > > The function returns No at line number 3 in Algorithm 5.4. This can only happen when the If condition at line 2 in Algorithm 5.4 is true. Therefore, there exists an atomic proposition p such that, $w_1 \in V_1(p)$ but $w_2 \notin V_2(p)$; or $w_1 \notin V_1(p)$ but $w_2 \in V_2(p)$. From truth definitions, we have either $(\mathcal{M}_1, w_1) \vDash p$ and $(\mathcal{M}_2, w_2) \nvDash p$; or $(\mathcal{M}_1, w_1) \nvDash p$ and $(\mathcal{M}_2, w_2) \vDash p$. But then $(\mathcal{M}_1, w_1) \not\rightleftarrows_d (\mathcal{M}_2, w_2)$ as they violate condition (1) of the definition of $\langle de \rangle$-bisimilarity.

> > > The function returns No at line number 12 in Algorithm 5.12. This can only happen if the condition at line 11 in Algorithm 5.12 is true. But since $|W_1| = 1$, $u_1 \neq w_1$ cannot hold. Hence this case cannot happen.

> > > The function returns No at line number 26 in Algorithm 5.12. By a similar argument as the previous case, since $|W_2| = 1$, $u_2 \neq w_2$ can not hold at line 25 in Algorithm 5.12. Hence this case cannot happen.

> > > Suppose the function returns No at line 16 in Algorithm 5.14. Then the condition at line 15 is true, i.e., *Found* $= 0$ and $w_1 R_1 u_1$. Since $|W_1| = 1$, there are several possibilities.

- $(w_1, w_1) \in R_1$ and $(w_2, w_2) \notin R_2$, which violates condition (3) of the definition of $\langle de \rangle$-bisimilarity, hence $(\mathcal{M}_1, w_1) \not\rightleftarrows_d (\mathcal{M}_2, w_2)$.
- $(w_1, w_1) \in R_1, (w_2, w_2) \in R_2$, $(w_1, w_2) \notin L$ and gen-Bisimilar$((\mathcal{M}_1, w_1), (\mathcal{M}_2, w_2), L \cup \{(w_1, w_2)\}, \varnothing, \langle de \rangle$-bisimilar$)$ = No. Then it may happen in the following cases.
  * Algorithm 5.1 returns No at line 13, this case violates condition (1) of the definition of $\langle de \rangle$-bisimilarity, hence $(\mathcal{M}_1, w_1) \not\rightleftarrows_d (\mathcal{M}_2, w_2)$.
  * Algorithm 5.1 returns No at line 47, which means Algorithm 5.12 returns No. But in Algorithm 5.12, the condition $w_1 \neq w_1$ cannot hold at line 11, and the condition $w_2 \neq w_2$ cannot hold at line 25. Thus this case cannot happen.
  * Algorithm 5.1 returns No at line 56, which means Algorithm 5.14 returns No. But in Algorithm 5.14, the condition $(w_1, w_2) \in L \cup \{(w_1, w_2)\}$ cannot hold at line 1, which means this case cannot happen.

> > > Suppose the function returns No at line 33 in Algorithm 5.14. By a similar argument as the previous case, $(\mathcal{M}_1, w_1)\not\underline{\leftrightarrow}_d(\mathcal{M}_2, w_2)$. This completes both sides of the base case.

> **Induction Hypothesis:** Suppose the theorem holds for $n \leq k$, which means $(\mathcal{M}_1, w_1)\underline{\leftrightarrow}_d(\mathcal{M}_2, w_2)$ iff gen-Bisimilar($(\mathcal{M}_1, w_1), (\mathcal{M}_2, w_2)$,∅,∅,$\langle de\rangle$-$bisimilar$) returns yes when $|W_1| = |W_2| \leq k$.

> **Induction Step:** Let $n = k + 1$

We will first prove that if $(\mathcal{M}_1, w_1)\underline{\leftrightarrow}_d(\mathcal{M}_2, w_2)$ then the function returns yes. Again we will prove this by contrapositivity.

> > Suppose the function returns No. Then it executes one of the 6 return No statements. But it can not return No at line 9, as we have assumed $|W_1| = |W_2|$. So the following cases can occur:

> > > The function returns No at line number 13. This can only happen when the If condition at line 2 in Algorithm 5.4 is true. But then, by an argument similar to that in the base case, $(\mathcal{M}_1, w_1)\not\underline{\leftrightarrow}_d(\mathcal{M}_2, w_2)$ as they violate condition (1) of the definition of $\langle de\rangle$-bisimilarity.

> > > The function returns No at line 12 in Algorithm 5.12. Then the condition at line 11 in Algorithm 5.12 is true. Therefore, there is some $u_1 \in W_1$ and $u_1 \neq w_1$ such that for all $u_2 \in W_2$ and $u_2 \neq w_2$, and the condition in line 5 is false, $i.e.$ there is an $u_1 \in W_1$ and $u_1 \neq w_1$ such that for all $u_2 \in W_2$ and $u_2 \neq w_2$, and we have gen-Bisimilar($(\mathcal{M}_1\backslash\{u_1\}, w_1), (\mathcal{M}_2\backslash\{u_2\}, w_2), \emptyset, \emptyset, \langle de\rangle\text{-}bisimilar$) returns No. But the model $\mathcal{M}_1' = \mathcal{M}_1\backslash\{u_1\}$ and $\mathcal{M}_2' = \mathcal{M}_2\backslash\{u_2\}$ have k points. Therefore, by induction hypothesis, $(\mathcal{M}_1\backslash\{u_1\}, w_1)\not\underline{\leftrightarrow}_d(\mathcal{M}_2\backslash\{u_2\}, w_2)$ for all $u_2 \in W_2$ and $u_2 \neq w_2$. This is a violation to condition (4) in the definition of $\langle de\rangle$-bisimilarity. Therefore, $(\mathcal{M}_1, w_1)\not\underline{\leftrightarrow}_d(\mathcal{M}_2, w_2)$.

> > > The function returns No at line 26 in Algorithm 5.12. By a similar argument as in the previous case, this leads to a violation of condition (5) in the definition of $\mathcal{L}de\rangle$-bisimilarity. Hence, $(\mathcal{M}_1, w_1)\not\underline{\leftrightarrow}_d(\mathcal{M}_2, w_2)$.

> > > The function returns No at line 16 in Algorithm 5.14. Then the condition at line 15 in Algorithm 5.14 is true for some $u_1 \in W_1$ with $w_1 R_1 u_1$. Therefore, the following cases arise:

> > > > The condition at line 5 in Algorithm 5.14 is false, i.e., for all $u_2 \in W_2$, $w_2 R_2 u_2$ is false for any $u_2 \in W_2$. This is a violation of condition (2) in the definition of

$\langle de \rangle$-bisimilarity and hence $(\mathcal{M}_1, w_1) \not\bumpeq_d (\mathcal{M}_2, w_2)$.

> > > > The condition at line 5 in Algorithm 5.14 is true, but the condition at line 7 in Algorithm 5.14 is false, i.e., $\exists u_1 \in W_1$ such that $w_1 R_1 u_1$, $\forall u_2 \in R_2$ such that $w_2 R_2 u_2$ and L is such that $(u_1, u_2) \notin L$ (and $(w_1, w_2) \notin L$ because line 6 can be executed only if condition in line 1 is true); we get gen-Bisimilar$((\mathcal{M}_1, u_1), (\mathcal{M}_2, u_2)), L \cup \{(w_1, w_2)\}, \varnothing, \langle de \rangle\text{-}bisimilar)$ returns No.

To prove:$(\mathcal{M}_1, w_1) \not\bumpeq_d (\mathcal{M}_2, w_2)$.

Proof by induction on $m = |W_1 \times W_2| - |L \cup \{(w_1, w_2)\}|$

> > > > > **Base case:** $|W_1 \times W_2| = |L \cup \{(w_1, w_2)\}|$

We need to prove that if $\exists u_1 \in W_1$ such that $w_1 R_1 u_1$, $\forall u_2 \in W_2$ such that $w_2 R_2 u_2$ and L is such that $(u_1, u_2) \notin L$ (and $(w_1, w_2) \notin L$ because line 6 can be executed only if the condition in line 26 is true), $|W_1 \times W_2| - |L \cup \{(w_1, w_2)\}| = 0$ and gen-Bisimilar$((\mathcal{M}_1, u_1), (\mathcal{M}_2, u_2)), L \cup \{(w_1, w_2)\}, \varnothing, \langle de \rangle\text{-}bisimilar)$ returns No, then $(\mathcal{M}_1, w_1) \not\bumpeq_d (\mathcal{M}_2, w_2)$. But since $|W_1 \times W_2| - |L \cup \{(w_1, w_2)\}| = 0$, we have $(u_1, u_2) \in L \cup \{(w_1, w_2)\}$. we notice that $(u_1, u_2) \notin L$ for all $u_2 \in W_2$ with $w_2 R_2 u_2$, then we have $(u_1, u_2) = (w_1, w_2)$ for all $u_2 \in W_2$ with $w_2 R_2 u_2$. Thus, $u_1 = w_1, u_2 = w_2$ and w_2 has only one successor, i.e., w_2. It follows that gen-Bisimilar$((\mathcal{M}_1, w_1), (\mathcal{M}_2, w_2)), L \cup \{(w_1, w_2)\}, \varnothing, \langle de \rangle\text{-}bisimilar)$ returns No, there are several possibilities.

- Algorithm 5.1 returns No at line 13, this case violates condition (1) of the definition of $\langle de \rangle$-bisimilarity, hence $(\mathcal{M}_1, w_1) \not\bumpeq_d (\mathcal{M}_2, w_2)$.
- Algorithm 5.1 returns No at line 47, which means Algorithm 5.12 returns No. Then in Algorithm 5.12, the condition at line 11 or 25 is false, which by inductive hypothesis implies $(\mathcal{M}_1, w_1) \not\bumpeq_d (\mathcal{M}_2, w_2)$ due to the violation of condition (4) or (5) in the definition of $\langle de \rangle$-bisimilarity.
- Algorithm 5.1 returns No at line 56, which means Algorithm 5.14 returns No. But in Algorithm 5.14, the condition $(w_1, w_2) \in L \cup \{(w_1, w_2)\}$ cannot hold at line 1, which means this case cannot happen.

> > > > > **Induction Hypothesis:** Suppose the claim holds for $m \leq l$, *i.e.*,

Suppose whenever $\exists u_1 \in W_1$ such that $w_1 R_1 u_1$, $\forall u_2 \in R_2$ such that $w_2 R_2 u_2$ and L is such that $(u_1, u_2) \notin L$ (and $(w_1, w_2) \notin L$ because line 6 can be executed only if the condition in line 1 is true) and $|W_1 \times W_2| - |L \cup \{(w_1, w_2)\}| \leq l$; and gen-Bisimilar$((\mathcal{M}_1, u_1), (\mathcal{M}_2, u_2)), L \cup \{(w_1, w_2)\}, \varnothing, \langle de \rangle\text{-}bisimilar)$ returns No, then $(\mathcal{M}_1, w_1)\underline{\not\leftrightarrow}_d(\mathcal{M}_2, w_2)$.

> > > > > **Induction step:** Suppose $m = l + 1$.

Since we have that gen-Bisimilar$((\mathcal{M}_1, u_1), (\mathcal{M}_2, u_2), L \cup \{(w_1, w_2)\}, \varnothing, \langle de\rangle$-$bisimilar)$ returns No. After checking Algorithm 5.1, it returns No either at one of 4 return No statements in Algorithm 5.1. If it returns No at lines 9, 13, and 47 then by the above cases, we have already proved that $(\mathcal{M}_1, u_1)\underline{\not\leftrightarrow}_d(\mathcal{M}_2, u_2)$ because they violate the previous lemma, condition (1), condition (4) or (5) in the definition of $\langle de\rangle$-bisimilarity respectively.

Suppose it returns No at line 56 in Algorithm 5.1, then Algorithm 5.14 returns No at line 16 or 33. We show the proof for the first case since the latter case can be dealt with in a similar way. Since Algorithm 5.14 returns No at line 16, then the condition at line 15 is true, i.e., *Found* = 0 and $u_1 R_1 v_1$ for some $v_1 \in W_1$. Since *Found* = 0, we have two cases in Algorithm 5.14. The first case is that the condition at line 5 is always false, which implies u_2 has no success, then $(\mathcal{M}_1, u_1)\underline{\not\leftrightarrow}_d(\mathcal{M}_2, u_2)$ because they violate condition (2) of the definition of $\langle de\rangle$-bisimilarity. The second case is that the condition at line 5 is true but at line 7 is false. Therefore, $\exists v_1 \in W_1$ such that $u_1 R_1 v_1$ and $\forall v_2 \in W_2$ such that $u_2 R_2 v_2$, L is such that $(v_1, v_2) \notin L \cup \{(w_1, w_2)\}$ (also $(u_1, u_2) \notin L \cup \{(w_1, w_2)\}$ because condition at line 1 has to be true), we have gen-Bisimilar$((\mathcal{M}_1, v_1), (\mathcal{M}_2, v_2)), L \cup \{(w_1, w_2), (u_1, u_2)\}, \varnothing, \langle de\rangle\text{-}bisimilar)$ returns No. Now by induction hypothesis, $(\mathcal{M}_1, v_1)\underline{\not\leftrightarrow}_d(\mathcal{M}_2, v_2)$ which implies $(\mathcal{M}_1, u_1)\underline{\not\leftrightarrow}_d(\mathcal{M}_2, u_2)$ and hence $(\mathcal{M}_1, w_1)\underline{\not\leftrightarrow}_d(\mathcal{M}_2, w_2)$.

> > > The function returns No at line 33 in Algorithm 5.14, then by an argument similar to the last case, $(\mathcal{M}_1, w_1)\underline{\not\leftrightarrow}_d(\mathcal{M}_2, w_2)$.

> We will now prove the remaining side, i.e., when $n = k + 1$, the function returns Yes, then $(\mathcal{M}_1, w_1)\underline{\leftrightarrow}_d(\mathcal{M}_2, w_2)$.

> Suppose $(\mathcal{M}_1, w_1)\underline{\not\leftrightarrow}_d(\mathcal{M}_2, w_2)$. Then these models must violate one of the five conditions in the definition of $\langle de\rangle$-bisimilarity.

> > Suppose they violate condition (1). There is some atomic proposition p such that either $(\mathcal{M}_1, w_1) \vDash p$ and $(\mathcal{M}_2, w_2) \nvDash p$; or $(\mathcal{M}_1, w_1) \nvDash p$ and $(\mathcal{M}_2, w_2) \vDash p$. From the truth definitions, we have $w_1 \in V_1(p)$ but $w_2 \notin V_2(p)$; or $w_1 \notin V_1(p)$ but $w_2 \in V_2(p)$. In this case, the function returns No at line 13 in Algorithm 5.1.

> > Suppose they violate condition (4). Then there is a point $u_1 \in W_1$ and $u_1 \neq w_1$ such that for any point $u_2 \in W_2$ and $u_2 \neq w_2$, $(\mathcal{M}_1\backslash\{u_1\}, w_1)\underline{\not\leftrightarrow}_d(\mathcal{M}_2\backslash\{u_2\}, w_2)$. In this

case for u_1, the condition at line 5 is never true by induction hypothesis ($\mathcal{M}_1\backslash\{u_1\}$ and $\mathcal{M}_2\backslash\{u_2\}$ have k edges, hence we can use induction hypothesis). Therefore, return No is executed in line 47 in Algorithm 5.1.

> > Suppose they violate condition (5), by a similar argument as the previous case, by the induction hypothesis, the function returns No in line 47 in Algorithm 5.1.

> > Suppose they violate condition(s) (2) and/or (3). We need to prove if $(\mathcal{M}_1, w_1) \not\leftrightarrows_d (\mathcal{M}_2, w_2)$ because they violate condition (2) and/or (3), but not (1) or (4)(5) in the definition of $\langle de\rangle$-bisimilarity, then we have that genBisimilar$((\mathcal{M}_1, u_1), (\mathcal{M}_2, u_2), L, \varnothing, \langle de\rangle\textit{-bisimilar})$ returns No, for $L = \varnothing$.

