EPISTEMIC DYNAMICS AND PROTOCOL INFORMATION

A DISSERTATION

SUBMITTED TO THE DEPARTMENT OF PHILOSOPHY

AND THE COMMITTEE ON GRADUATE STUDIES

OF STANFORD UNIVERSITY

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS

FOR THE DEGREE OF

DOCTOR OF PHILOSOPHY

Tomohiro Hoshi

June 2009

I certify that I have read this dissertation and that, in my opinion, it is fully adequate in scope and quality as a dissertation for the degree of Doctor of Philosophy.

(Johan van Benthem) Principal Adviser

I certify that I have read this dissertation and that, in my opinion, it is fully adequate in scope and quality as a dissertation for the degree of Doctor of Philosophy.

(Alexandru Baltag)

I certify that I have read this dissertation and that, in my opinion, it is fully adequate in scope and quality as a dissertation for the degree of Doctor of Philosophy.

(Solomon Feferman)

I certify that I have read this dissertation and that, in my opinion, it is fully adequate in scope and quality as a dissertation for the degree of Doctor of Philosophy.

(Krista Lawlor)

Approved for the University Committee on Graduate Studies.

For Kazuko and Kazumi Hoshi

Acknowledgements

Contents

List of Tables

List of Figures

Introduction

Knowledge and beliefs play a crucial role in human endeavor. They represent the world in certain ways and, as such, constitute an important basis for decisions about our actions. For this reason, these concepts have been of great importance in philosophical investigations and many disciplines of social sciences.

As various formal methods have been developed in the literature, the mathematical representation of knowledge and beliefs has become a topic of increasing interest. An individual knows or believes certain things based on the information she has. What type of mathematical models can we use for precise representations of her information, knowledge or beliefs? Also, when her environment involves other individuals, she may know what others know and such a part of knowledge depends on knowledge that others have. How can we represent the relations between her knowledge and others? Moreover, she may obtain new information during the course of her activities by e.g. observing certain facts in nature, interacting with other individuals, making inferences from what she previously knows, etc. How can we capture those informational changes in her knowledge?

Investigations on these questions have been addressed by the umbrella term, 'intelligent interaction', and have been studied in various fields of studies, including philosophy, computer science, artificial intelligence, and theoretical economics. This dissertation takes the formal tradition in the investigation of knowledge and beliefs and develops a logical framework that describes intelligent interaction.

Reasoning about Knowledge

To illustrate the kind of phenomena that are described by systems of intelligent interaction, let us start by the following example.

> You and I come back from lunch to our office and see a deck of cards on our desk with a card put face-down right next to the deck. Since it was not there before we left for lunch, we become curious about what the card is...

Now do you know what the card is? Of course, you don't, since the card was already put face-down when we came back. For the same reason, I do not know what the card is. Moreover, having lunch outside and coming back together with you, I know that you do not know what the card is. For the same reason, you do not know that I know what the card is. By a similar reasoning, we can give more complex descriptions about our states of knowledge: You know I know you do not know what the card is, I know you do not know I know what the card is, etc.

This illustrates that highly complex knowledge attributions can be made even in the simplest scenarios, let alone in more complex situations that the real world often presents. Therefore, we may seek a precise method to analyze reasoning about knowledge, which represents states of agents' information in such a way that the examination of statements about agents' knowledge can be systematically carried out.

Epistemic Dynamics

However this is not the end of the story. Intelligent interaction is a dynamic process in which agents' information changes over various sorts of informational events. For instance, consider what could happen in our example above. Seeing the card being placed on my desk, we may simply turn the card over. In that case, both you and I will come to know what the card is. Or I may try to trick you and peep into the card without showing it to you. In that case, I will come to know what the card is, while you won't. In general, various kinds of informational events can happen in the course of intelligent interaction and agents' informational states may change consequently.

In addition, the way that informational events change agents' information can be quite subtle. For illustration, consider another scenario in our example. Seeing the card on the desk, I try to trick you, but, this time, I hide my curiosity and peep into the card without making you realize that I do. In this case, I come to know what the card is, but you do not. Thus at the level of our knowledge about what the card is, we have the same result as the previous peeping scenario in the previous paragraph. What distinguishes the current scenario from the previous one is whether you will know that I know what the card is. In the current scenario, you will *not* know I know what the card is (since I secretly peeped); on the other hand, in the previous scenario, you will know I know what the card is (since you saw me peeping).

There are two kinds of subtleties that the examples illustrate. First, in both scenarios, I obtained the same information, that is, the information about what the card is. However we have different consequences in the two scenarios. This shows that the ways in which I obtained the information can make difference, even when the same information is obtained. Second, the difference of the two scenarios can be exhibited only at a higher-level of our knowledge. Indeed, our knowledge about the card is the same in both scenarios: I know the card in one scenario; but you do not in the other. It is only by our knowledge *about* what we know that the situations in the two scenarios can be differentiated (You know I know in one scenario but you do not know I know in the other).

Still another kind of subtlety can be highlighted by the following scenario. Suppose that what I tell you is always true and you are aware of it. After I peeped into the card on the desk, I could just tell you what the card was. Let us say the card was the ace of Diamonds. If that happens, you will come to know that the card is the ace of Diamonds. Inferring from this, one may expect, whatever I tell you, you will come to know it, since I tell you only true things. However this expectation turns out to be betrayed. Suppose, after you come into my office, I tell you "You do not know, but the card on the desk is the ace of Diamonds". What I just said is true. However, you will not come to know it, simply because the statement becomes false once you come to know what the card is. This example demonstrates a fact about subtlety in intelligent interaction: even when you obtain true information, you may not come to know it.

As these considerations suggest, in order to describe intelligent interaction in a precise manner, we need to get a good grasp on the mechanism about how informational events change agents' informational states. We call such a mechanism *epistemic dynamics*. To represent epistemic dynamics in intelligent interaction, we seek a method to represent informational events and the ways in which they change agents' information.

Protocol Information

Another important element in describing intelligent interaction is the information about what informational events *can* take place in the course of interaction. For instance, in our example, if I put the card into the deck next to it after I peep into it, you cannot any more obtain the information about what the card was by turning the card over. You will have to obtain the information in other ways, e.g. by asking me about it, etc. In general, various kinds of communicational and observational constraints are present in many situations of intelligent interaction. We call the kind of information in intelligent interaction *protocol information*.

The kind of information becomes particularly important when we ask whether agents can reach informational states of interest. In our example, can you know what the card is when you come into my office? Or can you know it without making me realize you do? Answers to these questions depend on how we fill in further details in our example. Suppose, as in the last paragraph, I put the card into the deck after I peeping into it. Mischievous as I am, I leave the office without telling you what the card is. If both of us have cell phones, you may ask me what the card was over the phone. In that case, the answer to the first question may be positive and the answer to the latter, negative. Or we may introduce other communicational and observational constraints into our story so as to make the answers to the questions as we like. Thus when we consider *reachability questions* of certain informational states, we need to specify protocol information involved in given situations of intelligent interaction.

The importance of protocol information in the light of reachability questions can be glimpsed in many famous puzzles. Here are two examples:

> **The Muddy Children Puzzle** (See e.g. [21])
>
> Several children are playing outside. After playing they come inside and their father says "At least one of you has mud on your forehead." Each child can only see other children's forehead but not his/her own. Their father repeats the following question "Do you know whether or not you have mud on your forehead?" The children are very intelligent and honest, and answer father's question at the same time. Can everybody know, over rounds of father's question, whether they have mud on their foreheads? If they can, how many question rounds are needed?
>
> **The Russian cards Problem** (See e.g. [72].)
>
> There are seven known cards. The first two players draw three cards each and the third player gets the remaining card. Can the first and second

players publicly inform each other about their cards without the third player learning their cards?

What makes these questions interesting are the constraints given in the communication scenarios. For instance, if children were allowed to say "You are dirty" or the first and second players secretly communicate, the answers to the reachability questions would be trivial (clearly positive).

Therefore protocol information is an important elements in describing intelligent interaction, together with epistemic dynamics. In order to fully describe intelligent interaction, we need not only a method to represent informational events and their effect on agent's information, but also a method to represent what informational event can take place in the course of intelligent interaction. The aim of this dissertation is to develop a formal framework that provides both kinds of methods to represent epistemic dynamics and protocol information together in one system.

Major Frameworks in Epistemic Logic

The aim of the dissertation is motivated by the fact that systems developed in the literature on intelligent interaction are oriented toward only one of the two aspects, but not toward *both*. Let us illustrate the point by looking at two major frameworks in the literature, *Dynamic Epistemic Logic* (DEL, e.g. [6, 26, 74]) and *Epistemic Temporal Logic* (ETL, e.g. [51]).

Both DEL and ETL appeal to the framework of *Epistemic Logic* (EL). EL is an application of *modal logic* and represents the informational states of agents by a set of epistemically possible states interconnected by some relations. Each relation corresponds to an agent and it represents the agent's indistinguishability between possible states. Roughly, when two states are connected, an agent cannot distinguish the states. In this setting, knowledge is usually interpreted as truth in all indistinguishable states (and other epistemic modalities, including beliefs, are interpreted in certain ways by appealing to the model). This is the interpretation of *Kripke models* in EL, and we call Kripke models *epistemic models* in the context of EL. Epistemic models have been widely applied in analyzing reasoning about knowledge, since Hintikka [37].

DEL: Event Models and Product Update

Although DEL and ETL share this basic representation, the systems represent temporal evolution of agents' informational states in different ways. First, DEL captures the temporal aspect of intelligent interaction by *event models* and *product update*. Event models are a certain kind of Kripke-models that represent informational events. Product update provides an algorithm by which to compute a new epistemic model from a given epistemic model and an event model. The new epistemic model obtained by the product update algorithm represents the new informational state of agents after the informational event captured by the event model happens. In abstract terms, DEL describes temporal evolution of agents' informational states by model transformations induced from event models via product update.

Therefore, DEL is well-suited for describing epistemic dynamics. Event models capture informational events and product update determines their informational effects by computing new epistemic models from them. Successive applications of event models to a given epistemic model represent how agents' informational states evolve in the course of intelligent interaction.

However, DEL does not provide a way of representing protocol information. In DEL, *any* event model can be applied to *any* epistemic model. In this sense, DEL assumes what we may call *universal protocol*, that is, *any informational event can happen at any moment*. To model protocol information, we must bring in some

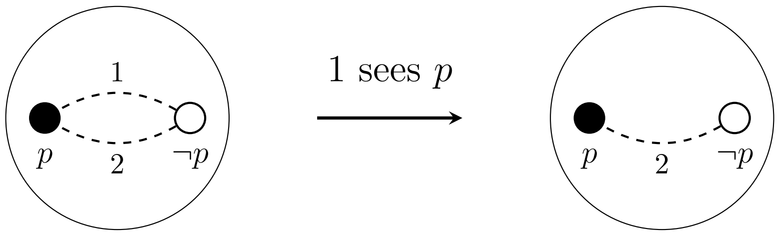

Figure 1: This figure visualizes how DEL represents temporal evolution of informational states. On the left, we have a simple epistemic model consisting of two indistinguishable points. It represents the situation in which two agents, 1 and 2, cannot tell whether p or $\neg p$. Suppose p is true and thus we are at the black world. The model on the right is obtained through product update based on the event model corresponding to "1 sees p". Consequently the dashed line corresponding to the agent 1 is eliminated. By seeing p, the agent 1 can now distinguish the current situation from the possible situation where p is not true.

additional structures, which are external to the basic framework of DEL.

ETL: Branching-Time Tree Structure

ETL provides an alternative representation. It represents temporal evolutions of agents' epistemic states by branching-time tree structures. Models of ETL consist of sequences of events, which are called *histories*. Each history represents temporal development of a given state and each node of a history represents a temporal moment of the development of the corresponding state. Nodes of tree structures are interconnected by indistinguishability relations to describe agents' informational states.

This way, ETL represents temporal evolutions of agents' informational states quite differently from the way DEL does. On the one hand, DEL represents moments of agents' informational states by distinct epistemic models. A new epistemic model is computed via product update every time an event model is applied. On the other hand, ETL represent the whole temporal evolution in single time-branching models.

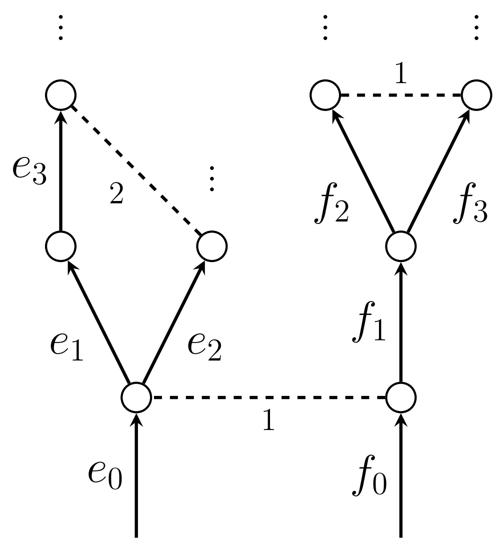

Figure 2: This figure visualizes how ETL models represent temporal evolution of an agent's informational states. Nodes represent moments of histories. Arrows represent temporal transitions from a given node to the next. The arrows are labeled by the names of the corresponding events. Dashed lines represents indistinguishable relations.

The time-branching structure makes ETL well-suited for describing protocol information. Intuitively, branches coming out of a given node represent what sequence of events can take place after the moment represented by the node. (In Figure, events, e_1 and e_2, can happen at the node after e_0, etc.) In this manner, ETL can straightforwardly capture relevant communicational constraints in various situations.

However, ETL does not provide a *systematic* method to represent informational events and their informational effects. In ETL models, events are considered to be unanalyzable elements. In order to represent an intended effect of an informational event, say, e, we must impose an appropriate structure for time-branching trees and agents' indistinguishability relations with respect to the event e. However ETL does not provide us with a procedure by which to figure out what the 'appropriate structure' is. For this reason, we need to come up with desired structures by considerations external to the framework of ETL. This is unlike DEL, since it gives a way to systematically represent informational events and produce new models that represent informational states after the events. In this sense, ETL is not suitable for analyzing epistemic dynamics, especially when compared to DEL.

Similar points can be made about other systems, such as *Interpreted System* (IS, [21]) and *STIT* ([9, 15, 13]). In abstraction, models in those systems can be thought of as consisting of points with temporal orders, which are also interconnected by indistinguishability relations for agents. Even though the models are constructed based on different primitives (local states for agents in IS and histories in STIT), we have to come up with appropriate constraints on those primitives by ourselves in order to represent informational events of our interest and their effects. Thus IS and STIT are well-suited for describing protocol information but less so for analyzing epistemic dynamics.

Merging the Frameworks

Given the situation in the literature on systems describing intelligent interaction, the first goal of the dissertation is to develop a formal framework that captures both epistemic dynamics and protocol information. To achieve this goal, we will make use of representational devices provided by the two major frameworks, DEL and ETL. In particular, we will take the ideas of event models and product update from DEL, and the time-branching structure from ETL. We will merge these mechanisms in one system and construct models that suitably represent the two crucial aspects in intelligent interaction.

The key idea of our framework is that successive applications of product update to epistemic models generate time-branching structures. Given an epistemic model, we assign to each state a set of sequences of event models, which we call a *protocol*. The protocol assigned to a given state is interpreted as the set of sequences of events that can take place at the state. Then, by applying the product update mechanism successively to the epistemic model based on the assigned protocols, we generate ETL tree structures. The generated tree structures represent all possible temporal evolutions of agents' initial informational states that accord with protocol information.

There are three perspectives from which we can view the models in our framework. First, our models are ETL models armed with powerful representational device for describing epistemic dynamics. Histories in ETL structures of our models consist of event models. In DEL, they are mathematical structures with interpretations as informational events and give rise to new epistemic models via product update to represent their intended informational effects. In our method sketched above, ETL tree structures are generated based on event models as such and this guarantees that the generated tree structures properly represent intended temporal evolutions of agents' informational states. This is exactly how event models are built into our

ETL tree structures. This feature and ETL structures of our models allows us to describe epistemic dynamics and protocol information at the same time. Thus, on this perspective, our framework can be thought of as a powerful tool to analyze the situations of intelligent interaction.

Second, our models generalize models in DEL. As mentioned above, DEL assumes *universal protocol* in the sense that any event can happen at any moment. By introducing protocols, our framework can have a freedom in constraining what event can happen in the course of intelligent interaction. This feature allows us to lift the assumption of universal protocol and to generalize the framework of DEL. Based on the perspective, we may now consider reinterpreting systems of DEL over the class of generalized models. This opens up investigation on new logical systems of DEL over the models in our framework.

Third, the idea of generating ETL tree structures by the mechanisms of DEL suggests the possibility of bridging the two distinct frameworks. In several places e.g. [26, 69, 70, 73], the question of how to best compare DEL and ETL has been investigated. In producing ETL models from models in ETL, our framework 'connects' models in DEL and models in ETL. This consideration leads us to investigate the relationship between DEL and ETL within our framework.

Philosophical Applications

Thus this dissertation will take on the three directions that the above perspectives point to: (1) applications, (2) logics, and (3) comparison between DEL and ETL. The projects given by (2) and (3) are of a formal character by nature. Our approach to (1) will be based on the following considerations.

As mentioned at the start, various approaches have been developed to investigate the concepts of knowledge and beliefs. However it has been noted (e.g. [36]) that philosophical and formal investigations have grown rather independently in the recent literature. The variety in approaches can be an advantage, since different approaches can reveal different aspects of the concepts. On the other hand, it can be an obstacle, when the different approaches are left without interaction.

This motivates the approach we will take for applications of our framework. We will see philosophical issues where epistemic concepts involve the aspects of epistemic dynamics and protocol information. By disentangling those aspects, we will try to provide clear visions on relevant philosophical problems. In giving philosophical applications of our framework, we hope not only that those examples illustrate that our framework provides a powerful tool for conceptual analysis, but also that our attempts will contribute to the interaction between philosophical investigation and formal approaches in epistemology.

Connections to Other Topics

Beyond philosophical applications, our framework can contribute to other formal investigations. First our approach is squarely within the logical tradition in epistemic logic and, as such, it can provide further modeling tools for investigations in computer science and theoretical economics, in which the framework of epistemic logic has been applied. Artificial intelligence and game theory are prime examples of the kind of disciplines. Furthermore some investigation has been made based on the framework of epistemic logic in such fields as cryptography ([72]), learning theory, etc.

Second, our framework can also provide a new approach in the study of epistemic logic itself. Various formal systems have been developed based on epistemic logic, and some of the general methodologies adopted in this dissertation can be applied to those systems. For instance, [67] demonstrates that our model construction based on protocols can be applied in *Dynamic Doxastic Logic*. In addition, our models build in syntactic structures by protocols and this feature may be exploited to bridge Dynamic Epistemic Logic and other systems of epistemic logics equipped with similar syntactic flavors. Examples include Justification Logic ([1, 3, 2, 24]), Logic of Awareness ([21, 22]), etc.

Outline of the Dissertation

Finally the structure of the dissertation is as follows. Part I develops our formal framework. In Chapter 1, we will start by reviewing DEL and ETL to introduce the formal machinery required for our framework, such as event models, product update, branching-time structures, etc. Having the formal systems, we will then provide the basic definitions of our framework. We represent protocol information by a set of sequences of event models, and call such a set *DEL-protocols*. Given an epistemic model, we assign a DEL-protocol to each state in the model, and generate ETL models that represent the temporal evolution of the original epistemic models. We call the ETL models *DEL-generated ETL models*. Based on the class of DEL-generated ETL models, we will reinterpret systems of DEL. This will set up a perspective that allows systematic comparisons between DEL and ETL. The main result in our study of the relationship between DEL and ETL will be the *representation theorem*, which states that the class of DEL-generated ETL models can be characterized as a special class of ETL models with some suitable properties.

In Chapter 2, we will study logics on the semantic framework developed in Chapter 1. The main goal of this chapter is to axiomatize the class of DEL-generated ETL models. We will develop our method of axiomatization by starting with the subclass of DEL-generated ETL models that consist only of public announcements (event models that induce model relativizations). We will call the system corresponding to the class of models, *TPAL*, (the name is an acronym for "Temporal Public Announcement

Logic"). Then we will generalize the method to the full class of DEL-generated ETL models. We will call the system *TDEL* (the name is an acronym for "Temporal Dynamic Epistemic Logic").

In Chapter 3, we will extend the systems developed in the previous chapter for wider applications. One kind of extensions are given by introducing new operators to TPAL and TDEL. The operators we will consider includes $\Diamond$ ("Some event can happen after which..."), $\Diamond^*$ ("Some sequence of events can happen after which..."), and P_ϵ ("ϵ has happened before which..."). These operators are useful to analyze various epistemic concepts. Another extension is given by generalizing the model construction introduced in Chapter 1. The notion of protocols introduced in Chapter 1 has a technical restriction on the kind of event models that constitute protocols. We will consider a way to lift the restriction and allow the full class of event models to be in protocols.

Part II of the dissertation develops philosophical applications of the formal framework developed in Part I. In Chapter 4, we will give a logical analysis on *Fitch's paradox* and its variant, *the idealism problem*. We undertake two tasks. The first task is to provide a philosophical framework for verificationism that does not imply the formulation of the knowability thesis, *every truth is knowable*, from which Fitch's paradox and the idealism problem are derived. The second task is to formalize the proposed framework by suitably interpreting a logical system in dynamic epistemic logic. Not only will this make explicit our theoretical commitments, but also it will allow us (i) to present a new formulation of the verificationist knowability thesis as a *provable* statement and (ii) to give a fine-grained logical analysis of alternative formulations of verificationist commitments to knowability.

In Chapter 5, we will deal with the *epistemic closure principle knowledge is closed under logical implication*. In epistemic logic, the principle, formulated as *if* φ *is known* *and φ logically implies ψ, then ψ is known*, is a problem, since it only applies to logically omniscient agents. This is called *the problem of logical omniscience*. Stalnaker argues that it seems infeasible to characterize the notion of knowledge that avoids the problem in the framework of epistemic logic. The first objective of the chapter is to challenge this claim and give a formalization of the desired notion of knowledge in our framework. In addition, the formalization of the notion of knowledge makes it possible to consider the representation of agents' making deductive inferences. Thus, the second objective of the chapter is to model the situations where agents make deductive inferences. We will use the formal representation to describe another perspective on the epistemic closure principle discussed in epistemology, *knowledge can be extended by deductive inference*. This will allow us to compare the two different perspectives on epistemic closure, one in epistemic logic and the other in epistemology, in our formal framework.

Sources of the Chapters

The main ideas and results in Chapter 1 and 2 builds on the joint works, [68] with Johan van Benthem, Jelle Gerbrandy, and Eric Pacuit, and [42] with Audrey Yap. Chapter 3 is an extension of [41]. Chapter 4 and 5 are based on [39, 40].

Part I

Formal Framework

Chapter 1

Merging Frameworks

As discussed in Introduction, there are two important aspects in describing intelligent interaction. One is the mechanism about how agents' informational states change over informational events. Since informational events of the simplest kind could affect agents' knowledge in a very delicate manner, it is crucial to get a good grasp on informational events and their epistemic effects. We call this aspect *epistemic dynamics*. The other is what informational events can take place in the course of agents' interaction. Various kinds of communication constraints are present in many situations of agents' intelligent interaction, and the information about such constraints is crucial to deal with *reachability questions*. We call this aspect *protocol information*.

Although various kinds of multi-agent intelligent systems have been developed so far, each system seems suitable for only one of the two aspects but less so for the other. For instance, *Dynamic Epistemic Logic* (DEL) describe epistemic dynamics well by *event models* and *product update*; however, DEL does not provide a machinery to describe protocol information. On the other hand, *Epistemic Temporal Logic* (ETL) describes protocol information well by its time-branching tree structures; however, ETL does not provide a machinery to systematically represent informational events and their informational effects.

The main purpose of this chapter is to develop a formal framework that describes both epistemic dynamics and protocol information together. We achieve this goal by merging DEL and ETL. Our key idea to put them together is that repeatedly applying product update with sequences of event models *generates* an ETL model. In this chapter, we will show how this idea can be made precise.

Furthermore, generating ETL models from DEL-models, our framework can provide a formal ground on which DEL and ETL can be compared in a precise manner. The main result of this chapter is the *representation theorem*, which characterizes the largest class of ETL models corresponding to DEL protocols in terms of notions of *Perfect Recall*, *No Miracles*, and *Bisimulation Invariance*.

We will proceed as follows. We start by introducing DEL and ETL and discussing how they represent intelligent interaction (Section 1.1- 1.3). Then we go on to merge the two systems and obtain the framework that we propose (Section 1.5). Having the framework, we will compare DEL and ETL (Section1.5) and prove the representation theorem.

1.1 Epistemic Logic

We start by introducing DEL and ETL to develop our framework. Both systems build on *Epistemic Logic* (EL) to represent informational states of agents. EL is an application of Modal Logic, which has been developed since the seminal work by Hintikka ([37]). Fix a finite set of agents $\mathcal{A}$ and a countable set of propositional letter At.

Definition 1.1.1 (Epistemic Models) An *epistemic model* is a triple $(W, \sim, V)$, where (i) W is a nonempty set, (ii) $\sim$, a function from $\mathcal{A}$ to $\wp(W \times W)$ and (iii) V, a valuation function on At, i.e $V : \mathsf{At} \to \wp(W)$. $\triangleleft$

W is interpreted as a set of epistemically possible situations. We call the elements in W in various ways, including *worlds*, *states*, *points*, etc. The relation $\sim$ assigns a binary relation on W for an agent in $\mathcal{A}$. By convention, we will write $\sim_i$ for $\sim (i)$ and $w \sim_i v$ for $(w, v) \in\sim (i)$. The intended interpretation of $w \sim_i v$ is "at w, i considers v possible." The valuation function V assigns to $p \in \mathsf{At}$ a subset of W. $V(p)$ represents the set of worlds where p is true. Therefore, V represents truth of propositional letters at worlds in W.

When $\sim$ assigns an equivalence relation on W to an agent i, an equivalence class induced by $\sim_i$ represents a set of worlds that an agent i cannot distinguish. For this reason, we often call $\sim_i$ an *indistinguishability relation* for an agent i. Although we do not assume $\sim$ assigns equivalence relations, we will often read $w \sim_i v$ as "w and v are indistinguishable for i". Also most of our examples below give models in which $\sim$ assigns equivalence relations.

Finally, given an epistemic model $\mathcal{M}$, we denote its domain, indistinguishability relation and valuation function also by $Dom(\mathcal{M})$, $\sim_{\mathcal{M}}$, and $V_{\mathcal{M}}$ respectively.

Example 1.1.2 (Office-Card Example: Epistemic Models) Figure 1.1 visualizes an example of epistemic models. The model consists of two worlds, w and v, represented by the two circles. The indistinguishability relations for the agent, 1 and 2, are equivalence relations visualized by dashed lines labelled with 1 and 2 respectively. The letters below the two circles represent the truth of propositional letters, p and q: p is true only at the black world; q is true at both worlds.

The model can be thought of as capturing the *office-card* example discussed in Introduction. You and I come in to our office and find a card being placed face-down on the desk. Suppose that the card is the ace of Diamonds. Let p be "The card on the desk is the ace of Diamonds." and Let q be "We are in the office." Given those assumptions, the black world in the model represents the situation, since p and q are

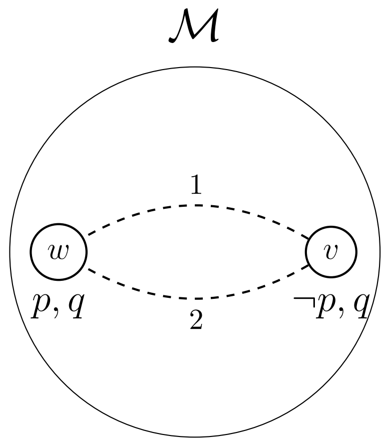

Figure 1.1: Epistemic Model

true there. However, you and I, 1 and 2 in the model cannot tell whether p or $\neg p$.

◁

Epistemic models represent informational states of agents by a set of worlds with a valuation for each propositional letter and indistinguishable relations assigned for agents. EL describes informational states represented by epistemic models by the following language.

Definition 1.1.3 (Language of EL) The language of EL consists of that of propositional logic (PL). Formulas of EL is inductively defined as follows:

$$\varphi ::= \top \mid p \mid \neg\varphi \mid \varphi \wedge \varphi \mid [i]\varphi$$

where $p \in \mathsf{At}$ and $i \in \mathcal{A}$. The dual $\langle i \rangle$ of $[i]$ and the other boolean operators are defined in the standard way. We denote the set of formulas in EL by $\mathcal{L}_{el}$. We call formulas in $\mathcal{L}_{el}$ *epistemic formulas.* ◁

Definition 1.1.4 (Truth in EL) Let $\mathcal{M} = (W, \sim, V)$ be an epistemic model. The truth of a formula $\varphi \in \mathcal{L}_{el}$ at w in $\mathcal{M}$, denoted by $\mathcal{M}, w \models \varphi$, is inductively defined as follows:

$$
\begin{array}{lll}
\mathcal{M}, w \models p & \text{iff} & w \in V(p) \quad (\text{with } p \in \mathsf{At}) \\
\mathcal{M}, w \models \neg\varphi & \text{iff} & \mathcal{M}, w \not\models \varphi \\
\mathcal{M}, w \models \varphi \wedge \psi & \text{iff} & \mathcal{M}, w \models \varphi \text{ and } \mathcal{M}, w \models \psi \\
\mathcal{M}, w \models [i]\varphi & \text{iff} & \forall v \in W : w \sim_i v \text{ and } \mathcal{M}, v \models \varphi
\end{array}
$$

◁

Thus $[i]\varphi$ is true at w iff φ is true at all worlds that i considers possible at w. We give different interpretations to the modal operator $[i]$ depending on our purpose of applications. Among them are "i knows φ," "To the best of i's information, φ." etc.[1] Given the former interpretation as standard, we will often call the modality the *knowledge modality* or *knowledge operator*. The dual operator $\langle i \rangle\varphi$ can be also read in various ways, such as "i considers φ possible", etc.

Example 1.1.5 (Office-Card Example: Truth in EL) Consider the model in Figure 1.1 again. At w, $[1]q$ and $[2]q$ are true, since q is true at both w and v. Therefore you know q and I know q. However, $[1]p$ and $[2]p$ are *false* since p is false at v, which we cannot distinguish from w. Therefore you do not know p and I do not know p. Similarly we can consider more complex formulas in the model and confirm that they provide right results. Readers are invited to verify the truth of $[1]\neg[2]p$, $[2]\neg[1]p$, $[1][2]\neg[1]p$, etc. at w ("You know I do not know p", "I know you do not know p", "You know I know you do not know p", etc.) ◁

1.2 Dynamic Epistemic Logic

Epistemic models, as defined in the previous section, are to describe *static* states of agents' information. Dynamic Epistemic Logic (DEL) introduces certain model

[1] The former reading is standard in epistemic logic. The latter reading is suggested by e.g. [66]

transformations to EL and represents dynamics of agents' informational states over informational events. To introduce the framework of DEL, we start with *Public Announcement Logic* (PAL, e.g. [53, 62]), since this simplest system of DEL exhibits the basic ideas of DEL well.

1.2.1 Public Announcement Logic

PAL describes dynamics of informational states of agents when true information is *publicly announced*. PAL represents public announcements as *model relativization*. Given an epistemic model $\mathcal{M}$ and a formula φ, *the public announcement of* φ, denoted by $!\varphi$, is the operation that eliminates the worlds in $\mathcal{M}$ where φ is *false*. This is illustrated in Figure 1.2. p is true at the black world and false at the white world. Through the public announcement $!p$, the white world is eliminated and a new epistemic model, represented on the right, is obtained.

PAL extends EL with operators of the form $[!\varphi]$. The language of PAL is thus defined as follows.

Definition 1.2.1 (Language of PAL) Formulas of PAL is inductively defined as follows:

$$\varphi \ ::= \top \mid p \mid \neg\varphi \mid \varphi \wedge \varphi \mid [i]\varphi \mid [!\varphi]\varphi$$

where $p \in \mathsf{At}$ and $i \in \mathcal{A}$. The duals, $\langle i \rangle$ and $\langle !\varphi \rangle$, of $[i]$ and $[!\varphi]$, and the other boolean operators are defined in the standard way. We denote the set of formulas in PAL by $\mathcal{L}_{pal}$. ◁

Definition 1.2.2 (Truth in PAL) The truth of formulas in PAL is defined by adding the inductive clause for the operator $[!\varphi]$ to the truth definition of EL (Definition 1.1.4). Given an epistemic model $\mathcal{M} = (W, \sim, V)$,

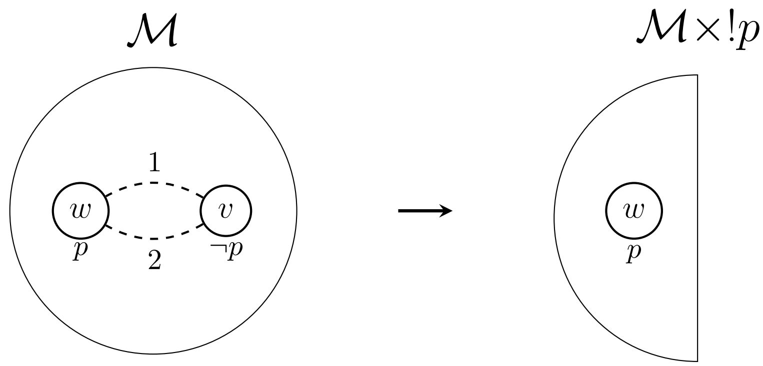

Figure 1.2: Public Announcement

$$\mathcal{M}, w \models [!\varphi]\psi \quad \text{iff} \quad \mathcal{M}, w \models \varphi \text{ implies } \mathcal{M}\times !\varphi, w \models \psi$$

where $\mathcal{M}\times !\varphi = (W', \sim', V')$ is defined by:

$$\begin{aligned} W' &= \{v \in W \mid \mathcal{M}, v \models \varphi\} \\ \sim'(i) &= \sim(i) \cap (W' \times W') \\ V'(p) &= V(p) \cap W'. \end{aligned}$$

$\triangleleft$

The intended readings of $[!\varphi]\psi$ and $\langle !\varphi\rangle\psi$ are respectively "After the truth of φ is publicly announced, ψ." and "The truth of φ can be publicly announced after which ψ." However readers should not be misled by the term "public announcement" here. $!\varphi$ is defined simply by model relativization to the worlds where φ is true and, depending on our purposes, we may capture other kinds of informational events by $!\varphi$, insofar as their informational effects can be so construed. Other possible readings of $[!\varphi]$ are: e.g. "after (publicly) observing φ, ...", "after (publicly) verifying the truth of φ", etc.

Example 1.2.3 (Office-Card Example: PAL) On this note, we can think of Figure 1.2 as a model of one of the scenarios in our office-card example. Suppose that the card on the desk is the ace of Diamonds. Our informational state when we come in to the office can be then represented by the epistemic model $\mathcal{M}$ on the left by interpreting p as "The card on the desk is the ace of Diamonds." Turning the card on the desk over, we will publicly observe that the card is the ace of Diamonds. This informational event changes our informational state into the state represented in the epistemic model $\mathcal{M}\times!\varphi$ on the right. Thus at w, we have $[!p][1]p$ and $[!p][2]p$ true (both 1 and 2 come to know that p).

Also note that, in the model in question, a formula $\varphi := p\wedge\neg[1]p$ (p but 1 does not know p) is true at w in $\mathcal{M}$, but false at w in $\mathcal{M}\times!\varphi$. Therefore, $[!\varphi][i]\varphi$ is false (at w in $\mathcal{M}$)! (False propositions can never be known.) Thus, as mentioned in Introduction, some truth may not be known after it is publicly announced. This phenomena will be closely studied in Chapter 4. ◁

1.2.2 Event Models and Product Update

Although public announcements represent a variety of informational events, there are still different kinds of informational events that cannot be properly captured by public announcements. For instance, observation may not have to be completely public. In our example, I may peep into the card on the desk while you do not. I may even try to do it secretly without making you realize I do. Dynamic Epistemic Logic (DEL) generalizes the framework of PAL by introducing the machinery of *event models* and *product update*.

Definition 1.2.4 (Event Model) An *event model* $\mathcal{E}$ is a tuple $(E, \rightarrow, \mathsf{pre})$, where (i) E is a finite nonempty set, (ii) $\rightarrow$, a function from $\mathcal{A}$ to $\wp(E\times E)$ and (iii) pre, a function from E to $\mathcal{L}_{el}$. ◁

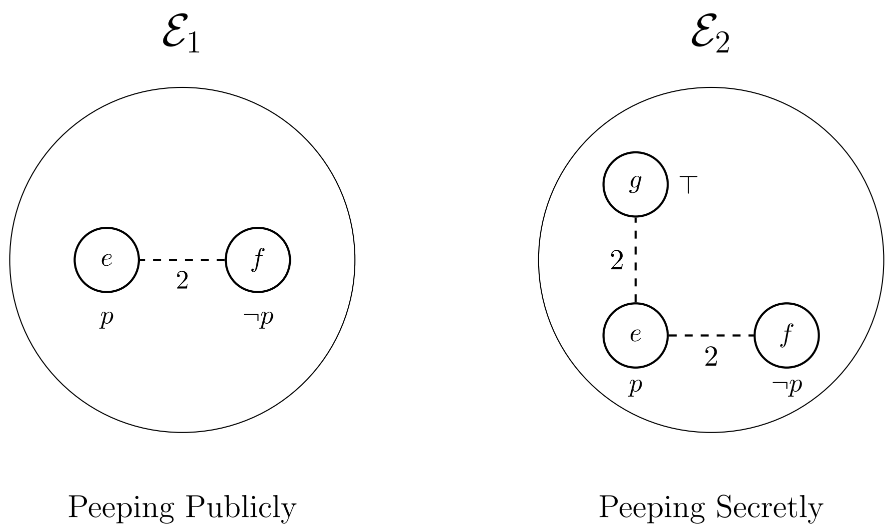

Figure 1.3: Event Models

The *domain* E of an event model can be considered as the set of events. Given two events, e and f, the intended interpretation of $(e, f) \in\rightarrow (i)$ is as "when e happens, an agent i considers it possible that f has happened." As discussed in the case of the indistinguishability relation of epistemic models, $\sim$ (Section 1.1), when $\rightarrow (i)$ is an equivalence class on E, an equivalence class induced by $\rightarrow (i)$ represents a set of events that i cannot distinguish: When an event in an equivalence class happens, i thinks that any of the events in the class has happened. Thus we will call $\rightarrow$ the *indistinguishability relation* for i over events. The function pre determines *preconditions* of events. Given $\mathsf{pre}(e) = \varphi$, an event e can happen at a world iff φ is true at the world. (More on preconditions below. See Section 1.2.3) Note that a precondition that pre maps to each event must be an epistemic formula (a formula in $\mathcal{L}_{el}$). This assumption can be lifted. See Chapter 3.

When $(e, f) \in\rightarrow (i)$, we write $e \rightarrow_i f$ by convention. Also given an event model $\mathcal{E}$, we denote its domain, indistinguishability relation, and precondition function by $Dom(\mathcal{E})$, $\rightarrow_{\mathcal{E}}$, and $\mathsf{pre}_{\mathcal{E}}$ respectively.

Example 1.2.5 (Office-Card Example: Event Models) Let us come back to the office-card example. Figure 1.3 visualizes the two event models, $\mathcal{E}_1$ and $\mathcal{E}_2$, which represent the two scenarios in which I peep into the card on the desk *in front of you* or *secretly from you*. Suppose that the card on the desk is the ace of Diamonds. Denote the proposition expressing this fact by p and let 1 and 2 be me and you respectively. First consider $\mathcal{E}_1$. It has two events, e and f, with preconditions p and $\neg p$ respectively. 1 can distinguish e and f ($\rightarrow_1$ partitions the model into $\{e\}$ and $\{f\}$), while 2 cannot ($\rightarrow_2$ partitions the model into the whole domain $\{e, f\}$). We can think of the event e in the event model as the event of 1's peeping into the card in front of 2. Suppose e happens. 1 thinks that e happened but not f, since he can distinguish e and f. Also e can happen only if p. Thus 1 comes to know that p. On the other hand, 2 cannot distinguish e and f. Thus she does not come to know that p after e. However, in the sense that these two event, e and f, are the only events that 2 considers possible, 2 come to know that 1 now know whether p or $\neg p$.

Next consider $\mathcal{E}_2$. It adds the third event, g, to the model on the left. Having $\top$ (a tautologous truth) as its precondition, g can be thought of as any trivial event that can happen no matter how the world is. Thus when e happens, 2 cannot tell whether 1 has peeped into the card (by which 1 would obtain the information that p or $\neg p$) or whether anything that informs 1 of what the card is has happened. Thus, unlike the model on the left, 2 does not come to know that 1 now know whether p or $\neg p$. The event e in the model on the right can be interpreted as the event of 1's peeping into the card without making 2 realizing that 1 does. $\lhd$

In DEL, these event models induce model transformations via *product update* to represent the informational effects of corresponding informational events.

Definition 1.2.6 (Product Update) The *product update* $\mathcal{M} \otimes \mathcal{E}$ of an epistemic model $\mathcal{M} = (W, \sim, V)$ and an event model $\mathcal{E} = (E, \rightarrow, \mathsf{pre})$ is the epistemic model

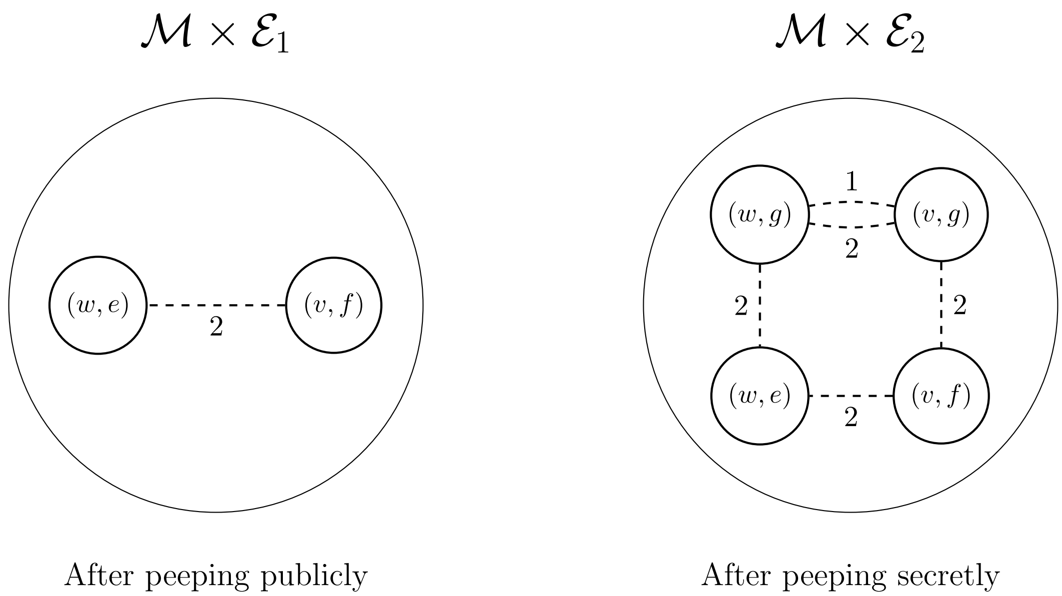

Figure 1.4: Product Update

$(W', \sim', V')$ with

1. $W' = \{(w, e) \mid w \in W, e \in E$ and $\mathcal{M}, w \models \mathsf{pre}(e)\}$,

2. $(w, e) \sim'_i (v, f)$ iff $w \sim_i v$ in $\mathcal{M}$ and $e \longrightarrow_i f$ in $\mathcal{E}$, and

3. $(w, e) \in V'(p) = w \in V(p)$ for all $p \in \mathsf{At}$. $\lhd$

Figure 1.4 represents the epistemic models obtained by transforming the model $\mathcal{M}$ in Figure 1.1 via product update based on the event models, $\mathcal{E}_1$ and $\mathcal{E}_2$, in Figure 1.3. The readers are invited to verify those results.

DEL captures temporal evolution of agents' informational states by model transformations induced from event models via *product update*. However, as seen in the above examples, events have their meanings relative to event models that they belong to. For this reason, DEL deals with pairs of event models and events in them. *A* *pointed event model* ϵ is a pair $(\mathcal{E}, e)$, where $\mathcal{E}$ is an event model and e is an event in $Dom(\mathcal{E})$.

Definition 1.2.7 (Language of DEL) Formulas of DEL is inductively defined as follows:

$$\varphi ::= \top \mid p \mid \neg\varphi \mid \varphi \wedge \varphi \mid [i]\varphi \mid [\mathcal{E}, e]\varphi$$

where $p \in \mathsf{At}$, $i \in \mathcal{A}$, and $(\mathcal{E}, e)$ is a pointed event model. The duals, $\langle i \rangle$ and $\langle \mathcal{E}, e \rangle$, of $[i]$ and $[\mathcal{E}, e]$, and the other boolean operators are defined in the standard way. We denote the set of formulas in PAL by $\mathcal{L}_{del}$. ◁

Definition 1.2.8 (Truth in DEL) The truth of formulas in DEL is defined by adding the inductive clause for the operator $[\mathcal{E}, e]$ to the truth definition of EL (Definition 1.1.4). Given an epistemic model $\mathcal{M} = (W, \sim, V)$,

$$\mathcal{M}, w \models [\mathcal{E}, e]\psi \quad \text{iff} \quad \mathcal{M}, w \models \mathsf{pre}_{\mathcal{E}}(e) \text{ implies } \mathcal{M} \otimes \mathcal{E}, (w, e) \models \psi.$$

◁

The intended readings of $[\mathcal{E}, e]\psi$ and $\langle \mathcal{E}, e \rangle\psi$ are respectively "After the event $(\mathcal{E}, e)$ happens, ψ." and "The event $(\mathcal{E}, e)$ can happen after which ψ."

Example 1.2.9 (Office-Card Example: Truth in DEL) Let us consider our office-card example in Figure 1.4. After peeping in public, you come to know I come to know whether p. This is expressed by the truth of $[\mathcal{E}_1, e][2]([1]p \vee \neg[1]p)$ at w in $\mathcal{M}$. The formula is true since $[2]([1]p \vee \neg[1]p)$ is true at (w, e) in $\mathcal{M} \times \mathcal{E}_1$. Also after peeping secretly, you *do not* come to know I know whether p. This is expressed by the truth of $[\mathcal{E}_2, e]\neg[2]([1]p \vee \neg[1]p)$ at w in $\mathcal{M}$. The formula is true since $\neg[2]([1]p \vee \neg[1]p)$ is true at (w, e) in $\mathcal{M} \times \mathcal{E}_2$. ◁

Remark 1.2.10 (Public Announcement) DEL generalizes PAL in the sense that public announcements can be captured by a certain kind of event models. The public announcement $!\varphi$ of a formula φ (in $\mathcal{L}_{el}$) can be thought of as the event model $\mathcal{E}_\varphi = (E, \rightarrow, \mathsf{pre})$, where (i) $E = \{e\}$, (ii) for each $i \in \mathcal{A}$, $\rightarrow_i = \{(e, e)\}$ (equivalence relation on e), and (iii) $\mathsf{pre}(e) = \varphi$. The product update of an epistemic model $\mathcal{M} = (W, \sim, V)$ with an event model $\mathcal{E}_\varphi$ produces a 'submodel' of $\mathcal{M}$ containing only the states where φ is true (in $\mathcal{M}$). More precisely, $\mathcal{M} \times \mathcal{E}_\varphi = (W', \sim', V')$ is:

- $W' = \{(w, e) \mid \mathcal{M}, w \models \varphi\}$
- $(w, e) \sim'_i (v, e)$ iff $w \sim_i v$
- $V'(p) = \{(w, e) \mid w \in V(p)\}$

Note that, by the above mentioned restriction on the precondition function on pre, only public announcements of epistemic formulas can be directly modeled in this way. (cf. Definition 2.1.2) This is not a substantial restriction in DEL, since formulas in DEL reduces equivalent to formulas in EL via *reduction axioms* as we will discuss later (Section 2.3 and 2.5.1). However the situation is different in our own framework. See also Remark 1.4.13. $\lhd$

Remark 1.2.11 (Identifying Event Models) Strictly speaking, there are as many distinct event models as there are distinct elements, according to Definition 1.2.4. For instance, two event models, $\mathcal{E}$ and $\mathcal{F}$, consisting of single reflexive elements e and f (with $e \neq f$) with precondition φ, can be identified with φ. However, we will identify event models, when they are *isomorphic*. Two event models $\mathcal{E}$ and $\mathcal{F}$ are isomorphic, if there is a one-to-one map from $\mathcal{E}$ onto $\mathcal{F}$ that preserves indistinguishability relations and precondition functions. Given that event models are finite (See below Remark 1.2.12), the class of all event models is countable. $\lhd$

Remark 1.2.12 (Finite Domain) In Definition 1.2.4, the domains of event models are defined to be finite. The main reason is that the standard reduction axioms for the DEL modality $[\mathcal{E}, e]$ (cf. [6]) contain a conjunction over all elements of $\mathcal{E}$ reachable from e. Now if this set is infinite, then the reduction axiom will not be a *formula* of $\mathcal{L}_{DEL}$ since it contains an infinite conjunction. We return to this issue in Section 1.5.2. $\lhd$

1.2.3 Protocol Information in DEL?

As we have seen above, DEL provides a good representational framework for epistemic dynamics. It represents static informational states of agents by epistemic models and informational events by event models. Product update transforms epistemic models into new models that represent informational states after informational events.

However DEL does not provide a machinery that is suitable for representing protocol information. There are two senses in which it does not. First, in DEL, there is no restriction on event models to be applied to given epistemic models. Any event model can be applied to any epistemic model and the epistemic model obtained by the process can be described by using corresponding event operators. As we saw above, the informational state after I peep into the card on the desk can be represented (in Figure 1.4) by an epistemic model. No matter what communication constraints we think of for the situation after my peeping, e.g. I put the card into the deck after peeping and leave the office, DEL does not forbid us from applying the public announcement of $!p$ (The card is the ace of Diamonds), which will yield the truth of $\langle !\varphi \rangle [2]p$ (p can be publicly announced after which you know p).

