Title	マルチエージェント・コミュニケーションのための様相論理に関する線形代数的意味論
Author(s)	秦野，亮
Citation	
Issue Date	2017-03
Type	Thesis or Dissertation
Text version	ETD
URL	http://hdl.handle.net/10119/14247
Rights	
Description	Supervisor東条　敏，情報科学研究科，博士

Linear Algebraic Semantics for Modal Logic of Multi-agent Communication

Ryo Hatano

Japan Advanced Institute of Science and Technology

Doctoral Dissertation

Linear Algebraic Semantics for Modal Logic of Multi-agent Communication

Ryo Hatano

Supervisor: Satoshi Tojo

School of Information Science
Japan Advanced Institute of Science and Technology

March, 2017

Abstract

In this thesis, we provide new computational bases for modal logic of multi-agent communication. One of the most important aspects of multi-agent communication is changes of agent's knowledge or belief [11]. Nowadays, such changes are well-discussed in terms of modal logic, as dynamic epistemic logic (**DEL**) [40]. Since **DEL** provides strong bases to handle agent's knowledge (or belief), development of such bases is an important subject for further studies. In order to obtain such bases, we focus on the following issues in this thesis.

The first issue is about the ordinary model-theoretic approach to Kripke semantics of modal logic. For some situation, we should be careful for, e.g., calculation of the truth of $\Box p$ at a 'dead-end' world where we cannot access any world. Since some conditions or necessary operations might be implicit in the proof of semantic properties, we sometimes might overlook them under the model-theoretic approach. To cover such a point, we propose to use a linear algebraic reformulation of Kripke semantics based on Fitting's approach [8]. His approach allows us to capture behaviors of the standard semantics in terms of Boolean matrix calculation explicitly. We show a matrix representation of Kripke semantics and its relevant properties, and also connect our argument to capture restricted form of quantifications in first-order logic.

The second issue is a deficiency of studies of proof theory for dynamic logic of relation changers (**DLRC**). **DLRC** is a recent approach to **DEL** and provides a general framework to capture many dynamic operators of **DEL** in terms of relation changing operation written by programs in propositional dynamic logic (**PDL**). However, the proof theory for **DLRC** is not well-studied except the sound and complete Hilbert-style axiomatization [39, 25]. Therefore, we propose the cut-free labelled sequent calculus for **DLRC**. We show that our sequent calculus is equipollent with the above Hilbert-style axiomatization.

The third issue is about the integration of the notion of structures among agents into **DEL**. When we study multi-agent communication system, we can naturally assume the existence of communication channels between agents such as phone numbers and email address. However, when we try to integrate such a notion of structures into **DEL**, we cannot avoid facing some problems, e.g., the decidability of resultant logic(s) and management of many indices, such as agent IDs and names of the possible worlds in our syntax and its semantics. Therefore, we propose to implement the above notion as a constant symbol, then we can define decidable and semantically complete doxastic logic with communication channels and dynamic operators. Moreover, we also propose a linear algebraic reformulation of these. Based on the framework of **DLRC**, we reformulate our proposed semantics in terms of Boolean matrices. In order to reformulate our semantics, we also provide matrix representation of programs in **PDL**.

Key Words: **Modal logic; Dynamic logic of relation changers; Linear algebraic reformulation of Kripke semantics; Labelled sequent calculus; Channel based agent communication**

Acknowledgments

First and foremost I would like to express my profound gratitude to my supervisor Satoshi Tojo for his patience, continuous support and guidance since the days I began working on my study in his laboratory. He gave me many opportunities and financial supports to improve my studies, e.g., travel supports of international conferences, teaching assistants of several lectures, tutors for international students. In addition, he encouraged me to continue my works even when I did serious failures. The days of studies in his laboratory are precious treasures of my life.

I would like to express my sincere gratitude to my advisor Katsuhiko Sano for all the things that he taught me. When he was an assistant professor at Tojo laboratory, his desk was located near mine. He always gave me very energetic and careful guidance when I ran into a trouble spot or had a question about my study. After he promoted to the position of associate professor at Hokkaido University, he continuously steered me in the right direction, and I deeply appreciate his kindness.

Beside my supervisor and advisor, I would like to thank Hajime Ishihara, Kazuhiro Ogata, and Yuko Murakami for being referees of my thesis. Thanks to their comments and encouragements, the thesis was improved from its preliminary version. In particular, Hajime Ishihara recommended me to study the soundness theorem with respect to a linear algebraic reformulation of Kripke semantics and Yuko Murakami pointed out possible further directions of this thesis, e.g., an extension of our approach to handle infinite norm of Boolean matrices. Kazuhiro Ogata gave me advice for educational aspects of my thesis. He was also an advisor for my minor research project that studies basics of distributed algorithms. I am grateful my second supervisor Hiroyuki Iida for his kind support. His comment about research questions was very beneficial to me and much appreciated. In connection with my publications, I would like to thank anonymous reviewers of ICAART2015, TTL2015, and AWPL-TPLC2016 for their helpful, thoughtful and detailed comments. In addition, it was my pleasure to receive comments on my presentation in TTL2015 conference from Johan van Benthem. He gave me some intriguing technical comments what I could not expect and send me a copy of a reference about a linear algebraic approach in the context of spacial logics.

It was a lot of pleasure to talk with visitors to Tojo laboratory. I would like to sincerely thank Robert Anthony Kowalski for teaching how to write an abstract of literature. I also wish to thank Minghui Ma and Jonni Virtema. Thanks to them, I could improve my scientific communication skills in English.

There were several reading groups at JAIST supervised by Katsuhiko Sano. I have studied many topics of, e.g., first-order logic, dynamic epistemic logic, dynamic logic and naive set theory in detail. The groups provided nice exercises for me to acquire mathematical foundations of this thesis. Therefore, a special acknowledgment goes to Katsuhiko Sano for these years, again. I also thank the following members for their efforts in the groups: Norihiro Arai, Pimolluck Jirakunkanok, Shoshin Nomura, Teeradaj Racharak, Tomoaki Hashizaki and Kei Shirakizawa.

The days of my studies were also supported by dozens of friends at JAIST. We have held a small seminar of linear algebraic semantics for this thesis to obtain feedbacks from students (Appendix A.6). I would like to thank the following participants of this experiment: Jun Takaba, Yuri Kido, Mingrui Yin, Masayuki Fujiwara, Tama Duangnamol, Hiroki Sudo, Takaya Watanabe, Yuuhei Shimizu, Nguyen Dinh Nghi, Hoshio Seki, Hoang Tran Xuan and Danilo S Carvalho. Thank you very much to my colleagues who have helped and encouraged me in many ways: Naoto Noguchi, Matsuo Konagaya, Nana Nishimura, Shota Hisai, Elise Myriam Olivia Perrotin, Mahdi Rohani Hezhad, Wenzhen Wang, Kong Yi, Shingo Hagiwara, Yoshiki Nakayama, Sakiko Yamasaki and the rest of all colleagues and friends who gave me valuable suggestions and kind encouragements.

For organizations, I would like to express deep gratitude to Japan Advanced Institute of Science and Technology. The institute provided me a really nice environment to study and many financial supports for seven years. Thank you to all the staffs in JAIST. In particular, I would like to thank Takio Hayashi and Megumi Sasaki (Health Care Center), Mun'delanji Vestergaard (Center for Graduate Education Initiative), William Holden and John Blake (Technical English Communication Education Program), Miyuki Sakurai and Nao Aoki (School of Information Science). I also would like to express my gratitude to Fumio Mizoguchi and Hiroyuki Nishiyama in Tokyo University of Science. They led me a road to researcher and gave me many chances to develop programming skills when I was an undergraduate student at TUS. Remarkably, I could integrate my old source codes that were originally developed at Nishiyama laboratory into a part of the implementation that is described in this thesis. I would like to acknowledge Misa Tamayama for continuous encouragement since I was a student at TUS. I am also grateful to Tatsuya Masamura. He introduced JAIST to me, and that was a beginning of my days of studies in the institute.

Finally but not the least, I would like to thank my family: Sho Hatano, Emi Hatano, Risa Hatano, Yusuke Shimbo, Atsuko Shimbo, Koji Tsutsui, Yumi Tsutsui, Hisae Hatano, Kenichi Hatano, Machiko Hatano, Yuko Hatano, Michiko Iwase, and Etuko Sasaki. Without their kind encouragements and continuous financial supports, I could not finish my thesis.

Ishikawa, March 2017

Ryo Hatano

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Background and Motivation

The aim of this thesis is to provide new computational tools of modal logic and its extensions, i.e., dynamic epistemic logics, in the context of multi-agent communication. One of the most important aspects of multi-agent communication is changes of agent's knowledge or belief [11]. Nowadays, such changes are well-discussed in terms of modal logic, as dynamic epistemic logic (**DEL**) [40].

Modal logic has its origin in the study of necessity and possibility by Aristotle, and the modern systematic approach to this logic was investigated by Lewis and Langford [23]. Then, the modern semantical approach to modal logic was established by Kripke [21].[1] Since his semantics can cover various modal logics, nowadays it is called Kripke semantics. A key idea of his semantics can be summarized as follows. Let us consider a *relational structure* consists of a set W of *possible worlds* and an *accessibility relation* $R \subseteq W \times W$. If we have such a structure, we can define the truth condition of a formula A with the modal operator $\Box$ which reflects the notion of necessity as follows:

$$\Box A \text{ is true at a world } w \in W \text{ iff } A \text{ is true at all possible worlds } R\text{-accessible from } w.$$

Similarly, we can also define the truth condition of the dual modal operator $\Diamond$ of $\Box$ which reflects the notion of possibility as follows:

$$\Diamond A \text{ is true at a world } w \in W \text{ iff } A \text{ is true at some possible worlds } R\text{-accessible from } w.$$

Studies of **DEL** can be regarded as a recent approach to multi-agent communication on the modal logic bases. **DEL** is known as a large family of logics that extend modal logic (in particular, epistemic logic) with dynamic operators. Such operators allow us to capture changes of agent's knowledge or belief over Kripke semantics (or its variant). For example, Plaza proposed public announcement logic (**PAL**) [32] which is a variant of **DEL**, and it has a *public announcement* operator. Using this operator, we can capture how an agent's knowledge change after a piece of information is *publicly* announced to all the agents in terms of changes (or updates) of a relational structure in Kripke semantics.

Since the above approaches in **DEL** provide strong bases to handle agent's knowledge (or belief), development of these bases is an important subject for further studies, e.g.,

[1] We note that similar approach was also proposed by Tarski and Jónsson [18, 17].

theoretical studies of mathematical logic and practical applications of artificial intelligence. Furthermore, some of the above bases will be important teaching topics for an education of future students. It is similar to the ordinary teaching topics of mathematical logic such as the truth-table calculation of propositional logic, etc.

This thesis is a challenge to find and provide candidates of such bases in terms of modal logic of multi-agent communication. To provide such bases, this thesis focuses on the following three questions.

Firstly, let us consider the ordinary model-theoretic approach to Kripke semantics. When we study Kripke semantics of modal logic, we often use the model-theoretic approach. Based on the model-theoretic definition of Kripke semantics, we can prove various semantic properties over Kripke semantics, e.g., the validity of a given formula. However, there are some cases that we have to prove a given property(s) very carefully, e.g., the truth of $\Box p$ at a 'dead-end' world where we cannot access any world.

(Q1) What is the effective approach(es) to overcome such situation?

Secondly, let us see a recent study of proof theory for **DEL**. As a recent approach to **DEL**, van Benthem and Liu proposed dynamic logic of relation changers (**DLRC**) [39, 25]. Their study provides a general framework to capture many dynamic operators of **DEL** in terms of relation changing operation written by programs in propositional dynamic logic (**PDL**). In addition, they also provided the sound and complete Hilbert-style axiomatization for **DLRC**. However, except this axiomatization, proof theory for **DLRC** is not well-studied.

(Q2) What kind of proof theory except Hilbert-style axiomatization can be provided to **DLRC**?

Thirdly, let us consider logical studies of **DEL** for multi-agent communication system. When we study (practical) multi-agent communication system, we can naturally assume the existence of communication channels between agents such as phone numbers or e-mail address [4, 13, 31, 20]. However, when we integrate such a notion among agents into **DEL**, we need to face some problems, e.g., the decidability of resultant logic(s) and management of many indices, such as agent IDs and names of the possible worlds in our syntax and its semantics.

(Q3) In order to overcome these issues on handling many indices, what kind of approach(es) can we take?

In the remaining of this chapter, let us see these matters in detail and why it is important for us to solve these questions.

1.1.1 Semantical Study for Modal Logic

Traditionally, we usually use the model-theoretic approach to study Kripke semantics of modal logic, and it is obviously important bases for many of us. So far, many semantic properties over Kripke semantics are studied by this approach. Some properties of modal logic are rather easy to understand for many of us since we can explain Kripke semantics

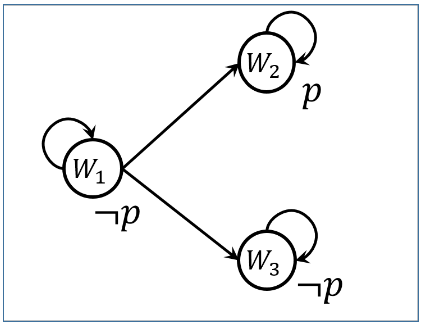


Figure 1.1: Example of Kripke Model

on a graphical representation of a Kripke model (see Figure 1.1). For example, we can calculate the truth value of a formula over a graphical representation of a Kripke model visually. Under this approach, however, there are some cases that we have to prove some properties very carefully. For example, they may include the truth of $\Box p$ at a 'dead-end' world where we cannot access any world, and the verification of the Euclideanness property (wRv and wRu jointly imply vRu, for all w, v, u). To show the truth of $\Box p$ at the dead-end world, we need to consider when the implication is vacuously true. We sometimes might overlook such a case during our proof. In addition, the verification of Euclideanness of a given frame might be the similar case. We need to check whether the frame satisfies the condition of Euclideanness very carefully because v and u in the above condition are possibly the same. If the cardinality of the domain of the model is larger, such checking might be more involved. What we can observe is that some conditions or necessary operations might be implicit in the proof of semantic properties, and we sometimes might overlook them under the model-theoretic approach. We might also face the similar situation to show a more general properties of modal logic such as the soundness for Kripke semantics, etc.

1.1.2 Proof Theoretical Study for Dynamic Epistemic Logic

As we mentioned, **DEL** models changes of agent's knowledge and belief over Kripke semantics, and public announcement logic (**PAL**) is one of such well-studied variants.

PAL has a public announcement operator $[!A]$ whose intuitive meaning is 'after the announcement that A, delete every non-A world from the domain of a Kripke model.' As a result, all links to the non-A world is also deleted from the model. We may also consider a variant of the public announcement operator, namely, the link-cutting public announcement operator, which keeps the original domain but changes the given accessibility (relation) alone. The link-cutting idea for public announcements was proposed and employed in the context of update semantics by Gerbrandy and Groeneveld [12], doxastic logic by van Ditmarsch et al. [40], preference logic and dynamic logic of relation changers (**DLRC**) by van Benthem and Liu [39, 25]. As for the study by van Benthem and Liu, the link-cutting idea is used to upgrade agent's preferences. Their preference logic has the preference modality $[a]$ of agent a, and it is used to describe what is better[2] to the agent.

[2] In their preference logic, the notion of *better*-ness for preference is supported by a binary preference relation (i.e., reflexive and transitive relation) over worlds. Common notions of preference play between propositions, but their approach emphasizes comparisons of worlds rather than propositions.

In particular, a formula $[a]A$ reads 'A is true at every world that is the agent a considers as least as good as the current world.' In this logic, dynamic operators can be regarded as preference upgrade operators. For example, the link-cutting operator for public announcements is defined as a preference upgrade operator and called the *link-cutting public update operator*. Intuitively, the update operator means 'after the announcement that A, remove all the links between A-worlds and non-A worlds from a given model.' Moreover, the update operator becomes an instance of *relation changers* in **DLRC**. Relation changers can be specified by programs in propositional dynamic logic (**PDL**) [14] without the iteration operator $*$. We may also write many other types of dynamic operators by programs, e.g., the link-adding operator. To acquire such a technically general result, we note that a set of agents in preference logic is regarded as a set of atomic programs.

There are several studies on proof theory of **DEL**s. The first Hilbert-style system of **PAL** is proposed by Plaza [32]. There is another Hilbert-style system of **PAL** which is provided by Wang and Cao [41]. Labelled tableau calculus for **PAL** is provided by Balbiani et al. [3] and its generalized version for non-normal **PAL** is given by Ma et al. [26]. The labelled formalism is also employed in sequent calculi for **PAL** such as [28, 2]. In addition, Frittella et al. [9] also proposed a sequent calculus for **DEL** based on display calculus which enjoys Belnap-style cut elimination. As for **PDL**, several cut-free sequent calculi are also proposed in [6, 16, 27]. Finally, Hilbert-style axiomatization for **DLRC** is provided by van Benthem and Liu [39, 25].

DLRC provides a general framework to capture many dynamic operators of **DEL**. It seems that **DLRC** can cover a broader range of **DEL**. However, in contrast to its potential, the proof theory for **DLRC** is not well-studied except the above axiomatization.

1.1.3 Logical Study for Multi-agent Communication

When we study multi-agent communication system, it forces us to manage an existence of communication channels between agents, such as phone numbers or e-mail addresses, although ordinary modal logic for the multi-agent system does not consider the notion of a channel. However, we can naturally assume the existence of such channels between agents, e.g., in [4, 13, 31, 20]. Communicability in those agents can be represented in a directed graph, where a vertex is an agent and an edge a channel.

There are several studies integrating the notion of structure among agents into **DEL**. Seligman et al. [35] proposed a two-dimensional modal logic which can handle both agents' knowledge and a friendship relation between agents. Based on the two-dimensional framework, Sano and Tojo [33] implemented the idea of communication channel in terms of a modal operator and studied belief changes of agents, where they raised the following requirements:

(R1) An effect of an informing action is restricted to some specified agents determined by communication channels.

(R2) An existence of communication channel between agents depends on a given situation, i.e., it is not constant or rigid for all situations.

One of the deficiencies of the two dimensional framework is that it is still unknown whether the resulting logics in [35, 33] are decidable, i.e., whether we can effectively test if a given formula is a theorem of a given logic. Another deficiency is a better formalism or notion for handling many indices such as agent IDs, names of possible worlds in our syntax and its semantics. When we study modal logic (or **DEL**) for multi-agent system, we need to manage such indices even if we do not consider the notion of channels.

1.2 Our Proposals

In the previous section, we have seen the background of our questions. In this section, we present our proposals to solve such questions. Our solutions provide new bases of modal logic of multi-agent communication.

1.2.1 Linear Algebraic Semantics for Modal Logic and Dynamic Logic of Relation Changers (Chapters 3 and 4)

As for a solution of our question (Q1) described in Section 1.1.1, we propose to use Fitting's linear algebraic approach to Kripke semantics [8]. A key idea of his approach is to represent an accessibility relation R by a Boolean square matrix and a valuation $V(p)$ of an atomic variable p by a Boolean column vector, provided the cardinality of the domain is finite.[3] As a result, we may compute the truth set of a formula by calculations over Boolean matrices. For example, the truth set of $\Diamond p$ is calculated by the multiplication of the square matrix of R and the vector of $V(p)$. Moreover, we may also verify the frame property of a given frame by the calculation of matrices. Since these calculations are based on the truth-table calculation of propositional logic, we can regard the calculations as an extended version of truth-table calculation. As a result, we can handle conditions such as a given model and operations during our proof explicitly. In addition, we may replace operations over the quantifications and the binary relation of first-order logic by the truth-table calculation of propositional logic and elementary calculations over Boolean matrices of linear algebra. Moreover, this extension allows us to calculate some restricted forms of quantifications (in Kripke semantics) *without* bound variables of first-order logic.

In connection with our question (Q2), we also propose to extend this approach to capture **DLRC** [39, 25]. Based on the above result, we extend the linear algebraic reformulation of Kripke semantics of modal logic to handle relation changers of **DLRC** that allows us to capture many dynamic operators of **DEL** in terms of relation changing operation(s). In addition, we can show the soundness theorem of the known Hilbert-style axiomatization of **DLRC** [39, 25] in terms of Boolean matrices. As a result, we can write and capture the proof of the soundness by simple equations over Boolean matrices.

Here, in order to clarify the position of our linear algebraic approach to Kripke semantics in this thesis, we summarize the other previous works as follows. Liau [24] introduced Boolean matrix operations for multiple agents' belief reasoning, revision, and fusion. Based on the matrix representation of belief states, he proposed a belief logic and its algebraic semantics. Similarly, Fusaoka et al. [10] introduced real-valued matrix operation for qualitative belief change in a multi-agent system. Based on the above studies, Tojo [37] proposed notions of Boolean matrices and vectors for the simultaneous informing

[3] We note that this assumption is often justified because most of the well-known modal logics, for example, **KT**, **S4** and **S5**, enjoy the finite model property, i.e., A is the theorem of a modal logic Λ iff A is valid for all *finite* models for the logic Λ.

action with communication channels. He showed that the notions of matrices could represent a public announcement [32] and a consecutive message passing. It can be regarded as an application of the linear algebraic approach to multi-agent communication. Then, Hatano et al. extended Tojo's idea to provide rigorous definitions in [15]. They proposed a decidable and semantically complete multi-agent doxastic logic with communication channels and its dynamic extensions with two informing action operators. With the help of van Benthem and Liu's idea of *relation changer* [39, 25], their dynamic operators can be regarded as program terms in propositional dynamic logic. Afterward, they provided a linear algebraic reformulation of the proposed semantics. In addition, a supporting software based on the above idea is also provided. Our proposal here expands Hatano et al.'s work [15] into **DLRC**. In connection with spatial logics and linear algebra, we refer to a survey by van Benthem and Bezhanishvili [38]. In the survey, they mentioned connections between modal logic and linear algebra over vector spaces $\mathbb{R}^n$. Different from our approach, they did not provide Kripke semantics in terms of Boolean matrices.

Finally, we comment a relevant study by Berghammer and Schmidt [5] to ours in terms of relational algebra. They proposed a relational algebraic approach to investigate finite models of non-classical logics such as multi-modal logics with common knowledge operators and computational tree logic. They interpret the logics in relation algebra with transitive closures whose representation is based on Boolean matrices. They also present applications of their tool `RELVIEW` based on the above idea for finite model checking tasks, e.g., the muddy children puzzles.

Their study seems very close to ours, although their approach is different from ours since they simply used Boolean matrices as an input and output interface to their computation system and did not compute the matrices directly.[4] For example, they defined a 'composition' of two Boolean square matrices which corresponds to two accessibility relations by a componentwise relational composition, although we define it by a multiplication of Boolean matrices. Another difference is that, in this thesis, we have more uniform list of the matrix representation of frame properties than that of Berghammer and Schmidt [5] and two types of correspondence between modal axioms and their matrix representations.

1.2.2 Cut-free Labelled Sequent Calculus for Dynamic Logic of Relation Changers (Chapter 4)

As for a solution of our question (Q2) explained in Section 1.1.2, we propose the cut-free labelled sequent calculus for **DLRC**. Similarly to [28, 2], our sequent calculus employs labelled formalism based on Kripke semantics, but it is extended to cover the notion of programs of **PDL**. This is because we can capture behaviors of semantic objects such as the truth of a formula at a certain world and links of an accessibility relation by corresponding syntactic objects explicitly. Labelled expressions are the most basic component of labelled sequent calculus. For example, for **PDL** [27], labelled expressions consist of:

$$x : A \qquad x\mathsf{R}_\alpha y \qquad x = y$$

[4] Relations can be represented by Boolean matrices [34], although ordinary works of relational formalization do not employ such representation, e.g., a study of relational formalization of non-classical logics by Orlowska [30].

where x and y are variables, and α is a program of **PDL**. We emphasize that these components are syntactic objects. An expression $x : A$ means that 'A formula A holds at state x,' $x\mathsf{R}y$ means that 'there is a link from x to y of program α,' and $x = y$ means that 'state x equals state y.' Based on these components, we can define labelled sequent calculus for **PDL**. The above feature allows us to analyze behaviors of syntactic proof corresponding to some semantic proof of the validity of a formula explicitly. We extend the above approach to handle relation changers of **DLRC**. In order to assure adequacy of our calculus, we have to show the soundness and completeness of our proof system. As a result, we will show that our sequent calculus is equipollent with the known sound and complete Hilbert-style system of **DLRC** [39, 25].

1.2.3 Linear Algebraic Semantics for Multi-agent Communication (Chapter 5)

As for a solution of our question (Q3) described in Section 1.1.3, we propose to implement the notion of communication channel as a constant symbol c_{ab} whose reading is 'there is a channel from agent a to agent b,' instead of a channel as a modal operator. Based on this approach, we can define decidable and semantically complete doxastic logic with communication channels and dynamic operators. As for dynamic operators, we implement two instances of relation changers [39, 25] whose effect is restricted to some specified agents determined by communication channels, i.e., these operators satisfy requirement (R1) described in Section 1.1.3.

Moreover, in order to handle many indices appears in our proposed logic effectively, we propose a linear algebraic representation of these. That is, with the help of the framework of **DLRC** [39, 25], we reformulate our semantics of the above doxastic logic and its dynamic extensions in terms of boolean matrices. In order to reformulate our semantics, we also provide a matrix representation of a program in **PDL**. Finally, we present matrix calculation algorithms for these operators.

We note that this proposal can be regarded as an application of the idea of the linear algebraic approach to **DLRC** (cf. Section 1.2.1) and integration of the notion of communication channels into **DEL** (this section).

1.3 Thesis Outline

The rest of this thesis is organized as follows. Chapter 2 introduces background needed in the rest of the thesis. Note that this chapter does not contain our original works. Section 2.1 recalls (multi-) modal logic which is a fundamental basis of the other logics presented in this thesis. We introduce syntax, Kripke semantics, frame definability, Hilbert-style axiomatization and the completeness theorem of modal logic. Section 2.2 also recalls definitions and notions of **PDL** and its iteration-free fragment. We use this fragment as a basis of **DLRC** in the later of this thesis. Section 2.3 reviews **PAL** as a reference of dynamic approach to modal logic. Finally, Section 2.4 introduces basic notions and properties of Boolean matrices for our linear algebraic semantics.

Chapter 3 concerns *linear algebraic semantics* for modal logic. First, we review the ordinary approach to study modal logic in Section 3.1. Second, we describe a matrix representation of Kripke semantics and its relevant properties, and also connect our argument to the concept of quantification in first-order logic in Section 3.2.

Chapter 4 studies both linear algebraic semantics and *cut-free labelled sequent calculus* for dynamic logic of relation changers. We provide our syntax and semantics for **DLRC** in Section 4.1. In Section 4.2, we present our linear algebraic reformulation of Kripke semantics for **DLRC**. We establish the soundness theorem in terms of Boolean matrices (Section 4.2.2). Then, we describe our sequent calculus **GDLRC** from a labelled formalism for that in Section 4.3 . In this section, we establish the following two theorems; all theorems of **HDLRC** are derivable in **GDLRC** (Theorem 4.11) and our sequent calculus enjoys the cut elimination theorem (Theorem 4.19). In Section 4.3.4, we also establish that our sequent calculus is sound for Kripke semantics (Theorem 4.23). Finally, we conclude that our sequent calculus is equipollent with the Hilbert-style axiomatization **HDLRC** (Corollary 4.24).

Chapter 5 investigates an extended instance of **DLRC** that deals with multi-agent communication. First, we introduce a static logic of agents' belief equipped with the notion of channel between agents in Section 5.1. In this section, we establish the following theorems; all the valid formulas on all the finite Kripke models for our syntax is completely axiomatizable (Theorem 5.11) and our proposed axiomatization is decidable (Theorem 5.12). In order to deal with changes of agents' belief via a communication channel, Section 5.2 provides two dynamic operators to our syntax of static logic with sets of reduction axioms. Moreover, for a better formalism for handling these semantics efficiently, we consider a linear algebraic representation of these. Following the idea explained in Chapter 4, we reformulate our proposed semantics of the doxastic static logic and its dynamic extensions in terms of boolean matrices in Section 5.3. With the help of [39, 25], we can regard our two dynamic operators as programs in **PDL** and also reformulates the semantics of two operators in terms of Boolean matrices. Finally, we use our boolean matrix reformulation to define algorithms for checking agent's belief at a given world and for rewriting a given Kripke model by one of our dynamic operators in Section 5.4.

Chapter 2

Preliminaries

This chapter introduces technical background that is needed in this thesis. In Section 2.1, we recall (multi-) modal logic as a starting point of our studies. We introduce syntax, Kripke semantics, frame definability, Hilbert-style axiomatization and the completeness theorem of modal logic. Then, in Section 2.2, we also recall propositional dynamic logic and its iteration free fragment. Besides syntax and Kripke semantics, we briefly review the sound and complete Hilbert-style axiomatizations of them. In Section 2.3, we review public announcement logic. Since **PAL** is based on epistemic logic, we define Kripke semantics over epistemic models. We also introduce the sound and complete Hilbert-style axiomatization for **PAL**. Finally, in Section 2.4, we introduce basic notions and properties of Boolean matrices for our linear algebraic semantics.

2.1 Modal logic

2.1.1 Syntax and Kripke Semantics

First of all, we recall the ordinary multi-modal logic. A *modal language* $\mathcal{L}_{\mathbf{ML}}$ consists of the following vocabulary: a countably infinite set PROP of (atomic) propositional variables, Boolean connectives $\neg, \to$, and a finite set MOD of modal operators. If the cardinality of MOD is 1, we say that the language $\mathcal{L}_{\mathbf{ML}}$ is *mono-modal*; otherwise *multi-modal*. A set $\mathsf{Form}_{\mathbf{ML}}$ of formulas of the language $\mathcal{L}_{\mathbf{ML}}$ are defined as follows:

$$\mathsf{Form}_{\mathbf{ML}} \ni A ::= p \mid \neg A \mid (A \to A) \mid \Box A$$

where $p \in \mathsf{PROP}$ and $\Box \in \mathsf{MOD}$. We will omit the parentheses whenever convenient. In addition, we introduce abbreviations for the conjunction $\wedge$, the disjunction $\vee$, the logical equivalence $\leftrightarrow$, the truth $\top$, the falsity $\bot$, and the dual operator $\Diamond$ of $\Box$ as follows:

$$\begin{array}{lcllcl} A \vee B & := & (\neg A) \to B, & A \wedge B & := & \neg(A \to \neg B), \\ \top & := & A \to A, & \bot & := & \neg\top, \\ \Diamond A & := & \neg\Box\neg A, & A \leftrightarrow B & := & (A \to B) \wedge (B \to A). \end{array}$$

A formula $\Box A$ stands for 'it is necessary that A holds,' and $\Diamond A$ stands for 'it is possible that A holds.' We use Γ, Σ to denote a set of formulas. Then, the conjunction of all formulas in a finite set Γ is denoted by $\bigwedge \Gamma$ where $\bigwedge \Gamma := \top$ if $\Gamma = \emptyset$.

A (Kripke) *frame* $\mathfrak{F}$ is a tuple $(W, (R_\Box)_{\Box \in \mathsf{MOD}})$ where W is a non-empty set, called the *domain of* $\mathfrak{F}$, and each $R_\Box \subseteq W \times W$ is a binary relation on W, called an *accessibility* *relation*. The elements of the domain of a frame are called *possible worlds*, *states*, etc. Let $\mathfrak{F} = (W, (R_\Box)_{\Box \in \mathsf{MOD}})$ be a frame. We use the notation $|\mathfrak{F}|$ to mean the domain W of $\mathfrak{F}$. Then, we also use the infix notation $wR_\Box v$ to mean $wR_\Box v$. If the domain W of $\mathfrak{F}$ is a finite set, we say that $\mathfrak{F}$ is a *finite frame*. And if we talk about mono-modal language, we simply denote a frame by (W, R) and the following definitions for our semantics are defined over such R. A (Kripke) *model* $\mathfrak{M}$ is a tuple $(\mathfrak{F}, V)$ where $\mathfrak{F}$ is a frame and $V : \mathsf{PROP} \to \mathcal{P}(|\mathfrak{F}|)$ is a *valuation function* which assigns a subset of the domain $|\mathfrak{F}|$ of $\mathfrak{F}$ to an atomic variable. Given any model $\mathfrak{M} = (\mathfrak{F}, V)$, the domain $|\mathfrak{F}|$ of $\mathfrak{M}$ is also denoted by $|\mathfrak{M}|$.

Given any model $\mathfrak{M} = (W, (R_\Box)_{\Box \in \mathsf{MOD}}, V)$, any possible world $w \in W$, and any formula A, the *satisfaction relation* $\mathfrak{M}, w \models A$ (read: *A is true at w in* $\mathfrak{M}$) is defined as follows:

$$\begin{array}{lll}
\mathfrak{M}, w \models p & \text{iff} & w \in V(p), \\
\mathfrak{M}, w \models \neg A & \text{iff} & \mathfrak{M}, w \not\models A, \\
\mathfrak{M}, w \models A \to B & \text{iff} & \mathfrak{M}, w \models A \text{ implies } \mathfrak{M}, w \models B, \\
\mathfrak{M}, w \models \Box A & \text{iff} & \text{for all } v \in W : wR_\Box v \text{ implies } \mathfrak{M}, v \models A.
\end{array}$$

By definition, the satisfaction relations for $\wedge$, $\vee$, $\leftrightarrow$, $\top$, $\bot$, and $\Diamond$ are derived as follows:

$$\begin{array}{lll}
\mathfrak{M}, w \models A \wedge B & \text{iff} & \mathfrak{M}, w \models A \text{ and } \mathfrak{M}, w \models B, \\
\mathfrak{M}, w \models A \vee B & \text{iff} & \mathfrak{M}, w \models A \text{ or } \mathfrak{M}, w \models B, \\
\mathfrak{M}, w \models A \leftrightarrow B & \text{iff} & \mathfrak{M}, w \models A \text{ is equivalent to } \mathfrak{M}, w \models B, \\
\mathfrak{M}, w \models \top, & & \\
\mathfrak{M}, w \not\models \bot, & & \\
\mathfrak{M}, w \models \Diamond A & \text{iff} & \text{for some } v \in W : wR_\Box v \text{ and } \mathfrak{M}, v \models A.
\end{array}$$

The *truth set* $[\![A]\!]_\mathfrak{M}$ is defined by $[\![A]\!]_\mathfrak{M} = \{\, w \in W \mid \mathfrak{M}, w \models A \,\}$. By definition, we can obtain the following:

$$\begin{array}{lll}
[\![p]\!]_\mathfrak{M} & := & V(p), \\
[\![\neg A]\!]_\mathfrak{M} & := & W \setminus [\![A]\!]_\mathfrak{M}, \\
[\![A \to B]\!]_\mathfrak{M} & := & W \setminus [\![A]\!]_\mathfrak{M} \cup [\![B]\!]_\mathfrak{M}, \\
[\![\Box A]\!]_\mathfrak{M} & := & \{\, w \in W \mid wRv \text{ implies } v \in [\![A]\!]_\mathfrak{M} \text{ for all } v \in W \,\}.
\end{array}$$

The notion of *validity* is defined over the various levels of semantical structure as follows:

- A is *valid in a model* $\mathfrak{M}$ (notation: $\mathfrak{M} \models A$) if $\mathfrak{M}, w \models A$ for all worlds $w \in W$.
- A is *valid in a frame* $\mathfrak{F}$ (notation: $\mathfrak{F} \models A$) if $(\mathfrak{F}, V) \models A$ for all valuations V for $\mathfrak{F}$.
- Γ is *valid in a frame* $\mathfrak{F}$ (notation: $\mathfrak{F} \models \Gamma$) if $\mathfrak{F} \models A$ for all $A \in \Gamma$.
- A is *valid in a class* $\mathbb{M}$ *of models* (notation: $\mathbb{M} \models A$) if $\mathfrak{M} \models A$ for all $\mathfrak{M} \in \mathbb{M}$.
- A is *valid in a class* $\mathbb{F}$ *of frames* (notation: $\mathbb{F} \models A$) if $\mathfrak{F} \models A$ for all $\mathfrak{F} \in \mathbb{F}$.

In connection with the validity of a formula and the truth set of that, the following proposition holds:

Proposition 2.1. *Given any model* $\mathfrak{M}$ *and any formula* A*,*

$$\mathfrak{M} \models A \leftrightarrow B \;\text{ iff }\; [\![A]\!]_\mathfrak{M} = [\![B]\!]_\mathfrak{M}.$$

2.1.2 Frame Definability

Given a set Γ of formulas of $\mathsf{Form}_{\mathbf{ML}}$, the frame class $\mathbb{F}$ is *defined* by Γ if for all frames $\mathfrak{F}$,

$$\mathfrak{F} \models \Gamma \text{ iff } \mathfrak{F} \in \mathbb{F}.$$

For simplicity, if the set Γ is the singleton $\{ A \}$, we say that the frame class $\mathbb{F}$ is defined by A (or A defines $\mathbb{F}$). We also say that the frame class $\mathbb{F}$ is (*modally*) *definable* if there exists some set Γ such that $\mathbb{F}$ is defined by Γ.

What kind of class(es) of frames are defined by which formulas? In modal logic, we can consider various conditions over an accessibility relation $R_\Box$ for a given frame $(W, (R_\Box)_{\Box \in \mathsf{MOD}})$. For example, Table 2.1 shows the well-known (frame) conditions on an accessibility relation and their names. These conditions are also known as *frame properties* since their names and conditions are also used in the level of frames. For example, given a frame $(W, (R_\Box)_{\Box \in \mathsf{MOD}})$, if each accessibility relation $R_\Box$ is reflexive, we also say that the frame $(W, (R_\Box)_{\Box \in \mathsf{MOD}})$ is reflexive. Then, we can consider about classes of frames which satisfy these frame properties. It is known that such classes are defined by well-known formulas as shown in the next proposition.

Proposition 2.2. *Each formulas listed in Table 2.1 defines a class of all frames which satisfy the corresponding frame property.*

Table 2.1: Correspondence between Frame Properties and Formulas

Name	Frame Condition	Formula	
Reflexive	$\forall w(wRw)$	$\mathrm{T}_\Box$	$\Box p \to p$
Symmetric	$\forall w, v(wRv$ implies $vRw)$	$\mathrm{B}_\Box$	$p \to \Box\Diamond p$
Transitive	$\forall w, v, u(wRv \& vRu$ imply $wRu)$	$4_\Box$	$\Box p \to \Box\Box p$
Serial	$\forall w \exists v(wRv)$	$\mathrm{D}_\Box$	$\Box p \to \Diamond p$
Euclidean	$\forall w, v, u(wRv \& wRu$ imply $vRu)$	$5_\Box$	$\Diamond p \to \Box\Diamond p$

Definition 2.3 (Lemmon-Scott Formulas [22, 7])**.** *Let $k, l, m, n \in \omega$. Given any formula A, $\Box^n A$ is inductively defined by $\Box^0 A := A$ and $\Box^{n+1} A := \Box\Box^n A$. Similarly, $\Diamond^n A$ is also defined by $\Diamond^0 A := A$ and $\Diamond^{n+1} A := \Diamond\Diamond^n A$. Then, the Lemmon-Scott formulas $G_{(k,l,m,n)}$ are defined as follows:*

$$\Diamond^k \Box^l p \to \Box^m \Diamond^n p.$$

From the Lemmon-Scott formulas, we can obtain every formulas listed in Table 2.1 as shown in Table 2.2.

Definition 2.4 (Lemmon-Scott Property)**.** *Let $k, l, m, n \in \omega$. Given any binary relation R, R^n is inductively defined by wR^0v iff $w = v$ and $R^{n+1} = R^n \circ R$ where $\circ$ is the relational composition. The Lemmon-Scott property $C_{(k,l,m,n)}$ is defined as follows:*

$$\forall u, v, w((wR^k v \& wR^m u) \textit{ implies } \exists x(vR^l x \& uR^n x)).$$

Proposition 2.5. *Let $k, l, m, n \in \omega$. The Lemmon-Scott formulas $G_{(k,l,m,n)}$ define a class of all frames which satisfy the Lemmon-Scott property $C_{(k,l,m,n)}$.*

Table 2.2: Instances of the Lemmon-Scott Formulas

Formulas	k	l	m	n
$\mathrm{T}_\Box$ $\Box p \to p$	0	1	0	0
$\mathrm{B}_\Box$ $p \to \Box\Diamond p$	0	0	1	1
$4_\Box$ $\Box p \to \Box\Box p$	0	1	2	0
$\mathrm{D}_\Box$ $\Box p \to \Diamond p$	0	1	0	1
$5_\Box$ $\Diamond p \to \Box\Diamond p$	1	0	1	1

Table 2.3: Hilbert-style Axiomatization HK

$(\mathbf{Taut})$	All instances of propositional tautologies
$(\mathbf{K}_\Box)$	$\Box(A \to B) \to (\Box A \to \Box B)$ $(\Box \in \mathsf{MOD})$
$(\mathbf{MP})$	From A and $A \to B$, infer B
$(\mathbf{Nec}_\Box)$	From A, infer $\Box A$ $(\Box \in \mathsf{MOD})$

2.1.3 Hilbert-style Axiomatization $\mathsf{HK}\Sigma$ and Completeness

In Table 2.3, we show the Hilbert-style system HK. The axioms and rules in Table 2.3 are given as *schemes*. Schemes are rules standing for infinitely many *instances*. For example, $\mathrm{K}_\Box$ $\Box(p \to q) \to (\Box p \to \Box q)$ (where $p, q \in \mathsf{PROP}$ and $\Box \in \mathsf{MOD}$) is a possible instance of the scheme $(\mathbf{K}_\Box)$ $\Box(A \to B) \to (\Box A \to \Box B)$, $(\Box \in \mathsf{MOD})$. In order to distinguish names of schemes from those of instance, we use the bold font with parenthesis for schemes, e.g., $(\mathbf{K}_\Box)$ is a name of scheme $\Box(A \to B) \to (\Box A \to \Box B)$ $(\Box \in \mathsf{MOD})$, and use the normal font for an instance on propositional variables, e.g., $\mathrm{K}_\Box$ is a name of instance $\Box(p \to q) \to (\Box p \to \Box q)$. In what follows, we consider a more general version of HK which consists of axioms and rules of HK with additional axiom schemes.

Definition 2.6. *Let* $\Box \in \mathsf{MOD}$. *Axiom schemes based on* $\mathrm{T}_\Box$, $\mathrm{B}_\Box$, $4_\Box$, $\mathrm{D}_\Box$ *and* $5_\Box$ *are defined as follows:*

$$
\begin{array}{ll}
(\mathbf{T}_\Box) & \Box A \to A \\
(\mathbf{B}_\Box) & A \to \Box\Diamond A \\
(\mathbf{4}_\Box) & \Box A \to \Box\Box A \\
(\mathbf{D}_\Box) & \Box A \to \Diamond A \\
(\mathbf{5}_\Box) & \Diamond A \to \Box\Diamond A
\end{array}
$$

Table 2.4: Various Systems of $\mathsf{HK}\Sigma$

Name	Axiom Schemes Σ
HK	$\emptyset$
HKT	$\{ (\mathbf{T}_\Box) \mid \Box \in \mathsf{MOD} \}$
$\mathsf{HS4}$	$\{ (\mathbf{T}_\Box), (\mathbf{4}_\Box) \mid \Box \in \mathsf{MOD} \}$
$\mathsf{HS5}$	$\{ (\mathbf{T}_\Box), (\mathbf{4}_\Box), (\mathbf{5}_\Box) \mid \Box \in \mathsf{MOD} \}$

Definition 2.7. *Let* $\Sigma \subseteq \{ \mathrm{T}_\Box, \mathrm{B}_\Box, 4_\Box, \mathrm{D}_\Box, 5_\Box \mid \Box \in \mathsf{MOD} \}$. *Then, let also* $\mathbf{\Sigma} \subseteq \{(\mathbf{T}_\Box), (\mathbf{B}_\Box), (\mathbf{4}_\Box), (\mathbf{D}_\Box), (\mathbf{5}_\Box) \mid \Box \in \mathsf{MOD}\}$ *be the finite set of axioms schemes corresponding to* Σ. *The Hilbert-style system* $\mathsf{HK}\mathbf{\Sigma}$ *is an axiomatic extension of* HK *that is the system* HK *extended with* $\mathbf{\Sigma}$.

Based on $\mathsf{HK}\mathbf{\Sigma}$, we can obtain a variety of systems that contain additional schemes in $\mathbf{\Sigma}$ defined the above. In Table 2.4, we show the well-known systems and their names as examples.

Definition 2.8. *A* derivation *in* $\mathsf{HK}\mathbf{\Sigma}$ *is a finite sequence of formulas which consist of an instance of an axiom or the result of applying an inference rule to formulas that occur earlier. If a formula* A *occurs in the last of a derivation in* $\mathsf{HK}\mathbf{\Sigma}$, *we say that* A *is* derivable (*or is a* theorem) *in* $\mathsf{HK}\mathbf{\Sigma}$ *and denote* $\vdash_{\mathsf{HK}\mathbf{\Sigma}} A$. *Given a set* $\Gamma \cup \{ A \} \subseteq \mathsf{Form}_{\mathbf{ML}}$ *of formulas, if there exists some finite subset* $\Gamma' \subseteq \Gamma$ *such that* $\vdash_{\mathsf{HK}\mathbf{\Sigma}} (\bigwedge \Gamma') \to A$, *we say that* A *is* derivable *from* Γ *in* $\mathsf{HK}\mathbf{\Sigma}$ *and denote* $\Gamma \vdash_{\mathsf{HK}\mathbf{\Sigma}} A$.

A derivation in $\mathsf{HK}\mathbf{\Sigma}$ in what follows, we regard $\Sigma \subseteq \{\mathrm{T}_\Box, \mathrm{B}_\Box, 4_\Box, \mathrm{D}_\Box, 5_\Box \mid \Box \in \mathsf{MOD}\}$ till end of this section.

Theorem 2.9 (Soundness). *Let* $\Sigma \subseteq \{ \mathrm{T}_\Box, \mathrm{B}_\Box, 4_\Box, \mathrm{D}_\Box, 5_\Box \mid \Box \in \mathsf{MOD} \}$ *and* $\mathbb{F}_\Sigma$ *be the class of all frames defined by* Σ (*cf. Proposition 2.2*). *For all* $A \in \mathsf{Form}_{\mathbf{ML}}$,

$$\text{if } \vdash_{\mathsf{HK}\mathbf{\Sigma}} A, \text{ then } \mathbb{F}_\Sigma \models A.$$

Definition 2.10. *Given* $\Gamma \subseteq \mathsf{Form}_{\mathbf{ML}}$. *We say that* Γ *is* $\mathsf{HK}\mathbf{\Sigma}$-inconsistent *if* $\Gamma \vdash_{\mathsf{HK}\mathbf{\Sigma}} \bot$ *and that* Γ *is* $\mathsf{HK}\mathbf{\Sigma}$-consistent *if* Γ *is not* $\mathsf{HK}\mathbf{\Sigma}$*-inconsistent, i.e.,* $\Gamma \nvdash_{\mathsf{HK}\mathbf{\Sigma}} \bot$. *Then,* Γ *is* maximal *if* $A \in \Gamma$ *or* $\neg A \in \Gamma$ *for all* $A \in \mathbf{ML}$. Γ *is a* maximally $\mathsf{HK}\mathbf{\Sigma}$-consistent set (*notation:* $\mathsf{HK}\mathbf{\Sigma}$*-MCS*) *if* Γ *is* $\mathsf{HK}\mathbf{\Sigma}$*-consistent and maximal.*

Proposition 2.11. *Let* Γ *be any* $\mathsf{HK}\mathbf{\Sigma}$*-MCS.*

1. $\Gamma \vdash_{\mathsf{HK}\mathbf{\Sigma}} A$ iff $A \in \Gamma$.
2. *if* $A \in \Gamma$ *and* $\vdash_{\mathsf{HK}\mathbf{\Sigma}} A \to B$, *then* $B \in \Gamma$.
3. $\neg A \in \Gamma$ iff $A \notin \Gamma$.
4. $A \to B \in \Gamma$ iff $A \notin \Gamma$ *or* $B \in \Gamma$.

Lemma 2.12 (Lindenbaum's Lemma). *If* Γ *is any* $\mathsf{HK}\mathbf{\Sigma}$*-consistent set, then there exists an* $\mathsf{HK}\mathbf{\Sigma}$*-MCS* Γ^+ *such that* $\Gamma \subseteq \Gamma^+$.

Definition 2.13. *For an axiomatic extension* $\mathsf{HK}\mathbf{\Sigma}$, *the* canonical model $\mathfrak{M}^{\mathsf{HK}\mathbf{\Sigma}} = (W^{\mathsf{HK}\mathbf{\Sigma}}, (R_\Box^{\mathsf{HK}\mathbf{\Sigma}})_{\Box \in \mathsf{MOD}}, V^{\mathsf{HK}\mathbf{\Sigma}})$ *is defined by:*

- $W^{\mathsf{HK}\mathbf{\Sigma}} := \{ \Gamma \mid \Gamma$ *is an* $\mathsf{HK}\mathbf{\Sigma}$*-MCS*$\}$, *i.e.,* $W^{\mathsf{HK}\mathbf{\Sigma}}$ *is the set of all* $\mathsf{HK}\mathbf{\Sigma}$*-MCSs.*
- $\Gamma R_\Box^{\mathsf{HK}\mathbf{\Sigma}} \Delta$ iff $\Box A \in \Gamma$ *implies* $A \in \Delta$ *for all formulas* A.
- $\Gamma \in V^{\mathsf{HK}\mathbf{\Sigma}}(p)$ iff $p \in \Gamma$.