Since $(\mathcal{M}_1, w_1) \not\leftrightarrows_d (\mathcal{M}_2, w_2)$ because they violate condition (2) and/or (3), therefore $\exists u_{11} \in W_1$, $w_1 R_1 u_{11}$, such that $\forall u_{21} \in W_2$, $w_2 R_2 u_{21}$, $(\mathcal{M}_1, u_{11}) \not\leftrightarrows_d (\mathcal{M}_2, u_{21})$ (if condition (2) is violated); or $\exists u_{12} \in W_2$, $w_2 R_2 u_{12}$, such that $\forall u_{11} \in W_1$, $w_1 R_1 u_{11}$, $(\mathcal{M}_1, u_{11}) \not\leftrightarrows_d (\mathcal{M}_2, u_{21})$. Now if $(\mathcal{M}_1, u_{11}) \not\leftrightarrows_d (\mathcal{M}_2, u_{21})$ because they violate conditions (1) or (4) (5), then by previous cases, the function returns No at line 13 or 47 respectively, and we will be done. Let us pick a general such pair (v_{11}, v_{21}). So, assume $(\mathcal{M}_1, v_{11}) \not\leftrightarrows_d (\mathcal{M}_2, v_{21})$ because they violate condition(s) (2) and/or (3). Therefore, again, $\exists u_{12} \in W_1$ with $v_{11} R_1 u_{12}$, such that $\forall u_{22} \in W_2$ with $v_{12} R_2 u_{22}$: $(\mathcal{M}_1, u_{12}) \not\leftrightarrows_d (\mathcal{M}_2, u_{22})$ (if they violate (2)); or $\exists u_{22} \in W_2$ with $v_{12} R_2 u_{22}$, such that $\forall u_{12} \in W_1$ with $v_{11} R_1 u_{12}$: $(\mathcal{M}_1, u_{12}) \not\leftrightarrows_d (\mathcal{M}_2, u_{22})$ (if they violate condition (3)). Again, choose a general such pair (v_{12}, v_{22}) from above such that $v_{11} R_1 v_{12}$ and $v_{21} R_2 v_{22}$ and $(\mathcal{M}_1, v_{12}) \not\leftrightarrows_d (\mathcal{M}_2, v_{22})$. Again, we are done if $(\mathcal{M}_1, v_{12}) \not\leftrightarrows_d (\mathcal{M}_2, v_{22})$ because they violate condition (1), (4) or (5). So, again, we can assume that they violate condition(s) (2) and /or (3). This can go on until we reach a leaf node, i.e., there is some k such that exactly one of the following is true: $v_{1k} R_1 v_{1k+1}$ for some $v_{1k+1} \in W_1$ or $v_{2k} R_2 v_{2k+1}$ for some $v_{2k+1} \in W_2$. Again the function returns No in both cases. The only case that remains is when there is no leaf nodes and there is some k such that $v_{1k} = v_{1l}$ or w_1 and $v_{2k} = v_{2l}$ or w_2 for some $l < k$. In this case, since v_{1i} and v_{2i} were some general node in the reachable part from w_1 and w_2, such that they do not violate condition (1), (4) or (5) in the definition of $\langle de\rangle$-bisimilarity, we have the following:

> > > $(\mathcal{M}_1, w_1)$ and $(\mathcal{M}_2, w_2)$ satisfy conditions (1), (4) and (5) in the definition of $\langle de\rangle$-bisimilarity.

> > > For every n, $\exists v_1 \in W_1$ such that $w_1 R_1^n v_1$ iff $\exists v_2 \in W_2$ such that $w_2 R_2^n v_2$; and

$(\mathcal{M}_1, v_1)$ and $(\mathcal{M}_2, v_2)$ satisfy condition (1), (4) and (5) from the definition of $\langle de \rangle$-bisimilarity. But these conditions are the same as the conditions in the definition of $\langle de \rangle$-bisimilarity. Hence, $(\mathcal{M}_1, w_1) \underline{\leftrightarrow}_d (\mathcal{M}_2, w_2)$ and the function does not return No in this case.

This completes the proof. ∎

As we showed in the case of $\langle gsb \rangle$-bisimulation, we can show that the function genBisimilar terminates and is in **PSPACE** for $state = \langle de \rangle\text{-}bisimilar$.

5.3.4 Bisimulation for other states

The main difference in the run of Algorithm 5.1, based on the different values of *state*, is that it calls different functions, namely, *checkEdgeDeletion* in case of $\langle sb \rangle$-bisimilar, *checkNodeDeletion* in case of $\langle de \rangle$-bisimilar, and so on. These functions check the analogous conditions (4) and (5) for different notions of bisimilarity (for $\langle ch \rangle$-bisimilar, the function checks condition ($\star$) rather than conditions (4) and (5)). The rest of the algorithm remains the same. Therefore the correctness proofs for other notions of bisimilarity are very similar to that of the case of $state = \langle gsb \rangle\text{-}bisimilar$. Thus, the complexity for all bisimilarity problems remains to be in **PSPACE**. So, we have the following main theorem of this work.

Theorem 5.4: Given two pointed relational models, $(\mathcal{M}_1, w_1)$ and $(\mathcal{M}_2, w_2)$, the problem of deciding whether they are $\langle up \rangle$-bisimilar is in **PSPACE** when $\langle up \rangle \in \{\langle sb \rangle, \langle gsb \rangle, \langle sw \rangle, \langle gsw \rangle, \langle br \rangle, \langle gbr \rangle, \langle de \rangle, \langle ch \rangle\}$.

5.4 Further remarks

In this work, we have presented several existing modal logics concerned with the model changing phenomena and studied the notion of model comparison or bisimulation for these logics from the algorithmic point of view. In the process, we have also shed some light on the complexity of these problems. However, we have only been able to show upper bound results, and not any lower bound result. We now provide some discussions on the lower bounds of these complexity problems, which are the natural next questions to answer. Of course, results on tight bounds will complete this whole study.

Before going there, let us first note that the complexity for checking whether given two pointed models are bisimilar, in basic modal logic, is known to be in polynomial time (Paige and Tarjan, 1987). What exactly makes the problem of $\langle up \rangle$-bisimilarity more complex (strictly more complex if **PTIME** is different than **PSPACE**)? The additional conditions (4) and (5) in the definition of $\langle up \rangle$-bisimilarity, compared to that of basic modal logic bisimilarity, requires a function that assigns a sequence of model-changing actions corresponding to one model to a sequence of model-changing actions in the other model. Formally, it requires a bijection $f : N(C_1) \to N(C_2)$ with $N(C)$ denoting the set of sequences of actions corresponding to the model-changing operator C. The function f should additionally satisfy the condition that any sequence of length n is mapped to a sequence of length n, for every n in $\mathbb{N}$. If $|C_1| = |C_2| = m$, then there are $2^{m^{2^m}}$ such functions. Given such a function, we need to check whether it satisfies the corresponding conditions for $\langle up \rangle$-bisimilarity on top of the models being bisimilar in the sense of basic modal logic. These conditions are what make this problem of $\langle up \rangle$-bisimilarity more complex. If we can show that every such function that satisfies the conditions for $\langle up \rangle$-bisimilarity is generated by a function $g : C_1 \to C_2$, then we believe that the complexity of $\langle up \rangle$-bisimilarity drops to the class **NP**. To draw an analogy, deciding whether given two graphs are isomorphic is in **NP**, but finding the isomorphism mapping may be more complex. This is equivalent to saying that given a small (with a number of elements bounded by a polynomial in the size of the input models) candidate generator of the relation $\langle up \rangle$-bisimilar, it may be efficient to check whether such a candidate can be extended to a full $\langle up \rangle$-bisimilar relation.

With regard to lower bound results, a general way one shows that a problem is $\mathcal{C}$-hard, for a complexity class $\mathcal{C}$, is by providing a polynomial-time reduction from an already known $\mathcal{C}$-complete problem to the problem in question. The problem for determining whether a fully quantified Boolean formula is true or false, that is, the *TQBF* problem (Sipser, 1996), is known to be **PSPACE**-complete, and has been used to prove **PSPACE**-hardness of many logic-related decision problems. However, finding such reduction functions from the *TQBF* and similar other **PSPACE**-complete problems to these model comparison problems have turned out to be quite involved. Complications arose even for showing **NP**-hardness results for these model comparison problems using **NP**-complete *SAT* and other problems. The main issue, as described above, is finding possibly simpler, more procedural conditions equivalent to the conditions (4) and (5) in the definition of $\langle up \rangle$-bisimulation that would facilitate such reduction results.

To end this chapter, let us get back to the general discussion of games on graphs. What does it mean to have a bisimulation between two game graphs with respect to two points on those two graphs? Evidently, whatever moves a player can make in one game, the same kind of moves can be made in the other game as well. Moreover, an alternation of the basic modality with the edge-deletion or node-deletion modality would describe a play in the game graphs with link deletion or point deletion, respectively. From the strategic viewpoint, the age-old copy strategy might be a relevant strategy to play on bisimilar game graphs. In fact, checking bisimilarity between different game graphs can be considered as a first step towards considering game-strategy equivalences between these games constituting structural changes in the underlying graphs. The related notion of $\langle up \rangle$-bisimulation contraction (van Benthem, 2010) concerns with the *simplest* graphs that are structurally equivalent to the original graphs. From the game perspectives, as mentioned earlier, we have the following question: What would the simplest graph in an equivalence class that would still show the essential structure of the game?

CHAPTER 6 LOGICS FOR PERSONALIZED ANNOUNCEMENTS

One of the most popular shows of 2020 has been the Netflix docudrama *The Social Dilemma*, which alarmed the public over the dangers of the prevalent engagement-based business model of popular social media (Wikipedia contributors, 2023). It explored, among others, the responsibility of platforms, facilitated by personalized advertizing and by the exploitation of the intrinsic properties of human cognition, such as our limited and selective attention. Although the dramatized story of family members drifting apart in their own bubbles has been criticized as overly simplistic and exaggerated (Girish, 2023), few can deny that our behavior and beliefs are affected by the design of social media. In particular, by exposing us to information that is tailored to our personal choices and preferences, these platforms can induce a bias in our news intake and thereby tend to influence and sometimes reinforce our beliefs (Flaxman et al., 2016).

Aside from typical users, influencers are also impacted by the design of online platforms. Influencers of Instagram, for instance, had a reason to worry when the platform tested hiding the number of 'likes' that a post receives. Indeed, their online success and popularity does not only depend on users' pre-existing attitude towards the content of a post (Paul, 2019), but also on parameters such as likes, views, shares, etc. It is thus increasingly clear that the way we access information is crucial in nowadays society and deserves in-depth investigation. To this end, formal-logic approaches represent a solid tool to analyse the intrinsic structural properties of social phenomena on a high level of abstraction and are thus promising for the analysis of the flow of information in applications in such social platform systems.

Logical approaches have been used extensively to study social networks, for example to analyse the propagation of opinions in a social network, the dynamics of the network's structure, and the entanglement of knowledge and social relation structure. See e.g. (Seligman et al., 2011; Smets and Velázquez-Quesada, 2017, 2018, 2019a,b; Baltag et al., 2019a; Smets and Velázquez-Quesada, 2020; Pedersen et al., 2020; Liu and Liao, 2021; Liu and Li, 2022). Our work continues this line of investigation by taking into account the dynamics of opinions as well as the constraints that can be imposed on the access to information that agents have. Different parameters pertain to the *access* of agents to information, and concern both the opinions or features of other agents (does your friend Alice like punk music?). The access of the agents to such pieces of information is crucial to the formation of their beliefs, which in turn underlie the evolution of social phenomena like the diffusion of fashions, the formation of echo chambers, pluralistic ignorance, polarization, etc. It is worth noting that we do not investigate the structural relationships between agents, but rather how the accessibility and spread of information through social media influences the evolution of beliefs.

The aforementioned studies in logic work with a certain level of abstraction and hence include idealizing assumptions that can be revisited now. For instance, it is assumed in some of the work that agents know who their friends are, including all of their features (Smets and Velázquez-Quesada, 2019a, 2020, 2018, 2019b). In reality, we do not necessarily keep track and have access to all this information, especially as it may come in large amounts and it evolves dynamically. Some modeling approaches explicitly account for the epistemic layer that underlies social network phenomena, and reflect that the agents' behaviour is dictated by their epistemic situation (Baltag et al., 2019a; Smets and Velázquez-Quesada, 2017). Yet, such frameworks started from the standard $\mathsf{S5}$ epistemic logic, which has advantages on the modeling side but also comes with well-known shortcomings when modeling the knowledge of real human agents see e.g. (Solaki, 2021). In addition to the epistemic aspects, when it comes to reasoning in online social networks, one would need to take into account the algorithmic design of the platform in use.

In our setting, we will focus on weaker doxastic logics and on the structural filtering of information. This type of filter pertains to the informational bounds placed by the designers of the social platform to sort the information and to ensure that browsing runs smooth and remains interesting and engaging for the users. More specifically, although the members of a given online network generate posts with the intention of reaching *all* of their connections, this rarely succeeds in practice. Given the multitude of online connections and the posts available to appear in one's feed, some form of filtering becomes necessary to maintain engagement. This necessity has revolutionized the use of *recommender systems* in sorting which posts appear to the users. Recommender systems rely heavily on *personalization* (Jannach et al., 2010). The subset of posts that will end up appearing on an agent's feed are the ones deemed sufficiently compatible with the preference profile built for this agent. For example, *user-based collaborative filtering* methods recommend posts to users on the basis of their similarity with agents with a similar profile; the *item-based collaborative filtering* method recommends posts to users on the basis of the similarity between these items (here: posts) and others in terms of how the users interacted with both; *content-based filtering* recommends posts with features the user has interacted favourably with in the past. Even in very compact networks, personalization results in similar agents becoming exposed to radically different feeds, a practice that has been blamed for the increasing polarization in social media (Vīķe-Freiberga et al., 2013).

It is worth emphasizing that not only does the relevant information selectively reach only some members of the intended audience, but additionally no member of the network is aware of the filtering outcome of the recommender system – largely due to the lack of transparency of how the chosen recommender system works. For example, neither the sender of a post nor their fellow friends are ever certain on who actually passed the filter and gained access to the posted piece of information.

In our formal analysis below, we will model such scenarios in terms of update methods coming from dynamic epistemic logic (Baltag et al., 1998; van Ditmarsch et al., 2008; van Benthem, 2011), which deals with knowledge and belief dynamics (partly) triggered by private information. To do so, we will use the convenient format of the so called edge-conditioned action models in (Bolander, 2018), where the filtering conditions can be treated as edge conditions on events that are partially observed by agents. In Appendix 6.4, we show how the edge-condition driven update could also in principle be represented in terms of the standard product update of a more classical kind used in dynamic epistemic logic.

More specifically, we will propose a logical framework to characterize the personalized distribution of information based on filtering conditions in social platforms, which consequently leads to the dynamics of agents' beliefs. It is crucial to acknowledge that the filtering mechanisms employed by social platforms are based on intricate statistical methods. The filtering conditions under consideration in this chapter are abstractions of the underlying concepts behind various filtering methods, expressed in terms of the components of the models we proposed below. This approach in this chapter can potentially yield further insights into the emergence of various phenomena, such as polarization, thereby providing valuable guidance for the design and refinement of filtering systems.

The rest of the paper can be summarized as follows: In Section 2, we introduce a framework based on dynamic epistemic logic, where announcements are personalised by semantically specified filtering conditions, from the static to dynamic setting and we give the axiomatizations. In Section 3, we further explore the extension of our existing framework to incorporate factors related to the agents themselves, namely the agent's cognitive capacity and attentive resources.