Second, one may try to adjust precondition functions of event models to represent communication constraints. One component of event models is a precondition function pre. The function is interpreted in such a way that an event e *can* happen iff $\mathsf{pre}(e) = \varphi$. Thus, in the above example, we may introduce a new propositional letter, say d, to represent whatever communicational or observational constraints there will be after my peeping, and say that the public announcement of p is in fact the public announcement of $p \wedge d$, since the public announcement of p can happen only if p is true and the condition d is satisfied. For instance, we may interpret d here as "the card is still on the desk in the office in front of us", and make d false to represent the situation after I put the card back into the deck of cards and leave the office. $\langle !(p \wedge d) \rangle [2]p$ then become *false*, since $p \wedge d$ is false in that case. Thus, after I put the card into the deck and leave the office, you *cannot* know what the card is by turning the card over or asking me what the card was.

Even though such an adjustment of precondition functions may yield satisfactory representations for certain cases of intelligent interaction, the strategy cannot be applied generally to represent protocol information. The main obstacle is that the informational events that can happen may change over time. In many interaction scenarios, the information about what can happen at a given moment depends on the information about what has happened earlier. For instance, our conversation may obey implicit rules such as "Do not repeat yourself", "Say p after q", etc. Protocol information of this kind cannot be captured by the above maneuver, since the preconditions of events are encoded by propositional letters, whose truth values are constant in DEL at a given world. A given world can evolve in various possible ways, depending on what event happens, and propositional letters cannot do the job of tracking how the world has evolved. For this reason, DEL is not suitable for capturing the temporality in protocol information.

1.3 Epistemic Temporal Logic

Epistemic Temporal Logic (ETL) provides an alternative framework to represent intelligent interaction. ETL represents temporal evolutions of agents' epistemic states by branching-time tree structures. Those structures describe how histories of given states evolve.

1.3.1 Branching-Time Tree Structure

Fix a finite set of agents $\mathcal{A}$ and a countable set of propositional letter At. Let Σ be a set of *events*. A *history* is a finite sequence of events from Σ. We write Σ^* for the set of histories built from elements of Σ. For a history h, we write he for the history h followed by the event e. Given $h, h' \in \Sigma^*$, we write $h \preceq h'$ if h is a *prefix* of h', i.e. there is some k such that $hk = h'$. $H \subseteq \Sigma^*$ is *closed under finite prefix* if, for every $h \in H$ and $h' \preceq h$, $h' \in H$. We denote the empty sequence by λ.

Definition 1.3.1 (ETL Models) Let Σ be a set of events. An *ETL model* is a tuple $(\Sigma, H, \sim, V)$ where (i) H does not contain λ and is a subset of Σ^* closed under finite prefix, (ii) $\sim$ is a function from $\mathcal{A}$ to $\wp(H \times H)$, and (iii) V is a function from At to $\wp(H)$. $\lhd$

H represents the temporal structure with $h' = he$ representing the temporal point after the event e has happened at the point h. For each $i \in \mathcal{A}$, the relation $\sim(i)$ (also denoted by $\sim_i$) represents the indistinguishability relation on histories for i. V is a valuation function on H. (cf. Definition 1.1.1 and 1.2.4).

Figure 1.5 visualizes two ETL models. Let us consider the models by using our example. Let e_0 and f_0 be respectively the event of the ace of Diamonds being placed on the desk and the event of a different card being placed face-down on the desk respectively. Let p be "the card on the desk is the ace of Diamonds." Thus, assume that, in both models, p is true at each node following e_0, while p is false at each node following f_0. Now when we come into the office, we do not know what the card is. Thus, the nodes e_0 and f_0 are indistinguishable to me and you (1 and 2 respectively) in both models. At the node e_0, we can turn over the card (the event represented

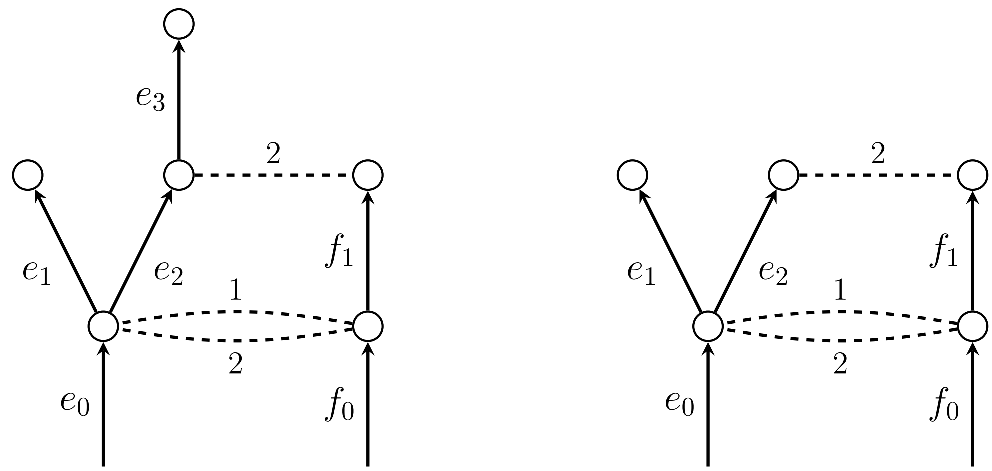

Figure 1.5: ETL Models

by e_1) or I can peep into the card in front of you (the event represented by e_2). If e_1 happens, we both know the card is the ace of Diamonds (we can distinguish the node e_0e_1, where p is true, from the nodes coming out of f_0). If e_2 happen, I come to know p but you don't, since you (2) cannot distinguish e_0e_2 from f_0f_1, where p is false. So far, the two models in Figure 1.5 are the same. They differ in what can happen at e_0e_2. The left model can be thought of as representing the situation where I stay in the office. Thus I can tell you what the card is (the event represented by e_3) and you will come to know p. On the other hand, the right model can be thought of as representing the situation where I leave the office after putting the card into the deck. No event can happen to change the informational states of ours.

Different modal languages describe ETL models (see, for example, [21, 38]). Here we give just the minimal language of ETL.

Definition 1.3.2 (Language of ETL) Let Σ be a set of events. Formulas $\mathcal{L}_{etl}$ are defined inductively as follows:

$$\varphi ::= \top \mid p \mid \neg\varphi \mid \varphi \wedge \varphi \mid [i]\varphi \mid [e]\varphi$$

where $i \in \mathcal{A}$, $e \in \Sigma$ and $p \in \mathsf{At}$. The dual, $\langle i \rangle$ and $\langle e \rangle$, and boolean connectives $(\vee, \rightarrow, \leftrightarrow)$ are defined in the standard way. We denote the set of formulas in ETL by $\mathcal{L}_{etl}$. ◁

It is often natural to extend the language $\mathcal{L}_{etl}$ with group knowledge operators (e.g., common or distributed knowledge) and more expressive temporal operators (e.g., arbitrary future or past modalities). This may lead to high complexity of the validity problem (cf. [29, 70]). We will study some of those operators in Chapter 2 and 3.

Definition 1.3.3 (Truth in ETL) Let $\mathcal{H} = (\Sigma, H, \sim, V)$ be an ETL model. The truth of a formula φ at a history $h \in H$, denoted $\mathcal{H}, h \models \varphi$, is defined inductively as follows:

$$\begin{array}{lll}
\mathcal{H}, h \models p & \text{iff} & h \in V(p) \quad (\text{with } p \in \mathsf{At}) \\
\mathcal{H}, h \models \neg\varphi & \text{iff} & \mathcal{H}, h \not\models \varphi \\
\mathcal{H}, h \models \varphi \wedge \psi & \text{iff} & \mathcal{H}, h \models \varphi \text{ and } \mathcal{H}, h \models \psi \\
\mathcal{H}, h \models [i]\varphi & \text{iff} & \forall h' \in H : h \sim_i h' \text{ implies } \mathcal{H}, h \models \varphi \\
\mathcal{H}, h \models [e]\varphi & \text{iff} & he \in H \text{ implies } \mathcal{H}, he \models \varphi
\end{array}$$

◁

The intended readings of $[e]\varphi$ and $\langle e \rangle\varphi$ are respectively "After the event e happens, φ." and "The event e can happen after which φ."

Example 1.3.4 (Office-Card Example: Truth in ETL) In the models discussed in Figure 1.5, $\langle e_1 \rangle [1]p$ is true at e_0, since e_0e_2 is in the model and $[1]p$ is true at e_0e_2. Also, $\langle e_3 \rangle \top$ is true at e_0e_2 in the left model but false in the right.[2] This is simply

[2] The formula reads as "The event e_3 can happen." It literally reads as "The event e_3 can happen

because $e_0e_2e_3$ is in the left model but not in the right. ◁

1.3.2 Epistemic Dynamics in ETL?

As the above examples illustrate, time-branching tree structures in ETL are suitable for representing protocol information. Each history represents temporal development of a given state (or world) and each node of a history represents a temporal moment of the development of the corresponding state. Branches coming out of a given node represent what can happen at the moment and express protocol information.

However, ETL does not provide a *systematic* method to represent informational events and their informational effects. In ETL models, events are unanalyzable primitive elements. In order to represent an intended effect of an informational event, say, e, we must impose an appropriate structure for time-branching trees and agents' indistinguishability relations with respect to the event e. This may be done for some simple cases as presented in Figure 1.5. However, there is no guarantee that we succeed in doing so for more complex situations. How can we come up with a right structure, say for the one of our scenarios, where I peep into the card secretly. In DEL, we could produce the epistemic model in Figure 1.4 that represents the informational state after the informational event by the mechanism of event models and product update. However, ETL does not provide such a systematic procedure and we need to come up with appropriate structures by considerations external to the framework of ETL. Therefore ETL is not suitable for analyzing epistemic dynamics.

1.4 Merging DEL and ETL

As we have seen, DEL is suitable for describing epistemic dynamics, while ETL is suitable for describing protocol information. To obtain a formal framework that

after which $\top$". However, $\top$, being a tautologous truth, is true at any node.

describes both aspects of intelligent interaction together, we merge DEL and ETL. Our key idea is that by repeatedly updating an epistemic model with event models, DEL in effect generates ETL models. First, to represent protocol information, we assign to each world of a given epistemic model a set of sequences of (pointed) event models. We call those assigned sets *protocols*. Sequences in protocols represent the sequences of events that can take place at a given world. Then, by applying the product update mechanism successively to the epistemic model based on the assigned protocols, we generate ETL tree structures. The generated tree structures represent all possible temporal evolutions of agents' initial informational states that accord with protocol information. Below we will make these ideas precise.

1.4.1 Protocols

Let $\mathbb{E}$ be the class of all pointed event models, i.e.

$$\mathbb{E} = \{(\mathcal{E}, e) \mid \mathcal{E} \text{ an event model and } e \in D(\mathcal{E})\}.$$

We denote the set of finite sequences of pointed event models by $\mathbb{E}^*$. (By considerations given in Remark 1.2.11, both $\mathbb{E}$ and $\mathbb{E}^*$ are countable. More on this in Chapter 3)

Definition 1.4.1 (DEL-Protocol) A *DEL-protocol* is a set $\mathsf{P} \subseteq \mathbb{E}^*$ closed under finite prefix. We denote by $Ptcl(\mathbb{E})$ the class of all DEL-protocols, i.e., $Ptcl(\mathbb{E}) = \{\mathsf{P} \mid \mathsf{P} \subseteq \mathbb{E}^* \text{ is closed under initial segments}\}$. ◁

Definition 1.4.2 (State-Dependent DEL-Protocol) Let $\mathcal{M}$ be an arbitrary epistemic model. A *state-dependent DEL-protocol on* $\mathcal{M}$, abbreviated by an *sd*-DEL-protocols is any function $p : Dom(\mathcal{M}) \to Ptcl(\mathbb{E})$. ◁

When there is no confusion, we will simply say protocols or sd-protocols for DEL-protocols or sd-DEL-protocols.

Sd-protocols significantly generalize the usual ETL setting where the *protocol* is assumed to be common knowledge among agents (cf. [21, 51]). An sd-protocol can assign different protocols to different worlds in a given epistemic model. Consequently, what event can happen at a given moment may not be even known by agents. On the other hand, if an sd-protocol p assigns the same protocol, say P, to each world of a given epistemic model, then the protocol P will be common knowledge. This is thus a special kind of sd-protocols and we will call them *uniform protocols*.

Definition 1.4.3 (Uniform Protocol) An sd-DEL-protocol p on $\mathcal{M}$ is a *uniform protocol* on $\mathcal{M}$, if, for all $w \in Dom(\mathcal{M})$, $p(w) = \mathsf{P}$ for some P. Clearly a given DEL-protocol P induces a uniform protocol on any epistemic model. For this reason, when there is no confusion, we drop the specification of epistemic models and call DEL-protocols *uniform protocols*. ◁

State-dependent and uniform protocols are two extreme cases with many interesting cases in between, where agents have only partial knowledge of the type of conversation, experimental protocol, or learning process they are in. One natural example is the assumption that all agents individually know the protocol: for each $w, v \in D(\mathcal{M})$, if wR_iv then $p(w) = p(v)$. In this chapter, we will restrict attention to state-dependent protocols and uniform protocols.

1.4.2 DEL-Generated ETL Models

We now present the main construction of this chapter: generating an ETL model from an initial epistemic model and a (state-dependent or uniform) DEL-protocol. We need some notations. We need to introduce some notations. Let $\sigma = (\mathcal{E}_1, e_1)(\mathcal{E}_2, e_2) \dots (\mathcal{E}_n, e_n) \in$

$\mathbb{E}^*$. We denote the length of σ by $\mathsf{len}(\sigma)$, i.e. $\mathsf{len}(\sigma) = n$. When $k \leq \mathsf{len}(\sigma)$, we write $\sigma_{(k)}$ for the initial segment of σ of length k, and σ_k for the kth component of σ. When $k > \mathsf{len}(\sigma)$ or $k = 0$, σ_k and $\sigma_{(k)}$ are the empty sequence λ. Also we write σ^L and σ^R for $\mathcal{E}_1 \cdots \mathcal{E}_n$ and $e_1 \cdots e_n$ respectively. Thus, for example, $(\sigma^L)_{(3)} = \mathcal{E}_1\mathcal{E}_2\mathcal{E}_3$ and $(\sigma^R)_3 = e_3$. Clearly, $(\cdot)^L, (\cdot)^R$ on the one hand and $(\cdot)_n, (\cdot)_{(n)}$ on the other commute. Thus, we omit parentheses when there is no danger of ambiguity.

Construction with Uniform Protocols

We start by constructing an ETL model from a *uniform* DEL-protocol since the definition is more transparent. However, we stress that the following two definitions are special cases of the more general construction given below (cf. Definition 1.4.8 and Definition 1.4.9).

Definition 1.4.4 (σ-Generated Epistemic Model) Given an epistemic model $\mathcal{M} = (W, \sim, V)$ and a finite sequence of pointed event models σ, we define the *σ-generated epistemic model*, $\mathcal{M}^\sigma = (W^\sigma, \sim^\sigma, V^\sigma)$ as $\mathcal{M} \times \sigma_1^L \times \sigma_2^L \otimes \ldots \otimes \sigma^L_{\mathsf{len}(\sigma)}$. $\triangleleft$

Definition 1.4.5 (ETL Model Generated from a Uniform DEL-Protocol) Let $\mathcal{M}$ be a pointed epistemic model and P a DEL protocol. The *ETL model generated by $\mathcal{M}$ and* P, $\mathsf{Forest}(\mathcal{M}, \mathsf{P})$ is an ETL model $(Dom(\mathcal{M}) \cup \mathbb{E}, H, \sim, V)$, where $(H', \sim', V')$ is such that

- $H' = \bigcup_{\sigma \in \mathsf{P}} W^\sigma$,
- for each $i \in \mathcal{A}$, $\sim'_i := \bigcup_{\sigma \in \mathsf{P}} \sim_i^\sigma$, and
- for each $p \in \mathsf{At}$, $V'(p) := \bigcup_{\sigma \in \mathsf{P}} V^\sigma(p)$

We will omit $Dom(\mathcal{M}) \cup \mathbb{E}$ and write $\mathsf{Forest}(\mathcal{M}, \mathsf{p}) = (H', \sim', V')$, where there is no confusion. Also we identify $(w, \sigma_1, \ldots, \sigma_{\mathsf{len}(\sigma)})$ in $\mathcal{M}^\sigma$ with a history $w\sigma$. $\triangleleft$

$\mathsf{Forest}(\mathcal{M}, \mathsf{P})$ represents all possible evolutions of the system obtained by updating $\mathcal{M}$ with sequences from P. It is straightforward to verify the following proposition.

Proposition 1.4.6 *For every epistemic model $\mathcal{M}$ and a uniform protocol P, $\mathsf{Forest}(\mathcal{M}, \mathsf{P})$ is an ETL model.*

Proof. The proposition immediately follows from the fact that every DEL protocol P is closed under prefixes by Definition 1.4.1. Indeed, when $w\sigma\epsilon$ (or $(w, \sigma_1, \dots, \sigma_{\mathsf{len}(\sigma)}, \epsilon)$) is in $\mathsf{Forest}(\mathcal{M}, \mathsf{P})$, $\sigma \in \mathsf{P}$. This means that $w\sigma$ is in $\mathsf{Forest}(\mathcal{M}, \mathsf{P})$, since $(w, \sigma_1, \dots, \sigma_{\mathsf{len}(\sigma)})$ is in $\mathcal{M}^\sigma$. Therefore, H is closed under finite prefix and $\mathsf{Forest}(\mathcal{M}, \mathsf{P})$ is an ETL model by Definition 1.3.1 QED

Example 1.4.7 (ETL Models Generated from Uniform Protocols) Here is a concrete illustration of the construction. Let $\mathcal{M}$ be an epistemic model that have three worlds, w, v, u, where p is true only at w and v and q is true only at w. An agent 1 cannot distinguish w and v and an agent 2 cannot distinguish v and u. Let P be a uniform protocol consisting of sequences of public announcements such that $\mathsf{P} = \{!p!q, !\neg p!\neg q\}$. $\mathsf{Forest}(\mathcal{M}, \mathsf{P})$ can be visualized as in Figure 1.6. At the bottom, we have three nodes (circled for emphasis) corresponding to $\mathcal{M}$. These three nodes are updated by the sequences of public announcements in P. Consider the sequence $!p!q$. After $\mathcal{M}$ is updated by $!p$, two nodes, $w!p$ and $v!p$, corresponding to $\mathcal{M}\times!p$ are created. Then the model is further updated by $!q$, which creates the node $w!p!q$ corresponding to $(\mathcal{M}\times!p)\times!q$. Similarly for the sequence $!\neg q!\neg p$.

◁

Construction with State-Dependent Protocols

Now we present the method to generate ETL models from sd-DEL-protocols in general. The basic intuition is the same here. We apply product update based on the

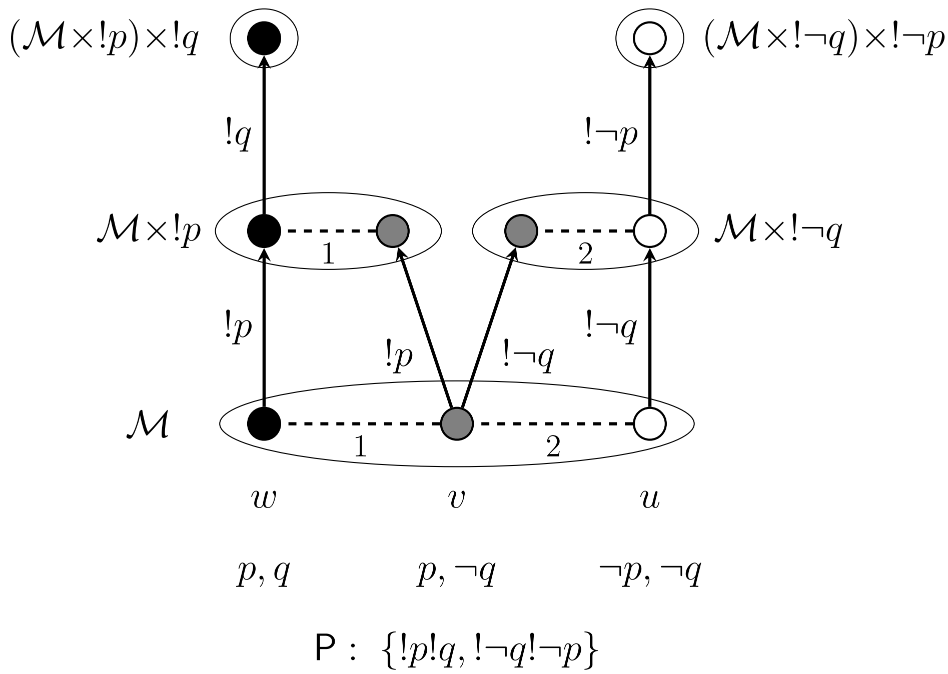

Figure 1.6: ETL Models Generated from Uniform Protocols

sequences of pointed event models that appear in protocols. The process was simple for uniform protocols, since what can happen was the same. For general sd-protocols, what can happen may differ between worlds. Thus, even if an event $(\mathcal{E}, e)$ is in the protocol at a world w in an epistemic model $\mathcal{M}$, it may not be in the protocol at another world v. In this case, whether or not the precondition of e is true at v, we cannot create the new node $v(\mathcal{E}, e)$. This means that, dealing with general sd-protocols, we cannot simply apply sequences of event models allowed in protocols. In applying product update with an event model, we need to exclude the worlds where the event model is not allowed to happen, as well as the world where the precondition is not satisfied. This is taken care of in the following definitions that generalize Definition 1.4.4 and 1.4.5. We will give definitions with comments for clarification, since definitions are much more complicated than the case for uniform protocols.

Definition 1.4.8 (σ^L-Generated Model) Let $\mathcal{M} = \langle W, \sim, V\rangle$ be an epistemic model and p, a state-dependant DEL-protocol on $\mathcal{M}$. Given a sequence $\sigma \in \mathbb{E}^*$, the σ^L*-generated model under* p,

$$\mathcal{M}^{\sigma^L,\mathsf{p}} = (W^{\sigma^L,\mathsf{p}}, \sim_i^{\sigma^L,\mathsf{p}}, V^{\sigma^L,\mathsf{p}}),$$

is defined by induction on the initial segment of σ^L:

- $W^{\sigma^L_{(0)},\mathsf{p}} := W$, for each $i \in \mathcal{A}$, $\sim_i^{\sigma^L_{(0)},\mathsf{p}} := \sim_i$ and $V^{\sigma^L_{(0)},\mathsf{p}} := V$.

(Thus we start with the initial epistemic model $\mathcal{M}$.)

- $w\tau \in W^{\sigma^L_{(n+1)},\mathsf{p}}$ iff

  1. $w \in W$,
  2. $\sigma^L_{(n+1)} = \tau^L$,
  3. $w\tau_{(n)} \in W^{\sigma^L_{(n)},\mathsf{p}}$,
  4. $\tau \in \mathsf{p}(w)$, and
  5. $\mathcal{M}^{\sigma^L_{(n)},\mathsf{p}}, w\tau_{(n)} \models \mathsf{pre}_{\tau^L_n}(\tau^R_{n+1})$.

(By Item 2, every element in $W^{\sigma^L_{n+1},\mathsf{p}}$ is of the form $w\tau$ with τ of the length $n+1$. Item 4 guarantees that only sequences of events models that are allowed by the protocol will be in $W^{\sigma^L_n,\mathsf{p}}$. Item 5 guarantees that the precondition of the event τ_{n+1} is satisfied at the previous stage $\mathcal{M}^{\sigma^L_{(n)},\mathsf{p}}$.)

- For each $w\tau, v\tau' \in W^{\sigma^L_{(n+1)},\mathsf{p}}$ $(0 < n < \mathsf{len}(\sigma^L))$, $w\tau \sim^{\sigma^L_{(n+1)}} v\tau'$ iff

  1. $w\tau_{(n)} \sim_i^{\sigma^L_{(n)},\mathsf{p}} v\tau'_{(n)}$, and
  2. $(\tau^R_{n+1}, (\tau'_{n+1})^R) \in \rightarrow (i)$ in τ^L_{n+1}.

(By Item 1, the two nodes in question must be indistinguishable at the previous stage $\mathcal{M}^{\sigma^L_{(n)},\mathsf{p}}$. By Item 2, they are indistinguishable in τ^L_{n+1} too. Item 2 also guarantees that $\tau^L = (\tau')^L = \sigma^L_{(n+1)}$)

- For each $p \in \mathsf{At}$, $V^{\sigma^L_{(n+1)},\mathsf{p}}(p) = \{w\sigma \in W^{\sigma^L_{(n+1)},\mathsf{p}} \mid w \in V(p)\}$. $\lhd$

(This clause guarantees that propositional valuation stays the same in the course of interaction)

Definition 1.4.9 (DEL-Generated ETL Model) Let $\mathcal{M} = (W, \sim, V)$ be an epistemic model and p a state-dependent DEL protocol on $\mathcal{M}$. An ETL model $\mathsf{Forest}(\mathcal{M}, \mathsf{p}) = (H, \sim', V')$ is defined as follows:

- $H \quad = \quad \{h \mid \text{there is a } w \in W,\ \sigma \in \bigcup_{w \in W} p(w) \text{ with } \ h = w\sigma \in W^{\sigma^L,\mathsf{p}}\}$.
- For all $h, h' \in H$ with $h = w\sigma$ and $h' = v\sigma'$, $h \sim_i h'$ iff $w\sigma \sim_i^{\sigma^L,\mathsf{p}} v\sigma'$.
- For each $p \in \mathsf{At}$ and $h = w\sigma \in H$, $h \in V'(p)$ iff $h \in V^{\sigma^L,\mathsf{p}}(p)$. $\lhd$

The readers are invited to verify that Definition 1.4.4 and Definition 1.4.5 are special cases of Definition 1.4.8 and Definition 1.4.9, respectively, when we restrict attention to uniform protocols (the details are left to the reader).

Proposition 1.4.10 *For every epistemic model $\mathcal{M}$ and an* sd-*DEL-protocol* p *on* $\mathcal{M}$, $\mathsf{Forest}(\mathcal{M}, \mathsf{p})$ *is an ETL model.*

Proof. Straightforward by the reasoning given in 1.4.6. QED

We illustrate this construction with another example.

Example 1.4.11 (DEL-Generated ETL Model) Let us illustrate the construction by the following example. Take an epistemic model $\mathcal{M}$ given in Example 1.4.7.

($\mathcal{M}$ consists of w, v, u, in which p is true only at w, v and q is true only at w.) Let p be an sd-protocol on $\mathcal{M}$ such that $\mathsf{p}(w) = \{!p![i]q\}$, $\mathsf{p}(v) = \{!p![i]q, !\neg q\}$, $\mathsf{p}(u) = \{!p, !\neg q!\top\}$. The ETL model we construct from $\mathcal{M}$ and p can be visualized as in Figure 1.7.

The basic procedure to produce this model is to (i) check what is permitted according to p as a public announcement at each stage, (ii) create a new node if what is permitted is in fact true at the stage and (iii) compute indistinguishability relation for the created stage.

We start from the first stage $\mathcal{M}$ (indicated by the solid line enclosing the three points). In all states in $\mathcal{M}$, $!p$ is assigned by p. Since p is true at w, v, we create the nodes $w!p$ and $v!p$, while we do not create the node "$u!p$" since p is false at u. Also we connect $w!p$ and $v!p$ by the indistinguishability relation (indicated by the horizontal dashed line), since they are indistinguishable in $\mathcal{M}$ $w \sim_1 v$ (where $\sim_1$ is assumed to be an equivalence relation) and $!p \rightarrow_i !p$ (since public announcements are single reflexive points). Note that the created nodes constitute the model obtained by applying $!p$ to $\mathcal{M}$, i.e. the model $\mathcal{M} \times !p$. In this second stage (indicated by the circle enclosing the two nodes), $![i]q$ is permitted and true at both nodes. Thus we produces the third stage consisting of $w!p![i]p$ and $v!p![i]p$. Similarly the nodes $v!\neg q$ and $u!\neg q$ are created since $\neg q$ are permitted and true at v, u, while $w!\neg q$ is not, since $!\neg q$ is neither permitted nor true at u. Furthermore, the node $u!\neg q!\top$ is created but the node "$v!\neg q!\top$" is not present, since $!\top$ is only permitted at $u!\neg q$ though $\top$, being a tautologous truth, is clearly true.

$\triangleleft$

Definition 1.4.12 (Class of DEL-Generated ETL Models) Given a class of state-dependent (or uniform) DEL protocols $\mathbf{X}$, let

$$\mathbb{F}(\mathbf{X}) = \{\mathsf{Forest}(\mathcal{M}, p) \mid \mathcal{M} \text{ an epistemic model and } p \in \mathbf{X}\}.$$

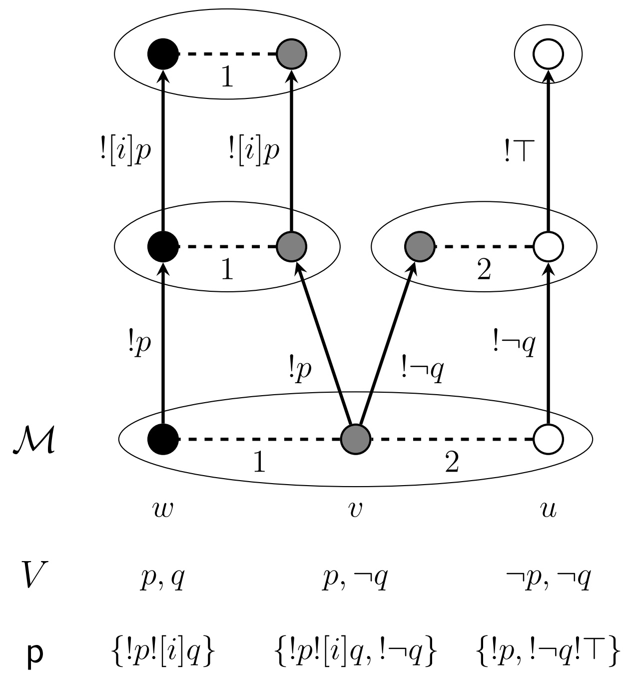

Figure 1.7: DEL-Generated ETL Models

Similarly, If X is a class of uniform protocols,

$$\mathbb{F}(\mathbf{X}) = \{\mathsf{Forest}(\mathcal{M}, \mathsf{P}) \mid \mathcal{M} \text{ an epistemic model and } \mathsf{P} \in \mathbf{X}\}.$$

In particular, when $\mathbf{X}$ is the class of all *sd*-protocols, we denote $\mathbb{F}(\mathbf{X})$ by $\mathbb{F}_{sd}$. Similarly, when $\mathbf{X}$ is the class of all uniform protocols, we denote $\mathbb{F}(\mathbf{X})$ by $\mathbb{F}_{uni}$. ◁

Also if $\mathbf{X} = \{p\}$ (respectively $\mathbf{X} = \{\mathsf{P}\}$) then we write $\mathbb{F}(p)$ (respectively $\mathbb{F}(\mathsf{P})$) instead of $\mathbb{F}(\{p\})$ (respectively $\mathbb{F}(\{\mathsf{P}\})$).

Remark 1.4.13 (Preconditions Given by Epistemic Formulas) As defined in Definition 1.2.4, precondition functions of event models assign to each event an epistemic formula (a formula in $\mathcal{L}_{el}$). This restriction is not substantial in DEL, since formulas in DEL reduce equivalently to formulas in EL by *reduction axioms.* (Section 2.3 and 2.5.1) On the other hand, in our framework, such axioms are not available.

We need to generalize the construction presented above to lift the assumption. We will discuss this in Chapter 3. ◁

1.5 Comparing DEL and ETL

In generating ETL models from DEL models, our framework can provide a formal ground on which DEL and ETL can be compared in a precise manner. For the rest of this chapter, we will give formal comparisons between DEL and ETL based on the framework that has been introduced in the previous sections.

1.5.1 Reinterpreting DEL-Operators as ETL-Operators

Our first observation is that, under a mild condition, we can think of the languages $\mathcal{L}_{del}$ and $\mathcal{L}_{etl}$ as the *same formal language*. In other words, We can reinterpret formulas in DEL as we interpret formulas in ETL. In particular, we can reinterpret the event operator $\langle \mathcal{E}, e \rangle$ in DEL as a labeled temporal modality in ETL as follows. Given $\mathsf{Forest}(\mathcal{M}, \mathsf{p}) \in \mathbb{F}_{sd}$ and h in $\mathsf{Forest}(\mathcal{M}, \mathsf{p})$,

$$\mathsf{Forest}(\mathcal{M}, \mathsf{p}), h \models \langle \mathcal{E}, e \rangle \varphi \quad \text{iff} \quad \mathsf{Forest}(\mathcal{M}, \mathsf{p}), h(\mathcal{E}, e) \models \varphi.$$

(The truth definitions for the other operators are as given in Definition 1.3.3.) The only thing that we have to make sure is that the set of events in ETL, Σ, contains $Dom(\mathcal{M})$ and $\mathbb{E}$ (the set of all pointed event models).

An easy induction shows that this model transformation preserves truth in the following sense.

Proposition 1.5.1 *Let $\mathbb{E}^*$ be the DEL-protocol consisting of* all *finite sequences of pointed event models in DEL. Let $\mathcal{M}$ an epistemic model with $w \in Dom(\mathcal{M})$ (and* *hence* (w) *is a history in* $\mathsf{Forest}(\mathcal{M}, \mathbb{E}^*)$)*: For any formula* $\varphi \in \mathcal{L}_{del}$*,*

$$\mathcal{M}, w \models \varphi \textit{ iff } \mathsf{Forest}(\mathcal{M}, \mathbb{E}^*), (w) \models \varphi.$$

Proposition 1.5.1 explains a common intuition about linking DEL to ETL.

1.5.2 Representation Theorem

Next, we will deal with the question which ETL models can be generated by DEL-protocols. We will show that DEL-generated ETL models have a number of special properties. Our main result is the *representation theorem* (Theorem 1.5.8) that characterizes the class of DEL-generated ETL models by certain properties. The result is an improvement of an existing characterization result found in [62] and provides a precise comparison between the DEL and ETL frameworks.

We start with the result from Van Benthem [62] which characterizes the ETL models resulting from consecutive updates with one single event model. The following properties come from the definition of product update (Definition 1.2.6).

Definition 1.5.2 (Synchronicity, Perfect Recall, Uniform No Miracles) Let $\mathcal{H} = (\Sigma, H, \sim, V)$ be an ETL model. $\mathcal{H}$ satisfies:

- **Synchronicity** iff for all $h, h' \in \mathsf{H}$, if $h \sim_i h'$ then $\mathsf{len}(h) = \mathsf{len}(h')$ ($\mathsf{len}(h)$ is the number of events in h).
- **Perfect Recall** iff for all $h, h' \in \mathsf{H}$, $e, e' \in \Sigma$ with $he, h'e' \in \mathsf{H}$, if $he \sim_i h'e'$, then $h \sim_i h'$
- **Uniform No Miracles** iff for all $h, h' \in \mathsf{H}$, $e, e' \in \Sigma$ with $he, h'e' \in \mathsf{H}$, if there are $h'', h''' \in \mathsf{H}$ with $h''e, h'''e' \in \mathsf{H}$ such that $h''e \sim_i h'''e'$ and $h \sim_i h'$, then $he \sim_i h'e'$. ◁

Additional properties vary depending on the class of DEL protocols considered.

Remark 1.5.3 (Alternative Definition of Perfect Recall) Van Benthem gives an alternative definition of Perfect Recall in [62]:

$$\text{if } he \sim_i h' \text{ then there is an event } f \text{ with } h' = h''f \text{ and } h \sim_i h''.$$

This property is equivalent over the class of ETL models to the above definition of Perfect Recall and synchronicity. We use the above formulation of Perfect Recall in order to stay closer to the computer science literature on verifying multi-agent systems (cf. [21]) and the game theory literature (cf. [11]). ◁

The next property reflects that preconditions of events are formulas of $\mathcal{L}_{el}$.

Definition 1.5.4 (Epistemic Bisimulation Invariance) Let $\mathcal{H} = (\Sigma, H, \sim, V)$ and $\mathcal{H}' = (\Sigma, H, \sim', V)$ be two ETL models. A relation $Z \subseteq H \times H'$ is an **epistemic bisimulation** provided that, for all $h \in H$ and $h' \in H'$, if hZh', then

(prop) h and h' satisfy the same propositional formulas,

(forth) for every $g \in H$, if $h \sim_i g$ then there exists $g' \in H'$ with $h' \sim_i g'$ and gZg'

(back) for every $g' \in H'$, if $h' \sim'_i g'$ then there exists $g \in H$ with $h \sim_i g$ and gZg'.

If Z is an epistemic bisimulation and hZh' then we say *h and h' are epistemically bisimilar*. An ETL model $\mathcal{H}$ satisfies *epistemic bisimulation invariance* iff for all epistemically bisimilar histories $h, h' \in \mathsf{H}$, if $he \in \mathsf{H}$ then $h'e \in \mathsf{H}$. ◁

Another property is needed since we are assuming that product update does not change propositional valuations (see Definition 1.2.4 and 1.2.6. An ETL model $\mathcal{H}$ satisfies *propositional stability* provided for all histories h in $\mathcal{H}$, events e with he in $\mathcal{H}$ and all propositional variables P, if P is true at h then P is true at he. We remark that this property is not crucial for the results in this section and can be dropped provided we allow product update to change the ground facts by revising Definition 1.2.4 and 1.2.6 (cf. [71]).

Finally, we need the following definition:

Definition 1.5.5 (Isomorphism between ETL models) An isomorphic map between two ETL models, $\mathcal{H} = (\Sigma, H, \sim, V)$ and $\mathcal{H}' = (\Sigma', H', \sim', V')$, is a one-to-one function f from Σ onto Σ' such that, for every $\sigma_1, \ldots, \sigma_n, \tau_1, \ldots, \tau_m \in \Sigma$, $i \in \mathcal{A}$ and $p \in \mathsf{At}$,

- if $\sigma_1 \ldots \sigma_n \sim_i \tau_1 \ldots \tau_m$, then $f(\sigma_1) \ldots f(\sigma_n) \sim'_i f(\tau_1) \ldots f(\tau_m)$, and
- if $\sigma_1 \ldots \sigma_n \in V(p)$, then $f(\sigma_1) \ldots f(\sigma_n) \in V'(p)$.

$\triangleleft$

Let $\mathcal{E}$ be a fixed event model and $\mathsf{P}_\mathcal{E}$ be the protocol that consists of all finite sequences of the repetition of $\mathcal{E}$. That is, $\mathsf{P}_\mathcal{E} = (\{(\mathcal{E}, e) \mid e \in Dom(\mathcal{E})\})^*$, where λ is the empty string.

Proposition 1.5.6 (van Benthem [62]) *An ETL model $\mathcal{H}$ is isomorphic to $\mathsf{Forest}(\mathcal{M}, \mathsf{P}_\mathcal{E})$ for some epistemic model $\mathcal{M}$ and event model $\mathcal{E}$ iff $\mathcal{H}$ satisfies propositional stability, synchronicity, perfect recall, uniform no miracles, as well as epistemic bisimulation invariance.*

We do not repeat the proof from [62] here since it is a specific case of our main representation theorem (Theorem 1.5.8) given below. But there are many further DEL-protocols of interest[3]. For example, let PAL be the class of all uniform protocols

[3]Van Benthem & Liu [69] suggest that iterating one large disjoint union of event model involving suitable preconditions can 'mimic' ETL style evolution for more complex protocols with varying event models. We do not pursue this claim here.

consisting of public announcements (with epistemic formula as preconditions). Recall that $\mathbb{F}(\mathsf{PAL}) = \{\mathsf{Forest}(\mathcal{M}, \mathsf{P}) \mid \mathcal{M}$ an epistemic model and $\mathsf{P} \in \mathsf{PAL}\}$. The class $\mathbb{F}(\mathsf{PAL})$ is one of the classes that we will closely study in Chapter 2. The class is characterized by the following representation theorem.

Proposition 1.5.7 (PAL-Generated Models) *An ETL model* $(\Sigma, H, \sim, V)$ *is isomorphic to some model in* $\mathbb{F}(\mathsf{PAL})$ *iff it satisfies the minimal properties of Theorem 1.5.8, and:*

- *for all* $h, h', he, h'e \in H$, *if* $h \sim_i h'$, *then* $he \sim_i h'e$ *(all events are reflexive)*
- *for all* $h, h' \in H$, *if* $he \sim_i h'e'$, *then* $e = e'$ *(no different events are linked).*

This result is also an easy variant of our representation theorem below.

Before proving the representation theorem, a few technical comments are in order. The following proof will construct a DEL-protocol from an ETL model satisfying certain properties. In particular, an event model will be constructed at each level of a given ETL model. Therefore, at each level of the ETL model we will need to specify a *formula* of $\mathcal{L}_{el}$ as a precondition for each primitive event e (cf. Definition 1.2.6). Thus, we already see the role that bisimulation invariance will play in the proof: without it, there is no hope of finding a formula of $\mathcal{L}_{el}$ for a precondition of an event e. However, as is well-known, epistemic bisimulation invariance alone is typically not enough to guarantee the existence of such a formula. More specifically, there are examples of *infinite* sets that are bisimulation closed but not definable by any formula of $\mathcal{L}_{el}$ (however, it will be definable by a formula of epistemic logic with *infinitary* conjunctions — see [10] for a discussion). Thus, if the set of histories at some level in which an event e can be executed is infinite, there may not be a formula of $\mathcal{L}_{el}$ that defines this set to be used as a precondition for e. Such a formula will exist under an appropriate **finiteness assumption**: at each level there are only finitely many histories in which e can be executed, i.e., for each n, the set $\{h \mid he \in \mathsf{H} \text{ and } \mathsf{len}(h) = n\}$ is finite.

Theorem 1.5.8 (Main Representation Theorem) *If an ETL model is isomorphic to some model in $\mathbb{F}_{uni}$ then it satisfies propositional stability, synchronicity, perfect recall, uniform no miracles, as well as epistemic bisimulation invariance.*

If an ETL model $\mathcal{H}$ satisfies the finiteness assumption, propositional stability, synchronicity, perfect recall, uniform no miracles, and epistemic bisimulation invariance, then $\mathcal{H}$ is isomorphic to some model in $\mathbb{F}_{uni}$.

Proof. Suppose that $\mathcal{H} = (\Sigma, H, \sim, V)$ is isomorphic to some model $\mathcal{H}' = (\Sigma', H', \sim', V) \in \mathbb{F}_{uni}$. It suffices to show that $\mathcal{H}'$ satisfies the specified conditions. We show that $\mathcal{H}'$ satisfies epistemic bisimulation invariance, and leave it to the reader to check that $\mathcal{H}$ satisfies the remaining properties. Let $\mathcal{M}$ and P be such that $\mathcal{H}' = \mathsf{Forest}(\mathcal{M}, \mathsf{P})$. Suppose that $h, h' \in H'$, h and h' are epistemically bisimilar, and $he \in H'$ for some event e. We must show $h'e \in H$. By construction (Definition 1.4.5), $h = se_1e_2 \cdots e_ne \in Dom(\mathcal{M} \times \mathcal{E}_1 \times \cdots \mathcal{E}_n \times \mathcal{E})$ where $(\mathcal{E}_1, e_1)(\mathcal{E}_2, e_2) \cdots (\mathcal{E}_n, e_n)(\mathcal{E}, e) \in \mathsf{P}$, $s \in D(\mathcal{M})$, for each $i = 1, \ldots, n$, $e_i \in Dom(\mathcal{E}_i)$ and $e \in Dom(\mathcal{E})$. In order to prove $h'e \in H$, it is enough to show $h'e \in Dom(\mathcal{M} \times \mathcal{E}_1 \times \cdots \mathcal{E}_n \times \mathcal{E})$. This follows from two facts: (1) $h' \in D(M \times \mathcal{E}_1 \times \cdots \times \mathcal{E}_n)$ and (2) $h' \models \mathsf{pre}(e)$. (2) follows from the fact that h and h' are epistemically bisimilar and $\mathsf{pre}(e)$ is assumed to be a formula of $\mathcal{L}_{el}$. (1) follows from the assumption that $h \sim^* h'$.

Suppose $\mathcal{H} = (\Sigma, H, \sim, V)$ is an ETL model satisfying the above properties. We must show there is an epistemic model $\mathcal{M}_{\mathcal{H}}$ and a DEL protocol $\mathsf{P}_{\mathcal{H}}$ such that $\mathcal{H} = \mathsf{Forest}(\mathcal{M}, \mathsf{P})$. For the initial epistemic model, let $\mathcal{M} = (W', \sim', V')$ with $W' = \{h \in H \mid \mathsf{len}(h) = 1\}$, for $h, h' \in W$, define $h \sim_i' h'$ provided $h \sim_i h'$, and for each $p \in \mathsf{At}$,

$V'(p) = V(p) \cap W$.

Call a history $h \in H$ *maximal* if there is no $h' \in H$ such that $h \prec h'$. Now let $\sim^*$ be the reflexive transitive closure of the union of the $\sim_i$ relations. For each maximal history $h \in H$, define the closure of h, denoted $C(h)$, to be the the smallest set that contains all finite prefixes of h, and if $h' \in C(h)$ and $h' \sim^* h''$, then also $h'' \in C(h)$. Note that by perfect recall, $C(h)$ is closed under finite prefixes and is completely connected with respect to the $\sim^*$ relation. It is easy to see that[4] $H = \bigcup\{C(h) \mid h \text{ is a maximal history}\}$.

We define, for each maximal history $h \in H$ and $j = 1, \ldots, \mathsf{len}(h)$, an event model $\mathcal{E}_j^h = (S_j^h, \rightarrow, \mathsf{pre})$ as follows:

1. $S_j^h = \{e \in \Sigma \mid \text{there is a history } h \text{ of length } j \text{ in } \mathsf{H} \text{ with } h = h' \cdot e\}$.

2. For each $e, e' \in S_j^h$, define $e \rightarrow_i e'$ provided there are histories h and h' of length j ending in e and e' respectively, such that $h \sim_i h'$.

3. For each $e \in S_j^h$, let $\mathsf{pre}(e)$ be the formula that characterizes the set $\{h \mid he \in H \text{ and } \mathsf{len}(h) = j\}$. Such a formula does exist, due to epistemic bisimulation invariance and the finiteness assumption.

Finally, let $\mathsf{P} = \{(\mathcal{E})_j^h \mid h \text{ is a maximal history in } H \text{ and } j \leq \mathsf{len}(h)\}$. Clearly, P is a DEL protocol and so is a uniform DEL-protocol. It is easy to see that $\mathsf{Forest}(\mathcal{M}, \mathsf{P})$ and $\mathcal{H}$ have the same set of histories. All that remains is to prove that the epistemic relations are the same in $\mathcal{H}$ and $\mathsf{Forest}(\mathcal{M}, \mathsf{P})$

Claim For each $h_1, h_2 \in H$, $h_1 \sim_i h_2$ in $\mathcal{H}$ iff $h_1 \sim_i h_2$ in $\mathsf{Forest}(\mathcal{M}, \mathsf{P})$.

Proof of Claim. The proof is by induction on the length of h and h' (which can

[4] Note that $C(h)$ only contains finite histories. According to Definition 1.3.1, H only contains *finite* histories. This restriction is not crucial, however, and our result remains true without it.

be assumed to be the same by synchronicity). If $\mathsf{len}(h) = 1$, the claim is immediate by the definition of M.

For the induction step, let $h_1 = h \cdot e$ and $h_2 = h' \cdot e'$. Suppose $h_1 \sim_i h_2$ in $\mathcal{H}$. Then by perfect recall, $h \sim_i h'$ in $\mathcal{H}$. So, by the induction hypothesis, $h \sim_i h'$ in $\mathsf{Forest}(\mathcal{M}, \mathsf{P})$ as well. By the definition given above, $e \rightarrow_i e'$ in the appropriate event model $\mathcal{E}_j^{h_m}$ for a maximal history h_m and $j = \mathsf{len}(h_1)$. It follows by the definition of product update that $h_1 \sim_i h_2$ in $\mathsf{Forest}(\mathcal{M}, \mathsf{P})$.

For the other direction, assume $h_1 \sim_i h_2$ in $\mathsf{Forest}(\mathcal{M}, \mathsf{P})$. Then, by definition of product update, $h \sim_i h'$ in $\mathsf{Forest}(\mathcal{M}, \mathsf{P})$ and $e \longrightarrow_i e'$ in the appropriate event model. By the way the event model is defined, there must be some x and x' with $x \cdot e \sim_i x' \cdot e'$ in $\mathcal{H}$, and therefore, by uniform no miracles, also $h \cdot e \sim_i h' \cdot e'$ in $\mathcal{H}$. QED (of Claim)

An immediate consequence is that $\mathcal{H}$ and $\mathsf{Forest}(\mathcal{M}, \mathsf{P})$ are the same model. QED

This Theorem identifies the *minimal properties* that any DEL generated model must satisfy, and thus it describes exactly what type of agent is presupposed in the DEL framework. The proof generalizes the one in van Benthem & Liu [69], which is an immediate special case. The proof of the characterization of PAL (Proposition 1.5.7) is also a simple variant. The details are left to the reader.

Note that the finiteness assumption can be dropped at the expense of allowing preconditions to come from a more expressive language (specifically, infinitary epistemic logic). Alternatively, we can define the preconditions to be *sets* of histories (instead of formulas of some logical language). A possible compromise is to work with state-dependent protocols instead of uniform protocols. More precisely, in the above proof, we set the precondition of $e \in S_j^h$ to be $\top$, and define a local DEL-protocol p so that, for all $w \in W$, $p(w) = \{(\mathcal{E})_j^h \mid h \text{ is a maximal history in } \mathsf{H} \text{ and } j \geq \mathsf{len}(h)\}$.[5]

[5]This construction suggests that the preconditions of events can be imitated by a trivial precondition, $\top$, and appropriate protocol constraints. For a further discussion about the relation between

Using this observation, we can argue in the same style as above to show the following representation theorems for state-dependent DEL protocols.