Lemma 2.14. *Given any* $\mathsf{HK}\mathbf{\Sigma}$*-MCS* Γ,

$$\text{if } \Box A \notin \Gamma, \text{ then } \{ \neg A \} \cup \{ B \mid \Box B \in \Gamma \} \nvdash_{\mathsf{HK}\mathbf{\Sigma}} \bot.$$

Lemma 2.15 (Truth Lemma). *Let* Γ *be any* $\mathsf{HK\Sigma}$*-MCS. For all* $A \in \mathsf{Form}_{\mathbf{ML}}$*,*

$$\mathfrak{M}^{\mathsf{HK\Sigma}}, \Gamma \models A \text{ iff } A \in \Gamma.$$

Lemma 2.16. *Let* $\mathfrak{M}^{\mathsf{HK\Sigma}} = (W^{\mathsf{HK\Sigma}}, (R_{\Box}^{\mathsf{HK\Sigma}})_{\Box\in\mathsf{MOD}}, V^{\mathsf{HK\Sigma}})$ *be the canonical model for an axiomatic extension* $\mathsf{HK\Sigma}$*. For all* $A \in \mathsf{Form}_{\mathbf{ML}}$*,*

$$\textit{if } \nvdash_{\mathsf{HK\Sigma}} A, \textit{ then } \mathfrak{M}^{\mathsf{HK\Sigma}} \not\models A.$$

Proof. Suppose that $\nvdash_{\mathsf{HK\Sigma}} A$. By supposition of $\nvdash_{\mathsf{HK\Sigma}} A$, $\{\neg A\}$ is $\mathsf{HK\Sigma}$-consistent, i.e., $\{\neg A\} \nvdash_{\mathsf{HK\Sigma}} \bot$. By Lemma 2.12, there exists an $\mathsf{HK\Sigma}$-MCS Γ such that $\{\neg A\} \subseteq \Gamma$. By Lemma 2.15, we obtain $\mathfrak{M}^{\mathsf{HK\Sigma}}, \Gamma \models \neg A$, i.e., $\mathfrak{M}^{\mathsf{HK\Sigma}}, \Gamma \not\models A$, as desired. □

Lemma 2.17. *Let* $\mathfrak{M}^{\mathsf{HK\Sigma}} = (W^{\mathsf{HK\Sigma}}, (R_{\Box}^{\mathsf{HK\Sigma}})_{\Box\in\mathsf{MOD}}, V^{\mathsf{HK\Sigma}})$ *be the canonical model for an axiomatic extension* $\mathsf{HK\Sigma}$*.*

$$\Gamma R_{\Box}^{\mathsf{HK\Sigma}} \Delta \text{ iff } A \in \Delta \textit{ implies } \Diamond A \in \Gamma \textit{ for all } A.$$

Lemma 2.18. *Given the canonical model* $\mathfrak{M}^{\mathsf{HK\Sigma}} = (W^{\mathsf{HK\Sigma}}, (R_{\Box}^{\mathsf{HK\Sigma}})_{\Box\in\mathsf{MOD}}, V^{\mathsf{HK\Sigma}})$ *for an axiomatic extension* $\mathsf{HK\Sigma}$*,*

1. *If* $\vdash_{\mathsf{HK\Sigma}} \Box A \to A$ *for all formulas* A*, then* $R_{\Box}^{\mathsf{HK\Sigma}}$ *is reflexive.*
2. *If* $\vdash_{\mathsf{HK\Sigma}} A \to \Box\Diamond A$ *for all formulas* A*, then* $R_{\Box}^{\mathsf{HK\Sigma}}$ *is symmetric.*
3. *If* $\vdash_{\mathsf{HK\Sigma}} \Box A \to \Box\Box A$ *for all formulas* A*, then* $R_{\Box}^{\mathsf{HK\Sigma}}$ *is transitive.*
4. *If* $\vdash_{\mathsf{HK\Sigma}} \Box A \to \Diamond A$ *for all formulas* A*, then* $R_{\Box}^{\mathsf{HK\Sigma}}$ *is serial.*
5. *If* $\vdash_{\mathsf{HK\Sigma}} \Diamond A \to \Box\Diamond A$ *for all formulas* A*, then* $R_{\Box}^{\mathsf{HK\Sigma}}$ *is Euclidean.*

Theorem 2.19 (Completeness). *Let* $\Sigma \subseteq \{\mathrm{T}_{\Box}, \mathrm{B}_{\Box}, 4_{\Box}, \mathrm{D}_{\Box}, 5_{\Box} \mid \Box \in \mathsf{MOD}\}$ *and* $\mathbb{F}_{\Sigma}$ *be the class of frames defined by* Σ*. For all* $A \in \mathsf{Form}_{\mathbf{ML}}$*,*

$$\textit{if } \mathbb{F}_{\Sigma} \models A, \textit{ then } \vdash_{\mathsf{HK\Sigma}} A.$$

Proof. By contraposition. Our goal is to show that if $\nvdash_{\mathsf{HK\Sigma}} A$, then $\mathfrak{F} \not\models A$ for some $\mathfrak{F} \in \mathbb{F}_{\Sigma}$. Suppose that $\nvdash_{\mathsf{HK\Sigma}} A$. It suffices to construct a counter model $(\mathfrak{F}, V)$ such that $(\mathfrak{F}, V), w \not\models A$ for some $w \in |\mathfrak{F}|$. By our supposition and Lemma 2.16, we get $\mathfrak{M}^{\mathsf{HK\Sigma}} \not\models A$, i.e., $(W^{\mathsf{HK\Sigma}}, (R_{\Box}^{\mathsf{HK\Sigma}})_{\Box\in\mathsf{MOD}}, V^{\mathsf{HK\Sigma}}) \not\models A$. It suffices to show $(W^{\mathsf{HK\Sigma}}, (R_{\Box}^{\mathsf{HK\Sigma}})_{\Box\in\mathsf{MOD}}) \in \mathbb{F}_{\Sigma}$. Since $\mathbb{F}_{\Sigma}$ is the class of all frames which satisfy each frame property corresponding to each formula of Σ, our goal is to show that $(W^{\mathsf{HK\Sigma}}, (R_{\Box}^{\mathsf{HK\Sigma}})_{\Box\in\mathsf{MOD}})$ satisfies each frame property corresponding to each formula of Σ. This is already shown by Lemma 2.18.

□

2.2 Propositional Dynamic Logic

2.2.1 Syntax and Kripke Semantics

This section reviews the syntax and the semantics of *regular* propositional dynamic logic (**PDL**) [14] and its iteration-free fragment. We extend multi-modal logic to handle the

Table 2.5: Reading of Programs

Program	Reading
$\alpha \cup \beta$	'choose either α or β nondeterministically and execute the chosen one.'
$\alpha; \beta$	'execute α, then execute β.'
$?A$	'test A, proceed if A is true, fail otherwise.'
α^*	'execute α a nondeterministically chosen finite number of times (zero or more).'

notion of programs that allows us to capture changes in dynamic systems. Therefore, this logic is an important basis for dynamic logic of relation changers to handle dynamics of agent's knowledge (or belief). Besides PROP, let $\mathsf{AP} = \{ a, b, \dots \}$ be a finite set of *atomic programs*. In addition to the propositional connectives, the language $\mathcal{L}_{\mathbf{PDL}}$ of (regular) propositional dynamic logic has the following operators: the (*sequential*) *composition* ;, the *nondeterministic choice* $\cup$, the *iteration* *, the *test* ?, and the *necessity* $[\alpha]$ *for program* α. The operators ; , $\cup$, * and ? are called *program constructors*. We regard a finite set MOD as $\{ [a] \mid a \in \mathsf{AP} \}$. A sets $\mathsf{Form}_{\mathbf{PDL}}$ of formulas and PR of programs over the language $\mathcal{L}_{\mathbf{PDL}}$ are defined by simultaneous induction as follows:

$$\begin{array}{rcl} \mathsf{Form}_{\mathbf{PDL}} \ni A & ::= & p \mid \neg A \mid (A \to A) \mid [\alpha]A \\ \mathsf{PR} \ni \alpha & ::= & a \mid \alpha \cup \alpha \mid \alpha; \alpha \mid ?A \mid \alpha^* \end{array}$$

where $p \in \mathsf{PROP}$ and $a \in \mathsf{AP}$. A formula $[\alpha]A$ stands for 'after executing α, it is necessary that A.' Then, readings of programs are shown in Table 2.5.

We introduce abbreviations for the conjunction $\wedge$, the disjunction $\vee$, the logical equivalence $\leftrightarrow$, the truth $\top$ and the falsity $\bot$ as did in Section 2.1.1 and also introduce the dual operator $\langle\alpha\rangle$ of $[\alpha]$ by $\langle\alpha\rangle A := \neg[\alpha]\neg A$. In addition, we can write some standard programming constructs by definitional abbreviation. For example,

$$\begin{array}{lcl} \text{if } A \text{ then } \alpha \text{ else } \beta & \overset{\text{def}}{\Leftrightarrow} & (?A; \alpha) \cup (?\neg A; \beta), \\ \text{while } A \text{ do } \alpha & \overset{\text{def}}{\Leftrightarrow} & (?A; \alpha)^*; \neg A. \end{array}$$

We also define $\mathbf{PDL}^-$ as iteration-free (or star-free) fragment of $\mathbf{PDL}$. That is, the language $\mathcal{L}_{\mathbf{PDL}^-}$ and $\mathsf{Form}_{\mathbf{PDL}^-}$ does not contain $*$ and $[\alpha^*]A$, respectively. In what follows, we regard definitions for $\mathbf{PDL}^-$ are also provided by those for $\mathbf{PDL}$ except $*$.

The semantics of $\mathbf{PDL}$ comes from that of modal logic (cf. Section 2.1.1). A model $\mathfrak{M}$ is a tuple $(W, (R_a)_{a \in \mathsf{AP}}, V)$ where W is a non-empty set of states, $R_a \subseteq W \times W$ is an *accessibility relation of atomic program* a, and $V : \mathsf{Prop} \to \mathcal{P}(W)$ is a valuation function.

Given any model $\mathfrak{M} = (W, (R_a)_{a \in \mathsf{AP}}, V)$, any state $w \in W$, and any formula A, the *satisfaction relation* $\mathfrak{M}, w \models A$ is defined by simultaneous induction as follows:

$$
\begin{array}{lll}
\mathfrak{M}, w \models p & \text{iff} & w \in V(p), \\
\mathfrak{M}, w \models \neg A & \text{iff} & \mathfrak{M}, w \not\models A, \\
\mathfrak{M}, w \models A \to B & \text{iff} & \mathfrak{M}, w \models A \text{ implies } \mathfrak{M}, w \models B, \\
\mathfrak{M}, w \models [\alpha]A & \text{iff} & \text{for all } v \in W : wR_\alpha v \text{ implies } \mathfrak{M}, v \models A, \\
wR_{\alpha\cup\beta}v & \text{iff} & wR_\alpha v \text{ or } wR_\beta v, \\
wR_{\alpha;\beta}v & \text{iff} & \text{for some } u \in W : wR_\alpha u \text{ and } uR_\beta v, \\
wR_{?A}v & \text{iff} & w = v \text{ and } \mathfrak{M}, w \models A, \\
wR_{\alpha^*}v & \text{iff} & \text{for some } 0 \leq n, wR_{\alpha^n}v,
\end{array}
$$

where α^n is n-fold composition of program α defined inductively by:

$$
\begin{array}{lll}
\alpha^0 & := & ?\top, \\
\alpha^{n+1} & := & \alpha; \alpha^n.
\end{array}
$$

The *truth set* $[\![A]\!]_{\mathfrak{M}}$ is defined by $[\![A]\!]_{\mathfrak{M}} = \{ w \in W \mid \mathfrak{M}, w \models A \}$. Then, we can obtain the following:

$$
\begin{array}{lll}
[\![p]\!]_{\mathfrak{M}} & := & V(p), \\
[\![\neg A]\!]_{\mathfrak{M}} & := & W \setminus [\![A]\!]_{\mathfrak{M}}, \\
[\![A \to B]\!]_{\mathfrak{M}} & := & (W \setminus [\![A]\!]_{\mathfrak{M}}) \cup [\![B]\!]_{\mathfrak{M}}, \\
[\![[\alpha]A]\!]_{\mathfrak{M}} & := & \{ w \in W \mid [\![\alpha]\!]_{\mathfrak{M}}(w) \subseteq [\![A]\!]_{\mathfrak{M}} \}, \\
[\![a]\!]_{\mathfrak{M}} & := & R_a, \\
[\![\alpha \cup \beta]\!]_{\mathfrak{M}} & := & [\![\alpha]\!]_{\mathfrak{M}} \cup [\![\beta]\!]_{\mathfrak{M}}, \\
[\![\alpha; \beta]\!]_{\mathfrak{M}} & := & [\![\alpha]\!]_{\mathfrak{M}} \circ [\![\beta]\!]_{\mathfrak{M}}, \\
[\![?A]\!]_{\mathfrak{M}} & := & \{ (w, v) \mid w = v \text{ and } w \in [\![A]\!]_{\mathfrak{M}} \}, \\
[\![\alpha^*]\!]_{\mathfrak{M}} & := & [\![\alpha]\!]^n_{\mathfrak{M}} \text{ for some } 0 \leq n.
\end{array}
$$

where $R \circ S$ is the relational composition of R with S, i.e., $(w, v) \in R \circ S$ iff $(w, u) \in R$ and $(u, v) \in S$ for some $u \in W$, $[\![\alpha]\!]_{\mathfrak{M}}(w) := \{ v \in W \mid (w, v) \in [\![\alpha]\!]_{\mathfrak{M}} \}$, and $[\![\alpha]\!]^n_{\mathfrak{M}}$ is defined by $[\![\alpha]\!]^0_{\mathfrak{M}} := [\![?\top]\!]_{\mathfrak{M}}$ and $[\![\alpha]\!]^{n+1}_{\mathfrak{M}} := [\![\alpha]\!]_{\mathfrak{M}} \circ [\![\alpha]\!]^n_{\mathfrak{M}}$. The notion of *validity* is almost identical to that of modal logic (see Section 2.1.1) where Kripke model $\mathfrak{M}$ is extended to handle $(W, (R_a)_{a \in \mathsf{AP}}, V)$. Except for this point, the notion of validity is also defined over the various levels of semantical structure, as in Section 2.1.1.

2.2.2 Hilbert-style Axiomatization HPDL

Table 2.6 shows the sound and complete Hilbert-style axiomatization **HPDL** [14]. Similarly to **HKΣ**, axioms and rules are given as schemes (cf. Section 2.1.3). Then, the sound and complete axiomatization $\mathbf{HPDL}^-$ is the result of dropping axioms ($[*]$) and ($\mathbf{Ind}_{[\alpha]}$) from **HPDL**. Definitions of a *derivation* in **HPDL** (or $\mathbf{HPDL}^-$) and *theorem* in **HPDL** (or $\mathbf{HPDL}^-$) are provided by the same manner as in **HKΣ** in Section 2.1.3. We denote A is a theorem in **HPDL** (or $\mathbf{HPDL}^-$) by $\vdash_{\mathbf{HPDL}} A$ (or $\vdash_{\mathbf{HPDL}^-} A$).

Here, we state the soundness and the completeness of **HPDL** as follows.

Theorem 2.20. *Let A be a formula in $\mathsf{Form}_{\mathbf{PDL}}$ and $\mathbb{M}$ be the class of all models.*

$$\mathbb{M} \models A \text{ iff } \vdash_{\mathbf{HPDL}} A.$$

It is known that the completeness of $\mathsf{H}\mathbf{PDL}$ can be shown by filtration technique with nonstandard Kripke frame. The readers are referred to [14] for the details. It is also known that the soundness and completeness also hold for $\mathsf{H}\mathbf{PDL}^{-}$. Since $\mathsf{H}\mathbf{PDL}^{-}$ becomes important in this thesis, we review the completeness of $\mathsf{H}\mathbf{PDL}^{-}$ in the remaining of this section. We can show the completeness of $\mathsf{H}\mathbf{PDL}^{-}$ by the method of the canonical model (cf. Section 2.1.3). Let $\Gamma \subseteq \mathsf{Form}_{\mathbf{PDL}^{-}}$. We use the same manner in Section 2.1.3 to define the *consistency* (Γ is $\mathsf{H}\mathbf{PDL}^{-}$-consistent), the *maximality* (Γ is maximal), and a *maximally consistent set* (Γ is a maximally $\mathsf{H}\mathbf{PDL}^{-}$-consistent set).

Definition 2.21. *For* $\mathsf{H}\mathbf{PDL}^{-}$, *the* canonical model $\mathfrak{M}^{\mathsf{H}\mathbf{PDL}^{-}} = (W^{\mathsf{H}\mathbf{PDL}^{-}}, (R_a^{\mathsf{H}\mathbf{PDL}^{-}})_{a \in \mathsf{AP}}, V^{\mathsf{H}\mathbf{PDL}^{-}})$ *is defined by:*

- $W^{\mathsf{H}\mathbf{PDL}^{-}} := \{\, \Gamma \mid \Gamma$ *is an* $\mathsf{H}\mathbf{PDL}^{-}$*-MCS*$\}$.
- $\Gamma R_a^{\mathsf{H}\mathbf{PDL}^{-}} \Delta$ iff $[a]A \in \Gamma$ *implies* $A \in \Delta$ *for all formulas* A.
- $\Gamma \in V^{\mathsf{H}\mathbf{PDL}^{-}}(p)$ iff $p \in \Gamma$.

In addition to Proposition 2.11 in terms of $\mathsf{H}\mathbf{PDL}^{-}$, we need the following proposition.

Proposition 2.22. *Let* Γ *be any* $\mathsf{H}\mathbf{PDL}^{-}$*-MCS.*

1. $[\alpha \cup \beta]A \in \Gamma$ iff $[\alpha]A \in \Gamma$ *and* $[\beta]A \in \Gamma$.
2. $[\alpha; \beta]A \in \Gamma$ iff $[\alpha][\beta]A \in \Gamma$.
3. $[?B]A \in \Gamma$ iff $(B \to A) \in \Gamma$.

Lemma 2.12 (Lindenbaum's Lemma) also holds in terms of $\mathsf{H}\mathbf{PDL}^{-}$.

Lemma 2.23 (Lindenbaum's Lemma)**.** *If* Γ *is any* $\mathsf{H}\mathbf{PDL}^{-}$*-consistent set, then there exists an* $\mathsf{H}\mathbf{PDL}^{-}$*-MCS* Γ^{+} *such that* $\Gamma \subseteq \Gamma^{+}$.

Then, we extend Lemma 2.15 (Truth Lemma) to handle programs in $\mathbf{PDL}^{-}$ as follows.

Lemma 2.24 (Truth Lemma)**.** *Let* Γ *be any* $\mathsf{H}\mathbf{PDL}^{-}$*-MCS.*

1. $\mathfrak{M}^{\mathsf{H}\mathbf{PDL}^{-}}, \Gamma \models A$ iff $A \in \Gamma$ *for all* $A \in \mathsf{Form}_{\mathbf{PDL}^{-}}$.
2. $R_\alpha^{\mathsf{H}\mathbf{PDL}^{-}} = S_\alpha^{\mathsf{H}\mathbf{PDL}^{-}}$ *for all* $\alpha \in \mathsf{PR}$, *where* $\Gamma S_\alpha^{\mathsf{H}\mathbf{PDL}^{-}} \Delta$ iff $[\alpha]A \in \Gamma$ *implies* $A \in \Delta$ *for all* $A \in \mathsf{Form}_{\mathbf{PDL}^{-}}$.

Theorem 2.25. *Let* A *be a formula in* $\mathsf{Form}_{\mathbf{PDL}^{-}}$ *and* $\mathbb{M}$ *be the class of all models.*

$$\mathbb{M} \models A \text{ iff } \vdash_{\mathsf{H}\mathbf{PDL}^{-}} A.$$

Proof. Since the soundness is easy to establish, we focus on the completeness with respect to the class of all models. We show the completeness by contraposition. Our goal is to show that if $\nvdash_{\mathsf{H}\mathbf{PDL}^{-}} A$, then $\mathbb{M} \not\models A$. With the help of the canonical model for $\mathsf{H}\mathbf{PDL}^{-}$ and Lemmas 2.23 and 2.24, we can show that if $\nvdash_{\mathsf{H}\mathbf{PDL}^{-}} A$, then $\mathfrak{M}^{\mathsf{H}\mathbf{PDL}^{-}} \not\models A$ (i.e., $\mathbb{M} \not\models A$) by the same argument of Lemma 2.16 in terms of $\mathsf{H}\mathbf{PDL}^{-}$. $\square$

Table 2.6: Hilbert-style Axiomatization $\mathsf{H}\mathbf{PDL}$

(**Taut**)	All instances of propositional tautologies
($\mathbf{K}_{[\alpha]}$)	$[\alpha](A \to B) \to ([\alpha]A \to [\alpha]B)$
($[\cup]$)	$[\alpha \cup \beta]A \leftrightarrow [\alpha]A \wedge [\beta]A$
($[;]$)	$[\alpha;\beta]A \leftrightarrow [\alpha][\beta]A$
($[?]$)	$[?B]A \leftrightarrow (B \to A)$
($[*]$)	$A \wedge [\alpha][\alpha^*]A \leftrightarrow [\alpha^*]A$
($\mathbf{Ind}_{[\alpha]}$)	$A \wedge [\alpha^*](A \to [\alpha]A) \to [\alpha^*]A$
(**MP**)	From A and $A \to B$, infer B
($\mathbf{Nec}_{[\alpha]}$)	From A, infer $[\alpha]A$

2.3 Public Announcement Logic

2.3.1 Syntax and Kripke Semantics

Public announcement logic (**PAL**) [32] is a variant of dynamic epistemic logics (**DEL**s) that models changes of agent's knowledge and belief over Kripke semantics. This section reviews the syntax and the semantics of **PAL**. We introduce the logic based on the standard epistemic logic of knowledge. Besides PROP, let G be a finite set of agents. In addition to the propositional connectives, the language $\mathcal{L}_{\mathbf{PAL}}$ of public announcement logic consists of the following operators: *knowledge operators* $[\mathsf{K}_a]$ $(a \in \mathsf{G})$ and *public announcement operator* $[!A]$. Then, we regard a finite set MOD as $\{ [\mathsf{K}_a] \mid a \in \mathsf{G} \}$. A set $\mathsf{Form}_{\mathbf{PAL}}$ of formulas of the language $\mathcal{L}_{\mathbf{PAL}}$ is defined by simultaneous induction as follows:

$$\mathsf{Form}_{\mathbf{PAL}} \ni A ::= p \mid \neg A \mid (A \to A) \mid [\mathsf{K}_a]A \mid [!A]A,$$

where $p \in \mathsf{PROP}$ and $a \in \mathsf{G}$. The knowledge operators $[\mathsf{K}_a]$ comes from epistemic logic. Formulas $[\mathsf{K}_a]A$ and $[!B]A$ stand for 'agent a knows that A,' and 'after the truthful announcement that B, A holds.' We introduce defined abbreviations for the conjunction $\wedge$, the disjunction $\vee$, the logical equivalence $\leftrightarrow$, the truth $\top$ and the falsity $\bot$ as did in Section 2.1.1 and also introduce the dual operator $\langle \mathsf{K}_a \rangle$ of $[\mathsf{K}_a]$ by $\langle \mathsf{K}_a \rangle A := \neg[\mathsf{K}_a]\neg A$.

In **PAL**, we use Kripke model $\mathfrak{M} = (W, (R_a)_{a \in \mathsf{G}}, V)$, but all the accessibility relations $R_a (a \in \mathsf{G})$ satisfy reflexivity, transitivity and symmetric, i.e., each R_a is the *equivalence relation.* Such a definition of the model comes from the semantics of epistemic logic, and the model is called an *epistemic model.* In what follows, we denote the equivalence relation R_a by $\sim_a$, and the model by $(W, (\sim_a)_{a \in \mathsf{G}}, V)$ explicitly. We also use infix notation $w \sim_a v$ to mean $(w, v) \in \sim_a$.

Given any epistemic model $\mathfrak{M} = (W, (\sim_a)_{a \in \mathsf{G}}, V)$, any state $w \in W$, and any formula A, the satisfaction relation $\mathfrak{M}, w \models A$ is defined as follows:

$$\begin{array}{lll}
\mathfrak{M}, w \models p & \text{iff} & w \in V(p), \\
\mathfrak{M}, w \models \neg A & \text{iff} & \mathfrak{M}, w \not\models A, \\
\mathfrak{M}, w \models A \to B & \text{iff} & \mathfrak{M}, w \models A \text{ implies } \mathfrak{M}, w \models B, \\
\mathfrak{M}, w \models [\mathsf{K}_a]A & \text{iff} & \text{for all } v \in W, w \sim_a v \text{ implies } \mathfrak{M}, v \models A, \\
\mathfrak{M}, w \models [!B]A & \text{iff} & \mathfrak{M}, w \models B \text{ implies } \mathfrak{M}^{!B}, w \models A,
\end{array}$$

where $\mathfrak{M}^{!B} = (W^{!B}, (\sim_a^{!B})_{a \in \mathsf{G}}, V^{!B})$ is defined by:

$$\begin{array}{lcl} W^{!B} & := & [\![B]\!]_{\mathfrak{M}} = \{\, w \in W \mid \mathfrak{M}, w \models B \,\}, \\ \sim_a^{!B} & := & \sim_a \cap\, [\![B]\!]_{\mathfrak{M}} \times [\![B]\!]_{\mathfrak{M}}, \\ V^{!B}(p) & := & V(p) \cap [\![B]\!]_{\mathfrak{M}}. \end{array}$$

The notion of validity is defined over various levels of semantic structure, as shown in Section 2.1.1. The difference is that we are using epistemic model. Namely, every accessibility relation is defined as an equivalence relation.

2.3.2 Hilbert-style Axiomatization HPAL

The sound and complete Hilbert-style system **HPAL** is shown in Table 2.7 [40].[1] From Table 2.7, we can find axiom schemes $\mathbf{T}_{[\mathsf{K_a}]}$, $\mathbf{4}_{[\mathsf{K_a}]}$ and $\mathbf{5}_{[\mathsf{K_a}]}$. These axioms come from axiom schemes **T**, **4** and **5** that we explained in Section 2.1.2 (cf. Table 2.1), but they are bit extended to handle knowledge operators $[\mathsf{K}_a]$ ($a \in \mathsf{G}$). Hence, we can regard the system **HPAL** as an axiomatic extension of **HS5** in terms of knowledge operators $[\mathsf{K}_a]$ ($a \in \mathsf{G}$). We use the same manner in Section 2.1.3 to define a *derivation* and a *theorem* in **HPAL**. Then, we denote A is a theorem in **HPAL** by $\vdash_{\mathbf{HPAL}} A$.

In what follows of this section, we review the standard argument of completeness in **PAL** that was shown in [40].

Definition 2.26. *The translation $t : \mathsf{Form}_{\mathbf{PAL}} \to \mathsf{Form}_{\mathbf{ML}}$ for* **PAL** *is defined by:*

$$\begin{array}{lcl} t(p) & = & p, \\ t(\neg A) & = & \neg t(A), \\ t(A \to B) & = & t(A) \to t(B), \\ t([\mathsf{K}_a]A) & = & [\mathsf{K}_a]t(A), \\ t([!A]p) & = & t(A \to p), \\ t([!A]\neg B) & = & t(A \to \neg[!A]B), \\ t([!A](B \to C)) & = & t([!A]B) \to t([!A]C), \\ t([!A][\mathsf{K}_a]B) & = & t(A \to [\mathsf{K}_a][!A]B), \\ t([!A][!B]C) & = & t([!(A \wedge [!A]B)]C). \end{array}$$

Lemma 2.27. *Given any formula $A \in \mathsf{Form}_{\mathbf{PAL}}$,*

$$\vdash_{\mathbf{HPAL}} A \leftrightarrow t(A).$$

Theorem 2.28. *Let A be a formula in $\mathsf{Form}_{\mathbf{PAL}}$ and $\mathbb{F}$ be the class of frames defined by $\{\, \mathrm{T}_{[\mathsf{K}_a]}, 4_{[\mathsf{K}_a]}, 5_{[\mathsf{K}_a]} \mid a \in \mathsf{G}$ and $[\mathsf{K}_a] \in \mathsf{MOD} \,\}$ (see also Table 2.1). Then, let $\mathbb{M}$ be the class of models which consists of models $(\mathfrak{F}, V)$ for all frames $\mathfrak{F} \in \mathbb{F}$ and all valuations V.*

$$\mathbb{M} \models A \text{ iff } \vdash_{\mathbf{HPAL}} A.$$

Proof. We divide our proof into the soundness part (the direction from right to left) and the completeness part (from left to right) as follows.

Soundness part We show that if $\vdash_{\mathbf{HPAL}}$, then $\mathbb{M} \models A$ for all A. One can easily check that each axiom in **HPAL** is valid in the class $\mathbb{M}$ of all epistemic models and each rule in **HPAL** preserves the validity in $\mathbb{M}$.

[1] There are another Hilbert-style axiomatization for **PAL** [41] which does not use recursion axioms.

Table 2.7: Hilbert-style Axiomatization $\mathsf{H}\mathbf{PAL}$

(**Taut**)	All instances of propositional tautologies
$(\mathbf{K}_{[\mathsf{K_a}]})$	$[\mathsf{K}_a](A \to B) \to ([\mathsf{K}_a]A \to [\mathsf{K}_a]B)$
$(\mathbf{T}_{[\mathsf{K_a}]})$	$[\mathsf{K}_a]A \to A$
$(\mathbf{4}_{[\mathsf{K_a}]})$	$[\mathsf{K}_a]A \to [\mathsf{K}_a][\mathsf{K}_a]A$
$(\mathbf{5}_{[\mathsf{K_a}]})$	$\neg[\mathsf{K}_a]A \to [\mathsf{K}_a]\neg[\mathsf{K}_a]A$
$([!]\mathbf{pr})$	$[!A]p \leftrightarrow (A \to p)$
$([!]\neg)$	$[!A]\neg B \leftrightarrow (A \to \neg[!A]B)$
$([!]\to)$	$[!A](B \to C) \leftrightarrow ([!A]B \to [!A]C)$
$([!][\mathsf{K_a}])$	$[!A][\mathsf{K}_a]B \leftrightarrow (A \to [\mathsf{K}_a][!A]B)$
$([!][!])$	$[!A][!B]C \leftrightarrow [!A \wedge [!A]B]C$
(**MP**)	From A and $A \to B$, infer B
$(\mathbf{Nec}_{[\mathsf{K_a}]})$	From A, infer $[\mathsf{K}_a]A$

Completeness part We show that if $\mathbb{M} \models A$, then $\vdash_{\mathsf{H}\mathbf{PAL}} A$ for all A. We can reduce the completeness of $\mathsf{H}\mathbf{PAL}$ to that of $\mathsf{HS5}$ (cf. Theorem 2.19 in Section 2.1.3) in terms of knowledge operators $[\mathsf{K}_a]$ $(a \in \mathsf{G})$. Fix any formula A and suppose that $\mathbb{M} \models A$. By the soundness part and Lemma 2.27, we get $\mathbb{M} \models A \leftrightarrow t(A)$. Then, by this and our assumption of $\mathbb{M} \models A$, we obtain that $\mathbb{M} \models t(A)$. By this and the completeness of $\mathsf{HS5}$ (Theorem 2.19) in terms of knowledge operators $[\mathsf{K}_a]$ $(a \in \mathsf{G})$, we have that $\vdash_{\mathsf{HS5}} t(A)$. Since $\mathsf{H}\mathbf{PAL}$ is an axiomatic extension of $\mathsf{HS5}$, we also have that $\vdash_{\mathsf{H}\mathbf{PAL}} t(A)$. Finally, by Lemma 2.27 and $\vdash_{\mathsf{H}\mathbf{PAL}} t(A)$, we obtain $\vdash_{\mathsf{H}\mathbf{PAL}} A$.

□

2.4 Boolean Matrix

Mathematical operations and some properties of Boolean matrices are slightly different from real-valued matrices. For example, the inverse operation of multiplication seems not well-defined,[2] and the addition of the same matrices satisfies idempotence, i.e., the resultant matrix of the addition is equal to the original one.

Throughout this paper, we use the symbol M, to denote a *Boolean matrix*, i.e., each element of the matrix belongs to the set $\{0, 1\}$. We use the symbol M as a superscript ^{M} with a symbol or expressions (e.g., X^M and $(X+Y)^M$) to denote a matrix representation of them. If the representing matrix is clear from the context, we omit 'M' from such 'X^M' and just write 'X'. Moreover, $M(m \times n)$ means the set of all (Boolean) $m \times n$ matrices, where m and n are the numbers of rows and columns, respectively. In the usual sense, $1 \times n$ and $m \times 1$ matrices are called Boolean row vector and column vector, respectively. Let M be an $m \times n$ matrix, $1 \leq i \leq m$ and $1 \leq j \leq n$. $M(i,j)$ denotes the element in the i-th row and j-th column entry. Moreover, $\boldsymbol{E}$, $\mathbf{0}$ and $\mathbf{1}$ denote the *unit square matrix*

[2] The inverse operation of the Boolean addition, i.e., subtraction is not well-defined over Boolean values. Consequently, subtraction for a Boolean matrix cannot make sense.

($\boldsymbol{E}(i,j) = 1$ if $i = j$; 0 otherwise), *complete matrix* ($\mathbf{1}(i,j) = 1$ for all i and j), and *zero matrix* ($\mathbf{0}(i,j) = 0$ for all i and j), respectively.[3]

The Boolean operations of addition '+', multiplication '·' and complement '$^-$' for the element of Boolean matrices correspond to the logical operations of '$\vee$', '$\wedge$' and '$\neg$', respectively. These operations are also defined to the level of matrices. Let $M, M_1, M_2 \in M(m \times n)$. For all i and j, the complement $\overline{M}$, the addition $M_1 + M_2$ and the conjunction $M_1 \wedge M_2$ are defined by:

$$\begin{array}{lcl} \overline{M}(i,j) & := & \overline{M(i,j)}, \\ (M_1 + M_2)(i,j) & := & M_1(i,j) + M_2(i,j), \\ (M_1 \wedge M_2)(i,j) & := & M_1(i,j) \cdot M_2(i,j). \end{array}$$

Given any $M_1 \in M(m \times l)$ and any $M_2 \in M(l \times n)$, the multiplication $M_1 M_2$ of matrices is defined by:

$$(M_1 M_2)(i,j) = \sum\nolimits_{1 \le k \le n} (M_1(i,k) \cdot M_2(k,j)).$$

The transposition ${}^t M$ is defined as: ${}^t M(i,j) = M(j,i)$ for all i and j. In the below, we summarize basic properties of addition, multiplication and transposition of Boolean matrices.

Proposition 2.29. *For any $M \in M(m \times n)$,*

1. $M = M + M$.
2. $M = \boldsymbol{E} M$.
3. $M = \mathbf{0} + M$.
4. $M = \mathbf{1} \wedge M$.
5. $\mathbf{0} = M \wedge \overline{M}$.
6. $\mathbf{0} = \mathbf{0} \wedge M$.
7. $\mathbf{1} = M + \overline{M}$.
8. $\mathbf{1} = \mathbf{1} + M$.

Proposition 2.30. *For any $N_1, N_2 \in M(l \times m)$ and $M_1, M_2, M_3 \in M(m \times n)$,*

1. $M = {}^t({}^t(M))$.
2. ${}^t(M_1 + M_2) = {}^t M_1 + {}^t M_2$.
3. ${}^t(M_1 M_2) = {}^t M_2 {}^t M_1$.
4. $M_1 + M_2 = M_2 + M_1$.
5. $M_1 \wedge M_2 = M_2 \wedge M_1$.
6. $(M_1 + M_2) \wedge M_3 = (M_1 \wedge M_3) + (M_2 \wedge M_3)$.

[3]Dimensions of those matrices depend on the context.

7. $N_1 \wedge (M_1 + M_2) = (N_1 \wedge M_1) + (N_1 \wedge M_2)$.
8. $(N_1 + N_2) \wedge M_1 = (N_1 \wedge M_1) + (N_2 \wedge M_1)$.
9. $\overline{M_1 \wedge M_2} = \overline{M_1} + \overline{M_2}$.
10. $\overline{M_1 + M_2} = \overline{M_1} \wedge \overline{M_2}$.

One can easily notice that the last two items of the above proposition are the well-known fact of De Morgan's law. These facts will be used in Section 3.2.2 to show some propositions of frame properties in matrix representation. For a more general introduction to Boolean matrix theory, see [19].

Chapter 3

Linear Algebraic Semantics for Modal Logic

In this chapter, we present linear algebraic reformulation of Kripke semantics. In Section 3.1, we recall the ordinary model-theoretic approach to Kripke semantics of modal logic. Next, in Section 3.2, we present a matrix representation of Kripke semantics and its relevant properties, and also connect our argument to the concept of quantification in first-order logic.

3.1 The Model-theoretic Approach to Kripke Semantics

In order to explain the idea of linear algebraic approach to Kripke semantics of modal logic shortly, we will stick with mono-modal language in this chapter (see also Section 2.1.1). As for syntax, we use a finite set PROP of *propositional variables* and a set $\mathsf{MOD} = \{ \Diamond \}$ of a *modal operator* $\Diamond$. Then, we define a set $\mathsf{Form}_{\mathbf{ML}}$ of *formulas* as follows:

$$\mathsf{Form}_{\mathbf{ML}} \ni A ::= p \mid \neg A \mid (A \vee A) \mid \Diamond A$$

where $p \in \mathsf{PROP}$ and $\Diamond \in \mathsf{MOD}$. For simplicity, we omit $\Diamond \in \mathsf{MOD}$ till the end of this chapter. As for semantics, we use the simple notation (W, R) for any *Kripke frame* and (W, R, V) for any *Kripke models*. Hence, definitions of the *satisfaction relation* $\mathfrak{M}, w \models A$ and the *validity* are also provided with respect to the above frames and models (cf. Section 2.1.1).

In order to compare calculations of the ordinary model-theoretic approach with that of our linear algebraic approach later, let us recall the model-theoretic approach with the following example which demonstrates the ordinary way to calculate the truth value of a formula.

Example 3.1. *Recall Figure 1.1 of Section 1.1.1, i.e., we define the model $\mathfrak{M}$ by:*

$$\begin{array}{rcl} W &=& \{ w_1, w_2, w_3 \}, \\ R &=& \{(w_1, w_1), (w_1, w_2), (w_1, w_3), (w_2, w_2), (w_3, w_3)\}, \\ V(p) &=& \{ w_2 \}. \end{array}$$

By definition, it is clear that $\Diamond p$ is true at w_1 and w_2, i.e., $\mathfrak{M}, w_1 \models \Diamond p$ and $\mathfrak{M}, w_2 \models \Diamond p$, respectively. The proof of $\mathfrak{M}, w_1 \models \Diamond p$ by the model-theoretic approach is proceed as follows. Let us rewrite our goal $\mathfrak{M}, w_1 \models \Diamond p$ by definition as:

$$\textit{For some } v \in W(w_1 R v \textit{ and } \mathfrak{M}, v \models p).$$

By the clause for propositional variable, this is equivalent to:

$$\textit{For some } v \in W(w_1 R v \textit{ and } v \in V(p)).$$

By definition, $V(p) = \{ w_2 \}$. In order to obtain our goal, it suffices to know if $(w_1, w_2) \in R$. Since this trivially holds, we conclude $\mathfrak{M}, w_1 \models p$, as required.

In this example, we used the notion of the existential quantification in our proof. Using our linear algebraic approach, we can show $\mathfrak{M}, w_1 \models \Diamond p$ without such a notion. In other words, the above model-theoretic proof can be represented by a simple calculation over Boolean matrices. We will see the details in Example 3.3 (Section 3.2.1).

3.2 Linear Algebraic Reformulation of Kripke Semantics

3.2.1 Kripke Semantics in Matrices

In this section, we establish a connection between Kripke semantics and its matrix representation with the help of Fitting's idea [8]. Regarding a possible world as a vertex and a tuple (v, u) in an accessibility relation as a directed edge, a frame (W, R) forms a directed graph. If the set of possible worlds is finite, the graph can be represented by a finite adjacency matrix [1] with boolean values, i.e., *Boolean* matrix. In order to focus our discussion on such matrices, we use the following convention.

Convention 3.2. *In what follows in this chapter, we restrict our attention to the finite Kripke models.*

Informally, Fitting's idea of reformulation of Kripke semantics can be summarized as follows: An accessibility relation (or frame) forms a directed graph and can be represented by a Boolean matrix. A valuation of a proposition (or a truth set of formula) can also be represented by a Boolean (column) vector. Then, propositional connectives correspond to Boolean operations over Boolean vectors, and $\Diamond$ operator corresponds to the multiplication of a Boolean matrix and a vector.

Example 3.3. *Recall a Kripke model $\mathfrak{M} = (W, R, V)$ in Example 3.1 (see also Figure 3.1). A Boolean vector which represents the truth set of $\Diamond p$ can be obtained by a multiplication of the matrix corresponding to R and the column vector corresponding to $V(p)$:*

$$\begin{bmatrix} 1 & 1 & 1 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix}.$$

[1] Let $V = \{ v_1, \dots, v_n \}$ and $E \subseteq V \times V$, we can form a finite directed graph (V, E). Then the adjacency matrix of E is a $n \times n$ square matrix such that its component $M(i, j) = 1$ if there is an edge from vertex i to vertex j, and 0 otherwise.

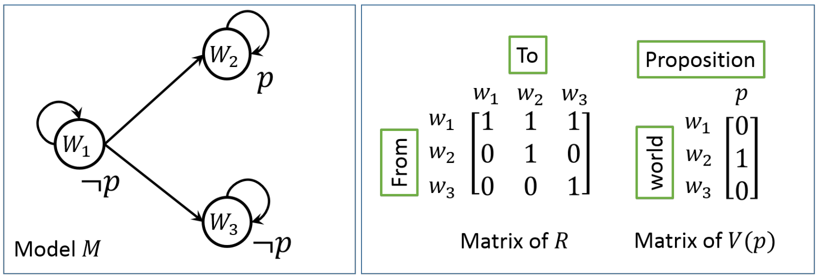


Figure 3.1: Kripke Model and its Boolean Matrix Representation

The resultant vector exactly corresponds to the truth set $[\![\Diamond p]\!]_{\mathfrak{M}} = \{\, w_1, w_2 \,\}$ *in Example 3.1.*

We also emphasize that the calculation of the semantics can be regarded as an extension of truth-table calculation. A truth value of $\Diamond p$ *at* w_1 *is computed by a multiplication of the row vector corresponding to* w_1 *row of the square matrix of* R *and the column vector of* $V(p)$*:*

$$\begin{bmatrix} 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} = 1.$$

The resultant value also corresponds to the result of $\mathfrak{M}, w_1 \models \Diamond p$ *in Example 3.1.*

Now, let us introduce our linear algebraic reformulation of Kripke semantics in full detail. Let $\mathfrak{F} = (W, R)$ be a (finite) Kripke frame and suppose that the cardinality of W is m and $W = \{\, w_1, w_2, \ldots, w_m \,\}$. A matrix representation of an accessibility relation $R^M \in M(m \times m)$ is defined by

$$R^M(i,j) = \begin{cases} 1 & \text{if } (w_i, w_j) \in R, \\ 0 & \text{if } (w_i, w_j) \notin R. \end{cases}$$

Intuitively, a row of the matrix means 'from' world and a column means 'to' world. In order to obtain a matrix representation of Kripke model, it suffices to consider a valuation function in terms of Boolean matrices. Given a Kripke model $\mathfrak{M} = (\mathfrak{F}, V)$ and an atomic proposition $p \in \mathsf{Prop}$, a matrix representation of $V(p)$ is defined to be a column vector $V(p)^M \in M(m \times 1)$ such that

$$V(p)^M(i) = \begin{cases} 1 & \text{if } w_i \in V(p), \\ 0 & \text{if } w_i \notin V(p). \end{cases}$$

The semantic clauses of each formula A can be defined by the computation over the column vector(s) $\|A\|_{\mathfrak{M}} \in M(m \times 1)$ inductively as follows:

$$\begin{array}{lcl} \|p\|_{\mathfrak{M}} & := & (V(p))^M, \\ \|\neg A\|_{\mathfrak{M}} & := & \overline{\|A\|_{\mathfrak{M}}}, \\ \|A \vee A\|_{\mathfrak{M}} & := & \|A\|_{\mathfrak{M}} + \|A\|_{\mathfrak{M}}, \\ \|\Diamond A\|_{\mathfrak{M}} & := & R^M \|A\|_{\mathfrak{M}}, \end{array}$$

where $p \in \mathsf{Prop}$. For simplicity, we drop the subscript '$\mathfrak{M}$' from $\|A\|_{\mathfrak{M}}$ if the underlying model is clear from the context. We say that A is valid on $\mathfrak{M}$ if $\|A\|_{\mathfrak{M}} = \mathbf{1}$. Note that we may extend our syntax and semantics to a multi-modal language. Let G be a finite set of indices. For syntax, we use $\Diamond_a$ (Box_a) operator in multi-modal language instead of $\Diamond$ ($\Box$) operator and the other operators are the same as mono-modal language. For semantics, an accessibility relation R is replaced by $(R_a)_{a \in \mathsf{G}}$, where $R_a \subseteq W \times W$, and their matrix representation becomes R_a^M. Therefore, $\|\Diamond_a A\| := R_a^M \|A\|$. In order to focus our attention on elementary properties of linear algebraic reformulation of Kripke semantics, we will not treat the multi-modal extension in this chapter. Instead, in Chapter 4, we will explain linear algebraic approach to dynamic logic of relation changers that is based on multi-modal language, i.e., propositional dynamic logic without iteration operator.

Proposition 3.4. *Given any finite Kripke model $\mathfrak{M}$ and any formula A of $\mathsf{Form}_{\mathbf{ML}}$,*

$$(\llbracket A \rrbracket_{\mathfrak{M}})^M = \|A\|_{\mathfrak{M}}.$$

Example 3.5. *Let R^M be a 2×2 matrix, $p \in \mathsf{Prop}$ and $V(p)^M$ be a 2×1 matrix. Let us write*

$$R^M := \begin{bmatrix} r_{11} & r_{12} \\ r_{21} & r_{22} \end{bmatrix} \text{ and } V(p)^M := \begin{bmatrix} x \\ y \end{bmatrix}.$$

Then,

$$\begin{aligned} \|\Box p\| &= \|\neg\Diamond\neg p\| &&= \overline{\|\Diamond\neg p\|} \\ &= \overline{R^M\|\neg p\|} &&= \overline{R^M \overline{V(p)^M}} \\ &= \overline{\begin{bmatrix} r_{11} & r_{12} \\ r_{21} & r_{22} \end{bmatrix} \overline{\begin{bmatrix} x \\ y \end{bmatrix}}} &&= \overline{\begin{bmatrix} r_{11} & r_{12} \\ r_{21} & r_{22} \end{bmatrix} \begin{bmatrix} \overline{x} \\ \overline{y} \end{bmatrix}} \\ &= \begin{bmatrix} \overline{r_{11}\overline{x} + r_{12}\overline{y}} \\ \overline{r_{21}\overline{x} + r_{22}\overline{y}} \end{bmatrix} &&= \begin{bmatrix} (\overline{r_{11}} + x) \cdot (\overline{r_{12}} + y) \\ (\overline{r_{21}} + x) \cdot (\overline{r_{22}} + y) \end{bmatrix}. \end{aligned}$$

Thus far, we have explained our linear algebraic reformulation of Kripke semantics. From Examples 3.3 and 3.5, we can observe that calculations of the truth set of a formula are based on truth-table calculation of propositional logic. Indeed, if we focus on propositional connectives and restrict the cardinality of the domain to 1, a matrix calculation of the truth set of a formula is essentially the same as the truth-table calculation of the formula. In this sense, we may regard matrix calculations of the truth set of a formula as an extended version of truth-table calculation of propositional logic. In the next section, we will explain that another type of an extended truth-table calculation, namely, the verification of frame properties in terms of Boolean matrices.

3.2.2 Modal Axioms in Matrices

In order to discuss various frame properties, we now explain that relational union and composition can be defined by matrix addition and multiplication as follows: given two binary relations $R, S \subseteq W \times W$,

$$(R \cup S)^M = R^M + S^M, \quad (R \circ S)^M = R^M S^M$$

where $R \circ S = \{\, (w, v) \mid (w, u) \in R \text{ and } (u, v) \in S \text{ for some } u \in W \,\}$. From an educational perspective, the reader may wonder if we should teach relation algebra after introducing our linear algebraic approach to modal logic since these operations are originally from Tarski's relation algebra [36]. However, this is not the case. Even if our target students do not have prior knowledge of relational composition and union, we can introduce these operations just as the corresponding operations to matrix addition and multiplication. Therefore, we may even introduce the notions from Tarski's relation algebra based on Boolean matrices.

In addition to the above correspondences, the following equivalences will be helpful in proving correspondence between modal formulas and their matrix representations (e.g., Proposition 3.11 in Section 3.2.2).

Proposition 3.6. *Given any $R, S \subseteq W \times W$,*

$$R \subseteq S \text{ iff } S = R \cup S \text{ iff } S^M = R^M + S^M.$$

Now we can reformulate well-known frame properties in terms of Boolean matrices.

Proposition 3.7. *Every frame property listed in Table 3.1 can be reformulated in terms of Boolean matrix with elementary matrix calculations as in the Table 3.1 where* **1** *means a column vector of all 1s.*

Table 3.1: Frame Properties and Their Matrix Representations

Name	Frame Condition	Formula	Matrix Reformulation
Reflexive	$\forall w(wRw)$	T $\Box p \to p$	$R = R + \boldsymbol{E}$
Symmetric	$\forall w, v(wRv \text{ implies } vRw)$	B $p \to \Box\Diamond p$	$R = {}^tR$ (or $R = {}^tR + R$)
Transitive	$\forall w, v, u(wRv \& vRu \text{ imply } wRu)$	4 $\Box p \to \Box\Box p$	$R = RR + R$
Serial	$\forall w \exists v(wRv)$	D $\Box p \to \Diamond p$	$R^tR = R^tR + E$ (or $\mathbf{1} = R\mathbf{1}$)[2]
Euclidean	$\forall w, v, u(wRv \& wRu \text{ imply } vRu)$	5 $\Diamond p \to \Box\Diamond p$	$R = {}^tRR + R$

We can verify the five frame properties of a given frame in Table 3.1 in terms of Boolean matrices.

Example 3.8. *Recall a matrix representation of an accessibility relation R in Example 3.1. For simplicity, we regard R as a square matrix of the relation. We can verify whether R satisfies certain frame properties listed in Table 3.1 by the computation over matrices. For example, let us check whether R satisfies transitivity. By $R = RR + R$, i.e.,*

$$\begin{bmatrix} 1 & 1 & 1 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 1 & 1 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} + \begin{bmatrix} 1 & 1 & 1 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} 1 & 1 & 1 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

[2] What we found is a more general result than [5]. Their seriality is described by $L \subseteq R; L$ where R is an accessibility relation, L denotes the universal relation, and $R; L$ denotes the relational composition of R with L. The corresponding matrix representation of their seriality is $L = RL$ where L and R are unit square matrix of a relation. If the cardinality of the domain is 2, we can describe their seriality by $[\mathbf{11}] = R\,[\mathbf{11}] = [R\mathbf{1}, R\mathbf{1}]$.

Hence, we may conclude that R satisfies transitivity. In a similar manner, we can also check whether R satisfies the other properties. Since $R = R + \boldsymbol{E}$ and $\mathbf{1} = R\mathbf{1}$, R satisfies reflexivity and seriality, respectively. However, by $R \neq {}^tR$, $E \neq {}^tRR{+}E$ and $R \neq {}^tRR{+}R$, this R does not satisfy symmetricity, and Euclideanness, respectively. Finally, we may conclude that the model satisfies reflexivity, seriality, and transitivity.

Note that the verification of frame properties in Example 3.8 can also be regarded as an extended truth-table calculation. This is because each verification of frame property in Example 3.8 is based on Boolean matrix calculation.

Next, we will establish well-known implications among frame properties in terms of Boolean matrices. In addition, we also show an ordinary proof for comparison.