6.1 Logics for personalization in social platforms

In what follows, we build a social-doxastic logical framework suitable for the dynamics of personalization in a social platform. As a result, it does justice to some type of limitations pertaining to the informational flow in the platform, namely the structural filtering limitations. We first present the static basis for our account, which we call the static logic of personalized announcements (*SLPA*), and we subsequently proceed to the development of a completely dynamic logic framework that captures the dynamics of belief due to the filtering condition in social platforms.

6.1.1 Static logic of personalized announcements

Our approach is based on the work of Smets and Velázquez-Quesada (2019a). We first introduce the syntax of the static logic of personalized announcements (*SLPA*), and then the semantics of this logic. For convenience, we introduce some notations. Let $Ag \neq \varnothing$ be a finite set of agents. Let $T \neq \varnothing$ be a finite set of topics about which the agents can form an opinion. We denote opinions on topics by $\{R_t\}_{t \in T}$ as a pairwise disjoint collection providing a finite non-empty set R_t of opinions for each $t \in T$. In addition, let R denote $\cup_{t \in T} R_t$.

Definition 6.1 (Opinion Model): Given Ag, T, and $\{R_t\}_{t \in T}$, an opinion model (OM) is a tuple $\mathbf{M} = \langle \mathbf{W}, \{\to_j\}_{j \in Ag}, \mathbf{V} \rangle$ where:

- $\mathbf{W}$ is a non-empty set of possible worlds.
- $\to_j$ is the doxastic accessibility relation for agent $j \in Ag$. Additionally, we ask that an agent considers at least a world possible, which means $\to_j$ is serial, i.e., for any $w \in \mathbf{W}$, there exists $u \in \mathbf{W}$ such that $w \to_j u$.
- $\mathbf{V} : W \times Ag \times T \to \mathcal{P}(R)$ is the valuation function, where $\mathbf{V}(w, i, t) \subseteq R_t$ indicates the opinions of agent i on topic $t \in T$ in world w.

Additionally, we require that an agent's valuation function values do not change across their doxastically indistinguishable worlds in opinion models:

1. If $w \to_i u$, then $\mathbf{V}(w, i, t) = \mathbf{V}(u, i, t)$, for any $t \in T$.

Example 6.1: Consider the following possible opinions on 3 topics that Alice ($\mathtt{A}$), Bob

(B) and Carol (C) can express on a social media platform :

- ChatGPT (C)
  - (d) The emergence of ChatGPT will cause a stagnation in the creative writing development of children.
  - (b) ChatGPT produces information that is not fact-checked and as such it enhances the spread of misinformation.
  - (c) The development of ChatGPT may lead to the emergence of machine worship and machine superstition.
  - (f) The implementation of ChatGPT in specific sectors holds the potential to considerably enhance productivity, which in turn may inadvertently lead to an escalation in unemployment rates.
  - (h) ChatGPT can engage in virtual chatting and role-playing, providing entertainment and interactive experiences that help with people's emotional and social development.
- Big Data (O)
  - (a) Big data can reinforce existing biases and perpetuate discrimination.
  - (g) Big data can lead to privacy and security issues.
  - (p) Big data can be used to identify patterns and trends that were previously unknown or difficult to detect.
  - (s) Big data can help in the development of personalized products and services.
- Virtual Reality (D)
  - (r) VR can enhance training and simulation by allowing individuals to practice real-world scenarios in a safe and controlled environment.
  - (l) VR can lead to users' detachment from the real world and loss of communication and social skills.
  - (i) VR can cause psychological issues such as hallucinations, anxiety, and dependence.
  - (o) VR may lead to disappointment with the real world, as it cannot be as exciting as the virtual world.

It is assumed that Alice, Bob and Carol form a group of users, and they don't know others' opinions on these topics, but they can make some judgments based on what others have posted in the past. The doxastic relations among them are depicted in Figure 6.1. In this figure, there are three worlds, w, u and v; the symbols A, B and C denote Alice, Bob and Carol, respectively. $w \xrightarrow{\mathtt{A}} u$ denotes that Alice at world w believes the atomic sentence that are true at world u, and it is similar for other labelled arrows.

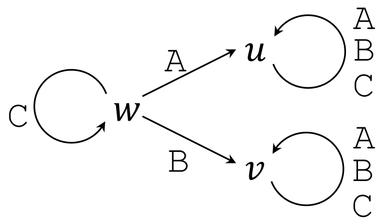


Figure 6.1 Doxastic relations among agents

Suppose w is the actual world, all opinions from Alice, Bob and Carol over these topics are as follows, as we show in world w in Figure 6.2, where the string $V_C : \{d, b, c, f\}$ below $\mathtt{A}$ denotes that the opinions of Alice over the topic C are d, b, c and f mentioned, and other notations in Figure 6.2,6.3, and 6.4 are similar.

- Alice : $C : \{d, b, c, f\}$; $O : \{a, g, p\}$; $D : \{r, l, i\}$.
- Bob: $C : \{d, f\}$; $O : \{a, g\}$; $D : \{r, l\}$.
- Carol: $C : \{f, h\}$; $O : \{p, s\}$; $D : \{l\}$.

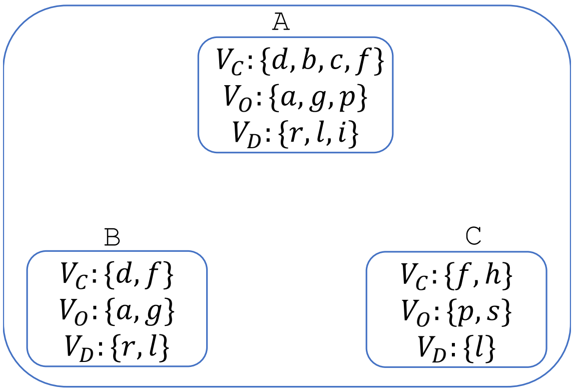


Figure 6.2 w

Alice believes that Bob and Carol have the following opinions. The world Alice believes is shown in world u in Figure 6.3.

- Bob: $C : \{d, c, f\}$; $O : \{a, p\}$; $D : \{r, l, o\}$.
- Carol: $C : \{f\}$; $O : \varnothing$; $D : \{l, o\}$.

While Bob believes that Alice and Carol have the following opinions. The world Bob believes is shown in world v in Figure 6.4.

- Alice: $C : \{d\}$; $O : \{a, p\}$; $D : \{r, o\}$.
- Carol: $C : \{f, h\}$; $O : \{p, s\}$; $D : \varnothing$.

Carol's beliefs about the other agents' opinions happen to be their actual opinions, which is exactly what happens in Figure 6.2.

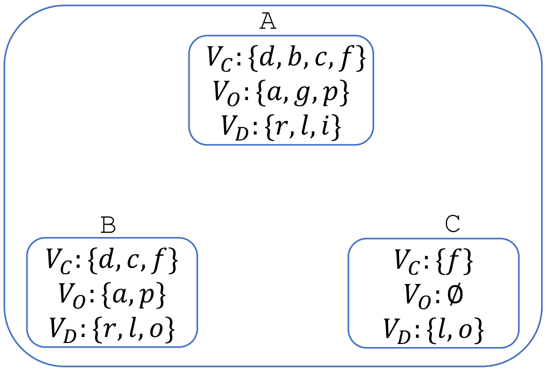


Figure 6.3 u

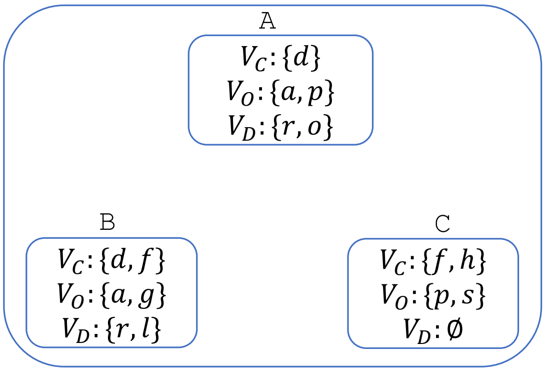


Figure 6.4 v

$\mathbf{M} = \langle \mathbf{W}, \{\rightarrow_j\}_{j \in Ag}, \mathbf{V} \rangle$ is an opinion model of the type that we have defined, where $Ag = \{\mathtt{A}, \mathtt{B}, \mathtt{C}\}$, $\mathbf{W} = \{w, u, v\}$, $\{\rightarrow_j\}_{j \in Ag}$ is shown in Figure 6.1, and the valuation function $\mathbf{V}$ is shown in Figure 6.2 - 6.4.

Definition 6.2: The language $\mathcal{L}$ of the static logic of personalized announcements (*SLPA*) is given by:

$$\phi ::= i_t^r \mid \neg\phi \mid \phi \wedge \phi \mid \mathtt{F}(j, i_t^r) \mid B_i\phi$$

where i_t^r is an atomic proposition letter, $i \in Ag$ and $r \in R_t$, for $t \in T$.

The boolean operators $\rightarrow, \vee$ are defined as usual. For $j \in Ag$, the dual operator $\hat{B}_j$ of B_j, is defined as follows: $\hat{B}_j\phi := \neg B_j \neg\phi$. Intuitively, i_t^r can be read as follows: "agent i makes a post about her opinion r on topic t". We introduce the access operator $\mathtt{F}(j, i_t^r)$ to express the accessibility of agent j to the post i_t^r. The guiding principle is that the truth value of $\mathtt{F}(j, i_t^r)$ rests upon whether the agent j is deemed sufficiently "compatible" with the post i_t^r by the recommender system of the platform in question, i.e., whether j passes the filtering method and would thus has the post i_t^r appearing in her feed.

Definition 6.3 (Semantic clauses): Given an OM $\mathbf{M}$ and world $w \in \mathbf{W}$:

- $\mathbf{M}, w \vDash i_t^r$ iff $r \in \mathbf{V}(w, i, t)$
- $\mathbf{M}, w \vDash \neg\phi$ iff $\mathbf{M}, w \nvDash \phi$
- $\mathbf{M}, w \vDash \phi \wedge \psi$ iff $\mathbf{M}, w \vDash \phi$ and $\mathbf{M}, w \vDash \psi$
- $\mathbf{M}, w \vDash B_i\phi$ iff $\mathbf{M}, u \vDash \phi$ for all $u \in \mathbf{W}$ such that $w \to_i u$
- $\mathbf{M}, w \vDash \mathtt{F}(j, i_t^r)$ iff there is a v with $w \to_j v$ such that $\mathbf{M}, v \vDash i_t^r$, and the **filtering condition** (as specified below) holds at w.

The evaluation of $\mathtt{F}(j, i_t^r)$ depends on the particular filtering condition in the recommender system that the online social platform is employing. The three parallel filtering conditions that we consider are: the *Radical Push Condition*, the *Conservative Push Condition* and the *Feature Push Condition*. We will elaborate on these filtering conditions and provide corresponding examples later on.

In order to later make sure that the seriality of the doxastic accessibility relation is preserved under product updates for personalized announcements, we have to ask that the formulas the agent has access to (pass the filtering condition) are consistent with the agent's beliefs. See the Consistency axiom in Table 6.1 as well.

In this way, we embed the reality of personalized information distribution, which is often largely determined by machine learning methods, in a logical (symbolic) framework paired with social platform analysis tools, in order to give a meta-analytical study of how such methods shape the informational dynamics, and thus the diffusion phenomena.

The **filtering conditions** Next, we elucidate the filtering condition for agent j when agent i is sending the post i_t^r, where $j \neq i$. Note that whenever $j = i$, we assume that the post sent by agent i will definitely appear in her own feeds. In other words, in this case, in addition to requiring the compatibility of the post content with agent i, there is no additional filtering condition required.

- **Radical Push Condition** This filtering condition is inspired by the so-called *item-based collaborative filtering* method of designing recommender systems. It relies on a similarity metrics between *items* (here: posts), calculated in terms of how they were interacted with by all agents. The method then predicts that the posts that should be recommended to the target agent are the ones that are similar to those the agent has interacted positively with. One everyday example application of this, encountered in many social media, can be summarized to "many users who liked that, like you, have also liked this". Given a threshold $\theta \in \mathbb{N}$, it can be interpreted as the lowest bound of consensus within a group. Specifically, if the number of individuals within the group who hold a certain belief exceeds the threshold, then that belief is considered to be held by the entire group.

At current state w, for some opinion r' held by agent j, if there are at least θ people in the group who hold or do not hold opinions r and r' simultaneously, agent j should treat opinions r and r' as equivalent, Therefore, since agent j holds opinion r', they should also tend to hold opinion r, and we should push opinion r to agent j accordingly.

In formal terms, there exists a witness $r' \in \mathbf{V}(w, j, t)$ such that

$$|\{k \in Ag \mid r' \in \mathbf{V}(w, k, t) \leftrightarrow r \in \mathbf{V}(w, k, t)\}| \geq \theta$$

We refer to the filtering condition described above as the *Radical Push Condition*.

- **Conservative Push Condition** This filtering condition follows the same principle as the previous one, which is "many users who liked that, like you, have also liked this", but the conservative push condition is more stringent. This condition not only requires a sufficient number of people to view opinion r and some opinion of agent j as equivalent, but also requires a sufficient number of people to view opinion r and all opinions of agent j as equivalent. Under this requirement, we have more reasons to believe that opinion r should also be a viewpoint that agent j is inclined to hold and would like to be pushed to them. Stated formally, every opinion r' held by agent j should be a witness, i.e., for all $r' \in \mathbf{V}(w, j, t)$ such that

$$|\{k \in Ag \mid r' \in \mathbf{V}(w, k, t) \leftrightarrow r \in \mathbf{V}(w, k, t)\}| \geq \theta$$

We call this filtering condition the *Conservative Push Condition*

Example 6.2: In Example 6.1, we focus on the topic C and consider whether the post from Bob that he holds opinion d over C will be recommended to Carol at world w. Fix a threshold $\theta = 2$,

- Using the *Radical Push Condition*: given that in world w, Carol holds opinion f, and Alice and Bob hold opinions d and f, the *Radical Push Condition* here is satisfied at w. Moreover, Carol believes that Bob holds the opinion d on topic C. Thus we have $\mathtt{F}(\mathtt{C}, \mathtt{B}^d_C)$ holds in world w, which means the post from Bob that he holds opinion d over C will be recommended to Carol at world w.
- Using the *Conservative Push Condition* Push: given that Carol holds opinion h and that no one holds opinions h and d together and actually one of them

holds opinion h iff the other agent does not hold opinion d. Thus the Conservative Push condition is satisfied at w, then we have that $\mathtt{F}(\mathtt{C}, \mathtt{B}^d_C)$ does not hold in world w, which means the post from Bob that he holds opinion d on C will not be recommended to Carol at world w.

Recommendation rules are indeed critical, as they shape the user's access to information. Actually, Carol does not hold the opinion d over topic C at w. However, if the social media platform uses the radical push condition, then Carol will receive this post, and realize that someone in her community holds the opinion d, and in the long run, this may have an impact on her. This outcome is not surprising, as the underlying notion of the radical push to "propagate opinions agreed upon by the majority to others" inevitably leads to such a result. Compared to the radical push condition, the conservative push condition may cause users to miss out on opinions that are truly interesting.

- **Feature Push Condition** This filtering condition is inspired by the so-called *content-based filtering* methods of designing recommender systems. It relies on the features of the items (here: posts) the target user has interacted favourably with. It then recommends posts that share these features.

  At the current state w, agent j does have opinions over topic t, then relative opinion r over t from agent i will push to agent j. In formal terms, that is to say,

$$\mathbf{V}(w, j, t) \neq \varnothing$$

  We call this filtering condition the *Feature Push Condition*.

  Example 6.3: In Example 6.1, at world u, Carol has no interest in topic O, which means $\mathbf{V}(u, \mathtt{C}, O) = \varnothing$. Thus the *Feature Push Condition* is not satisfied at u. Then any opinions on topic O from any other agents will not be recommended to Carol. Formally, $\mathtt{F}(\mathtt{C}, i^r_O)$ doesn't hold at world u for any $i \in Ag$, $r \in R_O$.

Please note that there are many filtering mechanisms across different platforms, such as *user-based filtering*, and so on. However, for the purpose of this study, we will only address the aforementioned three types of filters, and defer a more comprehensive analysis to future investigations.