Theorem 1.5.9 *An ETL model is isomorphic to some model in $\mathbb{F}_{sd}$ iff it satisfies propositional stability, synchronicity, perfect recall, and uniform no miracles.*

Theorem 1.5.10 *An ETL model $(\Sigma, H, \sim, V)$ is isomorphic to some model in $\mathbb{F}(\mathbf{PAL}_{uni})$ iff it satisfies the minimal properties of Theorem 1.5.9, and the additional properties of Proposition 1.5.7.*

1.6 Conclusion and Discussion

In this chapter, we have developed a formal framework that can describe two important aspects of intelligent interaction, epistemic dynamics and protocol information. We have achieved this by merging the two major systems in intelligent interaction, DEL and ETL. DEL describes epistemic dynamics well by the mechanism of event models and product update, and ETL uses tree structures to represent communicational or observational constraints that are present in various situations of intelligent interaction. These representational frameworks are combined in our framework and allow precise descriptions of epistemic dynamics and protocol information. In merging DEL and ETL, our framework also provides a precise comparison of the two systems. In particular we have proved that representations in DEL can be captured by a special class of ETL models.

Our representation theorems suggest a more general correspondence theory[6] relating natural properties of ETL frames to formulas in suitable modal languages. For instance, what should the language of ETL should be to express natural properties of ETL models? For instance, consider some of the properties mentioned in

preconditions and protocols, see Remark 2.2.15.

[6] [61] discusses related correspondence issues but without our connection to DEL protocols.

Theorem1.5.8. Synchronicity suggests the extension of $\mathcal{L}_{etl}$ with an operator that quantifies over the histories of the same length. (A similar operator is considered in Chapter 2) Perfect recall suggests the addition of an operator that refers to the previous nodes. (This operator is considered in Chapter 3) Theorem 1.5.8 demarcates some important properties of ETL models and this raises the issue of how to design the corresponding formal language to express the properties.

Another natural question to ask is whether a similar result can be obtained in systems that describe *belief revision*. While our framework merges the systems that are designed to describe knowledge, can we investigate a similar project for the systems that describe beliefs? Indeed, van Benthem & Dègremont [67] pursue the question. Recently, *Dynamic Doxastic Logic* (e.g [7, 64]) has been developed to represent agents' belief state and informational change, on the one hand, and temporal structures describing beliefs have been studied (e.g. [12]), on the other. [67] investigates the exact connection between the two kinds of systems. In their framework, doxastic temporal structures are generated by repeatedly updating doxastic models, and a representation theorem is proved to characterize the class of generated temporal structures.

Chapter 2

Logics

In the previous chapter, we have developed a method to generate time-branching tree structures by repeated applications of product update in order to represent temporal evolutions of agents' informational states. In Section 1.5.1, we reinterpreted the language of DEL over the class of these structures called *DEL-generated ETL models*. The goal of this chapter is to pursue this perspective further and study logics of DEL reinterpreted over classes of DEL-generated ETL models.

The main results in this chapter concern complete axiomatizations of classes of DEL-generated ETL models. Each set $\mathbf{X}$ of DEL-protocols induces a class $\mathbb{F}(X)$ of DEL-generated ETL models. This suggests the following natural questions:

- Which DEL protocols generate interesting ETL models?

- Can we axiomatize interesting classes of DEL-generated ETL models?

For some specific combinations of model classes and logical languages, the answers are already already known. For example, recall $\mathbb{E}^*$ is the set of *all* finite sequences of DEL event models — i.e., the forest of *all possible* DEL event structures. Then $\mathbb{F}(\mathbb{E}^*)$ is the class consisting of all DEL-generated ETL models. Its logic (with respect to the language $\mathcal{L}_{DEL}$) can be axiomatized using the well-known reduction axioms: indeed this is the standard completeness theorem for DEL: cf. [6].

In this chapter, we will closely investigate the class of ETL models generated from protocols consisting of *public announcements*. We will reinterpret *Public Announcement Logic* over the class and study the resulted logic, which we will call *Temporal Public Announcement Logic* (TPAL). As we saw in Chapter 1 (Section 1.2.1), public announcements are the simplest kind of model transformations in DEL. Nonetheless, a close study of models generated from them will reveal essential features of our framework and help us develop techniques that can be applied to other logics based on our framework. Indeed we will show that the methods developed for the axiomatization of TPAL can be generalized to obtain axiomatizations for logics over different subclasses of DEL-generated ETL models. We will call the resulted logical systems systems of *Temporal Dynamic Epistemic Logic* (TDEL).

We will proceed as follows. We will start by presenting the system TPAL (Section 2.1) and go on to study various semantic results, such as *model normalization* (Section 2.2). Then we will give axiomatization of TPAL and prove the completeness theorem (Section 2.3). We will also show that the satisfiability problem of TPAL is decidable and discuss how we can incorporate (relativized) common knowledge into the system. After this, we will provide the axiomatization of TPAL restricted to uniform protocols and prove that PAL can be faithfully embedded into TPAL (Section 2.4). Having these results in TPAL, we will extend our system to TDEL. We will first prove the completeness and decidability results for TDEL and its fragments *TDEL(X)* (Section 2.5). Then we will show how other results can be also generalized for TDEL (Section 2.6).

2.1 Temporal Public Announcement Logic

We will start by presenting the system of TPAL. First the definition of protocols must be restricted to public announcements.

Definition 2.1.1 (PAL-Protocol) Let PAL be the set of public announcements in $\mathbb{E}$, i.e. $\{!\varphi \mid \varphi \in \mathcal{L}_{el}\}$.[1] A *PAL-protocol* is a set $\mathsf{P} \subseteq \mathsf{PAL}^*$ closed under finite prefix. We denote the set of PAL-protocols by $Ptcl(\mathsf{PAL})$. A *state-dependent PAL-protocol* (*sd*-PAL-protocol) p on an epistemic model $\mathcal{M}$ is a function that assigns a PAL-protocol to each world in $\mathcal{M}$ a PAL-protocol. We denote the class of *sd*-PAL-protocols by $\mathbb{PAL}$. ◁

By the above notations, we can denote the classes of ETL models generated from *sd*-PAL-protocols and uniform PAL-protocols by $\mathbb{F}(\mathbb{PAL})$ and $\mathbb{F}(Ptcl(\mathsf{PAL}))$ respectively. Below we will axiomatize both $\mathbb{F}(\mathbb{PAL})$ and $\mathbb{F}(Ptcl(\mathsf{PAL}))$.

Definition 2.1.2 (Language of TPAL) Formulas of TPAL is inductively defined as follows:

$$\varphi ::= \top \mid p \mid \neg\varphi \mid \varphi \wedge \varphi \mid [i]\varphi \mid [!\psi]\varphi$$

where $p \in \mathsf{At}$, $i \in \mathcal{A}$ and $\psi \in \mathcal{L}_{el}$.[2] The duals, $\langle i \rangle$ and $\langle !\varphi \rangle$, of $[i]$ and $[!\varphi]$, and the other boolean operators are defined in the standard way. We denote the set of formulas in PAL by $\mathcal{L}_{tpal}$. ◁

By restricting our attention to public announcements, we can simplify many of the definitions in Section 1.4.2.

---

[1] The restriction to $\mathcal{L}_{el}$ here is placed here, since we deal with a subclass of DEL-generated ETL models and DEL-protocols are restricted to the event models with preconditions in $\mathcal{L}_{el}$. (See Definition 1.2.4 and Remark 1.4.13.) As we will see in Chapter 3, this restriction can be lifted by generalizing the method of generating ETL models.

[2] This last restriction is due to the definition of PAL-protocols. See Footnote 1.

Definition 2.1.3 (cf. Definition 1.4.8) Let $\mathcal{M} = (W, \sim, V)$ be an epistemic model, and p an *sd*-PAL-protocol on $\mathcal{M}$. We define

$$\mathcal{M}^{\sigma,p} = (W^{\sigma,\mathsf{p}}, \sim^{\sigma,\mathsf{p}}, V^{\sigma,\mathsf{p}})$$

by induction on the length of σ:

- $W^{\sigma_0,\mathsf{p}} = W$, for each $i \in \mathcal{A}$, $\sim_i^{\sigma_0,\mathsf{p}} = \sim_i$ and $V^{\sigma_0,\mathsf{p}} = V$.
- $w\sigma_{m+1} \in W^{\sigma_{m+1},\mathsf{p}}$ iff (1) $w \in W$, (2) $\mathcal{M}^{\sigma_m,\mathsf{p}}, w\sigma_m \models \varphi_{m+1}$, and also(3) $\sigma_{m+1} \in \mathsf{p}(w)$.
- For each $w\sigma_{m+1}, v\sigma_{m+1} \in W^{\sigma_{m+1},\mathsf{p}}$, $w\sigma_{m+1} \sim_i^{\sigma_{m+1},\mathsf{p}} v\sigma_{m+1}$ iff $w \sim_i v$.
- For each $p \in \mathsf{At}$, $V^{\sigma_{m+1},\mathsf{p}}(p) = \{w\sigma_{m+1} \in W^{\sigma_{m+1},\mathsf{p}} \mid w \in V(p)\}$. ◁

Definition 2.1.4 (cf. Definition 1.4.9) Let $\mathcal{M} = (W, \sim, V)$ be an epistemic model and p an *sd*-PAL-protocol on $\mathcal{M}$. A *PAL-generated ETL model* $\mathsf{Forest}(\mathcal{M}, \mathsf{p}) = (H, \sim', V')$ is defined as follows:

- $H = \{h \mid h \in W^{\sigma,\mathsf{p}}$ for some $\sigma \in \bigcup_{w \in W} \mathsf{p}(w)\}$.
- For all $h, h' \in H$ with $h = w\sigma$ and $h' = v\sigma$ for some $\sigma \in \bigcup_{w \in W} \mathsf{p}(w)$, $h \sim_i h'$ iff $h \sim_i^{\sigma,\mathsf{p}} h'$.
- For each $p \in \mathsf{At}$, $h \in V'(p)$ iff $h \in V^{\sigma,\mathsf{p}}(p)$, where $h = w\sigma$ for some $\sigma \in \bigcup_{w \in W} \mathsf{p}(w)$. ◁

Example 1.4.11 in Chapter 1 illustrates how ETL-generated models are generated from PAL-protocols. The readers are invited to consult the example.

Definition 2.1.5 (Truth) Let $\mathcal{H} \in \mathbb{F}(\mathbb{PAL})$ with $\mathcal{H} = \mathsf{Forest}(\mathcal{M}, p) = (H, \sim, V)$. For a history $h \in \mathsf{H}$, the truth of $\varphi \in \mathcal{L}_{tpal}$ is inductively defined as follows:

$$
\begin{array}{lll}
\mathcal{H}, h \models p & \text{iff} & h \in V(P) \quad (\text{with } P \in \mathsf{At}) \\
\mathcal{H}, h \models \neg\varphi & \text{iff} & \mathcal{H}, h \not\models \varphi \\
\mathcal{H}, h \models \varphi \wedge \psi & \text{iff} & \mathcal{H}, h \models \varphi \text{ and } \mathcal{H}, h \models \psi \\
\mathcal{H}, h \models [i]\varphi & \text{iff} & \forall h' \in H, h \sim_i h' \text{ implies } \mathcal{H}, h' \models \varphi \\
\mathcal{H}, h \models \langle !\psi \rangle \varphi & \text{iff} & h!\psi \in H \text{ and } \mathcal{H}, h!\psi \models \varphi
\end{array}
$$

◁

Based on this semantic framework, the main semantic notions are defined by the standard way.

Definition 2.1.6 (Semantic Notions) Let $\varphi \in \mathcal{L}_{el}$. φ is *satisfiable in* $\mathcal{M}$, if there is w in $\mathcal{M}$ such that $\mathcal{M}, w \models \varphi$. φ is *satisfiable* if φ is satisfiable in some $\mathcal{M}$. φ is *valid in* $\mathcal{M}$, written as $\mathcal{M} \models \varphi$, if $\mathcal{M}, w \models \varphi$ for all w in $\mathcal{M}$. φ is valid, written as $\models \varphi$, if φ is valid for all epistemic models. ◁

2.2 Semantic Results

Next we study semantic results of the system TPAL. We start out by seeing some semantic features that relate PAL and TPAL. Then we will make some simple observations in TPAL, which we will be used later in this chapter. These results will lead us to prove a truth-preservation result under a certain kind of model transformations, which we call *normalization*..

2.2.1 PAL and TPAL

We first make some observations that relate TPAL and PAL. The following proposition is an immediate consequence of Proposition 1.5.1.

Proposition 2.2.1 *Let $\mathcal{M}$ be an epistemic model. Let* PAL *be the set of public announcements in* $\mathbb{E}$*. Then, for any formula* φ *in* $\mathcal{L}_{tpal}$*,*

$$\mathcal{M}, w \models \varphi \text{ in } PAL \quad \text{iff} \quad \mathsf{Forest}(\mathcal{M}, \mathsf{PAL}^*), w \models \varphi \text{ in } TPAL.$$

This proposition also shows that the semantics framework of TPAL generalizes that of PAL. If we permit all formulas to be publicly announced by taking the uniform protocol PAL^*, then the truth of formulas in TPAL corresponds to that in the framework of PAL. Because of the generalization, some basic validities in PAL do not obtain in TPAL.

Proposition 2.2.2 (Public Announcement Operators) *The following properties hold in PAL but not in TPAL.*

(A) $\models \langle !p \rangle \langle !q \rangle \varphi \leftrightarrow \langle !(p \wedge q) \rangle \varphi$ *(with* $p, q \in \mathsf{At}$*)*

(B) $\models \langle !\varphi \rangle \leftrightarrow \varphi$ *(with* $\varphi \in \mathcal{L}_{el}$*)*

Proof. For PAL, **A** and **B** follows straightforwardly from the semantic definition of $\langle !\varphi \rangle$, as given Definition 1.2.2. The readers are invited to give counterexamples against **A** and **B** in TPAL. Also see the following discussion. QED

The validity of **A** in PAL shows that sequences of public announcements are identified with some single announcements in PAL. On the other hand, it is invalid in TAPAL, since a protocol may not allow the single announcement $!(p \wedge q)$ even when it allows the sequence of announcements $!p!q$. The validity of **B** in PAL reflects the general assumption in DEL that every event can happen if its precondition is true. (See Section 1.2.2) TPAL removes this assumption and invalidates the principle, while it assumes the truthfulness of announcements and validates the left-to-right direction. Because the invalidity of the principle, the standard reduction axioms in PAL do not hold. See below in Section 2.3.

Next, we will observe some simple properties of models in TPAL. First, the evaluation of *epistemic* formulas only depends on the 'current stage' of DEL-generated ETL models.

Observation 2.2.3 *Let* $\mathcal{H} = \mathsf{Forest}(\mathcal{M}, \mathsf{p}) \in \mathbb{F}(\mathbb{PAL})$. *For* $\varphi \in \mathcal{L}_{el}$, *for histories* h *in* $\mathsf{Forest}(\mathcal{M}, \mathsf{p})$ *with* $h = w\sigma$ *where* $w \in Dom(\mathcal{M})$ *and* $\sigma \in \mathsf{PAL}^*$,

$$\mathcal{H}, h\sigma \models \varphi \quad \textit{iff} \quad \mathcal{M}^{\sigma,p}, w\sigma \models \varphi.$$

Proof. By a straightforward induction on φ. QED

2.2.2 Simple Observations

Next we see some simple results that we will use later in this chapter. First, a formula $\varphi \in \mathcal{L}_{tpal}$ can describe, at most, what is true after a sequence of announcements bounded in length by the *depth* of φ.

Definition 2.2.4 (Depth of a Formula) Suppose $\varphi \in \mathcal{L}_{tpal}$. The *depth* of φ, denoted $d(\varphi)$, is defined as follows:

- $d(p) = 0$ with $p \in \mathsf{At}$
- $d(\neg\varphi) = d(\varphi)$
- $d(\varphi \wedge \psi) = max(d(\varphi), d(\psi))$
- $d([i]\varphi) = d(\varphi)$
- $d(\langle\psi\rangle\varphi) = 1 + d(\varphi)$

This definition is lifted to a set $X \subseteq \mathcal{L}_{tpal}$ of formulas as follows: $d(X) = \max\{d(\varphi) \mid \varphi \in X\}$. ◁

Given a protocol p on $\mathcal{M}$ and a sequence $\sigma \in \mathsf{PAL}^*$ with $\sigma \in \mathsf{p}(w)$ for some $w \in Dom(\mathcal{M})$, we define a protocol $\mathsf{p}_k^{\sigma<}$ on $\mathcal{M}^{\sigma,\mathsf{p}}$ so that $\mathsf{p}_k^{\sigma<}(w\sigma) = \{\tau \mid \sigma\tau \in \mathsf{p}(w)$ and $\mathsf{len}(\tau) \leq k\}$ for all $w\sigma \in Dom(\mathcal{M}^{\sigma,\mathsf{p}})$. $\mathsf{p}_k^{\sigma<}$ represents which sequences of public announcements of length k or less are allowed in p after σ. Also, we define $\mathsf{p}^{\sigma<}(w\sigma) = \{\tau | \sigma\tau \in \mathsf{p}(w)\}$ when not stating the upper bound. A straightforward induction gives the following result:

Observation 2.2.5 *Let $\mathcal{M}$ be an epistemic model, p a state-dependent protocol on $\mathcal{M}$. For all $w \in Dom(\mathcal{M})$ and $\sigma \in \bigcup_{w \in Dom(\mathcal{M})} \mathsf{p}(w)$,*

$$\mathsf{Forest}(\mathcal{M}, \mathsf{p}), w\sigma \models \varphi \quad \textit{iff} \quad \mathsf{Forest}(\mathcal{M}^{\sigma,\mathsf{p}}, \mathsf{p}_{d(\varphi)}^{\sigma<}), w\sigma \models \varphi$$

and

$$\mathsf{Forest}(\mathcal{M}, \mathsf{p}), w\sigma \models \varphi \quad \textit{iff} \quad \mathsf{Forest}(\mathcal{M}^{\sigma,\mathsf{p}}, \mathsf{p}^{\sigma<}), w\sigma \models \varphi.$$

Next, the histories relevant to evaluate the truth of a given formula $\varphi \in \mathcal{L}_{tpal}$ are only the ones that contain public announcements occurring in φ.

Definition 2.2.6 (Announcement Occurrence Set) The *announcement occurrence set* $AOC(\varphi)$ of a $TAPAL$-formula φ is defined inductively as follows:

- $AOC(p) = \emptyset$ with $p \in \mathsf{At}$
- $AOC(\neg\varphi) = AOC(\varphi)$
- $AOC(\varphi \wedge \psi) = AOC(\varphi) \cup AOC(\psi)$
- $AOC([i]\varphi) = AOC(\varphi)$

- $AOC(\langle !\psi\rangle\varphi) = \{!\psi\} \cup AOC(\varphi)$

Given a sequence $\sigma = !\varphi_1 \dots !\varphi_n \in \Sigma^*_{pal}$, we define

$$AOC(\sigma) := AOC(\varphi_1) \cup \cdots \cup AOC(\varphi_n).$$

Furthermore, given an *sd-PAL*-protocol p on $\mathcal{M} = (W, \sim, V)$, we define

$$AOC(\mathsf{p}) := \bigcup_{\{\sigma \mid \exists w \in W : \sigma \in \mathsf{p}(w)\}} AOC(\sigma).$$

◁

Given a state-dependent protocol p on a model $\mathcal{M}$, for $w \in Dom(\mathcal{M})$ define $(\mathsf{p}(w))_{AOC(\varphi)}$ as follows:

$$(\mathsf{p}(w))_{AOC(\varphi)} = \{\sigma \in \mathsf{p}(w) \mid \text{ for each } !\theta \text{ in } \sigma, !\theta \in AOC(\varphi)\}.$$

This set represents announceable sequences of announcements at w that only consist of public announcements occurring in φ. Now we can show the following by a straightforward induction.

Observation 2.2.7 *Suppose $\mathcal{M}$ is an epistemic model and p and q are two protocols on $\mathcal{M}$. Suppose $(\mathsf{p}(v))_{AOC(\varphi)} = (\mathsf{q}(v))_{AOC(\varphi)}$ for all $v \in Dom(\mathcal{M})$. Then for all $w \in Dom(\mathcal{M})$ and $\varphi \in \mathcal{L}_{tpal}$,*

$$\mathsf{Forest}(\mathcal{M}, \mathsf{p}), w \models \varphi \quad \textit{iff} \quad \mathsf{Forest}(\mathcal{M}, \mathsf{q}), w \models \varphi.$$

Finally we state the variant of Proposition 2.2.1. Given a formula $\varphi \in \mathcal{L}_{tpal}$ and an epistemic model $\mathcal{M}$, define p_φ so that, for all $w \in Dom(\mathcal{M})$, $\mathsf{p}_\varphi(w) = \{!\theta_1 \cdots \theta_k \mid !\theta_i \in AOC(\varphi)\ (1 \leq i \leq k) \text{ for some } k\}$. In the light of the above lemma, p_φ represents the sequences of public announcements that are relevant to the truth value of φ. We can show by an easy induction that the generated ETL model from p_φ preserves the truth value of φ in PAL in the following sense.

Observation 2.2.8 *Let $\varphi \in \mathcal{L}_{tpal}$. Then*

$$\mathcal{M}, w \models \varphi \textit{ in PAL} \quad \textit{iff} \quad \mathsf{Forest}(\mathcal{M}, \mathsf{p}_\varphi), w \models \varphi \textit{ in TPAL}.$$

2.2.3 Model Normalization

Next, we turn our attention to the following distinctive property of models in TPAL. Given a set X of public announcements, models in TPAL can be transformed so that they contain only the public announcements in X and public announcements with tautologous preconditions (call such public announcements *tautologous public announcements*), while the truth of the formulas expressed with public announcements in X is preserved. We call this model transformation *normalization*. This transformation is a general property of model in our framework and will be used in Chapter 3.

To formulate the transformation, we need some definitions. Let $\varphi_0, \varphi_1, \ldots$ and $\top_0, \top_1, \ldots$ be a pair of (possibly infinite) sequences of formulas in $\mathcal{L}_{el}$ such that (i) $\top_i$ is tautologous and (ii) $\varphi_i \neq \varphi_j$ and $\top_i \neq \top_j$ for all $i, j \geq 0$.

Definition 2.2.9 (Normalization of Sequences) Given a sequence $\sigma \in \mathsf{PAL}^*$, we define $\sigma[!\top_0/!\varphi_0, !\top_1/!\varphi_1, \ldots]$ to be the result of replacing all occurrences of $!\varphi_i$ in σ with $!\top_i$ for all i. $\triangleleft$

The idea of normalization is to replace public announcements with tautologous public announcements, preserving tree structures.

Definition 2.2.10 (Normalization of Models) Let p be an sd-protocol on $\mathcal{M}$. Also let $\mathcal{H} = \mathsf{Forest}(\mathcal{M}, \mathsf{p}) = (H, \sim, V)$. Define $\mathcal{H}[!\top_0/!\varphi_0, !\top_1/!\varphi_1, \ldots] = (H', \sim', V')$ by:

- $H' := \{h[!\top_0/!\varphi_0, !\top_1/!\varphi_1, \ldots] \mid h \in H\}$
- $(h[!\top_0/!\varphi_0, !\top_1/!\varphi_1, \ldots], g[!\top_0/!\varphi_0, !\top_1/!\varphi_1, \ldots]) \in \sim' (i)$ iff $(h, g) \in \sim (i)$
- $V'(p) := \{h[!\top_0/!\varphi_0, !\top_1/!\varphi_1, \ldots] \mid h \in V(p)\}$

$\triangleleft$

Now we need to confirm that, given that $\mathcal{H}$ is in $\mathbb{F}(\mathbb{PAL})$, $\mathcal{H}[!\top_0/!\varphi_0, !\top_1/!\varphi_1, \ldots]$ is also in $\mathbb{F}(\mathbb{PAL})$. Indeed, when $h!\varphi_i$ is in $\mathcal{H}$, $h!\top_i$ must be in $\mathcal{H}[!\top_0/!\varphi_0, !\top_1/!\varphi_1, \ldots]$ since the tautologous formula $!\top_i$ is guaranteed to be true at h. Also if $(h, g) \in \sim (i)$, the corresponding nodes for h and g will be indistinguishable by the construction of models in TPAL. We state this fact more precisely as follows.

Definition 2.2.11 (Protocol above σ in $\mathcal{H}$) Let $\mathcal{H} = \mathsf{Forest}(\mathcal{M}, \mathsf{p}) \in \mathbb{F}(\mathbb{PAL})$. Let $\sigma \in \mathsf{PAL}^*$. Then define $\mathsf{p}^{\mathcal{H},\sigma<}$ on $\mathcal{M}^{\mathsf{p},\sigma}$ so that

$$\mathsf{p}^{\mathcal{H},\sigma<}(w\sigma) = \{\tau \mid w\sigma\tau \text{ in } \mathcal{H}\}$$

$\triangleleft$

Observation 2.2.12 *Let $\varphi_0, \varphi_1, \ldots$ be a sequence of formulas in $\mathcal{L}_{el}$ and $\top_0, \top_1, \ldots$, a sequence of tautologous formulas in $\mathcal{L}_{el}$. Suppose, for every $i, j \leq 0$, if $i \neq j$, then $\varphi_i \neq \varphi_j$ and $\top_i \neq \top_j$. Let $\mathcal{H} = \mathsf{Forest}(\mathcal{M}, \mathsf{p})$. Put $\mathcal{G} = \mathcal{H}[!\top_0/!\varphi_0, !\top_1/!\varphi_1, \ldots]$.*

$$\mathcal{G} = \mathsf{Forest}(\mathcal{M}, \mathsf{p}^{\mathcal{G},\lambda<})$$

where λ is the empty sequence. $\triangleleft$

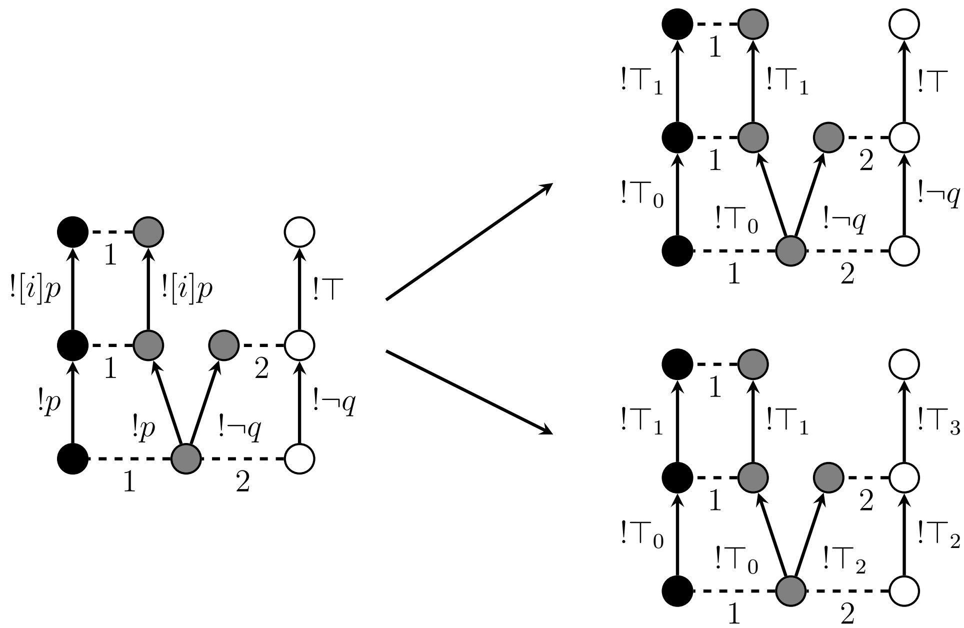

Figure 2.1: Model Normalization

Example 2.2.13 (Normalization) Let us illustrate the idea of normalization. In Figure 2.1, the model on the left visualizes a DEL-generated ETL model (the one discussed in Figure 1.7). Let us denote the model by $\mathcal{H}$. The upper-right model is the ETL model $\mathcal{H}[!\top_0/!p, !\top_1/![i]p]$ obtained by replacing $!p$ and $![i]p$ with $\top_0$ and $\top_0$ respectively. The lower-right model is the ETL model $\mathcal{H}[!\top_0/!p, !\top_1/![i]p, !\top_2/!\neg q, !\top_3/!\top]$ obtained by replacing $\neg q$ and $\top$ with $\top_2$ and $\top_3$ additionally. $\lhd$

Now we prove the truth-preservation result with respect to model normalization. Given a formula $\varphi \in \mathcal{L}_{tapal}$, even if we replace the announcements that do not occur in φ with "new" tautologous formulas in a DEL-generated ETL model, the truth of φ is preserved.

Proposition 2.2.14 (Normalization) *Let* $\mathcal{H} = \mathsf{Forest}(\mathcal{M}, \mathsf{p}) \in \mathbb{F}(\mathbb{PAL})$. *Let* X *be a finite subset of* PAL. *Furthermore, let* $!\varphi_0, !\varphi_1 \ldots$ *be an enumeration of public announcements in* $\mathsf{PAL} \backslash X$ *without repetition, and* $!\top_0, !\top_1, \ldots$ *be an enumeration of* *tautologous public announcements in* $\mathsf{PAL}\backslash X$ *without repetition. Then, for every* h *and TPAL-formula* φ *such that* $AOC(\varphi) \subseteq X$*,*

$$\mathcal{H}, h \models \varphi \quad \Leftrightarrow \quad \mathcal{H}[!\top_0/!\varphi_0, !\top_1/!\varphi_1, \dots], h[!\top_0/!\varphi_0, !\top_1/!\varphi_1, \dots] \models \varphi$$

Proof. Let $h = w\sigma$ with w in $\mathcal{M}$ and $\sigma \in \mathsf{p}(w)$. Denote by $\mathcal{H}' = (H', \sim', V')$ and h' respectively the normalization of $\mathcal{H} = (H, \sim, V)$ and the element (in $\mathcal{H}'$) corresponding h. The proof is by a straightforward induction on φ. The base and boolean cases are clear. For the case where φ is of the form $[i]\psi$, note that the normalization $[!\top_0/!\varphi_0, !\top_1/!\varphi_1, \dots]$ can be considered as an isomorphic map from H to H' by Definition 2.2.10, given our assumption that all $\top_i \neq \top_j$ and $\varphi_i \neq \varphi_j$ for all distinct i, j. Therefore we have $\{g \mid h' \sim_i' g'\} = \{g \mid h \sim_i g\}$. Given this consideration, the case of $[i]$ is immediate by IH.

Assume that φ is of the form $\langle !\theta \rangle \psi$. Assume LHS. Then we have $\mathsf{Forest}(\mathcal{M}, \mathsf{p}), h!\theta \models \psi$. Since $AOC(\psi) \subseteq AOC(\langle !\theta \rangle \psi) \subseteq X$, we can by IH obtain

$$\mathcal{H}[!\top_0/!\varphi_0, !\top_1/!\varphi_1, \dots], h!\theta[!\top_0/!\varphi_0, !\top_1/!\varphi_1, \dots] \models \psi.$$

Since $\theta \in AOC(\langle !\theta \rangle \psi)$, we have

$$\mathcal{H}[!\top_0/!\varphi_0, !\top_1/!\varphi_1, \dots], h[!\top_0/!\varphi_0, !\top_1/!\varphi_1, \dots]!\theta \models \psi.$$

Therefore, we have $\mathcal{H}[!\top_0/!\varphi_0, !\top_1/!\varphi_1, \dots], h[!\top_0/!\varphi_0, !\top_1/!\varphi_1, \dots] \models \langle !\theta \rangle \psi$. QED

Remark 2.2.15 (Preconditions and Protocol Information) As we will see below (Section 2.6.1), the idea of normalization can be generalized to the full class of DEL-generated ETL models and a similar result can be obtained for Temporal Dynamic Epistemic Logic.

One thing that this theorem illustrates is that preconditions of events can be imitated by trivial preconditions and appropriate adjustment of protocols. For instance, the public announcement $!p$ can only happen at the worlds where p is true and, as a result, it eliminates from a model the worlds at which p is false. However, we can make the trivial public announcement $!\top$ work in the same way by adjusting protocols so that $!\top$ can happen at the worlds where p is true. Note that the argument for Theorem 1.5.9 appealed to the same type of consideration. $\lhd$

Remark 2.2.16 (Normalization and Observation 2.2.7) Observation 2.2.7 should not be conceived as the generalization of Proposition 2.2.14. Observation 2.2.7 only refers to certain initial segments of PAL-protocols, whereas Proposition 2.2.14 refers to all parts of PAL-protocols. Also in Observation 2.2.7, two compared ETL models do not have to have the same tree structure, whereas an ETL model and its normalization share the same structure by construction. $\lhd$

2.3 Complete Axiomatization

Before we move on to the axiomatization of TPAL, the following fact about the system of PAL should be mentioned for contrast. In PAL, the following *reduction axioms* are valid:

$$\begin{array}{lcl} \langle !\theta\rangle p & \leftrightarrow & \theta \wedge p \quad (\text{with } p \in \mathsf{At}) \\ \langle !\theta\rangle \neg\varphi & \leftrightarrow & \theta \wedge \neg\langle !\theta\rangle\varphi \\ \langle !\theta\rangle(\varphi \wedge \psi) & \leftrightarrow & \langle !\theta\rangle\varphi \wedge \langle !\theta\rangle\psi \\ \langle !\theta\rangle[i]\varphi & \leftrightarrow & \theta \wedge [i](\theta \rightarrow \langle !\theta\rangle\varphi). \end{array}$$

Using these equivalences, we can transform any PAL-formulas equivalently into some EL-formulas ([6]). (We can push public announcement operators toward the right until they precedes propositional letters, and then apply the first reduction axiom.) Thus, with these axioms, the completeness of PAL is guaranteed by the completeness of EL. In fact, the type of compositional analysis via reduction axioms can be applied to the full DEL and provides a strong means to provide the complete axiomatization.

The validity of the reduction axioms in PAL depends crucially on the equivalence between the truth of a formula θ and the availability of $!\theta$, more precisely, $\theta \leftrightarrow \langle !\theta \rangle \top$. ($\langle !\theta \rangle \top$ literally reads as "$!\theta$ can happen after which $\top$". Given $\top$ is tautologous, we can read $\langle \theta \rangle \top$ as "$!\theta$ can happen.") For instance, in the first reduction axiom, $\langle !\theta \rangle p$ on the left, which reads as "$!\theta$ can happen after which p", is equated with $\theta \wedge p$ on the right, which only claims the truth of θ and p. However, as we saw in Proposition 2.2.2, the equivalence does not hold in TPAL. Consequently we cannot appeal to the compositional analysis via reduction axioms to axiomatize the system of TPAL. We need to redo the work.

2.3.1 Axiomatic System

Definition 2.3.1 (Axiomatization of TPAL) The axiomatization TPAL consists of the following axiom schemes and inference rules.

Axioms

PC Propositional validities

i**K** $[i](\varphi \rightarrow \psi) \rightarrow ([i]\varphi \rightarrow [i]\psi)$

!K $[!\theta](\varphi \rightarrow \psi) \rightarrow ([!\theta]\varphi \rightarrow [!\theta]\psi)$

R1 $\langle !\theta \rangle p \leftrightarrow \langle !\theta \rangle \top \wedge p$ (with $p \in \mathsf{At}$)

R2 $\langle !\theta \rangle \neg\varphi \leftrightarrow \langle !\theta \rangle \top \wedge \neg \langle !\theta \rangle \varphi$

R3 $\langle !\theta \rangle (\varphi \wedge \psi) \leftrightarrow \langle !\theta \rangle \varphi \wedge \langle !\theta \rangle \psi$ [3]

[3] **R3** follows from **!N** and **!K**. However we included it to make the contrast with the PAL reduction axioms explicit.

R4 $\langle !\theta\rangle[i]\varphi \leftrightarrow \langle !\theta\rangle\top \wedge [i](\langle !\theta\rangle\top \rightarrow \langle !\theta\rangle\varphi)$

A1 $\langle !\theta\rangle\top \rightarrow \theta$

Inference Rules

MP If $\vdash \varphi$ and $\vdash \varphi \rightarrow \psi$, then $\vdash \psi$.

i**N** If $\vdash \varphi$, then $\vdash [i]\varphi$ for any $i \in \mathcal{A}$.

!**N** If $\vdash \varphi$, then $\vdash [!\theta]\varphi$ for any $!\psi \in \mathsf{PAL}$. ◁

First note that **R1-4** are similar to the reduction axioms in PAL. However they differ from reduction axioms in PAL in terms of occurrences of the formulas of the form $\langle !\theta\rangle\top$ instead of simply θ. Because of this, formulas in TPAL do not reduce to formulas in EL. Second, A1 gives the only one way of the equivalence between the truth of a formula and its availability as a public announcement. A1 reads as "only true formulas can be announced". The converse is invalid.

Remark 2.3.2 (Without Uniform Substitution) Notice that TPAL does not satisfy uniform substitution. For one thing, axiom **R1** only applies to atomic propositions $p \in \mathsf{At}$. Furthermore, the preconditions of public announcements are in $\mathcal{L}_{el}$. Thus, for example, $\langle !\langle !\theta_1\rangle\top\rangle p \leftrightarrow \langle !\langle !\theta_2\rangle\top\rangle\top \wedge p$ is *not* an instance of axiom **R1**. This restriction will be lifted in Chapter 3. ◁

Before turning to the main result of this Section, we consider axiom R4 in more detail. Consider the following three variations of R4:

1. $\langle !\theta\rangle[i]\varphi \leftrightarrow !\theta \wedge [i]\langle !\theta\rangle\varphi$

2. $\langle !\theta\rangle[i]\varphi \leftrightarrow \langle !\theta\rangle\top \wedge [i](\theta \rightarrow \langle !\theta\rangle\varphi)$

3. $\langle !\theta\rangle[i]\varphi \leftrightarrow \langle !\theta\rangle\top \wedge [i](\langle !\theta\rangle\top \rightarrow \langle !\theta\rangle\varphi)$

Each of these axioms represent a different assumption about the underlying protocol and how that affects the agents' knowledge. The first is the usual PAL reduction axiom and assumes a specific protocol (which is common knowledge) where all true formulas are always available for announcement. The second (weaker) axiom is valid when there is a fixed protocol that is common knowledge (cf. Section 2.4.1). Finally, the third is an instance of **R4** and is thus true for all protocols.

2.3.2 Completeness Proof

Now our goal is to prove the following theorem:

Theorem 2.3.3 TPAL *is sound and strongly complete with respect to the class of ETL models* $\mathbb{F}(\mathbb{PAL})$.

The proof is a variant of the standard Henkin construction. We construct the canonical ETL model from the set of maximal consistent sets in TPAL (*mcs* below). The main idea is that each *mcs* defines sequences of 'legal' public announcements which we use to define a canonical state-dependent protocol. We start by defining the set of legal histories and a function λ_n that assigns maximally consistent sets to each node on a history.

Definition 2.3.4 (Legal Histories) Let W_0 be the set of all maximal consistent sets in TPAL. We define λ_n and H_n $(0 \leq n \leq d(\Sigma))$ are defined as follows:

- Set $H_0 = W_0$, and for each $w \in H_0$, $\lambda_0(w) = w$.
- Let $H_{n+1} = \{h!\theta \mid h \in H_n \text{ and } \langle !\theta\rangle\top \in \lambda_n(h)\}$. For each $h = h'!\theta \in H_{n+1}$, define $\lambda_{n+1}(h) = \{\varphi \mid \langle !\theta\rangle\varphi \in \lambda_n(h')\}$. ◁

We first confirm that each map λ_n is well-defined.

Lemma 2.3.5 *For each $n \geq 0$, for each $\sigma \in H_n$, $\lambda_n(\sigma)$ is maximally consistent.*

Proof. The proof is by induction on n. The case $n = 0$ is by definition. Suppose that the statement holds for H_n and λ_n. Suppose $\sigma \in H_{n+1}$ with $\sigma = \sigma'!\theta$. By the induction hypothesis, $\lambda_n(\sigma')$ is an *mcs*. Furthermore, by the construction of H_{n+1}, $\langle !\theta \rangle \top \in \lambda_n(\sigma)$. Therefore, $\lambda_{n+1}(\sigma) \neq \emptyset$. Let $\varphi \in \mathcal{L}_{tpal}$. Since $\lambda_n(\sigma')$ is an *mcs*, either $\langle !\theta \rangle \varphi \in \lambda_n(\sigma')$ or $\neg \langle !\theta \rangle \varphi \in \lambda_n(\sigma')$. If $\langle !\theta \rangle \varphi \in \lambda_n(\sigma')$, $\varphi \in \lambda_{n+1}(\sigma)$ by construction. If $\neg \langle !\theta \rangle \varphi \in \lambda_n(\sigma')$, by axiom **R2**, we have $\langle !\theta \rangle \neg \varphi \in \lambda_n(\sigma')$. Thus, by construction, $\neg \varphi \in \lambda_{n+1}(\sigma)$. Thus, for all $\varphi \in \mathcal{L}_{tpal}$, either $\varphi \in \lambda_{n+1}(\sigma)$ or $\neg \varphi \in \lambda_{n+1}(\sigma)$.

To show that λ_{n+1} is consistent, assume toward contradiction that there are formulas $\varphi_1, ..., \varphi_m \in \lambda_{n+1}(\sigma)$ such that $\vdash \bigwedge_{i=1}^m \varphi \to \bot$. Using standard modal reasoning, $\vdash \langle !\theta \rangle \top \to \bigvee_{i=1}^m \langle !\theta \rangle \neg \varphi_i$. Since $\langle !\theta \rangle \top \in \lambda_n(\sigma')$, we have $\bigvee_{i=1}^m \langle !\theta \rangle \neg \varphi \in \lambda_n(\sigma')$. And so, since $\lambda_n(\sigma')$ is a maximally consistent set, there is some j with $1 \leq j \leq m$ and $\langle !\theta \rangle \neg \varphi_j \in \lambda_n(\sigma')$. Using axioms **R2**, we have $\neg \langle !\theta \rangle \varphi_j \in \lambda_n(\sigma')$. By construction of $\lambda_{n+1}(\sigma)$ we have for each $i = 1, \ldots, m$, $\langle !\theta \rangle \varphi_i \in \lambda_n(\sigma')$. This contradicts the fact that $\lambda_n(\sigma')$ is consistent. QED

We now define a canonical ETL model $\mathcal{H}^{can}$. We start by defining $\mathcal{H}_0^{can} = (H_0, \sim^0, V^0)$. For this, we use the usual definitions:

- For $w, v \in H_0$, let $w \sim_i^0 v$ iff $\{\varphi \mid [i]\varphi \in w\} \subseteq v$.
- For each $P \in \mathsf{At}$ and $w \in H_0$, $P \in V^0(w)$ iff $P \in w$.

Definition 2.3.6 (Canonical Model) The canonical model $\mathcal{H}^{can} = (H^{can}, \sim^{can}, V^{can})$ is defined as follows:

- $H^{can} = \bigcup_{i=0}^{\infty} H_i$.
- For each $h, h' \in H^{can}$ with $h = w\sigma$ and $h' = w'\sigma'$, let $h \sim_i^{can} h'$ iff (1) $\sigma = \sigma'$ and (2) $w \sim_i^0 v$.

- For every $p \in \mathsf{At}$ and $h = w\sigma \in H^{can}$, $w\sigma \in V^{can}(p)$ iff $w \in V^0(p)$. $\lhd$

Given $h \in H^{can}$ with $h = w!\theta_1 \cdots !\theta_n$, we write $\lambda(h)$ for $\lambda_n(h)$. We now show that the canonical model $\mathcal{H}^{can}$ works as intended:

Lemma 2.3.7 (Truth Lemma) *For every $\varphi \in \mathcal{L}_{tpal}$, for each $h \in H^{can}$,*

$$\varphi \in \lambda(h) \quad \textit{iff} \quad \mathcal{H}^{can}, h \models \varphi.$$

Proof. We show by induction on the structure of $\varphi \in \mathcal{L}_{tpal}$ that for each $h \in H^{can}$, $\varphi \in \lambda(h)$ iff $\mathcal{H}^{can}, h \models \varphi$. The base and the boolean cases are straightforward. For the knowledge modality, let $h \in H^{can}$ with $h = w!\theta_1 \cdots !\theta_n$ and assume $[i]\psi \in \lambda(h)$. Suppose $h' \in H^{can}$ with $h \sim_i h'$. By construction of the canonical model, we know that $h' = v!\theta_1 \cdots !\theta_n$ for some $v \in H_0$ with $w \sim_i^0 v$. By Definition 2.3.4, since $[i]\psi \in \lambda(w!\theta_1 \cdots !\theta_n)$, we have $\langle !\theta_n \rangle [i]\psi \in \lambda(w!\theta_1 \cdots !\theta_{n-1})$. Using Axiom **R4**, we have $[i](\langle !\theta_n \rangle \top \to \langle !\theta_n \rangle \psi) \in \lambda(w!\theta_1 \cdots !\theta_{n-1})$. Continuing this way, we have

$$[i](\langle !\theta_1 \rangle \top \to \langle !\theta_1 \rangle (\langle !\theta_2 \rangle \top \to \langle !\theta_2 \rangle (\cdots \langle !\theta_{n-1} \rangle (\langle !\theta_n \rangle \top \to \langle !\theta_n \rangle \psi) \cdots)) \in w.$$

By Definition 2.3.6, since $h \sim_i^{can} h'$, we have $w \sim_i^0 v$. Hence,

$$\langle !\theta_1 \rangle \top \to \langle !\theta_1 \rangle (\langle !\theta_2 \rangle \top \to \langle !\theta_2 \rangle (\cdots \langle !\theta_{n-1} \rangle (\langle !\theta_n \rangle \top \to \langle !\theta_n \rangle \psi) \cdots) \in v.$$

Now note that

$$\langle !\theta_1 \rangle \top \in \lambda(w), \langle !\theta_2 \rangle \top \in \lambda(w!\theta_1), \ldots, \langle !\theta_n \rangle \top \in \lambda(w!\theta_1 ... !\theta_{n-1}).$$

Thus, we have

$$\langle !\theta_2 \rangle \top \to \langle !\theta_2 \rangle (\cdots \langle !\theta_{n-1} \rangle (\langle !\theta_n \rangle \top \to \langle !\theta_n \rangle \psi) \cdots) \in \lambda(v!\theta_1)$$

$$\langle !\theta_3\rangle\top \to \langle !\theta_3\rangle(\cdots\langle !\theta_{n-1}\rangle(\langle !\theta_n\rangle\top \to \langle !\theta_n\rangle\psi)\cdots) \in \lambda(v!\theta_1!\theta_2)$$

$$\vdots$$

$$\langle !\theta_n\rangle\psi \in \lambda(v!\theta_1\cdots!\theta_{n-1})$$

Therefore, $\psi \in \lambda(v!\theta_1\cdots!\theta_n) = \lambda(h')$. By the induction hypothesis, $\mathcal{H}^{can}, h' \models \psi$. Therefore, $\mathcal{H}^{can}, h \models [i]\psi$, as desired.

For the other direction, let $h \in H^{can}$ and assume $[i]\psi \notin \lambda(h)$. For simplicity, we let $h = w!\theta$ with $w \in W_0$ and $\theta \in \mathcal{L}_{el}$. The argument can easily be generalized to deal with the general case along the lines of the argument above. Since $\lambda(h)$ is an *mcs*, we have $\neg[i]\psi \in \lambda(h)$. Thus, by Definition 2.3.4, $\langle !\theta\rangle\neg[i]\psi \in \lambda(w)$. Using axiom **R2**, $\neg\langle !\theta\rangle[i]\psi \in \lambda(w)$; and so, by Axiom **R4**, $\neg\langle !\theta\rangle\top \vee \neg[i](\langle !\theta\rangle\top \to \langle !\theta\rangle\psi) \in \lambda(w)$. Since $\langle !\theta\rangle\top \in \lambda(w)$ by construction, it follows that $\neg[i](\langle !\theta\rangle\top \to \langle !\theta\rangle\psi) \in \lambda(w)$. Now consider the set $v_0 = \{\theta \mid [i]\gamma \in \lambda(w)\} \cup \{\neg(\langle !\theta\rangle\top \to \langle !\theta\rangle\psi)\}$. We claim that this set is consistent. Suppose not. Then, there are formulas $\gamma_1, \ldots, \gamma_m$ such that $\vdash \bigwedge_{j=1}^m \gamma_j \to \langle !\theta\rangle\top \to \langle !\theta\rangle\psi$ and for $j = 1, \ldots, m$, $[i]\gamma_j \in \lambda(w)$. By standard modal reasoning, $\vdash \bigwedge_{j=1}^m [i]\gamma_j \to [i](\langle !\theta\rangle\top \to \langle !\theta\rangle\psi)$. This implies that $[i](\langle !\theta\rangle\top \to \langle !\theta\rangle\psi) \in \lambda(w)$. However, this contradicts the fact that $\neg[i](\langle !\theta\rangle\top \to \langle !\theta\rangle\psi) \in \lambda(w)$, since $\lambda(w)$ is an *mcs*. Now using standard arguments (Lindenbaum's lemma), there exists a maximally consistent set v with $v_0 \subseteq v$. By the construction of v, we must have $w \sim_i^0 v$ and thus $w!\theta \sim_i^{can} v!\theta$. Also, since $\neg(\langle !\theta\rangle\top \to \langle !\theta\rangle\psi) \in v$, we have $\langle !\theta\rangle\top \in \lambda(v)$ and $\neg\langle !\theta\rangle\psi \in \lambda(v)$. Therefore, by axiom **R2**, $\langle !\theta\rangle\neg\psi \in \lambda(v)$. Hence $\neg\psi \in \lambda(v!\theta)$ and therefore $\psi \notin \lambda(v!\theta)$. By the induction hypothesis, $\mathcal{H}^{can}, v!\theta \not\models \psi$. This implies $\mathcal{H}^{can}, w!\theta \not\models [i]\psi$, as desired.