Proposition 3.9. *Reflexivity and Euclideanness jointly imply symmetry, i.e., $R = R + \boldsymbol{E}$ and $R = {}^tRR + R$ jointly imply $R = {}^tR$.*

Proof. Firstly, we observe that if R is reflexive, then the transposition ${}^tR$ is also reflexive, i.e., ${}^tR = {}^tR + E$. Secondly, we rewrite the equation of reflexivity, as follows:

$$\begin{array}{rcll} R & = & R + \boldsymbol{E} & \text{(by reflexivity)} \\ & = & ({}^tRR + R) + \boldsymbol{E} & \text{(by Euclideanness)} \\ & = & ({}^tR + \boldsymbol{E})R + \boldsymbol{E} & \\ & = & {}^tRR + \boldsymbol{E} & \text{(by reflexivity of } {}^tR). \end{array}$$

Afterward, we get ${}^tR = {}^tRR + \boldsymbol{E}$ by transposing both sides. Since both R and ${}^tR$ are equal to ${}^tRR + \boldsymbol{E}$, we finally obtain $R = {}^tR$. □

For comparison, we show an ordinary proof with quantifiers as follows. We show that for any $w, v \in W$, wRv implies vRw. Fix any w, v such that wRv. By reflexivity, wRw. By Euclideanness, we obtain vRw from wRv and wRw, as desired.

Proposition 3.10. *Reflexivity and Euclideanness jointly imply transitivity, i.e., $R = R + \boldsymbol{E}$ and $R = {}^tRR + R$ jointly imply $R = RR + R$.*

Proof. We rewrite the equation of Euclideanness as follows.

$$\begin{array}{rcll} R & = & {}^tRR + R & \text{(by Euclideanness)} \\ & = & RR + R & \text{(by Proposition 3.9).} \end{array}$$

□

We also show an ordinary proof with quantifiers as follows. We show that for any $w, v, u \in W$, wRv and vRu imply wRu. Fix any w, v such that wRv and vRu. By symmetry (Proposition 3.9), vRw. By Euclideanness, we obtain wRu from vRw and vRu.

In the above ordinary proofs, we had to select the appropriate variables for every application of the conditions of the frame properties. The selection of variables might sometimes be a cause of an error in the proof. On the other hand, we did not need to worry about the selection of variables in the above linear algebraic proofs.

In order to establish a relationship between modal axioms and frame properties, we follow the idea of Lemmon-Scott axioms or Geach axioms [22, 7]. Namely, we show that there are at least two types of correspondence between modal axioms and their corresponding matrix representations in the Table 3.1. For simplicity, we are omitting superscript M and regard R as a square matrix.

Proposition 3.11. *Let $n, m, l, k \in \mathbb{N}$ and $p \in \mathsf{Prop}$. For all frames $\mathfrak{F} = (W, R)$,*

$$\mathfrak{F} \models \Diamond^k \Box^l p \to \Box^m \Diamond^n p \text{ iff } ({}^tR)^m R^k + R^n ({}^tR)^l = R^n ({}^tR)^l.$$

Proof. Here R^{-1} denotes the inverse relation of R. We observe that ${}^t(R^M) = (R^{-1})^M$. Fix any frame $\mathfrak{F} = (W, R)$.

$$\begin{array}{ll} & \mathfrak{F} \models \Diamond^k \Box^l p \to \Box^m \Diamond^n p, \\ \text{iff} & \mathfrak{F} \models (\Diamond^{-1})^m \Diamond^k \Box^l p \to \Diamond^n p, \\ \text{iff} & \mathfrak{F} \models (\Diamond^{-1})^m \Diamond^k p \to \Diamond^n (\Diamond^{-1})^l p, \\ \text{iff} & (R^{-1})^m \circ R^k \subseteq R^n \circ (R^{-1})^l, \\ \text{iff} & (R^{-1})^m \circ R^k \cup R^n \circ (R^{-1})^l = R^n \circ (R^{-1})^l. \end{array}$$

This is equivalent to $({}^tR)^m R^k + R^n ({}^tR)^l = R^n ({}^tR)^l$. □

Using the above proposition, we can obtain matrix representations of reflexivity, symmetricity, transitivity, seriality, and Euclideanness in Table 3.1. In addition, we can obtain another matrix representation of seriality, i.e., $\mathbf{1} = R\mathbf{1}$, by the following proposition.

Proposition 3.12. *Let $m \in \mathbb{N}$, $p \in \mathsf{Prop}$ and $\mathbf{1}$ be a vector of all 1s. For all frames $\mathfrak{F} = (W, R)$,*

$$\mathfrak{F} \models \Box^m p \to \Diamond^m p \text{ iff } R^m \mathbf{1} = \mathbf{1}.$$

Proof. Fix any frame $\mathfrak{F} = (W, R)$.

$$\begin{array}{lll} \mathfrak{F} \models \Box^m p \to \Diamond^m p & \text{iff} & \mathfrak{F} \models \Diamond^m \neg p \vee \Diamond^m p, \\ & \text{iff} & \mathfrak{F} \models \Diamond^m (\neg p \vee p), \\ & \text{iff} & \mathfrak{F} \models \Diamond^m \top \leftrightarrow \top, \\ & \text{iff} & R^m \mathbf{1} = \mathbf{1}. \end{array}$$

□

3.2.3 Quantifications in Matrices

So far we have explained the matrix reformulation of Kripke semantics in modal logic. Now we begin to extend this approach to capture the behaviors of a universal quantifier $\forall$ and an existential quantifier $\exists$ in first-order logic.

Let us consider the case of the universal (or full) relation, i.e., $R = W \times W$. Then, the semantic clauses of $\Box$ and $\Diamond$ becomes:

$$\begin{array}{lll} \mathfrak{M}, w \models \Box A & \text{iff} & \forall v \in W (wRv \text{ implies } \mathfrak{M}, v \models A), \\ \mathfrak{M}, w \models \Diamond A & \text{iff} & \exists v \in W (wRv \text{ and } \mathfrak{M}, v \models A). \end{array}$$

Since R is the universal relation, wRv trivially holds. This implies that these clauses are not restricted by the accessibility relation R. Namely, the clauses can be regarded as:

$$\begin{array}{lll} \mathfrak{M}, w \models \Box A & \text{iff} & \forall v \in W (\mathfrak{M}, v \models A), \\ \mathfrak{M}, w \models \Diamond A & \text{iff} & \exists v \in W (\mathfrak{M}, v \models A). \end{array}$$

In this sense, we may regard the semantic clauses for $\Box$ and $\Diamond$ of modal logic as the ones for $\forall$ and $\exists$ of first-order logic, respectively.

We can establish a similar argument in terms of Boolean matrices. In the sense of the matrices, the universal relation R becomes the complete square matrix $\mathbf{1}$. As a result, computations of $\|\Diamond p\|$ and $\|\Box p\|$ come to reflect the above argument.

Example 3.13. *Let $W = \{ w_1, w_2, w_3 \}$, R be the universal relation and $V(p) = \{ w_2 \}$. Since there exists a world w_2 such that p holds, $\Diamond p$ also holds at every world, i.e.,*

$$\|\Diamond p\| := R^M \|p\| = R^M V(p)^M = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}.$$

However, since p does not hold at w_1 and w_3, $\Box p$ does not hold at every world, i.e.,

$$\|\Box p\| := \overline{R^M \overline{\|p\|}} = \overline{R^M \overline{V(p)^M}} = \overline{\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix}} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}.$$

Now, let us visualize the distinction between $\exists\forall$ and $\forall\exists$ of first-order logic by matrix representation. Let us consider the situation where $\exists y \forall x R(x, y)$, that is, there is some world y from which all the other worlds are accessible. Then, it means that the y-column is filled with 1s. This observation implies that the property of $\exists y \forall x R(x, y)$ is expressed in terms of Boolean matrix as $({}^t\overline{R})\mathbf{1} \neq \mathbf{1}$. In the similar way, in case $\forall x \exists y R(x, y)$, that is, for each row there must be at least one 1 (see Table 3.2). Thus, the property $\forall x \exists y R(x, y)$ of seriality is expressed in terms of Boolean matrix as: $R\mathbf{1} = \mathbf{1}$.

Table 3.2: Example of Nested Quantifications in terms of Matrices (in 3×3).

$\exists y \forall x R(x, y)$	$\forall x \exists y R(x, y)$
$\begin{bmatrix} 1 & 0 & 1 \\ 1 & 0 & 1 \\ 1 & 0 & 1 \end{bmatrix}$	$\begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 1 \end{bmatrix}$

Then, we also establish "$\exists\forall$ implies $\forall\exists$" in term of matrices.

Proposition 3.14. $({}^t\overline{R})\mathbf{1} \neq \mathbf{1}$ *implies* $R\mathbf{1} = \mathbf{1}$.

Proof. Let R be an $n \times n$ matrix. Let us write

$$R := \begin{bmatrix} r_{11} & \cdots & r_{1n} \\ \vdots & \ddots & \vdots \\ r_{n1} & \cdots & r_{nn} \end{bmatrix}.$$

We show the contrapositive implication and so assume $R\mathbf{1} \neq \mathbf{1}$. Now, the goal is to show $({}^t\overline{R})\mathbf{1} = \mathbf{1}$. The assumption implies that $[r_{i1} \ \cdots \ r_{in}] = {}^t\mathbf{0}$ for some $1 \leq i \leq n$. Fix such i. Then, $[\overline{r_{i1}} \ \cdots \ \overline{r_{in}}] = \overline{[r_{i1} \ \cdots \ r_{in}]} = {}^t\mathbf{1}$. Since

$${}^t\overline{R} := \begin{bmatrix} \overline{r_{11}} & \cdots & \overline{r_{n1}} \\ \vdots & \ddots & \vdots \\ \overline{r_{1n}} & \cdots & \overline{r_{nn}} \end{bmatrix}$$

$({}^t\overline{R})\mathbf{1} = \mathbf{1}$ holds by $[\overline{r_{i1}} \ \cdots \ \overline{r_{in}}] = {}^t\mathbf{1}$. $\square$

Chapter 4

Computational Tools for Dynamic Logic of Relation Changers

In this chapter, we provide cut-free labelled sequent calculus for dynamic logic of relation changers (**DLRC**). In Section 4.1, we provide our syntax, semantics and Hilbert-style axiomatization HDLRC for **DLRC**. Afterward, in Section 4.2, we present linear algebraic reformulation of Kripke semantics for **DLRC**. We show the soundness of HDLRC for Kripke semantics in terms of Boolean matrices (Section 4.2.2). In Section 4.3, we present our sequent calculus GDLRC from a labelled formalism for that. In this section, we establish that all theorems of HDLRC are derivable in GDLRC (Theorem 4.11) and our sequent calculus enjoys the cut elimination theorem (Theorem 4.19). Finally, in Section 4.3.4, we establish the soundness of GDLRC for Kripke semantics (Theorem 4.23) and conclude our sequent calculus is equipollent with the Hilbert-style axiomatization HDLRC (Corollary 4.24).

4.1 Dynamic Logic of Relation Changers

4.1.1 Syntax and Kripke Semantics

Our syntax of the language $\mathcal{L}_{\mathbf{DLRC}}$ of dynamic logic of relation changer (**DLRC**) is based on that of the star-free fragment of propositional dynamic logic ($\mathbf{PDL}^{-}$, see also Section 2.2). Besides the propositional connectives, modal operators for programs and and the program constructors of $\mathbf{PDL}^{-}$, the language $\mathcal{L}_{\mathbf{DLRC}}$ contains the next operators for relation changers: $[\mathsf{r}]$ and $:=$. Given a countably infinite set Atom of propositional variables and a finite set AP of atomic programs, we define the set $\mathsf{Form}_{\mathbf{DLRC}}$ of formulas, the set PR of programs, and the set RC of relation changers by simultaneous induction as follows:

$$\begin{array}{rcl} \mathsf{Form}_{\mathbf{DLRC}} \ni A & ::= & p \mid \neg A \mid A \to A \mid [\alpha]A \mid [\mathsf{r}]A \\ \mathsf{PR} \ni \alpha & ::= & a \mid \alpha \cup \alpha \mid \alpha ; \alpha \mid ?A \\ \mathsf{RC} \ni \mathsf{r} & ::= & (a := \alpha)_{a \in \mathsf{AP}} \end{array}$$

where $p \in \mathsf{Prop}$ and $a \in \mathsf{AP}$.

As we noted in Section 1.1.2, we regard each agent's preference as an atomic program. Therefore, we use the same reading of a formula $[\alpha]A$ and programs as in Section 2.2 (see also Table 2.5). Note that a relation changer r can be regarded as a function from AP to PR. Hence, when $\mathsf{r} = (a := \alpha_a)_{a \in \mathsf{AP}}$, we may use the notation $\mathsf{r}(a)$ to mean α_a.

Then a formula $[\mathsf{r}]A$ reads 'After changing an accessibility relation for each programs a by α_a, A holds.' For example, we may consider the following relation changer: given a set $\mathsf{AP} = \{a, b, c\}$ of atomic programs, we set $\mathsf{r} = (a := a, b := c, c := b)$ and so $\mathsf{r}(a) = a, \mathsf{r}(b) = c$ and $\mathsf{r}(c) = b$. The intended meaning of this $[\mathsf{r}]$ is that an accessibility relation of the program a is not changed, accessibility relations of b and c is overwritten (or exchanged) by relations of the (original) programs c and b, respectively.

Let us define Kripke semantics with our syntax. As in $\mathbf{PDL}^-$ (cf. Section 2.2), a *model* $\mathfrak{M}$ is a tuple $(W, (R_a)_{a\in\mathsf{AP}}, V)$ where W is a non-empty set of *possible worlds* (or states), $R_a \subseteq W \times W$ is an *accessibility relation*, and $V : \mathsf{Prop} \to \mathcal{P}(W)$ is a *valuation function*.

Given any model $\mathfrak{M} = (W, (R_a)_{a\in\mathsf{AP}}, V)$ and any possible world $w \in W$, the *satisfaction relation* $\mathfrak{M}, w \models A$ is defined by simultaneous induction as follows:

$$\begin{array}{lll}
\mathfrak{M}, w \models p & \text{iff} & w \in V(p), \\
\mathfrak{M}, w \models \neg A & \text{iff} & \mathfrak{M}, w \not\models A, \\
\mathfrak{M}, w \models A \to B & \text{iff} & \mathfrak{M}, w \models A \text{ implies } \mathfrak{M}, w \models B, \\
\mathfrak{M}, w \models [\alpha]A & \text{iff} & \mathfrak{M}, v \models A \text{ for all } v \text{ with } wR_\alpha v, \\
wR_{\alpha\cup\beta}v & \text{iff} & wR_\alpha v \text{ or } wR_\beta v, \\
wR_{\alpha;\beta}v & \text{iff} & wR_\alpha u \text{ and } uR_\beta v \text{ for some } u \in W, \\
wR_{?A}v & \text{iff} & w = v \text{ and } \mathfrak{M}, w \models A, \\
\mathfrak{M}, w \models [\mathsf{r}]A & \text{iff} & \mathfrak{M}^{\mathsf{r}}, w \models A,
\end{array}$$

where $\mathfrak{M}^{\mathsf{r}} = (W, (R^{\mathsf{r}}_a)_{a\in\mathsf{AP}}, V^{\mathsf{r}})$ and R^{r}_a and V^{r} are defined as:

- $R^{\mathsf{r}}_a := R_{\mathsf{r}(a)} = R_{\alpha_a}$when $\mathsf{r} = (a := \alpha_a)_{a\in\mathsf{AP}}$,
- $w \in V^{\mathsf{r}}(p)$ iff $w \in V(p)$.

As we described in our syntax, R^{r}_a is changed (or redefined) by that of a (new) program $\mathsf{r}(a)$ $(= \alpha_a)$. On the other hand, $V^{\mathsf{r}}(p)$ is not changed by a relation changer r since $V^{\mathsf{r}}(p)$ is defined by the original $V(p)$. The *truth set* $[\![A]\!]_{\mathfrak{M}}$ is defined by $[\![A]\!]_{\mathfrak{M}} = \{ w \in W \mid \mathfrak{M}, w \models A \}$. By definition, we can derive the same truth sets for $\neg$, $\to$, $[\alpha]$, $\cup$, ;, and ? as in $\mathbf{PDL}^-$ (see also Sections 2.2). Then, the truth set for a relation changer is derived by:

$$[\![[\mathsf{r}]A]\!]_{\mathfrak{M}} := [\![A]\!]_{\mathfrak{M}^{\mathsf{r}}}$$

where $\mathfrak{M}^{\mathsf{r}}$ is the model $\mathfrak{M}$ changed by the relation changer r that is defined in the above. The notion of the *validity* is provided over the above models and frames, as in the ordinary modal logic (cf. Section 2.1.1).

Example 4.1. *Here we show two examples of relation changers which were originally presented in [39, 25].*[1] *The first one is a relation changer for the link-cutting public update operator* $[\dagger A]$, *as we described in Section 1.1.2. Each accessibility relation of the updated model* $\mathfrak{M}^{\dagger A}$ *is defined as follows: for all* $a \in \mathsf{AP}$,

$$R^{\dagger A}_a := \{ (u, v) \in R_a \mid \mathfrak{M}, u \models A \text{ iff } \mathfrak{M}, v \models A \}$$

[1]The original definitions of atomic programs for these operators are defined as agent's preferences, and if we follow the original definition, we have to introduce additional frame properties to our logic. But, in order to make our story simple, we regard atomic programs semantically as ordinary binary relations. Thus, we do not assume any frame property since an accessibility relation might be changed arbitrarily by a dynamic operator.

The corresponding relation changer can be described as follows:

$$\mathsf{r}_{\dagger A} = (a := (?A; a; ?A) \cup (?\neg A; a; ?\neg A))_{a \in \mathsf{AP}}$$

The second one is a relation changer for a suggestion operator $[\sharp A]$*. Intuitively, the suggestion operator removes all links from* A*-worlds to non-*A *worlds from the original model* $\mathfrak{M}$*. Each accessibility relation of the updated model* $\mathfrak{M}^{\sharp A}$ *is defined as follows: for all* $a \in \mathsf{AP}$*,*

$$R_a^{\sharp A} := R_a \setminus \{ (u, v) \mid \mathfrak{M}, u \models A \text{ and } \mathfrak{M}, v \models \neg A \}.$$

The corresponding relation changer is given as follows:

$$\mathsf{r}_{\sharp A} = (a := (?\neg A; a) \cup (a; ?A))_{a \in \mathsf{AP}}$$

4.1.2 Hilbert-style Axiomatization HDLRC

In Table 4.1, we explicitly provide sound and complete Hilbert-style axiomatization **HDLRC** of dynamic logic of relation changers from [39, 25]. We note that the axioms from (**Taut**) to ([?]) and the rules (**MP**) and ($\mathbf{Nec}_{[\alpha]}$) in this table come from Hilbert-style axiomatization $\mathsf{H}\mathbf{PDL}^-$. Therefore, the axioms from ($[\mathsf{r}]p$) to ($[\mathsf{r}][?]$) and the rule ($\mathbf{Nec}_{[\mathsf{r}]}$) in this table are the additional items to $\mathsf{H}\mathbf{PDL}^-$. A *derivation* in **HDLRC** is a sequence of formulas which consist of an instance of an axiom or the result of applying an inference rule to formulas that occur earlier. If a formula A occurs in the last of a derivation, we say A is *derivable* in **HDLRC** (or is a theorem of that) and denote $\vdash_{\mathbf{HDLRC}} A$. The height of a derivation of A is defined as the length of a sequence of the derivation of that.

Definition 4.2. *The translation* $t : \mathsf{Form}_{\mathbf{DLRC}} \to \mathsf{Form}_{\mathbf{PDL}^-}$ *is defined by:*

$$\begin{array}{lcl}
t(p) & = & p, \\
t(\neg A) & = & \neg t(A), \\
t(A \to B) & = & t(A) \to t(B), \\
t([a]A) & = & [a]t(A), \\
t([\alpha \cup \beta]A) & = & t([\alpha]A) \wedge t([\beta]A), \\
t([\alpha; \beta]A) & = & t([\alpha][\beta]A), \\
t([?B]A) & = & t(B) \to t(A), \\
t([\mathsf{r}]p) & = & p, \\
t([\mathsf{r}]\neg A) & = & \neg t([\mathsf{r}]A), \\
t([\mathsf{r}](A \to B)) & = & t([\mathsf{r}]A) \to t([\mathsf{r}]B), \\
t([\mathsf{r}][a]A) & = & [\mathsf{r}(a)]t([\mathsf{r}]A), \\
t([\mathsf{r}][\alpha \cup \beta]A) & = & t([\mathsf{r}][\alpha]A) \wedge t([\mathsf{r}][\beta]A), \\
t([\mathsf{r}][\alpha; \beta]A) & = & t([\mathsf{r}][\alpha][\beta]A), \\
t([\mathsf{r}][?B]A) & = & t([\mathsf{r}]B) \to t([\mathsf{r}]A), \\
t([\mathsf{r}][\mathsf{r}']A) & = & t([\mathsf{r}]t([\mathsf{r}']A)).
\end{array}$$

Remark that this translation reflects the idea of 'inside-out' strategy. That is, we start rewriting the innermost occurrences of $[\mathsf{r}]$.

Lemma 4.3. *Given any formula* $A \in \mathsf{Form}_{\mathbf{DLRC}}$*,*

$$\vdash_{\mathbf{HDLRC}} A \leftrightarrow t(A).$$

Fact 4.4 ([39, 25]). *Let A be a formula in $\mathsf{Form}_{\mathbf{DLRC}}$ and $\mathbb{M}$ be the class of all models.*

$$\mathbb{M} \models A \text{ iff } \vdash_{\mathbf{HDLRC}} A.$$

Proof. **Soundness part** We show that if $\vdash_{\mathbf{HDLRC}} A$ then $\mathbb{M} \models A$ for all A. One can easily check that each axiom in $\mathsf{\mathbf{HDLRC}}$ is valid in the class $\mathbb{M}$ of all models and each rule in $\mathsf{\mathbf{HDLRC}}$ preserves the validity in $\mathbb{M}$.

Completeness part We show that if $\mathbb{M} \models A$, then $\vdash_{\mathbf{HDLRC}} A$ for all A. We can reduce the completeness of $\mathsf{\mathbf{HDLRC}}$ to that of $\mathsf{\mathbf{HPDL}}^-$ (cf. Theorem 2.25). Fix any formula A and suppose that $\mathbb{M} \models A$. By the soundness part and Lemma 4.3, we get $\mathbb{M} \models A \leftrightarrow t(A)$. Then, by this and our assumption of $\mathbb{M} \models A$, we obtain that $\mathbb{M} \models t(A)$. By this and the completeness of $\mathsf{\mathbf{HPDL}}^-$ (Theorem 2.25), we have that $\vdash_{\mathbf{HPDL}^-} t(A)$. Since $\mathsf{\mathbf{HDLRC}}$ is an axiomatic extension of $\mathsf{\mathbf{HPDL}}^-$, we also have that $\vdash_{\mathbf{HDLRC}} t(A)$. Finally, by this and Lemma 4.3, we obtain $\vdash_{\mathbf{HDLRC}} A$. □

Table 4.1: Hilbert-style Axiomatization $\mathsf{\mathbf{HDLRC}}$

(**Taut**)	All instances of propositional tautologies
($\mathbf{K}_{[\alpha]}$)	$[\alpha](A \to B) \to ([\alpha]A \to [\alpha]B)$
($[\cup]$)	$[\alpha \cup \beta]A \leftrightarrow [\alpha]A \wedge [\beta]A$
($[;]$)	$[\alpha;\beta]A \leftrightarrow [\alpha][\beta]A$
($[?]$)	$[?B]A \leftrightarrow (B \to A)$
($[\mathsf{r}]p$)	$[\mathsf{r}]p \leftrightarrow p$
($[\mathsf{r}]\neg$)	$[\mathsf{r}]\neg A \leftrightarrow \neg[\mathsf{r}]A$
($[\mathsf{r}] \to$)	$[\mathsf{r}](A \to B) \leftrightarrow ([\mathsf{r}]A \to [\mathsf{r}]B)$
($[\mathsf{r}][a]$)	$[\mathsf{r}][a]A \leftrightarrow [\mathsf{r}(a)][\mathsf{r}]A$
($[\mathsf{r}][\cup]$)	$[\mathsf{r}][\alpha \cup \beta]A \leftrightarrow [\mathsf{r}][\alpha]A \wedge [\mathsf{r}][\beta]A$
($[\mathsf{r}][;]$)	$[\mathsf{r}][\alpha;\beta]A \leftrightarrow [\mathsf{r}][\alpha][\beta]A$
($[\mathsf{r}][?]$)	$[\mathsf{r}][?B]A \leftrightarrow [\mathsf{r}](B \to A)$
(**MP**)	From A and $A \to B$, infer B
($\mathbf{Nec}_{[\alpha]}$)	From A, infer $[\alpha]A$
($\mathbf{Nec}_{[\mathsf{r}]}$)	From A, infer $[\mathsf{r}]A$

Proposition 4.5. *Let* $\mathsf{\mathbf{HDLRC}}^-$ *be the Hilbert-style axiomatization defined by all axioms and all inference rules of* $\mathsf{\mathbf{HDLRC}}$ *without axioms* ($[\mathsf{r}][\cup]$), ($[\mathsf{r}][;]$) *and* ($[\mathsf{r}][?]$). *Then, all theorems of* $\mathsf{\mathbf{HDLRC}}$ *are derivable in* $\mathsf{\mathbf{HDLRC}}^-$.

Proof. We show that axiom schemes ($[\mathsf{r}][\cup]$), ($[\mathsf{r}][;]$) and ($[\mathsf{r}][?]$) are derivable in $\mathsf{\mathbf{HDLRC}}$ without them. Here, we only sketch the proof for the case of axiom ($[\mathsf{r}][\cup]$), since the other two cases are shown similarly. First, we can derive the distribution of $[\mathsf{r}]$ over the conjunction $\wedge$, i.e., $[\mathsf{r}](A \wedge B) \leftrightarrow [\mathsf{r}]A \wedge [\mathsf{r}]B$, by axioms ($[\mathsf{r}] \to$) and ($[\mathsf{r}]\neg$). Moreover, we can also derive the distribution of $[\mathsf{r}]$ over the logical equivalence $\leftrightarrow$ by the distribution of

$[\mathsf{r}]$ over $\wedge$ and axiom ($[\mathsf{r}] \to$). Then, by axiom ($[\cup]$) and inference rule ($\mathbf{Nec}_{[\mathsf{r}]}$), we have $[\mathsf{r}]([\alpha \cup \beta] \leftrightarrow [\alpha]A \wedge [\beta]A)$, which implies axiom ($[\mathsf{r}][\cup]$) by the derived distribution of $[\mathsf{r}]$ over $\leftrightarrow$ and $\wedge$. $\square$

4.2 Linear Algebraic Semantics for Dynamic Logic of Relation Changers

4.2.1 Relation Changers in Matrices

As we noted in Section 3.2.1, we can extend our linear algebraic reformulation of Kripke semantics to handle multi-modal language. In this section, we define such an extension for **DLRC** that is based on $\mathbf{PDL}^{-}$.

Let $\mathfrak{M} = (W, (R_a)_{a \in \mathsf{AP}}, V)$ be a (finite) Kripke model where the cardinality of the domain W is m, i.e., $W = \{ w_1, w_2, \dots, w_m \}$. Then, we define matrix representations for an accessibility relation R_a and the valuation $V(p)$ as did in Section 3.2.1. Suppose that $R_a^M \in M(m \times m)$ and $V(p)^M \in M(m \times m)$. We inductively associate with each formula A of $\mathsf{Form}_{\mathbf{DLRC}}$ with a column vector $\|A\|_{\mathfrak{M}} \in M(m \times 1)$ as follows:

$$
\begin{array}{lll}
\|p\|_{\mathfrak{M}} & := & (V(p))^M, \\
\|\neg A\|_{\mathfrak{M}} & := & \overline{\|A\|_{\mathfrak{M}}}, \\
\|A \to B\|_{\mathfrak{M}} & := & \overline{\|A\|_{\mathfrak{M}}} + \|B\|_{\mathfrak{M}}, \\
\|[\alpha]A\|_{\mathfrak{M}} & := & \overline{\|\alpha\|_{\mathfrak{M}}\overline{\|A\|_{\mathfrak{M}}}}, \\
\|a\|_{\mathfrak{M}} & := & R_a^M, \\
\|\alpha \cup \beta\|_{\mathfrak{M}} & := & \|\alpha\|_{\mathfrak{M}} + \|\beta\|_{\mathfrak{M}}, \\
\|\alpha ; \beta\|_{\mathfrak{M}} & := & \|\alpha\|_{\mathfrak{M}}\|\beta\|_{\mathfrak{M}}, \\
\|?A\|_{\mathfrak{M}} & := & \begin{cases} 1 & \text{if } i = j \text{ and } \|A\|_{\mathfrak{M}}(i) = 1, \\ 0 & \text{otherwise}, \end{cases} \\
\|[\mathsf{r}]A\|_{\mathfrak{M}} & := & \|A\|_{\mathfrak{M}^{\mathsf{r}}}.
\end{array}
$$

It follows that:

$$
\begin{array}{lll}
\|A \vee A\|_{\mathfrak{M}} & := & \|A\|_{\mathfrak{M}} + \|A\|_{\mathfrak{M}}, \\
\|A \wedge A\|_{\mathfrak{M}} & := & \|A\|_{\mathfrak{M}} \wedge \|A\|_{\mathfrak{M}}, \\
\|\langle\alpha\rangle A\|_{\mathfrak{M}} & := & \|\alpha\|_{\mathfrak{M}}\|A\|_{\mathfrak{M}}, \\
\|\top\|_{\mathfrak{M}} & := & \mathbf{1}_{\mathfrak{M}}, \\
\|\bot\|_{\mathfrak{M}} & := & \mathbf{0}_{\mathfrak{M}},
\end{array}
$$

where $\mathbf{1}_{\mathfrak{M}}$ (or $\mathbf{0}_{\mathfrak{M}}$) is the column vector $\mathbf{1}$ (or $\mathbf{0}$) under the model $\mathfrak{M}$. As we noted in Section 3.2.1, we may drop the subscript '$\mathfrak{M}$' from $\|A\|_{\mathfrak{M}}$ if the underlying model is clear from the context. We say that A is valid in $\mathfrak{M}$ if $\|A\|_{\mathfrak{M}} = \mathbf{1}_{\mathfrak{M}}$.

Proposition 4.6. *Let $\mathfrak{M}$ be any finite Kripke model.*

1. $(\llbracket A \rrbracket_{\mathfrak{M}})^M = \|A\|_{\mathfrak{M}}$ *for any formula* $A \in \mathsf{Form}_{\mathbf{DLRC}}$.
2. $(\llbracket \alpha \rrbracket)_{\mathfrak{M}}^M = \|\alpha\|_{\mathfrak{M}}$ *for any program* $\alpha \in \mathsf{PR}$.

Proposition 4.7. *Given any finite Kripke model $\mathfrak{M}$ and any formula A,*

$$\|A \leftrightarrow B\|_{\mathfrak{M}} = \mathbf{1}_{\mathfrak{M}} \text{ iff } (\llbracket A \rrbracket_{\mathfrak{M}})^M = (\llbracket B \rrbracket_{\mathfrak{M}})^M \text{ iff } \|A\|_{\mathfrak{M}} = \|B\|_{\mathfrak{M}}.$$

4.2.2 Soundness of HDLRC for Kripke Semantics in Matrices

In this section, we establish the soundness of **DLRC** for Kripke semantics (Fact 4.4) using our linear algebraic approach.

Proof. To show the soundness of **HDLRC** for Kripke semantics, our goal is to show that

(1) all axioms in **HDLRC** are valid in $\mathbb{M}$,

(2) all inference rules in **HDLRC** preserve the validity in $\mathbb{M}$.

We show our goal by linear algebraic reformulation of Kripke semantics of **DLRC**. In what follows, we restrict our attention to the finite domain size first. However, we will see that our proof does not rely on the finiteness of the domain size. Hence, at the end of our proof, we may take the domain size to utmost limit, i.e., infinite domain size.

Case of axiom (**Taut**)
Since it is trivial, we skip this case.

Case of axiom ($\mathbf{K}_{[\alpha]}$)
We show that $\mathbb{M} \models [\alpha](A \to B) \to ([\alpha]A \to [\alpha]B)$ Fix any $\mathfrak{M} \in \mathbb{M}$.
Our goal is to show that:

$$\|[\alpha](A \to B) \to ([\alpha]A \to [\alpha]B)\| = \mathbf{1}.$$

This is shown by:

$$\begin{aligned}
\|[\alpha](A \to B) \to ([\alpha]A \to [\alpha]B)\| &= \overline{\|[\alpha](A \to B)\|} + \|([\alpha]A \to [\alpha]B)\| \\
&= \overline{\|[\alpha](A \to B)\|} + (\overline{\|[\alpha]A\|} + \|[\alpha]B)\|) \\
&= \overline{\overline{\|\alpha\|\overline{\|A \to B\|}}} + (\overline{\overline{\|\alpha\|\overline{\|A\|}}} + \overline{\|\alpha\|\overline{\|B\|}}) \\
&= \|\alpha\|(\overline{\|A \to B\|} + \overline{\|A\|}) + \overline{\|\alpha\|\overline{\|B\|}} \\
&= \|\alpha\|(\overline{\|A \to B\| \wedge \|A\|}) + \overline{\|\alpha\|\overline{\|B\|}} \\
&= \|\alpha\|(\overline{(\overline{\|A\|} + \|B\|) \wedge \|A\|}) + \overline{\|\alpha\|\overline{\|B\|}} \\
&= \|\alpha\|(\overline{(\overline{\|A\|} \wedge \|A\|) + (\|B\| \wedge \|A\|)}) + \overline{\|\alpha\|\overline{\|B\|}} \\
&= \|\alpha\|(\overline{\mathbf{0} + (\|B\| \wedge \|A\|)}) + \overline{\|\alpha\|\overline{\|B\|}} \\
&= \|\alpha\|((\overline{\|B\|} + \overline{\|A\|})) + \overline{\|\alpha\|\overline{\|B\|}} \\
&= (\|\alpha\|(\overline{\|B\|} + \|\alpha\|\overline{\|A\|})) + \overline{\|\alpha\|\overline{\|B\|}} \\
&= \mathbf{1} + \|\alpha\|\overline{\|A\|} \\
&= \mathbf{1}
\end{aligned}$$

Case of axiom ($[\cup]$)
We show that $\mathbb{M} \models [\alpha \cup \beta]A \leftrightarrow [\alpha]A \wedge [\beta]A$. Fix any $\mathfrak{M} \in \mathbb{M}$. By Proposition 4.7, it suffices to show that:

$$\|[\alpha \cup \beta]A\| = \|[\alpha]A \wedge [\beta]A\|.$$

This is shown by:

$$\begin{aligned}
\|[\alpha \cup \beta]A\| &= \overline{\|\alpha \cup \beta\| \overline{\|A\|}} \\
&= \overline{(\|\alpha\| + \|\beta\|)\overline{\|A\|}} \\
&= \overline{\|\alpha\|\overline{\|A\|} + \|\beta\|\overline{\|A\|}} \\
&= \overline{\|\alpha\|\overline{\|A\|}} \wedge \overline{\|\beta\|\overline{\|A\|}} \\
&= \|[\alpha]A \wedge [\beta]A\|
\end{aligned}$$

Case of axiom ([;])

We show that $\mathbb{M} \models [\alpha;\beta]A \leftrightarrow [\alpha][\beta]A$. Fix any $\mathfrak{M} \in \mathbb{M}$. By Proposition 4.7, it suffices to show that:

$$\|[\alpha;\beta]A\| = \|[\alpha][\beta]A\|.$$

This is shown by:

$$\begin{aligned}
\|[\alpha;\beta]A\| &= \overline{\|\alpha;\beta\|\overline{\|A\|}} \\
&= \overline{\|\alpha\|\|\beta\|\overline{\|A\|}} \\
&= \overline{\|\alpha\|\overline{\overline{\|\beta\|\overline{\|A\|}}}} \\
&= \overline{\|\alpha\|\overline{\|[\beta]A\|}} \\
&= \|[\alpha][\beta]A\|
\end{aligned}$$

Case of axiom ([?])

We show that $\mathbb{M} \models [?B]A \leftrightarrow (B \to A)$. Fix any $\mathfrak{M} \in \mathbb{M}$. Let n be the cardinality of the (finite) domain of $\mathfrak{M}$. By Proposition 4.7, it suffices to show that:

$$\|[?B]A\| = \|B \to A\|.$$

This is shown by:

$$\begin{aligned}
\|[?B]A\| &= \overline{\|?B\|\overline{\|A\|}} \\
&= \overline{\begin{bmatrix} r_{11} & 0 & \cdots & 0 \\ 0 & r_{22} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & r_{nn} \end{bmatrix} \overline{\begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_n \end{bmatrix}}} \\
&= \overline{\begin{bmatrix} r_{11} \\ r_{22} \\ \vdots \\ r_{nn} \end{bmatrix} \wedge \overline{\begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_n \end{bmatrix}}} \\
&= \overline{\|B\| \wedge \overline{\|A\|}} \\
&= \overline{\|B\|} + \|A\| \\
&= \|B \to A\|
\end{aligned}$$

Case of axiom ($[\mathsf{r}]p$)

We show that $\mathbb{M} \models [\mathsf{r}]p \leftrightarrow p$. Fix any $\mathfrak{M} \in \mathbb{M}$. By Proposition 4.7, it suffices to show that:

$$\|[\mathsf{r}]p\|_{\mathfrak{M}} = \|p\|_{\mathfrak{M}}.$$

This is shown by:

$$\begin{array}{rcl} \|[\mathsf{r}]p\|_{\mathfrak{M}} & = & \|p\|_{\mathfrak{M}^{\mathsf{r}}} \\ & = & V^{\mathsf{r}}(p)^{M} \\ & = & V(p)^{M} \\ & = & \|p\|_{\mathfrak{M}} \end{array}$$

Case of axiom $([\mathsf{r}]\neg)$

We show that $\mathbb{M} \models [\mathsf{r}]\neg A \leftrightarrow \neg[\mathsf{r}]A$. Fix any $\mathfrak{M} \in \mathbb{M}$. By Proposition 4.7, it suffices to show that:

$$\|[\mathsf{r}]\neg A\|_{\mathfrak{M}} = \|\neg[\mathsf{r}]A\|_{\mathfrak{M}}.$$

This is shown by:

$$\begin{array}{rcl} \|[\mathsf{r}]\neg A\|_{\mathfrak{M}} & = & \|\neg A\|_{\mathfrak{M}^{\mathsf{r}}} \\ & = & \overline{\|A\|_{\mathfrak{M}^{\mathsf{r}}}} \\ & = & \overline{\|[\mathsf{r}]A\|_{\mathfrak{M}}} \\ & = & \|\neg[\mathsf{r}]A\|_{\mathfrak{M}} \end{array}$$

Case of axiom $([\mathsf{r}]\rightarrow)$

We show that $\mathbb{M} \models [\mathsf{r}](A \rightarrow B) \leftrightarrow ([\mathsf{r}]A \rightarrow [\mathsf{r}]B)$. Fix any $\mathfrak{M} \in \mathbb{M}$. By Proposition 4.7, it suffices to show that:

$$\|[\mathsf{r}](A \rightarrow B)\|_{\mathfrak{M}} = \|[\mathsf{r}]A \rightarrow [\mathsf{r}]B\|_{\mathfrak{M}}.$$

This is shown by:

$$\begin{array}{rcl} \|[\mathsf{r}](A \rightarrow B)\|_{\mathfrak{M}} & = & \overline{\|A\|_{\mathfrak{M}^{\mathsf{r}}}} + \|B\|_{\mathfrak{M}^{\mathsf{r}}} \\ & = & \overline{\|[\mathsf{r}]A\|_{\mathfrak{M}}} + \|[\mathsf{r}]B\|_{\mathfrak{M}} \\ & = & \|[\mathsf{r}]A \rightarrow [\mathsf{r}]B\|_{\mathfrak{M}} \end{array}$$

Case of axiom $([\mathsf{r}][a])$

We show that $\mathbb{M} \models [\mathsf{r}][a]A \leftrightarrow [\mathsf{r}(a)][\mathsf{r}]A$. Fix any $\mathfrak{M} \in \mathbb{M}$. By Proposition 4.7, it suffices to show that:

$$\|[\mathsf{r}][a]A\|_{\mathfrak{M}} = \|[\mathsf{r}(a)][\mathsf{r}]A\|_{\mathfrak{M}}.$$

This is shown by:

$$\begin{array}{rcl} \|[\mathsf{r}][a]A\|_{\mathfrak{M}} & = & \|[a]A\|_{\mathfrak{M}^{\mathsf{r}}} \\ & = & \overline{\|a\|_{\mathfrak{M}^{\mathsf{r}}}\overline{\|A\|_{\mathfrak{M}^{\mathsf{r}}}}} \\ & = & \overline{\|\mathsf{r}(a)\|_{\mathfrak{M}}\overline{\|[\mathsf{r}]A\|_{\mathfrak{M}}}} \\ & = & \|[\mathsf{r}(a)][\mathsf{r}]A\|_{\mathfrak{M}} \end{array}$$

Case of axiom $([\mathsf{r}][\cup])$

We show that $\mathbb{M} \models [\mathsf{r}][\alpha \cup \beta]A \leftrightarrow [\mathsf{r}][\alpha]A \wedge [\mathsf{r}][\beta]A$. Fix any $\mathfrak{M} \in \mathbb{M}$. By Proposition 4.7, it suffices to show that:

$$\|[\mathsf{r}][\alpha \cup \beta]A\|_{\mathfrak{M}} = \|[\mathsf{r}][\alpha]A \wedge [\mathsf{r}][\beta]A\|_{\mathfrak{M}}.$$

This is shown by:

$$\begin{array}{rcl}
\|[\mathsf{r}][\alpha \cup \beta]A\|_{\mathfrak{M}} & = & \|[\alpha \cup \beta]A\|_{\mathfrak{M}^{\mathsf{r}}} \\
& = & \|[\alpha]A\|_{\mathfrak{M}^{\mathsf{r}}} \wedge \|[\beta]A\|_{\mathfrak{M}^{\mathsf{r}}} \\
& = & \|[\mathsf{r}][\alpha]A\|_{\mathfrak{M}} \wedge \|[\mathsf{r}][\beta]A\|_{\mathfrak{M}} \\
& = & \|[\mathsf{r}][\alpha]A \wedge [\mathsf{r}][\beta]A\|_{\mathfrak{M}}
\end{array}$$

Case of axiom ([r][;])

We show that $\mathbb{M} \models [\mathsf{r}][\alpha;\beta]A \leftrightarrow [\mathsf{r}][\alpha][\beta]A$. Fix any $\mathfrak{M} \in \mathbb{M}$. By Proposition 4.7, it suffices to show that:

$$\|[\mathsf{r}][\alpha;\beta]A\|_{\mathfrak{M}} = \|[\mathsf{r}][\alpha][\beta]A\|_{\mathfrak{M}}.$$

This is shown by:

$$\begin{array}{rcl}
\|[\mathsf{r}][\alpha;\beta]A\|_{\mathfrak{M}} & = & \|[\alpha;\beta]A\|_{\mathfrak{M}^{\mathsf{r}}} \\
& = & \|[\alpha][\beta]A\|_{\mathfrak{M}^{\mathsf{r}}} \\
& = & \|[\mathsf{r}][\alpha][\beta]A\|_{\mathfrak{M}}
\end{array}$$

Case of axiom ([r][?])

We show that $\mathbb{M} \models [\mathsf{r}][?B]A \leftrightarrow [\mathsf{r}](B \to A)$. Fix any $\mathfrak{M} \in \mathbb{M}$. By Proposition 4.7, it suffices to show that:

$$\|[\mathsf{r}][?B]A\|_{\mathfrak{M}} = \|[\mathsf{r}](B \to A)\|_{\mathfrak{M}}.$$

This is shown by:

$$\begin{array}{rcl}
& = & \|[\mathsf{r}][?B]A\|_{\mathfrak{M}} \\
& = & \|B \to A\|_{\mathfrak{M}^{\mathsf{r}}} \\
& = & \|[\mathsf{r}](B \to A)\|_{\mathfrak{M}}
\end{array}$$

Case of inference rule (MP)

We show that if $\mathbb{M} \models A$ and $\mathbb{M} \models A \to B$, then $\mathbb{M} \models B$. Fix any $\mathfrak{M} \in \mathbb{M}$. Suppose that $\mathfrak{M} \models A$ and $\mathfrak{M} \models A \to B$, i.e., $\|A\| = \mathbf{1}$ and $\overline{\|A\|} + \|B\| = \mathbf{1}$. Our goal is to show that $\mathfrak{M} \models B$, i.e., $\|B\| = \mathbf{1}$. This is trivial since:

$$\begin{array}{rcl}
\mathbf{1} & = & \overline{\|A\|} + \|B\| \\
& = & \overline{\mathbf{1}} + \|B\| \\
& = & \mathbf{0} + \|B\| \\
& = & \|B\|
\end{array}$$

Here is another proof that can capture the meaning of our assumptions:

$$\begin{array}{rcl}
\mathbf{1} \wedge \mathbf{1} & = & \|A\| \wedge \|A \to B\| \\
& = & \|A\| \wedge (\overline{\|A\|} + \|B\|) \\
& = & (\|A\| \wedge \overline{\|A\|}) + (\|A\| \wedge \|B\|) \\
& = & (\mathbf{1} \wedge \mathbf{0}) + (\mathbf{1} \wedge \|B\|) \\
& = & \|B\| = \mathbf{1}
\end{array}$$

Case of inference rule ($\mathbf{Nec}_{[\alpha]}$)

We show that if $\mathbb{M} \models A$, then $\mathbb{M} \models [\alpha]A$. Fix any $\mathfrak{M} \in \mathbb{M}$.

Suppose that $\mathfrak{M} \models A$, i.e., $\|A\| = \mathbf{1}$. We show that $\mathfrak{M} \models [\alpha]A$, i.e., $\|[\alpha]A\| = \mathbf{1}$. This is shown by:

$$\begin{array}{rcl} \|[\alpha]A\| & = & \overline{\|\alpha\|\overline{\|A\|}} \\ & = & \overline{\|\alpha\|\overline{\mathbf{1}}} \\ & = & \overline{\|\alpha\|\mathbf{0}} \\ & = & \overline{\mathbf{0}} = \mathbf{1} \end{array}$$

Case of inference rule ($\mathbf{Nec}_{[\mathsf{r}]}$)

We show that if $\mathbb{M} \models A$, then $\mathbb{M} \models [\mathsf{r}]A$. Suppose that $\mathbb{M} \models A$, i.e., $\|A\|_{\mathfrak{M}} = \mathbf{1}_{\mathfrak{M}}$ for any model $\mathfrak{M} \in \mathbb{M}$. We show that $\mathbb{M} \models [\mathsf{r}]A$, i.e., $\|[\mathsf{r}]A\|_{\mathfrak{M}} = \mathbf{1}_{\mathfrak{M}}$ for any model $\mathfrak{M} \in \mathbb{M}$. Fix any $\mathfrak{M} \in \mathbb{M}$. Our goal is to show that:

$$\begin{array}{rrcl} & \|[\mathsf{r}]A\|_{\mathfrak{M}} & = & \mathbf{1}_{\mathfrak{M}}, \\ \text{i.e.,} & \|A\|_{\mathfrak{M}^{\mathsf{r}}} & = & \mathbf{1}_{\mathfrak{M}^{\mathsf{r}}}. \end{array}$$

This is trivial by our assumption.

□

Remarkably, our linear algebraic approach to the soundness allows us to capture semantic proofs like syntactic way.

4.3 Labelled Sequent Calculus for Dynamic Logic of Relation Changers

4.3.1 Labelled Sequent Calculus GDLRC

In this section, we introduce the labelled formalism for our sequent calculus. Let $\mathsf{Var} = \{ x, y, z, \dots \}$ be a countably infinite set of *variables* and $L = (\mathsf{r}_1, \mathsf{r}_2, \dots, \mathsf{r}_n)$ be a (possibly empty) finite list of relation changers. If L is empty, we use ε to denote such an empty list. We may regard the list of relation changers as a history or a stack of that. Suppose that $\mathfrak{M}^L, w \models A$ is already defined, then

$$\mathfrak{M}^{(L,r)}, w \models A \text{ iff } \mathfrak{M}^L, w \models [\mathsf{r}]A.$$

Given variables $x, y \in \mathsf{Var}$, a list L of relation changers, a formula $A \in \mathsf{Form}$ and a program $\alpha \in \mathsf{PR}$, we define *labelled expressions* by:

$$\varphi ::= x :^L A \mid x \,\mathsf{R}^L_\alpha\, y \mid x = y$$

We say that $x :^L A$ is a *labelled formula*, $x \,\mathsf{R}^L_\alpha\, y$ is a *relational atom* and $x = y$ is an *equality atom*. A labelled formula $x :^L A$ means that 'after the successive updates of relation changers in L, A holds at state x,' a relational atom $x \,\mathsf{R}^L_\alpha\, y$ means that 'after the successive updates of relation changers in L, there is a link of program α from x to y,' and an equality atom means that 'state x equals to state y.' We note that labelled expressions are counterparts of objects in Kripke semantics. Namely, given a model $\mathfrak{M} = (W, (R_a)_{a \in \mathsf{AP}}, V)$ and if we regard $w, v \in W$ as counterparts of $x, y \in \mathsf{Var}$, a labelled formula $x :^L A$ corresponds to the satisfaction relation '$\mathfrak{M}^L, w \models A$,' a relational atom

$x \, \mathsf{R}^L_\alpha \, y$ corresponds to a link '$wR^L_\alpha v$' and an equality atom $x = y$ corresponds to an equality '$w = v$,' respectively. Let Γ and Δ be multisets of labelled expressions. We define a *labelled sequent* $\Gamma \Rightarrow \Delta$ by a pair (Γ, Δ) of labelled expressions, whose reading is 'if all of Γ hold, then some of Δ holds.' Based on these definitions, we define our labelled sequent calculus **GDLRC** by all initial sequents and all inference rules in Table 4.2. We also define the labelled sequent calculus **GDLRC**$^-$ by **GDLRC** without the cut rule. In order to refer both left and right rules shortly, we also use simple notation of the rule name without L and R, e.g., (at) means both (Lat) and (Rat) rules. The multisets Γ, Δ etc. in each inference rule in **GDLRC** are called the *context*. In the lower sequent of each inference rule, the labelled expression(s) not in the context is called the *principal labelled expression(s)*. A *derivation* $\mathfrak{D}$ in **GDLRC** is inductively defined as a finite tree generated by initial sequents and inference rules in Table 4.2. A labelled sequent in the root node of $\mathfrak{D}$ is called the *end sequent*. The *height* of a derivation $\mathfrak{D}$ is defined as the maximum length of branches in $\mathfrak{D}$ from the end sequent of that to an initial sequent. We say that $\Gamma \Rightarrow \Delta$ is *derivable* in **GDLRC** (notation: $\vdash_{\mathbf{GDLRC}} \Gamma \Rightarrow \Delta$) if there exists a derivation $\mathfrak{D}$ in **GDLRC** whose end sequent is $\Gamma \Rightarrow \Delta$. A formula A is a theorem of **GDLRC** if the labelled sequent $\Rightarrow x :^\varepsilon A$ is derivable in **GDLRC** for all $x \in \mathsf{Var}$. These definitions are also used for **GDLRC**$^-$.