6.1.2 Dynamic logic of personalized announcements

In what follows, we extend the basic static setting with the dynamics of information flow in online social platforms, and propose a dynamic logic of personalized announcements (*DLPA*). We present this logic by starting from the models.

First, we proposed so-called access models for describing the possible events in social platforms. More specifically, we utilize a filtering model for the personalized announcement of i^r_t to depict the potential events to agent j when agent i sends a post i^r_t, subject to the filtering conditions on the platform. Moreover, the belief states of agents are updated through the filtering model, which is reflected in our product models defined later. For convenience, we use the access model in the following, which is based on the edge-conditioned model in (Bolander, 2018). It is worth mentioning that the scenarios we would like to characterize can alternatively also be captured by action models in (van Benthem, 2011). Moreover, they share the same methodology and are equivalent, and we provide a proof of this fact in Appendix 6.4.

Definition 6.4 (Access model): An *Access model* is a tuple $\mathbf{C} := \langle \mathbf{E}, \{\mathbf{f_j}\}_{j\in Ag}, \mathbf{pre}\rangle$, where:

- $\mathbf{E}$ is a non-empty set of events.
- $\mathbf{f_j} : \mathbf{E}\times\mathbf{E} \to \mathcal{L}$ is the filtering function, assigning a condition to each event for each agent j.
- $\mathbf{pre} : \mathbf{E} \to \mathcal{L}$ is the precondition function.

Next, we propose specialized access models, i.e, the *filtering models*, for the scenarios that some agent sends a post i^r_t, and all agents in the social platform will receive this post only if the filtering condition is satisfied. Otherwise, they would have remained unaware of the event.

Definition 6.5 (Filtering model for the personalized announcement of i^r_t): A *filtering model for personalized announcements* is a tuple $\mathbf{C} := \langle \mathbf{E}, \{\mathbf{f_j}\}_{j\in Ag}, \mathbf{pre}\rangle$, where:

- $\mathbf{E} = \{e_0, e_1\}$ is a set of events, where event e_0 indicates that agent i does not send any post, while event e_1 indicates agent i sends the post i^r_t.
- The filtering function $\mathbf{f_j}(e, e')$ can be defined as follows. In particular, we ensure that agent j always receives the post i^r_t he sent, and others would receive the announcement if the platform system does not filter out the post for them.
  - for $j = i$, $\mathbf{f_j}(e, e') = \begin{cases} \top, \text{ if } e = e' = e_1 \\ \top, \text{ if } e = e' = e_0 \\ \bot, \textit{ otherwise} \end{cases}$
  - for $j \neq i$,

$$\mathbf{f_j}(e,e') = \begin{cases} \mathtt{F}(j, i_t^r), \text{ if } e = e' = e_1 \\ \neg\mathtt{F}(j, i_t^r), \text{ if } e = e_1 \text{ and } e' = e_0 \\ \bot \text{ if } e = e_0 \text{ and } e' = e_1 \\ \top, \text{ if } e = e' = e_0 \end{cases}$$

- **pre** is the precondition function. $\mathbf{pre}(e) = \begin{cases} i_t^r, \text{ if } e = e_1 \\ \top, \text{ if } e = e_0 \end{cases}$

Example 6.4: A filtering model for the post from Bob 'I hold opinion p on topic O'
We follow the example 6.1 and suppose that Bob is posting his opinion of p when it comes to topic C. The announcement of B_O^p will be effectively personalized by the algorithms of the social platform. This event can be captured via a filtering model, we focus on agent Carol and give a part of the filtering model in Figure 6.5.

$$\mathbf{f}_{\mathtt{C}}(e,e') = \begin{cases} \mathtt{F}(\mathtt{C}, \mathtt{B}_O^p), \text{ if } e = e' = e_1 \\ \neg\mathtt{F}(\mathtt{C}, \mathtt{B}_O^p), \text{ if } e = e_1 \text{ and } e' = e_0 \\ \bot \text{ if } e = e_0 \text{ and } e' = e_1 \\ \top, \text{ if } e = e' = e_0 \end{cases}$$

$$\mathbf{pre}(e) = \begin{cases} \mathtt{B}_O^p, \text{ if } e = e_1 \\ \top, \text{ if } e = e_0 \end{cases}$$

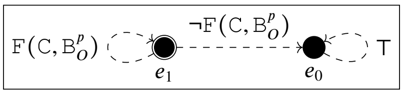


Figure 6.5 The personalized announcement of $\mathtt{B}_O^p$.

On the semantic level, we have proposed filtering models for the personalized announcement above. On the syntactic level, in order to incorporate personalized announcements, we add formulas of the form $[\mathbf{C}, e]\phi$ for a given filtering model $\mathbf{C}$, where e is an event in this model. The full dynamic language is as follows:

Definition 6.6: The dynamic language $\mathcal{L}^+$ of the dyanmic logic of personalized announcements (*DLPA*) is given by:

$$\phi ::= i_t^r \mid \neg\phi \mid \phi \wedge \phi \mid \mathtt{F}(j, i_t^r) \mid B_i\phi \mid [\mathbf{C}, e]\phi$$

where i_t^r is an atomic proposition letter, $i \in Ag$ and $r \in R_t$, for $t \in T$. $\mathbf{C}$ is a filtering model for for personalized announcement i_t^r and e is an event of the filtering model $\mathbf{C}$.

The satisfaction of dynamic modalities in formulas depends on the updating of the original opinion model, which is formally defined as follows:

Definition 6.7 (Updated model): Given an opinion model $\mathbf{M}$ and a filtering model $\mathbf{C}$ for personalized announcement i_t^r, the *updated model* is given by $\mathbf{M} \otimes \mathbf{C} := \langle \mathbf{W}', \{\rightarrow'_{\mathbf{j}}\}_{j \in Ag}, \mathbf{V}' \rangle$ where:

- $\mathbf{W}' = \{(w, e) \mid \mathbf{M}, w \models pre(e)\}$
- $(w, e) \rightarrow'_j (w', e')$ iff $w \rightarrow_j w'$, $\mathbf{M}, w \models \mathbf{f_j}(e, e')$, where $\mathbf{f_j}(e, e')$ denotes the filtering condition.
- $\mathbf{V}'((w, e), i, t) = \mathbf{V}(w, i, t)$, for $w \in \mathbf{W}$, $i \in Ag$, $t \in T$.

As mentioned earlier, the intuition is that when an agent posts an announcement on the social platform, the other agents in the community may update their beliefs through the filtering model. It is worth noting that condition 1 imposed on OMs, is preserved by product updates, next we show that the relation in the product model is serial, i.e., for any $i \in Ag$ and $w \in W'$, there is a world $v \in W'$, such that $w \rightarrow_i v$.

Lemma 6.1: The relation in the product model is serial.

Proof We aim to show that for any (w, e) in $\mathbf{M} \otimes \mathbf{C}$, there exists (u, e') in $\mathbf{M} \otimes \mathbf{C}$ such that $(w, e) \rightarrow_j (u, e')$.

For any (w, e) in $\mathbf{M} \otimes \mathbf{C}$, since $\mathbf{M}$ is serial, then for any agent k, there exists $u \in \mathbf{W}$ such that $w \rightarrow_k u$.

- for agent i,
  - we have $(w, e) \rightarrow'_i (u, e)$, since $\mathbf{f_i}(e, e) = \top$ for any $e \in \mathbf{C}$.
- for agent $j \in Ag - \{i\}$,
  - e is e_0, then $(w, e_0) \rightarrow'_j (u, e_0)$, since $\mathbf{f_j}(e_0, e_0) = \top$.
  - e is e_1, there are two cases. Suppose $\mathbf{M}, w \models \mathtt{F}(j, i^r_t)$, then $\mathbf{M}, w \models \hat{B}_j i^r_t$, it follows that $\mathbf{M}, v \models i^r_t$ for some $v \in \mathbf{W}$ with $w \rightarrow_j v$, thus $(v, e_1) \in \mathbf{W} \times \mathbf{E}$. Since $\mathbf{f}_j(e_1, e_1) = \mathtt{F}(j, i^r_t)$, then we have that $(w, e_1) \rightarrow'_j (v, e_1)$. Suppose $\mathbf{M}, w \models \neg\mathtt{F}(j, i^r_t)$, since $\mathbf{f}_j(e_1, e_0) = \neg\mathtt{F}(j, i^r_t)$, then $(w, e_1) \rightarrow'_j (u, e_0)$.

∎

The semantics of dynamic formulas is given as follows.

Definition 6.8 (Dynamic semantics): Given an opinion model $\mathbf{M}$ and a filtering model $\mathbf{C}$ for personalized announcements i^r_t. We define the truth of ϕ at w in $\mathbf{M}$ inductively as in Definition 6.3 with the additional clause:

$$\mathbf{M}, w \models [\mathbf{C}, e]\phi \text{ iff } \mathbf{M}, w \models pre(e) \text{ implies } \mathbf{M} \otimes \mathbf{C}, (w, e) \models \phi$$

We present an example of updated models as follows:

Example 6.5: Consider the opinion model in Example 6.1 and the filtering model in Example 6.4, and suppose that the filtering mechanism employed by the social platform is the *Feature Push* as mentioned above, then the product model $\mathbf{M}' = \langle \mathbf{W}', \{\rightarrow'_\mathbf{j}\}_{j \in Ag}, \mathbf{V}' \rangle$ is depicted in Figure 6.6.

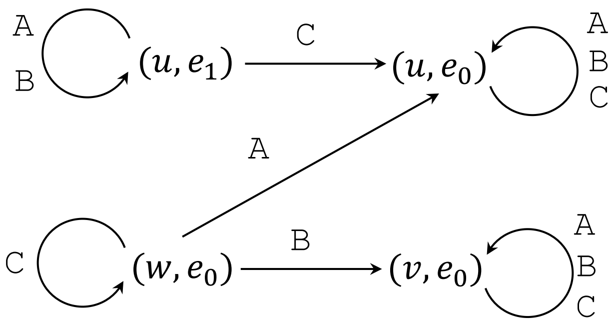


Figure 6.6 The product model

- For the domain, $\mathbf{W}' = \{(w, e_0), (u, e_0), (v, e_0), (w, e_0)\}$.
- For the relation, since $u \rightarrow_{\mathtt{C}} u$, and $\mathbf{M}, u \vDash \mathbf{f}_{\mathtt{C}}(e_1, e_0)$, where $\mathbf{f}_{\mathtt{C}}(e_1, e_0) = \neg\mathtt{F}(\mathtt{C}, \mathtt{B}^p_O) = \neg(\hat{B}_{\mathtt{C}}\mathtt{B}^p_o \wedge FP(\mathtt{C}, \mathtt{B}^p_O))$, then we have $(u, e_1) \rightarrow'_{\mathtt{C}} (u, e_0)$. Other relations can be checked in a similar fashion.
- For the valuation, $\mathbf{V}'((w, e_0), i, t) = \mathbf{V}(w, i, t)$ for $i \in \{\mathtt{A}, \mathtt{B}, \mathtt{C}\}$, which is depicted in Figure 6.2; $\mathbf{V}'((u, e_0), i, t) = \mathbf{V}'((u, e_1), i, t) = \mathbf{V}(u, i, t)$ for $i \in \{\mathtt{A}, \mathtt{B}, \mathtt{C}\}$, which is depicted in Figure 6.3; $\mathbf{V}'((v, e_0), i, t) = \mathbf{V}(v, i, t)$ for $i \in \{\mathtt{A}, \mathtt{B}, \mathtt{C}\}$, which is depicted in Figure 6.4.

6.2 Axiomatization

In this section, we propose sound and complete Hilbert-style proof systems for the logics of Section 6.1. The axiom system of the static logic *SLPA* is based on the minimal modal logic *K* with the axiom *D* for serial accessibility relations (known as *KD*), equipped with optional axioms for the corresponding access condition. The axiomatization of the dynamic logic *DLPA* expands that of the static logic with reduction axioms.

6.2.1 Complete axiomatization of SLPA

Please note that, as mentioned earlier, the filtering conditions are designed for various cases, and as such we have several axiom systems respectively. In this section, we employ the Radical Push for instance, and we will mention the axiom systems with Conservative Push or Feature Push later. In the interest of convenience, we provide the following abbreviations.

$$RP(j, i^r_t) := \bigvee_{r' \in R_t} (j^{r'}_t \wedge \bigvee_{\{i_1, \ldots, i_\theta\} \subseteq Ag} \bigwedge_{1 \leq n \leq \theta} (i^{r'}_{n_t} \leftrightarrow i^r_{n_t}))$$

$$CP(j, i_t^r) := \bigwedge_{r' \in R_t} (j_t^{r'} \to \bigvee_{\{i_1,\ldots,i_\theta\} \subseteq Ag} \bigwedge_{1 \le n \le \theta} (i_{n_t}^{r'} \leftrightarrow i_{n_t}^{r}))$$

$$FP(j, i_t^r) := \bigvee_{r' \in R_t} j_t^{r'}$$

PC
All instances of classical propositional tautologies
K
$B_i(\phi \to \psi) \to (B_i\phi \to B_i\psi)$
Consistency
$B_i\phi \to \neg B_i \neg\phi$
Positive Opinion-Belief
$i_t^r \to B_i i_t^r$
Negative Opinion-Belief
$\neg i_t^r \to B_i \neg i_t^r$
Radical Push axiom
$\mathtt{F}(j, i_t^r) \leftrightarrow \hat{B}_j i_t^r \wedge RP(j, i_t^r)$
Modus Ponens
From ϕ and $\phi \to \psi$, infer ψ
Necessitation of B_i
From ϕ infer $B_i\phi$

Table 6.1 The Hilbert-style proof system of SLPA with Radical Push axiom

Let us first discuss the axioms and rules in Table 6.1. The positive and negative Opinion-Belief axioms capture the OM condition 1 and the radical push axiom corresponds to the radical push condition. Please notice that the radical push axiom is optional, which depends on the specialized filtering condition employed by the social platform. We can substitute this axiom with any of the ones listed in Table 6.2, as needed. Other axioms and rules are for the axiom system *KD*.

Theorem 6.1: The axioms and rules in Table 6.1 form a sound and complete axiom system characterizing all valid formulas of *SLPA* over opinion models with radical push.

Proof For soundness, we can follow the standard way to show that all axioms and rules in Table 6.1 are valid. Please notice that the set of Ag and R_t are all finite, when we prove the soundness of Radical Push axiom.

For completeness, we can follow the canonical method in (Blackburn et al., 2001) to

Radical Push axiom
$\mathtt{F}(j, i^r_t) \leftrightarrow \hat{B}_j i^r_t \wedge RP(j, i^r_t)$
Conservative Push axiom
$\mathtt{F}(j, i^r_t) \leftrightarrow \hat{B}_j i^r_t \wedge CP(j, i^r_t)$
Feature Push axiom
$\mathtt{F}(j, i^r_t) \leftrightarrow \hat{B}_j i^r_t \wedge FP(j, i^r_t)$

Table 6.2 The axioms for filtering conditions

prove it. In the following, we give the definition of canonical models.

Definition 6.9: The canonical model $\mathbf{M^c}$ is a tuple $\langle S^c, \{\to^c_i\}_{i \in Ag}, V^c \rangle$ as follows.

- $S^c = \{\Gamma \mid \Gamma$ *is maximal consistent*$\}$
- $\Gamma \to^c_i \Delta$ iff for all formula α, $\alpha \in \Delta$ implies $\hat{B}_i \alpha \in \Gamma$
- $V^c(p) = \{\Gamma \in S^c \mid p \in \Gamma\}$.

It is sufficient to show the canonical model is an opinion model, since the Consistency axiom ensures the seriality of the doxastic accessibility relation, then we only need to show that the canonical model satisfies the condition 1.

Suppose that $w \to_i u$ and $\mathbf{V}(w, i, t) \neq \mathbf{V}(u, i, t)$, then there are two cases.