For the public announcement operator, assume that $\langle !\theta\rangle\psi \in \lambda(h)$. Since $\langle !\theta\rangle\top \in \lambda(h)$ (for $\neg\langle !\theta\rangle\top \in \lambda(h)$ makes $\lambda(h)$ inconsistent), $\psi \in \lambda(h!\theta)$. By the induction hypothesis, we have $\mathcal{H}^{can}, h!\theta \models \psi$, which implies $\mathcal{H}^{can}, h \models \langle !\theta\rangle\psi$. For the other direction, assume $\mathcal{H}^{can}, h \models \langle !\theta\rangle\psi$. Then, $\mathcal{H}^{can}, h!\theta \models \psi$. By the induction hypothesis, we have $\psi \in \lambda(h!\theta)$ and thus $\langle !\theta\rangle\psi \in \lambda(h)$. QED

All that remains is to show that canonical model $\mathcal{H}^{can}$ is in $\mathbb{F}(\mathbb{PAL})$.

Lemma 2.3.8 *$\mathcal{H}^{can}$ is in $\mathbb{F}(\mathbb{PAL})$. That is, there is an epistemic model $\mathcal{M}$ and state-dependent protocol $\mathsf{p} \in \mathbb{PAL}$ on $\mathcal{M}$ such that $\mathcal{H}^{can} = \mathsf{Forest}(\mathcal{M}, \mathsf{p})$.*

Proof. Let $\mathcal{M} = (W, \sim, V)$ and define $\mathsf{p}_{can} \in \mathbb{PAL}$ on $\mathcal{M}$ so that $\mathsf{p}_{can}(w) = \{\sigma \mid w\sigma \in H^{can}\}$. Suppose that $\mathcal{H}^{\mathsf{p}_{can}} = \mathsf{Forest}(\mathcal{M}, \mathsf{p}_{can})$. We claim that $\mathcal{H}^{can}$ and $\mathcal{H}^{\mathsf{p}_{can}}$ are the same model. For this, it suffices to show that for all $w \in W$ and $\sigma \in \mathsf{PAL}^*$ we have $w\sigma \in H^{can}$ iff $w\sigma \in W^{\sigma,\mathsf{p}_{can}}$ (cf. Definition 2.1.3). For this implies $H^{can} = H^{\mathsf{p}_{can}}$, where $H^{\mathsf{p}_{can}}$ is the domain of $\mathcal{H}^{\mathsf{p}_{can}}$. Then, by inspecting Definition 2.1.4 and Definition 2.3.6, we see that $\mathcal{H}^{can}$ and $\mathcal{H}^{\mathsf{p}_{can}}$ are the same model.

We show by induction on the length of $\sigma \in \mathsf{PAL}^*$ that for any $w \in W$, $w\sigma \in H^{can}$ iff $w\sigma \in W^{\sigma,\mathsf{p}_{can}}$. The base case ($\mathsf{len}(\sigma) = 0$) is clear. Assume that the claim holds for all σ with $\mathsf{len}(\sigma) = n$.

Given any $\sigma \in \mathsf{PAL}^*$ with $\mathsf{len}(\sigma) = n$, we first show by subinduction (on the structure of θ) that, for all $\theta \in \mathcal{L}_{el}$, $\mathcal{H}^{can}, w\sigma \models \theta$ iff $\mathcal{M}^{\sigma,\mathsf{p}_{can}}, w\sigma \models \theta$. The base and boolean cases are straightforward. Suppose that $\mathcal{H}^{can}, w\sigma \models [i]\gamma$. We must show $\mathcal{M}^{\sigma,\mathsf{p}_{can}}, w\sigma \models [i]\gamma$. Let $v\sigma \in W^{\sigma,\mathsf{p}_{can}}$ with $w\sigma \sim_i^{\sigma,\mathsf{p}_{can}} v\sigma$. By the main induction hypothesis, we have both $v\sigma \in H^{can}$ and $w\sigma \in W^{\sigma,\mathsf{p}_{can}}$. By Definition 2.1.3, since $w\sigma \sim_i^{\sigma,\mathsf{p}_{can}} v\sigma$, we have $w \sim_i^0 v$. Thus by Definition 2.3.6, $w\sigma \sim_i^{can} v\sigma$. Hence, $\mathcal{H}^{can}, v\sigma \models \gamma$. By the subinduction hypothesis, $\mathcal{M}^{\sigma,\mathsf{p}_{can}}, v\sigma \models \gamma$. Therefore, $\mathcal{M}^{\sigma,\mathsf{p}_{can}}, w\sigma \models [i]\gamma$. The other direction is similar.

Coming back to the main induction, assume $w\sigma!\theta \in H_{can}$. This implies that $\langle !\theta\rangle\top \in \lambda(w\sigma)$. By the Truth Lemma, we have $\mathcal{H}^{can}, w\sigma \models \langle !\theta\rangle\top$. This, together with axiom **A1**, implies $\mathcal{H}^{can}, w\sigma \models \theta$. From the above subinduction, it follows that $\mathcal{M}^{\sigma,\mathsf{p}_{can}}, w\sigma \models \theta$ (recall that $\theta \in \mathcal{L}_{EL}$ by definition). Thus, by the construction of

p_{can}, we have $w\sigma!\theta \in W^{\sigma!\theta,\mathsf{p}_{can}}$. This shows that if $w\sigma!\theta \in H^{can}$ then $w\sigma!\theta \in W^{\sigma!\theta,\mathsf{p}_{can}}$. The other direction is similar. This completes the proof. QED

The proof of the completeness theorem (Theorem 2.3.3) follows from Lemma 2.3.7 and Lemma 2.3.8 using a standard argument. The details are left to the reader.

2.3.3 Decidability via Finite Completeness Proof

We can modify the above proof to obtain a finite completeness proof. As a result, we can show that the satisfiability problem for TPAL is decidable. Our strategy is to construct a *finite* model from maximally consistent sets with respect to a suitable finite fragment of TPAL. In particular, we will associate what we call a *TPAL-closed set* with a given formula φ. The idea of the TPAL-closed set is based on the Fisher-Ladner closure in *Propositional Dynamic Logic* (PDL, [33, 32]) . Once the finite canonical model is constructed, the proof follows the idea of the full completeness proof from Section 2.3.2.

Definition 2.3.9 (TPAL-Closed Sets) Let X be a set of TPAL formulas. X is *TPAL-closed* if X satisfies the following closure conditions:

1. Closed under subformulas: If $\varphi \in X$ and ψ is a subformula of φ, then $\psi \in X$.

2. Closed under single negations: If $\varphi \in X$ and φ is of the form $\neg\psi$, then $\psi \in X$; and if $\varphi \in X$ and φ is not of the form $\neg\psi$, $\neg\varphi \in X$.

3. If $\langle !\theta\rangle\varphi \in X$, then $\langle !\theta\rangle\top \in X$.

4. If $\langle !\theta\rangle[i]\varphi \in X$, then $[i](\langle !\theta\rangle\top \to \langle !\theta\rangle\varphi) \in X$.

5. If $\varphi \in X$, then $\langle !\theta_1\rangle ... \langle !\theta_k\rangle\varphi \in X$ $(1 \leq k \leq d(X) - d(\varphi))$ where $\langle !\theta_i\rangle\top \in X$ for every $1 \leq i \leq k$. $\triangleleft$

Given a set $X \subseteq \mathcal{L}_{tpal}$, we denote by $(X)^{TPAL}$ the smallest expansion of X that is TPAL-closed. Note that provided that X is a finite set of formulas, $(X)^{TPAL}$ is also finite; also, $d(X) = d((X)^{TPAL})$. We denote by $(X)^{TPAL}_k$ with $0 \leq k \leq d(X)$ the set $\{\varphi \in (X)^{TPAL} \mid d(\varphi) \leq k\}$.

Here some remarks are in order about the closure conditions. The point of closure conditions is to make sure that formulas in closed sets can express 'enough' information about the truth of a formula of our interest in canonical models. In the above definition, the first two closure condition guarantees that closed sets contain enough source to say whether subformulas are true or false. By the third condition, closed sets can say whether public announcement in the sets can be made or not. By the fourth condition, closed sets contains enough source to express agents' future knowledge (cf Axiom **R4**). Finally the fifth condition guarantees that closed sets have enough *future* information to determine the truth of formulas in them.

Let Σ be a set of formulas in TPAL. We call a set $a \subseteq (\Sigma)^{TPAL}_k$ *an atom of depth k over Σ* ($0 \leq k \leq d(\Sigma)$), if a is TPAL-consistent and if $a \subset b \subseteq (\Sigma)^{TPAL}_k$, then b is inconsistent. We denote the set of the atoms of depth k over Σ as $At_k(\Sigma)$. Now it is easy to check the following properties of atoms.

Lemma 2.3.10 *Let Σ be a set of TPAL formulas. For every $a \in At_k(\Sigma)$, the following properties hold:*

1. *For all $\varphi \in (\Sigma)^{TPAL}_k$, $\varphi \in a$ or $\neg\varphi \in a$, but not both.*

2. *For all $\varphi \wedge \psi \in (\Sigma)^{TPAL}_k$, $\varphi \wedge \psi \in a$ iff $\varphi \in a$ and $\psi \in a$.*

3. *For all $\langle !\theta \rangle p \in (\Sigma)^{TPAL}_k$ with p a proposition letter, $\langle !\theta \rangle \varphi \in a$ iff $\langle !\theta \rangle \top \in a$ and $p \in a$.*

4. *For all $\langle !\theta \rangle \neg\varphi \in (\Sigma)^{TPAL}_k$, $\langle !\theta \rangle \neg\varphi \in a$ iff $\langle !\theta \rangle \top \in a$ and $\neg\langle !\theta \rangle \varphi \in a$.*

5. *For all* $\langle !\theta\rangle(\varphi \wedge \psi) \in (\Sigma)_k^{TPAL}$, $\langle !\theta\rangle(\varphi \wedge \psi) \in a$ *iff* $\langle !\theta\rangle\varphi \in a$ *and* $\langle !\theta\rangle\psi \in a$.

6. *For all* $\langle !\theta\rangle[i]\varphi \in (\Sigma)_k^{TPAL}$, $\langle !\theta\rangle[i]\varphi \in a$ *iff* $\langle !\theta\rangle\top \in a$ *and* $[i](\langle !\theta\rangle\top \to \langle !\theta\rangle\varphi) \in a$.

7. *For all* $\langle !\theta\rangle\varphi \in (\Sigma)_k^{TPAL}$, *if* $\langle !\theta\rangle\varphi \in a$, *then* $\langle !\theta\rangle\top \in a$.

8. *For all* $\langle !\theta\rangle\top \in (\Sigma)_k^{TPAL}$, *if* $\langle !\theta\rangle\top \in a$, *the* $!\theta \in a$.

Proof. Immediate from the definition of an atom and Definition 2.3.9. QED

Given a finite set Σ of TPAL-formulas, we construct a finite canonical model from the set $(\Sigma)^{TPAL}$. The construction follows exactly the construction from Section 2.3.2 (cf. Definition 2.3.4 and Definition 2.3.6). First, as in Definition 2.3.4 we construct maps λ_n^{fin} and sets H_n^{fin} $(0 \leq n \leq d(\Sigma))$ as follows:

- Let $H_0^{fin} = At_{d(\Sigma)}(\Sigma)$ and for each $a \in H_0^{fin}$, $\lambda_0^{fin}(a) = a$.
- Let $H_{n+1}^{fin} = \{\sigma!\theta \mid \sigma \in H_n^{fin} \text{ and } \langle !\theta\rangle\top \in \lambda_n^{fin}(\sigma)\}$. For every $\sigma = \sigma'!\theta \in H_{n+1}^{fin}$, define $\lambda_{n+1}^{fin}(\sigma) = \{\psi \mid \langle !\theta\rangle\psi \in \lambda_n^{fin}(\sigma')\}$.

Proposition 2.3.11 *For all* n, $\lambda_n^{fin}(\sigma) \in At_{d(\Sigma)-n}(\Sigma)$.

Proof. The proof is by induction on n. The base case is clear. For the inductive step, the argument is completely analogous to the proof of Lemma 2.3.5, given Lemma 2.3.10 and Definition 2.3.9. QED

We now define a finite canonical model $\mathcal{H}^{fin}$. This goes exactly like Definition 2.3.6 except for the domain, which is now $H^f in = \bigcup_{0 \leq i \leq d(\Sigma)} H_i^{fin}$. As in Section 2.3.2, we write $\lambda^{fin}(h)$ from $\lambda_n^{fin}(h)$ where n is the number of announcements in h. We use $\sim_i^{fin}$ and V^{fin} to denote the canonical relations and valuations in $\mathcal{H}^{fin}$, just as in Definition 2.3.6. All that remains to be proved are analogues of Lemma 2.3.7 and Lemma 2.3.8.

Lemma 2.3.12 (Finite Truth Lemma) *Let* $\varphi \in (\Sigma)^{TPAL}$*. For every history* h *in* H^{fin} *such that* $\mathsf{len}(h) \leq d(\Sigma) - d(\varphi) + 1$,

$$\varphi \in \lambda^{fin}(h) \text{ iff } \mathcal{H}^{fin}, h \models \varphi.$$

Proof. The proof is by induction on φ. Given Lemma 2.3.10 and the closure conditions in Definition 2.3.9, the proof is similar to the proof of Lemma 2.3.7. We only present the public announcement modality case. Readers are invited to verify that the argument holds for the other cases as well. In particular, note that the formulas used in the proof of Lemma 2.3.7 are in fact in the set $\lambda^{fin}(h)$.

Let φ be $\langle !\theta\rangle\psi$. First, assume that $\langle !\theta\rangle\psi \in \lambda^{fin}(h)$, where $\mathsf{len}(h) \leq d(\Sigma)-d(\varphi)+1$. Since $\langle !\theta\rangle\psi \in \lambda^{fin}(h)$ by Lemma 2.3.9, $\langle !\theta\rangle\top \in \lambda^{fin}(h)$. Thus, $h!\theta \in H^{fin}$ and $\psi \in \lambda^{fin}(h!\theta)$. Here note that $\mathsf{len}(h!\theta) = \mathsf{len}(h)+1 \leq d(\Sigma)-(d(\varphi)-1)+1 = d(\Sigma)-d(\psi)+1$. Thus, by induction, we have $\mathcal{H}^{fin}, h!\theta \models \psi$, which implies $\mathcal{H}^{fin}, h \models \langle !\theta\rangle\psi$. For the other direction, assume that $\mathcal{H}^{fin}, h \models \langle !\theta\rangle\psi$. This implies $\mathcal{H}^{fin}, h!\theta \models \psi$ with $\mathsf{len}(h!\theta) \leq d(\Sigma) - d(\psi) + 1$. By induction, $\psi \in \lambda^{fin}(h!\theta)$. By the construction of the canonical model, $\langle !\theta\rangle\psi \in \lambda^{fin}(h)$ as desired. QED

Lemma 2.3.13 $\mathcal{H}^{fin}$ *is an ETL model generated from an epistemic model and a PAL-protocol.*

Proof. The proof is similar to that of Lemma 2.3.8. QED

Putting everything together, it is not difficult to verify that:

Theorem 2.3.14 (Decidability of TPAL) *The satisfiability problem for the logic TPAL is decidable.*

2.3.4 Common Knowledge

So far, we have only considered the knowledge modality $[i]$ to describe agents' informational sates. However, other interesting informational states arise in multi-agent contexts. One such state is *common knowledge*. The notion was first studied by D. Lewis in [48] and formalized in [4]. Common knowledge has been one of the key epistemic notion in the literature.

We now describe how to incorporate the common knowledge operator into our axiomatic system. Our strategy has two components. First, as we did in the above completeness proof, we will make use of compositional analysis via reduction axioms in PAL. [71] provides the reduction axiom for *relativized common knowledge* in the context of PAL. We will modify the reduction axiom for TPAL and add it to TPAL together with other standard axioms associated with relativized common knowledge. Second, we will appeal to the finite completeness argument developed in the previous subsection. We will extend the closure condition in Definition 2.3.9 and show that it is enough to carry out the completeness argument.

First let us define common knowledge operators. Given a binary relation X, denote by X^+ the *transitive closure* of X, i.e. the smallest set containing X such that, if $(w,v),(v,u) \in X^+$, then (w,u). Let G be a set of agents in $\mathcal{A}$, i.e. $G \subseteq \mathcal{A}$. Given an ETL model $\mathcal{H} = (H, \sim, V)$, define $\sim_G := (\bigcup_{i \in G} \sim_i)^+$. The operator C_G, where $C_G\varphi$ reads as "φ is common knowledge among G", is defined by:

$$\mathcal{H}, h \models C_G\varphi \text{ iff for each } h' \in H, \text{ if } h \sim_G h' \text{ then } h' \models \varphi$$

Van Benthem, van Eijk and Kooi ([71]) discuss the technical issues that arise when axiomatizing Public Announcement Logic in languages with common knowledge. They introduce a new "relativized common knowledge" operator $C_G(\psi|\varphi)$ saying that every ψ-path (a path in which each step leads to a ψ-world) along the relation

$\sim_G$ ends in a state satisfying φ. More formally, let $[\![\varphi]\!]$ be the set of histories satisfying φ. Given a DEL-generated ETL model $\mathcal{H} = (H, \sim, V)$

$$\mathcal{H}, h \models C_G(\psi|\varphi) \text{ iff } \forall h' \in H,\ (h, h') \in (\bigcup_{i \in G} \sim_i\ \cap\ (H \times [\![\psi]\!]))^+ \text{ implies } \mathcal{H}, h' \models \varphi$$

The usual common knowledge operator $C_G\varphi$ can be defined as $C_G(\top|\varphi)$.

We denote by TPAL^C the extension of TPAL with the relativized common knowledge operator. We now provide the axiomatization TPAL^C of the extension. For convenience, we denote $\bigwedge_{i \in G}[i]\varphi$ by $E_G\varphi$ ("everybody in G knows φ").

Definition 2.3.15 (Axiomatization of TPAL^C) The axiomatization TPAL^C extends TPAL by the following axioms and the inference rule:

Axioms

CK $C_G(\varphi|\psi \to \chi) \to (C_G(\varphi|\psi) \to C_G(\varphi|\chi))$

C1 $C_G(\varphi|\psi) \leftrightarrow E_G(\varphi \to (\psi \wedge C_G(\varphi|\psi)))$

C2 $(E_G(\varphi \to \psi) \wedge C_G(\varphi|\psi \to E_G(\varphi \to \psi))) \to C_G(\varphi|\psi)$

R5 $\langle !\theta \rangle C(\psi|\varphi) \leftrightarrow \langle !\theta \rangle \top \wedge C(\langle !\theta \rangle \psi | \langle !\theta \rangle \varphi)$

Inference Rule

CN If $\vdash \varphi$, then $\vdash C_G(\psi|\varphi)$. $\lhd$

[71] provides the completeness proof for the extension of EL with the relativized common knowledge operator by **CK**, **C1-2** and **CN**. Then it reduces the extension of PAL with the operator by the following reduction axiom:

$$\langle !\theta \rangle C(\psi|\varphi) \leftrightarrow \theta \wedge C_G(\langle !\theta \rangle \psi | \langle !\theta \rangle \varphi).$$

The difference between this axiom and our **R5** is that θ is replaced by $\langle !\theta \rangle \top$ in **R5**. This is the maneuver that we appealed to when we axiomatized TPAL above in

Definition 2.3.1. Given this, it is not hard to see that the above axiom is valid on the class $\mathbb{F}(\mathbb{PAL})$.

Now the idea of the completeness proof is based on the finite completeness argument in Section 2.3.3. Thus, we will take a closure of a finite set X of our interest and construct a finite canonical model from maximally consistent sets in the closed set. With the addition of the relativized common knowledge operator, we need to add the following closure condition to the definition of TPAL-closed sets above (Definition 2.3.9):

- If $C_G(\psi|\varphi) \in X$, then $[i](\psi \to (\varphi \wedge C_G(\psi, \varphi))) \in X$ for all $i \in G$.

This condition is the same as the closure condition used in [71] to give the finite completeness argument for EL with the common knowledge operator.

With a TPAL-closure of a finite set, the canonical model is constructed in the way presented in Section 2.3.3. To see how the proof will go, it is helpful to inspect our completeness proof in TPAL given above. Consider the argument given in the proof of Lemma 2.3.7. To prove the knowledge modality case, our strategy was to go down along the history to the bottom level by appealing to **R4** and give the standard completeness argument inside the bottom epistemic model $\mathcal{H}_0^{can}$. In fact, the left-to-right direction of the argument can be characterized by the following three steps:

1. Assume $[i]\psi \in \lambda(w!\theta_1 \ldots !\theta_n)$.

2. By successive applications of **R4**, obtain

$$[i](\langle !\theta_1 \rangle \top \to \langle !\theta_1 \rangle (\ldots (\langle !\theta_n \rangle \top \to \langle !\theta_n \rangle \psi) \ldots) \in \lambda(w).$$

3. By the standard epistemic canonical model reasoning in $\mathcal{H}_0^{can}$, obtain

$$[i](\langle !\theta_1\rangle\top \to \langle !\theta_1\rangle(\dots(\langle !\theta_n\rangle\top \to \langle !\theta_n\rangle\psi)\dots) \in \lambda(v).$$

4. Obtain $\psi \in \lambda(v!\theta_1 \dots !\theta_n)$ by construction and conclude by IH that $\mathcal{H}^{can}, v!\theta_1 \dots !\theta_n \models \psi$.

The reasoning from 2 to 3 only requires the argument given in the completeness argument in the canonical model of epistemic logic. Here in particular, 3 is obtained from 2 based on the fact that the indistinguishability relation between w and v is established by the canonical model construction, $w \sim_i^{can} v$ in $\mathcal{H}_0^{can}$ iff $\{\varphi \mid [i]\varphi \in w\} \subseteq v$. From 1 to 2 and from 3 to 4, we need considerations special to TPAL and appeal to **R4** and the construction of the canonical model. This point applies to the right-to-left direction of the proof. (the construction of v_0 is completely analogous to what is done in the canonical model of epistemic logic).

Our proof of the completeness of TPAL^C below is based on this idea. Having **R5**, we can 'go down' the tree to the bottom and do the standard completeness argument at the bottom by using **CK**, **C1-3** and **CN**. After doing so, we 'come' up the tree back and apply the inductive hypothesis. In the following proof, we will not repeat the completeness argument at the bottom for relativized common knowledge given in [71].

Lemma 2.3.16 *Let $\varphi \in (\Sigma)^{TPAL}$. For every history h in H^{fin} such that $\mathsf{len}(h) \leq d(\Sigma) - d(\varphi) + 1$,*

$$\varphi \in \lambda^{fin}(h) \textit{ iff } \mathcal{H}^{fin}, h \models \varphi.$$

Proof. The proof is induction on the complexity of φ. We only do the case for relativized common knowledge. The other cases are done by the completeness arguments in Lemma 2.3.7 and 2.3.12. Assume that $C_G(\psi|\chi) \in \lambda^{fin}(w!\theta_1 \dots !\theta_n)$ with

$h = w!\theta_1 \ldots !\theta_n$. Let us write $!\theta_1 \ldots !\theta_n = !\vec{\theta}$. By repeated applications of **R5**, we obtain

$$C_G(\langle !\vec{\theta}\rangle\psi | \langle !\vec{\theta}\rangle\chi) \in \lambda(w).$$

where $\langle !\theta_1\rangle \ldots \langle !\theta_n\rangle$ is denoted by $\langle !\vec{\theta}\rangle$. From this, by the standard argument for the relativized common knowledge operator (see [71]), we can show that, for any v in a path from w along nodes where $\langle !\vec{\theta}\rangle\psi$ is true, $\langle !\vec{\theta}\rangle\chi \in \lambda(v)$ (and $C_G(\langle !\vec{\theta}\rangle\psi, \langle !\vec{\theta}\rangle\chi) \in \lambda(v)$). By construction, we have $\chi \in \lambda(v!\vec{\theta})$. By IH, $\mathcal{H}^{can}, h \models \chi$. On the other hand, by definition, $\mathcal{H}^{can}, v \models \langle !\vec{\theta}\rangle\psi$ iff $\mathcal{H}^{can}, v!\vec{\theta} \models \psi$. Therefore, we can say, for any $v!\vec{\theta}$ in a path from $w!\vec{\theta}$ along nodes where ψ is true, $\chi \in \lambda(v!\vec{\theta})$. Therefore, by inductive hypothesis, we are done.

For the other direction, assume that $\mathcal{H}^{can}, w!\vec{\theta} \models C_G(\psi|\chi)$. This implies $\mathcal{H}^{can}, w \models C_G(\langle !\vec{\theta}\rangle\psi | \langle !\vec{\theta}\rangle\chi)$. By the standard argument given in [71], we can show $C_G(\langle !\vec{\theta}\rangle\psi | \langle !\vec{\theta}\rangle\chi)) \in \lambda(w)$. Now, since $w!\vec{\theta}$ in $\mathcal{H}^{can}$ implies

$$\langle \theta_1\rangle\top \in \lambda(w), \ldots, \langle \theta_n\rangle\top \in \lambda(w!\theta_1 \ldots \theta_{n-1}),$$

we can apply **R5** successively and obtain $C_G(\psi|\chi) \in \lambda(w\vec{\theta})$. QED

We can make sure that the canonical model is in the right class of models as in Lemma 2.3.13. Therefore, we have the following result:

Theorem 2.3.17 TPAL^C *is sound and (weakly) complete with respect to* $\mathbb{F}_{sd}$. *Moreover, the satisfiability problem of* $TPAL^C$ *is decidable.*

2.4 Other Results in TPAL

We will now prove other important results in TPAL. First, we will axiomatize the class of ETL models generated from uniform PAL-protocols. Second, we will show that PAL can be faithfully embedded into TPAL.

2.4.1 Uniform Protocols

First we will axiomatize the class $\mathbb{F}(Ptcl(\mathsf{PAL}))$ of ETL models generated from uniform PAL-protocols. For this, we extend the language $\mathcal{L}_{tpal}$ with an existential modality. Let $E\varphi$ mean that "φ is true at some history with the same sequence of announcements". (cf. Chapter 1.6) We define this as follows. Let $\mathcal{H}$ be an ETL model generated by an epistemic model $\mathcal{M} = (W, \sim, V)$ and a (state-dependent or uniform) PAL-protocol. Let $w \in W$ and σ a sequence of announcements with $w\sigma$ in $\mathcal{H}$. Then we interpret the existential modality as follows:

$$\mathcal{H}, w\sigma \models E\varphi \;\text{ iff }\; \exists v \in W \;\text{ such that } v\sigma \text{ is in } \mathcal{H} \text{ and } \mathcal{H}, v\sigma \models \varphi.$$

This operator functions as an existential modality at each 'stage' of successive public announcements. The dual U of E is a universal modality in the same sense. We consider the extension TPAL^E of TPAL.

First let us remark that the introduction of this operator keeps the system of TPAL manageable. A complete axiomatization can be given in a similar way by adding the following axioms to TPAL as in Definition 2.3.1:

E1 $E(\varphi \to \psi) \to (E\varphi \to E\psi)$

E2 $\varphi \to E\varphi$

E3 $\varphi \to UE\varphi$

E4 $EE\varphi \to E\varphi$

E5 $U\varphi \to [i]\varphi$

R5 $\langle !\theta \rangle E\varphi \leftrightarrow \langle !\theta \rangle \top \wedge E\langle !\theta \rangle \varphi.$

R5 allows us to obtain the results corresponding to Lemma 2.3.7 and Lemma 2.3.8 with respect to uniform protocols. Axioms **E1-5** are the standard axiomatization of the existential modality. We denote the resulting axiomatization by TPAL^E

We now would like to axiomatize the class

$$\mathbb{F}(Ptcl(\mathsf{PAL})) = \{\mathsf{Forest}(\mathcal{M}, \mathsf{P}) \mid \mathcal{M} \text{ an epistemic model and } \mathsf{P} \in Ptcl(\mathsf{PAL})\}$$

For this, we extend the axiomatization TPAL^E with the following axiom:

Uni $\langle !\theta\rangle\top \to U(\theta \to \langle !\theta\rangle\top)$.

This axiom *characterizes* uniform protocols in the following sense. Let us say a state-dependent protocol $\mathsf{p} \in \mathbb{PAL}$ on a given model $\mathcal{M}$ *generates a uniform ETL model* if $\mathsf{Forest}(\mathcal{M}, \mathsf{p}) = \mathsf{Forest}(\mathcal{M}, \mathsf{P})$ for some $\mathsf{P} \in Ptcl(\mathsf{PAL})$.

Proposition 2.4.1 *The axiom* **Uni** *is valid on a frame* $\mathsf{Forest}(\mathcal{M}, \mathsf{p})$ *iff* p *generates a uniform ETL model.*

Proof. ($\Leftarrow$) Assume that p generates a uniform ETL model $\mathcal{H} = Forest(\mathcal{M}, \mathsf{p})$. Then there is some uniform protocol $\mathsf{P} \in Ptcl(\mathsf{PAL})$ such that $\mathcal{H} = \mathsf{Forest}(\mathcal{M}, \mathsf{P})$. Now suppose that $w \in Dom(\mathcal{M})$ and $\sigma \in \mathsf{PAL}^*$. Assume that $\mathcal{H}, w\sigma \models \langle !\theta\rangle\top$. Then, we have $w\sigma!\theta$ in $\mathcal{H}$. This means that $\sigma!\theta \in \mathsf{p}(w)$. Since p is uniform, there is some $\mathsf{P} \in Ptcl(\mathsf{PAL})$ such that $\mathcal{H} = \mathsf{Forest}(\mathcal{M}, \mathsf{P})$. Therefore $\sigma!\theta \in \mathsf{P}$. Now, let v be an arbitrary state in $\mathcal{M}$. If $\mathcal{H}, v\sigma \models \theta$, then, since $\sigma!\theta \in \mathsf{P}$, we have $v\sigma!\theta \in D(\mathcal{H})$. Hence $\mathcal{H}, v\sigma \models \langle !\theta\rangle\top$. Since v was arbitrary, we have $\mathcal{H}, w\sigma \models U(\theta \to \langle !\theta\rangle\top)$.

($\Rightarrow$) Assume that **Uni** is valid on an ETL model $\mathcal{H}^{\mathsf{p}} = \mathsf{Forest}(\mathcal{M}, \mathsf{p})$. Construct a protocol $\mathsf{P} = \{\sigma \mid w\sigma \text{ is in } \mathcal{H}^{\mathsf{p}} \text{ for some } w \in Dom(\mathcal{M})\}$. Clearly, P is closed under prefixes, so is in fact a protocol. We need to show that $\mathcal{H}^{\mathsf{p}} = \mathsf{Forest}(\mathcal{M}, \mathsf{P})$. For this, it suffices to show that, for all σ, $\mathcal{M}^{\sigma,\mathsf{p}} = \mathcal{M}^{\sigma,\mathsf{P}}$, equivalently (via definition) $W^{\sigma,\mathsf{p}} = W^{\sigma,\mathsf{P}}$. We prove this by induction on σ. First, the left-to-right inclusion is clear by the construction of P. For the other direction, the base case is clear. For if σ is the empty sequence, the inclusion clearly holds as $W^{\sigma,\mathsf{p}} = W^{\sigma,\mathsf{P}} = D(\mathcal{M})$. For the inductive step, assume that $w\sigma!\theta \in W^{\sigma!\theta,\mathsf{P}}$. Then we have $\mathcal{M}^{\sigma,\mathsf{P}}, w\sigma \models \theta$. By the induction hypothesis, we have $\mathcal{M}^{\sigma,\mathsf{p}}, w\sigma \models \theta$. Since $\theta \in \mathcal{L}_{el}$, it follows from Observation 2.2.3 that $\mathcal{H}^{\mathsf{p}}, w\sigma \models \theta$. Note that by the construction of P, there must be some $v \in Dom(\mathcal{M})$ such that $v\sigma!\theta \in W^{\sigma,\mathsf{p}}$. This implies that $\mathcal{H}^{\mathsf{p}}, v\sigma \models \langle!\theta\rangle\top$. Here, since Uni is valid in $\mathcal{H}^{\mathsf{p}}$, we have $\mathcal{H}^{\mathsf{p}}, v\sigma \models U(\theta \to \langle!\theta\rangle\top)$. Thus, it follows that $\mathcal{H}^{\mathsf{p}}, w\sigma \models !\theta \to \langle!\theta\rangle\top$. From the fact that $\mathcal{H}^{\mathsf{p}}, w\sigma \models \theta$, we then have $\mathcal{H}^{\mathsf{p}}, w\sigma \models \langle!\theta\rangle\top$, which is equivalent to $w\sigma!\theta$ in $\mathcal{H}^{\mathsf{p}}$, i.e., $w\sigma!\theta \in W^{\sigma!\theta,\mathsf{p}}$, as desired. QED

Let TPAL^{Uni} be the extension of TPAL^{E} with the axiom **Uni**. The following is an immediate consequence of a suitable truth lemma analogous to Lemma 2.3.7 and the above proposition:

Corollary 2.4.2 TPAL^{Uni} *is sound and strongly complete with respect to the class* $\mathbb{F}(Ptcl(\mathsf{PAL}))$.

Proof. The proof is similar to the one given in Section 2.3.2 (making use of the above proposition to show that the canonical model is generated by a uniform protocol). QED

2.4.2 Embedding PAL into TPAL

The introduction of the operator E allows us to obtain another interesting result. The relation between the original public announcement logic PAL and our new TPAL is not completely straightforward. Clearly all principles of TPAL are valid in PAL. Indeed, the inclusion seems proper, as standard public announcement logic is about special "full" protocols. But is it really *stronger* than TPAL? Using the existential modality of the previous section, we can answer this question almost in the negative by providing an effective semantic translation from PAL into TPAL^{E}

Given a formula φ, let $Ptcl(\varphi)$ be the set of formulas of the form:

$$U(\theta_1 \to \langle !\theta_1 \rangle(\theta_2 \to \langle !\theta_2 \rangle(\cdots \langle !\theta_{k-1} \rangle(\theta_k \to \langle \theta_k \rangle \top)\cdots)))$$

where $!\theta_i \in AOC(\varphi)$ $(1 \leq i \leq k)$ and $1 \leq k \leq d(\varphi)$.

The formulas in $Ptcl(\varphi)$ state that the public announcements that are relevant to the truth value of φ are all announceable at any node of a given ETL model.

Theorem 2.4.3 *For any formula $\varphi \in \mathcal{L}_{tpal}$,*

$$\models \varphi \text{ in PAL} \qquad \text{iff} \qquad \models \bigwedge Ptcl(\varphi) \to \varphi \text{ in TPAL.}$$

Proof. ($\Leftarrow$) Suppose $\models \bigwedge Ptcl(\varphi) \to \varphi$ in TPAL. Then, for all epistemic models $\mathcal{M}$ and all $w \in Dom(\mathcal{M})$, we have $\mathsf{Forest}(\mathcal{M}, \mathsf{PAL}^*), w \models \bigwedge Ptcl(\varphi) \to \varphi$, where PAL^* is the class of all finite sequences of public announcements. By Proposition 2.2.1, $\mathcal{M}, w \models \bigwedge Ptcl(\varphi) \to \varphi$ in PAL. Now, by Proposition 2.2.2, $Ptcl(\varphi)$ is valid in PAL. Hence, $\mathcal{M}, w \models \varphi$. Since $\mathcal{M}$ and w were arbitrary, we have $\models \varphi$ in PAL.

($\Rightarrow$) Suppose $\models \varphi$ in PAL. Let $\mathsf{Forest}(\mathcal{M}, p)$ be an arbitrary PAL-generated ETL model. Fix h in $\mathsf{Forest}(\mathcal{M}, p)$ and assume $\mathsf{Forest}(\mathcal{M}, p), h \models \bigwedge Ptcl(\varphi)$. Note that $h = w\sigma$ where $w \in Dom(\mathcal{M})$ and σ a sequence of formulas in PAL^*. Now consider the epistemic model $\mathcal{M}^{\sigma,\mathsf{p}}$. Since φ is valid in PAL, we have $\mathcal{M}^{\sigma,\mathsf{p}}, w\sigma \models \varphi$. By Observation 2.2.8, $\mathsf{Forest}(\mathcal{M}^{\sigma,\mathsf{p}}, \mathsf{p}_\varphi), w\sigma \models \varphi$. We now show that $\mathsf{Forest}(\mathcal{M}, \mathsf{p})$ contains the model $\mathsf{Forest}(\mathcal{M}^{\sigma,\mathsf{p}}, \mathsf{p}_\varphi)$.

Claim If h' is in $\mathsf{Forest}(\mathcal{M}^{\sigma,\mathsf{p}}, \mathsf{p}_\varphi)$, then h' is in $\mathsf{Forest}(\mathcal{M}, \mathsf{p})$.

Proof of Claim. We prove this claim by induction on the length of h' ($\mathsf{len}(h) \leq \mathsf{len}(h') \leq \mathsf{len}(h) + d(\varphi)$). For the base case, assume that $\mathsf{len}(h) = \mathsf{len}(h')$. If h' is in $Dom(\mathsf{Forest}(\mathcal{M}^{\sigma,\mathsf{p}}, \mathsf{p}_\varphi))$, then $h' \in Dom(\mathcal{M}^{\sigma,\mathsf{p}})$. Thus, h' in $\mathsf{Forest}(\mathcal{M}, \mathsf{p})$. For the inductive step, assume that h' is in $\mathsf{Forest}(\mathcal{M}^{\sigma,\mathsf{p}}, \mathsf{p}_\varphi)$. Then we have $h' = v\sigma!\theta_1...!\theta_n$ for $!\theta_i \in AOC(\varphi)$ $(1 \leq i \leq n)$ and $v \in Dom(\mathcal{M})$. Here, our assumption that $\mathsf{Forest}(\mathcal{M}, \mathsf{p}), w\sigma \models \bigwedge Ptcl(\varphi)$ implies

$$\mathsf{Forest}(\mathcal{M}, \mathsf{p}), w\sigma \models U(!\theta_1 \to \langle !\theta_1 \rangle (\cdots (!\theta_n \to \langle !\theta_n \rangle \top) \cdots))$$

and so,

$$\mathsf{Forest}(\mathcal{M}, \mathsf{p}), v\sigma \models !\theta_1 \to \langle !\theta_1 \rangle (\cdots (!\theta_n \to \langle !\theta_n \rangle \top) \cdots).$$

Also, we have assumed that h' is in $\mathsf{Forest}(\mathcal{M}^{\sigma,\mathsf{p}}, \mathsf{p}_\varphi)$, whose construction implies that $\mathcal{M}^{\sigma,\mathsf{p}}, v\sigma \models \theta_1$, ..., $\mathcal{M}^{\sigma!\theta_1...!\theta_{n-1},\mathsf{p}}, v\sigma!\theta_1...!\theta_{n-1} \models \theta_n$. Now by the induction hypothesis, $v\sigma$, $v\sigma!\theta_1$, ..., $v\sigma!\theta_1...!\theta_{n-1}$ are all in $\mathsf{Forest}(\mathcal{M}, \mathsf{p})$. This, together with Observation 2.2.3, implies that $\mathsf{Forest}(\mathcal{M}, \mathsf{p}), v\sigma \models \theta_1$, ..., $\mathsf{Forest}(\mathcal{M}, \mathsf{p}), v\sigma!\theta_1...!\theta_{n-1} \models \theta_n$ (since θ_1, ..., θ_n are all formulas in $\mathcal{L}_{el}$ as parts of protocols).

Thus, we have

$$\mathsf{Forest}(\mathcal{M}, \mathsf{p}), v\sigma \models \langle !\theta_1 \rangle (\theta_2 \to \langle !\theta_2 \rangle (\cdots (\theta_n \to \langle !\theta_n \rangle \top) \cdots)$$

$$\mathsf{Forest}(\mathcal{M}, \mathsf{p}), v\sigma!\theta_1 \models \langle !\theta_2 \rangle (\theta_3 \to \langle !\theta_3 \rangle (\cdots (!\theta_n \to \langle !\theta_n \rangle \top) \cdots)$$

$$\vdots$$

$$\mathsf{Forest}(\mathcal{M}, \mathsf{p}), v\sigma!\theta_1 \cdots !\theta_{n-1} \models \theta_n \to \langle !\theta_n \rangle \top.$$

$$\mathsf{Forest}(\mathcal{M}, \mathsf{p}), v\sigma!\theta_1 \cdots !\theta_{n-1} \models \langle !\theta_n \rangle \top.$$

Therefore, $h' = v\sigma!\theta_1...!\theta_n$ is in $\mathsf{Forest}(\mathcal{M}, \mathsf{p})$. QED (of Claim)

Now, by the preceding claim, $\mathsf{Forest}(\mathcal{M}, \mathsf{p})$ includes $\mathsf{Forest}(\mathcal{M}^{\sigma,\mathsf{p}}, \mathsf{p}_\varphi)$. Since we had $\mathsf{Forest}(\mathcal{M}^{\sigma,\mathsf{p}}, \mathsf{p}_\varphi), w\sigma \models \varphi$ as above, it follows from Observations 2.2.5 and 2.2.7 that

$\mathsf{Forest}(\mathcal{M}, \mathsf{p}) \models \varphi$. (Note that φ is in $\mathcal{L}_{tpal}$.) This completes the proof. QED

We do not know if we can do this reduction without the existential modality. Also, we have not solved the opposite question, whether TPAL can be faithfully embedded into PAL, though we think the answer is negative.

2.5 Temporal Dynamic Epistemic Logic

Now we will extend the logic of TPAL to the full class of DEL-generated ETL models $\mathbb{F}_{sd}$. We will call the resulted logical system, *Temporal Dynamic Epistemic Logic* (TDEL). Indeed many of the techniques in TPAL can be generalized and similar results can be obtained for TDEL. For illustration, we will first look at the axiomatization of TDEL in some details.

2.5.1 Axiomatization of TDEL

First we introduce the system of TDEL.

Definition 2.5.1 (Language of TDEL) Let $\mathbb{E}$ be the class of pointed event models. Formulas of TDEL is inductively defined as follows:

$$\varphi \ ::= \top \mid p \mid \neg\varphi \mid \varphi \wedge \varphi \mid [i]\varphi \mid [\epsilon]\varphi$$

where $p \in \mathsf{At}$, $i \in \mathcal{A}$ and $\epsilon \in \mathbb{E}$. The duals, $\langle i \rangle$ and $\langle !\varphi \rangle$, of $[i]$ and $[\epsilon]$, and the other boolean operators are defined in the standard way. We denote the set of formulas in TDEL by $\mathcal{L}_{tdel}$. ◁

Definition 2.5.2 (Truth) Let $\mathcal{H} \in \mathbb{F}_{sd}$ be a DEL-generated ETL model with

$$\mathcal{H} = \mathsf{Forest}(\mathcal{M}, \mathsf{p}) = (H, \sim, V).$$

The truth definition of the event model operator $\langle \epsilon \rangle$ is defined by:

- $\mathcal{H}, h \models \langle \epsilon \rangle \varphi$ iff $h\epsilon \in H$ and $\mathcal{H}, h\epsilon \models \varphi$.

The other operators are defined in the standard way as in Definition 2.1.5. ◁

Definition 2.5.3 (Axiomatization) The axiomatization TDEL of TDEL extends the axiomatization of EL (**PC**, i**K**, i**N**, **MP** in Definition 2.3.1) with the following axiom schemes and rule.[4]

Axioms

ϵ**K** $[\epsilon](\varphi \to \psi) \to ([\epsilon]\varphi \to [\epsilon]\psi)$

F1 $\langle \epsilon \rangle p \leftrightarrow \langle \epsilon \rangle \top \wedge p$

F2 $\langle \epsilon \rangle \neg \varphi \leftrightarrow \langle \epsilon \rangle \top \wedge \neg \langle \epsilon \rangle \varphi$

F3 $\langle \epsilon \rangle (\varphi \wedge \psi) \leftrightarrow \langle \epsilon \rangle \varphi \wedge \langle \epsilon \rangle \psi$

F4 $\langle \epsilon \rangle [i] \varphi \leftrightarrow \langle \epsilon \rangle \top \wedge \bigwedge_{\{e \in Dom(\epsilon^L) \mid (\epsilon^R, e) \in \to_{\epsilon^L}(i)\}} [i](\langle \epsilon^L, e \rangle \top \to \langle \epsilon^L, e \rangle \varphi)$

E1 $\langle \epsilon \rangle \top \to \mathsf{pre}_{\epsilon^L}(\epsilon^R)$

Inference Rule

ϵ**N** If $\vdash \varphi$, then $\vdash [\epsilon]\varphi$ with $\epsilon \in \mathbb{E}$. ◁

Here what was said about reduction axioms in PAL and axioms in TPAL (Section 2.3) applies to **F1-4** here. As PAL, DEL reduces to EL via compositional analysis

[4]Below **F3** follows from ϵ**K** and ϵ**N**. Nonetheless we put the axiom here to make the contrast explicit between TPAL and TDEL.

based on reduction axioms. For instance, the reduction axiom for the knowledge operator $[i]$ is:

$$\langle\epsilon\rangle[i]\varphi \leftrightarrow \mathsf{pre}_{\epsilon^L}(\epsilon^R) \wedge \bigwedge_{\{e\in Dom(\epsilon^L) \mid (\epsilon^R,e)\in Rel(\epsilon^L)(i)\}} [i](\langle\epsilon^L, e\rangle\top \rightarrow \langle\epsilon^L, e\rangle\varphi)$$

The difference from **F4** is in the right-hand side of the biconditional: $\langle\epsilon\rangle\top$ ("the event ϵ can happen") is replaced by $\mathsf{pre}_{\epsilon^L}(\epsilon^R)$ (the precondition of ϵ^R). The DEL-reduction axioms are valid as they are, since DEL assumes that if the precondition of a given event is satisfied, then the event can always happen. However TDEL lift this assumption. In TDEL, even if the precondition of a given event is satisfied, the event cannot always happen, unless it is 'permitted' by protocols. This is why we have **E1**, but not its converse. Readers are invited to verify that these are sound with respect to $\mathbb{F}_{sd}$.

2.5.2 Completeness Proof

The completeness proof can be given based on the methods used for the completeness of TPAL. To take care of the full class of event models, we have only to generalize the construction of canonical model (Definition 2.5.6) and prove an additional lemma (Proposition 2.5.7) to make sure that the argument for the truth lemma goes through in TDEL.

Definition 2.5.4 (Legal Histories) Let W_0 be the set of all TDEL-maximal consistent sets. We define λ_n and H_n as follows:

- Define $H_0 = W_0$ and for each $w \in H_0$, $\lambda_0(w) = w$.
- Let $H_{n+1} = \{h\epsilon \mid h \in H_n \text{ and } \langle\epsilon\rangle\top \in \lambda_n(h)\}$. For each $h = h'\epsilon \in H_{n+1}$, define $\lambda_{n+1}(h) = \{\varphi \mid \langle\epsilon\rangle\varphi \in \lambda_n(h')\}$.

Given $h \in H_n$, we write $\lambda(h)$ for $\lambda_n(h)$. ◁

Lemma 2.5.5 *For each $n \geq 0$, for each $\sigma \in H_n$, $\lambda_n(\sigma)$ is a maximally consistent set.* ◁

Proof. The same argument can be applied as in Lemma 2.3.5. QED

Let $\mathcal{H}^{can}_0 = (H_0, \sim^0, V^0)$, where $\sim^0$ and V^0 are defined by

- $w \sim^0_i v$ iff $\{\varphi \mid [i]\varphi \in w\} \subseteq v$.
- For each $p \in \mathsf{At}$ and $w \in H_0$, $p \in V(w)$ iff $p \in w$.

Definition 2.5.6 (Canonical Model) The canonical model $\mathcal{H}^{can}$ is a triple $(H^{can}, \sim^{can}, V^{can})$, where each item is defined as follows:

- $H^{can} := \bigcup_{i=0}^{\infty} H_i$.
- For each $w\sigma, w'\sigma' \in H^{can}$, $w\sigma \sim^{can}_i w'\sigma'$ iff $w\sigma \sim^{\sigma^L}_i w'\sigma'$, where $\sim^{\sigma^L}$ is defined by induction in the following way:
  - $\sim^{\sigma^L_{(0)}}_i = \sim^0_i$
  - For each $w\tau, v\tau' \in H_{n+1}$ $(0 < n < \mathsf{len}(\sigma^L))$, $w\tau \sim^{\sigma^L_{(n+1)}} v\tau'$ iff $w\tau_{(n)} \sim^{\sigma^L_{(n)}}_i v\tau'_{(n)}$ and $(\tau^R_{n+1}, (\tau')^R_{n+1}) \in Rel(\tau^L_{n+1})(i)$.
- For every $p \in \mathsf{At}$ and $h = w\sigma \in H^{can}$, $w\sigma \in V^{can}(p)$ iff $w \in V^0(p)$.

◁

The above definition simulates Definition 1.4.8 and 1.4.9. The construction guarantees that, at each stage along histories in the canonical model, $\sim^{can}_i$ respects updates made by corresponding event models. This makes the following proofs simpler.