Remark that all syntactic objects in the labelled sequent calculus **GDLRC** are finite, e.g., labelled expressions, sequents and derivations. We may regard **GDLRC** as a formalized version of Kripke semantics of **DLRC**. The crucial difference between the ordinary Kripke semantics and its formalized version is this finiteness.

Definition 4.8. *The length ℓ of formulas $p \in \mathsf{Prop}$ and $A, B \in \mathsf{Form}$, programs $a \in \mathsf{AP}$ and $\alpha, \beta \in \mathsf{PR}$, a relation changer $\mathsf{r} \in \mathsf{RC}$, and labelled expressions are defined as follows:*

$$\begin{array}{lcl}
\ell(p) & = & 1, \\
\ell(\neg A) & = & \ell(A) + 1, \\
\ell(A \to B) & = & \ell(A) + \ell(B) + 1, \\
\ell([\alpha]A) & = & \ell(\alpha) + \ell(A) + 1, \\
\ell([\mathsf{r}]A) & = & \ell(\mathsf{r}) + \ell(A) + 1, \\
\ell(a) & = & 1, \\
\ell(\alpha;\beta) & = & \ell(\alpha) + \ell(\beta) + 1, \\
\ell(\alpha \cup \beta) & = & \ell(\alpha) + \ell(\beta) + 1, \\
\ell(?A) & = & \ell(A) + 1, \\
\ell(\varepsilon) & = & 0, \\
\ell(\mathsf{r}) & = & \Sigma_{a \in \mathsf{AP}} \ell(\alpha_a) \ \textit{where}\ \mathsf{r} = (a := \alpha_a)_{a \in \mathsf{AP}}, \\
\ell(L) & = & \Sigma_{i=0}^{n} \ell(\mathsf{r}_i) \ \textit{where}\ L = (\mathsf{r}_0, \ldots, \mathsf{r}_n), \\
\ell(x :^L A) & = & \ell(L) + \ell(A), \\
\ell(x \, \mathsf{R}^L_\alpha \, y) & = & \ell(L) + \ell(\alpha), \\
\ell(x = y) & = & 0.
\end{array}$$

Proposition 4.9. *Let $p \in \mathsf{Prop}$, $A \in \mathsf{Form}$, $a \in \mathsf{AP}$, $\mathsf{r} \in \mathsf{RC}$ and $x, y \in \mathsf{Var}$.*

(i) $\ell(x :^\varepsilon [\mathsf{r}]A) > \ell(x :^{\mathsf{r}} A)$.

(ii) $\ell(x :^{\mathsf{r}} p) > \ell(x :^\varepsilon p)$.

(iii) $\ell(x \, \mathsf{R}^{\mathsf{r}}_a \, y) > \ell(x \, \mathsf{R}^\varepsilon_{\mathsf{r}(a)} \, y)$.

Table 4.2: Gentzen-style Sequent Calculus **GDLRC**

(Initial sequents)

$$x :^L A \Rightarrow x :^L A \qquad x\,\mathsf{R}^L_\alpha\, y \Rightarrow x\,\mathsf{R}^L_\alpha\, y \qquad x = y \Rightarrow x = y$$

(Structural rules)

$$\frac{\Gamma \Rightarrow \Delta}{\Gamma \Rightarrow \Delta, \varphi}\,(Rw) \qquad \frac{\Gamma \Rightarrow \Delta}{\varphi, \Gamma \Rightarrow \Delta}\,(Lw) \qquad \frac{\Gamma \Rightarrow \Delta, \varphi, \varphi}{\Gamma \Rightarrow \Delta, \varphi}\,(Rc) \qquad \frac{\varphi, \varphi, \Gamma \Rightarrow \Delta}{\varphi, \Gamma \Rightarrow \Delta}\,(Lc)$$

(Logical rules)

$$\frac{x :^L A, \Gamma \Rightarrow \Delta}{\Gamma \Rightarrow \Delta, x :^L \neg A}\,(R\neg) \qquad \frac{\Gamma \Rightarrow \Delta, x :^L A}{x :^L \neg A, \Gamma \Rightarrow \Delta}\,(L\neg)$$

$$\frac{x :^L A_1, \Gamma \Rightarrow \Delta, x :^L A_2}{\Gamma \Rightarrow \Delta, x :^L A_1 \to A_2}\,(R \to) \qquad \frac{\Gamma \Rightarrow \Delta, x :^L A_1 \quad x :^L A_2, \Gamma \Rightarrow \Delta}{x :^L A_1 \to A_2, \Gamma \Rightarrow \Delta}\,(L \to)$$

$$\frac{x\,\mathsf{R}^L_\alpha\, y, \Gamma \Rightarrow \Delta, y :^L A}{\Gamma \Rightarrow \Delta, x :^L [\alpha]A}\,(R[\alpha])^\dagger \qquad \frac{\Gamma \Rightarrow \Delta, x\,\mathsf{R}^L_\alpha\, y \quad y :^L A, \Gamma \Rightarrow \Delta}{x :^L [\alpha]A, \Gamma \Rightarrow \Delta}\,(L[\alpha])$$

†: y does not appear in the lower sequent.

$$\frac{\Gamma \Rightarrow \Delta, x :^{(L,\mathsf{r})} A}{\Gamma \Rightarrow \Delta, x :^L [\mathsf{r}]A}\,(R[\mathsf{r}]) \qquad \frac{x :^{(L,\mathsf{r})} A, \Gamma \Rightarrow \Delta}{x :^L [\mathsf{r}]A, \Gamma \Rightarrow \Delta}\,(L[\mathsf{r}])$$

(Program rules)

$$\frac{\Gamma \Rightarrow \Delta, x\,\mathsf{R}^L_{\alpha_i}\, y}{\Gamma \Rightarrow \Delta, x\,\mathsf{R}^L_{\alpha_1 \cup \alpha_2}\, y}\,(R\cup)_{i \in \{1,2\}} \qquad \frac{x\,\mathsf{R}^L_{\alpha_1}\, y, \Gamma \Rightarrow \Delta \quad x\,\mathsf{R}^L_{\alpha_2}\, y, \Gamma \Rightarrow \Delta}{x\,\mathsf{R}^L_{\alpha_1 \cup \alpha_2}\, y, \Gamma \Rightarrow \Delta}\,(L\cup)$$

$$\frac{\Gamma \Rightarrow \Delta, x\,\mathsf{R}^L_{\alpha_1}\, z \quad \Gamma \Rightarrow \Delta, z\,\mathsf{R}^L_{\alpha_2}\, y}{\Gamma \Rightarrow \Delta, x\,\mathsf{R}^L_{\alpha_1;\alpha_2}\, y}\,(R;) \qquad \frac{x\,\mathsf{R}^L_{\alpha_1}\, z, z\,\mathsf{R}^L_{\alpha_2}\, y, \Gamma \Rightarrow \Delta}{x\,\mathsf{R}^L_{\alpha_1;\alpha_2}\, y, \Gamma \Rightarrow \Delta}\,(L;)^\ddagger$$

‡: z does not appear in the lower sequent.

$$\frac{\Gamma \Rightarrow \Delta, x = y \quad \Gamma \Rightarrow \Delta, x :^L A}{\Gamma \Rightarrow \Delta, x\,\mathsf{R}^L_{?A}\, y}\,(R?) \qquad \frac{x = y, \Gamma \Rightarrow \Delta}{x\,\mathsf{R}^L_{?A}\, y, \Gamma \Rightarrow \Delta}\,(L?_1) \qquad \frac{x :^L A, \Gamma \Rightarrow \Delta}{x\,\mathsf{R}^L_{?A}\, y, \Gamma \Rightarrow \Delta}\,(L?_2)$$

(Equality rules)

$$\frac{\Gamma \Rightarrow \Delta, x = y \quad \Gamma \Rightarrow \Delta, x :^L p}{\Gamma \Rightarrow \Delta, y :^L p}\,(= at) \qquad \frac{x = x, \Gamma \Rightarrow \Delta}{\Gamma \Rightarrow \Delta}\,(= ref)$$

$$\frac{\Gamma \Rightarrow \Delta, x = y \quad y = x, \Gamma \Rightarrow \Delta}{\Gamma \Rightarrow \Delta}\,(= sym) \qquad \frac{\Gamma \Rightarrow \Delta, x = y \quad \Gamma \Rightarrow \Delta, y = z \quad x = z, \Gamma \Rightarrow \Delta}{\Gamma \Rightarrow \Delta}\,(= tra)$$

$$\frac{\Gamma \Rightarrow \Delta, x = y \quad \Gamma \Rightarrow \Delta, x\,\mathsf{R}^L_a\, z \quad y\,\mathsf{R}^L_a\, z, \Gamma \Rightarrow \Delta}{\Gamma \Rightarrow \Delta}\,(= rel_1)$$

$$\frac{\Gamma \Rightarrow \Delta, x = y \quad \Gamma \Rightarrow \Delta, z\,\mathsf{R}^L_a\, x \quad z\,\mathsf{R}^L_a\, y, \Gamma \Rightarrow \Delta}{\Gamma \Rightarrow \Delta}\,(= rel_2)$$

(Relation changer rules)

$$\frac{\Gamma \Rightarrow \Delta, x :^{L} p}{\Gamma \Rightarrow \Delta, x :^{(L,\mathsf{r})} p}\ (Rat) \qquad \frac{x :^{L} p, \Gamma \Rightarrow \Delta}{x :^{(L,\mathsf{r})} p, \Gamma \Rightarrow \Delta}\ (Lat)$$

$$\frac{\Gamma \Rightarrow \Delta, x\, \mathsf{R}^{L}_{\mathsf{r}(a)}\, y}{\Gamma \Rightarrow \Delta, x\, \mathsf{R}^{(L,\mathsf{r})}_{a}\, y}\ (Rrel) \qquad \frac{x\, \mathsf{R}^{L}_{\mathsf{r}(a)}\, y, \Gamma \Rightarrow \Delta}{x\, \mathsf{R}^{(L,\mathsf{r})}_{a}\, y, \Gamma \Rightarrow \Delta}\ (Lrel)$$

(Cut rule)

$$\frac{\Gamma \Rightarrow \Delta, \varphi \quad \varphi, \Sigma \Rightarrow \Pi}{\Gamma, \Sigma \Rightarrow \Delta, \Pi}\ (Cut)$$

Proof.

(i) The length of the left side is:

$$\ell(x :^{\varepsilon} [\mathsf{r}]A) = \ell(\mathsf{r}) + \ell(A) + 1.$$

The right side is:

$$\ell(x :^{\mathsf{r}} A) = \ell(\mathsf{r}) + \ell(A).$$

Therefore, $\ell(\mathsf{r}) + \ell(A) + 1 > \ell(\mathsf{r}) + \ell(A)$.

(ii) The length of the left side is:

$$\begin{array}{rcl} \ell(x :^{\mathsf{r}} p) & = & \ell(\mathsf{r}) + \ell(p), \\ & = & \ell(\mathsf{r}) + 1. \end{array}$$

The right side is:

$$\begin{array}{rcl} \ell(x :^{\varepsilon} p) & = & \ell(p), \\ & = & 1. \end{array}$$

Since $\ell(\mathsf{r}) > 0$, we obtain $\ell(\mathsf{r}) + 1 > 1$.

(iii) The length of the left side is:

$$\begin{array}{rcl} \ell(x\, \mathsf{R}^{\mathsf{r}}_{a}\, y) & = & \ell(\mathsf{r}) + \ell(a), \\ & = & \ell(\mathsf{r}) + 1. \end{array}$$

The right side is:

$$\ell(x\, \mathsf{R}^{\varepsilon}_{\mathsf{r}(a)}\, y) = \ell(\alpha_a).$$

Since $\ell(\mathsf{r}) \geq \ell(\alpha_a)$, we obtain $\ell(\mathsf{r}) + 1 > \ell(\alpha_a)$.

$\square$

4.3.2 All theorems of HDLRC are derivable in GDLRC

Lemma 4.10. *For any $n \in \mathbb{N}$ and any list $L \in (\mathsf{RC})^*$ of relation changers,*

(i) if $\ell(x :^{L} A) \leq n$, then $\vdash_{\mathbf{GDLRC}} x = y,\ x :^{L} A \Rightarrow y :^{L} A$ for any $x, y \in \mathsf{Var}$ and any $A \in \mathsf{Form}$.

(ii) *if* $\ell(x\,\mathsf{R}^{L}_{\alpha}\,y) \leq n$, *then* $\vdash_{\mathbf{GDLRC}} x = y,\ y\,\mathsf{R}^{L}_{\alpha}\,z \Rightarrow x\,\mathsf{R}^{L}_{\alpha}\,z$ *for any* $x, y, z \in \mathsf{Var}$ *and any* $\alpha \in \mathsf{PR}$.

Proof. The proofs of (i) and (ii) are done simultaneously by induction on $n \in \mathbb{N}$. First, we show base cases for both (i) and (ii).

Basis for (i) where $x :^{L} A$ is of the form $x :^{\varepsilon} p$ ($p \in \mathsf{Prop}$):

$$\dfrac{\dfrac{x = y \Rightarrow x = y}{x = y, x :^{\varepsilon} p \Rightarrow x = y}\,(Lw) \qquad \dfrac{x :^{\varepsilon} p \Rightarrow x :^{\varepsilon} p}{x = y, x :^{\varepsilon} p \Rightarrow x :^{\varepsilon} p}\,(Lw)}{x = y, x :^{\varepsilon} p \Rightarrow y :^{\varepsilon} p}\,(= at)$$

Basis for (ii) where $x\,\mathsf{R}^{L}_{\alpha}\,y$ is of the form $x\,\mathsf{R}^{\varepsilon}_{a}\,y$ ($a \in \mathsf{AP}$):
First, we construct the following derivation $\mathfrak{D}$.

$$\dfrac{\dfrac{y = x \Rightarrow y = x}{y = x, y\,\mathsf{R}^{\varepsilon}_{a}\,z \Rightarrow x\,\mathsf{R}^{\varepsilon}_{a}\,z, y = x}\,(w) \qquad \dfrac{y\,\mathsf{R}^{\varepsilon}_{a}\,z \Rightarrow y\,\mathsf{R}^{\varepsilon}_{a}\,z}{y = x, y\,\mathsf{R}^{\varepsilon}_{a}\,z \Rightarrow x\,\mathsf{R}^{\varepsilon}_{a}\,z, y\,\mathsf{R}^{\varepsilon}_{a}\,z}\,(w) \qquad \dfrac{x\,\mathsf{R}^{\varepsilon}_{a}\,z \Rightarrow x\,\mathsf{R}^{\varepsilon}_{a}\,z}{x\,\mathsf{R}^{\varepsilon}_{a}\,z, y = x, y\,\mathsf{R}^{\varepsilon}_{a}\,z \Rightarrow x\,\mathsf{R}^{\varepsilon}_{a}\,z}\,(w)}{y = x, y\,\mathsf{R}^{\varepsilon}_{a}\,z \Rightarrow x\,\mathsf{R}^{\varepsilon}_{a}\,z}\,(= rel_1)$$

Next, we obtain the following derivation by $\mathfrak{D}$.

$$\dfrac{\dfrac{x = y \Rightarrow x = y}{x = y, y\,\mathsf{R}^{\varepsilon}_{a}\,z \Rightarrow x\,\mathsf{R}^{\varepsilon}_{a}\,z, x = y}\,(w) \qquad \dfrac{\mathfrak{D}}{y = x, x = y, y\,\mathsf{R}^{\varepsilon}_{a}\,z \Rightarrow x\,\mathsf{R}^{\varepsilon}_{a}\,z}\,(Lw)}{x = y, y\,\mathsf{R}^{\varepsilon}_{a}\,z \Rightarrow x\,\mathsf{R}^{\varepsilon}_{a}\,z}\,(= sym)$$

Next, we show inductive steps for both (i) and (ii).

Case (i-a) where $x :^{L} A$ is of the form $x :^{L} \neg B$:

$$\dfrac{\dfrac{\dfrac{x = y \Rightarrow x = y}{x = y, y :^{L} B \Rightarrow x :^{L} B, x = y}\,(w) \qquad \dfrac{\begin{array}{c}\vdots\ \mathfrak{D}\\ y = x, y :^{L} B \Rightarrow x :^{L} B\end{array}}{y = x, x = y, y :^{L} B \Rightarrow x :^{L} B}\,(Lw)}{x = y, y :^{L} B \Rightarrow x :^{L} B}\,(= sym)}{x = y, x :^{L} \neg B \Rightarrow y :^{L} \neg B}\,(\neg)$$

where $\mathfrak{D}$ is obtained by induction hypothesis.

Case (i-b) where $x :^{L} A$ is of the form $x :^{L} A_1 \to A_2$:

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{x = y \Rightarrow x = y}{y :^{L} A_1, x = y \Rightarrow x :^{L} A_1, x = y}\,(w) \qquad \dfrac{\begin{array}{c}\vdots\ \mathfrak{D}_1\\ y = x, y :^{L} A_1, \Rightarrow x :^{L} A_1\end{array}}{y = x, y :^{L} A_1, x = y \Rightarrow x :^{L} A_1}\,(Lw)}{y :^{L} A_1, x = y \Rightarrow x :^{L} A_1}\,(= sym)}{y :^{L} A_1, x = y \Rightarrow y :^{L} A_2, x :^{L} A_1}\,(Rw) \qquad \dfrac{\begin{array}{c}\vdots\ \mathfrak{D}_2\\ x :^{L} A_2, x = y \Rightarrow y :^{L} A_2\end{array}}{x :^{L} A_2, y :^{L} A_1, x = y \Rightarrow y :^{L} A_2}\,(Lw)}{y :^{L} A_1, x = y, x :^{L} A_1 \to A_2 \Rightarrow y :^{L} A_2}\,(L\to)}{x = y, x :^{L} A_1 \to A_2 \Rightarrow y :^{L} A_1 \to A_2}\,(R\to)$$

where $\mathfrak{D}_1$ and $\mathfrak{D}_2$ are obtained by induction hypothesis.

Case (i-c) where $x :^L A$ is of the form $x :^L [\alpha]B$:

$$\dfrac{\dfrac{\dfrac{\begin{array}{c}\vdots\ \mathfrak{D}\\ y\,\mathsf{R}^L_\alpha\,z, x = y \Rightarrow x\,\mathsf{R}^L_\alpha\,z\end{array}}{y\,\mathsf{R}^L_\alpha\,z, x = y \Rightarrow z :^L B, x\,\mathsf{R}^L_\alpha\,z}\,(Rw) \quad \dfrac{z :^L B \Rightarrow z :^L B}{z :^L B, y\,\mathsf{R}^L_\alpha\,z, x = y \Rightarrow z :^L B}\,(w)}{y\,\mathsf{R}^L_\alpha\,z, x = y, x :^L [\alpha]B \Rightarrow z :^L B}\,(L[\alpha])}{x = y, x :^L [\alpha]B \Rightarrow y :^L [\alpha]B}\,(R[\alpha])$$

where $\mathfrak{D}$ is obtained by induction hypothesis.

Case (i-d) where $x :^L A$ is of the form $x :^L [\mathsf{r}]B$:

$$\dfrac{\begin{array}{c}\vdots\ \mathfrak{D}\\ x = y, x :^{(L,r)} A \Rightarrow y :^{(L,r)} A\end{array}}{x = y, x :^L [\mathsf{r}]B \Rightarrow y :^L [\mathsf{r}]B}\,([\mathsf{r}])$$

where $\mathfrak{D}$ is obtained by induction hypothesis. Note that $\ell(x :^{(L,r)} A) < \ell(x :^L [\mathsf{r}]A)$.

Case (i-e) where $x :^L A$ is of the form $x :^{(L,r)} p$:

$$\dfrac{\dfrac{\begin{array}{c}\vdots\ \mathfrak{D}\\ x = y, x :^\varepsilon p \Rightarrow y :^\varepsilon p\end{array}}{x = y, x :^L p \Rightarrow y :^L p}\,(at)}{x = y, x :^{(L,r)} p \Rightarrow y :^{(L,r)} p}\,(at)$$

where $\mathfrak{D}$ is obtained by induction hypothesis.

Case (ii-a) where $y\,\mathsf{R}^L_\alpha\,z$ is of the form $y\,\mathsf{R}^L_{\alpha_1\cup\alpha_2}\,z$:

$$\dfrac{\dfrac{\begin{array}{c}\vdots\ \mathfrak{D}_1\\ x = y, y\,\mathsf{R}^L_{\alpha_1}\,z \Rightarrow x\,\mathsf{R}^L_{\alpha_1}\,z\end{array}}{x = y, y\,\mathsf{R}^L_{\alpha_1}\,z \Rightarrow x\,\mathsf{R}^L_{\alpha_1\cup\alpha_2}\,z}\,(R\cup) \quad \dfrac{\begin{array}{c}\vdots\ \mathfrak{D}_2\\ x = y, y\,\mathsf{R}^L_{\alpha_2}\,z \Rightarrow x\,\mathsf{R}^L_{\alpha_2}\,z\end{array}}{x = y, y\,\mathsf{R}^L_{\alpha_2}\,z \Rightarrow x\,\mathsf{R}^L_{\alpha_1\cup\alpha_2}\,z}\,(R\cup)}{x = y, y\,\mathsf{R}^L_{\alpha_1\cup\alpha_2}\,z \Rightarrow x\,\mathsf{R}^L_{\alpha_1\cup\alpha_2}\,z}\,(L\cup)$$

where $\mathfrak{D}_1$ and $\mathfrak{D}_2$ are obtained by induction hypothesis.

Case (ii-b) where $y\,\mathsf{R}^L_\alpha\,z$ is of the form $y\,\mathsf{R}^L_{\alpha_1;\alpha_2}\,z$:

$$\dfrac{\dfrac{\dfrac{\begin{array}{c}\vdots\ \mathfrak{D}\\ x = y, y\,\mathsf{R}^L_{\alpha_1}\,w \Rightarrow x\,\mathsf{R}^L_{\alpha_1}\,w\end{array}}{x = y, y\,\mathsf{R}^L_{\alpha_1}\,w, w\,\mathsf{R}^L_{\alpha_2}\,z \Rightarrow x\,\mathsf{R}^L_{\alpha_1}\,w}\,(Lw) \quad \dfrac{w\,\mathsf{R}^L_{\alpha_2}\,z \Rightarrow w\,\mathsf{R}^L_{\alpha_2}\,z}{x = y, y\,\mathsf{R}^L_{\alpha_1}\,w, w\,\mathsf{R}^L_{\alpha_2}\,z \Rightarrow w\,\mathsf{R}^L_{\alpha_2}\,z}\,(Lw)}{x = y, y\,\mathsf{R}^L_{\alpha_1}\,w, w\,\mathsf{R}^L_{\alpha_2}\,z \Rightarrow x\,\mathsf{R}^L_{\alpha_1;\alpha_2}\,z}\,(R;)}{x = y, y\,\mathsf{R}^L_{\alpha_1;\alpha_2}\,z \Rightarrow x\,\mathsf{R}^L_{\alpha_1;\alpha_2}\,z}\,(L;)$$

where $\mathfrak{D}$ is obtained by induction hypothesis.

Case (ii-c) where $y\,\mathsf{R}^L_\alpha\,z$ is of the form $y\,\mathsf{R}^L_{?A}\,z$:

First, we construct the following derivation $\mathfrak{D}_1$.

$$\dfrac{\dfrac{x=y\Rightarrow x=y}{x=y,y=z\Rightarrow x=z,x=y}\,(w)\quad \dfrac{y=z\Rightarrow y=z}{x=y,y=z\Rightarrow x=z,y=z}\,(w)\quad \dfrac{x=z\Rightarrow x=z}{x=z,x=y,y=z\Rightarrow x=z}\,(Lw)}{x=y,y=z\Rightarrow x=z}\,(=tra)$$

Second, we also construct the following derivation $\mathfrak{D}_2$.

$$\dfrac{\dfrac{x=y\Rightarrow x=y}{x=y,y:^L A\Rightarrow x:^L A,x=y}\,(w)\quad \dfrac{\begin{matrix}\vdots\ \mathfrak{D}_3\\ y=x,y:^L A\Rightarrow x:^L A\end{matrix}}{y=x,x=y,y:^L A\Rightarrow x:^L A}\,(Lw)}{x=y,y:^L A\Rightarrow x:^L A}\,(=sym)$$

where $\mathfrak{D}_3$ is obtained by induction hypothesis. Finally, we obtain the following derivation by $\mathfrak{D}_1$ and $\mathfrak{D}_2$.

$$\dfrac{\dfrac{\mathfrak{D}_1}{x=y,y\,\mathsf{R}^L_{?A}\,z\Rightarrow x=z}\,(L?_1)\quad \dfrac{\mathfrak{D}_2}{x=y,y\,\mathsf{R}^L_{?A}\,z\Rightarrow x:^L A}\,(L?_2)}{x=y,y\,\mathsf{R}^L_{?A}\,z\Rightarrow x\,\mathsf{R}^L_{?A}\,z}\,(R?)$$

Case (ii-d) where $y\,\mathsf{R}^L_\alpha\,z$ is of the form $y\,\mathsf{R}^{(L,r)}_a\,z$:

$$\dfrac{\begin{matrix}\vdots\ \mathfrak{D}\\ x=y,y\,\mathsf{R}^L_{\mathsf{r}(a)}\,z\Rightarrow x\,\mathsf{R}^L_{\mathsf{r}(a)}\,z\end{matrix}}{x=y,y\,\mathsf{R}^{(L,r)}_a\,z\Rightarrow x\,\mathsf{R}^{(L,r)}_a\,z}\,(rel)$$

where $\mathfrak{D}$ is obtained by induction hypothesis.

□

Theorem 4.11. *For any $A\in\mathsf{Form}$ and any $x\in\mathsf{Var}$, if $\vdash_{\mathsf{HDLRC}} A$, then $\vdash_{\mathsf{GDLRC}}\Rightarrow x:^\varepsilon A$.*

Proof. We show our goal by induction of the height of the derivation of A in HDLRC. Here, we show only the cases of inference rule (**MP**) and axioms and rules for relation changers of HDLRC listed in Table 4.1, i.e., the axioms from ($[\mathsf{r}]p$) to ($[\mathsf{r}]?$) and the rule ($\mathbf{Nec}_{[\mathsf{r}]}$) in the table. In the following, we show the base cases.

Case of axiom ($[\mathsf{r}]p$)

$$\dfrac{\dfrac{\dfrac{\dfrac{x:^\varepsilon p\Rightarrow x:^\varepsilon p}{x:^{\mathsf{r}} p\Rightarrow x:^\varepsilon p}\,(Lat)}{x:^\varepsilon [\mathsf{r}]p\Rightarrow x:^\varepsilon p}\,(L[\mathsf{r}])}{\Rightarrow x:^\varepsilon [\mathsf{r}]p\to p}\,(R\to)\quad \dfrac{\dfrac{\dfrac{x:^\varepsilon p\Rightarrow x:^\varepsilon p}{x:^\varepsilon p\Rightarrow x:^{\mathsf{r}} p}\,(Rat)}{x:^\varepsilon p\Rightarrow x:^\varepsilon [\mathsf{r}]p}\,(R[\mathsf{r}])}{\Rightarrow x:^\varepsilon p\to [\mathsf{r}]p}\,(R\to)}{\Rightarrow x:^\varepsilon [\mathsf{r}]p\leftrightarrow p}\,(R\wedge)$$

Case of axiom ($[\mathsf{r}]\neg$)

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{x:^{\mathsf{r}} A\Rightarrow x:^{\mathsf{r}} A}{x:^\varepsilon [\mathsf{r}]A\Rightarrow x:^{\mathsf{r}} A}\,(L[\mathsf{r}])}{x:^{\mathsf{r}} \neg A\Rightarrow x:^\varepsilon \neg[\mathsf{r}]A}\,(\neg)}{x:^\varepsilon [\mathsf{r}]\neg A\Rightarrow x:^{\mathsf{r}} \neg[\mathsf{r}]A}\,(L[\mathsf{r}])}{\Rightarrow x:^\varepsilon [\mathsf{r}]\neg A\to \neg[\mathsf{r}]A}\,(R\to)\quad \dfrac{\dfrac{\dfrac{\dfrac{x:^{\mathsf{r}} A\Rightarrow x:^{\mathsf{r}} A}{x:^{\mathsf{r}} A\to x:^\varepsilon [\mathsf{r}]A}\,(R[\mathsf{r}])}{x:^\varepsilon \neg[\mathsf{r}]A\Rightarrow x:^{\mathsf{r}} \neg A}\,(\neg)}{x:^\varepsilon \neg[\mathsf{r}]A\Rightarrow x:^\varepsilon [\mathsf{r}]\neg A}\,(R[\mathsf{r}])}{\Rightarrow x:^\varepsilon \neg[\mathsf{r}]A\to [\mathsf{r}]\neg A}\,(R\to)}{\Rightarrow x:^\varepsilon [\mathsf{r}]\neg A\leftrightarrow \neg[\mathsf{r}]A}\,(R\wedge)$$

Case of axiom $([\mathsf{r}]\to)$

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{x :^{\mathsf{r}} A \Rightarrow x :^{\mathsf{r}} A}{x :^{\mathsf{r}} A \Rightarrow x :^{\mathsf{r}} B, x :^{\mathsf{r}} A}(Rw) \quad \dfrac{x :^{\mathsf{r}} B \Rightarrow x :^{\mathsf{r}} B}{x :^{\mathsf{r}} B, x :^{\mathsf{r}} A \Rightarrow x :^{\mathsf{r}} B}(Lw)}{x :^{\mathsf{r}} A \to B, x :^{\mathsf{r}} A \Rightarrow x :^{\mathsf{r}} B}(L\to)}{x :^{\varepsilon} [\mathsf{r}](A \to B), x :^{\varepsilon} [\mathsf{r}]A \Rightarrow x :^{\varepsilon} [\mathsf{r}]B}([\mathsf{r}])}{x :^{\varepsilon} [\mathsf{r}](A \to B) \Rightarrow x :^{\varepsilon} [\mathsf{r}]A \to [\mathsf{r}]B}(R\to)}{\Rightarrow x :^{\varepsilon} [\mathsf{r}](A \to B) \to ([\mathsf{r}]A \to [\mathsf{r}]B)}(R\to) \quad \dfrac{\dfrac{\dfrac{\dfrac{\dfrac{x :^{\mathsf{r}} A \Rightarrow x :^{\mathsf{r}} A}{x :^{\mathsf{r}} A \Rightarrow x :^{\mathsf{r}} B, x :^{\mathsf{r}} A}(Rw) \quad \dfrac{x :^{\mathsf{r}} B \Rightarrow x :^{\mathsf{r}} B}{x :^{\mathsf{r}} B, x :^{\mathsf{r}} A \Rightarrow x :^{\mathsf{r}} B}(Lw)}{x :^{\mathsf{r}} A \to B, x :^{\mathsf{r}} A \Rightarrow x :^{\mathsf{r}} B}(L\to)}{x :^{\varepsilon} [\mathsf{r}](A \to B), x :^{\varepsilon} [\mathsf{r}]A \Rightarrow x :^{\varepsilon} [\mathsf{r}]B}([\mathsf{r}])}{x :^{\varepsilon} [\mathsf{r}](A \to B) \Rightarrow x :^{\varepsilon} [\mathsf{r}]A \to [\mathsf{r}]B}(R\to)}{\Rightarrow x :^{\varepsilon} ([\mathsf{r}]A \to [\mathsf{r}]B) \to [\mathsf{r}](A \to B)}(R\to)}{\Rightarrow x :^{\varepsilon} [\mathsf{r}](A \to B) \leftrightarrow ([\mathsf{r}]A \to [\mathsf{r}]B)}(R\wedge)$$

Case of axiom $([\mathsf{r}][a])$

The direction from left to right is shown as follows.

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{x \,\mathsf{R}^{\varepsilon}_{\mathsf{r}(a)}\, y \Rightarrow x \,\mathsf{R}^{\varepsilon}_{\mathsf{r}(a)}\, y}{x \,\mathsf{R}^{\varepsilon}_{\mathsf{r}(a)}\, y \Rightarrow x \,\mathsf{R}^{\mathsf{r}}_{a}\, y}(Rrel)}{x \,\mathsf{R}^{\varepsilon}_{\mathsf{r}(a)}\, y \Rightarrow y :^{\mathsf{r}} A, x \,\mathsf{R}^{\mathsf{r}}_{a}\, y}(Rw) \quad \dfrac{y :^{\mathsf{r}} A \Rightarrow y :^{\mathsf{r}} A}{y :^{\mathsf{r}} A, x \,\mathsf{R}^{\varepsilon}_{\mathsf{r}(a)}\, y \Rightarrow y :^{\mathsf{r}} A}(Lw)}{x \,\mathsf{R}^{\varepsilon}_{\mathsf{r}(a)}\, y, x :^{\mathsf{r}} [a]A \Rightarrow y :^{\mathsf{r}} A}(L[a])}{x \,\mathsf{R}^{\varepsilon}_{\mathsf{r}(a)}\, y, x :^{\mathsf{r}} [a]A \Rightarrow y :^{\varepsilon} [\mathsf{r}]A}(R[\mathsf{r}])}{x :^{\mathsf{r}} [a]A \Rightarrow x :^{\varepsilon} [\mathsf{r}(a)][\mathsf{r}]A}(R[\mathsf{r}(a)])}{x :^{\varepsilon} [\mathsf{r}][a]A \Rightarrow x :^{\varepsilon} [\mathsf{r}(a)][\mathsf{r}]A}(L[\mathsf{r}])}{\Rightarrow x :^{\varepsilon} [\mathsf{r}][a]A \to [\mathsf{r}(a)][\mathsf{r}]A}(R\to)$$

The direction from right to left is shown as follows.

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{x \,\mathsf{R}^{\varepsilon}_{\mathsf{r}(a)}\, y \Rightarrow x \,\mathsf{R}^{\varepsilon}_{\mathsf{r}(a)}\, y}{x \,\mathsf{R}^{\mathsf{r}}_{a}\, y \Rightarrow x \,\mathsf{R}^{\varepsilon}_{\mathsf{r}(a)}\, y}(Lrel)}{x \,\mathsf{R}^{\mathsf{r}}_{a}\, y \Rightarrow y :^{\mathsf{r}} A, x \,\mathsf{R}^{\varepsilon}_{\mathsf{r}(a)}\, y}(Rw)}{x \,\mathsf{R}^{\mathsf{r}}_{a}\, y, x :^{\varepsilon} [\mathsf{r}(a)][\mathsf{r}]A \Rightarrow y :^{\mathsf{r}} A}(L[\mathsf{r}(a)]) \quad \dfrac{\dfrac{\dfrac{y :^{\mathsf{r}} A \Rightarrow y :^{\mathsf{r}} A}{y :^{\varepsilon} [\mathsf{r}]A \Rightarrow y :^{\mathsf{r}} A}(L[\mathsf{r}])}{y :^{\varepsilon} [\mathsf{r}]A, x \,\mathsf{R}^{\mathsf{r}}_{a}\, y \Rightarrow y :^{\mathsf{r}} A}(Lw)}{x :^{\varepsilon} [\mathsf{r}(a)][\mathsf{r}]A \Rightarrow x :^{\mathsf{r}} [a]A}(R[a])}{x :^{\varepsilon} [\mathsf{r}(a)][\mathsf{r}]A \Rightarrow x :^{\varepsilon} [\mathsf{r}][a]A}(R[\mathsf{r}])}{\Rightarrow x :^{\varepsilon} [\mathsf{r}(a)][\mathsf{r}]A \to [\mathsf{r}][a]A}(R\to)$$

Case of axiom $([\mathsf{r}][\cup])$

The direction from left to right is shown as follows. First, we construct the following derivation $\mathfrak{D}_1$

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{x \,\mathsf{R}^{\mathsf{r}}_{\alpha}\, y \Rightarrow x \,\mathsf{R}^{\mathsf{r}}_{\alpha}\, y}{x \,\mathsf{R}^{\mathsf{r}}_{\alpha}\, y \Rightarrow x \,\mathsf{R}^{\mathsf{r}}_{\alpha\cup\beta}\, y}(R\cup)}{x \,\mathsf{R}^{\mathsf{r}}_{\alpha}\, y \Rightarrow y :^{\mathsf{r}} A, x \,\mathsf{R}^{\mathsf{r}}_{\alpha\cup\beta}\, y}(Rw) \quad \dfrac{y :^{\mathsf{r}} A \Rightarrow y :^{\mathsf{r}} A}{y :^{\mathsf{r}} A, x \,\mathsf{R}^{\mathsf{r}}_{\alpha}\, y \Rightarrow y :^{\mathsf{r}} A}(Lw)}{x \,\mathsf{R}^{\mathsf{r}}_{\alpha}\, y, x :^{\mathsf{r}} [\alpha \cup \beta]A \Rightarrow y :^{\mathsf{r}} A}(L[\alpha\cup\beta])}{x :^{\mathsf{r}} [\alpha \cup \beta]A \Rightarrow x :^{\mathsf{r}} [\alpha]A}(R[\alpha])}{x :^{\mathsf{r}} [\alpha \cup \beta]A \Rightarrow x :^{\varepsilon} [\mathsf{r}][\alpha]A}(R[\mathsf{r}])$$

Second, we also construct the following derivation $\mathfrak{D}_2$.

$$
\begin{prooftree}
\AxiomC{$x\,\mathsf{R}^{\mathsf{r}}_{\beta}\,z \Rightarrow x\,\mathsf{R}^{\mathsf{r}}_{\beta}\,z$}
\RightLabel{$(R\cup)$}
\UnaryInfC{$x\,\mathsf{R}^{\mathsf{r}}_{\beta}\,z \Rightarrow x\,\mathsf{R}^{\mathsf{r}}_{\alpha\cup\beta}\,z$}
\RightLabel{(Rw)}
\UnaryInfC{$x\,\mathsf{R}^{\mathsf{r}}_{\beta}\,z \Rightarrow z :^{\mathsf{r}} A, x\,\mathsf{R}^{\mathsf{r}}_{\alpha\cup\beta}\,z$}
\AxiomC{$z :^{\mathsf{r}} A \Rightarrow z :^{\mathsf{r}} A$}
\RightLabel{(Lw)}
\UnaryInfC{$z :^{\mathsf{r}} A, x\,\mathsf{R}^{\mathsf{r}}_{\beta}\,z \Rightarrow z :^{\mathsf{r}} A$}
\RightLabel{$(L[\alpha\cup\beta])$}
\BinaryInfC{$x\,\mathsf{R}^{\mathsf{r}}_{\beta}\,z, x :^{\mathsf{r}} [\alpha\cup\beta]A \Rightarrow z :^{\mathsf{r}} A$}
\RightLabel{$(R[\beta])$}
\UnaryInfC{$x :^{\mathsf{r}} [\alpha\cup\beta]A \Rightarrow x :^{\mathsf{r}} [\beta]A$}
\RightLabel{$(R[\mathsf{r}])$}
\UnaryInfC{$x :^{\mathsf{r}} [\alpha\cup\beta]A \Rightarrow x :^{\varepsilon} [\mathsf{r}][\beta]A$}
\end{prooftree}
$$

Finally, we obtain the following derivation by $\mathfrak{D}_1$ and $\mathfrak{D}_2$.

$$
\begin{prooftree}
\AxiomC{$\mathfrak{D}_1$}
\AxiomC{$\mathfrak{D}_2$}
\RightLabel{$(R\wedge)$}
\BinaryInfC{$x :^{\mathsf{r}} [\alpha\cup\beta]A \Rightarrow x :^{\varepsilon} [\mathsf{r}][\alpha]A \wedge [\mathsf{r}][\beta]A$}
\RightLabel{$(L[\mathsf{r}])$}
\UnaryInfC{$x :^{\varepsilon} [\mathsf{r}][\alpha\cup\beta]A \Rightarrow x :^{\varepsilon} [\mathsf{r}][\alpha]A \wedge [\mathsf{r}][\beta]A$}
\RightLabel{$(R\to)$}
\UnaryInfC{$\Rightarrow x :^{\varepsilon} [\mathsf{r}][\alpha\cup\beta]A \to [\mathsf{r}][\alpha]A \wedge [\mathsf{r}][\beta]A$}
\end{prooftree}
$$

The direction from right to left is shown as follows.

$$
\begin{prooftree}
\AxiomC{$x\,\mathsf{R}^{\mathsf{r}}_{\alpha}\,y \Rightarrow x\,\mathsf{R}^{\mathsf{r}}_{\alpha}\,y$}
\RightLabel{(Rw)}
\UnaryInfC{$x\,\mathsf{R}^{\mathsf{r}}_{\alpha}\,y \Rightarrow y :^{\mathsf{r}} A, x\,\mathsf{R}^{\mathsf{r}}_{\alpha}\,y$}
\AxiomC{$y :^{\mathsf{r}} A \Rightarrow y :^{\mathsf{r}} A$}
\RightLabel{(Lw)}
\UnaryInfC{$y :^{\mathsf{r}} A, x\,\mathsf{R}^{\mathsf{r}}_{\alpha}\,y \Rightarrow y :^{\mathsf{r}} A$}
\RightLabel{$(L[\alpha])$}
\BinaryInfC{$x\,\mathsf{R}^{\mathsf{r}}_{\alpha}\,y, x :^{\mathsf{r}} [\alpha]A \Rightarrow y :^{\mathsf{r}} A$}
\RightLabel{$(L[\mathsf{r}])$}
\UnaryInfC{$x\,\mathsf{R}^{\mathsf{r}}_{\alpha}\,y, x :^{\varepsilon} [\mathsf{r}][\alpha]A \Rightarrow y :^{\mathsf{r}} A$}
\RightLabel{$(L\wedge)$}
\UnaryInfC{$x\,\mathsf{R}^{\mathsf{r}}_{\alpha}\,y, x :^{\varepsilon} [\mathsf{r}][\alpha]A \wedge [\mathsf{r}][\beta]A \Rightarrow y :^{\mathsf{r}} A$}
\AxiomC{$x :^{\mathsf{r}} \beta y \Rightarrow x :^{\mathsf{r}} \beta y$}
\RightLabel{(Rw)}
\UnaryInfC{$x :^{\mathsf{r}} \beta y \Rightarrow y :^{\mathsf{r}} A, x :^{\mathsf{r}} \beta y$}
\AxiomC{$y :^{\mathsf{r}} A \Rightarrow y :^{\mathsf{r}} A$}
\RightLabel{(Lw)}
\UnaryInfC{$y :^{\mathsf{r}} A, x :^{\mathsf{r}} \beta y \Rightarrow y :^{\mathsf{r}} A$}
\RightLabel{$(L[\beta])$}
\BinaryInfC{$x :^{\mathsf{r}} \beta y, x :^{\mathsf{r}} [\beta]A \Rightarrow y :^{\mathsf{r}} A$}
\RightLabel{$(L[\mathsf{r}])$}
\UnaryInfC{$x :^{\mathsf{r}} \beta y, x :^{\varepsilon} [\mathsf{r}][\beta]A \Rightarrow y :^{\mathsf{r}} A$}
\RightLabel{$(L\wedge)$}
\UnaryInfC{$x :^{\mathsf{r}} \beta y, x :^{\varepsilon} [\mathsf{r}][\alpha]A \wedge [\mathsf{r}][\beta]A \Rightarrow y :^{\mathsf{r}} A$}
\RightLabel{$(L\cup)$}
\BinaryInfC{$x\,\mathsf{R}^{\mathsf{r}}_{\alpha\cup\beta}\,y, x :^{\varepsilon} [\mathsf{r}][\alpha]A \wedge [\mathsf{r}][\beta]A \Rightarrow y :^{\mathsf{r}} A$}
\RightLabel{$(R[\alpha\cup\beta])$}
\UnaryInfC{$x :^{\varepsilon} [\mathsf{r}][\alpha]A \wedge [\mathsf{r}][\beta]A \Rightarrow x :^{\mathsf{r}} [\alpha\cup\beta]A$}
\RightLabel{$(R[\mathsf{r}])$}
\UnaryInfC{$x :^{\varepsilon} [\mathsf{r}][\alpha]A \wedge [\mathsf{r}][\beta]A \Rightarrow x :^{\varepsilon} [\mathsf{r}][\alpha\cup\beta]A$}
\RightLabel{$(R\to)$}
\UnaryInfC{$\Rightarrow x :^{\varepsilon} [\mathsf{r}][\alpha]A \wedge [\mathsf{r}][\beta]A \to [\mathsf{r}][\alpha\cup\beta]A$}
\end{prooftree}
$$

Case of axiom $([\mathsf{r}][;])$

The direction from left to right is shown as follows.

$$
\begin{prooftree}
\AxiomC{$x\,\mathsf{R}^{\mathsf{r}}_{\alpha}\,y \Rightarrow x\,\mathsf{R}^{\mathsf{r}}_{\alpha}\,y$}
\RightLabel{(Lw)}
\UnaryInfC{$x\,\mathsf{R}^{\mathsf{r}}_{\alpha}\,y, y\,\mathsf{R}^{\mathsf{r}}_{\beta}\,z \Rightarrow x\,\mathsf{R}^{\mathsf{r}}_{\alpha}\,y$}
\AxiomC{$y\,\mathsf{R}^{\mathsf{r}}_{\beta}\,z \Rightarrow y\,\mathsf{R}^{\mathsf{r}}_{\beta}\,z$}
\RightLabel{(Lw)}
\UnaryInfC{$x\,\mathsf{R}^{\mathsf{r}}_{\alpha}\,y, y\,\mathsf{R}^{\mathsf{r}}_{\beta}\,z \Rightarrow y\,\mathsf{R}^{\mathsf{r}}_{\beta}\,z$}
\RightLabel{$(R;)$}
\BinaryInfC{$x\,\mathsf{R}^{\mathsf{r}}_{\alpha}\,y, y\,\mathsf{R}^{\mathsf{r}}_{\beta}\,z \Rightarrow x\,\mathsf{R}^{\mathsf{r}}_{\alpha;\beta}\,z$}
\RightLabel{(Rw)}
\UnaryInfC{$x\,\mathsf{R}^{\mathsf{r}}_{\alpha}\,y, y\,\mathsf{R}^{\mathsf{r}}_{\beta}\,z \Rightarrow z :^{\mathsf{r}} A, x\,\mathsf{R}^{\mathsf{r}}_{\alpha;\beta}\,z$}
\AxiomC{$z\,\mathsf{R}^{\mathsf{r}}_{A} \Rightarrow z\,\mathsf{R}^{\mathsf{r}}_{A}$}
\doubleLine
\RightLabel{(Lw)}
\UnaryInfC{$z\,\mathsf{R}^{\mathsf{r}}_{A}, x\,\mathsf{R}^{\mathsf{r}}_{\alpha}\,y, y\,\mathsf{R}^{\mathsf{r}}_{\beta}\,z \Rightarrow z\,\mathsf{R}^{\mathsf{r}}_{A}$}
\RightLabel{$(L[\alpha;\beta])$}
\BinaryInfC{$x\,\mathsf{R}^{\mathsf{r}}_{\alpha}\,y, y\,\mathsf{R}^{\mathsf{r}}_{\beta}\,z, x :^{\mathsf{r}} [\alpha;\beta]A \Rightarrow z :^{\mathsf{r}} A$}
\RightLabel{$(R[\beta])$}
\UnaryInfC{$x\,\mathsf{R}^{\mathsf{r}}_{\alpha}\,y, x :^{\mathsf{r}} [\alpha;\beta]A \Rightarrow y :^{\mathsf{r}} [\beta]A$}
\RightLabel{$(R[\alpha])$}
\UnaryInfC{$x :^{\mathsf{r}} [\alpha;\beta]A \Rightarrow x :^{\mathsf{r}} [\alpha][\beta]A$}
\doubleLine
\RightLabel{$([\mathsf{r}])$}
\UnaryInfC{$x :^{\varepsilon} [\mathsf{r}][\alpha;\beta]A \Rightarrow x :^{\varepsilon} [\mathsf{r}][\alpha][\beta]A$}
\RightLabel{$(R\to)$}
\UnaryInfC{$\Rightarrow x :^{\varepsilon} [\mathsf{r}][\alpha;\beta]A \to [\mathsf{r}][\alpha][\beta]A$}
\end{prooftree}
$$

The direction from right to left is shown as follows.

$$
\begin{prooftree}
\AxiomC{$x \, \mathsf{R}^{\mathsf{r}}_{\alpha} \, z \Rightarrow x \, \mathsf{R}^{\mathsf{r}}_{\alpha} \, z$}
\doubleLine
\RightLabel{(w)}
\UnaryInfC{$x \, \mathsf{R}^{\mathsf{r}}_{\alpha} \, z, z \, \mathsf{R}^{\mathsf{r}}_{\beta} \, y \Rightarrow y :^{\mathsf{r}} A, x \, \mathsf{R}^{\mathsf{r}}_{\alpha} \, z$}
\AxiomC{$z \, \mathsf{R}^{\mathsf{r}}_{\beta} \, y \Rightarrow z \, \mathsf{R}^{\mathsf{r}}_{\beta} \, y$}
\RightLabel{(Rw)}
\UnaryInfC{$z \, \mathsf{R}^{\mathsf{r}}_{\beta} \, y \Rightarrow y :^{\mathsf{r}} A, z \, \mathsf{R}^{\mathsf{r}}_{\beta} \, y$}
\AxiomC{$y :^{\mathsf{r}} A \Rightarrow y :^{\mathsf{r}} A$}
\RightLabel{(Lw)}
\UnaryInfC{$y :^{\mathsf{r}} A, z :^{\mathsf{r}} [\beta]A \Rightarrow y :^{\mathsf{r}} A$}
\RightLabel{$(L[\beta])$}
\BinaryInfC{$z :^{\mathsf{r}} [\beta]A, z \, \mathsf{R}^{\mathsf{r}}_{\beta} \, y \Rightarrow y :^{\mathsf{r}} A$}
\RightLabel{(Lw)}
\UnaryInfC{$z :^{\mathsf{r}} [\beta]A, x \, \mathsf{R}^{\mathsf{r}}_{\alpha} \, z, z \, \mathsf{R}^{\mathsf{r}}_{\beta} \, y \Rightarrow y :^{\mathsf{r}} A$}
\RightLabel{$L[\alpha]$}
\BinaryInfC{$x \, \mathsf{R}^{\mathsf{r}}_{\alpha} \, z, z \, \mathsf{R}^{\mathsf{r}}_{\beta} \, y, x :^{\mathsf{r}} [\alpha][\beta]A \Rightarrow y :^{\mathsf{r}} A$}
\RightLabel{$(L;)$}
\UnaryInfC{$x \, \mathsf{R}^{\mathsf{r}}_{\alpha;\beta} \, y, x :^{\mathsf{r}} [\alpha][\beta]A \Rightarrow y :^{\mathsf{r}} A$}
\RightLabel{$(R[\alpha;\beta])$}
\UnaryInfC{$x :^{\mathsf{r}} [\alpha][\beta]A \Rightarrow x :^{\mathsf{r}} [\alpha;\beta]A$}
\doubleLine
\RightLabel{$([\mathsf{r}])$}
\UnaryInfC{$x :^{\varepsilon} [\mathsf{r}][\alpha][\beta]A \Rightarrow x :^{\varepsilon} [\mathsf{r}][\alpha;\beta]A$}
\RightLabel{$(R\to)$}
\UnaryInfC{$\Rightarrow x :^{\varepsilon} [\mathsf{r}][\alpha][\beta]A \to [\mathsf{r}][\alpha;\beta]A$}
\end{prooftree}
$$

Case of axiom $([\mathsf{r}][?])$

The direction from left to right is shown as follows.

$$
\begin{prooftree}
\AxiomC{$x = x \Rightarrow x = x$}
\RightLabel{$(=Ref)$}
\UnaryInfC{$\Rightarrow x = x$}
\RightLabel{(Lw)}
\UnaryInfC{$x :^{\mathsf{r}} B \Rightarrow x :^{\mathsf{r}} A, x = x$}
\AxiomC{$x :^{\mathsf{r}} B \Rightarrow x :^{\mathsf{r}} B$}
\RightLabel{(Rw)}
\UnaryInfC{$x :^{\mathsf{r}} B \Rightarrow x :^{\mathsf{r}} A, x :^{\mathsf{r}} B$}
\RightLabel{$(R?)$}
\BinaryInfC{$x :^{\mathsf{r}} B \Rightarrow x :^{\mathsf{r}} A, x \, \mathsf{R}^{\mathsf{r}}_{?B} \, x$}
\AxiomC{$x :^{\mathsf{r}} A \Rightarrow x :^{\mathsf{r}} A$}
\RightLabel{(Lw)}
\UnaryInfC{$x :^{\mathsf{r}} A, x :^{\mathsf{r}} B \Rightarrow x :^{\mathsf{r}} A$}
\RightLabel{$(L[?B])$}
\BinaryInfC{$x :^{\mathsf{r}} B, x :^{\mathsf{r}} [?B]A \Rightarrow x :^{\mathsf{r}} A$}
\RightLabel{$(R\to)$}
\UnaryInfC{$x :^{\mathsf{r}} [?B]A \Rightarrow x :^{\mathsf{r}} B \to A$}
\doubleLine
\RightLabel{$([\mathsf{r}])$}
\UnaryInfC{$x :^{\varepsilon} [\mathsf{r}][?B]A \Rightarrow x :^{\varepsilon} [\mathsf{r}](B \to A)$}
\RightLabel{$(R\to)$}
\UnaryInfC{$\Rightarrow x :^{\varepsilon} [\mathsf{r}][?B]A \to [\mathsf{r}](B \to A)$}
\end{prooftree}
$$

The direction from right to left is shown as follows.

$$
\begin{prooftree}
\AxiomC{$x :^{\mathsf{r}} B \Rightarrow x :^{\mathsf{r}} B$}
\RightLabel{(Rw)}
\UnaryInfC{$x :^{\mathsf{r}} B \Rightarrow y :^{\mathsf{r}} A, x :^{\mathsf{r}} B$}
\RightLabel{$(L?_2)$}
\UnaryInfC{$x \, \mathsf{R}^{\mathsf{r}}_{?B} \, y \Rightarrow y :^{\mathsf{r}} A, x :^{\mathsf{r}} B$}
\AxiomC{Lemma 4.10}
\noLine
\UnaryInfC{$\vdots$}
\noLine
\UnaryInfC{$x :^{\mathsf{r}} A, x = y \Rightarrow y :^{\mathsf{r}} A$}
\RightLabel{$(L?_1)$}
\UnaryInfC{$x :^{\mathsf{r}} A, x \, \mathsf{R}^{\mathsf{r}}_{?B} \, y \Rightarrow y :^{\mathsf{r}} A$}
\RightLabel{$(L\to)$}
\BinaryInfC{$x \, \mathsf{R}^{\mathsf{r}}_{?B} \, y, x :^{\mathsf{r}} B \to A \Rightarrow y :^{\mathsf{r}} A$}
\RightLabel{$(R[?B])$}
\UnaryInfC{$x :^{\mathsf{r}} B \to A \Rightarrow x :^{\mathsf{r}} [?B]A$}
\doubleLine
\RightLabel{$([\mathsf{r}])$}
\UnaryInfC{$x :^{\varepsilon} [\mathsf{r}](B \to A) \Rightarrow x :^{\varepsilon} [\mathsf{r}][?B]A$}
\RightLabel{$(R\to)$}
\UnaryInfC{$\Rightarrow x :^{\varepsilon} [\mathsf{r}](B \to A) \to [\mathsf{r}][?B]A$}
\end{prooftree}
$$

In the inductive step, we show the cases of inference rules in HDLRC. Here, we show only the case of inference rules (**MP**) and ($\mathbf{Nec}_{[\mathsf{r}]}$).