- there exists $r \in \mathbf{V}(w, i, t)$ and $r \notin \mathbf{V}(u, i, t)$. It follows that $\mathbf{M^c}, w \vDash i^r_t$ and $\mathbf{M^c}, u \nvDash \neg i^r_t$ by the definition of V^c, thus $\mathbf{M}, w \nvDash i^r_t \to B_i i^r_t$, which contradicts the Positive Opinion-Belief axiom.
- there exists $r \notin \mathbf{V}(w, i, t)$ and $r \in \mathbf{V}(u, i, t)$. It follows that $\mathbf{M^c}, w \nvDash i^r_t$ and $\mathbf{M^c}, u \vDash i^r_t$ by the definition of V^c, thus $\mathbf{M^c}, w \nvDash \neg i^r_t \to B_i \neg i^r_t$, which contradicts with the negative Opinion-Belief axiom.

∎

Next, we can obtain the Existence lemma and Truth lemma following the standard method, and we obtain the completeness result at last.

6.2.2 Complete axiomatization of DLPA

We will now proceed to provide a complete axiomatization for the full dynamic logic. The axiomatization is shown in Figure 6.3.

The reduction axioms play a crucial role in the completeness proof, as they allow us to reduce the dynamic properties of the models to their static counterparts. Note that all reduction axioms are valid, we only show that last two axioms are valid as follows.

Lemma 6.2: $[\mathbf{C}, e]\mathtt{F}(j, i^r_t) \leftrightarrow (pre(e) \to \mathtt{F}(j, i^r_t))$ is valid.

All axioms in Figure 6.1
Reduction axioms
$[\mathbf{C},e]i_t^r \quad \leftrightarrow \quad pre(e) \to i_t^r$
$[\mathbf{C},e](\phi_1 \wedge \phi_2) \quad \leftrightarrow \quad [\mathbf{C},e]\phi_1 \wedge [\mathbf{C},e]\phi_2$
$[\mathbf{C},e]\neg\phi \quad \leftrightarrow \quad pre(e) \to \neg[\mathbf{C},e]\phi$
$[\mathbf{C},e]\mathtt{F}(j,i_t^r) \quad \leftrightarrow \quad pre(e) \to \mathtt{F}(j,i_t^r)$
$[\mathbf{C},e]B_i\phi \quad \leftrightarrow \quad pre(e) \to \bigwedge_{e' \in E} (\mathbf{f}_i(e,e') \to B_i[\mathbf{C},e']\phi)$
All rules in Figure 6.1
Necessitation rule of $(\mathbf{C},e)$
From ϕ infer $[\mathbf{C},e]\phi$

Table 6.3 The Hilbert-style proof system of DLPA with Radical Push

Proof Note that the satisfiablility of $RP(j,i_t^r), CP(j,i_t^r)$ and $FP(j,i_t^r)$ at w are determined by $V(w,i,t)$ for $i \in Ag$, $r \in R_t$, where $t \in T$, thus $RP(j,i_t^r)$, $CP(j,i_t^r)$ and $FP(j,i_t^r)$ are satisfiable at w iff $RP(j,i_t^r), CP(j,i_t^r)$ and $FP(j,i_t^r)$ are satisfiable at (w,e) respectively, where $pre(e)$ holds at w. We prove this from two directions.

From left to right, given $\mathbf{M},w \vDash [\mathbf{C},e]\mathtt{F}(j,i_t^r)$, suppose $\mathbf{M},w \vDash pre(e)$, then $\mathbf{M} \otimes \mathbf{C},(w,e) \vDash \mathtt{F}(j,i_t^r)$. According to the truth condition of $\mathtt{F}(j,i_t^r)$, it follows that $\mathbf{M} \otimes \mathbf{C},(w,e) \vDash \hat{B}_j i_t^r$ and the parameterized access condition is satisfiable at (w,e). Since $\mathbf{M} \otimes \mathbf{C},(w,e) \vDash \hat{B}_j i_t^r$, then there exists (v,e') in $\mathbf{M} \otimes \mathbf{C}$ such that $(w,e) \to_j' (v,e')$ and $\mathbf{M} \otimes \mathbf{C},(v,e') \vDash i_t^r$. Thus we have that $w \to_j v$ and $\mathbf{M},v \vDash i_t^r$, which implies $\mathbf{M},w \vDash \hat{B}_j i_t^r$. Thus, $\mathbf{M},w \vDash \mathtt{F}(j,i_t^r)$.

From right to left, given that $\mathbf{M},w \vDash pre(e) \to \mathtt{F}(j,i_t^r)$, suppose $\mathbf{M},w \vDash pre(e)$, then $\mathbf{M},w \vDash \mathtt{F}(j,i_t^r)$. It follows that $\mathbf{M},w \vDash \hat{B}_j i_t^r$, then $\mathbf{M},u \vDash i_t^r$ for some $u \in \mathbf{W}$ with $w \to_j u$. Thus, there exists (u,e) in $\mathbf{M} \otimes \mathbf{C}$, such that $(w,e) \to_j' (u,e)$ and $\mathbf{M} \otimes \mathbf{C},(u,e) \vDash i_t^r$, which implies $\mathbf{M} \otimes \mathbf{C},(w,e) \vDash \hat{B}_j i_t^r$. Thus, $\mathbf{M} \otimes \mathbf{C},(w,e) \vDash \mathtt{F}(j,i_t^r)$. ∎

Lemma 6.3: $[\mathbf{C},e]B_i\phi \leftrightarrow (pre(e) \to \bigwedge_{e' \in E} (\mathbf{f}_i(e,e') \to B_i[\mathbf{C},e']\phi))$ is valid.

Proof Form left to right, given $\mathbf{M},w \vDash [\mathbf{C},e]B_i\phi$ [1], suppose $\mathbf{M},w \vDash pre(e)$ [2]. We want to show that $\mathbf{M},w \vDash \bigwedge_{e' \in E} (\mathbf{f}_i(e,e') \to B_i[\mathbf{C},e']\phi))$. Take any $e' \in E$ such that $\mathbf{M},w \vDash \mathbf{f}_i(e,e')$. It suffices to show that $\mathbf{M},w \vDash B_i[\mathbf{C},e']\phi)$, i.e. $\mathbf{M},w' \vDash [\mathbf{C},e']\phi)$, for all $w' \in \mathbf{W}$ such that $w \to_i w'$. That is, $\mathbf{M},w' \vDash pre(e')$ implies $\mathbf{M} \otimes \mathbf{C},(w',e') \vDash \phi$, for all $w' \in \mathbf{W}$ such that $w \to_i w'$.

From [1] we obtain: $\mathbf{M},w \vDash pre(e)$ implies $\mathbf{M} \otimes \mathbf{C},(w,e) \vDash B_i\phi$, i.e. $\mathbf{M},w \vDash pre(e)$

implies $\mathbf{M} \otimes \mathbf{C}, (w', e') \vDash \phi$, for all $(w', e') \in \mathbf{W}'$ such that $(w, e) \to_i' (w', e')$. That is, $\mathbf{M}, w \vDash pre(e)$ implies $\mathbf{M} \otimes \mathbf{C}, (w', e') \vDash \phi$, for all $(w', e') \in \mathbf{W}'$ such that $w \to_i w'$ and $\mathbf{M}, w \vDash \mathbf{f}_i(e, e')$. From [1],[2]: $\mathbf{M} \otimes \mathbf{C}, (w', e') \vDash \phi$, for all $(w', e') \in \mathbf{W}'$ such that $w \to_i w'$ and $\mathbf{M}, w \vDash \mathbf{f}_i(e, e')$ [3].

Now take arbitrary $e' \in E$ such that $\mathbf{M}, w \vDash \mathbf{f}_i(e, e')$ and arbitrary $w' \in \mathbf{W}$ such that $w \to_i w'$. Suppose that $\mathbf{M}, w' \vDash pre(e')$, i.e. $(w', e') \in \mathbf{W}'$. From [3], we obtain $\mathbf{M} \otimes \mathbf{C}, (w', e') \vDash \phi$, which is precisely what we wanted to show.

From right to left, given $\mathbf{M}, w \vDash pre(e) \to \bigwedge_{e' \in E} (\mathbf{f}_i(e, e') \to B_i[\mathbf{C}, e']\phi)$ [1], we want to show that $\mathbf{M}, w \vDash [\mathbf{C}, e]B_i\phi$, i.e. $\mathbf{M}, w \vDash pre(e)$ implies $\mathbf{M} \otimes \mathbf{C}, (w', e') \vDash \phi$, for all $(w', e') \in \mathbf{W}'$ such that $w \to_i w'$ and $\mathbf{M}, w \vDash \mathbf{f}_i(e, e')$.
Suppose $\mathbf{M}, w \vDash pre(e)$. Take arbitrary $w' \in \mathbf{W}$ and $e' \in \mathbf{E}$ such that $\mathbf{M}, w' \vDash pre(e')$, $w \to_i w'$ and $\mathbf{M}, w \vDash \mathbf{f}_i(e, e')$. From [1] we obtain that $\mathbf{M}, w \vDash B_i[\mathbf{C}, e']\phi$. As a result, $\mathbf{M}, w' \vDash [\mathbf{C}, e']\phi$, i.e. $\mathbf{M}, w' \vDash pre(e')$ implies $\mathbf{M} \otimes \mathbf{C}, (w', e') \vDash \phi$. Due to our assumption we obtain $\mathbf{M} \otimes \mathbf{C}, (w', e') \vDash \phi$ as desired. ∎

Theorem 6.2: The axioms and rules in Table 6.3 form a sound and complete axiom system characterizing the validities of *DLPA* over opinion models and (our class of) filtering models.

Proof Since all reduction axioms in Table 6.3 are valid, and the inference rule in Table 6.3 preserves the validity, thus soundness follows.

For completeness, since the reduction axioms define a validity-preserving translation from $\mathcal{L}^+$ to $\mathcal{L}$, which means for any formula in *SLPA*, there is an equivalent formula in *DLPA*. According to Theorem 6.1, the completeness follows. ∎

6.3 Conclusions and future work

In this chapter, our focus has been on agents in social platforms receiving only information that has passed the platform's filtering conditions. We have proposed a logical framework with both static and dynamic components to model the relevant events and how the personalization of announcements in social platforms affects agents' beliefs. Our treatment of the static logic of beliefs in this analysis was rather rudimentary, based on the weak axiom system *KD*. Hence, in future work we plan to extend this setting to incorporate full-fledged *KD45* belief, or by taking on board also conditional beliefs, extending our semantics to plausibility models for belief (van Benthem, 2007; Baltag and Smets, 2008).

In future and on-going work, we will address the other crucial phenomenon that we have noted in our initial discussion of the functioning of social platforms: the filtering effects of agent's attention span (Wickens, 2021). For this purpose, we can follow roughly the same methodology as that followed in this chapter. We plan to incorporate the attention of agents as a new parameter in our models, extending the more rudimentary approach of the awareness models of (Fagin and Halpern, 1987). As a first static perspective we will introduce a logic that reflects how on social platforms, given the limited attention available, agents only learn a subset of the information that they have access to. For the dynamic setting, we intend to introduce two elements: the focus of agents' attention and the cognitive cost of formulas. The attention focus will be used to discriminate which formulas an agent learns, for example, in a situation in which she has access to multiple announced formulas but has only enough cognitive resources to cover the cost of each of them separately, but not all of them together. In such a situation, the agent will only learn what is under the focus of her attention. More formally, we will extend models with attention functions, define updates of models that include attention and cost, enrich the languages used in this chapter accordingly, and prove that the resulting extended logics are sound and complete.

Finally, social scenarios where the dynamics of beliefs is affected by external information filtering and internal attention management are usually geared toward decisions that agents make or are encouraged to make. Thus, our analysis feeds into a study of personalized decision making in decision theory and game theory. Here we believe that the logic-based style of analysis in this chapter may profitably be integrated with existing logical analyses of games, whether existing or newly designed e.g., (van Benthem, 2014; van Benthem and Klein, 2022; van Benthem and Liu, 2020; Li et al., 2021; Gierasimczuk et al., 2009; Liu et al., 2016).

6.4 Appendix D: Connection between action models and edge-conditioned models

In this part, we prove a result announced but not proved in (Bolander, 2018) whose proof seems useful to have available officially. The class of model-changing updates using edge-conditioned models is actually the same as that produced by standard product update in DEL (Baltag et al., 1998; van Ditmarsch et al., 2008; van Benthem, 2011).

Firstly, we introduce the standard definitions of epistemic models, event models and product updates. Let G denote the set of agents, P denote the set of proposition letters.

Definition 6.10 (Epistemic model): An epistemic model M is a tuple (W, R, V), where

- W is a nonempty set of states,
- $R_i \subseteq W \times W$ for each $i \in G$,
- V is a valuation function from P to 2^W.

Definition 6.11 (Event model): An event model ε is a tuple $(E, S, Pre, Post)$, where

- E is a finite set of events,
- $S_i \subseteq E \times E$ for each $i \in G$,
- $Pre : E \to \mathcal{L}(P, G)$ assigns to each event a precondition.
- $Post : E \to \mathcal{L}(P, G)$ assigns to each event a postcondition. Postconditions are conjunctions of propositional literals, i.e., conjunctions of atomic propositions and their negations.

Definition 6.12 (Updated model): Let (M, w_0) be a pointed epistemic model, and (ε, e_0) be a pointed event model, and $M, w \vDash Pre(e_0)$. The updated model M^ε is a tuple (W_ε, R', V'), where

- $W_\varepsilon = \{(w, e) : M, w \vDash Pre(e)\}$,
- $(w, e)R'_j(v, f)$ iff $R_j wv$ and $S_j ef$,
- $(w, e) \in V'(p)$ iff $post(e) \vDash p$ or $(w \in V(p)$ and $post(e) \nvDash \neg p)$.

Next, we introduce the definitions of edge-conditioned event models and product updates.

Definition 6.13 (Edge-conditioned event model): An edge-conditioned event model ε is a tuple $(E, Q, Pre, Post)$, where E, $Pre, Post$ are defined as for standard event models, and $Q_i : E \times E \to \mathcal{L}(P, G)$ for each $i \in G$.

Definition 6.14 (Edge-conditioned updated model): Given a pointed epistemic model (M, w_0) and a pointed edge-conditioned event model (ε, e_0), and $M, w_0 \vDash Pre(e_0)$. The edge-conditioned updated model M^ε is a tuple (W^ε, R', V'), where W^ε, V' are defined as for standard updated models, and for each $j \in G$,

- $(w, e)R'_j(v, f)$ iff $R_j wv$ and $M, w \vDash Q_j(e, f)$.

Given an epistemic model $M = (W, R, V)$ and an event model $\varepsilon = (E, S, Pre, Post)$, then the standard DEL product update can be viewed as an edge-conditioned update.

We only need to define an edge-conditioned event $\varepsilon' = (E', Q, Pre', Post')$, where $E = E', Pre = Pre', Post = Post'$, and define the function Q_j as follows.

$$Q_j(e, e') = \begin{cases} \top & \textit{if } (e, e') \in S_j, \\ \bot & \textit{otherwise} \end{cases}$$

It is direct that $M, w \vDash Q_j(e, f)$ iff $(e, f) \in S_j$, then any standard DEL can be treated as an edge-conditioned updated model. Then we prove the other direction.

Proposition 6.1: Let $M = (W, R, V)$ be an epistemic model and $\varepsilon = (E, Q, Pre, Post)$ an edge-conditioned event model, then there is an event model ε' such that for any (w, e) in the edge-conditioned updated model M^{ε}, there is a bisimilar world in the updated model $M^{\varepsilon'}$.

Proof We construct $\varepsilon' = (E', S, Pre', Post')$ as follows.