Proposition 2.5.7 *Let* $w\sigma \sim_i^{can} v\tau$ *with* $w, v \in W^0$, $\sigma = \sigma_1 \dots \sigma_n$ *and* $\tau = \tau_1 \dots \tau_n$. *If* $[i]\varphi \in \lambda(w\sigma)$, *then*

$$[i](\langle\tau_1\rangle\top \to \langle\tau_1\rangle(\langle\tau_2\rangle\top \to \langle\tau_2\rangle(\dots(\langle\tau_n\rangle\top \to \langle\tau_n\rangle\varphi)\dots) \in \lambda(w).$$

Proof. By induction on n. When $n = 0$, σ, τ are empty and thus the claim clearly holds. For the inductive step, assume that $[i]\varphi \in \lambda(w\sigma)$. Then, by the construction of $\mathcal{H}^{can}$, $\langle\sigma_n\rangle[i]\varphi \in \lambda(w\sigma_{(n-1)})$. By **F4**, for all events e in $\sigma_n^L = (E, \to, V)$ such that $\sigma_n^R \to_i e$:

$$[i](\langle\sigma_n^L, e\rangle\top \to \langle\sigma_n^L, e\rangle\varphi) \in \lambda(w\sigma_{(n-1)}).$$

Here, since $w\sigma \sim_i^{can} v\tau$, we have $\sigma_n^R \to_i \tau_n^R$ by the construction of $\mathcal{H}^{can}$. By applying the IH, we are done. QED

This proposition makes sure that the argument given for the truth lemma in TPAL (Lemma 2.3.7) can be carried out for TDEL. In TPAL, we did not need to prove the lemma of this sort, since $w\sigma \sim_i v\tau$ obtains just in case $w \sim_i v$ and $\sigma = \tau$. Therefore, when $[i]\varphi \in \lambda(w\sigma)$, we have

$$[i](\langle\sigma_1\rangle\top \to (\dots(\langle\sigma_{\mathsf{len}(\sigma)}\rangle\top \to \langle\sigma_{\mathsf{len}(\sigma)}\varphi)\dots) \in \lambda(w)$$

and this implies by construction of $\sim_i$ that

$$(\langle\sigma_1\rangle\top \to (\dots(\langle\sigma_{\mathsf{len}(\sigma)}\rangle\top \to \langle\sigma_{\mathsf{len}(\sigma)}\varphi)\dots) \in \lambda(v).$$

From this, we could argue that $\varphi \in \lambda(v\sigma)$ in the argument. However, in TDEL, $w\sigma \sim_i v\tau$ does not generally imply that $\sigma = \tau$. Therefore, we needed the above proposition to guarantee

$$(\langle\tau_1\rangle\top \to (\dots(\langle\tau_{\mathsf{len}(\sigma)}\rangle\top \to \langle\tau_{\mathsf{len}(\sigma)}\varphi)\dots) \in \lambda(v).$$

given $w\sigma \sim_i v\tau$. This enables us to carry out the argument and obtain the truth lemma stated as follows:

Lemma 2.5.8 (Truth Lemma) *For every* $\varphi \in \mathcal{L}_{\mathsf{TDEL}}$ *and* $h \in \mathsf{H}^{can}$,

$$\varphi \in \lambda(h) \quad \textit{iff} \quad \mathcal{H}^{can}, h \models \varphi.$$

We can also prove that $\mathcal{H}^{can}$ is in $\mathbb{F}_{sd}$ by the argument given in TPAL for an analogous lemma (Lemma 2.3.8)

Lemma 2.5.9 *The canonical model* $\mathcal{H}^{can}$ *is in* $\mathbb{F}_{sd}$. *That is, there is an epistemic model* $\mathcal{M}$ *and local protocol* p *on* $\mathcal{M}$ *such that* $\mathcal{H}^{can} = \mathsf{Forest}(\mathcal{M}, \mathsf{p})$.

Therefore, we have:

Theorem 2.5.10 TDEL *is sound and strongly complete with respect to* $\mathbb{F}_{sd}$.

2.5.3 TDEL Restricted to Subclasses of Protocols

TDEL axiomatizes the class $\mathbb{F}_{sd}$. However, note that the completeness proof above does not depend on the fact that TDEL allows the *whole* class of pointed event models. Indeed, even if we restrict our attention to subclasses of pointed event models, the proof should work. However, here we have to be careful that we must at least have all the "relevant" pointed event models: if $(\mathcal{E}, e)$ is in the class of our interest, then $(\mathcal{E}, e')$ is also in for all e' in $\mathcal{E}$. Otherwise the knowledge modality case of Lemma 2.5.8 would fail, since all the "relevant" histories must be included in the canonical model.

Definition 2.5.11 (*e*-Closure) Let $X \subseteq \mathbb{E}$. Call X *e-closed* if, for all $\mathcal{E}$, if there is $\epsilon \in X$ such that $\epsilon^L = \mathcal{E}$, then, for every event e in $\mathcal{E}$, (ϵ^L, e) is in X. $\triangleleft$

Definition 2.5.12 (TDEL(X)) Denote by $\mathcal{L}_{tdel}(X)$ the fragment of $\mathcal{L}_{tdel}$ that only allows the event model operators $\langle\epsilon\rangle$ such that $\epsilon \in X$. Also, let $\mathsf{TDEL}(X)$ be axiomatized as in Definition 2.5.3 except that the axiom schemas and the $\epsilon\mathbf{N}$ rule can only be instantiated by the event models in X. $\lhd$

The following claim can follows from the above considerations.

Theorem 2.5.13 *Let X be an* e*-closed subclass of $\mathbb{E}$. Denote by $\mathbb{X}$ the class of* sd*-protocols whose values are subsets of X^*. (*sd*-protocols that only allows events in X.) Then $\mathsf{TDEL}(X)$ is sound and complete with respect to $\mathbb{F}(\mathbb{X})$.*

2.5.4 Decidability

Next, having the above completeness proof for TDEL(X), we can combine it with the finite completeness argument for TPAL (Section 2.3.3) and show that the satisfiability problem of TDEL(X) (with X e-closed) is decidable. The main idea in Section 2.3.3 was to construct the finite canonical model from a finite set of formulas that satisfies certain closure conditions. For the decidability of TDEL(X), we need to revise the closure conditions so that we can carry out the completeness argument of TDEL in the finite canonical model.

To state the closure conditions, first define the depth of a formula φ in TDEL as in TPAL (Definition 2.2.4) so that $d(\varphi)$ is the greatest length of the consecutive occurrences of event operators in φ.

Definition 2.5.14 (TDEL-Closed Sets) Let Σ be a set of formulas. Σ is *TDEL-closed* if (i) Σ is closed under subformulas and single negations (as in Definition 2.3.9) and (ii) satisfies the following conditions:

1. If $\langle\mathcal{E}, e\rangle\varphi \in \Sigma$, then $\langle\mathcal{E}, e'\rangle\top \in \Sigma$ for all e' in $\mathcal{E}$.

2. If $\langle \mathcal{E}, e\rangle \top \in \Sigma$, then $\mathsf{pre}(\mathcal{E})(e) \in \Sigma$.

3. If $\langle \mathcal{E}, e\rangle [i]\varphi \in \Sigma$, then $[i](\langle(\mathcal{E}, e'\rangle \top \to \langle \mathcal{E}, e'\rangle \varphi) \in \Sigma$ for all e' in $\mathcal{E}$ such that $(e, e') \in \to_{\mathcal{E}} (i)$.

4. If $\varphi \in \Sigma$, then $\langle !\epsilon_1\rangle ... \langle !\epsilon_k\rangle \varphi \in \Sigma$ $(1 \leq k \leq d(\Sigma) - d(\varphi))$ where $\langle \epsilon_i\rangle \top \in \Sigma$ for every $1 \leq i \leq k$. $\lhd$

Given a set Σ, denote by $(\Sigma)^{TDEL}$ the smallest set that contains Σ with the above closure properties. Since event models are finite (Definition 1.2.4), $(\Sigma)^{TDEL}$ is finite if Σ is finite.

Once this definition is given, the rest of the proof is similar to Section 2.3.3. Given a consistent formula φ in TDEL(X), we take $\{\varphi\}^{TDEL}$. Based on this set, we define *atoms* and construct finite canonical models in a way similar to Section 2.3.3. (Of course, the construction will use the canonical model construction in TDEL as in Definition 2.5.6, but not the one in TPAL.) The above closure conditions then guarantee that the rest of the argument can be carried out. Therefore, we can obtain the decidability of TDEL(X).

Theorem 2.5.15 (Decidability of TDEL(X)) *Let X be an* e*-closed set of pointed event models. The satisfiability problem for the logic TDEL(X) is decidable.*

2.5.5 Other Epistemic Operators?

The results in this section, together with the results in Section 2.3.4, suggest the possibility of incorporating (relativized) common knowledge operator into our system TDEL(X). If we have an axiom similar to **R5**, we will be able to axiomatize the system with relativized common knowledge as we argued in Section 2.3.4. Can we obtain such an axiom schema? Or generally can we obtain the corresponding axiom schemas every time we introduce new epistemic operators?

This question is answered in the context of DEL by van Benthem *et al* in [71]. They introduce a general algorithm to compute reduction axioms for epistemic operators expressible in their language, epistemic PDL. Therefore, we may ask the same question in the context of TDEL. Can we come up with a general algorithm to compute 'reduction'-like axioms? Many of the constructions we have seen so far in terms of TDEL suggests that it should be possible. However, we will leave the question for future research.

2.6 Generalization of Other Results in TDEL

The completeness proof is not the only technique that we can generalize for the full TDEL. The other results we saw in TPAL can be also extended. In this section, we will sketch how we can extend other results: model normalization, uniform protocols and embeddability.

2.6.1 Normalization

First, model normalization can be generalized to TDEL, based on the same idea as in TPAL. We replace preconditions of events by tautologous formulas without distorting the structures of ETL-trees. We can prove a truth-preservation result analogous to Proposition 2.2.14. Here we will not describe the full formal details, but sketch how to proceed.

Let $\alpha_0, \alpha_1, \dots$ and $\beta_0, \beta_1, \dots$ be a pair of (possibly infinite) sequences of pointed event models such that, for all $k, l \geq 0$, (i) the preconditions of β_k^R is a tautologous formula in $\mathcal{L}_{el}$ and (ii) $\alpha_k \neq \alpha_l$ and $\beta_k \neq \beta_l$. Given a sequence $h \in \mathbb{E}^*$, define $h[\beta_0/\alpha_0, \beta_1/\alpha_1, \dots]$ to be the sequence obtained by replacing all occurrences of α_k in H with β_k (for all k). Given an DEL-generated ETL model $\mathcal{H} = (H, \sim, V)$, define $\mathcal{H}[\beta_0/\alpha_0, \beta_1/\alpha_1, \dots] = (H', \sim', V')$ by:

$H' := \{h[\beta_0/\alpha_0, \beta_1/\alpha_1, \dots] \mid h \in H\}$

$\sim' (i) = \{(h[\beta_0/\alpha_0, \beta_1/\alpha_1, \dots], g[\beta_0/\alpha_0, \beta_1/\alpha_1, \dots]) \mid (h, g) \in \sim (i)\}$

$V'(p) = \{h[\beta_0/\alpha_0, \beta_1/\alpha_1, \dots] \sim' g[\beta_0/\alpha_0, \beta_1/\alpha_1, \dots] \mid h \in V(p)\}.$

Given a formula in $\mathcal{L}_{tdel}$, define the *event occurrence set* $EOC(\varphi)$ of φ to be the set of pointed event models occurring in φ. (cf. Definition 2.2.6)

Proposition 2.6.1 (Normalization in TDEL) *Let* $\mathcal{H} = \mathsf{Forest}(\mathcal{M}, \mathsf{p}) \in \mathbb{F}(\mathbb{E})$. *Let* X *be a finite subset of* $\mathbb{E}$. *Furthermore, let* $\alpha_0, \alpha_1 \dots$ *and* $\beta_0, \beta_1, \dots$ *be enumerations of elements in* $\mathbb{E} \backslash X$ *without repetition such that, for all* k*, the precondition of* β_k^R *is a tautologous formula in* $\mathcal{L}_{el}$. *Then, for every* h *and TDEL-formula* φ *such that* $EOC(\varphi) \subseteq X$,

$$\mathcal{H}, h \models \varphi \quad \Leftrightarrow \quad \mathcal{H}[\beta_0/\alpha_0, \beta_1/\alpha_1, \dots], h[\beta_0/\alpha_0, \beta_1/\alpha_1, \dots] \models \varphi$$

2.6.2 Uniform Protocols

We can also generalize the axiomatization of uniform PAL-protocols TPAL^{Uni} (Section 2.4.1) to obtain the axiomatization of uniform TDEL-protocols. Fix an e-closed set X below. We first need to generalize the definition of the existential operator E. In the context of TPAL, the existential modality was defined as follows:

$$\mathsf{Forest}(\mathcal{M}, \mathsf{p}), w\sigma \models E\varphi \text{ iff } \exists v \in W \text{ such that } v\sigma \text{ is in } \mathcal{H} \text{ and } \mathcal{H}, v\sigma \models \varphi.$$

where $w \in Dom(M)$ and σ is a sequence of public announcements (i.e. $\sigma \in \mathsf{PAL}^*$. The operator could be defined this way, since event models and events, so to speak, do not have to be distinguished in TPAL. Event models that represent public announcements contain single events and thus we do not have to specify which event in an public announcement event model we talk about. In the context of TDEL, we need to be explicit about the distinction. The definition is thus given as follows:

$$\mathsf{Forest}(\mathcal{M}, \mathsf{p}), w\sigma \models E\varphi \text{ iff } \exists v \in Dom(\mathcal{M})\exists\tau \in \mathbb{E}^* : \sigma^L = \tau^L \text{ and } \mathcal{H}, v\tau \models \varphi.$$

where w is in $\mathcal{M}$ and $\sigma \in \mathbb{E}^*$. Defined this way, $E\varphi$ reads as "φ is true at some history with the same sequence of product updates". We denote the dual of E by U and read $U\varphi$ as "φ is true at every history with the same sequence of product updates". Finally we denote the extension of TDEL(X) by TDEL^E(X).

The complete axiomatization of TDEL^E(X) can be obtained straightforwardly as in Section 2.4.1. We need to add the standard axiom schemas for the existential modality, **E1-5**, and the following axiom to $\mathsf{TDEL}(X)$:

F5 $\langle\mathcal{E}, e\rangle E\varphi \leftrightarrow \langle\mathcal{E}, e\rangle\top \wedge E\bigvee_{e' \in \{e' \in Dom(\mathcal{E}) | (e,e') \in \to_{\mathcal{E}}(i)\}}\langle\mathcal{E}, e'\rangle\varphi$

F5 is an analogue of **R5** in TPAL^E. (cf. also **F4** in TDEL) We denote the resulting axiomatization by $\mathsf{TDEL}^E(X)$.

We now would like to axiomatize the class $\mathbb{F}(Ptcl(X))$ of ETL models generated from uniform protocols in $Ptcl(X)$. (Remember $Ptcl(X) = \wp(X^*)$). For this, we extend the axiomatization $\mathsf{TDEL}^E(X)$ with the axiom that expresses the uniformity of protocols, as we did in the context of TPAL. The uniformity of protocols in the context of TDEL can be generalized by simply replacing public announcements in **Uni** with pointed event models, as expected:

Uni$_X$ $\langle\mathcal{E}, e\rangle\top \to U(\mathsf{pre}_{\mathcal{E}}(e) \to \langle\mathcal{E}, e\rangle\top)$, where $(\mathcal{E}, e) \in X$.

Let $\mathbb{X}$ be the class of *sd*-protocols whose values are subsets of X^*. We say a state-dependent protocol $\mathsf{p} \in \mathbb{X}$ *generates a uniform ETL model* if $\mathsf{Forest}(\mathcal{M}, \mathsf{p}) = \mathsf{Forest}(\mathcal{M}, \mathsf{P})$ for some $\mathsf{P} \in Ptcl(X)$. Then we can proceed as in the proof of Proposition 2.4.1 and prove the following:

Proposition 2.6.2 *Let* p *be in* $\mathbb{X}$. *The axiom* $\mathbf{Uni}_X$ *is valid in* $\mathsf{Forest}(\mathcal{M}, \mathsf{p})$ *iff* p *generates a uniform ETL model.*

Let $\mathsf{TDEL}^{Uni}(X)$ be the extension of $\mathsf{TDEL}^{E}(X)$ with the axiom $\mathbf{Uni}_X$. The following is an immediate consequence of a suitable truth lemma analogous to Lemma 2.3.7 and Proposition 2.6.2:

Corollary 2.6.3 *Let* X *be an* e*-closed set of pointed event models.* $\mathsf{TDEL}^{Uni}(X)$ *is sound and strongly complete with respect to the class* $\mathbb{F}(Ptcl(X))$.

2.6.3 Embedding DEL into TDEL

Finally, we can embed DEL into TDEL by generalizing the technique used in Section 2.4.2. The technique was, given a formula φ, to construct a formula that expresses that sequences of public announcements that are relevant to the truth value of φ are all allowed by protocols. In the context of TPAL, the 'relevant' public announcements were the ones occurring in φ. In the general setting of TDEL, the pointed event models relevant to the truth of a given formula φ are not only the ones that occur in φ. The set of relevant pointed event models must be e-closed, that is, if $(\mathcal{E}, e)$ is in X, then $(\mathcal{E}, e') \in X$ for all e' in $\mathcal{E}$.

Given a set of pointed event models X, denote by X^e the smallest set Y such that $X \subseteq Y$ and Y is e-closed. Also denote by $EOC(\varphi)$ the set of pointed event models occurring in a formula φ. Given a formula φ, let $Ptcl(\varphi)$ be the set of formulas of the form:

$$U(\mathsf{pre}_{\mathcal{E}_1}(e_1) \to \langle \mathcal{E}_1, e_1 \rangle (\mathsf{pre}_{\mathcal{E}_2}(e_2) \to \langle \mathcal{E}_2, e_2 \rangle (\cdots \langle \mathcal{E}_{k-1}, e_{k-1} \rangle (\mathsf{pre}_{\mathcal{E}_k}(e_k) \to \langle \mathcal{E}_k, e_k \rangle \top) \cdots)))$$

where $(\mathcal{E}_i, e_i) \in (EOC(\varphi))^e$ $(1 \leq i \leq k)$ and $1 \leq k \leq d(\varphi)$. (The operator U is as defined in Section 2.6.2.)

Having these machinery, we can proceed as in the proof of Theorem 2.4.3 and prove the following embeddability result.

Theorem 2.6.4 *For any formula* $\varphi \in \mathcal{L}_{tdel}$,

$$\models \varphi \textit{ in DEL} \qquad \textit{iff} \qquad \models \bigwedge Ptcl(\varphi) \rightarrow \varphi \textit{ in TDEL.}$$

2.7 Conclusion and Discussion

We have studied logics on classes of ETL models generated from classes of protocols. We started by investigating the logic TPAL over the class of PAL-generated ETL models. We can characterize our main results in TPAL as follows. First, we showed that model normalization preserves the truth of formulas in TPAL. Second, we axiomatized the class of all PAL-generated ETL models and then extended our system with relativized common knowledge. Third, by introducing the existential modality, we axiomatized the class of ETL models generated based on uniform protocols. Forth, we also used the existential modality to faithfully embed PAL into TPAL.

After studying TPAL in detail, we applied the techniques in TPAL to logics over other subclasses of DEL-generated ETL models and obtained similar results. For instance, Theorem 2.5.13 provides axiomatizations for logics over various subclasses of DEL-generated ETL models. Beyond public announcements, there are other informational events of our interest, such as secret communication ([50]), honest communication ([68]), etc. The theorem (and other results) present general methods in investigating logics of specific kinds of protocols.

Also there are other open questions that come out of our study in this chapter. For instance, Theorem 2.5.15 shows that the satisfiability problem for TDEL(X) is decidable. However, it does not give us the precise computational complexity of the systems. In addition, the theorem only shows the decidability of systems with state-dependent protocols. The decidability of systems with uniform protocols is still an open question. Furthermore, although Theorem 2.6.4 embeds DEL into our framework, it is unknown whether we can go the other way to embed TDEL into DEL. Finally, we discussed the question whether we can obtain an algorithm to compute 'reduction' axioms in TDEL(X), as [71] provides in the context of DEL (Section 2.5.5). We will leave these questions for future investigation.

Chapter 3

Extensions

In this chapter, we will consider extensions of the systems developed in the previous chapter. There are two kinds of extensions that we consider. One kind of extension concerns the language of TDEL. Our minimal language of TDEL have the two modalities, the knowledge operator $[i]$ and the labeled event operators $[\epsilon]$. There are some natural extensions that suggest themselves. For instance, both DEL and ETL deal with various kinds of *epistemic* operators other than $[i]$ and among them is *common knowledge*, which we have considered in the previous chapter (Section 2.3.4). Likewise, explicit time-branching structures in our models motivate *temporal* operators that have been considered in the system of ETL. One such operator is an operator that involves *quantification* over *future* events "Some event can happen after which…", "Some sequences of events can happen after which…", etc. Operators of this kind have been also considered in DEL in [5]. Another kind of operators describe what happened in the *past*, "Previously,…", "before the event e happens…", etc. From the perspective of DEL, such a kind of operators are interesting since DEL usually deals only with future operators.[1] Thus the first goal of this chapter is to consider

[1] We do not claim that past-operators have not been considered in the setting of DEL. Indeed, past-operators have been considered in [83].

the extensions of our system with the two kinds of temporal operators.

The other kind of extension that we consider in this chapter is a generalization of the model construction in our framework. Remember that the preconditions in event models are restricted to epistemic formulas (formulas in $\mathcal{L}_{el}$) and thus they cannot contain event operators. The restriction does not seem substantial in the context of DEL, since formulas in DEL reduces equivalently to epistemic formulas by reduction axioms. On the other hand, full reduction axioms are not available in TDEL, as we observed in the previous chapter, and our framework does not provide a way to express preconditions by formulas containing event operators. This restriction of our system can be a big obstacle in applications of our framework. For instance, in TPAL, we cannot deal with public announcements expressing what *will* be true in the future, etc. Therefore, the second goal of this chapter is to consider a way to lift the restriction and allow the full class of event models to be in protocols.

We will tackle these problems in the simplest setting of our framework, i.e. TPAL. We will proceed as follows. We will start out by extending TPAL with the generalized future operator "Some public announcements can be made after which ...". (Section 3.1) Next, we will deal with the extension with the past operator in TPAL. (Section 3.2) Then we will turn to the extension of models in TPAL to lift the assumption on preconditions (Section 3.3). After seeing the extensions of TPAL, we will discuss whether it is possible to give similar extensions in TDEL. (Section 3.4).

3.1 Quantifying over Public Announcements

We will study the extension of TPAL with the generalized public announcement operator "Some public announcement can be made after which...". The kind of operator can be motivated on various grounds. First, the operator has been one of the standard operators in ETL. Since our framework is based on ETL-time-branching tree structures, it is natural to ask how the operator behaves in the framework. Second, the operator has been recently considered in the literature on DEL. Balbiani *et al* [5] considers the extension of PAL with the operator, which they call *Arbitrary Public Announcement Logic* (APAL). It is interesting to compare the system and the corresponding extension of our system. Third, the generalized public announcement operators enable us to express various epistemic concepts of our interest. For instance, with the operator, we can express questions, such as whether there are some public announcements after which epistemic states of interest will be reached. That is, with the generalized public announcement operator, we can formulate the reachability question, which motivated our protocol-based semantic framework in the first place. Such a reachability question has a great importance to the notion of *knowability* (whether a given proposition is knowable), which we will discuss in Chapter 4.

3.1.1 Temporal Arbitrary Public Announcement Logic

To extend TPAL with the kind of operator in question, we need some preliminary considerations. In the framework of PAL, Balbiani *et al* [5] consider the operator $\Diamond$, where the intended reading of $\Diamond\varphi$ is "Some public announcement can be made after which φ." The semantics of the operator is given by:

$$\mathcal{M}, w \models \Diamond\varphi \quad \text{iff} \quad \exists\psi \in \mathcal{L}_{pal} : \mathcal{M}, w \models \langle !\psi\rangle\varphi.$$

They call the extension of PAL with the operator *Arbitrary Public Announcement* (APAL). The language of APAL is denoted by $\mathcal{L}_{apal}$.

Now, to consider such a generalized operator in TPAL, we start out by noting the following fact. In PAL, sequences of announcements are identified with some single announcements, in terms of the validity of the following schema:

$$\langle !\alpha\rangle\langle !\beta\rangle\varphi \leftrightarrow \langle !(\langle !\alpha\rangle\beta)\rangle\varphi.$$

However, in TPAL, this is not the case. TPAL invalidates the schema, since the corresponding single announcements may not be available even if sequences of announcements are available. (Proposition 2.2.2 in Chapter 2) Thus, we have to distinguish single announcements and sequences of announcements in the semantic framework of TPAL. This consideration motivates us to introduce two kinds of generalized public announcement operators to distinguish quantifications over single announcements and sequences of announcements.

Fix a set of agents $\mathcal{A}$ and a countable set of propositional letters At.

Definition 3.1.1 (Language of TAPAL) The language $\mathcal{L}_{tapal}$ of TAPAL extends $\mathcal{L}_{tpal}$ with the operators $\Diamond$ and $\Diamond^*$. The formulas in $\mathcal{L}_{tapal}$ is inductively defined by:

$$\varphi \ ::= \top \mid p \mid \neg\varphi \mid \varphi \wedge \varphi \mid [i]\varphi \mid \langle !\theta\rangle\varphi \mid \Diamond\varphi \mid \Diamond^*\varphi$$

where $p \in \mathsf{At}$, $i \in \mathcal{A}$ and $\theta \in \mathcal{L}_{el}$. The duals, $\Box$ and $\Box^*$, of $\Diamond$ and $\Diamond^*$ are defined in the standard way. The other operators are defined as mentioned in Definition 2.1.2. $\lhd$

The intended interpretations of $\Diamond\varphi$ and $\Box\varphi$ are "Some public announcement can be made after which φ is true" and "After every public announcement, φ is true." respectively. Also the intended interpretations of $\Diamond^*\varphi$ and $\Box^*\varphi$ are "Some sequences of public announcement can be made after which φ is true." and "After every sequence of public announcement, φ is true." respectively. (Sequences here are possibly empty.) We call the extension *Temporal Arbitrary Public Announcement Logic* (TAPAL).

Definition 3.1.2 (Truth) Let PAL and $\mathbb{PAL}$ be the class of public announcements in $\mathbb{E}$ and the class of state-dependent PAL-protocols.[2] Given $\mathcal{H} = (H, \sim, V) \in \mathbb{F}(\mathbb{PAL})$ and a history $h \in H$, the truth of a TAPAL-formula φ at h is inductively defined as follows. We only give the definitions for $\Diamond$, and $\Diamond^*$. The other definitions are as given in Definition 2.1.5:

$$\begin{array}{lll} \mathcal{H}, h \models \Diamond\varphi & \text{iff} & \exists !\psi \in \mathsf{PAL} : h!\psi \in H \text{ and } \mathcal{H}, h!\psi \models \varphi \\ \mathcal{H}, h \models \Diamond^*\varphi & \text{iff} & \exists \sigma \in \mathsf{PAL}^* : h\sigma \in H \text{ and } \mathcal{H}, h\sigma \models \varphi \end{array}$$

Consistency, satisfiability, validity etc. are defined in the standard way as in Definition 2.1.6. ◁

Remark 3.1.3 (Language of TAPAL) Some remarks are in order concerning the language of TAPAL. First note that the restriction, $\theta \in \mathcal{L}_{el}$, in Definition 3.1.1 makes the truth definition of the generalized operators $\Diamond$ and $\Diamond^*$ well-defined (as well as it reflects our notion of PAL-protocols and DEL-generated ETL models). For suppose all TAPAL-formulas are allowed in sd-protocols. Assume further that $\Box\varphi \in \mathsf{p}(w)$ for some w in a given model. By the truth definition in Definition 3.1.2, to determine the truth value of $\Box\varphi$, we need to know the truth value of $\Box\varphi$. A similar restriction is made in APAL in [5]. Second, given the restriction, we also needed to defined the public announcement operators to be formed only from the formulas in $\mathcal{L}_{el}$. By this, we do not allow formulas such as $\langle !\Diamond\psi\rangle\varphi$, which is allowed in APAL. ◁

[2] Thus we only consider public announcements $!\varphi$ with $\varphi \in \mathcal{L}_{el}$, as is in TPAL.

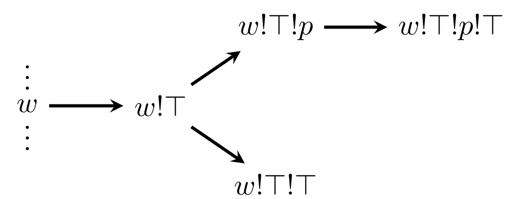

Figure 3.1: TAPAL-Model 1

3.1.2 Semantic Results

Next we see some basic semantic features of TAPAL in comparison with those in APAL. First, consider the following properties:

1. $\models \Box\varphi \to \varphi$
2. $\models \Box\varphi \to \Box\Box\varphi$
3. $\models \Box\Diamond\varphi \to \Diamond\Box\varphi$
4. $\models \Diamond\Box\varphi \to \Box\Diamond\varphi$

Proposition 3.1.4 *Generalized Operators*

(A) *All of the properties 1-4 hold in APAL.*

(B) *None of the properties 1-4 holds in TAPAL.*

(C) *The properties 1-2 hold, but 3 and 4 don't in TAPAL, when $\Diamond$ and $\Box$ are replaced with $\Diamond^*$ and $\Box^*$ respectively.*

Proof. The proofs of the properties A1-4 in $APAL$ are in [5]. We only do B3-4 and C3-4. The counterexamples are as follows:

B3 Let $\mathcal{M}, w \models p$. Define $\mathsf{p}(w) = \{!\top, !\top!\top, !\top!p, !\top!p!\top\}$. The model $\mathcal{H} = Forest(\mathcal{M}, \mathsf{p})$ can be represented by Figure 3.1. Here we have $\mathcal{H}, w!\top!p \models \langle!\top\rangle\top$, but $\mathcal{H}, w!\top!\top \not\models \langle!\top\rangle\top$. Therefore, we have $\mathcal{H}, w \models \Box\Diamond\langle!\top\rangle\top$, but $\mathcal{H}, w \models \Box\Diamond\neg\langle!\top\rangle\top$, i.e. $\mathcal{H}, w \not\models \Diamond\Box\langle!\top\rangle\top$.

B4 In Figure 3.1, $\mathcal{H}, w!\top!p \models \Box\top$, which yields $\mathcal{H}, w!\top \models \Diamond\Box\top$, but $\mathcal{H}, w!\top!\top \not\models \Diamond\top$, which yields $\mathcal{H}, w!\top \not\models \Box\Diamond\top$.

C3 Let $\mathcal{M}, w \models p$. Define $\mathsf{p}(w) = \{!\top, !\top!p, !\top!p!\top, !\top!p!\top!p, ...\}$. Let $\mathcal{H}$ be $Forest(\mathcal{M}, \mathsf{p})$. We claim that, for every h in $\mathcal{H}$, there exists $\sigma, \sigma' \in \mathsf{p}(w)$ such that $\mathcal{H}, h\sigma \models \langle !\top \rangle \top$ and $\mathcal{H}, h\sigma' \not\models \langle !\top \rangle \top$. To see this, note that every h ends with either $\top$ or p. If h ends with $!\top$, then put $\sigma =!p$ and $\sigma' = \emptyset$; if h ends with $!p$, then put $\sigma = \emptyset$ and $\sigma' =!\top$. This fact implies $\mathcal{H}, w \models \Box\Diamond\langle !\top \rangle \top$ and $\mathcal{H}, w \models \Box\Diamond\neg\langle !\top \rangle \top$. Thus, this model is a counterexample against 3.

C4 The models for B4 similarly works. QED

Now we will see the results concerning expressivity of the operators $\Diamond$ and $\Diamond^*$. First the operators 'implicitly' denote (sequences of) announcements by quantification, we do not have results similar to Proposition 2.2.7 in TPAL. This feature of the operators adds expressive power to TPAL as in the following result. Let $AOC(\varphi)$ be the *announcement occurrence set* of φ, the set of public announcements occurring in φ. (See Definition 2.2.6.)

Proposition 3.1.5 *Both TAPAL is strictly more expressive than TPAL.*

Proof. Consider the formula $\Diamond p$. Assume toward contradiction that this formula is equivalent to some TPAL-formula ψ. Since TPAL-formulas are finite, there are only a finite number of propositional letters, $q_1, q_2, ..., q_n$, used in ψ. Let q_{n+1} be a propositional letter that is distinct from all $q_1, q_2, ..., q_n$, and $\mathcal{M}$, an epistemic model with only a state w at which p, q_{n+1} are both true. Then define $\mathsf{p}_1, \mathsf{p}_2$ be $\mathsf{p}_1(w) = \emptyset$ and $\mathsf{p}_2(w) = \{!q_{n+1}\}$. Now consider $\mathcal{H}_1 = \mathsf{Forest}(\mathcal{M}, \mathsf{p}_1)$ and $\mathcal{H}_2 = \mathsf{Forest}(\mathcal{M}, \mathsf{p}_2)$.

Since $(\mathsf{p}_1(w))_{AOC(\psi)} = (\mathsf{p}_2(w))_{AOC(\psi)}$, it follows from Proposition 2.2.7 that ψ has the same value at $\mathcal{H}_1, w$ and $\mathcal{H}_2, w$. However, clearly $\mathcal{H}_1, w \not\models \Diamond p$ and $\mathcal{H}_2, w \models \Diamond p$. This is a contradiction. QED

On the other hand, since $\Diamond$ and $\Diamond^*$ are future-looking in the sense that the truth value of the formulas does not depend on the nodes below a point of evaluation, we

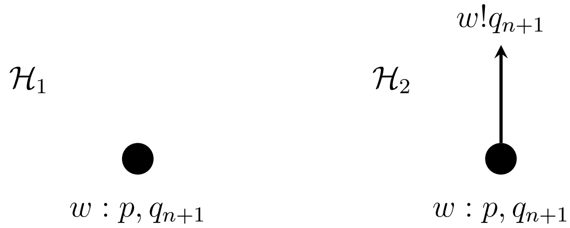

Figure 3.2: TAPAL-Model 2

can obtain a result similar to Proposition 2.2.5 in TPAL. However, we cannot place the explicit upper bound for every formula in TAPAL by the depth of the formula. This is because the operator $\Diamond^*$ quantifies over all finite sequences of future public announcements.

Observation 3.1.6 *Let $\mathcal{M}$ be an epistemic model, p a state-dependent protocol on $\mathcal{M}$. For all $w \in Dom(\mathcal{M})$ and $\sigma \in \bigcup_{w \in Dom(\mathcal{M})} p(w)$, if $\Diamond^*$ does not occur in φ, then*

$$\mathsf{Forest}(\mathcal{M}, \mathsf{p}), w\sigma \models \varphi \quad \textit{iff} \quad \mathsf{Forest}(\mathcal{M}^{\sigma,\mathsf{p}}, \mathsf{p}^{\sigma<}_{d(\varphi)}), w\sigma \models \varphi.$$

where $d(\varphi)$ is the highest number of nested occurrences of operators of the form $\langle !\theta \rangle$ or $\Diamond$. (cf Definition 2.2.4) If $\Diamond^*$ occur in φ, then

$$\mathsf{Forest}(\mathcal{M}, \mathsf{p}), w\sigma \models \varphi \quad \textit{iff} \quad \mathsf{Forest}(\mathcal{M}^{\sigma,\mathsf{p}}, \mathsf{p}^{\sigma<}), w\sigma \models \varphi.$$

This feature of $\Diamond^*$ yields the following proposition.

Proposition 3.1.7 *TAPAL is strictly more expressive than its fragment without $\Diamond^*$.*

Proof. Consider $\Box^*\langle !\top \rangle \top$. Assume toward contradiction that this formula is equivalent to some formula without occurrences of $\Diamond^*$. Let $\mathcal{M}$ be an epistemic model with only a state w. Let us denote as $!\top^k$ the sequence of k $!\top$'s. Let $d(\psi)$ be the highest number of nested occurrences of the operators $\langle !\theta \rangle$ and $\Diamond$ in ψ. Define

$$\mathcal{H}_1: \quad w \longrightarrow w!\top \longrightarrow \cdots \longrightarrow w!\top^{d(\psi)}$$

$$\mathcal{H}_2: \quad w \longrightarrow w!\top \longrightarrow \cdots \longrightarrow w!\top^{d(\psi)} \longrightarrow \cdots$$

Figure 3.3: TAPAL-Model 3

$\mathsf{p}_1, \mathsf{p}_2$ be such that $\mathsf{p}_1(w) = \{!\top^i | 0 \leq i \leq d(\psi)\}$ and $\mathsf{p}_2(w) = \{!\top^i | i \in \mathbf{N}\}$. Now consider $\mathcal{H}_1 = \mathsf{Forest}(\mathcal{M}, \mathsf{p}_1)$ and $\mathcal{H}_2 = \mathsf{Forest}(\mathcal{M}, \mathsf{p}_2)$. The models are visualized in Figure 3.3. Since $\mathcal{H}_2 = \mathsf{Forest}(\mathcal{M}, (\mathsf{p}_2)^{\lambda<}_{d(\psi)})$ with λ the empty sequence, it follows from Proposition 3.1.6 that ψ has the same value at $\mathcal{H}_1, w$ and $\mathcal{H}_2, w$. On the other hand, $\mathcal{H}_1, w\top^{d(\psi)} \not\models \langle !\top \rangle \top$, which implies $\mathcal{H}_1, w \not\models \Box^* \langle !\top \rangle \top$, whereas clearly $\mathcal{H}_2, w \models \Box^* \langle !\top \rangle \top$.

QED

Furthermore, the expressive power of $\Diamond$ and $\Diamond^*$ renders the systems non-compact, as in the case of APAL (see [5]).

Proposition 3.1.8 *TAPAL is not compact.* ◁

Proof. Straightforward by considering the set $\Gamma = \{\neg \langle !\theta \rangle p | \theta \in \mathcal{L}_{el}\} \cup \{\Diamond p\}$ or the set $\bigcup_{i=0}^{\infty} \Gamma_i \cup \{\Diamond^* p\}$, where $\Gamma_i = \{\neg \langle !\theta_0 \rangle ... \langle !\theta_i \rangle p | \theta_j \in \mathcal{L}_{el} \ (0 \leq j \leq i)\}$. QED

3.1.3 Axiomatization

Next we axiomatize the logic of TAPAL. Let $\Diamond^n$ and $\Box^n$ be the sequences of n $\Diamond$'s and $\Box$'s respectively. When $n = 0$, $\Diamond^n$ and $\Box^n$ denote φ. Also given $\sigma = \sigma_0 \ldots \sigma_{n-1} \in \mathsf{PAL}^*$, denote the sequences $\langle \sigma_0 \rangle \ldots \langle \sigma_{n-1} \rangle$ and $[\sigma_0] \ldots [\sigma_{n-1}]$ by $\langle \sigma \rangle$ and $[\sigma]$ respectively. When $n = 0$, $\langle \sigma \rangle \varphi$ and $[\sigma]\varphi$ denote φ. Finally, we define the *complexity* $|\varphi|$ of a TAPAL-formula φ by:

- $|p| = 0$ with p propositional.

- $|\neg\varphi| = |\Diamond\varphi| = |\Diamond^*\varphi| = |\varphi| + 1$
- $|\varphi \wedge \psi| = |\langle !\varphi\rangle\psi| = |\varphi| + |\psi| + 1.$

Definition 3.1.9 (Axiomatization TAPAL) The axiomatization TAPAL of TAPAL extends the axiomatization TPAL by the following axiom schemas and inference rules:

Axiom Schema

A2 $\langle !\chi\rangle\varphi \to \Diamond\varphi$ for any $!\chi \in \mathsf{PAL}$

A3 $\Diamond^*\varphi \leftrightarrow \varphi \vee \Diamond\Diamond^*\varphi$

Inference Rules

$R(\Box)$ If $\vdash \varphi \to [\sigma][!\top_0]\psi$, then $\vdash \varphi \to [\sigma]\Box\psi$, where $\top_0$ is a tautologous formula in $\mathcal{L}_{el}$ such that $!\top_0$ does not occur in φ or $[\sigma]\Box\psi$.

$R(\Box^*)$ If $\vdash \varphi \to [\sigma]\Box^k\psi$ for every k such that $0 \leq k \leq |\varphi|+1$, then $\vdash \varphi \to [!\sigma]\Box^*\psi$.

$\lhd$

Some remarks are in order about the axiomatization. First, **A2** expresses the fact that $\Diamond$ generalizes public announcement operators $\langle !\theta\rangle$. Second, A3 plays the role of Fixed Point Axiom as in PDL (See e.g. [10]), as can be seen by their schematic similarity.

Third, as we will discuss below (Corollary 3.6.7. See also Appendix 3.6.2), $R(\Box)$ is in fact equivalent to the following sound rule, which is a modification of the rule in the system APAL in [5]:

$R'(\Box)$ If $\vdash \varphi \to [\sigma][!p]\psi$ where p is in At such that $!p$ does not occur in φ or $[\sigma]\Box\psi$, then $\vdash \varphi \to [\sigma]\Box\psi$.

This form of the rule clarifies what the rule $R(\Box)$ is for. Observe the similarity between $R'(\Box)$ and the first-order rule:

FOQ If $\vdash \varphi \to \psi$ with no occurrence of x in φ, then $\vdash \varphi \to \forall x \psi$.

In fact, as we will see below in the completeness proof of TAPAL, the use of $R(\Box)$ is very similar to the use of this first-order rule in the completeness proof of first-order logic. Nonetheless, we chose $R(\Box)$ instead of $R'(\Box)$, since it extracts from the property of PAL-generated ETL models that they preserve truth over *model normalization* (Proposition 2.2.14) and the soundness proof becomes simpler when we appeal to the property.

Fourth, to see the role of $R(\Box^*)$, consider the following rule:

$R'(\Box^*)$ If $\vdash \varphi \to [\sigma]\Box^n\psi$ for all $n \geq 0$, then $\vdash \varphi \to [\sigma]\Box^*\psi$.

Given the semantic definition, it is straightforward to see that this infinitary rule is sound. The idea of our rule $R(\Box^*)$ is that we can extract a bound on n in the infinitary rule from the complexity $|\varphi|$ of the formula φ. (We will in fact use a more complicated notion of complexity, but it is bounded by the standard notion of complexity defined above. More on this in Appendix 3.6.3)

3.1.4 Soundness

The soundness of the axiom schemas and the necessitation rules are straightforward. Thus leaving the details of the proofs to the reader, we go on to sketch the soundness proofs of $R(\Box)$ and $R(\Box^*)$. The complete details are left to Appendix 3.6.

The Soundness of $R(\Box)$

To prove the soundness of $R(\Box)$, it suffices to show the following:

Theorem 3.1.10 (Soundness of $R(\Box)$) *If $\varphi \wedge \langle\sigma\rangle\Diamond\psi$ is satisfiable in $\mathbb{F}(\mathbb{PAL})$, then $\varphi \wedge \langle\sigma\rangle\langle !\top_0\rangle\psi$ is satisfiable in $\mathbb{F}(\mathbb{PAL})$, where $\top_0$ is a tautologous formula in $\mathcal{L}_{el}$ such that $!\top_o$ does not occur in φ or $\langle\sigma\rangle\Box\psi$.* $\lhd$

The idea of the proof can be sketched as follows. Suppose that $\varphi \wedge \langle\sigma\rangle\Diamond\psi$ is true at some h in $\mathcal{H}$. Then, φ is true at h. Also there is some $!\theta$ such that $h\sigma!\theta$ is in $\mathcal{H}$ and ψ is true at $h\sigma!\theta$. This situation is visualized in the left figure in Figure 3.4. We modify the model $\mathcal{H}$ by (i) taking the subtree starting from $h\sigma!\theta$ (the node labeled with ψ in the figure), (ii) obtaining the subtree with a new branch $!\top_0$ attached to its bottom, (iii) and *grafting* the new subtree to $h\sigma$ and the nodes connected to $h\sigma$ by indistinguishability relations in which $!\theta$ can happen. Let us denote the model obtained this way by $\mathcal{H}'$. $\mathcal{H}'$ is visualized in the right figure in Figure3.4. We claim that the formula $\varphi \wedge \langle\sigma\rangle\Diamond\psi$ is true at h in $\mathcal{H}'$. First, since TAPAL-formulas are 'future-looking', ψ is true at $h\sigma!\top_0$ by Proposition 3.1.6, since the structure of the new subtree is the same as the old subtree. Therefore, $\Diamond\psi$ is true at $h\sigma$, which implies that $\langle\sigma\rangle\Diamond\psi$ is true at h. Furthermore, the truth of φ is preserved over this transformation, since φ cannot distinguish the new and old subtrees by the assumption that $!\top_0$ does not occur in φ. Therefore, $\varphi \wedge \langle\sigma\rangle\Diamond\psi$ is indeed satisfiable.

All this can be made precise and the above claim can be proved to obtain the soundness of $R(\Box)$. The readers are invited to verify the details in Appendix 3.6.2.

The above soundness argument can be made when we replace $!\top_0$ by $!p_0$ when p_0 is a propositional letter such that $!p_0$ does not occur in φ or $[\sigma]\Box\psi$. We just have to adjust the valuation of p_0 appropriately so that it accord with the truth of θ in the above argument. Therefore, we can prove the soundness of $R'(\Box)$ mentioned above in a similar way. The following is an immediate consequence of these facts.

Corollary 3.1.11 *Let all $\varphi, \psi \in \mathcal{L}_{tapal}$ and $\sigma \in \mathsf{PAL}^*$. Also let $p_0, \top_0$ be a propositional letter and a tautologous formula in $\mathcal{L}_{el}$ such that neither $!p_0$ nor $\top_0$ occurs in*

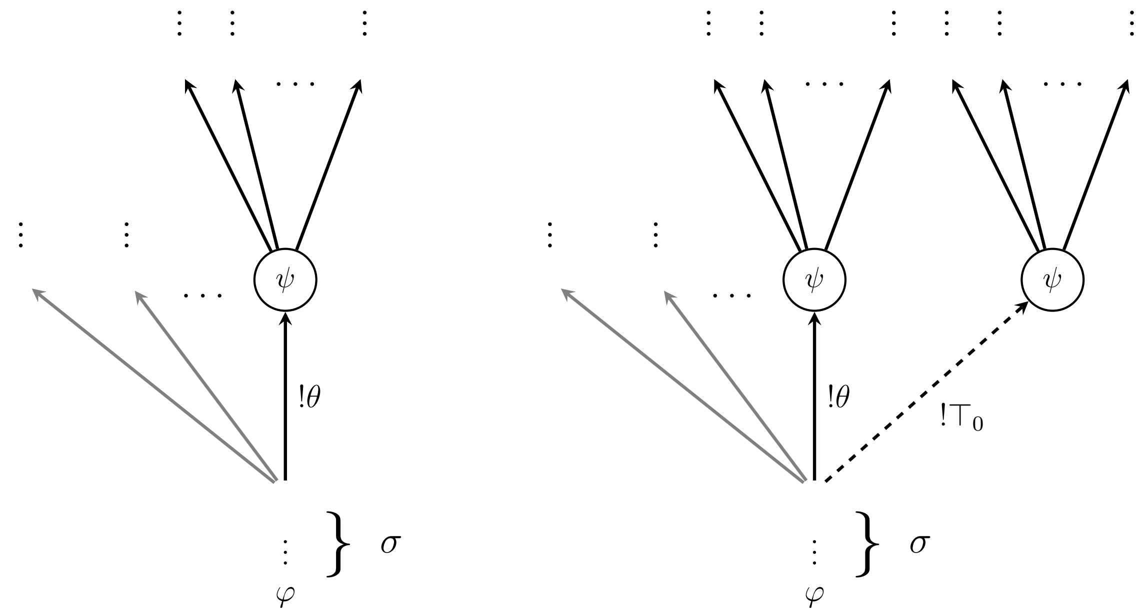

Figure 3.4: Grafting Subtrees

φ or $[\sigma]\Box\psi$.. Then

$$\vdash \varphi \rightarrow [\sigma][!p_0]\psi \;\Leftrightarrow \vdash \varphi \rightarrow [\sigma][!\top_0]\psi \;\Leftrightarrow \vdash \varphi \rightarrow [\sigma]\Box\varphi$$

◁

Proof. This follows immediate from the soundness of the rule $R'(\Box)$ and Theorem 3.6.6 via the semantic definition of $\Box$. QED

The benefit of using a tautologous formula $!\top_0$ is simply to do away with the process of adjusting valuation functions.

The Soundness of $R(\Box^*)$

As we mentioned above, the role of $R(\Box^*)$ can be most clearly seen by the infinitary version of the rule:

$R'(\Box^*)$ If $\vdash \varphi \to [\sigma]\Box^n\psi$ for all $n \geq 0$, then $\vdash \varphi \to [\sigma]\Box^*\psi$.

Given the semantic definition, it is straightforward to see that this infinitary rule is sound. The question then becomes how we can extract a bound on n in the infinitary rule from the complexity of the formula φ.

Let us start by observing the following fact. (The detailed proof of the fact is in Appendix 3.6.3.)

Proposition 3.1.12 (Reduction of $\Diamond^*$ to $\Diamond$) *For every $\varphi \in \mathcal{L}_{tapal}$, if $\Diamond^*\varphi$ is satisfiable in $\mathbb{F}(\mathbb{PAL})$, then $\Diamond^n\varphi$ is satisfiable in $\mathbb{F}(\mathbb{PAL})$ for $n = 0$ or $n = 1$.*

This proposition can be shown based on the following idea. If $\mathcal{H}, h \models \Diamond^*\varphi$, then there is some $\sigma \in \mathsf{PAL}^*$ such that $\mathcal{H}, h \models \langle\sigma\rangle\varphi$. By a similar argument given for Theorem 3.6.6, we take the subtree above h and form a new subtree with an appropriate $!\top_0$ attached to the bottom node. Then we graft it to h and the nodes related to h in which σ can happen. Since the structure of the new and old subtrees are the same, φ should be satisfied at $h!\top_0$. Consequently $\Diamond\varphi$ will be satisfied at h. This way, we can replace the sequence σ with $!\top_0$, so to speak.