Case of inference rule (MP):

Fix any $x \in \mathsf{Var}$. Suppose derivations $\mathfrak{D}_1$ for $\Rightarrow x :^{\varepsilon} A \to B$ and $\mathfrak{D}_2$ for $\Rightarrow x :^{\varepsilon} A$. We show that there is a derivation of $\Rightarrow x :^{\varepsilon} B$. This is shown by:

$$
\begin{prooftree}
\AxiomC{$\vdots \, \mathfrak{D}_1$}
\noLine
\UnaryInfC{$\Rightarrow x :^{\varepsilon} A \to B$}
\AxiomC{$x :^{\varepsilon} B \Rightarrow x :^{\varepsilon} B$}
\AxiomC{$\vdots \, \mathfrak{D}_2$}
\noLine
\UnaryInfC{$\Rightarrow x :^{\varepsilon} A$}
\RightLabel{$(L\to)$}
\BinaryInfC{$x :^{\varepsilon} A \to B \Rightarrow x :^{\varepsilon} B$}
\RightLabel{(Cut)}
\BinaryInfC{$\Rightarrow x :^{\varepsilon} B$}
\end{prooftree}
$$

Case of inference rule $(\mathbf{Nec}_{[\mathsf{r}]})$**:**

Fix any $x \in \mathsf{Var}$ and any $\mathsf{r} \in \mathsf{RC}$. Suppose a derivation $\mathfrak{D}$ of $\Rightarrow x :^{\varepsilon} A$. Our goal is to show the derivation $\mathfrak{D}'$ of $\Rightarrow x :^{\varepsilon} [\mathsf{r}]A$. Using rule $(R[\mathsf{r}])$ in GDLRC we can obtain the derivation $\mathfrak{D}'$ by $\mathfrak{D}$ where r is inserted into the head of the list of relation changers of each node in $\mathfrak{D}$.

□

4.3.3 Cut Elimination of GDLRC

Definition 4.12. *Let $w, x, y, z \in \mathsf{Var}$, $A \in \mathsf{Form}$, $\alpha \in \mathsf{PR}$, $L \in (\mathsf{RC})^*$ and φ be a sequent. A substitution $\varphi[y/x]$ (the result of substituting x in φ uniformly with y) is defined by:*

$$\begin{array}{rcl} z[y/x] & \equiv & \begin{cases} y & \text{if } z = x \\ z & \text{if } z \neq x \end{cases} \\ (z :^{L} A)[y/x] & \equiv & z[y/x] :^{L} A \\ (w\, \mathsf{R}^{L}_{\alpha}\, z)[y/x] & \equiv & w[y/x]\, \mathsf{R}^{L}_{\alpha}\, z[y/x] \\ (w = z)[y/x] & \equiv & w[y/x](= z[y/x]) \end{array}$$

A substitution $\Gamma[y/x]$ is also defined by:

$$\Gamma[y/x] \quad = \quad \{\, \varphi[y/x] \mid \varphi \in \Gamma \,\}.$$

Lemma 4.13. *If $\vdash_{\mathbf{GDLRC}} \Gamma \Rightarrow \Delta$ by a derivation $\mathfrak{D}$, then $\vdash_{\mathbf{GDLRC}} \Gamma[y/x] \Rightarrow \Delta[y/x]$ by a derivation $\mathfrak{D}'$ where the height of $\mathfrak{D}$ is equal to the height of $\mathfrak{D}'$.*

Definition 4.14 (Extended cut rule)**.** *Let $m, n \geq 0$ and φ be a labelled expression.*

$$\frac{\Gamma \Rightarrow \Delta, \varphi^{m} \quad \varphi^{n}, \Sigma \Rightarrow \Pi}{\Gamma, \Sigma \Rightarrow \Delta, \Pi}\ (Ecut)$$

We say that φ is a Ecut labelled expression*.*

We define GDLRC^{*} as GDLRC^{-} with the rule of $(Ecut)$.

Definition 4.15 (Ecut-bottom form for GDLRC^{*})**.** *A derivation $\mathfrak{D}$ in GDLRC^{*} is Ecut-bottom form if it has the following form:*

$$\dfrac{\dfrac{\vdots\, \mathfrak{D}_L}{\Gamma \Rightarrow \Delta, \varphi^{m}}\ \mathtt{rule}(\mathfrak{D}_L) \quad \dfrac{\vdots\, \mathfrak{D}_R}{\varphi^{n}, \Sigma \Rightarrow \Pi}\ \mathtt{rule}(\mathfrak{D}_R)}{\Gamma, \Sigma \Rightarrow \Delta, \Pi}\ (Ecut)$$

where $\mathtt{rule}(\mathfrak{D})$ is the last applied rule of a given derivation $\mathfrak{D}$ and there are no-application of $(Ecut)$ in $\mathfrak{D}_L$ nor $\mathfrak{D}_R$.

Definition 4.16. *Let $\mathfrak{D}$ be a derivation in GDLRC^{*} who has the Ecut-bottom form. The weight and the complexity of $\mathfrak{D}$ is defined by:*

$$\begin{array}{rcl} w(\mathfrak{D}) & = & |\mathfrak{D}_L| + |\mathfrak{D}_R|\ (\geq 2) \text{ where } |\cdot| \text{ is the number of nodes as a tree,} \\ c(\mathfrak{D}) & = & \ell(\varphi) \text{ where } \varphi \text{ is the labelled expressions in } \mathfrak{D}. \end{array}$$

Definition 4.17. *Let $c(\mathfrak{D}) = i$ and $w(\mathfrak{D}) = j$.*

$$(i, j) \geq (i', j') \quad \text{iff} \quad i = i' \text{ and } j \geq j'.$$

Lemma 4.18. *For any Ecut-bottom form $\mathfrak{D}$ in* **GDLRC****, there is a derivation $\mathfrak{D}'$ in* **GDLRC**$^-$ *such that the end sequent of $\mathfrak{D}$ is equal to that of $\mathfrak{D}'$.*

Proof. The proof is shown by the method of Ono and Komori [29]. Since $\mathfrak{D}$ is the Ecut-bottom form in **GDLRC***, it has the following form:

$$\dfrac{\dfrac{\vdots\, \mathfrak{D}_L}{\Gamma \Rightarrow \Delta, \varphi^m}\,\mathtt{rule}(\mathfrak{D}_L) \quad \dfrac{\vdots\, \mathfrak{D}_R}{\varphi^n, \Sigma \Rightarrow \Pi}\,\mathtt{rule}(\mathfrak{D}_R)}{\Gamma, \Sigma \Rightarrow \Delta, \Pi}\,(Ecut)$$

Let $c(\mathfrak{D}) = i$ (≥ 0) and $w(\mathfrak{D}) = j$ (≥ 2). We show the proof by double induction on $(c(\mathfrak{D}), w(\mathfrak{D}))$. Our goal is to obtain a derivation $\mathfrak{D}'$ in **GDLRC**$^-$ such that the end sequent of $\mathfrak{D}$ is equal to that of $\mathfrak{D}'$. Our proof is organized as follows: first, we consider the cases where $m = 0$ (for φ^m) or $n = 0$ (for φ^n) in $\mathfrak{D}$. Since we can easily show such cases by (Lw) and (Rw), we skip them. Second, we consider the cases where $m > 0$ and $n > 0$ in $\mathfrak{D}$. These cases can be divided into the following five sub cases.

(1) At least one of $\mathtt{rule}(\mathfrak{D}_L)$ and $\mathtt{rule}(\mathfrak{D}_R)$ is an initial sequent.

(2) At least one of $\mathtt{rule}(\mathfrak{D}_L)$ and $\mathtt{rule}(\mathfrak{D}_R)$ is a structural rule.

(3) At least one of $\mathtt{rule}(\mathfrak{D}_L)$ and $\mathtt{rule}(\mathfrak{D}_R)$ is a logical rule, a program rule, or a relation changer rule where the cut labelled expression is not principal.

(4) Both $\mathtt{rule}(\mathfrak{D}_L)$ and $\mathtt{rule}(\mathfrak{D}_R)$ are logical rules, program rules, or relation changer rules for the same connectives where the cut labelled expression is principal for both rules.

(5) At least one of $\mathtt{rule}(\mathfrak{D}_L)$ and $\mathtt{rule}(\mathfrak{D}_R)$ is an equality rule where the cut labelled expression is not principal.

For case (5), the equality rule $(= at)$ can be regarded as a right rule, but we do not have any corresponding left rule for that. In addition, there is no principal labelled expression in the other equality rules. Therefore we do not need to consider the case of 'both $\mathtt{rule}(\mathfrak{D}_L)$ and $\mathtt{rule}(\mathfrak{D}_R)$ are equality rules where the cut labelled expression is principal for both rules.'

Now, let us see the proof for each cases. We skip cases (1) and (2) since they are rather easy.

Case (3):

Suppose that at least one of $\mathtt{rule}(\mathfrak{D}_L)$ and $\mathtt{rule}(\mathfrak{D}_R)$ is a logical rule, a program rule, or a relation changer rule where the cut labelled expression is not principal. Here, we only show the sub cases for the logical rule $(R[\mathsf{r}])$ and the relation changer rules (Rat) and $(Rrel)$.

Case (3-1) where $\texttt{rule}(\mathfrak{D}_L)$ is $(R[\mathsf{r}])$:

The Ecut-bottom form of this case can be written as follows:

$$\dfrac{\dfrac{\overset{\vdots\, \mathfrak{D}_{L'}}{\Gamma \Rightarrow \Delta, x :^{(L,r)} A, \varphi^m}}{\Gamma \Rightarrow \Delta, x :^{L} [\mathsf{r}]A, \varphi^m}\,(R[\mathsf{r}]) \quad \overset{\vdots\, \mathfrak{D}_R}{\varphi^n, \Sigma \Rightarrow \Pi}}{\Gamma, \Sigma \Rightarrow \Delta, \Pi, x :^{L} [\mathsf{r}]A}\,(Ecut)$$

We can transform the above derivation into the following one:

$$\dfrac{\dfrac{\overset{\vdots\, \mathfrak{D}_{L'}}{\Gamma \Rightarrow \Delta, x :^{(L,r)} A, \varphi^m} \quad \overset{\vdots\, \mathfrak{D}_R}{\varphi^n, \Sigma \Rightarrow \Pi}}{\Gamma, \Sigma \Rightarrow \Delta, \Pi, x :^{(L,r)} A}\,(Ecut)}{\Gamma, \Sigma \Rightarrow \Delta, \Pi, x :^{L} [\mathsf{r}]A}\,(R[\mathsf{r}])$$

Since $\ell(x :^{L} [\mathsf{r}]A) > \ell(x :^{(L,r)} A)$ (cf. Proposition 4.9 (i)), the complexity of the above transformed derivation is less than that of the original one. Therefore, we can obtain a derivation for $\Gamma, \Sigma \Rightarrow \Delta, \Pi, x :^{L} [\mathsf{r}]A$ in $\mathbf{GDLRC}^-$ by induction hypothesis.

Case (3-2) where $\texttt{rule}(\mathfrak{D}_L)$ is (Rat):

The Ecut-bottom form of this case can be written as follows:

$$\dfrac{\dfrac{\overset{\vdots\, \mathfrak{D}_{L'}}{\Gamma \Rightarrow \Delta, x :^{L} p, \varphi^m}}{\Gamma \Rightarrow \Delta, x :^{(L,r)} p, \varphi^m}\,(Rat) \quad \overset{\vdots\, \mathfrak{D}_R}{\varphi^n, \Sigma \Rightarrow \Pi}}{\Gamma, \Sigma \Rightarrow \Delta, \Pi, x :^{(L,r)} p}\,(Ecut)$$

We can transform the above derivation into the following one:

$$\dfrac{\dfrac{\overset{\vdots\, \mathfrak{D}_{L'}}{\Gamma \Rightarrow \Delta, x :^{L} p, \varphi^m} \quad \overset{\vdots\, \mathfrak{D}_R}{\varphi^n, \Sigma \Rightarrow \Pi}}{\Gamma, \Sigma \Rightarrow \Delta, \Pi, x :^{L} p}\,(Ecut)}{\Gamma, \Sigma \Rightarrow \Delta, \Pi, x :^{(L,r)} p}\,(Rat)$$

Since $\ell(x :^{(L,r)} p) > \ell(x :^{L} p)$ (cf. Proposition 4.9 (ii)), the complexity of the above transformed derivation is less than that of the original one. Therefore, we can obtain a derivation for $\Gamma, \Sigma \Rightarrow \Delta, \Pi, x :^{(L,r)} p$ in $\mathbf{GDLRC}^-$ by induction hypothesis.

Case (3-3) where $\texttt{rule}(\mathfrak{D}_L)$ is $(Rrel)$:

The Ecut-bottom form of this case can be written as follows:

$$\dfrac{\dfrac{\overset{\vdots\, \mathfrak{D}_{L'}}{\Gamma \Rightarrow \Delta, x\, \mathsf{R}^{L}_{\mathsf{r}(a)}\, y, \varphi^m}}{\Gamma \Rightarrow \Delta, x\, \mathsf{R}^{(L,r)}_{a}\, y, \varphi^m}\,(Rrel) \quad \overset{\vdots\, \mathfrak{D}_R}{\varphi^n, \Sigma \Rightarrow \Pi}}{\Gamma, \Sigma \Rightarrow \Delta, \Pi, x\, \mathsf{R}^{(L,r)a}_{y}}\,(Ecut)$$

We can transform the above derivation into the following one:

$$\dfrac{\dfrac{\begin{array}{c}\vdots\,\mathfrak{D}_{L'}\\ \Gamma \Rightarrow \Delta, x\,\mathsf{R}^{L}_{\mathsf{r}(a)}\,y, \varphi^{m}\end{array} \quad \begin{array}{c}\vdots\,\mathfrak{D}_{R}\\ \varphi^{n}, \Sigma \Rightarrow \Pi\end{array}}{\Gamma, \Sigma \Rightarrow \Delta, \Pi, x\,\mathsf{R}^{L}_{\mathsf{r}(a)}\,y}\,(Ecut)}{\Gamma, \Sigma \Rightarrow \Delta, \Pi, x\,\mathsf{R}^{(L,r)}_{a}\,y}\,(Rrel)$$

Since $\ell(x\,\mathsf{R}^{(L,r)}_{a}\,y) > \ell(x\,\mathsf{R}^{\mathsf{r}}_{\mathsf{r}(a)}\,y)$ (cf. Proposition 4.9 (iii)), the complexity of the above transformed derivation is less than that of the original one. Therefore, we can obtain a derivation for $\Gamma, \Sigma \Rightarrow \Delta, \Pi, x\,\mathsf{R}^{(L,r)a}_{y}$ in GDLRC^{-} by induction hypothesis.

Case (4):

Suppose that both $\texttt{rule}(\mathfrak{D}_L)$ and $\texttt{rule}(\mathfrak{D}_R)$ are logical rules, program rules, or relation changer rules for the same connectives where the cut labelled expression is principal for both rules. Here, we only show the sub cases for the logical rules of $([\mathsf{r}])$ and relation changer rules of (rel) and (at).

Case (4-1) where $\texttt{rule}(\mathfrak{D}_L)$ is $(R[\mathsf{r}])$ and $\texttt{rule}(\mathfrak{D}_R)$ is $(L[\mathsf{r}])$:

The Ecut-bottom form of this case can be written as follows:

$$\dfrac{\dfrac{\begin{array}{c}\vdots\,\mathfrak{D}_{L'}\\ \Gamma \Rightarrow \Delta, x :^{(L,r)} A, (x :^{L} [\mathsf{r}]A)^{m-1}\end{array}}{\Gamma \Rightarrow \Delta, (x :^{L} [\mathsf{r}]A)^{m}}\,(R[\mathsf{r}]) \quad \dfrac{\begin{array}{c}\vdots\,\mathfrak{D}_{R'}\\ (x :^{L} [\mathsf{r}]A)^{n-1}, x :^{(L,r)} A, \Sigma \Rightarrow \Pi\end{array}}{(x :^{L} [\mathsf{r}]A)^{n}, \Sigma \Rightarrow \Pi}\,(L[\mathsf{r}])}{\Gamma, \Sigma \Rightarrow \Delta, \Pi}\,(Ecut)$$

From this derivation, we can construct the following derivation $\mathfrak{D}'_1$:

$$\dfrac{\begin{array}{c}\vdots\,\mathfrak{D}_{L'}\\ \Gamma \Rightarrow \Delta, x :^{(L,r)} A, (x :^{L} [\mathsf{r}]A)^{m-1}\end{array} \quad \dfrac{\begin{array}{c}\vdots\,\mathfrak{D}_{R'}\\ (x :^{L} [\mathsf{r}]A)^{n-1}, x :^{(L,r)} A, \Sigma \Rightarrow \Pi\end{array}}{(x :^{L} [\mathsf{r}]A)^{n}, \Sigma \Rightarrow \Pi}\,(L[\mathsf{r}])}{\Gamma, \Sigma \Rightarrow \Delta, \Pi, x :^{(L,r)} A}\,(Ecut)$$

Similarly, we can also construct the following derivation $\mathfrak{D}'_2$:

$$\dfrac{\dfrac{\begin{array}{c}\vdots\,\mathfrak{D}_{L'}\\ \Gamma \Rightarrow \Delta, x :^{(L,r)} A, (x :^{L} [\mathsf{r}]A)^{m-1}\end{array}}{\Gamma \Rightarrow \Delta, (x :^{L} [\mathsf{r}]A)^{m}}\,(R[\mathsf{r}]) \quad \begin{array}{c}\vdots\,\mathfrak{D}_{R'}\\ (x :^{L} [\mathsf{r}]A)^{n-1}, x :^{(L,r)} A, \Sigma \Rightarrow \Pi\end{array}}{x :^{(L,r)} A, \Gamma, \Sigma \Rightarrow \Delta, \Pi}\,(Ecut)$$

Since the weight of the Ecut-bottom form in $\mathfrak{D}'_1$ and $\mathfrak{D}'_2$ are smaller than the original one, we can obtain derivations for $\Gamma, \Sigma \Rightarrow \Delta, \Pi, x :^{(L,r)} A$ and $x :^{(L,r)} A, \Gamma, \Sigma \Rightarrow \Delta, \Pi$ in GDLRC^{-} by induction hypothesis. Then, by $\mathfrak{D}'_1$ and $\mathfrak{D}'_2$, we can transform the original derivation into the following one:

$$\dfrac{\dfrac{\mathfrak{D}'_1 \quad \mathfrak{D}'_2}{\Gamma, \Gamma, \Sigma, \Sigma \Rightarrow \Delta, \Delta, \Pi, \Pi}\,(Ecut)}{\Gamma, \Sigma \Rightarrow \Delta, \Pi}\,(c)$$

Since $\ell(x :^{L} [\mathsf{r}]A) > \ell(x :^{(L,r)} A)$ (cf. Proposition 4.9 (i)), the complexity of the above transformed derivation is less than that of the original one. Therefore, we can obtain a derivation for $\Gamma, \Sigma \Rightarrow \Delta, \Pi$ in $\mathbf{GDLRC}^{-}$ by induction hypothesis.

Case (4-2) where $\texttt{rule}(\mathfrak{D}_L)$ is (Rat) and $\texttt{rule}(\mathfrak{D}_R)$ is (Lat)):
The Ecut-bottom form of this case can be written as follows:

$$\dfrac{\dfrac{\begin{matrix}\vdots\ \mathfrak{D}_{L'}\\ \Gamma \Rightarrow \Delta, x :^{L} p, (x :^{(L,r)} p)^{m-1}\end{matrix}}{\Gamma \Rightarrow \Delta, (x :^{(L,r)} p)^{m}}\,(Rat) \quad \dfrac{\begin{matrix}\vdots\ \mathfrak{D}_{R'}\\ (x :^{(L,r)} p)^{n-1}, x :^{L} p, \Sigma \Rightarrow \Pi\end{matrix}}{(x :^{(L,r)} p)^{n}, \Sigma \Rightarrow \Pi}\,(Lat)}{\Gamma, \Sigma \Rightarrow \Delta, \Pi}\,(Ecut)$$

From this derivation, we can construct the following derivation $\mathfrak{D}'_1$:

$$\dfrac{\begin{matrix}\vdots\ \mathfrak{D}_{L'}\\ \Gamma \Rightarrow \Delta, x :^{L} p, (x :^{(L,r)} p)^{m-1}\end{matrix} \quad \dfrac{\begin{matrix}\vdots\ \mathfrak{D}_{R'}\\ (x :^{(L,r)} p)^{n-1}, x :^{L} p, \Sigma \Rightarrow \Pi\end{matrix}}{(x :^{(L,r)} p)^{n}, \Sigma \Rightarrow \Pi}\,(Lat)}{\Gamma, \Sigma \Rightarrow \Delta, \Pi, x :^{L} p}\,(Ecut)$$

Similarly, we can also construct the following derivation $\mathfrak{D}'_2$:

$$\dfrac{\dfrac{\begin{matrix}\vdots\ \mathfrak{D}_{L'}\\ \Gamma \Rightarrow \Delta, x :^{L} p, (x :^{(L,r)} p)^{m-1}\end{matrix}}{\Gamma \Rightarrow \Delta, (x :^{(L,r)} p)^{m}}\,(Rat) \quad \begin{matrix}\vdots\ \mathfrak{D}_{R'}\\ (x :^{(L,r)} p)^{n-1}, x :^{L} p, \Sigma \Rightarrow \Pi\end{matrix}}{x :^{L} p, \Gamma, \Sigma \Rightarrow \Delta, \Pi}\,(Ecut)$$

Since the weight of the Ecut-bottom form in $\mathfrak{D}'_1$ and $\mathfrak{D}'_2$ are smaller than the original one, we can obtain derivations for $\Gamma, \Sigma \Rightarrow \Delta, \Pi, x :^{L} p$ and $x :^{L} p, \Gamma, \Sigma \Rightarrow \Delta, \Pi$ in $\mathbf{GDLRC}^{-}$ by induction hypothesis. Then, by $\mathfrak{D}'_1$ and $\mathfrak{D}'_2$, we can transform the original derivation into the following one:

$$\dfrac{\dfrac{\mathfrak{D}'_1 \quad \mathfrak{D}'_2}{\Gamma, \Gamma, \Sigma, \Sigma \Rightarrow \Delta, \Delta, \Pi, \Pi}\,(Ecut)}{\Gamma, \Sigma \Rightarrow \Delta, \Pi}\,(c)$$

Since $\ell(x :^{(L,r)} p) > \ell(x :^{L} p)$ (cf. Proposition 4.9 (ii)), the complexity of the above transformed derivation is less than that of the original one. Therefore, we can obtain a derivation for $\Gamma, \Sigma \Rightarrow \Delta, \Pi$ in $\mathbf{GDLRC}^{-}$ by induction hypothesis.

Case (4-3) where $\texttt{rule}(\mathfrak{D}_L)$ is $(Rrel)$ and $\texttt{rule}(\mathfrak{D}_R)$ is $(Lrel)$:
The Ecut-bottom form of this case can be written as follows:

$$\dfrac{\dfrac{\begin{matrix}\vdots\ \mathfrak{D}_{L'}\\ \Gamma \Rightarrow \Delta, x\, \mathsf{R}^{L}_{\mathsf{r}(a)}\, y, (x\, \mathsf{R}^{(L,r)}_{a}\, y)^{m-1}\end{matrix}}{\Gamma \Rightarrow \Delta, (x\, \mathsf{R}^{(L,r)}_{a}\, y)^{m}}\,(Rrel) \quad \dfrac{\begin{matrix}\vdots\ \mathfrak{D}_{R'}\\ (x\, \mathsf{R}^{(L,r)}_{a}\, y)^{n-1}, x\, \mathsf{R}^{L}_{\mathsf{r}(a)}\, y, \Sigma \Rightarrow \Pi\end{matrix}}{(x\, \mathsf{R}^{(L,r)}_{a}\, y)^{n}, \Sigma \Rightarrow \Pi}\,(Lrel)}{\Gamma, \Sigma \Rightarrow \Delta, \Pi}\,(Ecut)$$

From this derivation, we can construct the following derivation $\mathfrak{D}'_1$:

$$\dfrac{\dfrac{\vdots\ \mathfrak{D}_{L'}}{\Gamma \Rightarrow \Delta, x\,\mathsf{R}^L_{\mathsf{r}(a)}\,y, (x\,\mathsf{R}^{(L,r)}_a\,y)^{m-1}} \quad \dfrac{\dfrac{\vdots\ \mathfrak{D}_{R'}}{(x\,\mathsf{R}^{(L,r)}_a\,y)^{n-1}, x\,\mathsf{R}^L_{\mathsf{r}(a)}\,y, \Sigma \Rightarrow \Pi}}{(x\,\mathsf{R}^{(L,r)}_a\,y)^{n}, \Sigma \Rightarrow \Pi}\,(Lrel)}{\Gamma, \Sigma \Rightarrow \Delta, \Pi, x\,\mathsf{R}^L_{\mathsf{r}(a)}\,y}\,(Ecut)$$

Similarly, we can also construct the following derivation $\mathfrak{D}'_2$:

$$\dfrac{\dfrac{\dfrac{\vdots\ \mathfrak{D}_{L'}}{\Gamma \Rightarrow \Delta, x\,\mathsf{R}^L_{\mathsf{r}(a)}\,y, (x\,\mathsf{R}^{(L,r)}_a\,y)^{m-1}}}{\Gamma \Rightarrow \Delta, (x\,\mathsf{R}^{(L,r)}_a\,y)^{m}}\,(Rrel) \quad \dfrac{\vdots\ \mathfrak{D}_{R'}}{(x\,\mathsf{R}^{(L,r)}_a\,y)^{n-1}, x\,\mathsf{R}^L_{\mathsf{r}(a)}\,y, \Sigma \Rightarrow \Pi}}{x\,\mathsf{R}^L_{\mathsf{r}(a)}\,y, \Gamma, \Sigma \Rightarrow \Delta, \Pi}\,(Ecut)$$

Since the weight of the Ecut-bottom form in $\mathfrak{D}'_1$ and $\mathfrak{D}'_2$ are smaller than the original one, we can obtain derivations for $\Gamma, \Sigma \Rightarrow \Delta, \Pi, x\,\mathsf{R}^L_{\mathsf{r}(a)}\,y$ and $x\,\mathsf{R}^L_{\mathsf{r}(a)}\,y, \Gamma, \Sigma \Rightarrow \Delta, \Pi$ in $\mathbf{GDLRC}^-$ by induction hypothesis. Then, by $\mathfrak{D}'_1$ and $\mathfrak{D}'_2$, we can transform the original derivation into the following one:

$$\dfrac{\dfrac{\mathfrak{D}'_1 \quad \mathfrak{D}'_2}{\Gamma, \Gamma, \Sigma, \Sigma \Rightarrow \Delta, \Delta, \Pi, \Pi}\,(Ecut)}{\Gamma, \Sigma \Rightarrow \Delta, \Pi}\,(c)$$

Since $\ell(x\,\mathsf{R}^{(L,r)}_a\,y) > \ell(x\,\mathsf{R}^{\mathsf{r}}_{\mathsf{r}(a)}\,y)$ (cf. Proposition 4.9 (iii)), the complexity of the above transformed derivation is less than that of the original one. Therefore, we can obtain a derivation for $\Gamma, \Sigma \Rightarrow \Delta, \Pi$ in $\mathbf{GDLRC}^-$ by induction hypothesis.

Case (5):

Suppose that at least one of $\mathtt{rule}(\mathfrak{D}_L)$ and $\mathtt{rule}(\mathfrak{D}_R)$ is an equality rule where the cut labelled expression is not principal. Here, we only show the sub cases for equality rules $(= at)$ and $(= ref)$.

Case (5-1) where $\mathtt{rule}(\mathfrak{D}_L)$ is $(= at)$:

The Ecut-bottom form can be written as follows:

$$\dfrac{\dfrac{\dfrac{\vdots\ \mathfrak{D}_{L'_1}}{\Gamma \Rightarrow \Delta, x = y, \varphi^m} \quad \dfrac{\vdots\ \mathfrak{D}_{L'_2}}{\Gamma \Rightarrow \Delta, x :^L p, \varphi^m}}{\Gamma \Rightarrow \Delta, y :^L p, \varphi^m}\,(= at) \quad \dfrac{\vdots\ \mathfrak{D}_{R}}{\varphi^n, \Sigma \Rightarrow \Pi}}{\Gamma, \Sigma \Rightarrow \Delta, \Pi, y :^L p}\,(Ecut)$$

From this derivation, we can construct the following derivation $\mathfrak{D}'_1$:

$$\dfrac{\dfrac{\vdots\ \mathfrak{D}_{L'_1}}{\Gamma \Rightarrow \Delta, x = y, \varphi^m} \quad \dfrac{\vdots\ \mathfrak{D}_{R}}{\varphi^n, \Sigma \Rightarrow \Pi}}{\Gamma, \Sigma \Rightarrow \Delta, \Pi, x = y}\,(Ecut)$$

Similarly, we can also construct the following derivation $\mathfrak{D}'_2$:

$$\dfrac{\dfrac{\vdots\,\mathfrak{D}_{L'_2}}{\Gamma \Rightarrow \Delta, x :^{L} p, \varphi^m} \quad \dfrac{\vdots\,\mathfrak{D}_R}{\varphi^n, \Sigma \Rightarrow \Pi}}{\Gamma, \Sigma \Rightarrow \Delta, \Pi, x :^{L} p}\ (Ecut)$$

Since the weight of the Ecut-bottom form in $\mathfrak{D}'_1$ and $\mathfrak{D}'_2$ are smaller than the original one, we can obtain derivations for $\Gamma, \Sigma \Rightarrow \Delta, \Pi, x = y$ and $\Gamma, \Sigma \Rightarrow \Delta, \Pi, x :^{L} p$ in **GDLRC**$^-$ by induction hypothesis. By $\mathfrak{D}'_1$ and $\mathfrak{D}'_2$, we can transform the original derivation into the following one:

$$\dfrac{\mathfrak{D}'_1 \quad \mathfrak{D}'_2}{\Gamma, \Sigma \Rightarrow \Delta, \Pi, y :^{L} p}\ (= at)$$

Since the transformed derivation does not contain $(Ecut)$, we can obtain a derivation for $\Gamma, \Sigma \Rightarrow \Delta, \Pi, y :^{L} p$ in **GDLRC**$^-$.

Case (5-2) where `rule`$(\mathfrak{D}_L)$ **is** $(= ref)$**:**

The Ecut-bottom form can be written as follows:

$$\dfrac{\dfrac{\dfrac{\vdots\,\mathfrak{D}_{L'}}{x = x, \Gamma \Rightarrow \Delta, \varphi^m}}{\Gamma \Rightarrow \Delta, \varphi^m}\ (= ref) \quad \dfrac{\vdots\,\mathfrak{D}_R}{\varphi^n, \Sigma \Rightarrow \Pi}}{\Gamma, \Sigma \Rightarrow \Delta, \Pi}\ (Ecut)$$

We can transform the above derivation into the following one:

$$\dfrac{\dfrac{\dfrac{\vdots\,\mathfrak{D}_{L'}}{x = x, \Gamma \Rightarrow \Delta, \varphi^m} \quad \dfrac{\vdots\,\mathfrak{D}_R}{\varphi^n, \Sigma \Rightarrow \Pi}}{x = x, \Gamma, \Sigma \Rightarrow \Delta, \Pi}\ (Ecut)}{\Gamma, \Sigma \Rightarrow \Delta, \Pi}\ (= ref)$$

Since the weight of the Ecut-bottom form in the above transformed derivation is smaller than the original one, we can obtain a derivation for $\Gamma, \Sigma \Rightarrow \Delta, \Pi$ in **GDLRC**$^-$ by induction hypothesis.

□

Theorem 4.19 (Cut Elimination)**.** *For any multisets* Γ *and* Δ *of labelled expressions, if* $\vdash_{\mathbf{GDLRC}} \Gamma \Rightarrow \Delta$*, then* $\vdash_{\mathbf{GDLRC}^-} \Gamma \Rightarrow \Delta$*.*

Proof. Since (Cut) is a special form of $(Ecut)$, we have that if $\vdash_{\mathbf{GDLRC}} \Gamma \Rightarrow \Delta$, then $\vdash_{\mathbf{GDLRC}^*} \Gamma \Rightarrow \Delta$. Thus, it suffices to show that if $\vdash_{\mathbf{GDLRC}^*} \Gamma \Rightarrow \Delta$, then $\vdash_{\mathbf{GDLRC}^-} \Gamma \Rightarrow \Delta$. Let $\mathfrak{D}$ be a derivation for $\Gamma \Rightarrow \Delta$ in **GDLRC***. Since **GDLRC*** contains $(Ecut)$, $\mathfrak{D}$ might contain some Ecut-bottom forms. If $\mathfrak{D}$ does not contain $(Ecut)$, it is trivial. Therefore, we suppose that $\mathfrak{D}$ contains some Ecut-bottom forms. Now, let us focus on the nearest uppermost occurrence of Ecut-bottom form $\mathfrak{D}'$ from an initial sequent in $\mathfrak{D}$. If we apply Lemma 4.18 to this $\mathfrak{D}'$, we can get a derivation $\mathfrak{D}''$ in **GDLRC**$^-$ whose initial sequent is the same as that of $\mathfrak{D}'$ in **GDLRC***. If we replace such $\mathfrak{D}'$ in $\mathfrak{D}$ by $\mathfrak{D}''$, we can eliminate an occurrence of $(Ecut)$ from $\mathfrak{D}$. Similarly, if we repeat the above procedure to each uppermost occurrence of Ecut-bottom form in $\mathfrak{D}$, we can obtain a derivation for $\Gamma \Rightarrow \Delta$ that does not contain $(Ecut)$, therefore $\vdash_{\mathbf{GDLRC}^-} \Gamma \Rightarrow \Delta$. □

4.3.4 Soundness of GDLRC for Kripke Semantics

To show the soundness of labelled sequent calculus **GDLRC** for Kripke semantics, we need to give a semantic interpretation of labelled expressions in terms of Kripke semantics. Thus, we lift our Kripke semantics to labelled expressions by an assignment function for variables.

Definition 4.20. *Let $\mathfrak{M} = (W, (R_a)_{a \in \mathsf{AP}}, V)$ be a model. We say $f : \mathsf{Var} \to W$ is an* assignment *on $\mathfrak{M}$. Given any model $\mathfrak{M}$ and any assignment f on $\mathfrak{M}$, we define $\mathfrak{M}, f \models \varphi$ by:*

$$\begin{array}{lll} \mathfrak{M}, f \models x :^L A & \text{iff} & \mathfrak{M}^L, f(x) \models A, \\ \mathfrak{M}, f \models x\,\mathsf{R}^L_\alpha\, y & \text{iff} & f(x) R^L_\alpha f(y), \\ \mathfrak{M}, f \models x = y & \text{iff} & f(x) = f(y). \end{array}$$

Definition 4.21. *A labelled sequent $\Gamma \Rightarrow \Delta$* holds in $\mathfrak{M}$ under f *(notation: $\mathfrak{M}, f \models \Gamma \Rightarrow \Delta$) if, whenever $\mathfrak{M}, f \models \gamma$ for all $\gamma \in \Gamma$, $\mathfrak{M}, f \models \delta$ for some $\delta \in \Delta$. We say that $\Gamma \Rightarrow \Delta$ is* valid in a model $\mathfrak{M}$ *(notation: $\mathfrak{M} \models \Gamma \Rightarrow \Delta$) if $\mathfrak{M}, f \models \Gamma \Rightarrow \Delta$ for any assignment f on $\mathfrak{M}$. We also say that $\Gamma \Rightarrow \Delta$ is* valid in a class $\mathbb{M}$ of models *(notation: $\mathbb{M} \models \Gamma \Rightarrow \Delta$) if $\mathfrak{M} \models \Gamma \Rightarrow \Delta$ for all models $\mathfrak{M} \in \mathbb{M}$.*

Lemma 4.22. *Let $\mathbb{M}_{\text{all}}$ be a class of all models. For any multisets Γ and Δ of labelled expressions,*

$$\textit{if } \vdash_{\mathbf{GDLRC}} \Gamma \Rightarrow \Delta, \textit{ then } \mathbb{M}_{\text{all}} \models \Gamma \Rightarrow \Delta.$$

Proof. Let φ be a labelled expression. We show the proof by induction on the height n of a derivation of $\Gamma \Rightarrow \Delta$ in **GDLRC**. Since the base case for initial sequents is trivial, we skip it. Then, for inductive steps, we show only the cases for $([\alpha])$, $([\mathsf{r}])$, $(;)$, (at), and (rel).

Case where the last applied rule of our derivation is $(L[\alpha])$:

Suppose that $\mathbb{M}_{\text{all}} \models \Gamma \Rightarrow \Delta, x\,\mathsf{R}^L_\alpha\, y$ and $\mathbb{M}_{\text{all}} \models y :^L A, \Gamma \Rightarrow \Delta$. We show that $\mathbb{M}_{\text{all}} \models x :^L [\alpha]A, \Gamma \Rightarrow \Delta$, i.e., $\mathfrak{M}, f \models x :^L [\alpha]A, \Gamma \Rightarrow \Delta$ for all $\mathfrak{M}$ and all f on $\mathfrak{M}$. Fix any $\mathfrak{M} = (W, (R_a)_{a \in \mathsf{AP}}, V)$ and f such that $\mathfrak{M}, f \models x :^L [\alpha]A$ and $\mathfrak{M}, f \models \gamma$ for all $\gamma \in \Gamma$. Our goal is to show that

$$\mathfrak{M}, f \models \delta \text{ for some } \delta \in \Delta.$$

By induction hypothesis, we have $\mathfrak{M}, f \models \Gamma \Rightarrow \Delta, x\,\mathsf{R}^L_\alpha\, y$. Since $\mathfrak{M}, f \models \gamma$ for all $\gamma \in \Gamma$, it follows that $\mathfrak{M}, f \models \delta$ for some $\delta \in \Delta$ or $\mathfrak{M}, f \models x\,\mathsf{R}^L_\alpha\, y$. If $\mathfrak{M}, f \not\models x\,\mathsf{R}^L_\alpha\, y$, we immediately obtain our goal. Hence, we consider the case where $\mathfrak{M}, f \models x\,\mathsf{R}^L_\alpha\, y$, i.e., $f(x) R^L_\alpha f(y)$. By definition of $\mathfrak{M}, f \models x :^L [\alpha]A$ and $f(x) R^L_\alpha f(y)$, we get $\mathfrak{M}^L, f(y) \models A$, i.e., $\mathfrak{M}, f \models y :^L A$. By induction hypothesis, we have $\mathfrak{M}, f \models y :^L A, \Gamma \Rightarrow \Delta$, which implies our goal.

Case where the last applied rule of our derivation is $(R[\alpha])$:

Suppose that $\mathbb{M}_{\text{all}} \models x\,\mathsf{R}^L_\alpha\, y, \Gamma \Rightarrow \Delta, y :^L A$ where y does not appear in Γ, Δ nor $x :^L [\alpha]A$. We show that $\mathbb{M}_{\text{all}} \models \Gamma \Rightarrow \Delta, x :^L [\alpha]A$, i.e., $\mathfrak{M}, f \models \Gamma \Rightarrow \Delta, x :^L [\alpha]A$ for all $\mathfrak{M}$ and all f on $\mathfrak{M}$. Fix any $\mathfrak{M} = (W, (R_a)_{a \in \mathsf{AP}}, V)$ and f such that $\mathfrak{M}, f \models \gamma$ for all $\gamma \in \Gamma$. Our goal is to show that $\mathfrak{M}, f \models \delta$ for some $\delta \in \Delta$

or $\mathfrak{M}, f \models x :^L [\alpha]A$. Suppose that $\mathfrak{M}, f \not\models \delta$ for all $\delta \in \Delta$. We show that $\mathfrak{M}, f \models x :^L [\alpha]A$, i.e., if $f(x)R^L_\alpha v$, then $\mathfrak{M}^L, v \models A$ for all $v \in W$. Fix any $v \in W$ such that $f(x)R^L_\alpha v$. Our goal is to show that $\mathfrak{M}^L, v \models A$. Define a new assignment function $f' : \mathsf{Var} \to W$ by

$$f'(z) = \begin{cases} v & \text{if } z = y, \\ f(z) & \text{otherwise.} \end{cases}$$

Since $f'(y)$ is equal to v, it suffices to show that $\mathfrak{M}^L, f'(y) \models A$. Since y does not appear in Γ nor Δ, our suppositions implies $\mathfrak{M}, f' \models \gamma$ for all $\gamma \in \Gamma$ and $\mathfrak{M}, f' \not\models \delta$ for all $\delta \in \Delta$. Since $x \neq y$ and $f(x)R^L_\alpha v$, we have $f'(x)R^L_\alpha f'(y)$, i.e., $\mathfrak{M}, f' \models x \,\mathsf{R}^L_\alpha\, y$. By induction hypothesis, we have $\mathfrak{M}, f' \models x \,\mathsf{R}^L_\alpha\, y, \Gamma \Rightarrow \Delta, y :^L A$, which implies $\mathfrak{M}, f' \models y :^L A$.

Case where the last applied rule of our derivation is ($L[\mathsf{r}]$):
Suppose that $\mathbb{M}_{\mathsf{all}} \models x :^{(L,\mathsf{r})} A, \Gamma \Rightarrow \Delta$. We show that $\mathbb{M}_{\mathsf{all}} \models x :^L [\mathsf{r}]A, \Gamma \Rightarrow \Delta$, i.e., $\mathfrak{M}, f \models x :^L [\mathsf{r}]A, \Gamma \Rightarrow \Delta$ for all $\mathfrak{M}$ and all f on $\mathfrak{M}$. Fix any $\mathfrak{M}$ and f such that $\mathfrak{M}, f \models x :^L [\mathsf{r}]A$ and $\mathfrak{M}, f \models \gamma$ for all $\gamma \in \Gamma$. Our goal is to show that $\mathfrak{M}, f \models \delta$ for some $\delta \in \Delta$. Since $\mathfrak{M}, f \models x :^L [\mathsf{r}]A$, we have $\mathfrak{M}, f \models x :^{(L,\mathsf{r})} A$. By induction hypothesis, we have $\mathfrak{M}, f \models x :^{(L,\mathsf{r})} A, \Gamma \Rightarrow \Delta$, which implies our goal.

Case where the last applied rule of our derivation is ($R[\mathsf{r}]$):
Suppose that $\mathbb{M}_{\mathsf{all}} \models \Gamma \Rightarrow \Delta, x :^{(L,\mathsf{r})} A$. We show that $\mathbb{M}_{\mathsf{all}} \models \Gamma \Rightarrow \Delta, x :^L [\mathsf{r}]A$, i.e., $\mathfrak{M}, f \models \Gamma \Rightarrow \Delta, x :^L [\mathsf{r}]A$ for all $\mathfrak{M}$ and all f on $\mathfrak{M}$. Fix any $\mathfrak{M}$ and f such that $\mathfrak{M}, f \models \gamma$ for all $\gamma \in \Gamma$. Our goal is to show that $\mathfrak{M}, f \models \delta$ for some $\delta \in \Delta$ or $\mathfrak{M}, f \models x :^L [\mathsf{r}]A$. Suppose that $\mathfrak{M}, f \not\models \delta$ for all $\delta \in \Delta$. Our goal is to show that $\mathfrak{M}, f \models x :^L [\mathsf{r}]A$, i.e.,

$$\begin{aligned} & \mathfrak{M}^L, f(x) \models [\mathsf{r}]A, \\ \text{i.e., } \ & \mathfrak{M}^{(L,\mathsf{r})}, f(x) \models A, \\ \text{i.e., } \ & \mathfrak{M}, f \models x :^{(L,\mathsf{r})} A. \end{aligned}$$

By induction hypothesis, we have $\mathfrak{M}, f \models \Gamma \Rightarrow \Delta, x :^{(L,\mathsf{r})} A$. Since $\mathfrak{M}, f \models \gamma$ for all $\gamma \in \Gamma$, it follows that $\mathfrak{M}, f \models \delta$ for some $\delta \in \Delta$ or $\mathfrak{M}, f \models x :^{(L,\mathsf{r})} A$. If $\mathfrak{M}, f \models x :^{(L,\mathsf{r})} A$, we immediately obtain our goal. Hence we consider the case where $\mathfrak{M}, f \models \delta$ for some $\delta \in \Delta$, but it contradicts to our supposition.

Case where the last applied rule of our derivation is ($L;$):
Suppose that $\mathbb{M}_{\mathsf{all}} \models x \,\mathsf{R}^L_{\alpha_1}\, z, z \,\mathsf{R}^L_{\alpha_2}\, y, \Gamma \Rightarrow \Delta$ where z does not appear in Γ, Δ nor $x \,\mathsf{R}^L_{\alpha_1;\alpha_2}\, y$. We show that $\mathbb{M}_{\mathsf{all}} \models x \,\mathsf{R}^L_{\alpha_1;\alpha_2}\, y, \Gamma \Rightarrow \Delta$, i.e., $\mathfrak{M}, f \models x \,\mathsf{R}^L_{\alpha_1;\alpha_2}\, y, \Gamma \Rightarrow \Delta$ for all $\mathfrak{M}$ and all f on $\mathfrak{M}$. Fix any $\mathfrak{M} = (W, (R_a)_{a \in \mathsf{AP}}, V)$ and f. Suppose that $\mathfrak{M}, f \models \gamma$ for all $\gamma \in \Gamma$ and $\mathfrak{M}, f \models x \,\mathsf{R}^L_{\alpha_1;\alpha_2}\, y$, i.e., $f(x)R^L_{\alpha_1;\alpha_2}f(y)$, i.e., $f(x)R^L_{\alpha_1}m$ and $mR^L_{\alpha_2}f(y)$ for some $m \in W$. Our goal is to show that

$$\mathfrak{M}, f \models \delta \text{ for some } \delta \in \Delta.$$

Define a new assignment function $f' : \mathsf{Var} \to W$ by

$$f'(w) = \begin{cases} m & \text{if } w = z, \\ f(w) & \text{otherwise.} \end{cases}$$

Since $f'(z)$ is equal to m, it suffices to show that $\mathfrak{M}, f' \models \delta$ for some $\delta \in \Delta$. Since $f(x)R^L_{\alpha_1}m$ and $mR^L_{\alpha_2}f(y)$ for some $m \in W$, it follows that $f'(x)R^L_{\alpha_1}f'(z)$ and $f'(z)R^L_{\alpha_2}f'(y)$, i.e., $\mathfrak{M}, f' \models x\,\mathsf{R}^L_{\alpha_1}\,z$ and $\mathfrak{M}, f' \models z\,\mathsf{R}^L_{\alpha_2}\,y$. Since $\mathfrak{M}, f \models \gamma$ for all $\gamma \in \Gamma$ and z does not appear in Γ, we get $\mathfrak{M}, f' \models \gamma$ for all $\gamma \in \Gamma$. By induction hypothesis, we have $\mathfrak{M}, f' \models x\,\mathsf{R}^L_{\alpha_1}\,z, z\,\mathsf{R}^L_{\alpha_2}\,y, \Gamma \Rightarrow \Delta$, which implies our goal.