- $E' = \{(\phi_1, ..., \phi_n, e) | \textit{for } e \in E, \exists f_i \in E \textit{ s.t. } Q_i(e, f_i) = \phi_i \textit{ for } i \in [1, n]\}$,
- $S_i(\phi_1, ..., \phi_n, e)(\psi_1, ..., \psi_n, f)$ iff $Q_i(e, f) = \phi_i$, where $\psi_i \in \Gamma_i^f$,
  $\Gamma_i^f = \{\phi_{i,j} | \exists f_j \textit{ s.t. } Q_i(f, f_j) = \phi_{i,j} \textit{ in } \varepsilon\}$,
- $Pre'(\phi_1, ..., \phi_n, e) = Pre(e) \wedge \bigwedge_{i \in [1,n]} \phi_i$,
- $Post'(\phi_1, ..., \phi_n, e) = Post(e)$.

Now, we have an update model $M^{\varepsilon'} = (W'', R'', V'')$ from M and event model ε'. Next, we show (w, e) in M^{ε} and $(w, \phi_1, ..., \phi_n, e)$ in $M^{\varepsilon'}$ are bisimilar, where $M, w \vDash \bigwedge_{i \in [1,n]} \phi_i$.

- $(w, e) \in V'(p)$ iff iff $Post(e) \vDash p$ or ($w \in V(p)$ and $Post(e) \nvDash \neg p$) iff $Post'(\phi_1, ..., \phi_n, e) \vDash p$ or ($w \in V(p)$ and $Post'(\phi_1, ..., \phi_n, e) \nvDash \neg p$) iff $(w, \phi_1, ..., \phi_n, e) \in V''(p)$.

Suppose $(w, e) R_i' (v, f)$, then we have $w R_i v$ and $M, w \vDash Q_i(e, f)$. Since $(w, \phi_1, ..., \phi_n, e) \in W''$, then $M, w \vDash \bigwedge_{i \in [1,n]} \phi_i$, thus we have $M, w \vDash \phi_i$.

Note that there is a unique formula in Γ_i^e is satisfiable at (M, w) for any w in M. (Γ_i^e consists of all possibilities under consideration by agent i for e, it's a partition of conditions. It follows that given an event e, there is a unique tuple $(\phi_1, ..., \phi_n) \in \Gamma_1^e \times ... \times \Gamma_n^e$ is satisfiable at M, w). It follows that $Q_i(e, f) = \phi_i$. According to the definition of S in ε', we have $S_i(\phi_1, ..., \phi_n, e)(\psi_1, ..., \psi_n, f)$ for any $\psi_j \in \Gamma_j^f$. It follows that we have $(w, \phi_1, ..., \phi_n, e) R_i'' (v, \psi_1, ..., \psi_n, f)$, where $M, v \vDash \bigwedge_{i \in [1,n]} \psi_i$.

For the back condition, suppose $(w, \phi_1, ..., \phi_n, e) R_i'' (v, \psi_1, ..., \psi_n, f)$, then $w R_i v$ and $(\phi_1, ..., \phi_n, e) R_i'' (\psi_1, ..., \psi_n, f)$, it follows that $Q_i(e, f) = \phi_i$. Since

$(w, \phi_1, \ldots, \phi_n, e) \in W''$, then $M, w \vDash \phi_i$, thus $(w, e) R'_i (v, f)$. By I.H. that (v, f) and $(v, \psi_1, \ldots, \psi_n, f)$ are bisimilar, then (w, e) and $(w, \phi_1, \ldots, \phi_n, e)$ are bisimilar. ∎

In the above, we demonstrated that the standard event models and the edge-conditioned event models are equivalent in terms of the class of model-changing operations they can describe. However, the edge-conditioned format is typically more succinct. Specifically, in the worst case, the event model can be exponentially less succinct than the edge-conditioned event model.

Moreover, there is an alternative approach in terms of generic language describing event models in (Belardinelli and Bolander, 2023).

CHAPTER 7 CONCLUSION AND FURTHER DIRECTIONS

7.1 Conclusion

In this thesis, we studied several game scenarios describing multi-agent interaction and related computational problems as well as model-theoretic aspects from the logical perspectives. More specifically, we addressed four major topics that we summarize below. We started off with a chapter on basic frameworks that we developed further in the subsequent chapters.

The first topic that we discussed involved sabotage games and in Chapter 3, we designed a new hybrid modal logic HSML to match this kind of games. We enriched the language of SML with additional nominals and the satisfaction operator, which enhanced the ability to characterize sabotage games. On the basis of the new language, we provided a complete *Hilbert*-style axiomatization. Taking into account the behavioral constraints that the player may have in more complicated sabotage games, we also introduced protocol models with restrictions on available edge deletions, and obtained the corresponding proof system. At the end, we clarified the connections between HSML-style logics of edge deletions and recent modal logics developed for describing stepwise point deletion in graphs (van Benthem et al., 2020).

In Chapter 4, we focused on our second topic, viz. distributed games. We used local and global models to describe these games from diverse perspectives, an essential feature involving distributed scenarios. We further proposed distributed game logics DGL and DGLEA to characterize reasoning *in* and *about* distributed games. We introduced two proof systems accordingly - for DGL we gave a strong completeness result, whereas for DGLEA we presented a weak completeness result that simplified the proof to a great extent. We also showed that both DGL and DGLEA have tractable model checking problems. We finished the chapter by proposing a similar framework to explore reasoning about strategies in distributed games.

Our third topic was explored in Chapter 5, where we zoomed out from sabotage games to a broader class of graph games, and focused on a common feature of many graph games: graph changes, that is, distinct structural and other changes in the underlying models. We extended the standard modal language with an additional operator expressing such model changes. The operator can be specified according to the model change we want to capture, for example, deletion of edges, swapping of directed edges, deletion of nodes, changes in the valuation function, among others. With respect to these languages, we concentrated on the notion of bisimulation or model comparison. We investigated this problem from an algorithmic viewpoint, providing a uniform algorithm for checking bisimulation for many such model-changing modal logics. Through our algorithmic study, we provided **PSPACE** upper bound results with respect to those modal logics.

In Chapter 6, we explored our final topic by proposing a concrete scenario in social platforms. We started off by proposing a notion of opinion models based on the work of Smets and Velázquez-Quesada (2019a). We followed up with a discussion on several mechanisms of information distribution in social platforms, and formulated them in the syntactic level. Accordingly, we proposed a static logic of personalized announcements (SLPA) and furthermore, a dynamic logic of personalized announcements (DLPA). We finished the chapter by presenting complete axiom systems for both these logics.

7.2 Further directions

The chapters also introduce new agenda items, some coming from the area of computation and some from the current social media scenarios on the internet. Many topics that remain to be investigated have been listed in our separate chapters. Of these, we would like to highlight three general themes below.

First, the study of graph games so far has concentrated largely on scenarios with perfect information where players can survey the whole graph and observe what is going on at every point. While this is realistic in many board games or simple recreational games, the assumption of complete observability quickly becomes unrealistic in practice. In that case, we must extend our study to versions of our games with *imperfect information*, where players' knowledge and ignorance about the state of the game becomes crucial (Perea, 2012; de Bruin, 2010; van Benthem, 2014). We have touched on this theme lightly in Chapter 3, in terms of extensive sabotage games, but a more thorough examination of this richer *epistemic perspective* needs to be undertaken, including what will be its effects on the main technical results obtained in this thesis.

While the preceding goal might be achieved with standard semantic tools from epistemic logic and dynamic-epistemic logic (Baltag et al., 1998; Baltag and Moss, 2004; van Ditmarsch et al., 2008; van Benthem, 2011), we also found a need for a more fine-grained view of the information available to players in the course of a game. This shows clearly in the need for an *attention dynamics* noted in Chapter 6, where players can access only limited portions of the semantic information at their disposal as provided by communication or other sources. We believe that the logical study of games as played in practice eventually needs such more fine-grained modeling, but the effects of doing so are largely unexplored, also in the broader logical literature on (in our view) closely related topics such as the dynamics of inference as an attention-driven informational process.

Finally, in a more standard game-theoretic view of the games studied in this thesis, a crucial role would be played by what players know or believe about the strategies played by the other players they are up against. Now this topic may not seem to arise in many of our games since they are determined as they stand, leading to impoverished equilibria where one player plays a winning strategy while it does not matter what the other player does. However, this can change when we impose more realistic constraints on what players are able to do, e.g., in terms of observation (Grossi and Turrini, 2012; Liu et al., 2016) or attention restrictions (Avoyan and Schotter, 2020). Moreover, it would be natural and easy to provide our graph games with more refined preferences in case we give them several goal regions arranged in some hierarchy. In all these cases more refined Nash equilibria (Osborne and Rubinstein, 1994) will arise that can be studied. Generally speaking, we expect that we will then have to make from the simpler logics of game boards that have been our main focus to richer genuine *game logics* that record the structure of the extensive games played over these boards.

REFERENCES

Aceto, L., Ingolfsdottir, A., and Srba, J. (2011). The algorithmics of bisimilarity. In *Advanced Topics in Bisimulation and Coinduction*, pages 100–172. Cambridge University Press.

Areces, C., Blackburn, P., and Marx, M. (2001). Hybrid logics: Characterization, interpolation and complexity. *Journal of Symbolic Logic*, 66:977–1010.

Areces, C., Fervari, R., and Hoffmann, G. (2012). Moving arrows and four model checking results. In *International Workshop on Logic, Language, Information, and Computation*, pages 142–153. Springer.

Areces, C., Fervari, R., and Hoffmann, G. (2014). Swap logic. *Logic Journal of IGPL*, 22(2):309–332.

Areces, C., Fervari, R., Hoffmann, G., and Martel, M. (2016). Relation-changing logics as fragments of hybrid logics. *Electronic Proceedings in Theoretical Computer Science*, 226:16–29.

Areces, C., Figueira, D., Figueira, S., and Mera, S. (2008). Expressive power and decidability for memory logics. In *Proceedings of the 15th International Workshop on Logic, Language, Information and Computation*, page 56–68. Springer-Verlag.

Areces, C. and ten Cate, B. (2007). Hybrid logics. In *Handbook of Modal Logic*, pages 821–868. Elsevier Science Publishers, Amsterdam.

Arora, S. and Barak, B. (2009). *Computational complexity: A modern approach*. Cambridge University Press.

Aucher, G. (2005). A combined system for update logic and belief revision. In *Intelligent Agents and Multi-Agent Systems*, pages 1–17. Springer Berlin Heidelberg.

Aucher, G., van Benthem, J., and Grossi, D. (2015). Sabotage modal logic: Some model and proof theoretic aspects. In *International Workshop on Logic, Rationality, and Interaction*, pages 1–13. Springer Berlin Heidelberg.

Aucher, G., van Benthem, J., and Grossi, D. (2018). Modal logics of sabotage revisited. *Journal of Logic and Computation*, 28(2):269–303.

Avoyan, A. and Schotter, A. (2020). Attention in games: An experimental study. *European Economic Review*, 124:103410.

Baier, C. and Katoen, J. P. (2008). *Principles of model checking*. MIT press.

Balcázar, J., Gabarró, J., and Sántha, M. (1992). Deciding bisimilarity is P-complete. *Formal Aspects of Computing*, 4(S1):638–648.

Baltag, A., Christoff, Z., Rendsvig, R. K., and Smets, S. (2019a). Dynamic epistemic logics of diffusion and prediction in social networks. *Studia Logica*, 107(3):489–531.

Baltag, A., Li, D., and Pedersen, M. Y. (2019b). On the right path: A modal logic for supervised learning. In *International Workshop on Logic, Rationality and Interaction*, pages 1–14. Springer.

Baltag, A. and Moss, L. S. (2004). Logics for epistemic programs. *Synthese*, 139(2):165–224.

Baltag, A., Moss, L. S., and Solecki, S. (1998). The logic of public announcements, common knowledge, and private suspicions. In *Proceedings of the 7th Conference on Theoretical Aspects of Rationality and Knowledge*, page 43–56. Morgan Kaufmann Publishers Inc.

Baltag, A. and Renne, B. (2016). Dynamic epistemic logic. *The Stanford Encyclopedia of Philosophy. Retrieved from https://plato.stanford.edu/archives/win2016/entries/dynamic-epistemic*.

Baltag, A. and Smets, S. (2008). A qualitative theory of dynamic interactive belief revision. In *Logic and the Foundations of Game and Decision Theory (LOFT 7)*, pages 11–58. Amsterdam University Press.

Barwise, J. and Seligman, J. (1997). *Information Flow: The Logic of Distributed Systems*. Cambridge University Press.

Belardinelli, G. and Bolander, T. (2023). Attention! Dynamic epistemic logic models of (in)attentive agents. In *Proceedings of the 2023 International Conference on Autonomous Agents and Multiagent Systems*, page 391–399. ACM.

Belle, V. and Lakemeyer, G. (2010). Reasoning about imperfect information games in the epistemic situation calculus. In *Proceedings of the Twenty-Fourth AAAI Conference on Artificial Intelligence*, page 255–260. AAAI Press.

Berwanger, D., Mathew, A. B., and van den Bogaard, M. (2018). Hierarchical information and the synthesis of distributed strategies. *Acta Informatica*, 55(8):669–701.

Beutner, R., Finkbeiner, B., and Hecking-Harbusch, J. (2019). Translating asynchronous games for distributed synthesis. In *30th International Conference on Concurrency Theory*. Schloss Dagstuhl - Leibniz-Zentrum für Informatik.

Blackburn, P., de Rijke, M., and Venema, Y. (2001). *Modal Logic*. Cambridge University Press.

Blackburn, P. and Ten Cate, B. (2006). Pure extensions, proof rules, and hybrid axiomatics. *Studia Logica*, 84:277–322.

Bolander, T. (2018). Seeing is believing: Formalising false-belief tasks in dynamic epistemic logic. In *Jaakko Hintikka on knowledge and game-theoretical semantics*, pages 207–236. Springer.

Bonanno, G. (2001). Branching time, perfect information games, and backward induction. *Games and Economic Behavior*, 36:57–73.

Bunte, O., Groote, J. F., Keiren, J. J. A., Laveaux, M., Neele, T., de Vink, E. P., Wesselink, W., Wijs, A., and Willemse, T. A. C. (2019). The mCRL2 toolset for analysing concurrent systems. In *Tools and Algorithms for the Construction and Analysis of Systems*, pages 21–39. Springer International Publishing.

Cleaveland, R., Parrow, J., and Steffen, B. (1993). The concurrency workbench: A semantics-based tool for the verification of concurrent systems. *ACM Transactions on Programming Languages and Systems*, 15(1):36–72.

Coulouris, G., Dollimore, J., Kindberg, T., and Blair, G. (2012). *Distributed Systems: Concepts and Design*. Addison-Wesley Longman Publishing Co., Inc.

Das, R. and Ramanujam, R. (2021). A logical description of strategizing in social network games. *Logics for New-Generation AI*, page 107.

de Bruin, B. (2010). *Explaining Games: The Epistemic Programme in Game Theory*. Springer.

Duchet, P. and Meyniel, H. (1993). Kernels in directed graphs: a poison game. *Discrete Mathematics*, 115(1-3):273–276.

Fagin, R. and Halpern, J. Y. (1987). Belief, awareness, and limited reasoning. *Artificial Intelligence*, 34(1):39–76.

Fervari, R. (2014). *Relation-Changing Modal Logics*. PhD thesis, Universidad Nacional de Córdoba.

Fine, K. (1975). Normal forms in modal logic. *Notre Dame Journal of Formal Logic*, 16(2):229–237.

Flaxman, S., Goel, S., and Rao, J. M. (2016). Filter Bubbles, Echo Chambers, and Online News Consumption. *Public Opinion Quarterly*, 80(S1):298–320.

Garavel, H., Lang, F., Mateescu, R., and Serwe, W. (2013). CADP 2011: A toolbox for the construction and analysis of distributed processes. *International Journal on Software Tools for Technology Transfer*, 15(2):89–107.

Gerbrandy, J. and Groeneveld, W. (1997). Reasoning about information change. *Journal of Logic, Language and Information*, 6:147–169.

Ghosh, S., Konar, N., and Ramanujam, R. (2017). Strategy composition in dynamic games with simultaneous moves. In *International Conference on Agents and Artificial Intelligence*, pages 624–631. Springer Cham.