In the light of this observation, it might be expected that the following claim holds:

Claim If $\varphi \wedge \langle\sigma\rangle\Diamond^*\psi$ is satisfiable in $\mathbb{F}(\mathbb{PAL})$, then $\varphi \wedge \langle\sigma\rangle\Diamond^n\psi$ is satisfiable in $\mathbb{F}(\mathbb{PAL})$ for $n = 0$ or $n = 1$.

Unfortunately this claim does not hold, due to the semantics of the $\Box$-operator. For simplicity, consider the case where σ is empty. Take $\Box\theta \wedge \Diamond^*\psi$. If this formula is satisfiable, then there will be a sequence τ after which ψ is satisfied. Here even if we appeal to the grafting method as in Proposition 3.6.8, we may not obtain the satisfiability of the whole formula $\Box\theta \wedge \Diamond^*\psi$. For the new node added to the model as a result of grafting must be quantified by $\Box$ in $\Box\theta$ and there is no guarantee that the node satisfies θ.

What this example illustrates is that, in general, the formula φ in $\varphi \wedge \langle\sigma\rangle\Diamond^*\psi$ may 'refer' to the nodes between the current node h and $h\tau$, where τ is a sequence of announcements, whose existence is claimed by $\Diamond^*$ in the formula. When this 'reference' is made by $\Box$, we cannot safely graft as we did for Proposition 3.6.8.

However how 'high up' in the tree φ can 'refer' can be read off from the complexity of φ. In particular, what is problematic is the occurrences of $\Box$ in φ and we need to know the highest number of nested occurrences of $\Box$ in φ.[3] Once we know such a number for φ, we can safely graft above the height that φ can refer to, as we did for Proposition 3.6.8. This way, we can put the bound on n in the infinitary rule based on the complexity of φ in $\varphi \wedge \langle\sigma\rangle\Diamond^*\psi$ to obtain $R(\Box)$.

All this can be made precise, but we will leave the complete details to Appendix 3.6.3. Here we state the theorem that we prove for the soundness of $R(\Box^*)$. Let $ibi(\varphi)$ be the critical number about the occurrences of $\Box$ in φ.

Theorem 3.1.13 *If $\varphi\wedge\langle\sigma\rangle\Diamond^*\psi$ is satisfiable in $\mathbb{F}(\mathbb{PAL})$, then $\varphi\wedge\langle\sigma\rangle\Diamond^k\psi$ is satisfiable in $\mathbb{F}(\mathbb{PAL})$ for some k such that $0 \leq k \leq ibi(\varphi)\dot{-}\mathsf{len}(\sigma) + 1$, where $a\dot{-}b = a - b$ if $a - b > 0$; $a\dot{-}b = 0$ otherwise.*

From this, we can immediately obtain the soundness of $R(\Box^*)$.

Corollary 3.1.14 *Soundness of $R(\Box^*)$ $R(\Box^*)$ is sound with respect to the class $\mathbb{F}(\mathbb{PAL})$.*

[3] This is measured in a suitable form of the formula. The basic idea is simply to get negation signs pushed in front of atomic formulas and then count the nested occurrences. See Appendix 3.6.3.

3.1.5 Completeness

Finally we prove the (weak) completeness of TAPAL. The basic idea of the proof is the same as the one give for TPAL in Section 2.3.2. However, some extra care must be taken for TAPAL, when we construct the canonical model. We need to construct the canonical model from the set of the maximal consistent sets Σ with the following properties.

Definition 3.1.15 (Saturation wrt $\Diamond$) A set Σ of formula is *saturated with respect to* $\Diamond$, if, for every sentence of the form $\langle\sigma\rangle\Diamond\varphi$ with $\sigma \in \mathsf{PAL}^*$, $\langle\sigma\rangle\Diamond\varphi \in \Sigma$ implies that there is some formula θ such that $\langle\sigma\rangle\langle!\theta\rangle\varphi \in \Sigma$. $\lhd$

Definition 3.1.16 (Saturation wrt $\Diamond^*$) A set Σ of formulas is *saturated with respect to* $\Diamond^*$, if, for every formula of the form $\langle\sigma\rangle\Diamond^*\varphi$ with $\sigma \in \mathsf{PAL}^*$, $\langle\sigma\rangle\Diamond^*\varphi \in \Sigma$ implies that there is some n such that $\langle\sigma\rangle\Diamond^n\varphi \in \Sigma$. $\lhd$

The motivation for these properties is to make sure that there are formulas that "witness" $\Diamond$ and $\Diamond^*$ in every formula in a given maximally consistent set. Here the analogy mentioned in the above remark (Section 3.1.9) between $R(\Box)$ and the first-order rule comes back again. In the proof below, when we construct a maximal consistent set from a consistent formula, we add witnessing formulas for the formulas of the above form. The consistency of the resulting set with witnessing formulas will be guaranteed by the rule $R(\Box)$, and this is very similar to the way that the first-order rule in question (or its equivalent) is used in the completeness proof of first-order logic. Similarly, $R(\Box^*)$ gives a witness for $\Diamond^*\varphi$ by finding an appropriate n for $\Diamond^n\varphi$ to be added, consistently, to a set, when we construct maximally consistent sets. These roles of the two rules are clear in the proof of the following lemma.

Lemma 3.1.17 (Lindenbaum Lemma) *Every consistent TAPAL-formula φ can be expanded to a maximal consistent set saturated with respect to $\Diamond$ and $\Diamond^*$.*

Proof. Let $\alpha_0, .\alpha_1...$ be an enumeration of the TAPAL-formulas such that $\alpha_0 = \varphi$. We construct a sequence $\Sigma_0, \Sigma_1, ...$ of sets as follows:

- $\Sigma_0 = \emptyset$
- If $\Sigma_n \cup \{\alpha_n\}$ is inconsistent, then $\Sigma_{n+1} = \Sigma_n$.
- If $\Sigma_n \cup \{\alpha_n\}$ is consistent and α_n is neither of the form $\langle\sigma\rangle\Diamond\psi$ nor of the form $\langle\sigma\rangle\Diamond^*\psi$, then $\Sigma_{n+1} = \Sigma_n \cup \{\alpha_n\}$.
- If $\Sigma_n \cup \{\alpha_n\}$ is consistent and α_n is of the form $\langle\sigma\rangle\Diamond\psi$, then $\Sigma_{n+1} = \Sigma_n \cup \{\langle\sigma\rangle\Diamond\psi, \langle\sigma\rangle\langle!\top_0\rangle\psi\}$ for a tautologous formula $\top_0$ in $\mathcal{L}_{el}$ such that $!\top_0$ does not occur in $\langle\sigma\rangle\Diamond\psi$ or any $\theta \in \Sigma_n$. Such a tautologous formula exists since Σ_n is finite and we have a countable number of tautologous formulas in $\mathcal{L}_{el}$.
- If $\Sigma_n \cup \{\alpha_n\}$ is consistent and α_n is of the form $\langle\sigma\rangle\Diamond^*\psi$, then take k such that $\Sigma_n \cup \{\langle\sigma\rangle\Diamond^*\psi, \langle\sigma\rangle\Diamond^k\psi\}$ is consistent and put $\Sigma_{n+1} = \Sigma_n \cup \{\langle\sigma\rangle\Diamond^*\psi, \langle\sigma\rangle\Diamond^k\psi\}$.

We show by induction that Σ_n is consistent for $n \geq 1$. The base case is given by the assumption that φ is consistent. Assume that Σ_n is consistent for an arbitrary n. Clearly it suffices to show the following claims:

Claim 1: Σ_{n+1} is consistent, if α_n is of the form $\langle\sigma\rangle\Diamond\psi$.

Claim 2: If $\Sigma_n \cup \{\alpha_n\}$ is consistent and α_n is of the form $\langle\sigma\rangle\Diamond^*\psi$, there is some m such that $\Sigma_n \cup \{\alpha_n, \langle\sigma\rangle\Diamond^m\psi\}$ is consistent.

Proof of Claim 1 Suppose Σ_{n+1} is inconsistent. Then, there must be some formulas $\psi_1, \psi_2, ..., \psi_l \in \Sigma_m \cup \{\langle\sigma\rangle\Diamond\psi\}$ such that

$$\vdash (\psi_1 \wedge ... \wedge \psi_l) \rightarrow \neg\langle\sigma\rangle\langle!\top_0\rangle\psi.$$

However, this implies

$$\vdash (\psi_1 \wedge ... \wedge \psi_l) \to [\sigma][!\top_0]\neg\psi.$$

Since $\top_0$ is chosen so that $\top_0$ does not occur in $[\sigma]\Box\psi$ or any $\theta \in \Sigma_n$, we can apply $R(\Box)$ to obtain

$$\vdash (\psi_1 \wedge ... \wedge \psi_l) \to [\sigma]\Box\neg\psi$$

This gives us $\Sigma_m \vdash [\sigma]\Box\neg\psi$ and $\Sigma_m \vdash \neg\langle\sigma\rangle\Diamond\psi$. However this contradicts the assumption that $\Sigma_n \cup \{\alpha_n\}$ is consistent.

Proof of Claim 2: Suppose toward contradiction that there is no such m. Then, for all $m \geq 0$, we have:

$$\vdash \bigwedge \Sigma_n \to \neg\langle\sigma\rangle\Diamond^m\psi.$$

where $\bigwedge \Sigma_n$ is a conjunction of the formulas in Σ_{m-1}. This implies that, for all m,

$$\vdash \bigwedge \Sigma_n \cup \{\alpha_n\} \to [\sigma]\Box^m\neg\psi$$

and by $R(\Box^*)$

$$\vdash \bigwedge \Sigma_n \cup \{\alpha_n\} \to [\sigma]\Box^*\neg\psi.$$

Therefore, we have $\Sigma_n \cup \{\alpha_n\} \vdash [\sigma]\Box^*\neg\psi$ and thus $\Sigma_n \cup \{\alpha_n\} \vdash \neg\langle\sigma\rangle\Diamond^*\psi$. This contradicts our assumption that $\Sigma_n \cup \{\langle\sigma\rangle\Diamond^*\psi\}$ is consistent.

Now take $\Sigma' = \bigcup_{i=0}^{\infty} \Sigma_i$. The maximality and saturation with respect to $\Diamond$ and $\Diamond^*$ is clear by the construction. The consistency is shown in the standard way by the consistency of Σ_n for $n \geq 1$. QED

Having this lemma, we can construct the canonical model from the set of maximally consistent sets that are saturated with respect to $\Diamond$ and $\Diamond^*$ in the same was as Section 2.3.2. The saturation properties of maximally consistent sets are needed to prove the truth lemma for the canonical model. Before proving the truth lemma, we need the following proposition.

Proposition 3.1.18 *Let* $\sigma \in \mathsf{PAL}^*$ *and* $\mathsf{len}(\sigma) = n$. *Then,*

1. $\vdash \langle\sigma\rangle\varphi \to \Diamond^n\varphi$.
2. $\vdash \Diamond^n\varphi \to \Diamond^*\varphi$.

Proof. Straightforward. The proof for the second appeals to the axiom A3. QED

Let $\mathcal{G}_{can}$ be the canonical model constructed as in Section 2.3.2.

Lemma 3.1.19 *(Truth Lemma) For every formula* $\varphi \in \mathcal{L}_{tapal}$,

$$\varphi \in \lambda(h) \quad \textit{iff} \quad \mathcal{G}_{can}, h \models \varphi.$$

Proof.: The proof is by induction on φ. We only give the cases for $\Diamond$ and $\Diamond^*$. The argument for the other cases are given in the proof of Lemma 2.3.7.

Assume that φ is of the form $\Diamond\psi$. First assume that $\Diamond\psi \in \lambda(h)$. Given the construction of the canonical model in Section 2.3.2, each $\lambda(h)$ is maximally consistent set and clearly saturated with respect to $\Diamond$ and $\Diamond^*$. Therefore, we have $\langle !\theta\rangle\psi \in \lambda(h)$ for some θ. By the construction of $\mathcal{G}_{can}$, we have $\psi \in \lambda(h!\theta)$. By IH, we obtain $\mathcal{G}_{can}, h!\theta \models \psi$. Therefore, we have $\mathcal{G}_{can}, h \models \Diamond\psi$ by truth definition.

For the other direction, assume that $\mathcal{G}_{can}, h \models \Diamond\psi$. By definition, there is some θ such that $h!\theta \in G_{can}$ and $\mathcal{G}_{can}, h!\theta \models \psi$. By IH, we have $\psi \in \lambda(h!\theta)$, which, by the construction of $\mathcal{G}_{can}$, implies $\langle !\theta\rangle\varphi \in \lambda(h)$. This implies by A2 that $\Diamond\varphi \in \lambda(h)$.

Next, assume that $\Diamond^*\psi \in \lambda(h)$. Since $\lambda(h)$ is a maximally consistent set saturated with respect to $\Diamond^*$, there is some $k \geq 0$ such that $\Diamond^k\psi \in \lambda(h)$ Now, since $\lambda(h)$ is also saturated with respect to $\Diamond$, we have $\langle !\theta_1\rangle \ldots \langle !\theta_k\rangle\psi \in \lambda(h)$. Thus, by the construction of canonical model, we have $\psi \in \lambda(h!\theta_1 \ldots !\theta_k)$, which implies by IH that $\mathcal{G}_{can}, h!\theta_1 \ldots !\theta_k \models \psi$. This gives us $\mathcal{G}_{can}, h \models \Diamond^*\psi$.

Assume that $\mathcal{G}_{can}, h \models \Diamond^*\psi$. By definition, this is equivalent to saying that there is some σ such that $h\sigma \in \mathcal{G}_{can}$ and $\mathcal{G}_{can}, h\sigma \models \psi$. By IH, we have $\psi \in \lambda(h\sigma)$, which, by the construction of λ, implies $\langle\sigma\rangle\psi \in \lambda(h)$. By Proposition 3.1.18, we have that $\Diamond^*\psi \in \lambda(h)$. QED

The rest of the argument is similar to Section 2.3.2. Therefore, we obtain:

Theorem 3.1.20 (Completeness) *TAPAL is weakly complete with respect to* $\mathbb{F}(\mathbb{PAL})$.

3.2 Describing Past

So far we have considered only 'future-looking' operators, $\langle\epsilon\rangle$, $\Diamond$, and $\Diamond^*$. The semantic definitions of these operators only depend on what *is* or *will be* true in given models and, in this sense, our language did not provide a way to describe what *was* the case in the past. This is because we have reinterpreted the language of DEL with our models in the first place. DEL captures the temporal transition of informational states by product update and, once models are updated, the information about the previous models is (at least partially) lost. For this reason, the language of DEL, admittedly, is limited to the descriptions about what will happen after informational events happen, but not what was the case in the past. However, in our framework, DEL-generated ETL models have the forest structures that encode all successive stages of update by event models. With the temporal structures, we can naturally think about the operator that states what *was* the case prior to a given temporal point. Indeed, languages of ETL often contain the operators that describe prior temporal points in time-branching tree structures. These considerations motivate us to extend our DEL-based language and consider past-operators in our framework. In this section, we will consider the extension of TPAL with past-operators.

3.2.1 TPAL with Labelled Past Operators

Fix a set of agents $\mathcal{A}$ and a countable set of propositional letter At.

Definition 3.2.1 (Language of TPAL+P) The language $\mathcal{L}^{p}_{tpal}$ of TPAL+P extends $\mathcal{L}_{tpal}$ with the operators $P_{!\theta}$. The formulas in $\mathcal{L}^{p}_{tpal}$ is inductively defined by:

$$\varphi \ ::= \top \mid p \mid \neg\varphi \mid \varphi \wedge \varphi \mid [i]\varphi \mid \langle !\theta \rangle\varphi \mid P_{!\theta}\varphi$$

where $p \in \mathsf{At}$, $i \in \mathcal{A}$ and $\theta \in \mathcal{L}_{el}$. The dual $\hat{P}_{!\theta}$ of the operator $P_{!\theta}$ is defined in the standard way. The other operators are defined as mentioned in Definition 2.1.2. $\triangleleft$

The intended reading of $P_{!\theta}\varphi$ is "the public announcement $!\theta$ has been made, before which φ". The intended reading of the dual $\hat{P}_{!\theta}\varphi$ is as "Before the public announcement $!\theta$ has been made, φ." However, this reading should be not taken as implying that $!\theta$ has in fact happened. If the announcement $!\theta$ did not happen, we define $\hat{P}_{!\theta}$ vacuously true. We call the resulting system *Temporal Public Announcement with the Labelled Past Operators* and denote by *TDEL+P*.

Definition 3.2.2 (Truth) Given $\mathcal{H} = (H, \sim, V) \in \mathbb{F}(\mathbb{PAL})$ and a history $h \in H$, the truth of formulas in TPAL+P is inductively defined as follows. We only give the definitions for $P_{!\theta}$. The other truth definitions are as given in Definition 2.1.5:

$$\mathcal{H}, h \models P_{!\theta}\varphi \quad \text{iff} \quad \exists h' \text{ such that } h = h'!\theta \text{ and } \mathcal{H}, h' \models \varphi.$$

Consistency, satisfiability, validity etc. are defined in the standard way as in Definition 2.1.6. ◁

3.2.2 Semantic Results

Next, we make some simple semantic observations in TPAL+P and prove the normalization theorem for TPAL, as we need it for the axiomatization of TPAL+P. First the following proposition is straightforward to verify based on the truth definition in TPAL+P.

Proposition 3.2.3 (Validities) *The following validities obtain in TPAL+P.*

1. $\models \langle !\theta \rangle P_{!\theta}\varphi \leftrightarrow \langle !\theta \rangle \top \wedge \varphi$.
2. $\models \neg \langle !\theta' \rangle P_{!\theta} \top$, *where* $\theta' \neq \theta$.
3. $\models [!\theta][i]P_{!\theta}\theta$.

Item 3 is contrastive to the *in*validity of the formula $[!\theta][i]\theta$ (After the announcement of $!\theta$, i knows θ). $[!\theta][i]\theta$ is invalid in the presence of the formulas that become false after being publicly announced. The prime example of such a formula is $p \wedge \neg [i]p$. Although agents cannot *know* what has become false, they can always know that it *was* true before it is announced. This is what Item 3 says.

Next, let us observe that the past-operator adds expressive power to TPAL. Denote by $!\top^k$ the sequence of k $!\top$'s. Given an epistemic model $\mathcal{M}$ and a world w in $\mathcal{M}$, define p be such that$\mathsf{p}(w) = \{!\top^k | 0 \leq i \leq d(\psi)\}$. Now consider $\mathcal{H} = \mathsf{Forest}(\mathcal{M}, \mathsf{p})$, which is visualized in Figure 3.5. Since TPAL does not describe the past, every formula true at a node in the model is true at another. (Indeed any subtree of the model is isomorphic to the whole model.) However, in TPAL+P, we can distinguish each point of the model, say $w!\top^k$, by the formula $P^k_{!\top}\top$, where $P^k_{!\theta}$ denote k consecutive occurrences of $P_{!\theta}$.

$$\mathcal{H}: \quad w \longrightarrow w!\top \longrightarrow \cdots \longrightarrow w!\top^k \longrightarrow \cdots$$

Figure 3.5: Expressivity of TPAL+P

Proposition 3.2.4 (Expressivity) *TPAL+P is strictly more expressible than TPAL.*

Finally the normalization result similar to Proposition 2.2.14 can be obtained for TPAL+P, which we will need for our axiomatization of TPAL+P. Recall that, given a model $\mathcal{H} \in \mathbb{F}(\mathsf{PAL})$ and a history h in $\mathcal{H}$, $\mathcal{H}[!\top_0/!\varphi_0, !\top_1/!\varphi_1, \ldots]$ and $h[!\top_0/!\varphi_0, !\top_1/!\varphi_1, \ldots]$ were the model and history obtained by replacing all occurrences of φ_i with $\top_i$ in $\mathcal{H}$. (See Definition 2.2.10) Also recall that, given AOC let us extend the notion of *announcement occurrence set* (Definition 2.2.6) to TPAL+P by the clause:

- $AOC(P_{!\theta}\varphi) = \{!\theta\} \cup AOC(\varphi)$

so that $AOC(\varphi)$ in general yields the set of public announcements occurring in φ.

Proposition 3.2.5 (Normalization) *Let* $\mathcal{H} = \mathsf{Forest}(\mathcal{M}, \mathsf{p}) \in \mathbb{F}(\mathbb{PAL})$. *Let* X *be a finite subset of* PAL. *Furthermore, let* $!\varphi_0, !\varphi_1 \ldots$ *be an enumeration of public announcements in* $\mathsf{PAL} \backslash X$ *without repetition, and* $!\top_0, !\top_1, \ldots$ *be an enumeration of tautologous public announcements in* $\mathsf{PAL} \backslash X$ *without repetition. Then, for every* h *and a formula* $\varphi \in \mathcal{L}^p_{tpal}$ *such that* $AOC(\varphi) \subseteq X$,

$$\mathcal{H}, h \models \varphi \quad \Leftrightarrow \quad \mathcal{H}[!\top_0/!\varphi_0, !\top_1/!\varphi_1, \ldots], h[!\top_0/!\varphi_0, !\top_1/!\varphi_1, \ldots] \models \varphi$$

Proof. The proof is by induction on φ. The cases other than P_ϵ are as in Proposition 2.2.14. Thus, assume $\mathcal{H}, h \models P_{!\theta}\psi$. Then there must be some h' such that $h = h'!\theta$ and $\mathcal{H}, h' \models \psi$. By the IH,

$$\mathcal{H}[!\top_0/!\varphi_0, !\top_1/!\varphi_1, \ldots], h'[!\top_0/!\varphi_0, !\top_1/!\varphi_1, \ldots] \models \psi.$$

Since $!\theta \in X$,

$$(h'\theta)[!\top_0/!\varphi_0, !\top_1/!\varphi_1, \ldots] = h\prime[!\top_0/!\varphi_0, !\top_1/!\varphi_1, \ldots]\epsilon.$$

Thus,

$$\mathcal{H}[!\top_0/!\varphi_0, !\top_1/!\varphi_1, \ldots], h'!\theta[!\top_0/!\varphi_0, !\top_1/!\varphi_1, \ldots] \models P_{!\theta}\psi.$$

The other direction is similar. QED

3.2.3 Axiomatization

To present the axiomatization of TDEL+P, we need some definitions.

Definition 3.2.6 (Past-Depth) Given a formula φ, the *past-depth* $pd(\varphi)$ of the formula φ is defined as follows:

- $pd(p) = 0$ for p propositional.
- $pd(\neg\varphi) = d(\varphi)$
- $pd(\varphi \wedge \psi) = max\{d(\varphi), d(\psi)\}$
- $pd([i]\varphi) = d(\varphi)$
- $pd(\langle !\theta \rangle \varphi) = d(\varphi) - 1$
- $pd(P_{!\theta}\varphi) = \max(d(\varphi), 0) + 1$

◁

The intuition behind this definition is that if a formula has a past-depth n, we would have to go n-steps into the past from the current point of the ETL-tree in order to verify it. Thus, the final clause reflects the intended meaning. Had the definition instead been $pd(P_{!\theta}\varphi) = d(\varphi) + 1$, this would not have worked, $P_{!\theta_1}\langle !\theta_2\rangle\langle !\theta_3\rangle p$. That definition would mistakenly have set the past-depth as -1 instead of 1.

Definition 3.2.7 (Axiomatization of TPAL+P) The axiomatization of TPAL+P extends the axiomatization TPAL (Definition 2.3.1) with the following axiom schemas and inference rule.

Axioms

$P!\mathbf{K}$ $\quad \hat{P}_{!\theta}(\varphi \to \psi) \to (\hat{P}_{!\theta}\varphi \to \hat{P}_{!\theta}\psi)$

P1 $\quad \langle !\theta\rangle P_{!\theta}\varphi \leftrightarrow \langle !\theta\rangle \top \wedge \varphi$

P2 $\quad \langle !\theta\rangle P_{!\theta'}\varphi \to \bot$ if $!\theta \neq !\theta'$

Inference Rules

$P!\mathbf{N}$ $\quad$ If $\vdash \varphi$, then $\vdash \hat{P}_{!\theta}\varphi$ for $!\theta \in$ PAL.

$\mathbf{R}(P)$ $\quad$ If $\vdash [!\theta_1]\dots[!\theta_{pd(\varphi)}]\varphi$ for every $!\theta_1, \dots, !\theta_{pd(\varphi)}$ such that each $!\theta_i$ is in $AOC(\varphi)$ or in $!T(\varphi)$, then $\vdash \varphi$, where $!T(\varphi)$ is a set of $pd(\varphi)$ tautologous public announcements not in $AOC(\varphi)$.

$\triangleleft$

3.2.4 Soundness Proof

P1 and **P2** correspond to Item 1 and 2 in Proposition 3.2.3. The rule $\mathbf{R}(P)$ is equivalent to the following statement.

Lemma 3.2.8 *If φ is satisfiable in $\mathbb{F}$(PAL), then $\langle !\theta_1\rangle \dots \langle !\theta_{pd(\varphi)}\rangle\varphi$ is satisfiable in $\mathbb{F}$(PAL) for some $!\theta_1, \dots, !\theta_{pd(\varphi)} \in AOC(\varphi) \cup !T(\varphi)$, where $!T(\varphi)$ is a set of $pd(\varphi)$ tautologous public announcements not in $AOC(\varphi)$.*

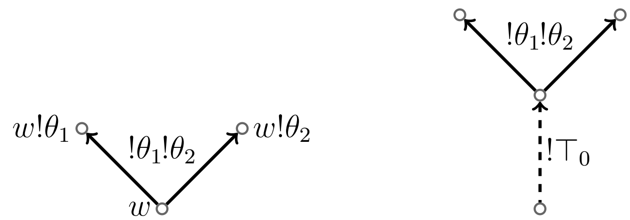

Figure 3.6: Lifting Histories by $!\top$.

The basic idea of the proof will be as follows. Assuming $\mathcal{H}, h \models \varphi$, we first apply the normalization method based on Proposition 3.2.5. Then, if φ is satisfied in the model at a sufficiently long history (i.e. strictly longer than $d(\varphi)$), then we can satisfy $\langle !\theta_1 \rangle \dots \langle !\theta_{d(\varphi)} \rangle \varphi$ by tracing down the history, using the truth definition of the future operator. By adjusting the normalization at the beginning, we can make each $!\theta_i$ of the form specified in $\mathsf{R(X)}$.

However, if φ is satisfied at a history that is not long enough, then we construct a new model from $\mathcal{H}$ by 'lifting' the root nodes of the trees in $\mathcal{H}$ with the sequence of tautologous public announcements $\tau =!\top_1 \dots !\top_{pd(\varphi)}$. The new model preserves the structures above the sequence τ, since applying tautologous public announcements (uniformly at every world) keeps the structure of the original model unchanged. In addition, the added sequence τ cannot be 'referred' by the formula φ, since each $!\top_i$ in τ is a public announcement that does not occur in φ.

To illustrate this, consider the evaluation of the formula $\varphi = P_{!\theta_1} \neg P_{!\theta_2} \top$, with past-depth 2. On the left model in Figure 3.6, φ is satisfied at world $w!\theta_1$, where $\mathsf{len}(w!\theta_1) = 2$. To obtain a length of 3 (as would be required for the soundness claim) for the history at which φ is satisfied, we lift the model by a public announcement $!\top_0$, which does not occur in φ. The model obtained by this operation is visualized by the right figure in Figure 3.6.

To give the soundness proof based on the above idea, we need to recall some definitions. Let p be a state-dependent protocol on $\mathcal{M}$. Given $\sigma \in \mathsf{PAL}^*$, we define a local protocol $\mathsf{p}_{\sigma<}$ on $\mathcal{M}^{\sigma,\mathsf{p}}$ so that, for all $v\tau \in Dom(\mathcal{M}^{\sigma,\mathsf{p}})$, $p_{\sigma<}(v\tau) = \{\rho \mid v\tau\rho \in \mathsf{p}(w)$ where $w \in Dom(\mathcal{M})\}$. Given an ETL model $\mathsf{Forest}(\mathcal{M}, \mathsf{p})$ and a sequence σ, the model $\mathsf{Forest}(\mathcal{M}^{\sigma,\mathsf{p}}, \mathsf{p}_{\sigma<})$ can be seen as a submodel of $\mathsf{Forest}(\mathcal{M}, \mathsf{p})$ that describes what will happen in $\mathsf{Forest}(\mathcal{M}, \mathsf{p})$ after the sequence σ of events have happened. Now we prove Lemma 3.4.3.

Proof. Let $\mathcal{M} = (W, \sim, V)$ be an epistemic model and p, a state-dependent protocol on $\mathcal{M}$. Put $\mathcal{H} = \mathsf{Forest}(\mathcal{M}, \mathsf{p})$. Assume $\mathcal{H}, h \models \varphi$. Suppose $\mathsf{len}(h) \geq pd(\varphi) + 1$. Let Σ_0 be the set of the last $pd(\varphi)$ elements of the sequence h that are not in $AOC(\varphi)$. Then apply Proposition 3.2.5 so that the elements in Σ_0 are replaced by elements in $!T(\varphi)$. This can be done, since $|\Sigma_0| \leq |!T(\varphi)|$ by definition. Denoting by $\mathcal{H}'$ and h' respectively the obtained normalization of $\mathcal{H}$ and the element in $\mathcal{H}'$ corresponding to h, we have:

$$s\mathcal{H}', h' \models \varphi.$$

Now, since $\mathsf{len}(h') \geq dp(\varphi) + 1$ by our assumption, we obtain:

$$\mathcal{H}', h'_{\mathsf{len}(h')-dp(\varphi)} \models \langle h'_{\mathsf{len}(h')-dp(\varphi)+1}\rangle \dots \langle h'_{\mathsf{len}(h')}\rangle\varphi.$$

Since each of $h'_{\mathsf{len}(h')-pd(\varphi)+1}, \dots, h'_{\mathsf{len}(h')}$ is either in $AOC(\varphi)$ or $!T(\varphi)$ by our construction, we are done in this case.

Next, suppose $\mathsf{len}(h) < pd(\varphi) + 1$. Put $k = pd(\varphi) - \mathsf{len}(h) + 1$. Let $!\top_0$ be an element in $!T(\varphi)$. Also denote by $!\top_0^k$ the sequence of k $!\top_0$'s. Construct a state-dependent protocol p^+ on $\mathcal{M}$ so that $\mathsf{p}^+(w)$ is the set obtained by taking the closure under finite prefix on $\{!\top_0^k\sigma \mid \sigma \in \mathsf{p}(w)\}$. Then, by the construction, for all σ (possibly empty):

$$\mathsf{Forest}(\mathcal{M}^{!\top_0^k,\mathsf{p}^+}, \mathsf{p}^+_{!\top_0^k<}), (w!\top_0^k)\sigma \models \varphi \text{ iff } \mathsf{Forest}(\mathcal{M}, \mathsf{p}), w\sigma \models \varphi$$

where w is in $\mathcal{M}$. We would like to show, for all σ,

$$\mathsf{Forest}(\mathcal{M}^{!\top_0^k,\mathsf{p}^+}, \mathsf{p}^+_{!\top_0^k<}), (w!\top_0^k)\sigma \models \varphi \text{ iff } \mathsf{Forest}(\mathcal{M}, \mathsf{p}^+), w!\top_0^k\sigma \models \varphi.$$

Once this is shown, we can argue as above in the case $\mathsf{len}(h') \geq pd(\varphi) + 1$. Indeed, if $h = w\sigma$, then $\mathsf{len}(w!\top_0^k\sigma) = d(\varphi) + 1$.

To show this, we prove the following general claim: for all σ and formulas ψ such that $!\top_0$ does not occur in ψ,

$$\mathsf{Forest}(\mathcal{M}^{!\top_0^k,\mathsf{p}^+}, \mathsf{p}^+_{!\top_0^k<}), (w!\top_0^k)\sigma \models \psi \text{ iff } \mathsf{Forest}(\mathcal{M}, \mathsf{p}^+), w!\top_0^k\sigma \models \psi.$$

The proof is by a straightforward induction on the complexity of ψ. We will only do the past-modality case. Suppose that ψ is of the form $P_{!\theta}\chi$. Assume that σ is empty. By our assumption, $!\theta \neq !\top_0$. Therefore, the RHS and LHS of the biconditional are simply false. Next, assume that σ is non-empty. Further suppose the LHS of the biconditional. Then the last element of σ, $\sigma_{\mathsf{len}(\sigma)}$, must be $!\theta$ by the truth definition of the past-operator. Thus we have

$$\mathsf{Forest}(\mathcal{M}^{!\top_0^k,\mathsf{p}^+}, \mathsf{p}^+_{!\top_0^k<}), w!\top_0^k\sigma_{(\mathsf{len}(\sigma)-1)} \models \chi.$$

By IH, this is equivalent to

$$\mathsf{Forest}(\mathcal{M}, \mathsf{p}^+), w!\top_0^k\sigma_{(\mathsf{len}(\sigma)-1)} \models \chi.$$

Therefore, we have

$$\mathsf{Forest}(\mathcal{M}, \mathsf{p}^+), w!\top_0^k\sigma_{(\mathsf{len}(\sigma))} \models P_{!\theta}\chi.$$

This completes the proof. QED

3.2.5 Completeness Proof

The basic idea of the completeness proof for TPAL+P is the same as the completeness proof for TPAL. The canonical model is constructed from the set of maximally consistent set as in Section 2.3.2. Then we prove the truth lemma stated as follows:

Lemma 3.2.9 (Truth Lemma) *Let $\mathcal{H}^{can}$ be the canonical model. For every formula φ in* TPAL+P *and $h \in \mathcal{H}^{can}$ such that* $\mathsf{len}(h) > pd(\varphi)$,

$$\varphi \in \lambda(h) \text{ iff } \mathcal{H}^{can}, h \models \varphi$$

Proof. We will only consider the past modality case. The other cases are given in the same way as the proof of Lemma 2.3.7.) Let $h = h'!\theta$ for some $\mathsf{len}(h) \geq pd(\varphi) + 1$, where $!\theta \in \mathsf{PAL}$. Let φ be of the form $P_{!\chi}\psi$.

Assume then that $P_{!\chi}\psi \in \lambda(h)$. By the definition of canonical model, $\langle !\theta \rangle P_{!\chi}\psi \in \lambda(h')$. If $!\theta \neq !\chi$, then by **P2**, $\bot \in \lambda(h')$, which contradicts the consistency of $\lambda(h')$. Thus, assume $!\theta = !\chi$. Then, by **P1**, we have $\psi \in \lambda(h')$. By the IH, $\mathcal{H}^{can}, h' \models \psi$ (note $\mathsf{len}(h') \geq pd(\psi) + 1$). Since $h'!\theta \in \mathcal{H}^{can}$ and $!\theta = !\chi$, the truth definition implies that $\mathcal{H}^{can}, h \models P_{!\chi}\psi$.

For the other direction, assume that $\mathcal{H}^{can}, h \models P_{!\chi}\psi$. By truth definition, we have $!\theta = !\chi$, and also $\mathcal{H}, h' \models \psi$. By the IH, we have $\psi \in \lambda(h')$. And by the construction of the canonical model, we have $\langle !\theta \rangle \top \in \lambda(h')$. Thus, by **P1**, we have $\langle !\theta \rangle P_{!\chi}\psi \in \lambda(h')$, which by construction implies that $P_{!\chi}\psi \in \lambda(h)$. QED

We can also prove the lemma corresponding to Lemma 2.3.8 in order to guarantees that the canonical model is in the class $\mathbb{F}(\mathbb{PAL})$ of PAL-generated ETL models. Now, we cannot conclude the completeness immediately from this, since we are not sure yet that, given a formula of past-depth n, we have a maximal consistent set that contains φ, which is assigned to a history long enough to apply the truth lemma. That is where we need to appeal to the rule $\mathbf{R}(P)$.

Theorem 3.2.10 TPAL+P(X) *is complete with respect to* $\mathbb{F}(\mathbb{PAL})$.

Proof. Let φ be consistent. Then $\langle !\theta_1\rangle \dots \langle !\theta_{pd(\varphi)}\rangle\varphi$ is consistent for some sequence of $!\theta_1, \dots, !\theta_{pd(\varphi)}$ as specified in the rule $\mathbf{R}(P)$. For suppose otherwise. Then for every such $!\theta_1, \dots, !\theta_{pd(\varphi)}$, $\langle !\theta_1\rangle \dots \langle !\theta_{pd(\varphi)}\rangle\varphi$ is inconsistent and thus $\vdash [!\theta_1] \dots [!\theta_{pd(\varphi)}]\neg\varphi$. By $\mathbf{R}(P)$, $\vdash \neg\varphi$. This contradicts the consistency of φ. Thus $\langle !\theta_1\rangle \dots \langle !\theta_{pd(\varphi)}\rangle\varphi$ is consistent for some $!\theta_1 \dots !\theta_{pd(\varphi)}$. Take such a formula $\langle !\theta_1\rangle \dots \langle !\theta_{pd(\varphi)}\rangle\varphi$. Since the formula is consistent, by Lindenbaum's Lemma, we have a maximally consistent set containing it. Furthermore, note that $pd(\theta) = 0$. Thus, by the truth lemma, there is some history h of length 1 such that

$$\mathcal{H}^{can}, h \models \langle !\theta_1\rangle \dots \langle !\theta_{pd(\varphi)}\rangle\varphi$$

This gives us the result that $\mathcal{H}^{can}, h!\theta_1 \dots !\theta_{pd(\varphi)} \models \varphi$. QED

3.3 Announcements about Announcements

Next we will discuss the extension of PAL-generated ETL models. In general, DEL-generated ETL models are constructed from epistemic models based on DEL-protocols. DEL-protocols consist of sets of finite sequences of (pointed) event models and event models as defined in Chapter 1 (Definition 1.2.4) have preconditions expressed by epistemic formulas. This restriction does not seem substantial in DEL, since every DEL-formulas are equivalent to epistemic formulas via reduction axioms. However, such a reduction is not available in our framework and thus the restriction in our context can be an obstacle when we apply the framework to describe intelligent interaction. Indeed, why can we not model public announcements, say, about future truths that obtain after public announcements? Why can preconditions of events depend on future truths? In this section, we will tackle this problem in the context of TPAL.

To lift the assumption, we need to generalize the construction of PAL-generated ETL models.

3.3.1 Higher-Order Public Announcements

Let us start by seeing difficulties in extending our model construction method beyond epistemic formulas. Suppose we extend the notion of PAL-protocols so that they can contain formulas with public announcements. For instance, let p be an *sd*-PAL protocol on a given epistemic model $\mathcal{M}$ in this extended sense. Assume that the protocol assigned by p at a given world w in $\mathcal{M}$ indeed contains $!(\langle !p\rangle\langle !q\rangle\top)$. To determine whether we construct the node $w!(\langle !p\rangle\langle !q\rangle\top)$, that is, whether $!(\langle !p\rangle\langle !q\rangle\top)$ is announceable, we need to know whether the formula is true at w. However, to determine whether the formula is true at w, we need to know in advance whether p is true at w and whether p allows $!p$ at w. Moreover, we need to know whether q is true at $w!p$ and whether p allows $!q$ after $!p$ at w (if the node $w!p$ is generated). Unless these things are known in advance, we cannot determine whether $!(\langle !p\rangle\langle !q\rangle\top)$ is announceable at w. As this example illustrates, once we lift the restriction and allow formulas with public announcements in protocols, we cannot simply generate ETL-trees straight up from the bottom epistemic model, as we did in the previous chapters.

The main difficulty here can be summarized by the following points. First, if φ contains announcement operators, thus making $!\varphi$ a "higher-order" public announcement about the "lower-order" public announcements contained in $!\varphi$, then we need to know in advance about the announceability of the lower-order announcements in order to determine the announceability of $!\varphi$. To determine whether $!\langle !p\rangle\top$ is announceable, we need to know whether $!p$ is announceable in the first place. Second, $!\varphi$ may 'refer' to the announceability of lower-order public announcements in the *future* as well as the current announceability of lower-order public announcements.

In our example, to determine whether $!\langle !p\rangle\langle !q\rangle\top$ is announceable *currently*, we need to know whether $!p$ is *currently* announceable *and* whether $!q$ *will* be announceable after $!p$ is announced. Therefore, we need to know in advance the announceability of relevant sequences of announcements in order to determine the announceability of $!\varphi$. This means that we need to know the structure of the tree above the current node at least concerning the lower-order announcements mentioned in φ, whereas we only needed to know the structure of the 'current stage' in the model construction developed in the previous chapters.

Therefore, we need to generalize the construction of PAL-generated ETL models to take into account higher order announcements. The key idea is to construct ETL structures by induction on the *orders* of announcements occurring in the announcement sequences. Having an epistemic model $\mathcal{M}$ and a protocol p, we first construct ETL-trees from $\mathcal{M}$ by the above construction method, based on the initial segments of the sequences given by p that consist only of epistemic formulas. That is, we begin the construction by dealing with only the "first-order" announcements. Then we add nodes to the resulted trees, based on the second-order announcements that refer to the first-order announcements. This second construction process can be carried out, since the truth values of the formulas we need to know to make the second-order announcements, i.e. the ones of first-order announcements, will have been determined at this point after the first-order construction process. We then continue this way by constructing nodes of second-order announcements after nodes for first-order announcements, until all first- and second-order announcements are taken care of. Then we next goes on to the third-order announcement process and continue similarly. And so forth. Below we make this idea more precise.

3.3.2 Generalization of PAL-Generated ETL Models

To extend PAL-generated ETL models, we need to extend the notion of protocols. We start by specifying our language. Fix a set of agents $\mathcal{A}$ and a countable set of propositional letters At.

Definition 3.3.1 (Language of TPAL$^+$) The language $\mathcal{L}^+_{tpal}$ of TPAL$^+$ is the language of PAL and, thus, the formulas in $\mathcal{L}^+_{tpal}$ is inductively defined by:

$$\varphi ::= \top \mid p \mid \neg\varphi \mid \varphi \wedge \varphi \mid [i]\varphi \mid \langle !\varphi \rangle \varphi$$

where $p \in \mathsf{At}$ and $i \in \mathcal{A}$. The other operators are defined as mentioned in Definition 2.1.2. $\lhd$

Note that we finally drop the restriction. φ in $!\varphi$ can be now any PAL-formula and contain public announcements. Next we redefine the notion of protocols and relevant notations accordingly.

Definition 3.3.2 (PAL-Protocol Extended) Let PAL^+ be the set i.e. $\{!\varphi \mid \varphi \in \mathcal{L}_{tpal}\}$. A *PAL-protocol* is a set $\mathsf{P} \subseteq (\mathsf{PAL}^+)^*$ closed under finite prefix. We denote the set of PAL-protocols by $Ptcl(\mathsf{PAL}^+)$. A *state-dependent PAL-protocol* (*sd*-PAL-protocol) p on an epistemic model $\mathcal{M}$ is a function that assigns a PAL-protocol to each world in $\mathcal{M}$ a PAL-protocol. We denote the class of *sd*-PAL-protocols in this extended sense by $\mathbb{PAL}^+$. $\lhd$

Next, we introduce the notion of *orders*.

Definition 3.3.3 (Order of Formulas) *The order* $o(!\varphi)$ of a public announcement $!\varphi \in \mathsf{PAL}^+$ is defined inductively as follows:

- $o(!p) = 1$ with $p \in \mathsf{At}$

- $o(!(\varphi \wedge \psi)) = max(o(!\varphi), o(!\psi))$
- $o(!\neg\varphi) = o(!\varphi)$
- $o(![i]\varphi) = o(!\varphi)$
- $o(!(\langle!\varphi\rangle\psi)) = max(o(!\varphi) + 1, o(!\psi))$ ◁

For example, $o(!\langle!p\rangle\top) = 2$, $o(!(\langle!q\rangle\langle!\langle!p\rangle\top\rangle\top)) = 3$ etc. The order of a given public announcement indicates the greatest number of nested "!" operators. Given a sequence $\sigma = \varphi_0 \dots \varphi_{n-1} \in \Sigma_{pal}$, we define the order $o(\sigma)$ of σ by

$$o(\sigma) = (o(!\varphi_0), \dots o(!\varphi_{n-1})).$$

We denote the set of the orders of sequences by $\mathbb{O}$.

Definition 3.3.4 (Lexicographic Ordering on Orders) We define the ordering $\ll$ on the set of orders $\mathbb{O}$ lexicographically as follows. For every pair of sequences in PAL^+, $\sigma = \sigma_0 \dots \sigma_{n-1}$ and $\tau = \tau_0 \dots \tau_{m-1}$, $o(\sigma) \ll o(\tau)$ if

1. $\sigma \prec \tau$ (σ is a proper initial segment of τ as defined above) or
2. There is some $i \in \mathbb{N}$ such that
   - for all $j \in \mathbb{N}$, $j < i \rightarrow o(\sigma_j) = o(\tau_j))$, and
   - $o(\sigma_i) < o(\tau_i)$. ◁

Finally we need some notations.

Definition 3.3.5 (Union of Models) Let $\mathbb{F} = \{\mathcal{H}_k\}_{k \in I}$ be a family of ETL-models $\mathcal{H}_k = (H_k, \sim_k, V_k)$. The union $\bigcup_{k \in I} \mathcal{H}_k$ of ETL-models in $\mathbb{F}$ is a triple $(H, \sim, V)$:

- $H = \bigcup_{k \in I} H_k$

- $\sim (i) \;\; = \bigcup_{k\in I} \sim_k (i)$
- For all $p \in \mathsf{At}$, $V(p) = \bigcup_{k\in I} V_k(p)$. ◁

Recall that, given a sequence σ, we denote by $\sigma_{(k)}$ $(0 \le k \le m)$ the initial segment of σ of length k and by σ_k $(1 \le k \le m)$ the k-th element of σ.

Definition 3.3.6 (σ-Generated Models) Let $\mathcal{M} = (W, \sim, V)$ and p be an epistemic model and an sd-PAL protocol on $\mathcal{M}$ respectively. For every sequence $\sigma \in \mathsf{PAL}^*$ and every order $x \in \mathbb{O}$, we define *the σ-generated model* $\mathcal{H}^{\sigma,\mathsf{p}} = (H^{\sigma,\mathsf{p}}, \sim^{\sigma,\mathsf{p}}, V^{\sigma,\mathsf{p}})$ and *the order-x-fragment model* $\mathcal{H}^{\mathsf{p}}_x = (H^{\mathsf{p}}_x, \sim^{\mathsf{p}}_x, V^{\mathsf{p}}_x)$ by simultaneous induction as follows:

1. $\mathcal{H}^{\lambda,\mathsf{p}} = \mathcal{M}$, $\mathcal{H}^{\mathsf{p}}_\lambda = \mathcal{M}$
2. $\mathcal{H}^{\mathsf{p}}_{o(\tau)} = \bigcup\{\mathcal{H}^{\tau',\mathsf{p}} \mid o(\tau') \ll o(\tau)\}$
3. $H^{\sigma_{(n+1)},\mathsf{p}} = H^{\mathsf{p}}_{o(\sigma_{(n+1)})} \cup \{w\sigma_{(n+1)} \mid \mathcal{H}^{\mathsf{p}}_{o(\sigma_{(n+1)})}, w\sigma_{(n)} \models \sigma_{n+1} \text{ and } \sigma_{(n+1)} \in \mathsf{p}(w)\}$
4. $(w\tau, v\tau') \in \sim^{\sigma_{(n+1)},\mathsf{p}} (i)$ iff $(w, v) \in \sim (i)$ and $\tau = \tau'$
5. $V^{\sigma_{(n+1)},\mathsf{p}}(p) = \{w\tau \in H^{\sigma_{(n+1)},\mathsf{p}} \mid w \in V(p)\}$ ◁

In Item 3 in the above definition guarantees, the precondition of the $n+1$-th public announcement σ_{n+1} can properly be evaluated, since the construction in Item 2 guarantees that $\mathcal{H}^{\mathsf{p}}_{o(\sigma_{(n+1)})}$ reflects the evaluations of all public announcements of lower-order, which are necessary to evaluate the precondition of σ_{n+1}. Also the precondition is evaluated in $\mathcal{H}^{\mathsf{p}}_{o(\sigma_{(n+1)})}$, which is an ETL model. Formulas of $\mathcal{L}^+_{tpal}$ or $\mathcal{L}_{pal}$ are interpreted as ETL-formulas as in TPAL. Note that the evaluation of preconditions in Definition 2.1.3 for the construction of PAL-generated ETL models were evaluated in epistemic models.

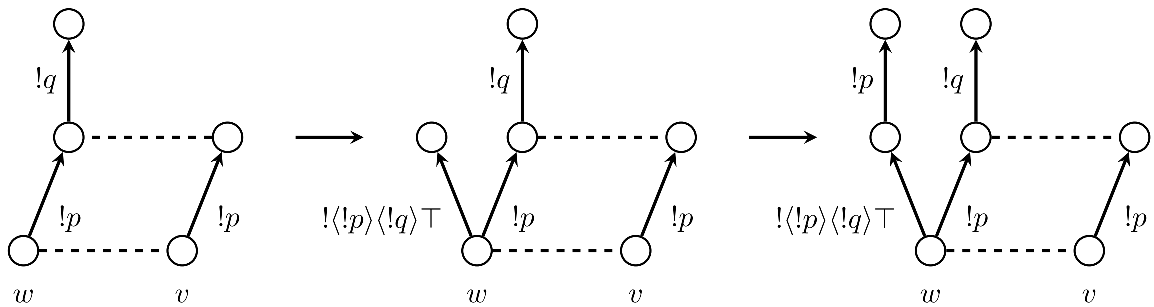

Figure 3.7: Generalizing PAL-Generated ETL Models

Definition 3.3.7 (PAL-Generated ETL Models) An ETL-model $Forest(\mathcal{M}, \mathsf{p})$ generated from an epistemic model $\mathcal{M} = (W, \sim, V)$ based on a sd-protocol p is defined by:

$$Forest(\mathcal{M}, \mathsf{p}) := \bigcup_{w \in W} \bigcup_{\sigma \in \mathsf{p}(w)} \mathcal{H}^{\sigma, \mathsf{p}}$$

We call a *PAL-generated ETL model* an ETL model generated this way. We denote by $\mathbb{F}(\mathbb{PAL}^+)$ the class of PAL-generated ETL models. $\triangleleft$

Example 3.3.8 (Example) Let $\mathcal{M}$ be an epistemic model consisting of two indistinguishable points (for an agent) w, v, in which p is true at both w and v and q is true only at w. Define a protocol p so that $\mathsf{p}(w) = \{!p!q, !\langle !p\rangle\langle !q\rangle\top\}$ and $\mathsf{p}(v) = \{!p!q, !\langle !p\rangle\langle !q\rangle\top\}$.