Case where the last applied rule of our derivation is (R;):

Suppose that $\mathbb{M}_{\text{all}} \models \Gamma \Rightarrow \Delta, x\,\mathsf{R}^L_{\alpha_1}\,z$ and $\mathbb{M}_{\text{all}} \models \Gamma \Rightarrow \Delta, z\,\mathsf{R}^L_{\alpha_2}\,y$. We show that $\mathbb{M}_{\text{all}} \models \Gamma \Rightarrow \Delta, x\,\mathsf{R}^L_{\alpha_1;\alpha_2}\,y$, i.e., $\mathfrak{M}, f \models \Gamma \Rightarrow \Delta, x\,\mathsf{R}^L_{\alpha_1;\alpha_2}\,y$ for all $\mathfrak{M}$ and all f on $\mathfrak{M}$. Fix any $\mathfrak{M} = (W, (R_a)_{a\in\mathsf{AP}}, V)$ and f such that $\mathfrak{M}, f \models \gamma$ for all $\gamma \in \Gamma$. Our goal is to show that $\mathfrak{M}, f \models \delta$ for some $\delta \in \Delta$ or $\mathfrak{M}, f \models x\,\mathsf{R}^L_{\alpha_1;\alpha_2}\,y$, i.e.,

$$\begin{array}{rl} & f(x)R^L_{\alpha_1;\alpha_2}f(y), \\ \text{i.e.,} & f(x)R^L_{\alpha_1}w \text{ and } wR^L_{\alpha_2}f(y) \text{ for some } w \in W. \end{array}$$

By induction hypothesis, we have $\mathfrak{M}, f \models \Gamma \Rightarrow \Delta, x\,\mathsf{R}^L_{\alpha_1}\,z$, which implies $\mathfrak{M}, f \models \delta$ for some $\delta \in \Delta$ or $\mathfrak{M}, f \models x\,\mathsf{R}^L_{\alpha_1}\,z$. If $\mathfrak{M}, f \models \delta$ for some $\delta \in \Delta$, we immediately obtain our goal. Hence, we consider the case where $\mathfrak{M}, f \models x\,\mathsf{R}^L_{\alpha_1}\,z$. Since $\mathfrak{M}, f \not\models \delta$ for all $\delta \in \Delta$, we get $\mathfrak{M}, f \models x\,\mathsf{R}^L_{\alpha_1}\,z$, i.e., $f(x)R^L_{\alpha_1}f(z)$. Similarly, by induction hypothesis, we have $\mathfrak{M}, f \models \Gamma \Rightarrow \Delta, z\,\mathsf{R}^L_{\alpha_2}\,y$, which implies $f(z)R^L_{\alpha_2}f(y)$. By $f(x)R^L_{\alpha_1}f(z)$ and $f(z)R^L_{\alpha_2}f(y)$, we obtain our goal.

Case where the last applied rule of our derivation is (Lat):

Suppose that $\mathbb{M}_{\text{all}} \models x :^L p, \Gamma \Rightarrow \Delta$. We show that $\mathbb{M}_{\text{all}} \models x :^{(L,\mathsf{r})} p, \Gamma \Rightarrow \Delta$, i.e., $\mathfrak{M}, f \models x :^{(L,\mathsf{r})} p, \Gamma \Rightarrow \Delta$ for all $\mathfrak{M}$ and all f on $\mathfrak{M}$. Fix any $\mathfrak{M} = (W, (R_a)_{a\in\mathsf{AP}}, V)$ and f such that $\mathfrak{M}, f \models x :^{(L,\mathsf{r})} p$ and $\mathfrak{M}, f \models \gamma$ for all $\gamma \in \Gamma$. Our goal is to show that $\mathfrak{M}, f \models \delta$ for some $\delta \in \Delta$. Then,

$$\begin{array}{rll} \mathfrak{M}, f \models x :^{(L,\mathsf{r})} p & \text{iff} & \mathfrak{M}^{(L,\mathsf{r})}, f(x) \models p, \\ & \text{iff} & f(x) \in V^{(L,\mathsf{r})}(p), \\ & \text{iff} & f(x) \in V^L(p), \\ & \text{iff} & \mathfrak{M}^L, f(x) \models p, \\ & \text{iff} & \mathfrak{M}, f \models x :^L p. \end{array}$$

So, $\mathfrak{M}, f \models x :^L p$. By induction hypothesis, we have $\mathfrak{M}, f \models x :^L p, \Gamma \Rightarrow \Delta$, which implies our goal.

Case where the last applied rule of our derivation is (Rat):

Suppose that $\mathbb{M}_{\text{all}} \models \Gamma \Rightarrow \Delta, x :^L p$. We show that $\mathbb{M}_{\text{all}} \models \Gamma \Rightarrow \Delta, x :^{(L,\mathsf{r})} p$, i.e., $\mathfrak{M}, f \models \Gamma \Rightarrow \Delta, x :^{(L,\mathsf{r})} p$ for all $\mathfrak{M}$ and all f on $\mathfrak{M}$. Fix any $\mathfrak{M} = (W, (R_a)_{a\in\mathsf{AP}}, V)$ and f such that $\mathfrak{M}, f \models \gamma$ for all $\gamma \in \Gamma$. Our goal is to show that $\mathfrak{M}, f \models \delta$ for some $\delta \in \Delta$ or $\mathfrak{M}, f \models x :^{(L,\mathsf{r})} p$, i.e.,

$$\begin{array}{rl} & \mathfrak{M}^{(L,\mathsf{r})}, f(x) \models p, \\ \text{i.e.,} & f(x) \in V^{(L,\mathsf{r})}(p), \\ \text{i.e.,} & f(x) \in V^L(p), \\ \text{i.e.,} & \mathfrak{M}, f \models x :^L p. \end{array}$$

By induction hypothesis, we have $\mathfrak{M}, f \models \Gamma \Rightarrow \Delta, x :^L p$, which implies our goal.

Case where the last applied rule of our derivation is ($Lrel$)

Suppose that $\mathbb{M}_{\text{all}} \models x\, \mathsf{R}^{L}_{\mathsf{r}(a)}\, y, \Gamma \Rightarrow \Delta$. We show that $\mathbb{M}_{\text{all}} \models x\, \mathsf{R}^{(L,\mathsf{r})}_{a}\, y, \Gamma \Rightarrow \Delta$, i.e., $\mathfrak{M}, f \models x\, \mathsf{R}^{(L,\mathsf{r})}_{a}\, y, \Gamma \Rightarrow \Delta$ for all $\mathfrak{M}$ and all f on $\mathfrak{M}$. Fix any $\mathfrak{M} = (W, (R_a)_{a\in\mathsf{AP}}, V)$ and f such that $\mathfrak{M}, f \models x\, \mathsf{R}^{(L,\mathsf{r})}_{a}\, y$ and $\mathfrak{M}, f \models \gamma$ for all $\gamma \in \Gamma$. Our goal is to show that $\mathfrak{M}, f \models \delta$ for some $\delta \in \Delta$. Then,

$$\begin{array}{rll} \mathfrak{M}, f \models x\, \mathsf{R}^{(L,\mathsf{r})}_{a}\, y & \text{iff} & f(x) R^{(L,\mathsf{r})}_{a} f(y), \\ & \text{iff} & f(x) R^{L}_{\mathsf{r}(a)} f(y), \\ & \text{iff} & \mathfrak{M}, f \models x\, \mathsf{R}^{L}_{\mathsf{r}(a)}\, y. \end{array}$$

So, $\mathfrak{M}, f \models x\, \mathsf{R}^{L}_{\mathsf{r}(a)}\, y$. By induction hypothesis, we have $\mathfrak{M}, f \models x\, \mathsf{R}^{L}_{\mathsf{r}(a)}\, y, \Gamma \Rightarrow \Delta$, which implies our goal.

Case where the last applied rule of our derivation is ($Rrel$):

Suppose that $\mathbb{M}_{\text{all}} \models \Gamma \Rightarrow \Delta, x\, \mathsf{R}^{L}_{\mathsf{r}(a)}\, y$. We show that $\mathbb{M}_{\text{all}} \models \Gamma \Rightarrow \Delta, x\, \mathsf{R}^{(L,\mathsf{r})}_{a}\, y$, i.e., $\mathfrak{M}, f \models \Gamma \Rightarrow \Delta, x\, \mathsf{R}^{(L,\mathsf{r})}_{a}\, y$ for all $\mathfrak{M}$ and all f on $\mathfrak{M}$. Fix any $\mathfrak{M} = (W, (R_a)_{a\in\mathsf{AP}}, V)$ and f such that $\mathfrak{M}, f \models \gamma$ for all $\gamma \in \Gamma$. Our goal is to show that $\mathfrak{M}, f \models \delta$ for some $\delta \in \Delta$ or $\mathfrak{M}, f \models x\, \mathsf{R}^{(L,\mathsf{r})}_{a}\, y$, i.e.,

$$\begin{array}{rl} & f(x) R^{(L,\mathsf{r})}_{a} f(y), \\ \text{i.e.,} & f(x) R^{L}_{\mathsf{r}(a)} f(y), \\ \text{i.e.,} & \mathfrak{M}, f \models x\, \mathsf{R}^{L}_{\mathsf{r}(a)}\, y. \end{array}$$

By induction hypothesis $\mathfrak{M}$ and f, we have $\mathfrak{M}, f \models \Gamma \Rightarrow \Delta, x\, \mathsf{R}^{L}_{\mathsf{r}(a)}\, y$, which implies our goal.

□

Theorem 4.23 (Soundness)**.** *If $\vdash_{\mathbf{GDLRC}^-} \Rightarrow x :^{\varepsilon} A$ for all $x \in \mathsf{Var}$, then A is valid on all models.*

Proof. Suppose $\vdash_{\mathbf{GDLRC}^-} \Rightarrow x :^{\varepsilon} A$ for all $x \in \mathsf{Var}$. We show that A is valid on all models, i.e., $\mathfrak{M}, w \models A$ for all $\mathfrak{M} = (W, (R_a)_{a\in\mathsf{AP}}, V)$ and $w \in W$. Fix any $\mathfrak{M}$ and $w \in W$. Define a new assignment function $f' : \mathsf{Var} \to W$ by $f'(y) = w$ if $y = x$; $f'(y) = v$ otherwise. By Lemma 4.22, we get that if $\vdash_{\mathbf{GDLRC}} \Gamma \Rightarrow \Delta$, then $\mathfrak{M}, f' \models \Gamma \Rightarrow \Delta$ for any multisets of Γ and Δ of labelled expressions. Since $\vdash_{\mathbf{GDLRC}^-} \Rightarrow x :^{\varepsilon} A$ for all $x \in \mathsf{Var}$, we get $\mathfrak{M}, f' \models \Rightarrow x :^{L} A$ which implies $\mathfrak{M}, w \models A$. □

Finally, by Fact 4.4 and our Theorems 4.11-4.23, we obtain the following corollary.

Corollary 4.24. *Given any formula A, the following are equivalent:*

(*i*) *A is valid on all Kripke models,*

(*ii*) $\vdash_{\mathbf{HDLRC}} A$,

(*iii*) $\vdash_{\mathbf{GDLRC}} \Rightarrow x :^{\varepsilon} A$ *for all* $x \in \mathsf{Var}$,

(*iv*) $\vdash_{\mathbf{GDLRC}^-} \Rightarrow x :^{\varepsilon} A$ *for all* $x \in \mathsf{Var}$.

Proof. Firstly, the direction from (*i*) to (*ii*) is established by Fact 4.4 (**HDLRC** is sound and complete). Secondly, the direction from (*ii*) to (*iii*) is shown by Theorem 4.11 (all theorems of **HDLRC** are derivable in **GDLRC**). Thirdly, the direction from (*iii*) to (*iv*) is shown by Theorem 4.19 (cut-elimination holds for **GDLRC**). Finally, the direction from (*iv*) to (*i*) is shown by Theorem 4.23 (**GDLRC** is sound for Kripke semantics). $\square$

Chapter 5

Linear Algebraic Semantics for Multi-agent Communication

In this chapter, we try to combine idea of the linear algebraic approach to **DLRC** (Chapter 4) with integration of the notion of communication channels into **DEL**. First, we integrate the notion of the channels into **DEL**. In Section 5.1, we introduce a static logic of agents' belief equipped with the notion of channel between agents and establish that all the valid formulas on all the *finite* Kripke models for our syntax is completely axiomatizable (Theorem 5.11). We also show that our proposed axiomatization is decidable (Theorem 5.12). In Section 5.2, in order to deal with changes of agents' belief via communication channel, we provide two dynamic operators to our syntax of static logic with sets of reduction axioms. Afterward, we introduce extension of our linear algebraic approach. In Section 5.3, with the help of the idea of **DLRC** [39], we reveal that we can regard our two dynamic operators as programs in $\mathbf{PDL}^{-}$ and also reformulates the semantics with two operators in terms of Boolean matrices. Finally, in Section 5.4, we present algorithms for checking agent's belief at a given world and for rewriting a given Kripke model using our Boolean matrix reformulation.

5.1 Doxastic Logic with Communication Channels

5.1.1 Syntax and Semantics

Doxastic logic is a variant of epistemic logic that concerns agent's beliefs rather than knowledges (for epistemic logic of knowledge, see also Section 2.3). In this section, we introduce the doxastic logic with the notion of communication channels between agents (for short, $\mathbf{ML}_{\mathsf{c}}$). Let PROP be a finite set of propositional variables and G be a fixed finite set of agents. Besides the propositional connectives, the language $\mathcal{L}_{\mathbf{ML}_{\mathsf{c}}}$ contains the following operators: *belief operators* $[\mathsf{B}_a]$ ($a \in \mathsf{G}$) and *channel constants* c_{ab} ($a, b \in \mathsf{G}$). We regard a finite set MOD as $\{\, [\mathsf{B}_a] \mid a \in \mathsf{G} \,\}$. Then, a set $\mathsf{Form}_{\mathbf{ML}_{\mathsf{c}}}$ of formulas of the language $\mathcal{L}_{\mathbf{ML}_{\mathsf{c}}}$ is inductively defined as follows:

$$\mathsf{Form}_{\mathbf{ML}_{\mathsf{c}}} \ni A ::= p \mid \mathsf{c}_{ab} \mid \neg A \mid A \vee B \mid [\mathsf{B}_a]A$$

where $p \in \mathsf{PROP}$, $a, b \in \mathsf{G}$ and $[\mathsf{B}_a] \in \mathsf{MOD}$. We introduce the defined abbreviations for propositional connectives $\wedge, \rightarrow, \leftrightarrow$ as in Section 2.1.1 and the dual operator $\langle \mathsf{B}_a \rangle$ of $[\mathsf{B}_a]$ by $\langle \mathsf{B}_a \rangle A := \neg[\mathsf{B}_a]\neg A$. c_{ab} stands for 'there is a communication channel from a to b.'

Then, $[\mathsf{B}_a]p$ and $\langle \mathsf{B}_a \rangle A$ stand for 'agent a believes that p' and for 'agent a considers it possible that A,' respectively.

Next, let us provide Kripke semantics with our syntax. In this logic, we extend a standard Kripke model with a set of (communication) channels. That is, a *model* $\mathfrak{M}$ is a tuple $(W, (R_a)_{a \in \mathsf{G}}, (C_{ab})_{a,b \in \mathsf{G}}, V)$ where W is a non-empty set of worlds, $R_a \subseteq W \times W$, $C_{ab} \subseteq W$ is a *channel relation* such that $C_{aa} = W$ for all $a \in \mathsf{G}$, and $V : \mathsf{Prop} \to \mathcal{P}(W)$ is a valuation function. Note that we require $C_{aa} = W$ for all $a \in \mathsf{G}$ in order to capture our notion of communication channel. A *frame* (denoted by $\mathfrak{F}$, etc.) is the result of dropping a valuation function from a model, as usual.

Given any model $\mathfrak{M}$, any world $w \in W$, and any formula A, we define the *satisfaction relation* $\mathfrak{M}, w \models A$ inductively as follows:

$$\begin{array}{lll}
\mathfrak{M}, w \models p & \text{iff} & w \in V(p), \\
\mathfrak{M}, w \models \mathsf{c}_{ab} & \text{iff} & w \in C_{ab}, \\
\mathfrak{M}, w \models \neg A & \text{iff} & \mathfrak{M}, w \not\models A, \\
\mathfrak{M}, w \models A \vee B & \text{iff} & \mathfrak{M}, w \models A \text{ or } \mathfrak{M}, w \models B, \\
\mathfrak{M}, w \models [\mathsf{B}_a]A & \text{iff} & \text{for all } v \in W : wR_av \text{ implies } \mathfrak{M}, v \models A.
\end{array}$$

The *truth set* $[\![A]\!]_{\mathfrak{M}}$ is defined by $[\![A]\!]_{\mathfrak{M}} = \{ w \in W \mid \mathfrak{M}, w \models A \}$. The notion of the *validity* is also provided with respect to the above frames and models as in Section 2.1.1.

Proposition 5.1. *For any $a \in \mathsf{G}$, c_{aa} is always valid in any Kripke model $\mathfrak{M}$.*

Proof. Fix any $a \in \mathsf{G}$, any $\mathfrak{M}$ and any $w \in |\mathfrak{M}|$. We show $\mathfrak{M}, w \models \mathsf{c}_{aa}$, i.e., $w \in C_{aa}$. By definition, $C_{aa} = |\mathfrak{M}|$ and it follows that $w \in C_{aa}$. ☐

Example 5.2. *Let $\mathsf{G} = \{ a, b \}$. Define $\mathfrak{M}$ (see Figure 5.1) by:*

$$\begin{array}{lll}
W & = & \{ w_1, w_2, w_3 \}, \\
R_a & = & \{ (w_1, w_1), (w_1, w_2), (w_1, w_3), (w_2, w_2), (w_3, w_3) \}, \\
R_b & = & W \times W, \\
V(p) & = & \{ w_2 \}, \\
C_{ab} & = & \{ w_1, w_2 \}, \\
C_{ba} & = & \emptyset, \\
C_{aa} & = & C_{bb} = W.
\end{array}$$

Agent a believes p in w_2 and $\neg p$ in w_3, but he/she is not sure of p or $\neg p$ in w_1. On the other hand, agent b does not believe p nor $\neg p$ at all the worlds. There are channels from a to b in w_1 and w_2, but there is no channel between them in w_3.

5.1.2 Hilbert-style Axiomatization $\mathsf{H}\mathbf{K}_\mathsf{c}$

In Table 5.1, we present the sound and complete Hilbert-style axiomatization $\mathsf{H}\mathbf{K}_\mathsf{c}$. We can regard the system $\mathsf{H}\mathbf{K}_\mathsf{c}$ as an axiomatic extension of $\mathsf{H}\mathbf{K}\boldsymbol{\Sigma}$ in terms of belief operators $[\mathsf{B}_a]$ ($a \in \mathsf{G}$). Therefore, we use the same manner in Section 2.1.3 to define a *derivation* and *theorem* in $\mathsf{H}\mathbf{K}_\mathsf{c}$. Then, we denote A is a theorem in $\mathsf{H}\mathbf{K}_\mathsf{c}$ by $\vdash_{\mathsf{H}\mathbf{K}_\mathsf{c}} A$. Similarly to $\mathsf{H}\mathbf{K}\boldsymbol{\Sigma}$, we can show the completeness of $\mathsf{H}\mathbf{K}_\mathsf{c}$ by the method of the canonical model (cf. Section 2.1.3), where the model is extended with a set of channels. We can also use the same manner in Section 2.1.3 to define a *maximally consistent set* ($\mathsf{H}\mathbf{K}_\mathsf{c}$-*MCS*, we also use the word $\mathsf{H}\mathbf{K}_\mathsf{c}$-*consistent* in what follows).

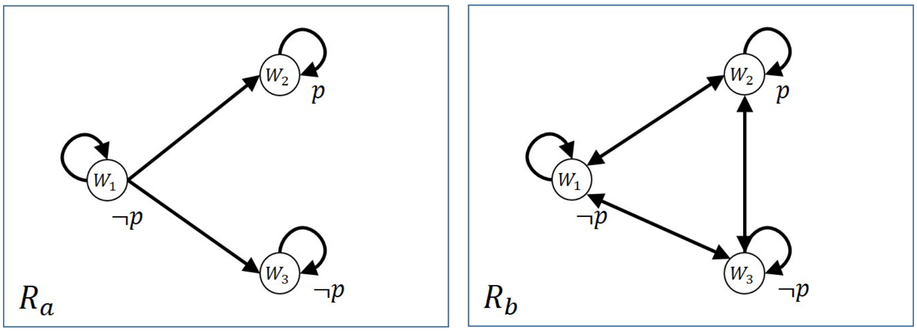


Figure 5.1: Accessibility Relations of Agents a and b.

Table 5.1: Hilbert-style Axiomatization $\mathsf{H}\mathbf{K}_{\mathsf{c}}$ of Static Logic

(**Taut**)	A, A is a tautology
($\mathbf{K}_{[\mathsf{B}]}$)	$[\mathsf{B}_a](A \to B) \to ([\mathsf{B}_a]A \to [\mathsf{B}_a]B) \quad (a \in \mathsf{G})$
(**Selfchn**)	$\mathsf{c}_{aa} \quad (a \in \mathsf{G})$
(**MP**)	From A and $A \to B$, infer B
($\mathbf{Nec}_{[\mathsf{B}]}$)	From A, infer $[\mathsf{B}_a]A \quad (a \in \mathsf{G})$

Definition 5.3. *For the axiomatic extension $\mathsf{H}\mathbf{K}_{\mathsf{c}}$, the canonical model $\mathfrak{M}^{\mathsf{H}\mathbf{K}_{\mathsf{c}}} = (W^{HKc}, (R_a^{\mathsf{H}\mathbf{K}_{\mathsf{c}}})_{a \in \mathsf{G}}, (C_{ab}^{\mathsf{H}\mathbf{K}_{\mathsf{c}}})_{a,b \in \mathsf{G}}, V^{\mathsf{H}\mathbf{K}_{\mathsf{c}}})$ is defined by:*

- $W^{\mathsf{H}\mathbf{K}_{\mathsf{c}}} := \{ \Gamma \mid \Gamma$ *is an* $\mathsf{H}\mathbf{K}_{\mathsf{c}}$*-MCS*$\}$.
- $\Gamma R_a^{\mathsf{H}\mathbf{K}_{\mathsf{c}}} \Delta$ iff $[\mathsf{B}_a]A \in \Gamma$ *implies* $A \in \Delta$ *for all formulas* A.
- $\Gamma \in C_{ab}^{\mathsf{H}\mathbf{K}_{\mathsf{c}}}$ iff $\mathsf{c}_{ab} \in \Gamma$.
- $\Gamma \in V^{\mathsf{H}\mathbf{K}_{\mathsf{c}}}(p)$ iff $p \in \Gamma$.

Lemma 2.12 (Lindenbaum's Lemma) also holds in terms of $\mathsf{H}\mathbf{K}_{\mathsf{c}}$.

Lemma 5.4 (Lindenbaum's Lemma)**.** *If Γ is any $\mathsf{H}\mathbf{K}_{\mathsf{c}}$-consistent set, then there exists an $\mathsf{H}\mathbf{K}_{\mathsf{c}}$-MCS Γ^+ such that $\Gamma \subseteq \Gamma^+$.*

Then, we can show the following equivalence as in Lemma 2.15 (Truth Lemma).

Lemma 5.5 (Truth Lemma)**.** *Given any formula A and any $\mathsf{H}\mathbf{K}_{\mathsf{c}}$-MCS Γ,*

$$\mathfrak{M}^{\mathsf{H}\mathbf{K}_{\mathsf{c}}}, \Gamma \models A \text{ iff } A \in \Gamma.$$

Proof. This is shown by induction on A. We only show the case for $A \equiv \mathsf{c}_{ab}$. Our goal is to show that:

$$\begin{aligned} & \mathfrak{M}^{\mathsf{H}\mathbf{K}_{\mathsf{c}}}, \Gamma \models \mathsf{c}_{ab} \text{ iff } \mathsf{c}_{ab} \in \Gamma, \\ \text{i.e.,} \quad & \Gamma \in C_{ab}^{\mathsf{H}\mathbf{K}_{\mathsf{c}}} \text{ iff } \mathsf{c}_{ab} \in \Gamma, \\ \text{i.e.,} \quad & \mathsf{c}_{ab} \in \Gamma \text{ iff } \mathsf{c}_{ab} \in \Gamma. \end{aligned}$$

This is trivial. One can show that the remaining cases by the same manner as in Lemma 2.15. □

Proposition 5.6. *Let* $(W^{HKc}, (R_a^{\mathsf{HK_c}})_{a\in\mathsf{G}}, (C_{ab}^{\mathsf{HK_c}})_{a,b\in\mathsf{G}}, V^{\mathsf{HK_c}})$ *be the canonical model. For all* $a \in \mathsf{G}$,

$$C_{aa}^{\mathsf{HK_c}} = W^{\mathsf{HK_c}}.$$

Proof. Since c_{aa} is always valid in any Kripke model (Proposition 5.1), this is trivial by definition. ☐

In order to show the completeness of $\mathsf{HK_c}$, it is important to restrict our attention to the *finite* Kripke models. This is because our linear algebraic reformulation of Kripke semantics is based on *finite matrix*. Hence, we use filtration technique to obtain the completeness of $\mathsf{HK_c}$ with respect to finite models.

Definition 5.7. *Given any formulas* A *in* $\mathsf{Form}_{\mathbf{ML_c}}$*, we define the subformulas* $Sub(A) : \mathsf{Form}_{\mathbf{ML_c}} \to \mathcal{P}(\mathsf{Form}_{\mathbf{ML_c}})$ *by:*

$$\begin{array}{lcl} Sub(p) & := & \{\, p \,\}, \\ Sub(\mathsf{c}_{ab}) & := & \{\, \mathsf{c}_{ab} \,\}, \\ Sub(\neg A) & := & Sub(A) \cup \{\, \neg A \,\}, \\ Sub(A \cup B) & := & Sub(A) \cup Sub(B) \cup \{\, A \cup B \,\}, \\ Sub([\mathsf{B}_a]A) & := & Sub(A) \cup \{\, [\mathsf{B}_a]A \,\}. \end{array}$$

We also define $Sub(\cdot)$ *for the set* Γ *of formulas as follows:*

$$Sub(\Gamma) := \bigcup_{A\in\Gamma} Sub(A).$$

We say that the set Γ *of formulas is closed under taking subformulas if* $Sub(A) \subseteq \Gamma$ *for all formulas* $A \in \Gamma$.

Definition 5.8. *Let* $\mathfrak{M} = ckripke$ *be a model and* Γ *be a finite set of formulas that is closed under taking subformulas. Without loss of generality, we can assume that* $\mathsf{c}_{aa} \in \Gamma$ *for all agents* a *occurring in* Γ *(otherwise, we can just add* c_{aa}*s to* Γ *for all as occurring in* Γ *where note that the number of such as is finite since* G *is finite). We define the equivalence relation* $\sim_\Gamma$ *on* W *by:*

$$w \sim_\Gamma v \text{ iff } (\mathfrak{M}, w \models A \text{ iff } \mathfrak{M}, v \models A) \textit{for all } A \in \Gamma.$$

We define the equivalence class of $w \in W$ *with respect to* $\sim_\Gamma$ *by:*

$$[w] := \{\, v \in W \mid w \sim_\Gamma v \,\}.$$

Definition 5.9 (Filtration). *Let* $\mathfrak{M} = ckripke$ *be a model and* Γ *be a finite set of formulas that is closed under taking subformulas. The model* $\mathfrak{M}^\Gamma := (W^\Gamma, (R_a^\Gamma)_{a\in AG}, (C_{ab}^\Gamma)_{a,b\in\mathsf{G}}, V^\Gamma)$ *is a* filtration of $\mathfrak{M}$ through Γ *if it satisfies the following conditions:*

- $W^\Gamma := W/\sim_\Gamma = \{\, [w] \mid w \in W \,\}$.
- $[w]R_a^\Gamma[w']$ iff $w' R_a v'$ *for some* $w' \in [w]$ *and* $v' \in [v]$.
- $[w] \in C_{ab}^\Gamma$ iff $w \in C_{ab}$.
- $[w] \in V^\Gamma(p)$ iff $w \in V(p)$.

Remark that C^{Γ}_{aa} always holds, since we assumed that $\mathsf{c}_{aa} \in \Gamma$ for all as occurring in Γ. Remark also that the size of W^{Γ} is less than or equal to $2^{\#\Gamma}$, hence finite.

Theorem 5.10 (Filtration Theorem). *Let $\mathfrak{M} = (W, (R_a)_{a\in\mathsf{G}}, (C_{ab})_{a,b\in\mathsf{G}}, V)$ be a model and Γ be a finite set of formulas that is closed under taking subformulas. For any $w \in W$ any formulas $A \in \Gamma$,*

$$\mathfrak{M}, w \models A \text{ iff } \mathfrak{M}^{\Gamma}, [w] \models A.$$

Theorem 5.11. *Let A be a formulas in $\mathsf{Form}_{\mathbf{ML_c}}$ and $\mathbb{M}$ be the class of all* finite *Kripke models.*

$$\mathbb{M} \models A \text{ iff } \vdash_{\mathsf{HK_c}} A.$$

Proof. Since the soundness is easy to establish, we focus on the completeness with respect to the class of all finite Kripke models. We establish the completeness of $\mathsf{HK_c}$ with respect to finite models by contraposition and the technique of filtration. Our goal is to show that if $\not\vdash_{\mathsf{HK_c}} A$, then $\mathbb{M} \not\models A$. With the help of the canonical model for $\mathsf{HK_c}$ and Lemmas 5.4 and 5.5, we can obtain that $\mathfrak{M}^{\mathsf{HK_c}} \not\models A$ by the same argument of Lemma 2.16 in terms of $\mathsf{HK_c}$. Since the domain of the canonical model $\mathfrak{M}^{\mathsf{HK_c}}$ is infinite, we use the filtration technique to boil the model down to a finite model. By Theorem 5.10 (Filtration Theorem) to $\mathfrak{M}^{\mathsf{HK_c}} \not\models A$, we obtain $(\mathfrak{M}^{\mathsf{HK_c}})^{\Gamma}, [w] \not\models A$, i.e., $\mathbb{M} \not\models A$.

□

Theorem 5.12. $\mathsf{HK_c}$ *is decidable.*

Proof. By Theorem 5.12, we have that if $\not\vdash_{\mathsf{HK_c}} A$, then there exists a model $\mathfrak{M}$ such that $\mathfrak{M} \not\models A$, i.e., the model $\mathfrak{M}$ is a finite counter model. Since we can recursively check if a given finite model satisfies the condition $C_{aa} = W$ for all agents $a \in \mathsf{G}$ (note that G is finite), we can construct an effective procedure generating all the finite Kripke models and checking if A is falsified at some point of a finite model. Together with an effective procedure of enumerating all the theorems of $\mathsf{HK_c}$, we obtain the decision procedure of Theoremhood of $\mathsf{HK_c}$. □

5.2 Dynamic Operators for Channel Communication

This section introduces two dynamic operators which allows us to talk about agents' belief changes in terms of informing action. The first dynamic operator (semi-private announcement) specifies both the sender and the receiver, but the second operator (introspective announcement via channel) just specified the sender agents and we need to calculate the receivers of the information via communication channels.

5.2.1 Semi-private Announcement

One of the most well-known dynamic operators is public announcement operator [32] (see also Section 2.3), but our operator of this section differs from it by the following requirement:

(R3) Our introducing operators are *semi-private* or *non-public* announcements to some specific agents. We assume that an agent a can send a message to an agent b only when there is a channel from a to b.

Table 5.2: Hilbert-style Axiomatization $\mathsf{H}\mathbf{K}_{\mathsf{c}[\,\cdot\downarrow^a_b]}$

In addition to all the axioms and rules of $\mathsf{H}\mathbf{K}_{\mathsf{c}}$, we add:

$$
\begin{array}{lcl}
[A\downarrow^a_b]p & \leftrightarrow & p, \\
[A\downarrow^a_b]\mathsf{c}_{cd} & \leftrightarrow & \mathsf{c}_{cd}, \\
[A\downarrow^a_b]\neg B & \leftrightarrow & \neg [A\downarrow^a_b]B, \\
[A\downarrow^a_b](B \vee C) & \leftrightarrow & [A\downarrow^a_b]B \vee [A\downarrow^a_b]C, \\
[A\downarrow^a_b][\mathsf{B}_c]B & \leftrightarrow & [\mathsf{B}_c][A\downarrow^a_b]B \quad (c \neq b) \\
[A\downarrow^a_b][\mathsf{B}_b]B & \leftrightarrow & ((\mathsf{c}_{ab} \wedge [\mathsf{B}_a]A) \to [\mathsf{B}_b](A \to [A\downarrow^a_b]B)) \wedge \\
 & & (\neg(\mathsf{c}_{ab} \wedge [\mathsf{B}_a]A) \to [\mathsf{B}_b][A\downarrow^a_b]B)
\end{array}
$$

($\mathbf{Nec}_{[A\downarrow^a_b]}$) From B, infer $[A\downarrow^a_b]B$

When an agent informs one of the other agents of something, our basic assumption is that we need a (context-dependent) channel between those agents. The notion of channel was formalized as channel propositions c_{ab}.

Let us denote our intended dynamic operator by $[A\downarrow^a_b]$, whose reading is 'after the agent a informs the agent b of the message A via channel'. Our intended reading of $[A\downarrow^a_b]B$ is 'after the agent a informs the agent b to A, B'. Then, we expand our syntax $\mathcal{L}_{\mathbf{ML}_{\mathsf{c}}}$ with a dynamic operator $[A\downarrow^a_b]$. Let us call such the logic based on our expanded syntax by $\mathbf{ML}_{\mathsf{c}[\,\cdot\downarrow^a_b]}$ and denote the set of all formulas of $\mathbf{ML}_{\mathsf{c}[\,\cdot\downarrow^a_b]}$ by $\mathsf{Form}_{\mathbf{ML}_{\mathsf{c}[\,\cdot\downarrow^a_b]}}$. Given any Kripke model $\mathfrak{M} = (W, (R_a)_{a\in\mathsf{G}}, (C_{ab})_{a,b\in\mathsf{G}}, V)$ and any world $w \in W$, we define the satisfaction relation of $[A\downarrow^a_b]B$ by:

$$\mathfrak{M}, w \models [A\downarrow^a_b]B \text{ iff } \mathfrak{M}^{A\downarrow^a_b}, w \models B$$

where $\mathfrak{M}^{A\downarrow^a_b} = (W, (R_c^{A\downarrow^a_b})_{c\in\mathsf{G}}, (C_{ab})_{a,b\in\mathsf{G}}, V)$ and $(R_c^{A\downarrow^a_b})_{c\in\mathsf{G}}$ is defined as:

- If $c = b$, then for all $x \in W$,

$$R_b^{A\downarrow^a_b}(x) := \begin{cases} R_b(x) \cap [\![A]\!]_{\mathfrak{M}} & \text{if } \mathfrak{M}, x \models [\mathsf{B}_a]A \wedge \mathsf{c}_{ab}, \\ R_b(x) & \text{otherwise.} \end{cases}$$

- If $c \neq b$, then $R_c^{A\downarrow^a_b} := R_c$.

Semantically speaking, $[A\downarrow^a_b]$ restricts b's attention to the A's worlds if there is a channel from the agent a to b and agent a believes A. Otherwise, the action $[A\downarrow^a_b]$ will not change b's belief.

In Table 5.2, we present the Hilbert-style system $\mathsf{H}\mathbf{K}_{\mathsf{c}[\,\cdot\downarrow^a_b]}$ for $\mathbf{ML}_{\mathsf{c}[\,\cdot\downarrow^a_b]}$. Since we can regard the system $\mathsf{H}\mathbf{K}_{\mathsf{c}[\,\cdot\downarrow^a_b]}$ as an axiomatic extension of $\mathsf{H}\mathbf{K}_{\mathsf{c}}$, we use the same manner in Section 5.1.2 to define a *derivation* and a *theorem* in $\mathsf{H}\mathbf{K}_{\mathsf{c}[\,\cdot\downarrow^a_b]}$. We denote A is a theorem in $\mathsf{H}\mathbf{K}_{\mathsf{c}[\,\cdot\downarrow^a_b]}$ by $\vdash_{\mathsf{H}\mathbf{K}_{\mathsf{c}[\,\cdot\downarrow^a_b]}} A$.

Definition 5.13. *The translation* $t : \mathsf{Form}_{\mathbf{ML}_{\mathsf{c}[\,\cdot\downarrow^a_b]}} \to \mathsf{Form}_{\mathbf{ML}_{\mathsf{c}}}$ *for* $\mathbf{ML}_{\mathsf{c}[\,\cdot\downarrow^a_b]}$ *is defined by:*

$$
\begin{array}{lcl}
t(p) & = & p, \\
t(\mathsf{c}_{ab}) & = & \mathsf{c}_{ab}, \\
t(\neg A) & = & \neg t(A), \\
t(A \vee B) & = & t(A) \vee t(B), \\
t([\mathsf{B}_a]A) & = & [\mathsf{B}_a]t(A), \\
t([A\downarrow^a_b]p) & = & p, \\
t([A\downarrow^a_b]\mathsf{c}_{cd}) & = & \mathsf{c}_{cd}, \\
t([A\downarrow^a_b]\neg B) & = & \neg t([A\downarrow^a_b]B), \\
t([A\downarrow^a_b](B \vee C)) & = & t([A\downarrow^a_b]B) \vee t([A\downarrow^a_b]C), \\
t([A\downarrow^a_b][\mathsf{B}_c]B) & = & [\mathsf{B}_c]t([A\downarrow^a_b]B), \qquad (c \neq b) \\
t([A\downarrow^a_b][\mathsf{B}_b]B) & = & ((\mathsf{c}_{ab} \wedge [\mathsf{B}_a]t(A)) \to [\mathsf{B}_b](t(A) \to t([A\downarrow^a_b]B)))\wedge \\
 & & (\neg(\mathsf{c}_{ab} \wedge [\mathsf{B}_a]t(A)) \to [\mathsf{B}_b]t([A\downarrow^a_b]B)), \\
t([A\downarrow^a_b][A\downarrow^c_d]B) & = & t([A\downarrow^a_b]t([A\downarrow^c_d]B)).
\end{array}
$$

Lemma 5.14. *Given any formula* $A \in \mathsf{Form}_{\mathbf{ML}_{\mathsf{c}[\,\cdot\downarrow^a_b]}}$*,*

$$\vdash_{\mathsf{H}\mathbf{K}_{\mathsf{c}[\,\cdot\downarrow^a_b]}} A \leftrightarrow t(A).$$

Theorem 5.15. *Let* A *be a formulas in* $\mathsf{Form}_{\mathbf{ML}_{\mathsf{c}[\,\cdot\downarrow^a_b]}}$ *and* $\mathbb{M}$ *be the class of all finite Kripke models.*

$$\mathbb{M} \models A \text{ iff } \vdash_{\mathsf{H}\mathbf{K}_{\mathsf{c}[\,\cdot\downarrow^a_b]}} A.$$

Proof. We divide our proof into the soundness part (the direction from right to left) and the completeness part (from left to right) as follows.

Soundness part We show that if $\vdash_{\mathsf{H}\mathbf{K}_{\mathsf{c}[\,\cdot\downarrow^a_b]}}$, then $\mathbb{M} \models A$ for all A. One can easily check that each axiom in $\mathsf{H}\mathbf{K}_{\mathsf{c}[\,\cdot\downarrow^a_b]}$ is valid in the class $\mathbb{M}$ of all finite models and each rule in $\mathsf{H}\mathbf{K}_{\mathsf{c}[\,\cdot\downarrow^a_b]}$ preserves the validity in $\mathbb{M}$.

Completeness part We show that if $\mathbb{M} \models A$, then $\vdash_{\mathsf{H}\mathbf{K}_{\mathsf{c}[\,\cdot\downarrow^a_b]}} A$ for all A. We can reduce the completeness of $\mathsf{H}\mathbf{K}_{\mathsf{c}[\,\cdot\downarrow^a_b]}$ to that of $\mathsf{H}\mathbf{K}_{\mathsf{c}}$ (cf. Theorem 5.11). Fix any formula A and suppose that $\mathbb{M} \models A$. By the soundness part and Lemma 5.14, we get $\mathbb{M} \models A \leftrightarrow t(A)$. Then, by this and our assumption of $\mathbb{M} \models A$, we get $\mathbb{M} \models t(A)$. By this and the completeness of $\mathsf{H}\mathbf{K}_{\mathsf{c}}$ (Theorem 5.11), we have that $\vdash_{\mathsf{H}\mathbf{K}_{\mathsf{c}}} t(A)$. Since $\mathsf{H}\mathbf{K}_{\mathsf{c}[\,\cdot\downarrow^a_b]}$ is an axiomatic extension of $\mathsf{H}\mathbf{K}_{\mathsf{c}}$, we also have that $\vdash_{\mathsf{H}\mathbf{K}_{\mathsf{c}[\,\cdot\downarrow^a_b]}} t(A)$. Finally, by Lemma 5.14 and $\vdash_{\mathsf{H}\mathbf{PAL}} t(A)$, we obtain $\vdash_{\mathsf{H}\mathbf{PAL}} A$.

□

Example 5.16. *In Example 5.2, we obtain the truth of* $[p\downarrow^a_b][\mathsf{B}_b]p$ *at* w_2*, i.e., 'after agent* a *informs agent* b *of the message* A *via channel, agent* b *comes to believe* p*' in* w_2*. Figure 5.2 is the updated model of* $\mathfrak{M}$ *by* $[p\downarrow^a_b]$*. On the other hand, agent* a *does not have any channel to* b *in* w_3*, and so, the accessible worlds from* w_3 *will be unchanged even after the update of* $\mathfrak{M}$ *by* $[p\downarrow^a_b]$*. Therefore,* $[p\downarrow^a_b][\mathsf{B}_b]p$ *is false at* w_3*. Similarly, agent* a *does not believe* $\neg p$ *in* w_1*, i.e.,* $[\mathsf{B}_a]\neg p$ *fails in* w_1*, and so, the informing action* $[p\downarrow^a_b]$ *will not change the accessible worlds from* w_1*.*

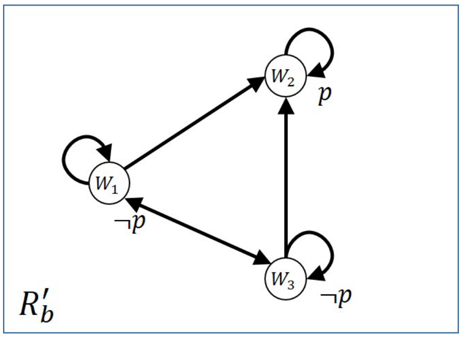


Figure 5.2: Updated Accessibility Relation of Agent b.

5.2.2 Introspective Announcement

In the dynamic operator $[B{\downarrow}^a_b]$, we specified a and b as the sender and the receiver of the information A, respectively. Even so, we may consider the situation where more than one agents, say a and b, send a piece of information to the other agents, and who will receive the information may change, depending on communication channels between agents. In this sense, we do not specify the receivers in advance here. Rather, we calculate the receivers of the information from the senders and the communication channels.

Let us denote such a dynamic operator by $[A{\downarrow}^{\mathsf{H}}]$ ($\mathsf{H} \subseteq \mathsf{G}$) whose reading is 'after a group H of agents sends a piece A of information via communication channels'. Then, we expand our syntax $\mathcal{L}_{\mathbf{ML}_{\mathsf{c}}}$ with a dynamic operator $[A{\downarrow}^{\mathsf{H}}]$ ($\mathsf{H} \subseteq \mathsf{G}$). Let us call the logic based on our expanded syntax by $\mathbf{ML}_{\mathsf{c}[\,\cdot{\downarrow}^{\mathsf{H}}]}$ and denote the set of all formulas of $\mathbf{ML}_{\mathsf{c}[\,\cdot{\downarrow}^{\mathsf{H}}]}$ by $\mathsf{Form}_{\mathbf{ML}_{\mathsf{c}[\,\cdot{\downarrow}^{\mathsf{H}}]}}$. Given any Kripke model $\mathfrak{M} = (W, (R_a)_{a\in\mathsf{G}}, (C_{ab})_{a,b\in\mathsf{G}}, V)$ and any world $w \in W$, we define the satisfaction relation of $[A{\downarrow}^{\mathsf{H}}]B$ by:

$$\mathfrak{M}, w \models [A{\downarrow}^{\mathsf{H}}]B \text{ iff } \mathfrak{M}^{A{\downarrow}^{\mathsf{H}}}, w \models B,$$

where $\mathfrak{M}^{A{\downarrow}^{\mathsf{H}}} = (W, (R_a^{A{\downarrow}^{\mathsf{H}}})_{a\in\mathsf{G}}, (C_{ab})_{a,b\in\mathsf{G}}, V)$ and $R_a^{A{\downarrow}^{\mathsf{H}}}$ is defined as follows: for all $w \in W$,

$$R_a^{A{\downarrow}^{\mathsf{H}}}(w) := \begin{cases} R_a(w) \cap [\![A]\!]_{\mathfrak{M}} & \text{if } \mathfrak{M}, w \models [\mathsf{B}_b]A \wedge \mathsf{c}_{ba} \text{ for some } b \in \mathsf{H}, \\ R_a(w) & \text{otherwise.} \end{cases}$$

Definition 5.17. *The translation* $t : \mathsf{Form}_{\mathbf{ML}_{\mathsf{c}[\,\cdot{\downarrow}^{\mathsf{H}}]}} \to \mathsf{Form}_{\mathbf{ML}_{\mathsf{c}}}$ *for* $\mathbf{ML}_{\mathsf{c}[\,\cdot{\downarrow}^{\mathsf{H}}]}$ *is defined by:*

$$\begin{array}{lcl}
t(p) & = & p, \\
t(\mathsf{c}_{ab}) & = & \mathsf{c}_{ab}, \\
t(\neg A) & = & \neg t(A), \\
t(A \vee B) & = & t(A) \vee t(B), \\
t([\mathsf{B}_a]A) & = & [\mathsf{B}_a]t(A), \\
t([A{\downarrow}^{\mathsf{H}}]p) & = & p, \\
t([A{\downarrow}^{\mathsf{H}}]\mathsf{c}_{ab}) & = & \mathsf{c}_{ab}, \\
t([A{\downarrow}^{\mathsf{H}}]\neg B) & = & \neg t([A{\downarrow}^{\mathsf{H}}]B), \\
t([A{\downarrow}^{\mathsf{H}}](B \vee C)) & = & t([A{\downarrow}^{\mathsf{H}}]B) \vee t([A{\downarrow}^{\mathsf{H}}]C), \\
t([A{\downarrow}^{\mathsf{H}}][\mathsf{B}_a]B) & = & (\bigvee_{b\in\mathsf{H}} (\mathsf{c}_{ba} \wedge [\mathsf{B}_b]t(A)) \to [\mathsf{B}_a](t(A) \to t([A{\downarrow}^{\mathsf{H}}]B))) \\
 & & \wedge(\neg \left(\bigvee_{b\in\mathsf{H}}(\mathsf{c}_{ba} \wedge [\mathsf{B}_b]t(A))\right) \to [\mathsf{B}_a]t([A{\downarrow}^{\mathsf{H}}]B)), \\
t([A{\downarrow}^{\mathsf{H}}][B{\downarrow}^{\mathsf{I}}]C) & = & t([A{\downarrow}^{\mathsf{H}}]t([B{\downarrow}^{\mathsf{I}}]C)).
\end{array}$$

Table 5.3: Hilbert-style Axiomatization $\mathsf{H}\mathbf{K}_{\mathsf{c}[\,\cdot\downarrow^{\mathsf{H}}]}$

In addition to all the axioms and rules of $\mathsf{H}\mathbf{K}_{\mathsf{c}}$, we add:

$$\begin{array}{lcl}
[A\downarrow^{\mathsf{H}}]p & \leftrightarrow & p, \\
[A\downarrow^{\mathsf{H}}]\mathsf{c}_{ab} & \leftrightarrow & \mathsf{c}_{ab}, \\
[A\downarrow^{\mathsf{H}}]\neg B & \leftrightarrow & \neg[A\downarrow^{\mathsf{H}}]B, \\
[A\downarrow^{\mathsf{H}}](B \vee C) & \leftrightarrow & [A\downarrow^{\mathsf{H}}]B \vee [A\downarrow^{\mathsf{H}}]C, \\
[A\downarrow^{\mathsf{H}}][\mathsf{B}_a]B & \leftrightarrow & (\bigvee_{b\in\mathsf{H}}(\mathsf{c}_{ba} \wedge [\mathsf{B}_b]A) \to [\mathsf{B}_a](A \to [A\downarrow^{\mathsf{H}}]B)) \\
 & & \wedge(\neg(\bigvee_{b\in\mathsf{H}}(\mathsf{c}_{ba} \wedge [\mathsf{B}_b]A)) \to [\mathsf{B}_a][A\downarrow^{\mathsf{H}}]B)
\end{array}$$

($\mathbf{Nec}_{[A\downarrow^{\mathsf{H}}]}$) From B, infer $[A\downarrow^{\mathsf{H}}]B$

Lemma 5.18. *Given any formula $A \in \mathsf{Form}_{\mathbf{ML}_{\mathsf{c}[\,\cdot\downarrow^{\mathsf{H}}]}}$,*

$$\vdash_{\mathsf{H}\mathbf{K}_{\mathsf{c}[\,\cdot\downarrow^{\mathsf{H}}]}} A \leftrightarrow t(A).$$

Theorem 5.19. *Let A be a formulas in $\mathsf{Form}_{\mathbf{ML}_{\mathsf{c}[\,\cdot\downarrow^{\mathsf{H}}]}}$ and $\mathbb{M}$ be the class of all finite Kripke models.*

$$\mathbb{M} \models A \text{ iff } \vdash_{\mathsf{H}\mathbf{K}_{\mathsf{c}[\,\cdot\downarrow^{\mathsf{H}}]}} A.$$

Proof. With the help of the translation for $\mathsf{H}\mathbf{K}_{\mathsf{c}[\,\cdot\downarrow^{\mathsf{H}}]}$ and Lemma 5.18, we can show the completeness for $\mathsf{H}\mathbf{K}_{\mathsf{c}[\,\cdot\downarrow^{\mathsf{H}}]}$ over the class $\mathbb{M}$ of all the finite Kripke models by the same argument of Theorem 5.15. $\square$

Example 5.20. *In Example 5.2, let $\mathsf{H} = \{a\}$ be a group of senders. Then, when we focus on the world w_2, we can calculate the receivers by the calculation just before this example and specify the receivers as $\{a, b\}$, since there is a channel from a to b in w_2 and a believes p in w_2. So, we obtain the truth of $[p\downarrow^{\mathsf{H}}][\mathsf{B}_b]p$ at w_2, i.e., 'after the group of agent H sends a piece p of information via communication channel, agent b comes to believe p' in w_2. Moreover, the updated model of $\mathfrak{M}$ by $[p\downarrow^{\mathsf{H}}]$ is the same as Figure 5.2.*

However, when we change the group of senders to $\mathsf{H}' = \{b\}$, agent b does not believe p in w_2 (i.e., $[\mathsf{B}_b]p$ is false in w_2), and so, the accessible worlds from w_2 will be unchanged even after the update of $\mathfrak{M}$ by $[p\downarrow^{\mathsf{H}'}]$. Therefore, $[p\downarrow^{\mathsf{H}'}][\mathsf{B}_b]p$ is still false at w_2.