Gierasimczuk, N., Kurzen, L., and Velázquez-Quesada, F. R. (2009). Learning and teaching as a game: A sabotage approach. In *Proceedings of the Third International Conference on Logic, Rationality, and Interaction*, pages 119–132. Springer, Heidelberg.

Girish, D. (2023). 'The Social Dilemma' Review: Unplug and Run. *The New York Times. Retrieved from https://www.nytimes.com/2020/09/09/movies/the-social-dilemma-review.html*.

Grädel, E. (2011). Back and forth between logic and games. In *Lectures in Game Theory for Computer Scientists*, pages 99–145. Cambridge University Press.

Grossi, D. and Rey, S. (2019). Credulous acceptability, poison games and modal logic. In *Proceedings of the 18th International Conference on Autonomous Agents and MultiAgent Systems*, page 1994–1996. International Foundation for Autonomous Agents and Multiagent Systems (IFAAMAS).

Grossi, D. and Turrini, P. (2012). Short sight in extensive games. In *Adaptive Agents and Multi-Agent Systems*, pages 805–812.

Halpern, J. Y. and Moses, Y. (1992). A guide to completeness and complexity for modal logics of knowledge and belief. *Artificial intelligence*, 54(3):319–379.

Harrenstein, P., Van der Hoek, W., Meyer, J. J., and Witteveen, C. (2003). A modal characterization of Nash equilibrium. *Fundamenta Informaticae*, 57(2-4):281–321.

Holliday, W. H., Hoshi, T., and Icard, T. F. (2011). Schematic validity in dynamic epistemic logic: Decidability. In *Proceedings of the Third International Conference on Logic, Rationality, and Interaction*, pages 87–96.

Hopcroft, J. E., Motwani, R., and Ullman, J. D. (2001). Introduction to automata theory, languages, and computation. *SIGACT News*, 32(1):60–65.

Hoshi, T. (2014). Public announcement logics with constrained protocols. In *Proceedings of the 8th Conference on Logic and the Foundations of Game and Decision Theory*. Springer-Verlag.

Jannach, D., Zanker, M., Felfernig, A., and Friedrich, G. (2010). *Recommender Aystems: An Introduction*. Cambridge University Press.

Kanellakis, P. C. and Smolka, S. A. (1983). CCS expressions, finite state processes, and three problems of equivalence. In *Proceedings of the Second Annual ACM Symposium on Principles of Distributed Computing*, page 228–240. Association for Computing Machinery.

Kooi, B. (2007). Expressivity and completeness for public update logics via reduction axioms. *Journal of Applied Non-Classical Logics*, 17:231 – 253.

Kooi, B. and Renne, B. (2011). Arrow update logic. *The Review of Symbolic Logic*, 4(4):536–559.

Kvasov, D. (2016). On sabotage games. *Operation Research Letters*, 44(2):250–254.

Li, D. (2020). Losing connection: The modal logic of definable link deletion. *Journal of Logic and Computation*, 30(3):715–743.

Li, D., Ghosh, S., and Liu, F. (2023). Action-information interplay in the cops and robber game. Manuscript.

Li, D., Ghosh, S., Liu, F., and Tu, Y. (2021). On the subtle nature of a simple logic of the hide and seek game. In *Logic, Language, Information, and Computation*, pages 201–218. Springer International Publishing.

Liu, C., Liu, F., Su, K., and Zhu, E. (2016). A logical characterization of extensive games with short sight. *Theoretical Computer Science*, 612:63–82.

Liu, F. and Li, D. (2022). Ten-year history of social network logics in china. *Asian Studies*, 10:121–146.

Liu, F. and Liao, B. (2021). Reasoning in social settings. *Journal of Logic and Computation*, 31(4):1023–1025.

Löding, C. and Rohde, P. (2003a). Model checking and satisfiability for sabotage modal logic. In *FST TCS 2003: Foundations of Software Technology and Theoretical Computer Science*, pages 302–313. Springer Berlin Heidelberg.

Löding, C. and Rohde, P. (2003b). Solving the sabotage game is Pspace-hard. In *Mathematical Foundations of Computer Science 2003*, pages 531–540. Springer, Berlin.

Mera, S. F. (2009). *Modal Memory Logics*. PhD thesis, Université Henri Poincaré-Nancy 1.

Mierzewski, K. (2018). Sabotage games in the random graph, Department of Philosophy, Stanford University. Working paper.

Mohalik, S. and Walukiewicz, I. (2003). Distributed games. In *FST TCS 2003: Foundations of Software Technology and Theoretical Computer Science*, pages 338–351. Springer Berlin Heidelberg.

Muscholl, A. and Schewe, S. (2013). Unlimited decidability of distributed synthesis with limited missing knowledge. In *Mathematical Foundations of Computer Science 2013*, pages 691–703. Springer Berlin Heidelberg.

Nowakowski, R. and Winkler, P. (1983). Vertex-to-vertex pursuit in a graph. *Discrete Mathematics*, 43(2-3):235–239.

Osborne, M. J. and Rubinstein, A. (1994). *A Course in Game Theory*. MIT press.

Paige, R. and Tarjan, R. E. (1987). Three partition refinement algorithms. *SIAM Journal on Computing*, 16(6):973–989.

Paul, K. (2019). Instagram tests hiding how many people like a post. that has influencers worried. *The Guardian. Retrieved from https://www.theguardian.com/technology/2019/nov/15/instagram-likes-influencers-social-media*.

Pedersen, M. Y., Smets, S., and Ågotnes, T. (2020). Further steps towards a logic of polarization in social networks. In *Logic and Argumentation*, pages 324–345. Springer International Publishing.

Perea, A. (2012). *Epistemic Game Theory: Reasoning and Choice*. Cambridge University Press.

Plaza, J. A. (1989). Logics of public communications. In *Proceedings of the 4th International Symposium on Methodologies for Intelligent Systems (ISMIS 1989)*, page 201–216. Oak Ridge National Laboratory, ORNL/DSRD-24.

Plaza, J. A. (2007). Logics of public communications. *Synthese*, 158:165–179.

Ramanujam, R. (1996). Locally linear time temporal logic. In *Proceedings 11th Annual IEEE Symposium on Logic in Computer Science*, pages 118–127. IEEE Computer Society.

Ramanujam, R. and Simon, S. (2010). A communication based model for games of imperfect information. In *International Conference on Concurrency Theory*, pages 509–523. Springer.

Renardel de Lavalette, G. R. (2004). Changing modalities. *Journal of Logic and Computation*, 14(2):251–275.

Rohde, P. (2005). *On Games and Logics over Dynamically Changing Structures*. PhD thesis, Rheinisch-Westfälische Technische Hochschule Aachen.

Seligman, J., Liu, F., and Girard, P. (2011). Logic in the community. In *Logic and Its Applications*, pages 178–188. Springer Berlin Heidelberg.

Sipser, M. (1996). Introduction to the theory of computation. *ACM Sigact News*, 27(1):27–29.

Smets, S. and Velázquez-Quesada, F. R. (2017). How to make friends: A logical approach to social group creation. In *International Workshop on Logic, Rationality, and Interaction*, pages 377–390. Springer Berlin Heidelberg.

Smets, S. and Velázquez-Quesada, F. R. (2018). The creation and change of social networks: A logical study based on group size. In *Dynamic Logic. New Trends and Applications*, pages 171–184. Springer International Publishing.

Smets, S. and Velázquez-Quesada, F. R. (2019a). A logical analysis of the interplay between social influence and friendship selection. In *International Workshop on Dynamic Logic*, pages 71–87. Springer.

Smets, S. and Velázquez-Quesada, F. R. (2019b). A logical study of group-size based social network creation. *Journal of Logical and Algebraic Methods in Programming*, 106:117–140.

Smets, S. and Velázquez-Quesada, F. R. (2020). A closeness-and priority-based logical study of social network creation. *Journal of Logic, Language and Information*, 29(1):21–51.

Solaki, A. (2021). *Logical Models for Bounded Reasoners*. PhD thesis, University of Amsterdam.

Thiagarajan, P. and Walukiewicz, I. (2002). An expressively complete linear time temporal logic for Mazurkiewicz traces. *Information and Computation*, 179(2):230–249.

Thompson, D. (2020). Local fact change logic. In *Knowledge, Proof and Dynamics*, pages 73–96. Springer Singapore.

van Benthem, J. (1996). *Exploring Logical Dynamics*. CSLI Publications.

van Benthem, J. (2001). Games in dynamic-epistemic logic. *Bulletin of Economic Research*, 53(4):219–248.

van Benthem, J. (2002). Extensive games as process models. *Journal of logic, language and information*, 11(3):289–313.

van Benthem, J. (2005). 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. Springer, Heidelberg.

van Benthem, J. (2007). Dynamic logic for belief revision. *Journal of Applied Non-Classical Logics*, 17:129–155.

van Benthem, J. (2010). *Modal Logic for Open Minds*. CSLI Publications.

van Benthem, J. (2011). *Logical Dynamics of Information and Interaction*. Cambridge University Press.

van Benthem, J. (2014). *Logic in Games*. MIT press.

van Benthem, J., Gerbrandy, J., Hoshi, T., and Pacuit, E. (2009). Merging frameworks for interaction. *Journal of Philosophical Logic*, 38:491–526.

van Benthem, J. and Klein, D. (2022). Logics for Analyzing Games. *The Stanford Encyclopedia of Philosophy. Retrieved from https://plato.stanford.edu/archives/win2022/entries/logics-for-games*.

van Benthem, J., Li, L., Shi, C., and Yin, H. (2022). Hybrid sabotage modal logic. *Journal of Logic and Computation*. exac006.

van Benthem, J. and Liu, F. (2007). Dynamic logic of preference upgrade. *Journal of Applied Non-Classical Logics*, 17(2):157–182.

van Benthem, J. and Liu, F. (2020). Graph games and logic design. In *Knowledge, Proof and Dynamics*, pages 125–146. Springer, Singapore.

van Benthem, J., Mierzewski, K., and Zaffora Blando, F. (2020). The modal logic of stepwise removal. *The Review of Symbolic Logic*, page 1–28.

van Benthem, J., van Eijck, J., and Kooi, B. P. (2006). Logics of communication and change. *Information and Computation*, 204:1620–1662.

van Ditmarsch, H., van der Hoek, W., and Kooi, B. (2008). *Dynamic Epistemic Logic*. Springer, Dordrecht.

van Ditmarsch, H., van Eijck, J., Sietsma, F., and Wang, Y. (2012). On the logic of lying. In *Games, Actions and Social Software: Multidisciplinary Aspects*, pages 41–72. Springer Berlin Heidelberg.

van Ditmarsch, H. P., van der Hoek, W., and Kooi, B. P. (2005). Dynamic epistemic logic with assignment. In *Proceedings of the Fourth International Joint Conference on Autonomous Agents and Multiagent Systems*, page 141–148. Association for Computing Machinery.

van Steen, M. and Tanenbaum, A. S. (2017). *Distributed Systems*. CreateSpace Independent Publishing Platform.

Vīķe-Freiberga, V., Däubler-Gmelin, H., Hammersley, B., and Maduro, L. M. P. P. (2013). A free and pluralistic media to sustain european democracy. European Commission.

Wang, Y. (2010). *Epistemic Modelling and Protocol Dynamics*. PhD thesis, University of Amsterdam.

Wickens, C. (2021). Attention: Theory, principles, models and applications. *International Journal of Human–Computer Interaction*, 37(5):403–417.

Wikipedia contributors (2023). The social dilemma. *Wikipedia, The Free Encyclopedia. Retrieved from https://en.wikipedia.org/w/index.php?title=The_Social_Dilemma&oldid=1152378558*.

Zaffora Blando, F., Mierzewski, K., and Areces, C. (2020). The modal logics of the poison game. In *Knowledge, Proof and Dynamics*, pages 3–23. Springer Singapore.

Ågotnes, T. and Ditmarsch, H. (2011). What will they say?—Public announcement games. *Synthese*, 179:57–85.

摘 要

博弈对于社会互动来说是一种强大的范式, 同时它对于分析逻辑推理和计算中的核心概念而言，也是好的模型。博弈提供了一个多功能平台，用以模拟各种场景。探索博弈建模，从玩家和建模者的视角分析与博弈相关的推理，研究不同信息获取方式对主体认知活动的影响，以及解决与博弈相关的计算问题，都具有实用价值和学术价值。本文特别探讨博弈图、博弈“棋盘”变化和在不同场景下对博弈元素的逻辑分析。

首先，我们特别设计了相对应的逻辑系统，来分析两类博弈。我们考虑的第一类博弈是图博弈，特别是破坏式博弈。这些博弈通过在博弈过程中改变作为博弈“棋盘”的图，模拟了主体通过消除错误路径向某个既定目标前进的场景。破坏式模态逻辑对这类博弈进行了行之有效的分析，但是公理化问题依然悬而未决。我们从混合逻辑中借助恰如其分的表达力工具对破坏式模态逻辑语言进行小幅扩张，从而为有效性提供了一个完全的公理化形式。

我们接下来研究的博弈类型，也就是分布式博弈，关注了互动情境中相当特别的方面：在博弈进行时，玩家对博弈的局部性的内部视角与建模者对博弈的全局性的外部视角之间的差异。在本文的第四章中，我们使用特殊的逻辑语言研究这些“分布式博弈”，这些语言使得我们准确地描述局部和全局视角，并详细地展示它们如何相互作用。

本文的剩余主题进一步探索了两个方向。

首先，在第五章中，我们注意到，那些目前被我们发展的特殊博弈逻辑，也就是第一部分中的破坏式模态逻辑，可以看作是一类更广泛的逻辑的实例，这类逻辑带有的模态算子可以描述各种模型变化操作的效果。这类逻辑已经被用于建模行动以及信息流，就这点而言，不仅在具体的系统上，而且在一般的模型论和证明论主题意义上，相关文献都广泛存在。我们探讨给定的有穷模型之间的相应的互模拟问题的复杂性究竟如何?

本论文的最后一个主题涉及到我们第一部分关于博弈场景的逻辑的另一个延伸。我们进行了一个实际案例研究，局部和全局的多主体视角都在此案例中进行了实际展示：即在全局的推荐系统的运作中，局部的个体玩家参与交互。在第六章中，我们展示了如何使用沟通交流涉及过滤动作下的动态认知逻辑来完全规范和分析过滤动态。

最后，我们强调了一些尚未解决的问题，以便在现有研究的基础上进一步探索。

这些问题涉及不完美信息的博弈，信息充足情形下的注意力动态变化，博弈中的策略推理，等等。

关键词：图博弈；分布式博弈；模型变化的逻辑；互模拟；过滤机制

ABSTRACT

Games are a powerful paradigm for social interaction, but at the same time also good models for analyzing crucial notions in logical reasoning and computation. They provide a versatile platform for simulating diverse scenarios. Exploring game modeling, analyzing game-related reasoning from both player and modeler perspectives, investigating the impact of varying information access on agents' epistemic activities, and addressing computational problems related to games all hold both practical and academic significance. This dissertation specifically delves into game graphs, game board change, and the logical analysis of game elements across various scenarios.

First, we analyze two sorts of games in terms of especially designed corresponding logical systems. The first kind of games we consider are graph games: in particular, sabotage games. These games where the graph that serves as the game board can change in the course of play model scenarios where agents are pushed toward some desirable goal by removing false paths. Sabotage modal logics have provided effective analysis for such games, but the problem of axiomatization remains unresolved, we provide a complete axiomatization for the validities in the language of sabotage modal logic slightly extended with just enough expressive devices from hybrid logic.

Our next kind of game, i.e., the distributed game, concerns quite different aspects of interactive scenarios: the difference between players' local internal view and the modeler‘s global external view of the game as it proceeds. In Chapter 4 of this thesis we study these 'distributed games' with special logical languages allowing us to describe local and global perspectives precisely, and show in detail how they interact.

The remaining topics of the thesis explore two further directions.