The following figures illustrate the construction process. The model on the left is obtained by calculating the first-order public announcement. The model on the right is obtained by calculating the second-order public announcements. The model on the right is the ETL model generated from $\mathcal{M}$ by p as specified.

$\triangleleft$

3.3.3 Representation Theorem

The construction of PAL-generated ETL models given above in Definition3.3.6 and 3.3.7 is the generalization of the construction given in Chapter 2 (Definition 2.1.3 and 2.1.3) in the sense that it applies to a wider class of protocols beyond the ones consisting only of epistemic formulas. Indeed, by inspecting the definitions of both constructions, we can easily observe the following.

Observation 3.3.9 *For every ETL model, if it is isomorphic to a model in* $\mathbb{F}(\mathbb{PAL})$, *then it is isomorphic to a model in* $\mathbb{F}(\mathbb{PAL}^+)$. *I.e.* $\mathbb{F}(\mathbb{PAL}) \subseteq \mathbb{F}(\mathbb{PAL}^+)$.

Now the question becomes whether the inclusion is a proper one. The answer turns out to be negative. It is straightforward to observe the following.

Observation 3.3.10 *For every ETL model, if it is isomorphic to a model in* $\mathbb{F}(\mathbb{PAL}^+)$, *then it satisfies propositional stability, synchronicity, perfect recall, and uniform no miracles and the following properties:*

- *for all* $h, h', he, h'e \in H$, *if* $h \sim_i h'$, *then* $he \sim_i h'e$ *(all events are reflexive)*
- *for all* $h, h' \in H$, *if* $he \sim_i h'e'$, *then* $e = e'$ *(no different events are linked).*

(For the definitions of the four properties, see also Section 1.5.2.) Recall that the properties were the properties that characterizes the class of PAL-generated ETL models, as we saw in Theorem 1.5.10. Thus by the two observations and the representation theorem, we obtain:

Theorem 3.3.11 *For every ETL model, it is isomorphic to a model in* $\mathbb{F}(\mathbb{PAL}^+)$ *iff it satisfies the properties specified in Observation 3.3.10. Therefore, we have* $\mathbb{F}(\mathbb{PAL}) = \mathbb{F}(\mathbb{PAL}^+)$.

This means that our new construction does not produce more ETL models than the old construction does. Nonetheless, extending the notion of protocols beyond epistemic formulas has a conceptual advantage. When we apply our systems to describe realistic situations of intelligent interaction, the possibilities of public announcements about future truth must frequently be considered. Indeed, our new construction method will be useful in philosophical applications of our systems, as we will see in Part II.

3.3.4 Axiomatization

We may also ask whether we can axiomatize the class of PAL-generated ETL models by the new language $\mathcal{L}^{+}_{tpal}$. The answer to this question is positive. To obtain the axiomatization, we only need to modify the axioms and inference rules in the axiomatization TPAL so that they allow instances of formulas in TPAL^{+}. Then the completeness proof can be carried out in a way similar to the completeness proof of TPAL, except for the lemma to make sure that the canonical model is in the intended class of PAL-generated ETL models, i.e. $\mathbb{F}(\mathbb{PAL}^{+})$.

In the completeness proof, the canonical model is constructed from the set of maximally consistent sets. We read off the temporal structures based on the information contained in each maximally consistent set. In particular, we construct the set of histories H_n and the function λ_n that assigns suitable maximally consistent sets to the constructed histories. by simultaneous induction by

- $H_{n+1} = \{h!\theta \mid h \in H_n \text{ and } \langle !\theta \rangle \top \in \lambda_n(h)\}$
- For each $h = h'!\theta \in H_{n+1}$, $\lambda_{n+1}(h) = \{\varphi \mid \langle !\theta \rangle \varphi \in \lambda_n(h')\}$.

After proving the truth lemma based on this construction, we need to carry out the argument to show that the model is in $\mathbb{F}(\mathbb{PAL})$. However, the argument in TPAL

(Lemma 2.3.8) crucially depends on the fact that the protocols only contain epistemic formulas. Therefore, to give an argument for TPAL$^+$, we need to do away with the assumption and work with our new construction. Some extra care must be taken for this, but the argument can be given and the extended version of the axiomatization TPAL can be proved to be complete with respect to $\mathbb{F}(\mathbb{PAL}^+)$. Denote the extended axiomatization by TPAL^+.

Theorem 3.3.12 TPAL^+ *is sound and (strongly) complete with respect to* $\mathbb{F}(\mathbb{PAL}^+)$*.*

We will leave the details of the proof to Appendix 3.7.

3.4 Discussions: Extensions in TDEL

We have considered various extensions of TPAL in the previous sections. Can we give similar extensions to TDEL? First we will discuss this question in terms of the two kinds of language extensions in TDEL: the extensions of TDEL with the generalized DEL-operator "Some event can happen after which..." and the labelled past operator "The event ϵ has happened before which...". After the language extensions in TDEL, we will consider how to generalize the construction of DEL-generated ETL models.

3.4.1 Extending TDEL with Generalized Event Operators

We start with the generalized event operator "Some event can happen after which...". By a similar consideration we gave about the generalized public announcement operators, two kinds of operators are motivated to distinguish single events and sequences of events. Thus, let us introduce the following operators, $\Diamond$ and $\Diamond^*$. Fix an e-closed set of pointed event model X and consider TDEL(X). Then, we define the two operators in TDEL(X) as follows:

$$\mathcal{H}, h \models \Diamond\varphi \quad \text{iff} \quad \exists\epsilon \in X : h\epsilon \text{ is in } \mathcal{H} \text{ and } \mathcal{H}, h\epsilon \models \varphi.$$

$$\mathcal{H}, h \models \Diamond^*\varphi \quad \text{iff} \quad \exists\sigma \in X^* : h\sigma \text{ is in } \mathcal{H} \text{ and } \mathcal{H}, h\sigma \models \varphi.$$

The intended reading of $\Diamond\varphi$ and $\Diamond^*$ are respectively "some (single) event can happen after which…" and "some sequence of event can happen after which". The duals, $\Box$ and $\Box^*$, are defined in the standard way and their intended readings of $\Box\varphi$ and $\Box^*\varphi$ are respectively "After any event, φ" and "After any sequence of events, φ". We denote the extension of TDEL(X) with by $TADEL(X)$. (The name comes from the fact that TPAL extends to TAPAL.)

Axiomatization

Can we apply the technique that we developed for TAPAL for TADEL(X)? One natural thing to try is to extend the axiomatization of TDEL(X) with axioms and inference rules similar to the ones introduced in TAPAL. The special part of the completeness proof in TAPAL that is to show consistent sets can be expanded to maximally consistent sets that are saturated with respect to $\Diamond$ and $\Diamond^*$. The argument here is carried out in terms of the special rules $R(\Box)$ and $R(\Box^*)$.

$R(\Box)$ If $\vdash \varphi \to [\sigma][!\top_0]\psi$ where $\top_0$ is a tautologous formula in $\mathcal{L}_{el}$ such that $!\top_0$ does not occur in φ or $[\sigma]\Box\psi$, then $\vdash \varphi \to [\sigma]\Box\psi$.

$R(\Box^*)$ If $\vdash \varphi \to [\sigma]\Box^k\psi$ for every k such that $0 \leq k \leq |\varphi|+1$, then $\vdash \varphi \to [!\sigma]\Box^*\psi$.

Once the argument is taken care of, the truth lemma can be proved by using the additional axioms **A2-3**:

A2 $\langle !\chi\rangle\varphi \to \Diamond\varphi$ for any $\chi \in \mathcal{L}_{pal}$

A3 $\Diamond^*\varphi \leftrightarrow \varphi \vee \Diamond\Diamond^*\varphi$

The rest of the proof is the same as TPAL.

Now we can reasonably expect that, if similar axioms and inference rules are available in TDEL(X), then we can give the completeness proof for TADEL(X) based on the completeness argument for TDEL(X). First, the counterpart of **A2** and **A3** in TADEL(X) are

E2 $\langle\epsilon\rangle\varphi \rightarrow \Diamond\varphi$ for any $\epsilon \in X$ and

E3 $\Diamond^*\varphi \leftrightarrow \varphi \vee \Diamond\Diamond^*\varphi$.

Given the truth definition of $\Diamond$ and $\Diamond^*$, it is straightforward to see that **E2** and **E3** are sound in TDEL(X).

Extending R(□)

Next, let us consider the counterpart of $R(\Box)$. Suppose $\varphi \wedge \langle\sigma\rangle\Diamond\psi$ is satisfied at some history h. Then there is some public announcement $!\theta$ such that ψ is satisfied at $h\sigma!\theta$. The key idea of the soundness of $R(\Box)$ was that we can create a new history starting with $h\sigma!\top_0$ for a tautologous public announcement $!\top_0$ that does not occur in the other part of the formula and make ψ satisfied at $h\sigma!\top_0$ without changing the truth value of other formulas.

To apply this idea to TDEL(X), we need some definitions.

Definition 3.4.1 (Event Type) Two event models, $\mathcal{E} = (E, \rightarrow, \mathsf{pre})$ and $\mathcal{E}' = (E', \rightarrow', \mathsf{pre}')$, are of *the same type*, if $(E, \rightarrow)$ and $(E', \rightarrow')$ are isomorphic. The *event type* $t(\mathcal{E})$ of an event model $\mathcal{E}$ is the class of all event models of the same type as $\mathcal{E}$. Given an event model $\mathcal{E}$, denote the type of $\mathcal{E}$ by $type(\mathcal{E})$. $\lhd$

Definition 3.4.2 (Representative) An event model $\mathcal{E} = (E, \rightarrow, \mathsf{pre})$ of an event type t is a *representative* of t, if there is a tautologous formula φ in $\mathcal{L}_{el}$ such that

$\mathsf{pre}(e) = \varphi$ for all $e \in E$. Given an event type t and a tautologous formula φ, we denote the representative of t with the tautologous precondition φ by $r(t, \varphi)$. $\lhd$

Recall that we identify isomorphic event model (Remark 1.2.11), $r(t, \varphi)$ is considered to be a unique event model. Given that there are countably many tautologous epistemic formulas, there are countably many $r(t, \varphi)$ for a given t.

In TDEL(X), there might be other *types* of event models than public announcements. Thus, to 'witness' the satisfiable formula $\Diamond\psi$, we need to choose a right type of event models. Suppose that $\varphi \wedge \langle\sigma\rangle\Diamond\varphi$ is satisfied in TDEL(X) at a history h in some model. Then there is a pointed event model $\epsilon = (\mathcal{E}, e) \in X$ such that ψ is true at $h\sigma\epsilon$. To proceed as in the soundness proof of R($\Box$), we need to take a representative of t, and replace ϵ with $(r(type(\epsilon^L), \chi), e^R)$ for some appropriate tautologous formula χ. Therefore, we can expect that, if $\varphi \wedge \langle\sigma\rangle\Diamond\psi$ is satisfiable, then $\varphi \wedge \langle\sigma\rangle\langle(r(t, \chi), e)\rangle\psi$ is satisfiable where $r(t, \chi)$ does not occur in $\varphi \wedge \langle\sigma\rangle\Diamond\psi$.

Based on this idea, the counterpart of $R(\Box)$ in TADEL(X) can be expected to be as follows. Let χ be a tautologous epistemic formula that does not occur in $\varphi \to [\sigma]\Box\psi$ or the precondition of events occurring in it.

$\mathbf{R}_X(\Box)$ If $\vdash \varphi \to [\sigma][r(type(\epsilon^L), \chi), \epsilon^R]\psi$ for every $\epsilon \in X$, then $\vdash \varphi \to [\sigma]\Box\psi$.

Clearly, we need some restrictions on the set X of pointed event models in TADEL(X). First, $R_X(\Box)$ is an infinitary rule if X consists of infinitely many event models. X must consists of a finite number of event types to obtain finite axiomatization for TADEL(X). Second, appropriate representative event modelsA should be in X. Otherwise, we cannot take $r(type(\epsilon^L), \chi)$ within TDEL(X).

One natural restriction derived from these considerations is that X is a union of finitely many event types. To state the restriction more precisely, let t be an event type. Then let $\mathbb{E}_t$ be the set of pointed event models $(\mathcal{E}, e)$ where $\mathcal{E} \in t$. Then X should be of the form $\bigcup_{i=1}^{n} \mathbb{E}_{t_i}$ for some natural number n.

Extending $R(\Box^*)$

Finally, what about $R(\Box^*)$? As in TAPAL, given the truth definition of $\Diamond^*$, it is straightforward to see the following infinitary version of $R(\Box)$ is sound in TADEL(X):

$R'(\Box^*)$ If $\vdash \varphi \to [\sigma]\Box^n\psi$ for all $n \geq 0$, then $\vdash \varphi \to [\sigma]\Box^*\psi$.

Now the question is whether we can put a finite bound on n to make the rule finitary.

Recall the argument for the soundness of $R(\Box^*)$ in TAPAL in Section 3.1.4. The key idea was to 'replace' a sequence of public announcements with a single tautologous public announcement. If $\Diamond\varphi$ is satisfied at a history h, then there is a sequence σ that φ is satisfied at $h\sigma$. σ here can be replace with a tautologous public announcement $!\top_0$ to satisfy $h!\top_0$.

This argument can be carried out in TAPAL, since public announcements are a kind of event models that correspond to model relativizations. A sequence of model relativizations amounts to a model relativization. Thus the model transformation that σ induces can be imitated by a single announcement $!\top_0$ by adjusting PAL-protocols. However, in TADEL(X), we have different event types. To carry out the soundness argument in TADEL(X), we need to guarantee that we can always find an event model of a right event type to imitate the sequence of event models.

One way to guarantee this is to restrict the set of pointed event models X in TDEL(X) to be closed under compositions. That is, if two event models, $\mathcal{E}_1, \mathcal{E}_2$, are in X then $\mathcal{E}_1 \times \mathcal{E}_2$ must be in X. However this restriction does not seem to square well with the restrictions that were suggested for $R(\Box)$, as a brief consideration suggests that the closure under composition can easily lead to infinity of event types except for some special cases (the event type of public announcement is closed under composition, etc.). Thus, we have a dilemma to obtain the axiomatization of TADEL(X) by extending TAPAL: on the one hand, we would like to have finitely many event types to make $R_X(\Box)$) a finitary rule; on the other, to make $R'(\Box^*)$ finitary, we need to have a set of event types closed under composition. It is an open question whether there are natural conditions on X to carry out the completeness argument in TADEL(X) as in TAPAL. We leave this question for future research.

3.4.2 Extending TDEL with Labelled Past Operators

Next we consider the labelled past operator. Fix an e-closed set of pointed event model X and consider TDEL(X). Then, given a pointed event model $\epsilon \in X$, we can define the labelled-past operator P_ϵ as follows:

$$\mathcal{H}, h \models P_\epsilon \varphi \quad \text{iff} \quad \exists \epsilon \in X \exists h' : \; h = h'\epsilon \text{ and } \mathcal{H}, h' \models \varphi.$$

The intended reading of $P_\epsilon \varphi$ is respectively "The event ϵ has happened before which. . . ". The dual $\hat{P}_\epsilon$ is defined in the standard way and the intended reading of $\hat{P}_\epsilon \varphi$ is "Before the event ϵ, φ."[4] We denote the extension of TDEL(X)+P.

Axiomatization

Can we extend TPAL+P to TDEL(X)+P? The special part of the completeness proof in TPAL+P was to guarantee that, if φ is consistent, then there is some sequence $!\theta_1, \ldots, !\theta_{pd(\varphi)}$ such that $\langle !\theta_1 \rangle \ldots \langle !\theta_{pd(\varphi)} \rangle \varphi$ is consistent. The argument was carried out by the following rule.

R(P) If $\vdash [!\theta_1] \ldots [!\theta_{pd(\varphi)}]\varphi$ for every $!\theta_1, \ldots, !\theta_{pd(\varphi)}$ such that each $!\theta_i$ is in $AOC(\varphi)$ or in $!T(\varphi)$, then $\vdash \varphi$, where $!T(\varphi)$ is a set of $pd(\varphi)$ tautologous public announcements not in $AOC(\varphi)$.

[4] This reading should be not taken as implying that ϵ has in fact happened. If the event ϵ did not happen, we define $\hat{P}_\epsilon$ vacuously true. See also Section 3.2.1.

The rest of the argument is the same as in TPAL in the presence of the additional axioms:

P1 $\langle\epsilon\rangle P_{\epsilon}\varphi \leftrightarrow \langle\epsilon\rangle\top \wedge \varphi$

P2 $\langle\epsilon\rangle P_{\epsilon'}\varphi \rightarrow \bot$ if $\epsilon \neq \epsilon'$

The soundness of **P1** and **P2** in TDEL(X)+P is clear given the truth definition of P_{ϵ}. Can we extend $\mathbf{R}(P)$? The answer is positive. Let us first state the version of $\mathbf{R}(P)$ in TDEL(X)+P. We need some definitions. Let φ be a formula in TDEL(X)+P. Define the past-depth $pd(\varphi)$ of φ as the highest number of nested occurrences of past-operators. Also let denote the set of pointed event models that occur in φ by $EOC(\varphi)$. Given a set X of pointed event models, define $type(X) := \{type(\epsilon^L) \mid \epsilon \in X\}$. (the set of all event types in X.)

Let φ be a formula in TDEL(X)+P. Given an event type t, take a set of $pd(\varphi)$ distinct representatives of t that are not in $EOC(\varphi)$, and let $T_t(\varphi)$ be the set of all pointed event models of the form $(\mathcal{E}, e)$ such that $\mathcal{E}$ is one of the representatives. Then the following is the counterpart of $\mathbf{R}(P)$ in TDEL(X):

$\mathbf{R}_X(P)$ If $\vdash [\epsilon_1]\ldots[\epsilon_{pd(\varphi)}]\varphi$ for every $\epsilon_1, \ldots, \epsilon_{pd(\varphi)}$ such that each ϵ_i is in $EOC(\varphi)$ or in $T_X(\varphi)$, then $\vdash \varphi$, where $T_X(\varphi)$ is the set $\bigcup_{t \in type(X)} T_t(\varphi)$.

This rule generalizes $\mathbf{R}_X(P)$ by taking representatives for every types in X. We defined the set $T_X(\varphi)$, since we may have distinct event types in TDEL(X)+P and we have to take enough (as many as $d(\varphi)$) representatives for each event type. Thus, $\mathbf{R}_X(P)$ is an infinitary rule, if there are infinitely many types in X. Also, we need to guarantee that representatives of event types are in X. These considerations, as in the case of TADEL(X), suggest the restrictions on the set X that X is a union of finite event types.

Now it suffices to show the following for the soundness of $\mathbf{R}_X(P)$.

Lemma 3.4.3 *If φ is satisfiable, then $\langle !\epsilon_1 \rangle \ldots \langle !\epsilon_{pd(\varphi)} \rangle \varphi$ is satisfiable for some $!\epsilon_1, \ldots, !\epsilon_{pd(\varphi)}$ such that each ϵ_i is in $EOC(\varphi)$ or in $T_X(\varphi)$.*

This can be proved by the argument for the soundness of $\mathbf{R}(P)$. We had two key ingredients in the argument. First, we needed to normalize the model that satisfies φ. This part of the argument can be taken care of also in TDEL(X), since the normalization theorem for TDEL can be obtained, as stated in Section 2.6.1 (Theorem 2.6.1). Based on the normalization theorem, it is straightforward to obtain the normalization theorem for TDEL(X)+P by the argument in the above normalization theorem for TPAL+P (Proposition 3.2.5). Second, we needed to 'lift' histories to obtain histories that are long enough. This part of the argument depends on tautologous public announcements. Thus to carry out the same argument in TDEL(X)+P, we require the set of public announcements PAL to be included in the set of events X.

In sum, the soundness rule can be proved with the restriction that X is a finite union of finite event types and X includes the set of pubic announcements. Based on $\mathbf{R}_X(P)$, we can carry out the completeness argument and provide the axiomatization for TDEL(X)+P. The details of the completeness argument in TDEL(X)+P are provided in Hoshi and Yap [42].

3.4.3 Events with Future Preconditions

Next, we will discuss the extension of TDEL so that the event operators containing event operators are allowed. There are two main difficulties in extending TDEL based on the above methods: *the problem of infinite regress* and *the problem of order incompatibility*.

The Problem of Infinite Regress

The first problem we deal with is the one pertinent to DEL itself. We call the problem *the problem of infinite regress.* It arises when we allow arbitrary DEL-formulas to be preconditions of events. For instance, let $\mathcal{E}$ be the event model consisting of a single point e. Suppose the precondition of e is $\langle \mathcal{E}, e\rangle\top$. Then problem is that we encounter "infinite regress", when we try to determine, say, whether the formula $\langle \mathcal{E}, e\rangle\top$ is true at a given point. To determine the truth value, we have to determine whether the precondition of e is true. However the precondition is again $\langle \mathcal{E}, e\rangle\top$. This way, we can never determine the truth value of the formula.

The problem is not limited to this specific example. Indeed, we can generate infinite regress, which involve multiple event models or even infinitely many event models. For instance, take two event models, $\mathcal{E}_1$ and $\mathcal{E}_2$, that contain events, e_1 and e_2, respectively. Suppose $\mathsf{pre}_{\mathcal{E}_1}(e_1) = \langle e_2\rangle\top$ and $\mathsf{pre}_{\mathcal{E}_2}(e_2) = \langle e_1\rangle\top$. To determine whether $\langle e_1\rangle\top$ is true at a given point, we need to determine whether the precondition of e_1, i.e. $\langle e_2\rangle\top$, is true. However to determine whether $\langle e_2\rangle\top$ is true, we need to determine the precondition of e_2, i.e. $\langle e_1\rangle\top$ is true. And so forth. This way, we can generate various examples of infinite regress and all those cases must be avoided when we allow event operators in preconditions of events.

The main idea to avoid the problem is to delimit the *legal* classes of pointed event models by some appropriate properties on the set of event models X in TDEL(X). One natural property suggests itself in view of the problem of infinite regress. An event e is *well-founded* in a set of pointed models events X if

- the precondition of e is epistemic formula, or
- all events occurring in e are in X and well-founded.

A set of pointed event models X is well-founded if all events in X are well-founded in X. It is straightforward to see that the problem of infinite regress does not arise, if the set of event models X in TDEL(X) is well-founded.

The Problem of Order Incompatibility

The other problem is *the problem of order incompatibility.* This problem can be illustrated by the following example. Let $\mathcal{M}$ be an epistemic model consisting of two indistinguishable points (for an agent i) w, v, in which p is true at w but not at v. By allowing the formulas containing event operators as preconditions, let $\mathcal{E}$ be the event model consisting of indistinguishable points (for an agent i), e_1 and e_2, whose preconditions are defined respectively by $\top$ and $\langle !\top\rangle\langle !\top\rangle\top$. Then, define a *sd*-DEL-protocol p so that $\mathsf{p}(w) = \{(\mathcal{E}, e_1)![i]A\}$ and $\mathsf{p}(v) = \{(\mathcal{E}, e_2), !\top!\top\}$ (where $!\varphi$ denotes the public announcement of φ as usual).

If we faithfully adopt the method we adopted for $TPAL^+$, we start constructing an ETL-models from the events with epistemic preconditions. This model obtained by the process is represented one the left side of Figure 3.4.3. Once this construction process is done, the next construction process would be to consider events of higher-order. In our current example, the precondition of e_2 contains public announcements inside and thus e_2 is treated at this stage. The model obtained by this process is represented on the right hand side of Figure 3.4.3. (Since $\langle !\top\rangle\langle !\top\rangle\top$, which is the precondition of e_2, is true at v in the model on the left, we create node $v(\mathcal{E}, e_2)$ as in the model on the right.) However, note that, in this model, even though there is the node $w(\mathcal{E}, e_1)![i]p$, $[i]p$ is not any more true at $w(\mathcal{E}, e_1)$, since it is now indistinguishable from $v!\top$. Thus, this model will violates the truthfulness axiom $\langle \mathcal{E}, e\rangle\top \rightarrow \mathsf{pre}_{\mathcal{E}}(e)$.

What this problem shows is that the part of DEL-generated ETL models constructed by treating lower-order events can be incompatible with the part of DEL-generated ETL models constructed by treating higher-order events. In the presence of this problem, the construction given in TPAL$^+$ cannot straightforwardly applied to the general case of TDEL. We will leave the question about how to give a proper

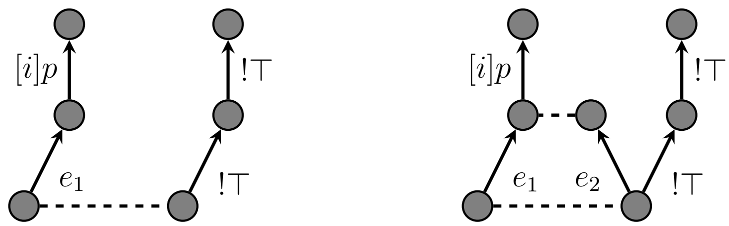

Figure 3.8: Extending TDEL 1

construction in TDEL for future research.

3.5 Conclusion and Discussion

We have considered two kinds of extensions of TPAL in this chapter. One kind was to extend the language of TPAL with operators describing temporal structures of our models. We considered the two kinds of temporal operators, generalized public announcement operators and labelled past operators, and gave axiomatizations for both extensions, TADEL(X) and TDEL(X)+P. Then we considered the extension of PAL-generated ETL models and generalized our construction method to allow public announcements containing public announcement operators. We have seen that, although the new construction method provides possible advantage in applications of our framework, it produces the same class of ETL models as the old construction method does. Finally, we discussed whether and how the methods to extend TPAL can be applied to TDEL. We saw some problems to be investigated for future research.

We conclude this chapter by listing further open questions:

Complexity Are TAPAL and TPAL+P decidable? If so, what are their complexities? For instance, APAL, the extension of PAL with the generalized public announcement operator is proved to be undecidable in [25]. What about TAPAL in the view of the result?

Common Knowledge Can we incorporate the common knowledge operator to TAPAL and TPAL+P? In the presence of the undecidability result in [49], would TAPAL plus common knowledge be axiomatizable?

Combination Can we combine TAPAL and TPAL+P? We consider the generalized public announcement operator and labelled past operator separately. Can we have both operators together in one system?

Extensions to TDEL Can we extend TDEL in the ways that TPAL was extended in this chapter? (as discussed in Section 3.4)

3.6 Appendix 1: Soundness of TAPAL

We provide details for the soundness of the axiomatization TAPAL. In particular we give the soundness proof of the rules $\mathbf{R}(\Box)$ and $\mathbf{R}(\Box^*)$.

3.6.1 Grafting

First, we need to formalize the model transformation of *grafting*. Given a sequence of public announcements σ, let $AOC(\sigma)$ be the set of public announcement that occur in σ, i.e.

$$AOC(\sigma) = AOC(\sigma_1) \cup \cdots \cup AOC(\sigma_{\mathsf{len}(\sigma)}).$$

Given a protocol p on $\mathcal{M}$, let $AOC(\mathsf{p})$ be the set of public announcements that occur in p, i.e.

$$AOC(\mathsf{p}) = \bigcup_{\{\sigma \mid \exists w \in Dom(\mathcal{M}): \sigma \in \mathsf{p}(w)\}} AOC(\sigma).$$

Definition 3.6.1 (Grafting) *The model $\mathcal{H}^{[\sigma\tau\mapsto\sigma!\top_0]}$ obtained by grafting $\mathcal{H}$ with respect to $\sigma\tau \mapsto \sigma!\top_0$* is a triple $(H^{[\sigma\tau\mapsto\sigma!\top_0]}, \sim^{[\sigma\tau\mapsto\sigma!\top_0]}, V^{[\sigma\tau\mapsto\sigma!\top_0]})$ defined by:

- $H^{[\sigma\tau\mapsto\sigma!\top_0]} := H \cup \{w\sigma!\top_0 v \mid \exists v \in \mathsf{PAL}^* : \ w\sigma\tau v \in H \text{ and } w \text{ in } \mathcal{M}\}$

- $(h, h') \in \sim^{[\sigma\tau \mapsto \sigma !\top_0]} (i)$ iff
  - $(h, h') \in \sim (i)$, or
  - $h = w\sigma!\top_0 \upsilon$, $h' = v\sigma!\top_0 \upsilon'$ and $(\sigma\tau\upsilon, v\sigma\tau\upsilon') \in \sim (i)$.
- $h \in V^{[\sigma\tau \mapsto \sigma !\top_0]}(p)$ iff
  - $h \in V(p)$
  - $h = w\sigma!\top_0 \upsilon$ and $w\sigma\tau\upsilon \in V(p)$. ◁

The idea of grafting is as discussed in Section3.1.4. Given a sequence $\sigma\tau$, we "take branches" in the ETL-model above $\sigma\tau$ in $\mathcal{H}$. Then we concatenate the "new" tautologous formula $!\top_0$ at the bottom of the branches and "graft" the branches to the corresponding nodes of the form $w\sigma$ with w in the base epistemic model.

Observation 3.6.2 *Let $\mathcal{G} = \mathcal{H}^{[\sigma\tau \mapsto \sigma !\top_0]}$. Then*

$$\mathcal{G} = \mathsf{Forest}(\mathcal{M}, \mathsf{p}^{\mathcal{G},\lambda<})$$

where λ is the empty sequence. (For the definition of $\mathsf{p}^{\mathcal{G},\lambda<}$, see Definition 2.2.11) ◁

Proof. By the similar reasoning given to obtain Observation 2.2.12. QED

Proposition 3.6.3 (Preservation at Grafted Branches) *For every $\varphi \in \mathcal{L}_{tapal}$,*

$$\mathcal{H}, w\sigma\tau \models \varphi \quad \Leftrightarrow \quad \mathcal{H}^{[\sigma\tau \mapsto \sigma !\top_0]}, w\sigma!\top_0 \models \varphi$$

◁

Proof. The proof is straightforward by Proposition 3.1.6 and the construction of $\mathcal{H}^{[\sigma\tau \mapsto \sigma !\top_0]}$. QED

The proposition gives a truth-preservation result concerning grafted models, i.e. the truth of formulas are preserved at the bottom of newly grafted branches. However, grafting does not preserve truth in general. We must be careful when transforming models by grafting to preserve the truth of formulas of our interest. We will see more on this below when we prove the soundness theorems.

3.6.2 Soundness of R(□)

The intuition behind the soundness proof is as described in Section 3.1.4. We start by observing that the normalization result can be obtained in TAPAL.

Proposition 3.6.4 (Normalization) *Let* $\mathcal{H} = \mathsf{Forest}(\mathcal{M}, \mathsf{p}) \in \mathbb{F}(\mathbb{PAL})$. *Let* X *be a finite subset of* $\mathcal{L}_{el}$. *Furthermore, let* $\varphi_0, \varphi_1 \ldots$ *be an enumeration of the formulas in* $\mathcal{L}_{el} \backslash X$ *without repetition, and* $\top_0, \top_1, \ldots$ *be an enumeration of tautologous formulas in* $\mathcal{L}_{el} \backslash X$ *without repetition. Then, for every* h *and TAPAL-formula* φ *such that* $AOC(\varphi) \subseteq X$,

$$\mathcal{H}, h \models \varphi \quad \Leftrightarrow \quad \mathcal{H}[!\top_0/!\varphi_0, !\top_1/!\varphi_1, \ldots], h[!\top_0/!\varphi_0, !\top_1/!\varphi_1, \ldots] \models \varphi$$

$\triangleleft$

Proof. Straightforward induction on φ. See also Proposition 2.2.14. QED

Next, the following fact stating that grafting with respect to $\sigma\tau \mapsto \sigma!\top_0$ where $\mathsf{len}(\tau) = 1$ preserves the truth of TAPAL-formulas.

Lemma 3.6.5 *Grafting with* $\mathsf{len}(\tau) = 1$ *Let* p *be an sd-protocol on* $\mathcal{M} = (W, \sim, V)$. *Let* $\mathcal{H} = \mathsf{Forest}(\mathcal{M}, \mathsf{p})$ *and* $w\tau\sigma!\psi$ *in* $\mathcal{H}$ *where* $w \in W$. *For every* φ, *if a tautologous formula* $\top \in \mathsf{PAL}^*$ *is not in* $AOC(\varphi) \cup AOC(\mathsf{p})$, *then*

$$\mathcal{H}, w\tau \models \varphi \quad \Leftrightarrow \quad \mathcal{H}^{[\tau\sigma!\psi \mapsto \tau\sigma!\top_0]}, w\tau \models \varphi.$$

Proof. Straightforward by induction on φ. QED

Theorem 3.6.6 (Soundness of $R(\Box)$) *If $\varphi \land \langle\sigma\rangle\Diamond\psi$ is satisfiable in $\mathbb{F}(\mathbb{PAL})$, then $\varphi \land \langle\sigma\rangle\langle !\top_0\rangle\psi$ with $!\top_0 \notin AOC(\varphi) \cup AOC(\sigma) \cup AOC(\psi)$ is satisfiable in $\mathbb{F}(\mathbb{PAL})$.* $\lhd$

Proof. Assume that $\varphi \land \langle\sigma\rangle\Diamond\psi$ is satisfiable. Thus, let $\mathsf{Forest}(\mathcal{M}, \mathsf{p}), h \models \varphi \land \langle\sigma\rangle\Diamond\psi$. This implies $\mathsf{Forest}(\mathcal{M}, \mathsf{p}), h \models \varphi \land \langle\sigma\rangle\langle\alpha\rangle\psi$ for some $\alpha \in \mathsf{PAL}$. Now take

$$X := AOC(\varphi \land \langle\sigma\rangle\langle\alpha\rangle\psi).$$

Also let $Taut$ be the set of tautologous formulas in $\mathcal{L}_{el}$. Take $Taut' := Taut \backslash X$. Then enumerate the elements in $Taut'$ and let $\top'_0, \top'_1, \dots$ be the result of the enumeration. Also take an enumeration of $\mathcal{L}_{pal} \backslash X$ without repetition so that $\top'_0$ comes as the first element. We write the enumeration as $\top'_0, \varphi'_1, \varphi'_2, \dots$. Then apply Proposition 3.6.4 by taking the following parameters:

- $X := AOC(\varphi \land \langle\alpha\rangle\psi)$
- $\varphi_0 := \top'_0, \varphi_1 := \varphi'_1, \dots, \varphi_i := \top'_i, \dots$
- $\top_0 := \top'_1, \top_1 := \top'_2, \dots, \top_i := \top'_{i+1}, \dots$.

Then, by this application of Proposition 3.6.4 together with Observation 2.2.12, we obtain

$$\mathsf{Forest}(\mathcal{M}, \mathsf{p}'), h' \models \varphi \land \langle\sigma\rangle\langle\alpha\rangle\psi$$

for some p' such that $!\top'_0 \notin AOC(\mathsf{p}')$. Now, since this implies $\mathsf{Forest}(\mathcal{M}, \mathsf{p}'), h'\sigma\alpha \models \psi$, we can apply Lemma 3.6.5 (or Proposition 3.6.3) to obtain

$$\mathsf{Forest}(\mathcal{M}, \mathsf{p}')^{[\tau\sigma\alpha \mapsto \tau\sigma !\top'_0]}, h'\sigma !\top'_0 \models \psi$$

Similarly, by applying Lemma 3.6.5 to $\mathsf{Forest}(\mathcal{M}, \mathsf{p}'), h' \models \varphi$, we can obtain

$$\mathsf{Forest}(\mathcal{M}, \mathsf{p}')^{[\tau\sigma\alpha \mapsto \tau\sigma !\top_0']}, h' \models \varphi.$$

By Observation 3.6.2, the model $\mathsf{Forest}(\mathcal{M}, \mathsf{p}')^{[\tau\sigma\alpha \mapsto \tau\sigma !\top_0]}$ is in $\mathbb{F}(\mathbb{PAL})$ and, therefore, $\varphi \wedge \langle\sigma\rangle\langle !\top_0\rangle\psi$ is satisfiable in $\mathbb{F}(\mathbb{PAL})$. QED

Corollary 3.6.7 *Let all $\varphi, \psi \in \mathcal{L}_{tapal}$ and $\sigma \in \mathsf{PAL}^*$. Also let $p, \top_0$ be a propositional letter and a tautologous formula in $\mathcal{L}_{el}$ such that $!p, !\top_0 \notin AOC(\varphi) \cup AOC(\psi) \cup AOC(\sigma)$. Then*

$$\vdash \varphi \to [\sigma][!p]\psi \;\Leftrightarrow \vdash \varphi \to [\sigma][!\top_0]\psi \;\Leftrightarrow \vdash \varphi \to [\sigma]\Box\varphi$$

$\lhd$

Proof. This follows immediate from the soundness of the rule $R'(\Box)$ given in [41] and Theorem 3.6.6 via the semantic definition of $\Box$. QED

3.6.3 The Soundness of R($\Box^*$)

Next, we deal with **R($\Box^*$)**. We start by proving the observation discussed in Section 3.1.4.

Proposition 3.6.8 (Reduction of $\Diamond^*$ to $\Diamond$) *For every $\varphi \in \mathcal{L}_{tapal}$, if $\Diamond^*\varphi$ is satisfiable in $\mathbb{F}(\mathbb{PAL})$, then $\Diamond^n\varphi$ is satisfiable in $\mathbb{F}(\mathbb{PAL})$ for $n = 0$ or $n = 1$.*

Proof. If $\mathcal{H}, h \models \Diamond^*\varphi$ with $h = w\tau$, then there is some $\sigma \in \Sigma^*_{pal}$ such that $\mathcal{H}, h \models \langle\sigma\rangle\varphi$. If σ is empty, we are done. Thus, assume that σ is not empty. By the method used in the proof of Theorem 3.6.6, obtain a tautologous formula $\top_0$, a model $\mathcal{H}'$, and a history h' in $\mathcal{H}'$ such that $\top_0$ does not occur in $\mathcal{H}'$ and $\mathcal{H}', h' \models \Diamond^*\varphi$. Then by a similar argument given in the proof of Theorem 3.6.6, we obtain

$$(\mathcal{H}')^{[\tau\sigma \mapsto \tau!\top_0]}, h'!\top_0 \models \varphi.$$

This implies the satisfiability of $\Diamond\varphi$. QED

Corollary 3.6.9 *For every $\varphi \in \mathcal{L}_{tapal}$,*

$$\vdash \Box\varphi \quad \Leftrightarrow \quad \vdash \Box^*\varphi.$$

◁

Next we define the notion of *initial box iteration* to indicate the occurrence of $\Box$ that must be taken care of in the soundness proof of $\mathbf{R}(\Box)$.

Observation 3.6.10 *Every $TAPAL$-formula is equivalent to some formula of $TAPAL$ built up by the following inductive definition:*

$$\varphi \ ::= \top \mid p \mid \neg p \mid \varphi \wedge \varphi \mid \varphi \wedge \varphi \mid \langle i \rangle\varphi \mid [i]\varphi \mid \langle !A \rangle\varphi \mid [!\theta]\varphi \mid \Diamond\varphi \mid \Box\varphi \mid \Diamond^*\varphi \mid \Box^*\varphi.$$

where $p \in \mathsf{At}$, $i \in \mathcal{A}$ and $\theta \in \mathcal{L}_{pal}$.

Proof. Immediate by the definitions of the dual operators and the standard boolean equivalences. QED

Thus, we can interchangeably use the inductive definition in Definition 3.1.1 and the one given here.

Definition 3.6.11 (Initial Box Iteration) The initial box iteration $ibi(\varphi)$ of a $TAPAL$-formula φ is defined inductively as follows:

- $ibi(p) = ibi(\neg p) = 0$ for p propositional
- $ibi(\varphi \wedge \psi) = ibi(\varphi \vee \psi) = max(ibi(\varphi), ibi(\psi))$
- $ibi(\langle i \rangle \varphi) = ibi([i]\varphi) = ibi(\varphi)$
- $ibi(\langle !A \rangle \varphi) = ibi([!A]\varphi) = 0$
- $ibi(\Diamond \varphi) = 0$
- $ibi(\Box) = ibi(\varphi) + 1$
- $ibi(\Diamond^* \varphi) = ibi(\Box^* \varphi) = ibi(\varphi)$ ◁

Now we explain the basic idea of the soundness proof for $R(\Box^*)$ below. Suppose $\varphi \wedge \langle \sigma \rangle \Diamond^* \psi$ is true at $w\tau$ in $\mathcal{H}$ (w in the base epistemic model $\mathcal{M}$). Then, ψ is true at $w\tau\sigma\upsilon$ for some υ. Now we graft the model $\mathcal{H}$ with respect to $\tau\sigma\upsilon_0 \mapsto \tau\sigma!\top_0$. This will preserve the truth of $\varphi \wedge \langle \sigma \rangle \Diamond^* \psi$ by Lemma 3.6.5. After this, we again graft at $\tau\sigma!\top_0$ in the similar way, and repeat grafting that way as many times as $ibi(\varphi)$, i.e. the number of the $\Box$-operators that must be taken care of. Once we graft $ibi(\varphi)$ times, we finally apply the grafting method for $\Diamond^*$-operator as we did in Proposition 3.6.8. This process of iterated grafting preserves the truth of $\varphi \wedge \langle \sigma \rangle \Diamond^* \psi$ and thus we can put the desired bound k for the satisfiability of the formula $\varphi \wedge \langle \sigma \rangle \Diamond^k \psi$ given the satisfiability of the formula $\varphi \wedge \langle \sigma \rangle \Diamond^* \psi$. Below we make this idea more precise.

Lemma 3.6.12 (Grafting for $\Box$) *Let p be an sd-PAL-protocol on $\mathcal{M} = (W, \sim, V)$ and φ a TAPAL-formula. Let $w\sigma \in \mathsf{p}(w)$. Put $\mathcal{H} = \mathsf{Forest}(\mathcal{H}, \mathsf{p})$ and $ibi(\varphi) = m$. Also let τ be a sequence of TAPAL-formula such that $\mathsf{len}(\tau) \geq m$. Finally let $\top_0 \notin AOC(\varphi) \cup AOC(\mathsf{p})$. Then, for every $\upsilon \in \Sigma_{pal}^*$ and $w \in W$, if $w\sigma\tau\upsilon$ is in $\mathcal{H}$,*

$$\mathcal{H}, w\sigma \models \varphi \quad \Rightarrow \quad \mathcal{H}^{\sigma\tau\upsilon \mapsto \sigma\tau!\top_0}, w\sigma \models \varphi.$$

Proof. The proof can be given by straightforward induction on φ in terms of the equivalent formulation of the formulas in $TAPAL$ as in Observation 3.6.10. QED

Theorem 3.6.13 *If $\varphi \wedge \langle\sigma\rangle\Diamond^*\psi$ is satisfiable in $\mathbb{F}(\mathbb{PAL})$, then $\varphi \wedge \langle\sigma\rangle\Diamond^k\psi$ is satisfiable in $\mathbb{F}(\mathbb{PAL}^+)$ for some k such that $0 \leq k \leq ibi(\varphi)\dot{-}\mathsf{len}(\sigma) + 1$, where $a\dot{-}b = a - b$ if $a - b > 0$; $a\dot{-}b = 0$ otherwise.*

Proof. Let $\mathcal{H} = \mathsf{Forest}(\mathcal{M}, \mathsf{p})$ and $w\tau$ in $\mathcal{H}$ with w in $\mathcal{M}$. Assume that $\mathcal{H}, w\tau \models \varphi \wedge \langle\sigma\rangle\Diamond^*\psi$. By the semantics of $\Diamond^*$, there is some $\upsilon = \upsilon_0 \dots \upsilon_{n-1}$ such that

$$\mathcal{H}, w\tau \models \varphi \wedge \langle\sigma\rangle\langle\upsilon\rangle\psi. \tag{3.1}$$

If $ibi(\varphi)\dot{-}\mathsf{len}(\sigma) \geq \mathsf{len}(\upsilon)$, we are done since we have $\mathcal{H}, w\tau \models \varphi \wedge \langle\sigma\rangle\Diamond^k\psi$ for some $k \leq ibi(\varphi)\dot{-}\mathsf{len}(\sigma) + 1$.

Thus suppose $ibi(\varphi)\dot{-}\mathsf{len}(\sigma) < \mathsf{len}(\upsilon)$. Let $a = \mathsf{len}(\sigma)$ and $b = ibi(\varphi)$. Then take a sequence of distinct tautologous formulas in $\mathcal{L}_{pal}$, $\top_0, \dots, \top_{[(b-a)-1]+1}$. By a similar argument given in the proof of Theorem 3.6.6, we can assume that $\top_0, \dots, \top_{[(b-a)-1]+1} \notin AOC(\mathsf{p})$. Then, define

$$\mathcal{H}' = (\dots(\mathcal{H}^{[w\tau\sigma\upsilon_0 \mapsto w\tau\sigma!\top_0]})\dots)^{[w\tau\sigma!\top_0\dots!\top_{(b-a)-2}\upsilon_{(b-a)-1} \mapsto w\tau\sigma!\top_0\dots!\top_{(b-a)-2}!\top_{(b-a)-1}]}$$

By repeatedly applying Lemma 3.6.5, we have

$$\mathcal{H}', w\tau \models \varphi \tag{3.2}$$

Also since (3.1) implies

$$\mathcal{H}', w\tau\sigma\upsilon_0 \dots \upsilon_{(b-a)-1} \models \langle\upsilon_{(b-1)} \dots \upsilon_{n-1}\rangle\psi,$$

by repeatedly applying Lemma 3.6.3, we have

$$\mathcal{H}', w\tau\sigma!\top_0 \ldots !\top_{(b-a)-1} \models \langle \upsilon_{(b-a)} \ldots \upsilon_{n-1}\rangle\psi.$$

and thus $\mathcal{H}', w\tau\sigma!\top_0 \ldots !\top_{(b-a)-1} \models \Diamond^*\psi$. Here consider the model

$$\mathcal{H}'' := (\mathcal{H}')^{[w\tau!\top_0 \ldots !\top_{(b-a)-1}\upsilon_{b-a}\ldots\upsilon_{n-1} \mapsto w\tau!\top_0\ldots!\top_{(b-a)-1}!\top_{b-a}]}$$

By the argument given in the proof of Proposition 3.6.8, this implies

$$\mathcal{H}'', w\tau!\top_0 \ldots !\top_{(b-a)-1}!\top_{b-a} \models \psi.$$

This gives us

$$\mathcal{H}'', w\tau \models \Diamond^{b-a+1}\psi.$$

In addition, (3.2) together with Lemma 3.6.12 implies

$$\mathcal{H}'', w\tau \models \varphi.$$

Therefore, we have $\varphi \wedge \Diamond^{b-a+1}$ is satisfied in $\mathcal{H}''$, which is clearly in $\mathbb{F}(\mathbb{PAL})$ by construction (and Observation 2.2.12 and 3.6.2). QED

Corollary 3.6.14 (Soundness of $R(\Box^*)$) *$R(\Box^*)$ is sound with respect to the class $\mathbb{F}(\mathbb{PAL})$.*

Proof. Immediate from the above theorem and the fact that

$$ibi(\varphi)\dot{-}\mathsf{len}(\sigma) + 1 \leq ibi(\varphi) + 1 \leq |\varphi| + 1.$$

QED

3.7 Appendix 2: Completeness of TPAL over $\mathbb{F}(\mathbb{PAL}^+)$

The axiomatization TPAL^+ of TPAL$^+$ consists of the axiom schemas and inference rules in TPAL. In TPAL^+, the schemas and rules can be instantiated by any formula in TPAL$^+$. We refer to the schemas and inference rules in TPAL^+ by their names in TPAL. The idea of the completeness proof is the same as in TPAL. The canonical model is constructed and the truth lemma is proved in a similar way. A special care must be taken in order to prove that the canonical model in TPAL$^+$ is in the class $\mathbb{F}(\mathbb{PAL}^+)$. For this, we first need to prove some properties of PAL-generated models. Given two sequences σ and τ, we denote by $\sigma\tau$ the sequence obtained by concatenating σ and τ in order.

Proposition 3.7.1 *Given an epistemic model $\mathcal{M}$ and $\mathsf{p}\in \mathbb{PAL}^+$ on $\mathcal{M} = (W, \sim, V)$, define $\mathcal{H}^{\mathsf{p}}_x$ for every $x \in \mathbb{N}^*$ as defined in 3.3.6. Let $y, z \in \mathbb{O}$, $n \geq 1$. Further, suppose $yn \ll z$. For every $h \in H^{\mathsf{p}}_{yn}$ and every $\varphi \in \mathcal{L}^+_{tpal}$ with $o(\varphi) \leq n$,*

$$\mathcal{H}^{\mathsf{p}}_{yn}, h \models \varphi \Leftrightarrow \mathcal{H}^{\mathsf{p}}_z, h \models \varphi.$$

Proof. First, observe that, by Definition 3.3.6, $h \in H^{\mathsf{p}}_{yn}$ implies that $h \in H^{\mathsf{p}}_z$ (by the assumption that $yn \ll z$). Thus, on the assumption that $h \in H^{\mathsf{p}}_{yn}$, we have $h \in H^{\mathsf{p}}_{yn} \Leftrightarrow h \in H^{\mathsf{p}}_z$. Denote this fact by (i). We show the claim by induction on φ. The base and boolean cases are clear. Suppose φ is of the form $[i]\varphi$. Assume LHS. Let $(h, h') \in \sim^{\mathsf{p}}_{yn} (i)$. Then, by IH, $\mathcal{H}^{\mathsf{p}}_z, h' \models \psi$. Here, by the construction in Definition 3.3.6 and the fact (i), it follows that $(h, h') \in \sim^{\mathsf{p}}_{yn} (i) \Leftrightarrow (h, h') \in \sim^{\mathsf{p}}_z (i)$. Thus we have $\mathcal{H}^{\mathsf{p}}_z, h \models [i]\psi$. The other way is similar.