5.3 Linear Algebraic Semantics for Channel Communication

5.3.1 Syntax and Semantics

Given a Kripke model $\mathfrak{M}$ with a domain $W = \{w_1, \ldots, w_m\}$, we may easily rewrite semantic clauses of $[A\downarrow^a_b]$ and $[\mathsf{H}\downarrow^A]$ in terms of matrix such as:

$$\begin{array}{rcl}
\|[A\downarrow^a_b]B\|_{\mathfrak{M}} & := & \|B\|_{\mathfrak{M}^{A\downarrow^a_b}}, \\
\|[A\downarrow^{\mathsf{H}}]B\|_{\mathfrak{M}} & := & \|B\|_{\mathfrak{M}^{A\downarrow^{\mathsf{H}}}},
\end{array}$$

where $\|[A\downarrow^a_b]B\|_{\mathfrak{M}}$ and $\|[A\downarrow^{\mathsf{H}}]B\|_{\mathfrak{M}}$ are matrices in $M(m \times 1)$. In general, it is not so clear whether we can capture processes of changing the given model $\mathfrak{M}$ by dynamic operators of **DEL** in terms of operations over matrices. However, in the semantics of $[A{\downarrow}^a_b]$ and $[A{\downarrow}^{\mathsf{H}}]$ ($\mathsf{H} \subseteq \mathsf{G}$), we keep the domain of a model, channel relations, and a valuation for propositional variables but *redefine* the accessibility relation $(R_a)_{a\in\mathsf{G}}$. In this sense, we may say that those operations are relation changers of **DLRC** [39, 25]. As we explained in Chapter 4, **DLRC** provides a general framework of changing agents' accessibility relations in terms of programs of $\mathbf{PDL}^-$. In addition, we have provided linear algebraic reformulation of Kripke semantics for **DLRC**. Therefore, if we extend our linear algebraic reformulation of that to handle the notion of communication channels, we can capture processes of changing $\mathfrak{M}$ to $\mathfrak{M}^{A{\downarrow}^a_b}$ and $\mathfrak{M}^{A{\downarrow}^{\mathsf{H}}}$ in terms of operations over matrices.

In what follows of this section, we expand our syntax $\mathcal{L}_{\mathbf{ML}_\mathsf{c}}$ with terms of $\mathbf{PDL}^-$ first. Then, we explain the main idea of van Benthem and Liu [39, 25] to regard our dynamic operators $[A{\downarrow}^a_b]$ and $[\mathsf{H}{\downarrow}^A]$ as relation changers. Let us call the logic of $\mathbf{PDL}^-$-extension of $\mathcal{L}_{\mathbf{ML}_\mathsf{c}}$ by $\mathbf{PDLc}^-$. We define the set $\mathsf{Form}_{\mathbf{PDLc}^-}$ of formulas, the set PR of programs by simultaneous induction as follows:

$$\begin{array}{rcl} \mathsf{Form}_{\mathbf{PDLc}^-} \ni A & ::= & p \mid \mathsf{c}_{ab} \mid \neg A \mid A \vee A \mid [\alpha]A \\ \mathsf{PR} \ni \alpha & ::= & a \mid (\alpha \cup \alpha) \mid (\alpha;\alpha) \mid ?A \end{array}$$

where $p \in \mathsf{PROP}$ and $a, b \in \mathsf{G}$. Note that we regard a as an atomic program (for agent a). Here, $[a]$ corresponds to the previous belief operator $[\mathsf{B}_a]$. So, in what follows, we also write $[\mathsf{B}_a]$ for $[a]$, if no confusion arises from the context.

Given any model $\mathfrak{M} = (W, (R_a)_{a\in\mathsf{G}}, (C_{ab})_{a,b\in\mathsf{G}}, V)$, any world $w \in W$, and any formula $A \in \mathsf{Form}_{\mathbf{PDLc}^-}$, the *satisfaction relation* $\mathfrak{M}, w \models A$ is naturally obtained from that of $\mathbf{ML}_\mathsf{c}$ and $\mathbf{PDL}^-$ (see also Sections 5.1 and 2.2). It follows that:

$$\begin{array}{lcl} [\![p]\!]_\mathfrak{M} & := & V(p), \\ [\![\mathsf{c}_{ab}]\!]_\mathfrak{M} & := & C_{ab}, \\ [\![\neg A]\!]_\mathfrak{M} & := & W \setminus [\![A]\!]_\mathfrak{M}, \\ [\![A \vee B]\!]_\mathfrak{M} & := & [\![A]\!]_\mathfrak{M} \cup [\![B]\!]_\mathfrak{M}, \\ [\![[\alpha]A]\!]_\mathfrak{M} & := & \{\, w \in W \mid [\![\alpha]\!]_\mathfrak{M}(w) \subseteq [\![A]\!]_\mathfrak{M} \,\}, \\ [\![a]\!]_\mathfrak{M} & := & R_a, \\ [\![\alpha \cup \alpha']\!]_\mathfrak{M} & := & [\![\alpha]\!]_\mathfrak{M} \cup [\![\alpha']\!]_\mathfrak{M}, \\ [\![\alpha;\alpha']\!]_\mathfrak{M} & := & [\![\alpha]\!]_\mathfrak{M} \circ [\![\alpha']\!]_\mathfrak{M}, \\ [\![?A]\!]_\mathfrak{M} & := & \{\, (w,v) \mid w = v \text{ and } w \in [\![A]\!]_\mathfrak{M} \,\}, \end{array}$$

where $[\![\alpha]\!]_\mathfrak{M}(w) := \{\, v \in W \mid (w,v) \in [\![\alpha]\!]_\mathfrak{M} \,\}$.

Recall that if relation changing operations are written in terms of programs generated from atomic programs by the composition ;, the union $\cup$ and the test $?A$, then we can automatically generate the set of reduction axioms (as in Tables 5.2 and 5.3) to assure the completeness of **DLRC** (see also Section 4.1). Let us suppose that our relation changer for a relation R_a ($= [\![a]\!]_\mathfrak{M}$) is written in terms of a program α_a ($a \in \mathsf{G}$). Then, we may use the notation $[(a := \alpha_a)_{a\in\mathsf{G}}]$ to mean our dynamic operator which changes an original relation R_a into a new relation R'_a via α_a for all agents $a \in \mathsf{G}$. Then, our key equivalence for generating the reduction axioms takes the following form:

$$[(a := \alpha_a)_{a\in\mathsf{G}}][b]A \leftrightarrow [\alpha_b][(a := \alpha_a)_{a\in\mathsf{G}}]A.$$

where we generalize van Benthem and Liu's equivalence [39] for a single agent to multi-agents.

Table 5.4: Hilbert-style Axiomatization $\mathsf{H}\mathbf{PDL}^{-}_{\mathsf{c}}$

(**Taut**)	All instances of propositional tautologies
($\mathbf{K}_{[\alpha]}$)	$[\alpha](A \to B) \to ([\alpha]A \to [\alpha]B)$
($[\cup]$)	$[\alpha \cup \beta]A \leftrightarrow [\alpha]A \wedge [\beta]A$
($[;]$)	$[\alpha;\beta]A \leftrightarrow [\alpha][\beta]A$
($[?]$)	$[?B]A \leftrightarrow (B \to A)$
(**Selfchn**)	$\mathsf{c}_{aa} \quad (a \in \mathsf{G})$
(**MP**)	From A and $A \to B$, infer B
($\mathbf{Nec}_{[\alpha]}$)	From A, infer $[\alpha]A$

Example 5.21. *In the semantics of* $[A{\downarrow}^a_b]$*, we have rewritten the accessibility relations* $(R_a)_{a\in\mathsf{G}}$ *into the new ones* $(R_a^{A{\downarrow}^a_b})_{a\in\mathsf{G}}$*. We may reformulate the semantics in terms of binary relations.*

- *Let* $c = b$*. Then,* $R_c^{A{\downarrow}^a_b} := (R_c \cap [\![\mathsf{c}_{ac} \wedge [\mathsf{B}_a]A]\!] \times [\![A]\!]) \cup (R_c \cap [\![\neg(\mathsf{c}_{ac} \wedge [\mathsf{B}_a]A)]\!] \times W)$*.*
- *Let* $c \neq b$*. Then,* $R_c^{A{\downarrow}^a_b} := R_c$*.*

Then, the corresponding relation changer agent b *to* $[A{\downarrow}^a_b]$ *is the following programs. When* $c = b$*,*

$$\alpha_b := (?(\mathsf{c}_{ab} \wedge [\mathsf{B}_a]A); b; ?A) \cup ?(\neg(\mathsf{c}_{ab} \wedge [\mathsf{B}_a]A); b).$$

If we employ the previous definitional abbreviation, we may write α_b *as:*

$$\alpha_b := \textbf{ if } \mathsf{c}_{ab} \wedge [\mathsf{B}_a]A \textbf{ then } b; ?A \textbf{ else } b.$$

When $c \neq b$*, the relation changer for agent* c *for* $[A{\downarrow}^a_b]$ *is:* $\alpha_c := c$*. Then, we may regard* $[A{\downarrow}^a_b]$ *as* $[(a := \alpha_a)_{a\in\mathsf{G}}]$*.*

Example 5.22. *Let* a *be any agent. The corresponding relation changer to* $[A{\downarrow}^{\mathsf{H}}]$ *is the following program term.*

$$\alpha_b^{A{\downarrow}^{\mathsf{H}}} := (?B; b; ?A) \cup (?\neg B; b)\,,$$

where $B := \bigvee_{a\in\mathsf{H}}(\mathsf{c}_{ab} \wedge [\mathsf{B}_a]A)$*. By the previous definitional abbreviation, we may write* $\alpha_b^{A{\downarrow}^{\mathsf{H}}}$ *as:*

$$\alpha_b^{A{\downarrow}^{\mathsf{H}}} := \textbf{if } \left(\bigvee\nolimits_{a\in\mathsf{H}}(\mathsf{c}_{ab} \wedge [\mathsf{B}_a]A)\right) \textbf{ then } b; ?A \textbf{ else } b.$$

Then, we may regard $[A{\downarrow}^{\mathsf{H}}]$ *as* $[(a := \alpha'_a)_{a\in\mathsf{G}}]$*.*

5.3.2 Relation Changers for Channel Communication in Matrices

Now, let us reformulate the semantics of $\mathbf{PDL}\mathsf{c}^{-}$ with our dynamic operators $[A{\downarrow}^a_b]$ and $[A{\downarrow}^{\mathsf{H}}]$ ($\mathsf{H} \subseteq \mathsf{G}$) in terms of Boolean matrices. First, let us recall a matrix representation of the ordinary (multi-agent) Kripke model as follows (cf. Sections 2.4 and 3.2.1). Let $\mathfrak{M} = (W, (R_a)_{a\in\mathsf{G}}, V)$ be a Kripke model where the cardinality of the domain W is m, i.e.,$W = \{ w_1, w_2, \ldots, w_m \}$. We define matrix representations R_a^M and $V^M(p)$ for an accessibility relation R_a and the valuation $V(p)$ as did in Section 3.2.1, respectively. Based on these definition, let us provide a matrix representation of the Kripke model (or frame) with the notion of communication channels. Since our model $(W, (R_a)_{a\in\mathsf{G}}, (C_{ab})_{a,b\in\mathsf{G}}, V)$ (with the cardinality of the domain W is m) is extended with a channel relation $C_{ab} \subseteq W$ ($a, b \in \mathsf{G}$) and the definition of C_{ab} is similar to that of the valuation $V(p) \subseteq W$, we can naturally obtain a matrix representation $C^M_{ab} \in M(m \times 1)$ (= a column vector) as follows.

$$C^M_{ab}(i) = \begin{cases} 1 & \text{if } w_i \in C_{ab}, \\ 0 & \text{if } w_i \notin C_{ab}. \end{cases}$$

Now it is ready to rewrite Kripke semantics to our syntax in terms of matrix. Given any finite Kripke model $\mathfrak{M} = (W, (R_a)_{a\in\mathsf{G}}, (C_{ab})_{a,b\in\mathsf{G}}, V)$. Suppose that $R^M_a \in M(m \times m)$ and $V(p)^M \in M(m \times m)$. We inductively associate each formula A of $\mathsf{Form}_{\mathbf{PDLc}^-}$ except c_{ab} with a column vector $\|A\|_{\mathfrak{M}} \in M(m \times 1)$ as did in Section 4.2.1. Then, we also associate the formula c_{ab} with a column vector $\|\mathsf{c}_{ab}\|_{\mathfrak{M}} \in M(m \times 1)$ as follows:

$$\|\mathsf{c}_{ab}\|_{\mathfrak{M}} \;:=\; C^M_{ab}.$$

Proposition 5.23. *Let A and B be formulas. Then, $[\![?(A \wedge B)]\!] = [\![?A; ?B]\!]$. Therefore, $\|?(A \wedge B)\| = \|?A; ?B\|$.*

Example 5.24. *Let us see whether our matrix representation of model update for semi-private announcement works on Example 5.2. As is the same as in Example 5.16, we consider the update by $[p{\downarrow}^a_b]$. There are channel between agent a and b, and agent a believes that p at w_2. By Proposition 5.23, the first part of a matrix calculation of $R_b^{A\downarrow^a_b}$ becomes:*

$$\begin{aligned} &\|?(\mathsf{c}_{ab} \wedge [\mathsf{B}_a]p)\| R^M_b \|?p\| = \|?\mathsf{c}_{ab}\|\|?[\mathsf{B}_a]p\| R^M_b \|?p\| \\ &= \begin{pmatrix} 1&0&0\\0&1&0\\0&0&0 \end{pmatrix} \begin{pmatrix} 0&0&0\\0&1&0\\0&0&0 \end{pmatrix} \begin{pmatrix} 1&1&1\\1&1&1\\1&1&1 \end{pmatrix} \begin{pmatrix} 0&0&0\\0&1&0\\0&0&0 \end{pmatrix} = \begin{pmatrix} 0&0&0\\0&1&0\\0&0&0 \end{pmatrix} \end{aligned}$$

Then, the remaining part of $R_b^{A\downarrow^a_b}$ becomes:

$$\begin{aligned} &\|?\neg(\mathsf{c}_{ab} \wedge [\mathsf{B}_a]p)\| R^M_b \\ &= \begin{pmatrix} 1&0&0\\0&0&0\\0&0&1 \end{pmatrix} \begin{pmatrix} 1&1&1\\1&1&1\\1&1&1 \end{pmatrix} = \begin{pmatrix} 1&1&1\\0&0&0\\1&1&1 \end{pmatrix} \end{aligned}$$

Then, we combine both results to obtain updated relation $R_b^{A\downarrow^a_b}$ of agent b as:

$$\begin{aligned} &\|R_b^{A\downarrow^a_b}\| = \|?(\mathsf{c}_{ab} \wedge [\mathsf{B}_a]p)\| R^M_b \|?p\| + \|?\neg(\mathsf{c}_{ab} \wedge [\mathsf{B}_a]p)\| R^M_b \\ &= \begin{pmatrix} 0&0&0\\0&1&0\\0&0&0 \end{pmatrix} + \begin{pmatrix} 1&1&1\\0&0&0\\1&1&1 \end{pmatrix} = \begin{pmatrix} 1&1&1\\0&1&0\\1&1&1 \end{pmatrix} \end{aligned}$$

This coincides with the result of Example 5.16 (see Figure 5.2)

Example 5.25. *Let us also see whether our matrix representation of model update for introspective announcement via channels works on Example 5.2. As in the same as Example 5.20, we consider the effect of* $[p{\downarrow}^{\mathsf{H}}]$ *over* $\mathfrak{M}$ *of Example 5.2. Let* $\mathsf{H} = \{ a \}$ *be the group of senders and focus on agent* b*. Then,*

$$[\![A]\!] = [\![\bigvee_{k\in\mathsf{H}}(\mathsf{c}_{kb} \wedge [\mathsf{B}_k]p)]\!] = [\![\mathsf{c}_{ab} \wedge [\mathsf{B}_a]p]\!] = \{ w_2 \}.$$

That is, the group H *of senders can send to* b *the piece of information* p *at* w_2 *alone. Thus, the first part of our matrix representation of the updated relation* $R_b^{A\downarrow^{\mathsf{H}}}$ *is:*

$$\|?A\|R_b^M\|?p\| = \begin{bmatrix} 0&0&0\\0&1&0\\0&0&0 \end{bmatrix} \begin{bmatrix} 1&1&1\\1&1&1\\1&1&1 \end{bmatrix} \begin{bmatrix} 0&0&0\\0&1&0\\0&0&0 \end{bmatrix} = \begin{bmatrix} 0&0&0\\0&1&0\\0&0&0 \end{bmatrix}$$

Then, the second part of $R_b^{A\downarrow^{\mathsf{H}}}$ *becomes:*

$$\|?\neg A\|R_b^M = \begin{bmatrix} 1&0&0\\0&0&0\\0&0&1 \end{bmatrix} \begin{bmatrix} 1&1&1\\1&1&1\\1&1&1 \end{bmatrix} = \begin{bmatrix} 1&1&1\\0&0&0\\1&1&1 \end{bmatrix}$$

By combining these results, we obtain:

$$\begin{aligned} \|R_b^{A\downarrow^{\mathsf{H}}}\| &= \|?A\|R_b^M\|?p\| + \|?\neg A\|R_b^M \\ &= \begin{bmatrix} 0&0&0\\0&1&0\\0&0&0 \end{bmatrix} + \begin{bmatrix} 1&1&1\\0&0&0\\1&1&1 \end{bmatrix} = \begin{bmatrix} 1&1&1\\0&1&0\\1&1&1 \end{bmatrix} \end{aligned}$$

which is the same result as Example 5.20 for agent b*.*

5.4 Algorithms for Channel Communication in Matrices

This section introduces two naive algorithms. One of them calculates the truth value of a formula $[\mathsf{B}_a]p$ and the other one calculates the relation updates by $[p{\downarrow}^a_b]$. For both algorithms, we assume that an input model $\mathfrak{M} = (W, (R_a)_{a\in\mathsf{G}}, (C_{ab})_{a,b\in\mathsf{G}}, V)$ is represented in terms of a boolean matrix. For simplicity, we do not care about the recursive calculation of the given formula A. We have implemented these algorithms into our software that is described in Appendix A.3.

Algorithm 1 Calculation of $\|[\mathsf{B}_a]p\|(i)$

```
procedure IS-BELIEVED-AT
    input 𝔐, w_i ∈ W, a ∈ G, p ∈ PROP
    ‖[B_a]p‖ := R_a^M \overline{\overline{V(p)}^M}
    return True if ‖[B_a]p‖(i) ≠ 0; False otherwise
end procedure
```

Here we comment just on Algorithm 2. In order to update an accessibility relation of agent b, the algorithm loops to find agent b. If the algorithm finds him/her, a model

Algorithm 2 Calculation of $[p{\downarrow}^a_b]$

procedure SEMI-PRIVATE-ANNOUNCEMENT
  input $\mathfrak{M}$, $a, b \in \mathsf{G}$, $p \in \mathsf{PROP}$
  for $c \in \mathsf{G}$ **do**
    if $c = b$ **then**
      $X :=$ Test(C^M_{ab})
      $Y :=$ Test$(\|[\mathsf{B}_a]p\|)$
      $Z :=$Test$(V(p)^M)$
      $(R^{p\downarrow^a_b}_b)^M := XYR^M_b Z + \overline{XY}R^M_b$
    else
      $(R^{p\downarrow^a_b}_c)^M := {R_c}^M$
    end if
  end for
  return $\mathfrak{M}^{p\downarrow^a_b} = (W, (R^{p\downarrow^a_b}_a)_{a\in\mathsf{G}}, (C)_{a,b\in\mathsf{G}}, V)$
end procedure

updating procedure (for a single agent) will be started, otherwise it just put $R^{p\downarrow^a_b}_c = R_c$. At the beginning of the updating procedure, the algorithm generates test matrices through `Test` function where an input of the function is a column vector. The function enumerates the elements of the input vector in the diagonal components of an output matrix, and fills 0 in the non-diagonal components of the matrix. Then, the algorithm calculates the updated accessibility relation of agent b in terms of boolean matrix. Note that $\|?\neg A\|$ can be calculated as $\overline{\|?A\|}$. Finally, the algorithm returns the updated model $\mathfrak{M}^{p\downarrow^a_b}$.

Chapter 6

Conclusions and Further Directions

6.1 Conclusions

As we explained in our introduction, the aim of this thesis is to provide new computational tools of modal logic of multi-agent communication. To achieve this goal, we have focused on three questions (Q1) - (Q3) (Section 1.1), and studied them. We can summarize our answers for these questions and technical contributions in the following three items:

(A1) Linear Algebraic Semantics for Modal Logic(s) and DLRC:

As for our solutions to question (Q1), we have studied a linear algebraic approach to Kripke semantics of modal logic in Chapter 3. Based on Fitting's linear algebraic reformulation of Kripke semantics of modal logic [8], we can represent an accessibility relation R by a Boolean matrix, and a valuation $V(p)$ of an atomic variable p by a Boolean column vector, provided the cardinality of the domain is finite. Then we can calculate the truth set of a formula by calculations over these matrices.

For this basis, we have added a linear algebraic reformulation of frame properties and shown some basic facts of that. In order to obtain the frame properties in terms of Boolean matrices, we have shown two types of correspondence between matrix reformulation of frame properties and corresponding modal axioms (Propositions 3.11 and 3.12). These correspondences allow us to capture all of the five well-known frame properties, i.e., reflexivity, symmetricity, transitivity, seriality and Euclideanness, in terms of Boolean matrices (Proposition 3.7). In addition, we can verify these frame properties of a given frame by calculations over Boolean matrices (Example 3.8) and can show implication among them (Proposition 3.9 and 3.10).

We have also shown a method to capture some restricted form of quantifications (in Kripke semantics) without bound variables of first-order logic by Boolean matrices. If we consider the case of the universal relation $R = W \times W$ in Kripke semantics, we may regard the semantic clauses of $\Diamond$ and $\Box$ of modal logic as those of $\forall$ and $\exists$ of first-order logic. Based on this idea, we can capture the distinction between $\exists\forall$ and $\forall\exists$ of first-order logic by matrix representation (Proposition 3.14).

In addition, in connection with question (Q2), we have also proposed linear algebraic reformulation of Kripke semantics for **DLRC** [39, 25] in Chapter 4. Based on the result for (Q1), we extend our approach to handle relation changers of **DLRC** that allows us to capture many dynamic operators of **DEL** in terms of relation changing operation(s). We have shown the soundness theorem of the known Hilbert-style axiomatization of **DLRC** [39, 25] in terms of Boolean matrices (Section 4.2.2). As a result, we can write and capture various proof of semantic properties of Kripke semantics by simple calculations over Boolean matrices. These are our main results of this thesis.

(A2) Cut-free Labelled Sequent Calculus for DLRC:
As for our solution to question (Q2), we have studied a labelled sequent calculus for dynamic logic of relation changers in Chapter 4. Based on labelled formalism for Kripke semantics of **DLRC**, we have defined our labelled sequent calculus (with the cut rule, Table 4.2). We have shown that all theorems of the known Hilbert-style system of this logic [39, 25] are also theorems of our sequent calculus (Theorem 4.11). Then, with the help of the method of Ono and Komori [29], we have shown that cut-elimination theorem for our sequent calculus (Theorem 4.19). In addition, we have also shown that our sequent calculus also holds soundness for Kripke semantics (Theorem 4.23) since we have defined equivalence between the ordinary Kripke semantics and labelled expressions of our calculus by the notion of assignment function. Finally, we have obtained that our sequent calculus is equipollent with the above sound and complete Hilbert-style axiomatization of this logic (Corollary 4.24) [39, 25].

(A3) Linear Algebraic Semantics for Multi-agent Communication:
As for our solution to question (Q3), we have studied an approach to integrate the notion of communication channels into **DEL** in Chapter 5. We have introduced the static doxastic logic with communication channels (where we always assume self-channel on all agents) with the complete axiomatization $\mathbf{K}_\mathsf{c}$ that is also decidable (Theorems 5.11 and 5.12). Then, we have also extended such static logic with two dynamic operators $[A{\downarrow}^a_b]$ (semi-private announcement) and $[A{\downarrow}^H]$ (introspective announcement) with reduction axioms (so extensions of both of them enjoy completeness results, Theorems 5.15 and 5.19). A key feature of our dynamic operators are *non-public*, i.e., effects of announcements are restricted to some specified agents determined by communication channels.

Moreover, in order to handle many indices that appear in our proposed logic(s) effectively, we have also studied linear algebraic representation of these. We have provided **PDL**-extension (without the iteration operator) of the above logic since we can define our dynamic operators as relation changers written in programs of **PDL** [39]. Finally, we have provided matrix reformulation of Kripke semantics of **PDL**-extension including communication channels, program constructors of **PDL** and our two dynamic operators (Example 5.3.2). Then, in order to show the basic idea for practical implementation, we have introduced two algorithms (Section 5.4). One of them calculates the truth value of a formula $[\mathsf{B}_a]p$ (Algorithm 1) and the other one calculates relation updates by $[p{\downarrow}^a_b]$ (Algorithm 2).

6.2 Further Directions

A further direction of our linear algebraic approach will be to use a similar approach to investigate the other types of modal logics. We may expand our mono-modal syntax into multi-modal one to cover, e.g., description logic [1] and dynamic epistemic logic [40].

As for description logic, a family of *roles* (say "has a child") generates both box-type and diamond-type modal operators. Therefore, we can capture the semantics of these operators by a set of the corresponding adjacency Boolean matrices to the roles. We can also cover some topics of dynamic epistemic logics [40] by our approach, where multi-modal operators are employed for describing agents' knowledge or beliefs (see also Chapter 5).

Another direction of our approach will be to handle the infinite norm of matrices. If we extend our approach to handle such matrices, we can show the soundness theorem for Kripke semantics by Boolean matrices without the limitation of the finiteness of matrices (cf. Section 4.2.2). In addition, if we free from such a limitation, it might be possible to prove the completeness theorem of modal logics via the soundness theorem on matrices.

In connection with our study of labelled sequent calculus **GDLRC**, we are planning to provide a direct proof of completeness of that without the cut rule. In particular, there must be a semantic proof of cut elimination. In addition, we can also consider another complete Hilbert-style axiomatization for **DLRC** by the method of Wang and Cao [41]. The method was used to obtain the complete axiomatization for **PAL** without reduction axioms. Is it possible to apply their method to obtain the complete axiomatization for **DLRC** without reduction axioms?

As for the variant of **GDLRC**, we may consider restricted version of ours, e.g., the intuitionistic labelled system, or extended version of that, e.g., the labelled system with the iteration operator $*$. In order to investigate such extensions or restrictions effectively, implementation of theorem prover might also be helpful to obtain rules or conditions for that.

Appendix A

Linear Algebraic Approach to Teach Modal Logic

In this appendix, we investigate our linear algebraic approach for educational purpose. In Section A.2, we explain which teaching topics of modal logics can be taught to students using our approach and how we can teach such topics. In Section A.3, we present our supporting software to avoid involved calculations on matrices. Finally, in Section A.6, we report our teaching experiment based on our linear algebraic approach.

A.1 Introduction

Modal logics are often taught to students as one of the advanced topics after propositional logic and first-order logic. This is because Kripke semantics of modal logics relies on knowledge of quantification and the binary relation of first-order logic and model theory. In particular, the notions of existential quantification and universal quantification are used to define the semantics of $\Diamond$ and $\Box$ operators over the Kripke model, respectively. Moreover, quantifications are also used to define the conditions of frame properties, e.g., reflexivity, i.e., for all w, wRw holds (wRv stands for 'there is a link from w to v'). In general, the notion of quantification and the binary relation of first-order logic are taught to students in a course of mathematical logic that contains such topics. Such a course are often provided by, for example, the departments of philosophy, computer science, and mathematics. Hence, the ordinary target students of a course on modal logic are usually assumed to belong to such departments. They assumed to have already learned the syntactic notions of quantification and a binary relation of first-order logic and the model-theoretic explanation of that.

So far, many topics of Kripke semantics are taught to students by the model-theoretic approach. However, as we mentioned in Section 1.1.1, this approach might confuse our students. For example, teaching the topics of the truth of $\Box p$ at a 'dead-end' world and the verification of the Euclideanness property are such candidates. In addition, if the cardinality of the domain of the model is larger, the situation for our students might be more involved.

As for a solution of this problem, we propose to use Fitting's linear algebraic approach to Kripke semantics [8] for education. Based on our technical contribution (A1) (see also Chapter 3), we may replace required prior knowledge of the quantifications and binary relation of first-order logic by the truth-table calculation of propositional logic and elementary calculations of Boolean matrices of linear algebra. We can teach many elementary topics of Kripke semantics for modal logics by calculations over Boolean matrices. We will explain how to teach modal logic using our approach in Section A.2. In this appendix, our target students are those who have prior knowledge of both linear algebra and propositional logic. In general, they are first or second year undergraduate students at the departments of computer science, electrical engineering, and physics. They might not be familiar with first-order logic.

We claim that our linear algebraic approach is helpful to students who have prior knowledge of both linear algebra and propositional logic. Using our approach, they can learn modal logics based on their acquired knowledge without prior knowledge of first-order logic. Moreover, ordinary target students, who have already learned first-order logic, may deepen their understanding of the subject from a different perspective. In order to support our approach for educational purpose, we show our supporting software in Section A.3. In addition, in order to test our approach and obtain feedbacks from students, we held a small seminar to teach elementary topics of modal logic in terms of our approach. We will explain these results in Section A.6, and will also show our lecture material in Appendix B.

A.2 Teaching Topics on Modal Logic by Linear Algebraic Approach

When we teach modal logics to students, the following topics are often covered:

1. Syntax: how to read modal operators, how to define formulas, the dual definition of modal operators, and the distinction between nested modalities.

2. Kripke semantics: a graphical representation of a Kripke model, the satisfaction relation, how to compute the truth value of a formula at a world, the validity and the satisfiability of a given formula, a counter-model construction, frame properties (reflexivity, symmetricity, transitivity, seriality and Euclideanness), and the correspondence between frame properties and formulas (T, B, D, 4 and 5).

3. Proof theory: Hilbert-style systems, tableau methods, natural deductions, sequent calculi, extensions by modal axioms (T, B, D, 4 and 5).

4. Possible further topics: bisimulation, finite model property, and decidability, complexity, soundness and completeness theorem of modal logics.

Here we focus our attention on elementary topics of items 1-3. For item 1, we should teach how to read modal operators at first. We introduce $\Box$ and $\Diamond$ operators and teach how to read them, i.e., we read $\Box$ as 'it is necessary that' and $\Diamond$ as 'it is possible that.' Then, we also teach the other readings of modal operators. For example, we read $\Box$ operator as 'it is believed that' in doxastic logic, 'it is known that' in epistemic logic, 'it is obligatory that' in deontic logic, 'it will always be the case that' and 'it has always been the case that' in temporal logic. Afterward, we should teach how to write a formula of modal logic by a BNF grammar. If $\Diamond$ operator is contained in the syntax, then we can define the other $\Box$ operator as the dual of the operator $\Diamond$, e.g., $\Box p := \neg\Diamond\neg p$. In addition, we should also teach the distinction between nested modalities, e.g., $\Box\Box p$, $\Box\Diamond p$ and $\Diamond\Box p$. In connection with epistemic logic, we also teach what the positive introspection ($\Box p \to \Box\Box p$) and the negative introspection ($\neg\Box p \to \Box\neg\Box p$) mean. The positive introspection stands for 'If agent knows, he/she knows what he/she knows', and negative introspection stands for 'If agent do not know, he/she knows that he/she do not know.'

For item 2, in order to teach Kripke semantics, we use the model-theoretic approach. A good point of modal logic is that we can calculate the truth value of a formula over a graphical representation of a Kripke model visually. However, such graphical approach sometimes might not work well, e.g., a calculation of the truth value of the formula $\Box p$ at the 'dead-end' world where we cannot access any world. In such a case, we should follow the definition of the satisfaction. We often give a brief introduction to the above five frame properties by the graphical approach intuitively, and then we explain frame conditions of a frame property by the model-theoretic approach rigorously. We also explain well-known implications among the frame properties, e.g., reflexivity and Euclideanness jointly imply transitivity (cf. Section 3.2.2). Afterward, we should explain correspondence between frame properties and valid formulas, e.g., a frame satisfies Euclideanness if and only if 5 ($\Diamond p \to \Box\Diamond p$) is valid on the frame.

For proof theory of item 3, we should introduce the basic proof system first. For example, the Hilbert-style base system is defined by propositional tautology, the distribution axiom for $\Box$ operator, $(\Box(p \to q) \to (\Box p \to \Box q))$, modus ponens (from A and $A \to B$, we may infer B), and the necessitation rule for $\Box$ operator (from A, we may infer $\Box A$). In addition, we also teach what a proof of the theorem is and what the notion of theorem on the base system is.

Next, we should teach additional well-known modal axioms, i.e., T, B, D, 4, and 5. From these five axioms, we also teach that we can consider 32 different combinations of the axioms, but we can reduce them substantially to 15 combinations. By the 15 combinations of axioms, we can determine 15 different modal logics. For example, we can determine **HKT**, **HKD**, **HK45**, **HS4**, and **HS5**. Thereafter, we explain some extensions of the base system. For example, if we add the axioms T, B, and 4 to the above Hilbert-style base system, it becomes Hilbert-style system for **HS5**.

Our linear algebraic approach can cover some of the above topics. In particular, many topics of Kripke semantics (item 2) and soundness of proof theory (item 3) can be covered. But the topics of syntax (item 1) and proof theory (item 3) cannot be covered. We assume that our target students have prior knowledge of propositional logic and linear algebra, and so our approach might be effective for them. In the following sections, we compare the ordinary model-theoretic approach with our linear algebraic approach.

- In Section A.4.1, we start with a calculation of the truth value of a formula. We also explain how we can verify the validity of a formula on a model in Section A.4.2.
- In Section A.5.1, we explain how to verify frame properties of a frame. In this section, we also mention that we can check whether the frame satisfies reflexivity, seriality, and symmetricity at a glance by the form of a matrix of an accessibility relation.
- In Section A.5.2, we explain how to show the correspondence between frame properties and valid formulas.

For each section, we also explain how to use our software for educational purposes. Finally, in Section A.6, we explain our teaching experiment and feedbacks from students.

A.3 Supporting Software to Teach Modal Logic

In Section 3.2.2, we regarded a calculation of the truth set of a formula and the verification of frame properties listed in Table 3.1 as an extended truth-table calculation, i.e., a computation on Boolean matrices. However, similarly to the case of the ordinary model-theoretical approach, we have to give more efforts to compute matrices if the length of a given formula becomes longer or the dimension of a matrix becomes bigger. Such efforts might be required when lecturers provide exercises or prepare teaching materials. If we wish to avoid such efforts on calculations, we had better to implement some supporting tools. In this section, we introduce our supporting software to overcome this issue. We provide an overview and a short instruction on our software in the remaining sections.

We have implemented a supporting software based on our linear algebraic reformulation of Kripke semantics by Java™ 8 programming language and opened for the public.[1] The features of our software can be summarized as follows:

1. We can edit a matrix representation of a Kripke model by a graphical user interface easily.

2. A computation program of the truth set of a formula on a model is provided. We can obtain a Boolean vector representation of the truth set of the formula written in TeX style, e.g., 'p `¥land` q.'[2] From this vector, we can obtain a truth value of the formula for each world and also verify the validity of the formula on the model.[3]

3. A verification program of frame properties is also provided. We can verify all frame properties listed in Table 3.1 at once.

4. A visualization program is provided. By the program, and we can obtain a graphical representation of a Kripke model via Graphviz.[4]

The provided programs might be helpful for educational purposes. For example, lecturers can use our software to design exercises and lecture materials. In addition, students can use our software to study modal logics by themselves. Notice that unlike `RELVIEW` tool [5], if we know how to input a formula and a model into our software, we can work with modal logic by our software without any more preparation. `RELVIEW` tool is designed to solve computation tasks of relation algebra. In order to work with modal logic by `RELVIEW` tool, we need to provide definitions of formulas and semantics of modal operators based on relational operations to `RELVIEW` tool by the internal language of it. For example, we need to define $\Box$ operator by `box(S, v) = - (S * -v)` where `S` is a matrix for an accessibility relation, `v` is a vector for valuation and the operators `-` and `*` are relational complementation and composition, respectively. However, in the context of the efficiency of computation tasks, we do not claim any superiority of our program over `RELVIEW`.

[1] `http://cirrus.jaist.ac.jp:8080/soft/bc`.

[2] As a matter of practical convenience, there are insert buttons of a proposition, logical connectives, and modal axioms at the next to the parameter box of the calculator. Hence, we can input the above vocabulary of modal logic written in TeX style into the parameter box easily.

[3] We note that our software was originally introduced in [15] to support computation tasks for dynamic logic of multi-agent communication.

[4] `http://www.graphviz.org/`

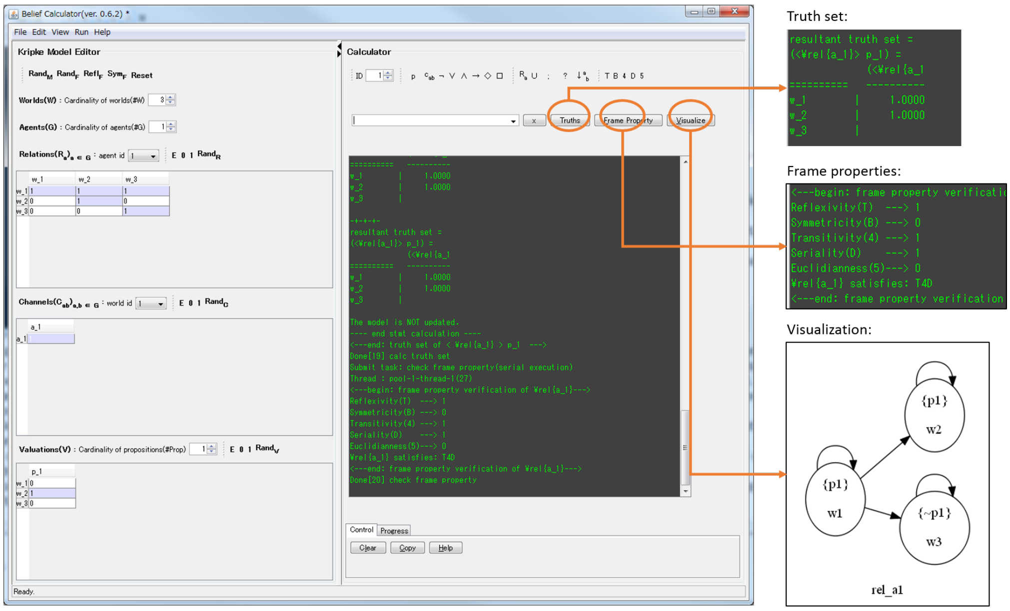


Figure A.1: Overview of Our Implementation

Figure A.1 shows a sample of the graphical user interface of our software.[5] The interface is divided into two parts. The left side of the interface is an editor for Kripke model, and the right side is a calculator for the computation tasks that we mentioned in the above list of features.

A Kripke model editor allows us to manage a model easily. The design of the editor reflects our approach; namely, we can input the model into the editor by the matrix representation of the model. A general workflow to input the parameters of the model is described as follows:

1. Input the cardinality of possible worlds and propositions into corresponding parameter boxes.

2. Input 0 or 1 into each component of matrices for relations, and valuations.[6]

As a matter of practical convenience, each component of matrices works as either a button or a text field. We may either switch the values of matrices 0 and 1 by clicking the component or enter a truth value to the component directly. Each component turns to blue if it has the value 1, white if 0. The colored matrices are helpful since these matrices allow us to recognize some (frame) properties of matrices at a glance (see Section A.5.1). In addition, there are buttons `E`, `0`, `1`, and `Rand` to set the values of each matrix as unit square matrix, zero matrix, complete matrix and randomly generated matrix, respectively.

Once parameters are entered to the editor side, we can use the calculator side to solve several computation tasks. The calculator has functions which solve tasks of the following kind:

[5]Displaying parameters are corresponding to the formula and the model in Example 3.1.

[6]Matrices of channels are used to define communication channels among agents in [15]. In this chapter, we leave the matrix to **1**, i.e., the unit square matrix, and this stands for 'every agent has communication channels each other' (cf. Figure A.1). Since this is out of focus of the present chapter, we can ignore this matrix.

1. Visualization of a Kripke model.
2. Computation of the truth set of a formula.
3. Verification of frame properties listed in Table 3.1 of a frame

The function for visualizing a Kripke model can be executed by clicking `Visualize` button on the calculator. With the help of 'Graphviz', the function yields and saves a picture of the graphical representation of the model under appropriate directory. Afterward, our software displays the picture on the screen. We will explain details and applications of the other two functions for educational purposes in the following Section A.4 (computation of a truth set of a formula) and Section A.5 (verification of frame properties), respectively.

A.4 Computation of Truth Sets and Validity

One of the most basic topics of Kripke semantics for modal logic is to calculate the truth value of a formula at a given world. In connection with this topic, we have the following topics which should be taught to students:

- The truth value of a formula at a given world
- The validity and the invalidity of a formula on a model

In this section, we explain to follow the above topics.

A.4.1 Truth Value of Formula at World

In modal logic, the truth value of a formula is computed at each possible world. In general, we explain to students how the truth value of a formula is computed by the ordinary model-theoretic approach as in Example 3.1 (Section 3.1). If a given formula is simple and the domain of a given model is small, we can calculate the truth value of a formula visually. This is one of the best points of modal logic. In this approach, we firstly draw a picture of a graphical representation of a Kripke model, and next we calculate the truth value of a formula on the picture.

For example, let us consider a Kripke model $\mathfrak{M}_1$ by $W = \{ w_1, w_2, w_3 \}$, $R = \{(w_1, w_1), (w_1, w_2), (w_1, w_3), (w_2, w_2)\}$, and $V(p) = W$ (see Figure A.2). In the model, $\Box p$ is true at every world. In order to teach the truth of $\Box p$ at a given world on a model visually, we often use the graphical representation of a Kripke model such as a picture of the model $\mathfrak{M}_1$ shown in the left side of the Figure A.2. By tracing the links of an accessibility relation from w_1 and w_2, students can obtain the truth value of $\Box p$ at w_1 and w_2, respectively. However, the graphical approach is intuitive but sometimes misleading. For example, some students might be confused how to obtain the truth value of the formula $\Box p$ at w_3. This is because the world w_3 is a 'dead-end,' i.e., a world where we cannot access any world, and so we cannot find any link to the other worlds from the picture. In such a case, we should use the ordinary model-theoretic approach. By the explanation of this approach, students eventually understand why $\Box p$ trivially holds at w_3. Namely, our goal is to show: for all $v \in W$, $w_3 R v$ implies $\mathfrak{M}_1, v \models A$. But, by definition of R, there are no world $v \in W$ such that $w_3 R v$. Therefore, the above implication is vacuously true, and we can conclude that $\Box p$ trivially holds at w_3. However, this proof might be unnatural

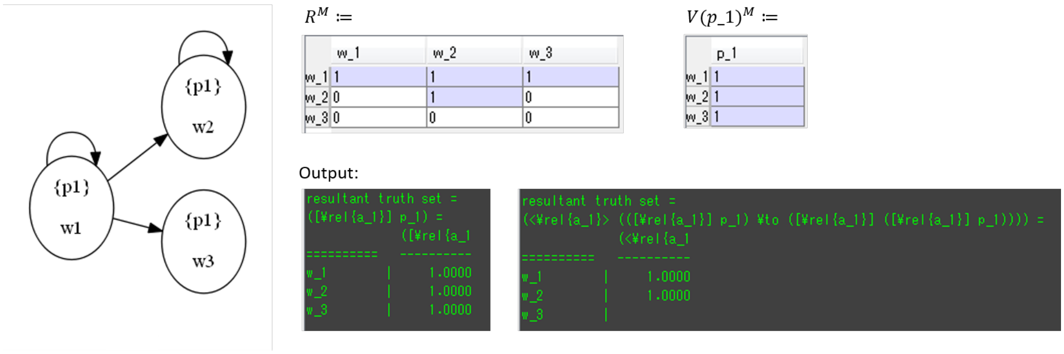


Figure A.2: The World w_3 is Now 'Dead-end'

for some students. In such case, we may also explain the proof by the negation of an assumption. Namely, we assume that $\mathfrak{M}_1, w_3 \not\models \Box p$. Then this is equivalent to:

$$\begin{array}{ll} & \text{it is not the case that for all } v \in W,\ w_3 R v \text{ implies } \mathfrak{M}_1, v \models A, \\ \text{iff} & \text{for some } v \in W,\ \text{it is not the case that } w_3 R v \text{ implies } \mathfrak{M}_1, v \models A, \\ \text{iff} & \text{for some } v \in W,\ w_3 R v \text{ and } \mathfrak{M}_1, v \not\models A. \end{array}$$

Hence, we obtain $w_3 R v$ for some $v \in W$. But there is no world $v \in W$ such that $w_3 R v$ by definition of R, a contradiction. Therefore, the graphical approach sometimes might not work well, and the model-theoretic approach gives us the more rigorous explanation. However, we need to rely on the notion of 'vacuously hold' or the argument by contradiction in this approach.

On the other hand, if we employ our linear algebraic approach to obtain the truth value of the above formula, we do not need to rely on such notions explicitly. As we mentioned in Section 3.2.1, we can compute the truth value of a formula by an extended truth-table calculation, i.e., a Boolean matrix calculation. In order to teach how to obtain the truth value of $\Box p$ at w_3 by our approach, at first we should show the following matrix representations of R and $V(p)$ to students:

$$R^M := \begin{bmatrix} 1 & 1 & 1 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}, V(p)^M := \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}.$$

Then, we can teach the computation of the truth set of $\Box p$ by:

$$\|\Box p\| = \overline{R^M \overline{V(p)^M}} = \overline{\begin{bmatrix} 1 & 1 & 1 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \overline{\begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}}} = \overline{\begin{bmatrix} 1 & 1 & 1 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}} = \overline{\begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}.$$

Afterward, we can extract the computation of w_3 from the above computation as:

$$\overline{\begin{bmatrix} 0 & 0 & 0 \end{bmatrix} \overline{\begin{bmatrix} 1 \end{bmatrix}}} = \overline{\begin{bmatrix} 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} 0 \end{bmatrix}} = \overline{\begin{bmatrix} 0 \end{bmatrix}} = \begin{bmatrix} 1 \end{bmatrix}.$$

We may also explain that the truth value of $\Box p$ eventually must be true at the dead-end world since $\overline{\begin{bmatrix} 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x \end{bmatrix}}$ (where x is a truth value of p at the dead-end world) always returns

1. As we can see above, we can compute the truth set of a formula using our approach easily. On the other hand, in order to obtain the truth set of a formula, in the ordinary model-theoretic approach we need to calculate the truth value of the formula for every world.

Furthermore, we may use our software to obtain the truth set of a formula quickly. Our software provides a function to compute the truth set of a formula. Inputs of the function are matrices of a Kripke model and a formula written in TEX style, e.g., 'p `¥land` q.' An output is a vector corresponding to the desired truth set. We can compute the truth set of a given formula by clicking the button `Truths` on the calculator. If the computation procedure finishes successfully, a resultant vector appears on a terminal window. See the bottom side of the Figure A.2. We can find the vector $\|\Box p\| = \mathbf{1}$ of the truth set. We can also find the more involved computation result in the bottom right side of the figure, i.e., the vector $\|\Diamond(\Box p \to \Box\Box p)\| = {}^{t}\left[110\right]$ of the truth set. Since the function yields intermediate computation results, it can be helpful for educational purposes. For example, students can use this function for their self-study. Lecturers can also use this function to provide exercises and write lecture materials. In addition, we can obtain the following solutions from the vector of the truth set of a formula:

1. the truth value of a formula at a world.
2. the validity of a formula on a model.

That is, if the n-th component of the vector is 1, the formula is true at n-th world (item 1). In the right side of the Figure A.2, we can find the truth value 1 of a formula $\Box p$ at the row of w_3 of the vector. If the vector is filled with 1, the formula is valid on a model (item 2). Otherwise, the formula is invalid on the model. For the validity and the invalidity, we explain them in Section A.4.2.

A.4.2 Validity and Invalidity of Formula on Model

In connection with the topic of the truth value of a formula, the validity of a formula on a model is another important topic which should be taught to students. This is because the notion of the validity of a formula is used to explain the notions of a counter-model to a formula, the satisfiability of a formula on a model, and the correspondence between frame properties and (valid) formulas.

In the ordinary model-theoretic approach, we explain that a formula is valid on a model if the formula is true at every world. For example, let us recall the model $\mathfrak{M}_1$ which we used in Section A.4.1 (see also Figure A.2) and verify whether the formula $\Box p$ is valid on the model. After the calculation of the truth value of the formula $\Box p$ for each world, we obtain $\Box p$ is true at every world, and can conclude that $\Box p$ is valid on the model $\mathfrak{M}_1$. During the above calculation, we have to repeat the similar argument.