First, in Chapter 5 we note that the special game logics developed so far, i.e., sabotage model logics in our first part, can be seen as instances of a much broader class of logics with modalities that describe the effects of various operations of model change. Such logics have been used for modeling both action and information flow, and there is a broad literature on both specific systems and general model-theoretic and proof-theoretic themes running through all of these. We explore what is the precise complexity of testing for the appropriate notions of bisimulation between given finite models.

Our final topic in this thesis concerns another extension of the concerns in our first part on logics for game scenarios. We undertake a practical case study where local and global multi-agent perspectives play in practice: namely, in the functioning of global recommender systems interacting with local individual users. In Chapter 6 we show how the filtration dynamics can be specified and analyzed completely in dynamic-epistemic logics of communication involving filtering actions.

Finally, we highlight some unresolved issues for further exploration based on existing research. These involve games with imperfect information, attention dynamics in situations with abundant information, strategic reasoning in games, and so forth.

Keywords: graph game; distributed game; model-changing logic; bisimulation; filtering mechanism

SAMENVATTING

Spelen zijn een krachtig paradigma voor sociale interactie, maar tegelijkertijd ook goede modellen voor het analyseren van cruciale begrippen in logisch redeneren en rekenen met computers. Ze bieden een veelzijdig platform voor het simuleren van een breed scala van interactieve scenario's. Het verkennen van spelmodellen, het analyseren van spelgerelateerd redeneren vanuit zowel het perspectief van de spelers als de externe modelleur, het onderzoeken van de effecten van verschillen in toegang tot informatie op de epistemische activiteiten van agenten, en het exploreren van computationele problemen met betrekking tot spelen hebben zowel praktisch als academisch belang. Dit proefschrift onderzoekt specifiek spelen gebaseerd op grafen, de dynamiek van veranderingen in het spelbord gedurende het spel, en de logische analyse van strategische aspecten van spelen in verschillende scenario's.

Om te beginnen analyseren we twee soorten spelen in termen van speciaal ontworpen bijbehorende logische systemen. Het eerste soort spellen dat we bestuderen zijn graafspelen, in het bijzonder sabotagespelen. Deze spelen, waarbij een graaf dient als het speelbord dat kan veranderen tijdens het spel, modelleren scenario's waarin agenten naar een wenselijk doel worden gedreven door vruchteloze paden te verwijderen. 'Sabotage modale logica's' in de literatuur hebben reeds effectieve analyses geleverd voor dergelijke spelen, maar het probleem van axiomatisering der geldige redeneerprincipes bleef nog onopgelost. In Hoofdstuk 3 van dit proefschrift presenteren wij een volledige axiomatisering voor de geldigheden van de sabotage modale logica, met gebruik van een taal die lichtelijk is uitgebreid met enkele syntactische operatoren uit de zogenaamde 'hybride logica'.

Ons tweede genre, namelijk het gedistribueerde spel, heeft betrekking op geheel andere aspecten van interactieve scenario's: het verschil tussen de lokale interne kijk van spelers en de globale externe kijk van de modelbouwer op het spel terwijl het plaats vindt. In Hoofdstuk 4 van dit proefschrift bestuderen we deze 'gedistribueerde spelen' met speciale logische talen waarmee we lokale en globale perspectieven nauwkeurig kunnen beschrijven, en laten we gedetailleerd zien hoe ze met elkaar in wisselwerking staan.

De verdere onderwerpen van het proefschrift verkennen de volgende twee richtingen.

Ten eerste merken we in Hoofdstuk 5 op dat de speciale spellogica's die tot nu toe zijn ontwikkeld in dit proefschrift, zoals de modale sabotage logica's in ons eerste deel, gezien kunnen worden als instanties van een veel bredere klasse van logica's met modaliteiten die de effecten van verschillende operaties van modelverandering beschrijven. Dergelijke logica's zijn gebruikt voor het modelleren van zowel acties als informatiestromen, en er is een brede literatuur over zowel specifieke systemen als algemene modeltheoretische en bewijstheoretische thema's die door al deze systemen heen lopen. We onderzoeken wat de precieze complexiteit is van het testen van de juiste noties van bisimulatie tussen gegeven eindige modellen, iets wat kan worden gezien als een logisch geïnspireerde analyse van spelequivalentie.

Ons laatste onderwerp in dit proefschrift betreft een andere uitbreiding van de onderwerpen in ons eerste deel over logica's voor spelscenario's. We richten de aandacht op een praktische case-studie waarin lokale en globale perspectieven van verchillende actoren concreet spelen: namelijk, in het functioneren van internet aanbevelingssystemen die in wisselwerking staan met lokale individuele gebruikers. In Hoofdstuk 6 laten we zien hoe de dynamiek van filteren door een sociaal platform volledig kan worden gespecificeerd en geanalyseerd in passend uitgebreide dynamisch-epistemische logica's van communicatie.

Tot slot belichten we enkele onopgeloste brede kwesties voor verder onderzoek op basis van de resultaten gepresenteerd in dit proefschrift. Deze kwesties omvatten spelen met onvolledige informatie, dynamiek van aandacht in situaties met een overdaad aan informatie, strategisch redeneren in spelen, en diverse andere richtingen.

Trefwoorden: graafspel; gedistribueerd spel; logica van modelverandering; bisimulatie; filtermechanisme

ACKNOWLEDGEMENTS

First and foremost, I would like to express my deepest gratitude to my supervisors, Johan van Benthem, Sonja Smets, and Sujata Ghosh. I am deeply grateful to Johan. From him, I not only acquired a wealth of knowledge in logic, but also learned the art of conducting research. I am truly inspired by his expertise and personal charm. I extend my sincere thanks to Sonja for being my supervisor at UvA and guiding my studies at ILLC. Her impressive work in various domains left a profound impact on me, and her guidance and support expanded both my research and academic horizons. I am also indebted to Sujata, who dedicated considerable time and effort in guiding me. Her mentorship has been invaluable, and I have witnessed in her the multifaceted qualities that make an outstanding scholar.

All of them have provided immense assistance throughout my PhD studies and the dissertation process. Without their feedback, help, and meticulous guidance, this work would not have been possible.

In addition, I must express my gratitude to Fenrong Liu for providing me with the opportunity to embark on my doctoral research and providing critical direction and perspective in the early stages of my study. Her guidance and all-encompassing support have been instrumental in shaping my academic journey. Without her invaluable mentorship, I cannot imagine what my doctoral career would have been like.

I would also like to thank the members of the committee, Qi Feng, Malvin Gattinger, Davide Grossi, Fenrong Liu, Robert van Rooij and Donghua Zhu.

I express my thanks to all co-authors of the papers included in my dissertation: Gaia Belardinelli, Sujata Ghosh, Shreyas Gupta, Fenrong Liu, Chenwei Shi, Sonja Smets, Anthia Solaki, R. Ramanujam, Johan van Benthem, Haoxuan Yin.

I am appreciative of other professors who taught me so much at the beginning of my doctoral career: Jeremy Seligman, Martin Stokhof, Dag Westerståhl and Junhua Yu. Additionally, I extend my gratitude to more professors who have taught or supported me during my doctoral journey: Alexandru Baltag, Nick Bezhanishvili, Beishui Liao, Yde Venema, Lu Wang, Junwei Yu and Yan Zhang.

To the numerous individuals who have continuously supported me during my doctoral career, I extend my heartfelt thanks. Special thanks to my two classmates, Yiyan Wang and Jialiang Yan, for their meaningful discussions and sincere help. Additionally, I extend my appreciation to Qian Chen, Fengxiang Cheng, Mingliang Chu, Penghao Du, Rui Fan, Xiaoxuan Fu, Yinlin Guan, Qingyu He, Zhenkun Hu, Dazhu Li, Xuan Li, Yuqi Liu, Yang Sun, Fei Xue, Lingyuan Ye and many others.

Lastly, I am deeply thankful to my family, especially my parents, Shaogong Li and Guirong Yang, and my significant other, Yichen Liu, for their unwavering love and support throughout this journey.

RÉSUMÉ AND ACADEMIC ACHIEVEMENTS

Résumé

Lei Li was born on the 2nd of Dec 1992 in Henan, China. He began his bachelor's study in School of Mathematics and Statistics, Henan University in September 2010, majoring in Information and Computing Science, and got a Bachelor of Science degree in June 2014. He began his master's study in the Institute of Philosophy, Renmin University in September 2015, and got a Master of Philosophy degree in Logic in June 2019. In September 2019, he started to pursue a doctorate in the Department of Philosophy, Tsinghua University. In January 2021, he was admitted to the jointly awarded doctorate program of Tsinghua University and the Institute for Logic, Language and Computation, University of Amsterdam.

Academic Achievements

[1]. Johan van Benthem, Lei Li, Chenwei Shi, Haoxuan Yin (2022). Hybrid sabotage modal logic. *Journal of Logic and Computation*, exac006, https://doi.org/10.1093/logcom/exac006.

[2]. Sujata Ghosh, Shreyas Gupta, Lei Li (2022). Bisimulation in model-changing modal logics: An algorithmic study. *Journal of Logic and Computation*, *Accepted*.

[3]. Sujata Ghosh, Lei Li, Fenrong Liu, R. Ramanujam (2023). A modal logic to reason in distributed games. *Manuscript*.

[4]. Gaia Belardinelli, Lei Li, Sonja Smets, Anthia Solaki (2023). Logics for personalized announcements. *Manuscript*.

Titles in the ILLC Dissertation Series:

ILLC DS-2018-04: **Jelle Bruineberg**
Anticipating Affordances: Intentionality in self-organizing brain-body-environment systems

ILLC DS-2018-05: **Joachim Daiber**
Typologically Robust Statistical Machine Translation: Understanding and Exploiting Differences and Similarities Between Languages in Machine Translation

ILLC DS-2018-06: **Thomas Brochhagen**
Signaling under Uncertainty

ILLC DS-2018-07: **Julian Schlöder**
Assertion and Rejection

ILLC DS-2018-08: **Srinivasan Arunachalam**
Quantum Algorithms and Learning Theory

ILLC DS-2018-09: **Hugo de Holanda Cunha Nobrega**
Games for functions: Baire classes, Weihrauch degrees, transfinite computations, and ranks

ILLC DS-2018-10: **Chenwei Shi**
Reason to Believe

ILLC DS-2018-11: **Malvin Gattinger**
New Directions in Model Checking Dynamic Epistemic Logic

ILLC DS-2018-12: **Julia Ilin**
Filtration Revisited: Lattices of Stable Non-Classical Logics

ILLC DS-2018-13: **Jeroen Zuiddam**
Algebraic complexity, asymptotic spectra and entanglement polytopes

ILLC DS-2019-01: **Carlos Vaquero**
What Makes A Performer Unique? Idiosyncrasies and commonalities in expressive music performance

ILLC DS-2019-02: **Jort Bergfeld**
Quantum logics for expressing and proving the correctness of quantum programs

ILLC DS-2019-03: **András Gilyén**
Quantum Singular Value Transformation & Its Algorithmic Applications

ILLC DS-2019-04: **Lorenzo Galeotti**
The theory of the generalised real numbers and other topics in logic

ILLC DS-2019-05: **Nadine Theiler**
Taking a unified perspective: Resolutions and highlighting in the semantics of attitudes and particles

ILLC DS-2019-06: **Peter T.S. van der Gulik**
Considerations in Evolutionary Biochemistry

ILLC DS-2019-07: **Frederik Möllerström Lauridsen**
Cuts and Completions: Algebraic aspects of structural proof theory

ILLC DS-2020-01: **Mostafa Dehghani**
Learning with Imperfect Supervision for Language Understanding

ILLC DS-2020-02: **Koen Groenland**
Quantum protocols for few-qubit devices

ILLC DS-2020-03: **Jouke Witteveen**
Parameterized Analysis of Complexity

ILLC DS-2020-04: **Joran van Apeldoorn**
A Quantum View on Convex Optimization

ILLC DS-2020-05: **Tom Bannink**
Quantum and stochastic processes

ILLC DS-2020-06: **Dieuwke Hupkes**
Hierarchy and interpretability in neural models of language processing

ILLC DS-2020-07: **Ana Lucia Vargas Sandoval**
On the Path to the Truth: Logical & Computational Aspects of Learning

ILLC DS-2020-08: **Philip Schulz**
Latent Variable Models for Machine Translation and How to Learn Them

ILLC DS-2020-09: **Jasmijn Bastings**
A Tale of Two Sequences: Interpretable and Linguistically-Informed Deep Learning for Natural Language Processing

ILLC DS-2020-10: **Arnold Kochari**
Perceiving and communicating magnitudes: Behavioral and electrophysiological studies

ILLC DS-2020-11: **Marco Del Tredici**
Linguistic Variation in Online Communities: A Computational Perspective

ILLC DS-2020-12: **Bastiaan van der Weij**
Experienced listeners: Modeling the influence of long-term musical exposure on rhythm perception

ILLC DS-2020-13: **Thom van Gessel**
Questions in Context

ILLC DS-2020-14: **Gianluca Grilletti**
Questions & Quantification: A study of first order inquisitive logic

ILLC DS-2020-15: **Tom Schoonen**
Tales of Similarity and Imagination. A modest epistemology of possibility

ILLC DS-2020-16: **Ilaria Canavotto**
Where Responsibility Takes You: Logics of Agency, Counterfactuals and Norms

ILLC DS-2020-17: **Francesca Zaffora Blando**
Patterns and Probabilities: A Study in Algorithmic Randomness and Computable Learning

ILLC DS-2021-01: **Yfke Dulek**
Delegated and Distributed Quantum Computation

ILLC DS-2021-02: **Elbert J. Booij**
The Things Before Us: On What it Is to Be an Object

ILLC DS-2021-03: **Seyyed Hadi Hashemi**
Modeling Users Interacting with Smart Devices

ILLC DS-2021-04: **Sophie Arnoult**
Adjunction in Hierarchical Phrase-Based Translation

ILLC DS-2021-05: **Cian Guilfoyle Chartier**
A Pragmatic Defense of Logical Pluralism

ILLC DS-2021-06: **Zoi Terzopoulou**
Collective Decisions with Incomplete Individual Opinions

ILLC DS-2021-07: **Anthia Solaki**
Logical Models for Bounded Reasoners

ILLC DS-2021-08: **Michael Sejr Schlichtkrull**
Incorporating Structure into Neural Models for Language Processing

ILLC DS-2021-09: **Taichi Uemura**
Abstract and Concrete Type Theories

ILLC DS-2021-10: **Levin Hornischer**
Dynamical Systems via Domains: Toward a Unified Foundation of Symbolic and Non-symbolic Computation

ILLC DS-2021-11: **Sirin Botan**
Strategyproof Social Choice for Restricted Domains

ILLC DS-2021-12: **Michael Cohen**
Dynamic Introspection

ILLC DS-2021-13: **Dazhu Li**
Formal Threads in the Social Fabric: Studies in the Logical Dynamics of Multi-Agent Interaction

ILLC DS-2022-01: **Anna Bellomo**
Sums, Numbers and Infinity: Collections in Bolzano's Mathematics and Philosophy

ILLC DS-2022-02: **Jan Czajkowski**
Post-Quantum Security of Hash Functions

ILLC DS-2022-03: **Sonia Ramotowska**
Quantifying quantifier representations: Experimental studies, computational modeling, and individual differences

ILLC DS-2022-04: **Ruben Brokkelkamp**
How Close Does It Get?: From Near-Optimal Network Algorithms to Suboptimal Equilibrium Outcomes

ILLC DS-2022-05: **Lwenn Bussière-Carae**
No means No! Speech Acts in Conflict

ILLC DS-2023-01: **Subhasree Patro**
Quantum Fine-Grained Complexity

ILLC DS-2023-02: **Arjan Cornelissen**
Quantum multivariate estimation and span program algorithms

ILLC DS-2023-03: **Robert Paßmann**
Logical Structure of Constructive Set Theories

ILLC DS-2023-04: **Samira Abnar**
Inductive Biases for Learning Natural Language

ILLC DS-2023-05: **Dean McHugh**
Causation and Modality: Models and Meanings