In the linear algebraic approach, we may explain that a formula is valid if a vector of the truth set of a formula is $\mathbf{1}$, in other words, the vector does not contain 0. For example, through the calculation of the truth set $\|\Box p\|$ on the above model $\mathfrak{M}_1$, we obtain the vector $\mathbf{1}$. Therefore, we can conclude that the formula $\Box p$ is valid on the model. Let us suppose a model $\mathfrak{M}'_1$ by the model $\mathfrak{M}_1$ where $V(q) = \{ w_2 \}$. We can also show that $\Diamond(p \lor q) \leftrightarrow \Diamond p \lor \Diamond q$ is valid on $\mathfrak{M}'_1$. It suffices to show $[\![\Diamond(p \lor q)]\!] = [\![\Diamond p \lor \Diamond q]\!]$ (cf. Section 2.1.1). Since

$$\|\Diamond(p \lor q)\| = R^M(V(p)^M V(q)^M) = (R^M V(p)^M) + (R^M V(q)^M) = \|\Diamond p \lor \Diamond q\|,$$

we obtain $\|\Diamond(p \vee q)\| = \|\Diamond p \vee \Diamond q\|$, i.e., $[\![\Diamond(p \vee q)]\!] = [\![\Diamond p \vee \Diamond q]\!]$, therefore the commutativity of $\Diamond$ over disjunction is reduced to the distributivity of matrix multiplication. Moreover, we can also show $[\![\Diamond\bot]\!] = [\![\bot]\!]$ by $\|\Diamond\bot\| = R^M\mathbf{0} = \mathbf{0} = \|\bot\|$.

From the notion of the validity of a formula on a model, we should also teach invalidity of that on the model. That is, a formula is invalid on a model if the formula is not valid on the model. In other words, there is some world such that a formula is not true. For example, let us define a model $\mathfrak{M}_2$ by $W = \{ w_1, w_2, w_3 \}$, $R = \{ (w_1, w_1), (w_1, w_2), (w_1, w_3), (w_2, w_2), (w_3, w_3) \}$ and $V(p) = \{ w_2 \}$. The model is the same as the model that we explained in Example 3.1 (Section 3.1). Then, the formula $\Box p$ is no longer valid on the model $\mathfrak{M}_2$ since $\Box p$ is false at w_1 and w_3. Therefore, $\Box p$ is invalid on the model $\mathfrak{M}_2$. In the model-theoretic approach, we need to find such worlds w_1 or w_3 by the calculation of the truth value of the formula for each world. Although in the linear algebraic approach, we only need to find 0 in the vector of the truth set of the formula, i.e.,

$$\overline{\begin{bmatrix}1&1&1\\0&1&0\\0&0&1\end{bmatrix}\overline{\begin{bmatrix}0\\1\\0\end{bmatrix}}} = \overline{\begin{bmatrix}1&1&1\\0&1&0\\0&0&1\end{bmatrix}\begin{bmatrix}1\\0\\1\end{bmatrix}} = \overline{\begin{bmatrix}1\\0\\1\end{bmatrix}} = \begin{bmatrix}0\\1\\0\end{bmatrix}.$$

Since the vector of the truth set contains 0, we can conclude that the formula $\Box p$ is invalid on the model $\mathfrak{M}_2$. If we use our software, we can check the validity of a formula easily. As we mentioned in Section A.4.1, we can compute the truth set of a formula on a model, and from the resultant vector we can check the validity of the formula on the model. For example, in Figure A.2, the resultant vector of the truth set of the formula $\Box p$ does not contain 0. Therefore, the formula is valid on the model of the figure.

From the invalidity of a formula, we may also explain that a model is a counter-model to the formula. We say that a model $\mathfrak{M}$ is a counter-model to a formula A if A is invalid on the model $\mathfrak{M}$. For example, the above model $\mathfrak{M}_2$ is a counter-model to the formula $\Box p$. Of course we can investigate the validity of a formula on the larger model easily. Such investigation can be a good exercise to some students who want to study finite model checking.

A.5 Verification of Frame Properties

As we mentioned in Section A.4, the truth value of a formula is determined for each possible world. In particular, if the formula contains modal operators, the resultant truth value is affected by the properties of a given accessibility relation. Therefore, it is important to explain topics of various properties of frames to students. In this section, we explain the following topics that should be taught to students:

1. The verification of frame properties of a given frame.

2. The validity of a formula on a frame which satisfies one of the frame properties listed in Table 3.1.

A.5.1 Frame Properties on Frame

After giving a brief introduction to frame properties of a frame visually, we should explain how to verify them. The ordinary approach to teach the verification of frame properties

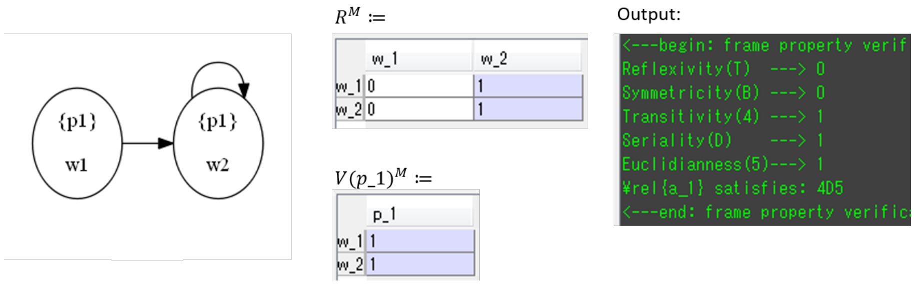


Figure A.3: The Frame Satisfies Seriality, Transitivity and Euclideanness

is to use both the visual and the model-theoretic approach. For example, let us define a frame $\mathfrak{F}$ by $W = \{ w_1, w_2 \}, R = \{ (w_1, w_2), (w_2, w_2) \}$ (see Figure A.3). The frame $\mathfrak{F}$ satisfies seriality, transitivity and Euclideanness. Since the cardinality of the accessibility relation as a set is enough small, we can use the graphical approach to explain that the frame satisfies the above frame properties. If we show a graphical representation of the frame to students, they might easily realize that the frame satisfies seriality and transitivity. However, it might be difficult to realize if the frame satisfies Euclideanness. In such case, we should switch our explanation to the model-theoretic approach. When we show the frame satisfies Euclideanness, we should check whether the frame satisfies the frame condition of the frame property, i.e., wRv and wRu imply vRu for any $w, v, u \in W$. By definition of the accessibility relation R, we have the following implications:

- $w_1 R w_2$ and $w_1 R w_2$ imply $w_2 R w_2$ ($w = w_1$, $v = w_2$, $u = w_2$).
- $w_2 R w_2$ and $w_2 R w_2$ imply $w_2 R w_2$ ($w = w_2$, $v = w_2$, $u = w_2$).

The other implications, e.g., $w_1 R w_1$ and $w_1 R w_1$ imply $w_1 R w_1$ ($w = w_1$, $v = w_1$, $u = w_1$), trivially hold since the antecedent of the implication is false by definition of R. Therefore, we can conclude that the frame $\mathfrak{F}$ satisfies wRv and wRu imply vRu for any $w, v, u \in W$, i.e., Euclideanness.

If we teach how to verify the frame properties of a frame by our linear algebraic approach, we should show the verification of matrix reformulation of a frame property. Similarly to Example 3.8 (Section 3.2.2), we can explain the verification of Euclideanness of the above frame $\mathfrak{F}$ by $R = {}^tRR + R$, i.e.,

$$\begin{aligned}
\begin{bmatrix} 0 & 1 \\ 0 & 1 \end{bmatrix} &= \begin{bmatrix} 0 & 0 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} 0 & 1 \\ 0 & 1 \end{bmatrix} + \begin{bmatrix} 0 & 1 \\ 0 & 1 \end{bmatrix} \\
&= \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix} + \begin{bmatrix} 0 & 1 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 1 \end{bmatrix}.
\end{aligned}$$

In addition, we may also mention that the more simplified method to verify some frame properties of a frame. We can check whether a given frame satisfies reflexivity, seriality, and symmetricity by the form of a matrix of an accessibility relation of the frame. This is because if the frame satisfies the above three properties, then the matrix of an accessibility relation has the following features:

- Reflexivity: every diagonal component of the matrix consist of 1.

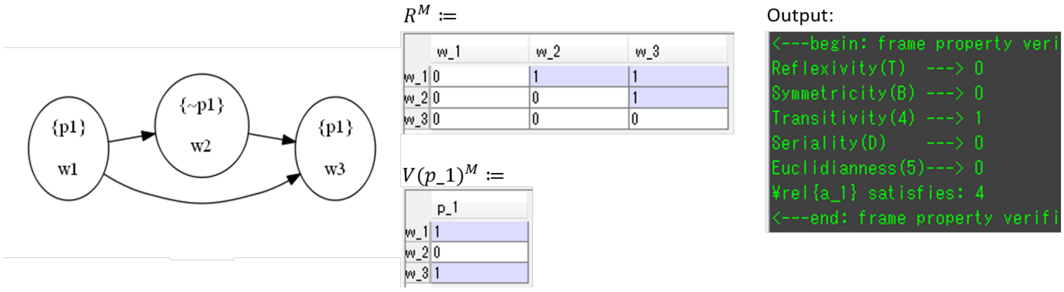


Figure A.4: How to Satisfy both Transitivity and Seriality by Adding an Edge?

- Seriality: every row contains at least one occurrence of 1.
- Symmetricity: a matrix is a mirror image in the diagonal line.

For example, we can recognize that the matrix R of $\mathfrak{F}$ in Figure A.3 satisfies seriality but not reflexivity and symmetricity at a glance.

We may also teach the above matrix computation with our software, which has a function to verify every frame properties listed in Table 3.1 (Section 3.2.2). An input parameter of the function is a matrix of an accessibility relation of a frame, and an output is a list of the verification results of each property. After entering the input parameters, it is ready to verify the property. If the dimension of the matrix of an accessibility relation is enough small, we can also use the simplified method to verify some frame properties as we mentioned before. Since the color of each component is blue if it is 1, we can easily recognize whether the matrix satisfies the above features of frame properties. For example, see the matrix of R^M at the center of Figure A.3. We can observe that the matrix actually satisfies seriality since every row contains a blue component, i.e., 1. If we wish to use the function of verification of frame properties, we can execute it by clicking the button `Frame Property` on the calculator. If the function finishes successfully, a resultant list will be displayed in a terminal window (see right side of Figure A.3). In the list, if the given frame satisfies a frame property, 1 appears at the right side of the name of the property, otherwise, 0 appears. In addition, the name of the modal axiom also appears at the next to the name of the corresponding frame property. At the right side of the Figure A.3, we can see that the frame satisfies seriality, transitivity and Euclideanness.

Since we can easily manipulate an input matrix of a model by the model editor and obtain the result of the verification of frame properties of the model quickly, for example, we can design or solve the following exercises.

Example A.1. *Suppose a model of Figure A.4 which satisfies transitivity.*

1. *In order to satisfy both transitivity and seriality, which edge should we add to the model?* (*answer: add an edge from w_3 to itself.*)
2. *In order to satisfy Euclideanness, how should we modify the model?* (*answer: delete every edge from the model.*)
3. *How to remove every frame property from the model by one edge deletion?* (*answer: delete an edge from w_1 to w_3.*)

If we try to compute possible solutions of the above exercises without supporting tools, we have to compute frame conditions with various changes over and over again. Therefore, our software might be helpful to avoid such efforts.

A.5.2 Frame Properties and Valid Formulas

In this section, we focus our attention on the modal axioms T, B, D, 4 and 5. If we know whether a given frame satisfies some properties, we can determine which formulas are valid. For example, each frame property listed in Table 3.1 (Section 3.2.2) has the corresponding formula. If a model satisfies seriality, then the corresponding formula D ($\Box p \to \Diamond p$) is valid on the model. In order to explain the correspondence between frame properties and valid formulas, we use both verification of the validity of a formula on a model in Section A.4.2 and of frame properties of a frame in Section A.5.1.

For example, let us recall the frame $\mathfrak{F}$ of previous Section A.5.1, and define a model $\mathfrak{M}_3$ by $\mathfrak{F}$ and $V(p) = W$ (see Figure A.3). The model satisfies seriality, transitivity, and Euclideanness. In addition, the formulas D ($\Box p \to \Diamond p$) for seriality, 4 ($\Box p \to \Box\Box p$) for transitivity and 5 ($\Diamond p \to \Box\Diamond p$) for Euclideanness are valid on the model $\mathfrak{M}_3$, respectively. In order to teach the the correspondence between frame properties and valid formulas smoothly, the above properties are sometimes provided as an assumption. Otherwise, we should start to explain from the verification of all frame properties listed in Table 3.1 of the frame $\mathfrak{F}$. In this example, we suppose that the above properties are given. Then, we can start our explanation from the verification of the validity of the above formulas. To make our discussion simpler, we focus our attention on the validity of the formula 5 ($\Diamond p \to \Box\Diamond p$) only. Under the model-theoretic approach, we have to check the truth value of the formula 5 for each world. Through the similar discussion in Section A.4.2, we can eventually conclude that the formula 5 is valid on the model $\mathfrak{M}_3$. Similarly, we may explain the validity of the formula by our linear algebraic approach. By

$$
\begin{aligned}
\|\Diamond p \to \Box\Diamond p\| &= \overline{\|\Diamond p\|} + \|\Box\Diamond p\| \\
&= \overline{R^M V(p)^M} + \overline{R^M(\overline{R^M V(p)^M})} \\
&= \overline{\begin{bmatrix} 01 \\ 01 \end{bmatrix}\begin{bmatrix} 1 \\ 1 \end{bmatrix}} + \overline{\begin{bmatrix} 01 \\ 01 \end{bmatrix}\left(\overline{\begin{bmatrix} 01 \\ 01 \end{bmatrix}\begin{bmatrix} 1 \\ 1 \end{bmatrix}}\right)} \\
&= \overline{\begin{bmatrix} 1 \\ 1 \end{bmatrix}} + \overline{\begin{bmatrix} 01 \\ 01 \end{bmatrix}\overline{\begin{bmatrix} 1 \\ 1 \end{bmatrix}}} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} + \overline{\begin{bmatrix} 0 \\ 0 \end{bmatrix}} = \begin{bmatrix} 1 \\ 1 \end{bmatrix},
\end{aligned}
$$

we can conclude that the formula 5 is valid on the model $\mathfrak{M}_3$. In a similar manner, we can also verify whether the formulas 4 ($\Box p \to \Box\Box p$) and D ($\Box p \to \Diamond p$) are valid on the model, respectively. However, we need to give the effort to calculate matrices, thus we may use our software to check the validity of the formulas quickly.

We should also explain that if a formula which defines a frame property is not valid, then the corresponding frame property is not satisfied in the frame of the model. Remark that we still focus on the axioms T, B, D, 4 and 5. For example, let us define a model $\mathfrak{M}_4$ by $W = \{ w_1, w_2 \}$, $R = \{ (w_1, w_2) \}$ and $V(p) = W$ (see Figure A.5). Then the frame of the model $\mathfrak{M}_4$ does not satisfy seriality and Euclideanness since the formulas D ($\Box p \to \Diamond p$) and 5 ($\Diamond p \to \Box\Diamond p$) are no longer valid on the model $\mathfrak{M}_4$, respectively. To

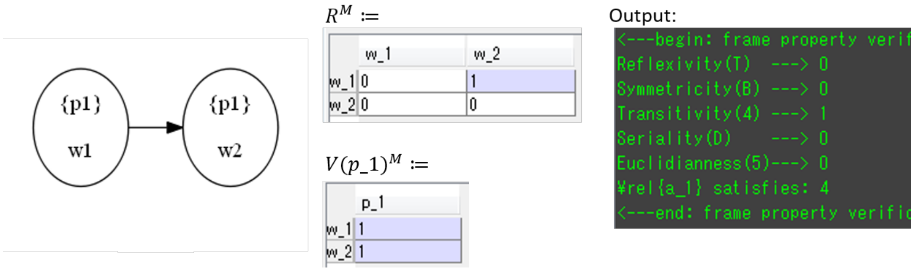


Figure A.5: The Formula 5 ($\Diamond p \to \Box\Diamond p$) is Invalid and the Frame Does Not Satisfy Euclideanness

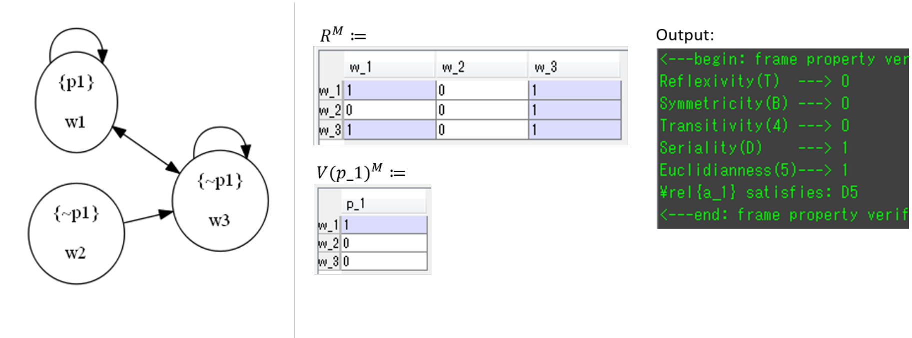


Figure A.6: Which Frame Properties are Satisfied on the Frame?

show that the frame does not satisfy Euclideanness, we should find a link which violates the frame condition of the frame property. By the ordinary model-theoretic approach, we can find that the following implication does not hold: $w_1 R w_2$ and $w_1 R w_2$ imply $w_2 R w_2$ ($w := w_1$, $v := w_2$, $u := w_2$). Therefore, the frame does not satisfy Euclideanness. The result is also the same in the linear algebraic approach by $R \neq {}^t RR + R$, i.e.,

$$\begin{aligned}\begin{bmatrix}0&1\\0&0\end{bmatrix} &\neq \begin{bmatrix}0&0\\1&0\end{bmatrix}\begin{bmatrix}0&1\\0&0\end{bmatrix}+\begin{bmatrix}0&1\\0&0\end{bmatrix}\\ &= \begin{bmatrix}0&0\\0&1\end{bmatrix}+\begin{bmatrix}0&1\\0&0\end{bmatrix} = \begin{bmatrix}0&1\\0&1\end{bmatrix}.\end{aligned}$$

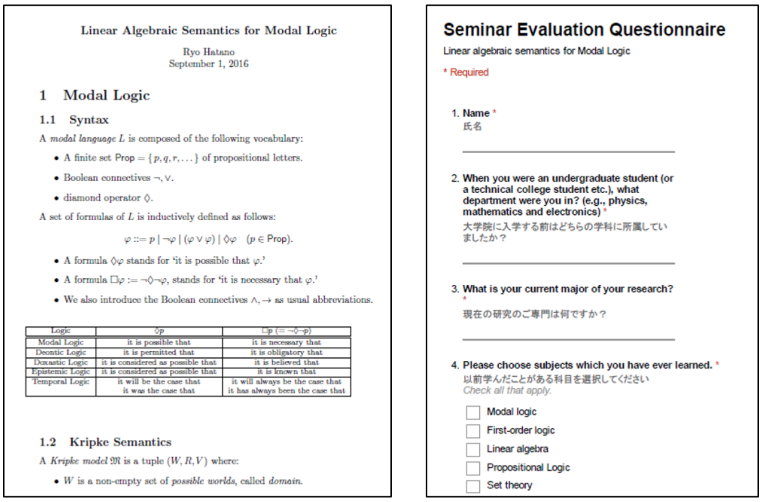
The above calculation seems easy but it takes a bit of our time. Hence, we may use our software to verify which frame properties are satisfied on the frame quickly.

At the end, for example, we can design the following exercise with the help of our software.

Example A.2. *Let us define a Kripke model by* $W = \{ w_1, w_2, w_3 \}, R = \{(w_1, w_1), (w_1, w_3), (w_2, w_3), (w_3, w_1), (w_3, w_3)\}$ *and* $V(p_1) = \{ w_1 \}$ *(see Figure A.6).*

1. *Enumerate satisfying frame properties (listed in Table 3.1). (answer: the frame satisfies seriality and Euclideanness.)*

2. *Verify whether the formula 5 ($\Diamond p_1 \to \Box\Diamond p_1$) is valid on the model. (answer: 5 is valid on the model.)*



Linear Algebraic Semantics for Modal Logic

Ryo Hatano
September 1, 2016

1 Modal Logic

1.1 Syntax

A *modal language* L is composed of the following vocabulary:

- A finite set $\mathsf{Prop} = \{p, q, r, \dots\}$ of propositional letters.
- Boolean connectives $\neg, \vee$.
- diamond operator $\Diamond$.

A set of formulas of L is inductively defined as follows:

$$\varphi ::= p \mid \neg\varphi \mid (\varphi \vee \varphi) \mid \Diamond\varphi \quad (p \in \mathsf{Prop}).$$

- A formula $\Diamond\varphi$ stands for 'it is possible that φ.'
- A formula $\Box\varphi := \neg\Diamond\neg\varphi$, stands for 'it is necessary that φ.'
- We also introduce the Boolean connectives $\wedge, \rightarrow$ as usual abbreviations.

Logic	$\Diamond p$	$\Box p \; (= \neg\Diamond\neg p)$
Modal Logic	it is possible that	it is necessary that
Deontic Logic	it is permitted that	it is obligatory that
Doxastic Logic	it is considered as possible that	it is believed that
Epistemic Logic	it is considered as possible that	it is known that
Temporal Logic	it will be the case that it was the case that	it will always be the case that it has always been the case that

1.2 Kripke Semantics

A *Kripke model* $\mathfrak{M}$ is a tuple (W, R, V) where:

- W is a non-empty set of *possible worlds*, called *domain*.

Seminar Evaluation Questionnaire

Linear algebraic semantics for Modal Logic

* Required

1. Name *
   氏名
2. When you were an undergraduate student (or a technical college student etc.), what department were you in? (e.g., physics, mathematics and electronics) *
   大学院に入学する前はどちらの学科に所属していましたか？
3. What is your current major of your research? *
   現在の研究のご専門は何ですか？
4. Please choose subjects which you have ever learned. *
   以前学んだことがある科目を選択してください
   Check all that apply.
   - [ ] Modal logic
   - [ ] First-order logic
   - [ ] Linear algebra
   - [ ] Propositional Logic
   - [ ] Set theory

Figure A.7: Our Lecture Material and Questionnaire Form

3. *Let us delete an edge from w_3 to w_1. Afterward, verify again the formula* 5 *on the model.* (*answer:* 5 *is invalid on the model.*)

4. *Verify whether the frame satisfies Euclideanness.* (*answer: the frame does not satisfy Euclideanness.*)

When we teach modal logic, we sometimes need to consider involved exercises as above. With the help of some supporting tools, we can avoid our efforts by hand and eliminate human errors from our teaching materials.

A.6 Feedbacks from Students

In order to obtain feedbacks from students, we have held a small seminar to teach elementary topics of modal logic using our approach. The participants of this seminar were 15 graduate students from our university. For reference, we have opened our lecture material and feedbacks from students for the public (see also Figure A.7 and Appendix B).[7] In the lecture, we taught the following topics to the students:

1. Truth-table calculation of propositional logic, how to read modal operators, how to define formulas, and the dual-definition of modal operators.

2. A graphical representation of Kripke model, linear algebraic reformulation of Kripke semantics, and computation of truth sets and validity of a given formula on a model.

3. Matrix representation of the five frame properties in Table 3.1 (reflexivity, symmetricity, transitivity, seriality and Euclideanness), verification of them, and the correspondence between these frame properties and formulas (T, B, D, 4 and 5).

These topics are selected from teaching topics explained in Sections A.2, A.4 and A.5. We used Examples 3.1, 3.3 and 3.8, and examples of the truth value of a formula $\Box p$

[7] `http://cirrus.jaist.ac.jp:8080/soft/ttl`

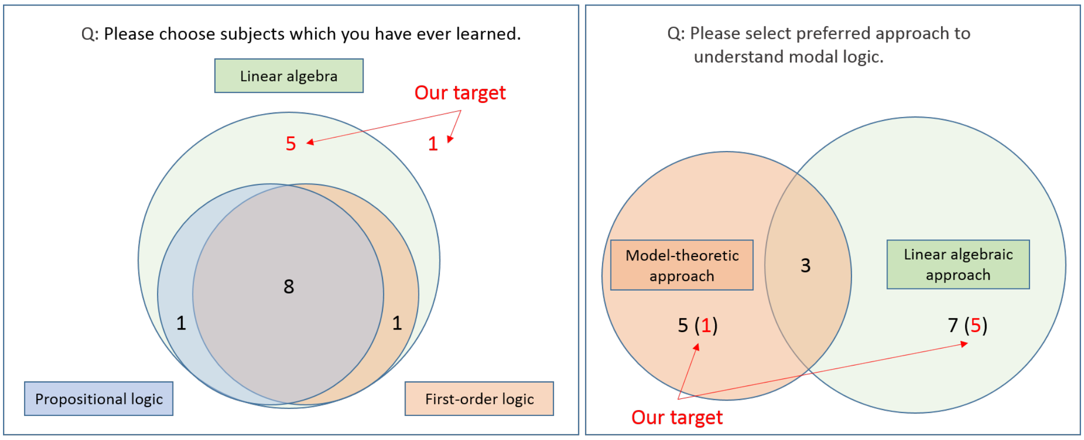


Figure A.8: Prior Knowledge of Students and Their Preferred Approaches

at a dead-end world in Section A.4.1 and the verification of Euclideanness property in Section A.5.1. In order to compare our linear algebraic approach with the ordinary model-theoretic approach, we also provide a short explanation of the model-theoretic approach for each topic. Finally, we demonstrated our supporting software.

After the seminar, we have conducted a survey using a questionnaire form (see the right side of Figure A.7). The questionnaire form consists of the following items:

1. Past and current affiliation, past and current major of research and subjects which he/her has ever learned.
2. Levels of understanding of the topics and his/her intriguing topics.
3. Preferred approach to learn modal logic.
4. Effectiveness of our approach and supporting software to learn modal logic.

We have collected 15 completed questionnaire forms, and the results and opinions can be summarized as follows. For item 1, we found that 14 students have already learned linear algebra, and 6 students have never learned first-order logic (see the left diagram of Figure A.8).[8] In what follows, we regard the latter 6 students (i.e., one-third of the participants) as our target students since we taught truth-table calculation at the beginning of the seminar. For item 2, most of our students answered that they could understand the topics of the seminar (see the left graph of Figure A.9). In particular, the topic of the verification of frame properties using matrices attracts the interest of 12 students (see the right graph of Figure A.9). For item 3, the results were different from our expectation. Our approach was bit preferred than the model-theoretic approach; 7 students preferred to use linear algebraic approach, 5 students preferred to use the model-theoretic approach and 3 students preferred to use both approaches (see the right diagram of Figure A.8). In particular, 5 students of our target preferred the linear algebraic approach. Their opinion is that simple matrix calculations allow them to understand the elementary notion of

[8] In the questionnaire, we also asked a question that whether students know modal logic and set theory. As a result, we found that 6 students and 4 students have already learned modal logic and set theory, respectively. Since these students also have already learned first-order logic, we have merged them into the same group of students who know first-order logic in the left diagram of Figure A.8.

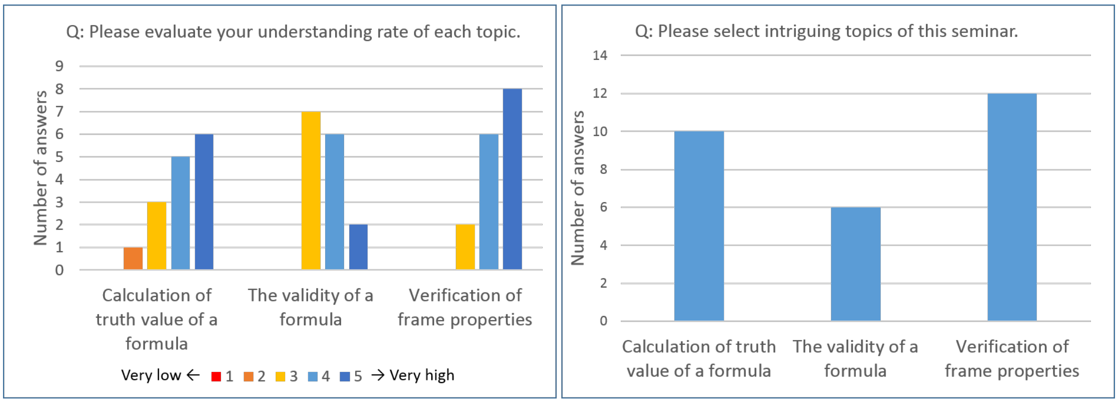


Figure A.9: Interesting Topics to Students

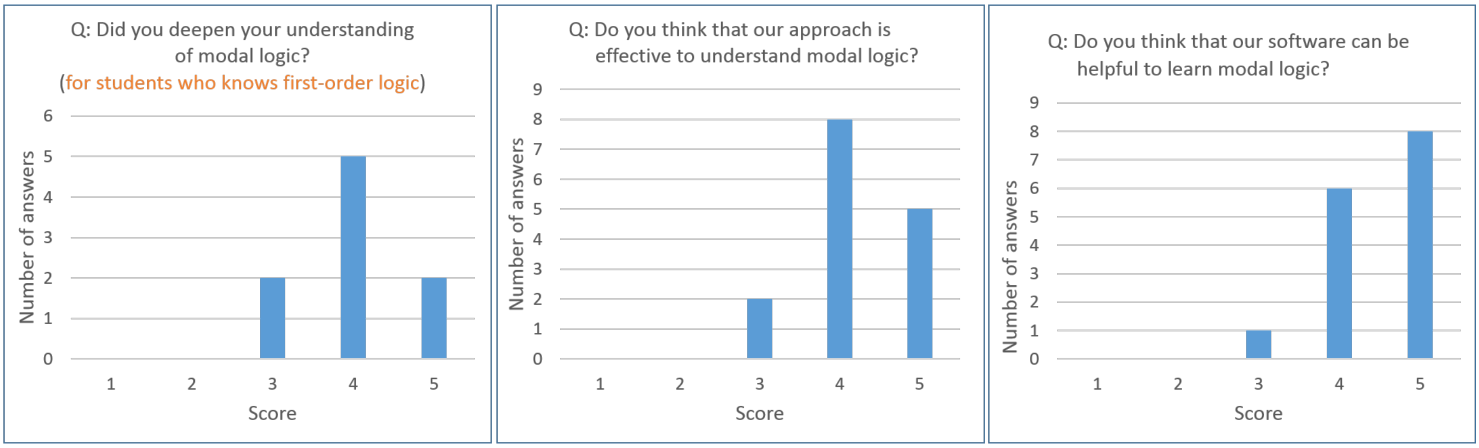


Figure A.10: Effectiveness of Our Teaching Approach and Software to Learn Modal Logic

modal logic since they are not familiar with set theory and felt difficult to understand the model-theoretic treatment of Kripke semantics. This result indicates that our target students could learn elementary part of modal logic using our approach. On the other hand, 7 students who have already learned first-order logic were divided into two groups; 4 students preferred model-theoretic approach and 3 students preferred both approaches. But these 7 students also answered that they could deepen their understanding of modal logic from the linear algebraic perspective (see the left graph of Figure A.10). For item 4, most of our students are agreed that our approach is effective to learn modal logic (see the center graph of Figure A.10). They also answered that our software can be helpful to their study since they wished to avoid involved calculation of matrices (see the right graph of Figure A.10).

As a result, students eventually got a positive impression to learn modal logic using our approach. The above feedbacks indicate that our approach must be efficient for both our target students and those who have already learned first-order logic. The result also indicates that our approach has the potential of expanding the range of our target students. For example, in Japan, many high-school students learn basic calculations of real-valued matrices. Therefore, we may teach elementary topics of modal logic and graph theory to advanced high school students in terms of matrices.

A.7 Concluding Remarks

In this chapter, we have also investigated educational applications based on our technical contributions. We can summarize our results as following two items.

Teaching Modal Logic from The Linear Algebraic Viewpoint:
In connection with our technical contribution (A1), we have also studied a linear algebraic approach to teach modal logic to students in Chapter 3. Based on our linear algebraic approach, we can teach many elementary topics of Kripke semantics for modal logics by simple calculations over Boolean matrices. Our target students are those who have prior knowledge of linear algebra and propositional logic. They can learn modal logics based on their acquired knowledge without learning first-order logic. In addition, students who have already learned first-order logic, can also deepen their understanding of the subject from a different perspective. In order to claim this, we have explained which topics of modal logics can be taught to students using our approach and why the approach is helpful for educational purposes (Appendix A.2). Finally, we have taught some elementary topics of modal logic to our students using our approach, and collected feedbacks from them (Appendix A.6). The feedbacks indicates that our approach must be efficient for both our target students and the ordinary target students of a course of modal logic.

Supporting Software and Algorithms for Linear Algebraic Semantics:
In connection with our technical contributions (A1) and (A3), we have developed a supporting software based on our linear algebraic reformulation of Kripke semantics for modal logic (Appendix A.3). Our software allows us to manipulate a matrix representation of Kripke model easily and to solve computation tasks such as the calculation of truth set of a formula, verification of frame properties of a given frame, or visualization of Kripke model. In addition to modal logic, our software also supports to handle **PDL**-extension of doxastic logic with communication channels and semi-private announcement operator. Since we have opened our software including source code for the public, it might be helpful for education or further studies.

Appendix B

Lecture Material for Teaching Linear Algebraic Semantics

Linear Algebraic Semantics for Modal Logic

Ryo Hatano

B.1 Modal Logic

B.1.1 Syntax

A *modal language* L is composed of the following vocabulary:

- A finite set $\mathsf{PROP} = \{ p, q, r, \dots \}$ of propositional letters.
- Boolean connectives $\neg, \vee$.
- diamond operator $\Diamond$.

A set of formulas of L is inductively defined as follows:

$$A ::= p \mid \neg A \mid (A \vee A) \mid \Diamond A \quad (p \in \mathsf{PROP}).$$

- A formula $\Diamond A$ stands for 'it is possible that A.'
- A formula $\Box A := \neg\Diamond\neg A$, stands for 'it is necessary that A.'
- We also introduce the Boolean connectives $\wedge, \rightarrow$ as usual abbreviations.

Logic	$\Diamond p$	$\Box p$ $(= \neg\Diamond\neg p)$
Modal Logic	it is possible that	it is necessary that
Deontic Logic	it is permitted that	it is obligatory that
Doxastic Logic	it is considered as possible that	it is believed that
Epistemic Logic	it is considered as possible that	it is known that
Temporal Logic	it will be the case that it was the case that	it will always be the case that it has always been the case that

B.1.2 Kripke Semantics

A *Kripke model* $\mathfrak{M}$ is a tuple (W, R, V) where:

- W is a non-empty set of *possible worlds*, called *domain*.
- $R \subseteq W \times W$ is an *accessibility relation*.
- $V : \mathsf{PROP} \to \mathcal{P}(W)$ is a *valuation function*.

A *frame* is the result of dropping a valuation function from a model, i.e., (W, R). Given any model $\mathfrak{M} = (W, R, V)$ and any possible world $w \in W$, the *satisfaction relation* $\mathfrak{M}, w \models A$ is defined inductively as follows:

$$\begin{array}{lll} \mathfrak{M}, w \models p & \text{iff} & w \in V(p), \\ \mathfrak{M}, w \models \neg A & \text{iff} & \mathfrak{M}, w \not\models A, \\ \mathfrak{M}, w \models A \vee B & \text{iff} & \mathfrak{M}, w \models A \text{ or } \mathfrak{M}, w \models B, \\ \mathfrak{M}, w \models \Diamond A & \text{iff} & \text{for some } v \in W, wRv \text{ and } \mathfrak{M}, v \models A. \end{array}$$

- A *truth set* $[\![A]\!]_{\mathfrak{M}}$ is defined by $[\![A]\!]_{\mathfrak{M}} = \{\, w \in W \mid \mathfrak{M}, w \models A \,\}$.
- A is *valid* on a model $\mathfrak{M}$ if $\mathfrak{M}, w \models A$ for all worlds $w \in W$.

Example B.1. *We define a Kripke model $\mathfrak{M}$ by:*

$$\begin{array}{rcl} W & = & \{\, w_1, w_2, w_3 \,\}, \\ R & = & \{(w_1, w_1), (w_1, w_2), (w_1, w_3), (w_2, w_2), (w_3, w_3)\}, \\ V(p) & = & \{\, w_2 \,\}. \end{array}$$

It is clear that $\Diamond p$ is true at w_1 (and w_2), i.e., $\mathfrak{M}, w_1 \models \Diamond p$.

Proof. (The ordinary model-theoretic approach) By

$$\begin{array}{lll} \mathfrak{M}, w_1 \models \Diamond p & \text{iff} & \text{For some } v \in W(w_1 Rv \text{ and } \mathfrak{M}, v \models p), \\ & \text{iff} & \text{For some } v \in W(w_1 Rv \text{ and } v \in V(p)), \\ & \text{iff} & \text{For some } v \in W(w_1 Rv \text{ and } v \in \{\, w_2 \,\}), \end{array}$$

it suffices to know if $(w_1, w_2) \in R$. Trivial. $\square$

B.2 Linear Algebraic Semantics

B.2.1 Boolean Matrix

Here, we use the symbol M, to denote a *Boolean matrix*, i.e., each element of the matrix belongs to the set $\{\, 0, 1 \,\}$.

- The superscript ^{M} with a symbol or expressions (e.g., X^M and $(X + Y)^M$) denotes a matrix representation of them.
- $M(m \times n)$ denotes the set of all (Boolean) $m \times n$ matrices, where m and n are the numbers of rows and columns, respectively.

- $M(i,j)$ denotes the element in the i-th row and j-th column entry.
- The Boolean operations of '$+$', '$\cdot$' and '$^-$' for the component of Boolean matrices correspond to the logical operations of '$\vee$', '$\wedge$' and '$\neg$.'

Let $M, M_1, M_2 \in M(m \times n), M_3 \in M(m \times l)$ and $M_4 \in M(l \times n)$.

$$\begin{array}{lcl}
{}^tM(i,j) & := & M(j,i), \\
\overline{M}(i,j) & := & \overline{M(i,j)}, \\
(M_1 + M_2)(i,j) & := & M_1(i,j) + M_2(i,j), \\
(M_3M_4)(i,j) & := & \sum_{1 \le k \le n}(M_3(i,k) \cdot M_4(k,j)).
\end{array}$$

B.2.2 Kripke Semantics in Matrices

Let $\mathfrak{M} = (W, R, V)$ be a Kripke model and $W = \{ w_1, w_2, \dots, w_m \}$. Matrix representations of an accessibility relation $R^M \in M(m \times m)$ and a valuation $V(p)^M \in M(m \times 1)$ are defined by

$$R^M(i,j) = \begin{cases} 1 & \text{if } (w_i, w_j) \in R, \\ 0 & \text{if } (w_i, w_j) \notin R. \end{cases} \qquad V(p)^M(i) = \begin{cases} 1 & \text{if } w_i \in V(p), \\ 0 & \text{if } w_i \notin V(p). \end{cases}$$

The semantic clauses of each formula A can be defined by the computation over matrices inductively as follows:

$$\begin{array}{lcl}
\|p\| & := & (V(p))^M \quad (p \in \mathsf{Prop}), \\
\|\neg A\| & := & \overline{\|A\|}, \\
\|A \vee B\| & := & \|A\| + \|B\|, \\
\|\Diamond A\| & := & R^M\|A\|.
\end{array}$$

Example B.2. *Recall a Kripke model $\mathfrak{M} = (W, R, V)$ in Example B.1. A Boolean vector of the truth set of $\Diamond p$ is obtained by*

$$\|\Diamond p\| = R^M\|V(p)\| = \begin{bmatrix} 1 & 1 & 1 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix}.$$

Exercise 1. *Let us define a Kripke model $\mathfrak{M} = (W, R, V)$ by:*

$$\begin{array}{rcl}
W & = & \{ w_1, w_2, w_3 \}, \\
R & = & \{ (w_1, w_1), (w_1, w_2), (w_1, w_3), (w_2, w_2) \}, \\
V(p) & = & W.
\end{array}$$

1. *Calculate the truth set of $\Diamond\Diamond p \to \Diamond p$ (Hint: $p \to q := \neg p \vee q$).*
2. *Calculate the truth set of $\Box p$.*
3. *Calculate the truth set of $\Box p \to \Box\Box p$.*

B.3 Frame Properties in Matrix

B.3.1 Verification of Frame Properties

Every frame property listed in Table B.1 can be reformulated in terms of Boolean matrix, where $\mathbf{1}$ and $\boldsymbol{E}$ denotes a complete vector ($\mathbf{1}(i,1) = 1$ for all i) and a unit square matrix ($\boldsymbol{E}(i,j) = 1$ if $i = j$; 0 otherwise), respectively.

Table B.1: Frame Properties and Corresponding Valid Formulas

Name	Frame Condition	Formula	Matrix Reformulation
Reflexive	$\forall w(wRw)$	T $\Box p \to p$	$R = R + \boldsymbol{E}$
Symmetric	$\forall w, v(wRv$ implies $vRw)$	B $p \to \Box\Diamond p$	$R = {}^tR$ (or $R = {}^tR + R$)
Transitive	$\forall w, v, u(wRv \& vRu$ imply $wRu)$	4 $\Box p \to \Box\Box p$	$R = RR + R$
Serial	$\forall w \exists v(wRv)$	D $\Box p \to \Diamond p$	$R({}^tR) = R({}^tR) + E$ (or $\mathbf{1} = R\mathbf{1}$)
Euclidean	$\forall w, v, u(wRv \& wRu$ imply $vRu)$	5 $\Diamond p \to \Box\Diamond p$	$R = {}^tRR + R$

Example B.3. *Recall an accessibility relation R in Example B.1, and let us regard R as a square matrix of the relation. Let us check whether R satisfies reflexivity. By $R = R + \boldsymbol{E}$, i.e.,*

$$\begin{bmatrix}1&1&1\\0&1&0\\0&0&1\end{bmatrix} = \begin{bmatrix}1&1&1\\0&1&0\\0&0&1\end{bmatrix} + \begin{bmatrix}1&0&0\\0&1&0\\0&0&1\end{bmatrix}$$

$$= \begin{bmatrix}1&1&1\\0&1&0\\0&0&1\end{bmatrix}.$$

Hence, R satisfies reflexivity.

We can also check whether a given frame satisfies reflexivity, seriality, and symmetricity by the form of a matrix of an accessibility relation of the frame.

- Reflexivity: every diagonal component of the matrix consist of 1.
- Seriality: every row contains at least one occurrence of 1.
- Symmetricity: a matrix is a mirror image in the diagonal line.

Exercise 2. *Let us also use an accessibility relation R in Example B.1.*

1. *Verify whether R satisfies symmetricity, transitivity, seriality and Euclideanness by matrix reformulation of frame properties.*
2. *Verify again whether R satisfies reflexivity, seriality and symmetricity by the form of matrix.*

B.3.2 Correspondence between Frame Properties and Valid Formulas

- If a frame $\mathfrak{F}$ satisfies a frame property in Table B.1, then the corresponding formula is valid on a model $(\mathfrak{F}, V)$.
- If a formula which defines a frame property is not valid on a model $(\mathfrak{F}, V)$, then the corresponding frame property in Table B.1 does not satisfy in the frame $\mathfrak{F}$.

Bibliography

[1] Franz Baader, Deborah L McGuinness, and Daniele Nardi. *The Description Logic Handbook: Theory, implementation, and applications*. Cambridge University Press, 2003.

[2] Philippe Balbiani, Vincent Demange, and Didier Galmiche. A sequent calculus with labels for PAL. *Advances in Modal Logic*, 6, 2014.

[3] Philippe Balbiani, Hans van Ditmarsch, Andreas Herzig, and Tiago De Lima. Tableaux for public announcement logic. *Journal of Logic and Computation*, 20(1):55–76, 2010.

[4] Jon Barwise and Jerry Seligman. *Information flow: the logic of distributed systems*. Cambridge University Press, 1997.

[5] Rudolf Berghammer and Renate A. Schmidt. Investigating finite models of non-classical logics with relation algebra and relview. In *Proceedings of the 2006 International Conference on Theory and Applications of Relational Structures As Knowledge Instruments - Volume 2*, TARSKI'02-05, pages 31–49, Berlin, Heidelberg, 2006. Springer-Verlag.

[6] Robert Bull. Cut elimination for propositional dynamic logic without *. *Zeitschrift fuer Mathematische Logik und Grundlagen der Mathematik*, 38:85–100, 1992.

[7] Max J Cresswell and George Edward Hughes. *A new introduction to modal logic*. Routledge, 2012.

[8] Melvin Fitting. Bisimulations and boolean vectors. In *Advances in Modal Logic*, pages 97–126. King's College Publications, 2003.

[9] Sabine Frittella, Giuseppe Greco, Alexander Kurz, Alessandra Palmigiano, and Vlasta Sikimíc. Multi-type display calculus for dynamic epistemic logic. *Journal of Logic and Computation*, 2014.

[10] Akira Fusaoka, Katsunori Nakamura, and Mitsunari Sato. On a linear framework for belief dynamics in multi-agent environments. In *Computational Logic in Multi-Agent Systems*, pages 41–59. Springer, 2007.

[11] Peter Gärdenfors. *Belief revision*, volume 29. Cambridge University Press, 2003.

[12] Jelle Gerbrandy and Willem Groeneveld. Reasoning about information change. *Journal of Logic, Language and Information*, 6(2):147–169, 1997.

[13] Shingo Hagiwara, Mikito Kobayashi, and Satoshi Tojo. Belief updating by communication channel. In *Proceedings of the 7th international conference on Computational logic in multi-agent systems*, pages 211–225. Springer-Verlag, 2006.

[14] David Harel, Dexter Kozen, and Jerzy Tiuryn. *Dynamic logic*. MIT press, 2000.

[15] Ryo Hatano, Katsuhiko Sano, and Satoshi Tojo. Linear algebraic semantics for multi-agent communication. In *Proceedings of the 7th International Conference on Agents and Artificial Intelligence*, volume 1, pages 172–181, 2015.

[16] Brian Hill and Francesca Poggiolesi. A contraction-free and cut-free sequent calculus for propositional dynamic logic. *Studia Logica*, 94(1):47–72, 2010.

[17] Bjarni Jónnson and Alfred Tarski. Boolean algebras with operators. part ii. *American Journal of Mathematics*, 74(1):127–162, 1952.

[18] Bjarni Jónsson and Alfred Tarski. Boolean algebras with operators. part i. *American Journal of Mathematics*, 73(4):891–939, 1951.

[19] K.H. Kim. *Boolean matrix theory and applications*. Monographs and textbooks in pure and applied mathematics. Dekker, 1982.

[20] Mikito Kobayashi and Satoshi Tojo. Agent communicability in belief update logic. In *Proc. the 6th International Workshop on Declarative Agent Languages and Technologies*, pages 206–221. Citeseer, 2008.

[21] Saul A. Kripke. Semantical analysis of modal logic i, normal modal propositional calculi. *Zeitschrift fur mathematische Logik und Grundlagen der Mathematik*, 9(5-6):67–96, 1963.

[22] E.J. Lemmon and D.S. Scott. *An Introduction to Modal Logic: The Lemmon Notes*. American philosophical quarterly monograph series. B. Blackwell, 1977.

[23] Clarence Irving Lewis and Cooper Harold Langford. *Symbolic Logic*. The Century Co., 1932.

[24] Churn-Jung Liau. Matrix representation of belief states: An algebraic semantics for belief logics. *International Journal of Uncertainty, Fuzziness and Knowledge-based Systems*, 12(05):613–633, 2004.

[25] Fenrong Liu. *Reasoning about preference dynamics*, volume 354. Springer Science & Business Media, 2011.

[26] Minghui Ma, Katsuhiko Sano, François Schwarzentruber, and Fernando R Velázquez-Quesada. Tableaux for non-normal public announcement logic. *Logic and Its Applications*, 8923:132–145, 2015.

[27] Paolo Maffezioli and Alberto Naibo. Proof theory of epistemic logic of programs. *Logic and Logical Philosophy*, 23(3):301–328, 2013.

[28] Paolo Maffezioli and Sara Negri. A Gentzen-style analysis of public announcement logic. In *Proceedings of the International Workshop on Logic and Philosophy of Knowledge, Communication and Action*, pages 293–313, 2010.

[29] Hiroakira Ono and Yuichi Komori. Logics without the contraction rule. *The Journal of Symbolic Logic*, 50(01):169–201, 1985.

[30] Ewa Orlowska. Relational formalisation of nonclassical logics. In Chris Brink, Wolfram Kahl, and Gunther Schmidt, editors, *Relational Methods in Computer Science*, pages 90–105. Springer Vienna, Vienna, 1997.

[31] Eric Pacuit, Rohit Parikh, et al. Reasoning about communication graphs. *Interactive Logic*, 1:135–157, 2007.

[32] J. A. Plaza. Logics of public communications. In M. L. Emrich, M. S. Pfeifer, M Hadzikadic, and Z. W. Ras, editors, *Proceedings of the 4th International Symposium on Methodologies for Intelligent Systems*, pages 201–216, 1989.

[33] Katsuhiko Sano and Satoshi Tojo. Dynamic epistemic logic for channel-based agent communication. In *Logic and Its Applications*, pages 109–120. Springer, 2013.

[34] Gunther Schmidt and Thomas Ströhlein. Relations and graphs: Discrete mathematics for computer science. *EATCS Monographs on Theoretical Computer Science. Springer-Verlag*, 1993.

[35] J. Seligman, F. Liu, and P. Girard. Logic in the community. In *Proceedings of the 4th Indian Conference on Logic and its Applications*, volume 6521 of *Lecture Notes in Computer Science*, pages 178–188, 2011.

[36] Alfred Tarski. On the calculus of relations. *The Journal of Symbolic Logic*, 6(3):73–89, 1941.

[37] Satoshi Tojo. Collective belief revision in linear algebra. In *Computer Science and Information Systems (FedCSIS), 2013 Federated Conference on Computer Science and Information Systems*, pages 175–178. IEEE, 2013.

[38] Johan van Benthem and Guram Bezhanishvili. Modal logics of space. In *Handbook of spatial logics*, pages 217–298. Springer, 2007.

[39] Johan van Benthem and Fenrong Liu. Dynamic logic of preference upgrade. *Journal of Applied Non-Classical Logics*, 17(2):157–182, 2007.

[40] Hans van Ditmarsch, Wiebe van der Hoek, and Barteld Pieter Kooi. *Dynamic epistemic logic*, volume 337. Springer, 2007.

[41] Yanjing Wang and Qinxiang Cao. On axiomatizations of public announcement logic. *Synthese*, 190(1):103–134, 2013.

Publications

[1] Ryo Hatano, Katsuhiko Sano, and Satoshi Tojo. Linear algebraic semantics for multi-agent communication. In Proceedings of the 7th International Conference on Agents and Articial Intelligence, volume 1, pages 172-181, 2015.

[2] Ryo Hatano, Katsuhiko Sano, and Satoshi Tojo. Teaching modal logic from linear algebraic viewpoints. In Proceedings of the 4th International Conference on Tools for Teaching Logic, pages 55-64, 2015.

[3] Ryo Hatano and Katsuhiko Sano. Cut free labelled sequent calculus for dynamic logic with relation changers. In Proceedings of the Joint Conference of the 3rd Asian Workshop on Philosophical Logic & the 3rd Taiwan Philosophical Logic Colloquium, 2016.

[4] Ryo Hatano, Katsuhiko Sano, and Satoshi Tojo. Teaching Modal Logic from the Linear Algebraic Viewpoint. The IfColog Journal of Logics and their Applications (FLAP), volume 4, number 1, pages 111-148, 2017.