Next suppose φ is of the form $\langle !\theta\rangle\psi$. First, LHS is equivalent to $\mathcal{H}^{\mathsf{p}}_{yn}, h!\theta \models \psi$. Furthermore, since $o(!\varphi) \leq n$, we have $o(!\theta) \leq n$ by the definition of o. By this fact and the construction in Definition 3.3.6, $h \in H^{\mathsf{p}}_{yn}$ implies that $h!\theta \in H^{\mathsf{p}}_{yn} \Leftrightarrow h!\theta \in H^{\mathsf{p}}_z$

(by the same reasoning as for the fact (i)). Thus, we can apply IH and obtain $\mathcal{H}^{\mathsf{p}}_{yn}, h!\theta \models \psi \Leftrightarrow \mathcal{H}^{\mathsf{p}}_{z}, h!\theta \models \psi$. This gives us the equivalence between LHS and RHS. QED

Let $\mathcal{M}_0 = (W_0, \sim_0, V_0)$ be the base epistemic model, from which the canonical ETL-model is constructed. Also, let $\mathcal{G} = (G, \approx, U)$ be the canonical model. Define p_0 on $\mathcal{G}$ so that $\mathsf{p}_0(w) = \{\sigma | w\sigma \in G\}$ for all $w \in W_0$. Given $\mathcal{M}_0$ and p_0, generate $\mathcal{H}^{\sigma,\mathsf{p}_0}$ and $\mathcal{H}^{\mathsf{p}_0}_x$ for a sequence σ of public announcements ($\sigma \in (\mathsf{PAL}^+)^*$ and $x \in \mathbb{O}$, as defined in Definition 3.3.6. For simplicity, we write $\mathcal{H}^{\sigma}$ and $\mathcal{H}_x$ respectively for $\mathcal{H}^{\sigma,\mathsf{p}_0}$ and $\mathcal{H}^{\mathsf{p}_0}_x$. Also let $\mathcal{H} = (H, \sim, V) = \mathsf{Forest}(\mathcal{M}_0, \mathsf{p}_0)$.

Proposition 3.7.2 *Let $w \in W_0$ and $\sigma \in (\mathsf{PAL}^+)^*$. Assume $v\sigma \in G \Leftrightarrow v\sigma \in H^{\sigma}$ for every $v \in W_0$ (Denote by "Assumption 1"). Then, for every $\varphi \in \mathcal{L}^+_{tpal}$,*

$$\mathcal{G}, w\sigma \models \varphi \Leftrightarrow \mathcal{H}_{o(\sigma)o(!\varphi)}, w\sigma \models \varphi.$$

Proof. We go by induction on φ. The base and boolean cases are straightforward. Suppose that φ is of the form $[i]\psi$. Assume $\mathcal{G}, w\sigma \models [i]\psi$. Let w' be such that $(w, w') \in \sim_0 (i)$. Then we have $(w\sigma, w'\sigma) \in \approx (i)$ by construction, and thus $\mathcal{G}, w'\sigma \models \psi$. Thus, by IH, $\mathcal{H}_{o(\sigma)o(!\varphi)}, w'\sigma \models \psi$. Put $\mathcal{H}_{o(\sigma)o(!\psi)} = \mathcal{H}' = (H', \sim', V')$. Here, note, for every $u \in W_0$, $u\sigma \in H^{\sigma} \Leftrightarrow u\sigma \in H'$, by the construction in Definition 3.3.6. Therefore, Assumption 1 implies that, for any u, $(w\sigma, u\sigma) \in \sim' (i) \Leftrightarrow (w\sigma, u\sigma) \in \approx (i)$. This gives us $\mathcal{H}', w\sigma \models [i]\psi$. Here $\mathcal{H}' = \mathcal{H}_{o(\sigma)o(!\psi)} = \mathcal{H}_{o(\sigma)o(![i]\psi)}$, since $o(!\psi) = o(![i]\psi)$ by the definition of o. Thus, we obtain the LHS-RHS direction. The other direction is similar.

Next, suppose φ is of the form $\langle !\theta \rangle \psi$. First, we claim that Assumption 1 implies $v\sigma!\theta \in G \Leftrightarrow v\sigma!\theta \in H^{\sigma!\theta}$ for all $v \in W_0$.

Proof of the claim: $v\sigma!\theta \in G$ implies $\langle !\theta \rangle \top \in \lambda(w\sigma)$, and by A1, $\theta \in \lambda(w\sigma)$. By truth lemma, we have $\mathcal{G}, v\sigma \models \theta$. Thus, by IH, $\mathcal{H}_{o(\sigma)o(!\theta)}, v\sigma \models \theta$. Since we have $\sigma!\theta \in \mathsf{p}_0(v)$ by the construction of p_0, we have $v\sigma!\theta \in H^{\sigma!\theta}$. The other direction is similar.

Now, assume the LHS of the biconditional. It implies that $\mathcal{G}, w\sigma!\theta \models \psi$. By the claim, we can apply IH and obtain $\mathcal{H}_{o(\sigma!\theta)o(!\psi)}, w\sigma!\theta \models \psi$. Here, note that $o(\sigma!\theta)o(!\psi) = o(\sigma)o(!\theta)o(!\psi) \ll o(\sigma)o(!\varphi)$ since $o(!\theta), o(!\psi) < o(!\langle !\theta \rangle \psi)$. Thus, applying Proposition 3.7.1, we obtain $\mathcal{H}_{o(\sigma)o(!\langle !\theta \rangle \psi)}, w\sigma!\theta \models \psi$. Therefore, we have $\mathcal{H}_{o(\sigma)o(!\langle !\theta \rangle \psi)}, w\sigma \models \langle !\theta \rangle \psi$, as desired. The RHS-LHS direction is similar. QED

Lemma 3.7.3 (Canonicity) *The canonical model $\mathcal{G}$ is in $\mathbb{F}(\mathbb{PAL})$.*

Proof. It suffices to show the following claim:

Claim 1: For every $w \in W_0$ and every $\sigma \in (\mathsf{PAL}^+)^*$, $w\sigma \in G \Leftrightarrow w\sigma \in H^\sigma$.

For this implies $G = H$ and then, by inspecting the constructions of PAL-generated ETL-models and the canonical model, we see that $\mathcal{G} = \mathcal{H}$.

We go by complete induction on the order of σ. The base case ($o(\sigma) = \lambda$) is clear by the construction of the canonical model and Definition 3.3.6. Assume that the claim holds for every τ such that $o(\tau) \ll o(\sigma)$. Let $\sigma = \sigma_1 \dots \sigma_k$. Suppose that $w\sigma_1 \dots \sigma_k \in G$. This implies $\mathcal{G}, w\sigma_1 \dots \sigma_{k-1} \models \sigma_k$ (by truth lemma) and $\sigma_1 \dots \sigma_k \in \mathsf{p}_0(w)$. Also IH implies that, for every $v \in W_0$, $w\sigma_1 \dots \sigma_{k-1} \in G \Leftrightarrow w\sigma_1 \dots \sigma_{k-1} \in H^{\sigma_1 \dots \sigma_{k-1}}$. By the construction in Definition 3.3.6, this is equivalent to:

$$\text{For every } v \in W_0,\ w\sigma_1 \dots \sigma_{k-1} \in G \Leftrightarrow w\sigma_1 \dots \sigma_{k-1} \in H_{o(\sigma_1 \dots \sigma_{k-1})}.$$

Thus, we can apply Proposition 3.7.2 and obtain $\mathcal{H}_{o(\sigma_1 \dots \sigma_{k-1})o(\sigma_k)}, w\sigma_1 \dots \sigma_{k-1} \models \sigma_k$. Given $\sigma_1 \dots \sigma_k \in \mathsf{p}_0(w)$, we have $w\sigma_1 \dots \sigma_k \in H^{\sigma_1 \dots \sigma_k}$ by Definition 3.3.6. The other way is similar. QED

Part II

Applications

Chapter 4

Knowability Paradox

In Part II, we have developed a formal framework that represents both epistemic dynamics and protocol information. We will now give applications of the formal system to some philosophical problems, in which relevant epistemic concepts can be seen as involving aspects of epistemic dynamics and protocol information. By formalizing the epistemic concepts in our system, we will try to throw new light on the philosophical problems. In giving philosophical applications of our framework, we hope not only that those examples illustrate that our framework provides a powerful tool for conceptual analysis, but also that our attempts will contribute to the interaction between philosophical investigation and formal approaches in epistemology. Our first application concerns the *knowability paradox*.

Fitch's argument, *If there is some unknown truth, then there is some unknowable truth* ([23]), poses a problem for recent verificationist accounts of semantic antirealism. In claiming that the meaningfulness of statements consists in the existence of their verification procedures, these accounts seem to be committed to the knowability thesis, *Every truth is knowable*. For if a true statement has a verification procedure, the procedure will provide a way through which the truth of the statement can come to be known. However, this thesis implies, via Fitch's argument, the counterintuitive claim *every truth is known*. This problem has come to be known as *Fitch's paradox*. Although various kinds of accounts have been produced to deal with Fitch's paradox, each account has been at least controversial in some relevant respects. The main purpose of this chapter is to propose an alternative account that avoids the problematic features of the previous approaches.

We achieve this goal by undertaking two tasks. The first task will be to show that a verificationist account does not have to be committed to the formulation of the knowability thesis.[1] We will do this by providing a philosophical framework that does not imply the knowability thesis, while preserving the verificationist thesis of anti-realist semantic accounts. Consequently, our approach will avoid the charges against some of the previous accounts that they are not motivated by verificationism.

Our verificationist framework will introduce two notions concerning verification procedures: *successful executability* and *self-retainingness*. First, if there is a verification procedure, an epistemic agent may *execute* the procedure by performing relevant actions based on it. However, in certain cases, even if a given statement is true, there are some critical constraints that preclude the *successful* executions of its procedure by the epistemic agent. Thus the truth of a given statement does not imply the successful executability of its verification procedure and does not necessarily provide us with a way through which we can come to know the truth of the statement. Second, even if the verification procedure of a given true statement is successfully executable, this still does not guarantee the knowability of the statement. We say a statement is *self-retaining* if its verification procedure, whenever it is successfully executable, can be successfully executed without changing the truth value of the statement. If

[1] An argument against such a commitment has been recently expressed in the verificationist account by Hand in [30]. Our approach shares the basic insights with his account, but is still distinct from his approach in several respects. We will compare our approach with his account. See Section 4.2.

a statement is not self-retaining, it is not knowable since it may become false after the successful execution of its verification procedure. Therefore, in our framework, the truth of a given statement does not imply its knowability, because its verification procedure may *not* be successfully executable or it may *not* be self-retaining.

Without implying the knowability thesis, the above philosophical account also blocks another problem discussed as a variation of Fitch's paradox. That is, *every truth is knowable* implies the necessary falsity of statements such as "there are no epistemic beings" If it is true then it could never be known due to the absence of epistemic beings in the situation, but then verificationism requires the necessary existence of an epistemic being to avoid inconsistency. This puzzle is known as *the idealism problem* ([57, 31, 30]). By having the distinction between successful executability and self-retainingness, our account can identify the different sources of the problems for Fitch's paradox and the idealism problem: Fitch's paradox is due to the existence of non-self-retaining statements; the idealism problem arises since the verification procedure of the statement in question is not successfully executable when the statement is true.

After the presentation of the philosophical account, our second task will be to formalize the key notions in the framework developed in Part II. In particular, we will use the system TAPAL developed in Chapter 3. First, we will interpret the successful execution of a verification procedure of a statement φ as an epistemic processes of eliminating the possibility that φ is false. This way we can formalize successful executions as public announcements. Given this interpretation, the notion of successful executability can be captured by appealing to protocols, since it concerns whether the relevant epistemic process, successfully executing verification procedures, can happen. Finally the notion of self-retainingness is a notion that involves an aspect of epistemic dynamics, since it depends on the informational states after successfully executing verification procedures. The language of TAPAL is suitable for expressing such a dynamics of agents' informational states.

There are two things that we can achieve through the formalization of our philosophical account. First the formalized account will enable us to avoid the concern raised against some of the previous accounts against the knowability paradoxes. That is, there is no guarantee that they are free of counterexamples. In our formalized account, we will be able to state a new formulation of the knowability thesis as a *provable fact* in TAPAL: a statement is knowable if it is successfully executable and self-retaining. Second, we will show that the framework of TAPAL provides a fine-grained logical analysis concerning alternative formulations of the knowability thesis. We can show that some formulations apparently similar to our new knowability thesis are in fact stronger and thus generate more theoretical burdens on verificationist accounts when the formulations are endorsed.

We proceed as follows. We will start by reviewing Fitch's paradox and the idealism problem and give a quick summary of the previously proposed accounts and their problems (Section 4.1). We will then provide our verificationist framework by clarifying and motivating the notions of successful executability and self-retainingness (Section 4.2). Next, we will move on to formalize the notions by interpreting TAPAL and give a new formalization of the knowability thesis (Section 4.3). Having the formalization, we will give a logical analysis on the new knowability thesis (Section 4.4) and discuss possible objections against our approach (Section 4.5).

4.1 The Paradoxes of Knowability and Previous Solutions

In this section, we will first review *Fitch's paradox* and its variation, *the idealism problem*. Then we will give a short survey of the previous accounts on Fitch's paradox. In particular, we will see the approaches based on *logical revisions*, *semantics reformulation*, *syntactic restriction* and *dynamic epistemic logic*. Since the purpose of the expositions is to motivate our project, we will only get to the main ideas of the previous accounts. For a more comprehensive survey of the literature, see [14].

4.1.1 Paradoxes

Fitch gives an argument to show that if there is some unknown truth, then there is some unknowable truth ([23]). This argument can be formulated as follows. Let $\Diamond\varphi$ read as "it is possible that φ" and $K\varphi$ as "φ is known (by somebody at some time)". Suppose that φ is unknown. Then $\varphi \wedge \neg K\varphi$ must be true. However, suppose $K(\varphi \wedge \neg K\varphi)$. From the two principles, $K(\varphi \wedge \psi) \vdash K\varphi \wedge K\psi$ and $K\varphi \vdash \varphi$, it follows that $K\varphi \wedge \neg K\varphi$, which is a contradiction. Thus we obtain $\vdash \neg K(\varphi \wedge \neg K\varphi)$. Here, by the necessitation rule (if $\vdash \varphi$, then $\vdash \Box\varphi$, where $\Box$ is the dual of $\Diamond$, i.e. $(\neg\Diamond\neg)$, and thus reads as "it is necessary that..."), it follows that $\vdash \Box\neg K(\varphi \wedge \neg K\varphi)$, which by duality ($\Box\neg\varphi \leftrightarrow \neg\Diamond\varphi$) implies

$$\vdash \neg\Diamond K(\varphi \wedge \neg K\varphi). \tag{4.1}$$

Thus, for all φ, if φ is an unknown truth, then the fact that φ is an unknown truth is an *unknowable* truth.

This argument was brought into a wide philosophical discussion by Hart and McGinn in [34], where they applied it to derive the counterintuitive claim that every truth is known, from the verificationist knowability thesis *every truth is knowable*. The argument is characterized in the following way. First, the knowability thesis can be schematically represented as

$$\varphi \to \Diamond K\varphi \tag{4.2}$$

By instantiating this with $\psi \wedge \neg K\psi$, we obtain

$$(\psi \wedge \neg K\psi) \to \Diamond K(\psi \wedge \neg K\psi). \tag{4.3}$$

On the other hand, Fitch's argument yields the negation of the consequent, i.e. (4.1) above. Therefore, we have $\neg(\psi \wedge \neg K\psi)$. This implies $\psi \to K\psi$ (in classical logic). This is counterintuitive since it reads as *every truth is known*. This problem has received a wide attention and come to be known as *Fitch's paradox*. How can verificationism survive Fitch's paradox?

Another problem that has been discussed as a variation of Fitch's paradox is *the idealism problem* ([57, 31, 30]). If the knowability thesis is a consequence of the general principle, *the meaningfulness of statements consists in the existence of their verification procedures*, it must concern not only the *actual* truth but also the *possible* truth. For if the principle is a correct semantic principle, then whatever statement turns out to be true in a given possible situation, its verification procedure would give us a way in which we can come to know its truth. Assuming that this is the case, consider the statement *there is no epistemic being*. If this statement is true in some counterfactual circumstance, it would have to be knowable by the knowability thesis. However, in such a circumstance, there would be no epistemic being by assumption, so it is questionable how anything could possibly be known. Thus, if the statement is possibly true, the statement would be a counterexample against the knowability thesis. Thus to keep consistency, verificationism must maintain the necessary falsity of the statement and, as a consequence, the necessary existence of some epistemic beings. How can verificationists avoid such a commitment to the necessary existence of epistemic beings?

4.1.2 Logical Revision

Logical revision approaches attempt to block some of the logical inferences used in the argument of Fitch's paradox by revising the base logic. Here are the list of inferences in the above argument that have been considered for logical revisions:

- *Epistemic Logic*: $K\varphi \vdash \varphi$ (*factivity*) and $K(\varphi \wedge \psi) \vdash K\varphi \wedge K\psi$ (*distribution*)
- *Intuitionistic Logic*: the classical step from $\neg(\psi \wedge \neg K\psi)$ to $\psi \rightarrow K\psi$, (e.g. [76, 79, 80])
- *Paraconsistent Logic*: Reductio to obtain $\vdash \neg K(\varphi \wedge \neg K\varphi)$. ([8])

Each of these approaches has been criticized. First, for the epistemic logic revision, it has been shown that the same effect as Fitch's paradox can be derived without these two rules (e.g. [57, 58, 81]). Second, for the intuitionistic logic revision, it has been pointed out that intuitionistic logic derives some other counterintuitive epistemic claims ([52]). Although the implausibility of some of the consequences has been explained away on the intuitionistic interpretation ([79, 80] etc.), it is unclear whether those intuitionistic reinterpretations are not *ad hoc* ([45]). Thus there is no reason to think that all the consequences of the knowability thesis can be suitably explained. Third, for the paraconsistent logic revision, it is highly controversial how the adoption of paraconsistent logic can be fully motivated on verificationist grounds. We can admit that the solution blocks Fitch's paradox, but the solution is not be satisfactory if we seek a *verificationist* account that avoids the paradox.

4.1.3 Semantic Reformulation

Another approach is to reformulate the knowability thesis based on some semantic intuition. A prime example of such an account is given by Edgington [20]. She reformulates the knowability principle by:

ENT $\forall s(In(\varphi, s) \rightarrow \exists s' In(K(In(\varphi, s)), s'))$

where this reads as "For all situation s, if φ is true at s, then there is some situation s' such that it is known in s' that φ is true in s." Let φ be $p \wedge \neg Kp$ in the above schema. Then there seems to be nothing paradoxical about $In(K(In(p \wedge \neg Kp, s)), s')$ for a pair of distinct situations, s distinct from s'. Thus the schema avoids Fitch's paradox.

However, it has been questioned whether a detailed semantic account along this line can be developed in a philosophically satisfactory way. First, it is unclear whether one can always specify, to an adequate degree, a situation s' distinct from the situation s ([78, 81]). Second, some questions have been raised about whether the proposed reformulation works for more complicated scenarios ([77]).

4.1.4 Syntactic Restriction

As we saw above, the knowability paradox (and its variation) arises from statements of certain forms. In the light of this, some have proposed to syntactically restrict the class of formulas to which the knowability thesis applies. Tennant provides an account based on this approach ([57, 58, 60]). Call a statement p *cartesian* if $Kp \nvdash \bot$, i.e. Kp is not provably inconsistent. Then the following restricted form of the knowability principle avoids Fitch's paradox:

TKT $\varphi \rightarrow \Diamond K\varphi$ where φ is cartesian.

For as we saw in Fitch's argument, $K(p \wedge \neg Kp)$ proves a contradiction and thus the statement in question $p \wedge \neg Kp$ is not cartesian. Dummett [19] also presents another way of restricting the knowability principle syntactically.

Accounts of this kind have been objected to on the following respects. First, the proposed syntactic restrictions seem *ad hoc*. To be taken as a verificationist response to Fitch's paradox, the restrictions must be motivated on some verificationist basis ([31, 30]). Second, although the accounts may as well avoid the knowability paradox, they do not exclude the possibility of counterexamples. Indeed, Williamson presents some problematic cases for TKT in [82]. Although Tennant replies to Williamson's putative counterexample in [59], the worry about possible counterexamples still remains unless the syntactic restrictions are grounded in some principled way.

4.1.5 Dynamic Epistemic Logic

Fitch's paradox has also been analyzed within the framework of dynamic epistemic logic. Van Benthem ([63]) explains how the type of formulas in Fitch's paradox fail to satisfy the knowability thesis in the dynamic setting where the agents' epistemic states change as they obtain new information. Appealing to *public announcement logic* (PAL, [53, 27, 74]), i.e. the extension of epistemic logic with the operator $\langle \varphi \rangle$, where $\langle \varphi \rangle \psi$ reads as "The announcement that φ can be made after which ψ is true", he analyzes different versions of the knowability principle. Also Balbiani *et al* ([5]) take the dynamic epistemic logic approach further and formally study the question what kinds of formulas satisfy the knowability schema.

The approach that we will take below sits in this dynamic epistemic logic tradition. We hope that the current paper contributes to the relevant literature on the following points. First, the studies given in the tradition, for better or worse, have not paid enough attention to the philosophical aspect of Fitch's paradox: i.e. the paradox has been raised as an objection against the verificationist account of semantic anti-realism. In fully considering this aspect, the current paper adds another serious philosophical application of dynamic epistemic logic to the relevant literature. Second, our framework as such will be able to deal with the idealism problem, which has not been analyzed in the tradition. The framework of TAPAL allows us to capture the relevant aspects of the problem.

4.2 Verificationism without the Knowability Thesis

As we have seen, the previous accounts have been considered as problematic or at least controversial. Thus, in proposing an alternative, we have to take into account the objections raised against those accounts. For this reason, we will first emphasize the following features of the account that we will propose below. Our account:

- preserves the relevant principles of the knowledge operator and the classical propositional logic.
- does not appeal to a semantic framework that has not been fully developed. For instance, we do not invoke references to the possible situations in which the relevant knowledge is realized, etc.
- precludes the possibility of counterexamples. As we will see below, our formulation of the knowability thesis is provable in the logical framework that we will adopt.
- avoids the charges of being *ad hoc* or not motivated by verificationism. We provide an account that maintains the verificationist semantic thesis and explain the failure of the knowability thesis systematically.

Our main idea to achieve this goal is to show that *verificationism can be held without any commitment to the knowability thesis formulated by KT*. Let us start by seeing the argument that derives KT from the verificationist assumption:

1. Every meaningful statement has an verification procedure.

2. If a true statement has an verification procedure, then the statement is knowable.

3. Therefore, every true statement is knowable.

Our task is to deny the second premise, while preserving the first premise, which is the statement of the verificationist anti-realist semantics.

4.2.1 Proposal

To do so, we introduce two notions concerning statements and their verification procedures: *successful executability* and *self-retainingness*. We start out by characterizing the notion of *verification procedures*. According to the verificationist semantic anti-realism, every meaningful statement has a canonical method of verifying the statement. For instance, the statement "It is raining outside", being meaningful, has its canonical method of verification, e.g. direct observations, etc. Such a method of verification is called *a verification procedure.*

We will make some assumptions about this notion in relation to the way the world is. First, if we go outside and make a direct observation, that will tell us whether it is raining or not. Generally, we may assume that, given a statement, its verification procedure will present some noticeable signs to us in one way or another concerning the result of performing it. In particular, we may assume that they will present some signs to us about the result at least when the statement is true, whereas it does not have to when it is not. Second, if it is raining outside, the direct observation *would* tell us that it is raining; if not, it would not. Verification procedures must be at least constrained this way in terms of the way the world is. Based on this consideration, we assume that the signs that *would* be revealed by a verification procedure must be determined by the way the world is when the procedure is performed, independently of who performs it, whether it is in fact performed by some epistemic agents, etc. To describe this assumed "sign-determination" relationship between verification procedures and the world, we say that the procedure *yields* the value *success*, if the corresponding statement is true. In these terms, we can recapitulate our assumptions as follows: *the signs that verification procedures would present to us are solely determined by the way the world is and verification procedures yield the value success whenever the corresponding statements are true.*

On the other hand, the notion of executability concerns the relationship between verification procedures and epistemic agents. Verification procedures must be such that epistemic agents can take instructions from them in one way or another and perform actions according to the instructions. If the verification procedure of "It is raining outside" is to make a direct observation, I go outside and observe the situation outside. To describe such an activity made by epistemic agents following the procedure, we say an agent *executes* a verification procedure. When an agent executes a verification procedure and the procedure yields the value *success*, we say the agent *successfully executes* the verification procedure. Furthermore, we say that a verification procedure is *executable* if some agent *can* execute the procedure. Similarly, a verification procedure is *successfully executable* if some agent can successfully execute it. Here, as one may object, we have only loosely defined the notion of (successful) executability, since we keep open the reading of the "can" in the definition. However, this is enough for our purpose, since we only deal with the extreme case of unexecutability, i.e. the idealism problem, as we will see in a moment. We may leave further refinement of this point up to particular verificationist accounts.

Having introduced these notions, we claim that, even if a statement is true (thus its verification procedure yields the value *success*), the procedure might not be successfully executable. Indeed, the peculiarity of the statement "there is no epistemic being" in the idealism problem presents an extreme case in point. Suppose, as we should in the idealism problem, that the statement "there is no epistemic being" is meaningful. In the actual world, the verification procedure of this statement is executable but not successfully executable, since the statement is supposedly false. On the other hand, in all possible situations in which this statement is true, the successful executability of *any* verification procedures is *logically* impossible, since there is no epistemic being in the situation. Therefore, in general, the truth of a statement does not imply the successful executability of its verification procedure, let alone the knowability of its truth.

Next, note that the notions introduced so far describe verification procedures (and the relevant items) independently of the dynamism of agents' epistemic states. When we bring this fact into the picture, another kind of knowability failure is elucidated. When epistemic agents execute verification procedures, they may obtain new information by performing the relevant actions. In some cases, the execution of a verification procedure changes the epistemic states of relevant agents to the extent that it changes the truth value of the corresponding statement. If some true statement becomes false by an execution of its procedure, then the statement cannot possibly be known simply because it is false. The prime example of such statements is a statement of the form "$p \wedge \neg Kp$." Assume that this statement is true and its verification procedure is executable. When the procedure is executed by some agent, the procedure must yield the value *success* by our stipulations. However, after the successful execution of the procedure, the truth of p must now be known (by the agent who successfully executed the procedure), which makes the whole statement false! Thus statements of the form

"$p \wedge \neg Kp$", if true, change their truth value after the successful execution of the corresponding verification procedures. To capture such a property of statements, we say a statement is *self-retaining* if, whenever its procedure is successfully executable, it is successfully executable without changing the truth value of the statement. Therefore, even if a true statement is *not* self-retaining, it may not be knowable even when its verification procedure is successfully executable.

Thus, we have highlighted the two reasons that the knowability of a true statement fails. First, its verification procedure may not be successfully executable. There may be some serious constraints that prevent the procedure from its successfully execution. Second, even if successful executability is guaranteed, the statement may still not be knowable, since the statement may not be self-retaining. Hence, the assumption 2 fails in the above argument for the knowability thesis. We claim that verificationism need not be committed to the knowability thesis.

4.2.2 Hand's Verificationist Account

Before we move on to formalize our framework, we shall mention the verificationist account proposed by Hand in [30], which is similar to our present approach. Hand argues, verificationism is never committed to the knowability thesis and thus Fitch's paradox as well as the idealism problem does not present a serious challenge raise to verificationism *per se*. To establish this, he makes the distinction between verification procedures and the *performance* of them. The bare existence of verification procedures does not guarantee that qualified epistemic agents can perform them and come to know that the corresponding statements are true. Fitch's paradox and the idealism problem reveal that there are statements such that the performability of their procedures is precluded by the truth of the very statements. Such statements are not knowable because of the violation of performability, but this is not a problem for verificationism, since it does not claim that every truth is knowable, but instead that every truth is *epistemic* in the sense that every true statement has an verification procedure.

On the one hand, the similarity between Hand's verificationist account and ours confirms that our account is much in the spirit of verificationist approaches. His distinction roughly corresponds to our distinction between verification procedures *and* successful executability plus self-retainingness. On the other hand, our present account has several advantages. First, our account will present the relevant notions in a precise manner and provide an explicit verificationist knowability thesis in a more concrete form than just saying *every truth is epistemic*. This is the task we undertake in the next section. Second and more importantly, our framework captures an essential distinction between the sources of the problems for the idealism problem and Fitch's paradox. In our terms, the source of the problem in the former is that the relevant statement is never successfully executable. Thus the problem arises when we equivocate between the truth of a statement and the successful executability of its verification procedure. Fitch's paradox results from the existence of statements that are not self-retaining. It becomes a problem when we do not consider the dynamic property of agents epistemic states. Hand's account based on performability does not provide a way to pin down the distinct sources of these problems.

4.3 TAPAL: Verificationistic Interpretation

Now to make our account more precise, we will attempt to formalize the philosophical framework proposed above. For this purpose, we will use the system of TAPAL developed in Chapter 3. Below we will give verificationistic interpretation to the system and represent the key notions of our philosophical account.

4.3.1 Interpreting TAPAL

TAPAL extends TPAL with the generalized operator $\Diamond$ and $\Diamond^*$. For our current purpose below, we will only use the operator $\Diamond$. Also, for our purpose, we will restrict our attention to the single agent case and denote the epistemic operator by K. Let us start by giving verificationist interpretations of the operators in TAPAL. We provide the list of intended readings of the operators.

1. $K\varphi$: "φ is known (by somebody at some time)."

2. $\langle !\theta \rangle \varphi$: "The verification procedure of θ can be successfully executed, after which φ is true." We say "The successful execution $!\theta$ can be made after which φ is true." for short.

3. $\Diamond\varphi$: "The verification procedure of some statement can be successfully executed, after which φ is true." We say "Some successful execution can be made after which φ is true." for short.

4. $[!\theta]\varphi$: "After the successful execution of the verification procedure of θ, φ is true." We say "After the successful execution $!\theta$, φ is true." for short.

5. $\Box\varphi$ "For every statement, after the successful execution of its verification, φ is true." We say "After every successful execution, φ is true." for short.

Also we assume that propositional letters refer to atomic propositions about the world, whose truth values are determined independently of the epistemic state of agents. Given that our purpose is to analyze the relevant epistemic concepts, such as knowledge, verification procedures, etc., we take this familiar assumption to mark off the objects of our analysis from unanalyzable atomic propositions.

Given these readings, let us consider how to express the notions in our philosophical framework. First, for every formula φ, the intended reading of φ is "φ is true".

In our philosophical framework, this is equivalent to "The verification procedure of φ yields the value *success*." Also, "$\langle !\theta \rangle \top$" reads as "The verification procedure of θ can be successfully executed after which $\top$ is true." Since $\top$ is always true, we can interpret the formula as saying "the verification procedure of θ is successfully executable." Furthermore, that a given formula φ is self-retaining can be expressed by an implication "The successful executability of φ implies that the successful execution of φ can be made after which φ is true." The antecedent is formalized as $\langle !\varphi \rangle \top$, as we have seen, and the consequent, $\langle !\varphi \rangle \varphi$. *$\varphi$ is self-retaining* is put as "$\langle !\varphi \rangle \top \vdash \langle !\varphi \rangle \varphi$". Finally, we interpret "φ is knowable" as meaning something like "as we learn things, we could come to know φ". When we learn some true statements, we learn them by checking in one way or another whether they are true or not. That is, in our terms, we learn them by successfully executing their verification procedures. Thus, given that "$\Diamond K \varphi$" reads as "Some successful execution can be made after which φ is known", we interpret "$\Diamond K \varphi$" as "φ is knowable."

4.3.2 Intended Semantics

Next we will interpret the intended semantics for TAPAL in verificationist terms. Recall that the models of TAPAL are PAL-generated ETL models. (Definition 2.1.3 and 2.1.4) PAL-generated ETL models, constructed from epistemic models based on PAL-protocols, represent possible temporal evolutions of agents' informational states over sequences of public announcements permitted by PAL-protocols. We give the following interpretation to the models of TAPAL for our current application.

First epistemic models represent agents' states of knowledge and encode what agents know. Models of TAPAL thus describe dynamics of agents' state of knowledge over time. Second, we interpret the successful execution of the verification procedure of a statement φ as an epistemic processes of eliminating the possibility that φ is false. This way we can formalize the successful execution of φ by the public announcement

$!\varphi$. Third, we capture the executability of verification procedures by PAL-protocols. The notion of executability concerns protocol information about whether the relevant epistemic process, execution of verification procedures, can happen. Thus we interpret PAL-protocols as the sequences of executable (not necessarily successfully) verification procedures at a given state. Consequently, if the verification procedure of φ is executable according to a given protocol and φ is true, then φ is successfully executable. Each node in a given PAL-generated ETL model represents a state after sequences of successful executions.

To get familiarized with these interpretations, let us review the construction of PAL-generated ETL models with the verificationist readings. Consider the model consisting of two states w and v that are indistinguishable for a given agent. Suppose p, q, r are true at w, while p, r are true and q is false at v. Given this model, we assign protocols to each state so that the protocol at w is $\{!p!q, !r\}$ and the one at v is $\{!p!q\}$. The PAL-generated ETL model obtained from the epistemic model based on the protocol is visualized in Figure 4.1. The construction of the model with our verificationist interpretation can be illustrated as follows. First having p true and executable at w, we generate the node $w!p$, which is the node to which the $!p$-arrow from w points. Similarly we construct $v!p$ above v. In addition, since w and v are indistinguishable and p is successfully executable at both worlds, they must be indistinguishable too. Thus, we connect $w!p$ and $v!p$ by the indistinguishability relation. These two states represent the epistemic state after the successful execution $!p$. Next, $!q$ is executable at $w!p$ and $v!p$, since w and v have the sequence $!p!q$. This time, $!q$ is successfully executable only at $w!p$. (Remember the truth value of propositional letters are persistent over executions.) Thus, we only generate $w!p!q$ but not $v!p!q$. As a result, $w!p!q$ singly constitutes the new epistemic state after the sequence $!p!q$ of successful executions. Finishing up the sequence $!p!q$, we finally work on the sequence $!r$. We generate $w!r$ since $!r$ is successfully executable, but do not

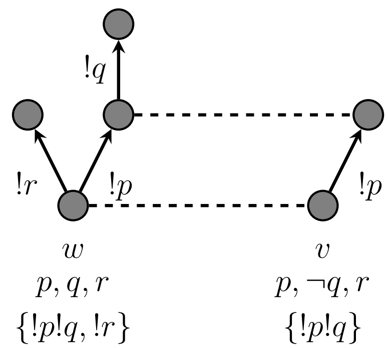

Figure 4.1: TAPAL with Verificationist Interpretations

generate the node $v!r$ since $!r$ is not successfully executable at v, though true.

As we saw in Chapter 3, PAL-generated ETL models are in general triples of the form $(H, \sim, V)$, where H is a set of trees of the above form, $\sim$ is the indistinguishability relation on the nodes in H, and V is a propositional valuation at each node of the trees in H. The semantics of the formulas in TAPAL is as defined in Definition 3.1.2. Here we only give the definition for the operators, K, $\langle\varphi\rangle$ and $\Diamond$. Let $\mathcal{H} = (H, \sim, V)$ and h be a node in H:

- $\mathcal{H}, h \models K\varphi$ iff for all h' in H, if $h \sim h'$, then $\mathcal{H}, h' \models \varphi$.
- $\mathcal{H}, h \models \langle !\theta\rangle\varphi$ iff $h!\theta \in H$ and $\mathcal{H}, h!\theta \models \varphi$.
- $\mathcal{H}, h \models \Diamond\varphi$ iff there is some ψ such that $h!\theta \in H$ and $\mathcal{H}, h\theta \models \varphi$.

Given the interpretation of our models, it is straightforward to see how the definition for the successful execution operator $\langle !\theta\rangle$ captures the intended meaning of the operator: at the history h, $\langle !\theta\rangle\varphi$ is true when ψ is true at $h!\theta$, i.e., after the successful execution *theta* at h. Similarly for $\Diamond$.

4.3.3 Deductive System

Next we will now interpret the axiomatization TAPAL (Definition 3.1.9) and justify it based on our verificationist interpretation. We will focus on the relevant part of the axiomatization. Consider the following axioms and rules in TAPAL: **Axioms**

R1 $\langle !\theta \rangle p \leftrightarrow \langle !\theta \rangle \top \wedge p$, where p is propositional.

R2 $\langle !\theta \rangle \neg \varphi \leftrightarrow \langle !\theta \rangle \top \wedge \neg \langle !\theta \rangle \varphi$

R3 $\langle !\theta \rangle K \varphi \leftrightarrow \langle !\theta \rangle \top \wedge K(\langle !\theta \rangle \top \rightarrow \langle !\theta \rangle \varphi)$

A1 $\langle !\theta \rangle \top \rightarrow \theta$

A2 $\langle !\theta \rangle \varphi \rightarrow \Diamond \varphi$

R1 reflects our assumption concerning propositional letters, i.e. they refer to atomic propositions about the world whose truths are determined independent of the epistemic state of agents. By this assumption, the truth of atomic propositions is persistent over any execution of a verification procedure. Thus, the equivalence states: if p is true, it is true after the successful execution of any verification procedure, and vice versa. **R2** simply follows from the usual meaning of negation, given the reading of the relevant operators: The successful execution of θ can be made after which φ is false iff the successful execution of θ can be made and it is not that φ becomes true after the successful execution of θ. **R3** presents the key observation in dynamic epistemic logic: if an agent comes to know ψ after the successful execution of φ, then (i) φ can be successfully executed and (ii) the agent knows (before the successful execution) that if φ is successfully executed, then ψ is true after the execution; and vice versa. Accepting this principle amounts to assuming two well-known principles concerning the relationship between agents and successful executions: *perfect recall* and *no miracles*. In Section 4.5.1, we will argue that these two principles are safe to assume in our present context of discussing the knowability paradoxes. **A1** captures the idea that the successful executability of a verification procedure implies that the verification procedure yields the value *success*, i.e. the corresponding statement is true. Note that the implication is only one way, since the truth of a statement does not imply its successful executability in our verificationist framework. **A2** captures the reading of $\Diamond$: if φ is the case after the successful execution $!\theta$, then φ is the case after some successful execution.

4.4 Logical Analysis of the Knowability Thesis

Having interpreted TAPAL in verificationistic terms, we are now ready to give a logical analysis about the verificationist knowability thesis. First we will give a new formulation of the knowability thesis as a probable fact in the interpreted system and see how it avoids Fitch's paradox and the idealism problem. Then, we will make a fine-grained comparison between the new knowability thesis and its alternative formulations.

4.4.1 New Knowability Thesis

The following statement is provable in TAPAL: for every formula φ, if φ is self-retaining and the verification procedure of φ is successfully executable, then φ is knowable, which can be formally put as:

NKT If $\vdash [!\varphi]\varphi$, then $\vdash \langle !\varphi \rangle \top \to \Diamond K\varphi$.

First, it must be noted that $\vdash [!\varphi]\varphi$ is equivalent to $\langle\varphi\rangle\top \vdash \langle\varphi\rangle\varphi$, and thus states that φ is a self-retaining statement.[2] With this equivalence, it can be now straightforwardly seen that NKT formulates the thesis we stated. Now we prove NKT:

Proof. Assume $\vdash [!\varphi]\varphi$. By the above proof, this is equivalent to $\vdash \langle!\varphi\rangle\top \to \langle!\varphi\rangle\varphi$. By epistemic logic, $\vdash K(\langle!\varphi\rangle\top \to \langle!\varphi\rangle\varphi)$. Now assume (toward the derivation of $\Diamond K\varphi$) that $\langle!\varphi\rangle\top$. Then we have $\langle!\varphi\rangle\top \wedge K(\langle!\varphi\rangle\top \to \langle!\varphi\rangle\varphi)$. By **R3**, We have $\langle!\varphi\rangle K\varphi$. By **A2**, we have $\Diamond K\varphi$. Thus, $\vdash \langle!\varphi\rangle\top \to \Diamond K\varphi$. QED

This result underscores the discussion concerning the notions of *successful execution* and *self-retainingness* in Section 4.2. There we explained the failure of the knowability thesis *Every truth is knowable* by pointing out that the verification procedure of a true statement may not be successfully executable or the statement may not be self-retaining. The current result now shows that successful executability and self-retainingness are sufficient for knowability.

Since this characterizes what our philosophical framework maintains concerning knowability in contrast with the original knowability thesis, we adopt the above formulation, NKT, as a new verificationist knowability thesis. Since it is a provable fact in our deductive system, one can find no counterexample, insofar as one accepts our verificationist framework, or more precisely, the theoretical commitment concerning the relevant notions that is made by accepting the principles in TAPAL.

[2]This can be shown by:

$$\begin{array}{llll} \langle!\varphi\rangle\top \vdash \langle!\varphi\rangle\varphi & \Leftrightarrow & \vdash \langle!\varphi\rangle\top \to \langle!\varphi\rangle\varphi & \\ & \Leftrightarrow & \vdash \neg(\langle!\varphi\rangle\top \wedge \neg\langle!\varphi\rangle\varphi) & \text{(by propositional logic)} \\ & \Leftrightarrow & \vdash \neg\langle!\varphi\rangle\neg\varphi & \text{(by R2)} \\ & \Leftrightarrow & \vdash [!\varphi]\varphi & \text{(by duality)} \end{array}$$

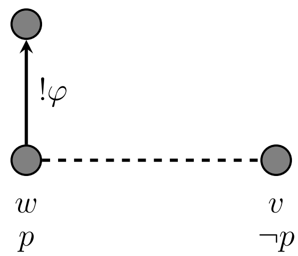

Figure 4.2: Counterexample against $\vdash [!\varphi]\varphi$

4.4.2 Fitch's Paradox and the Idealism Problem

NKT blocks Fitch's paradox, simply because $[!\varphi]\varphi$ is not derivable in TAPAL where $\varphi := p \wedge \neg Kp$. Such a φ vacuously satisfies NKT by falsifying the antecedent. In the presence of Theorem 3.1.20, this can be shown by giving a model that satisfies its negation $\neg[!\varphi]\varphi$, equivalently $\langle !\varphi\rangle\neg\varphi$. An example of such a model is given as follows. Let $\mathcal{M}$ be a model consisting of two indistinguishable states w, v where p is true at w but not at v. Assign $\{!\varphi\}$ to both w and v. Then the corresponding PAL-generated ETL model $\mathcal{H} := \mathsf{Forest}(\mathcal{M}, \mathsf{p})$ is visualized in Figure 4.2. In the model, we have $\mathcal{H}, w!\varphi \models p$, since $\mathcal{M}, w \models p$. Thus, since there is no indistinguishable node from $w!\varphi$ except for itself, $\mathcal{H}, w!\varphi \models Kp$. This implies $\mathcal{H}, w!\varphi \models \neg(p \wedge \neg Kp)$. By the semantic definition of $\langle !\varphi\rangle$, we obtain $\mathcal{H}, w \not\models \langle !\varphi\rangle\neg\varphi$.

Also, the idealism problem does not arise for NKT. To see this, let p be "there is no epistemic being". Given the meaning of p, we have at least:

$$p \to \neg\Diamond\top.$$

For this says that if there is no epistemic being, then there will be no successful executions. Now assume that p is successfully executable, so $\langle !p\rangle\top$. Then it implies p by **A1**. By the above implication, we immediately obtain $\neg\Diamond\top$. On the other hand, by instantiating **A2**, we obtain $\langle !p\rangle\top \to \Diamond\top$. With our assumption, it follows from this that $\Diamond\top$. Therefore, our assumption leads to contradiction $\langle !p\rangle\top \vdash \bot$. Thus, p satisfies NKT vacuously by falsifying $\langle !p\rangle\top$. Thus it does not commit us to the knowability of p.

Moreover, it is worth noting that the formula $\neg\Diamond\top$ is consistent in TAPAL. Given an epistemic model $\mathcal{M}$, set a state w and assign the empty set to w. Then, we see $\neg\Diamond\top$ must be true at w (since the formula says that there is no successful execution). This shows that our theoretical commitment in TAPAL does not preclude the possibility of "there is no epistemic being" being true.

4.4.3 Comparison with Alternatives

We adopted NKT as the new knowability principle for verificationism. To elucidate what it says, we will compare it to other plausible knowability statements. Particularly, we will consider two alternative principles, LKT and WKT. We first list all three principles:

NKT For all φ, If $\vdash [!\varphi]\varphi$, then $\vdash \langle !\varphi\rangle\top \to \Diamond K\varphi$

LKT For all φ, $\vdash \langle !\varphi\rangle\varphi \to \Diamond K\varphi$.

WKT For all φ, if $\langle !\varphi\rangle\varphi \nvdash \bot$, then $\vdash \langle !\varphi\rangle\top \to \Diamond K\varphi$.

These principles look very similar. With the intended interpretation of TAPAL, we could express these principles in slightly different ways. First, NKT says "For every statement, if it is self-retaining, then it is knowable on the condition that its verification procedure is in fact successfully executable." Second, LKT says "For every statement, if the successful execution of its verification procedure can be made after which it does not change its truth value, then it is knowable." Third, WKT says "For every statement, if the assumption that it does not change its truth value after the successful execution of its verification procedure does not lead to contradiction, then the statement is knowable on the condition that it is in fact successfully executable." However, it seems very unclear how much difference, if any, is being made explicit by these English translations.

Moreover, as we look into the properties of these principles, we find further similarities. First, all the formulations are consistent with TAPAL in the sense that, even if we stipulate one of the principles in TAPAL, the system will remain consistent. The consistency of NKT is immediate by the fact that it can be proved in TAPAL and the consistency of LKT and WKT is given by constructing models that satisfy them. There are many non-trivial models that satisfy these principles, but, for simplicity, we can simply take empty protocols assigned to each state of a given epistemic model to satisfy these principle. For such a model will vacuously satisfy the stipulated conditionals. Second, the three formulations are also similar in the sense that they all avoid Fitch's paradox and the idealism problem. We have already seen how NKT avoids the problems above. To see how LKT and WKT avoid Fitch's paradox, note that, with $\varphi := p \wedge \neg Kp$, we can prove that $\langle \varphi \rangle \varphi \vdash \bot$. [3] This makes such a φ vacuously satisfy LKT and WKT. For the idealism problem, set the propositional letter p as is done in the above argument for NKT. Then we have $\langle !p \rangle \top \vdash \bot$. This again makes p satisfy LKT and WKT vacuously.

Despite these similarities, LKT and WKT are different from NKT. Let an epistemic model consist of w, v, u, which are indistinguishable to each other. Further let p, q be true at w, $\neg p, q$, at v, and $\neg p, \neg q$, at u. Assign to all w, v, u the protocol $\{!\psi\}$ with $\psi := p \vee \neg Kq$. The resulting PAL-generated ETL model $\mathcal{H}$ can be visualized in Figure 4.3. In the model, we have $\mathcal{H}, w \models \langle !\psi \rangle \psi$ but $\mathcal{H}, v \models \langle !\psi \rangle \neg \psi$. This gives $\mathcal{H}, w \models \langle !\psi \rangle \neg K\psi$. Given that $!\psi$ is the only permissible execution, $\mathcal{H}, w \models$

[3] By **A1** and propositional reasoning, $\langle !\varphi \rangle \top \vdash p$. This implies propositionally $\vdash \langle !\varphi \rangle \top \rightarrow \langle !\varphi \rangle \top \wedge p$. By **R1**, $\vdash \langle !\varphi \rangle \top \rightarrow \langle !\varphi \rangle p$. Thus by epistemic logic, we have $\vdash K(\langle !\varphi \rangle \top \rightarrow \langle !\varphi \rangle p)$. On the other hand, by standard modal reasoning, $\langle !\varphi \rangle \varphi \vdash \langle !\varphi \rangle \neg Kp$. By R3 with some propositional reasoning, we obtain $\langle !\varphi \rangle \varphi \vdash \neg K(\langle !\varphi \rangle \top \rightarrow \langle !\varphi \rangle p)$. Thus, $\langle !\varphi \rangle \varphi \vdash \bot$.

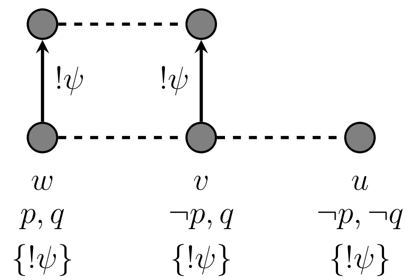

Figure 4.3: Counterexample to LKT and WKT

$\langle !\psi\rangle\top \wedge \neg\Diamond K\psi$. This yields a counterexample to LKT. Also, since $\mathcal{H}, w \models \langle !\psi\rangle\psi$, we have $\langle !\psi\rangle\psi \not\vdash \bot$. Thus, this model violates WKT too.

The above arguments show that LKT and WKT imply NKT but not vice versa. For the arguments show that the classes of the models that satisfy LKT and WKT are proper subclasses of the class of models that satisfy NKT, i.e. the class of all PAL-generated ETL models. Therefore, we can say that LKT and WKT require stronger theoretical commitments than NKT does. NKT, being a provable fact in TAPAL, does not constrain the models beyond the general semantic constraints of the TAPAL framework. For instance, it is completely neutral over the permissible structures of protocols, the structures of epistemic models, etc. Unless we find some independent philosophical reasons that force us to restrict ourselves to particular structures of the models, we should allow as little theoretical commitment as possible concerning the knowability principle by adopting NKT, and leave more logical possibilities for individual accounts of verificationist semantic anti-realism. This is the reason that we have adopted NKT as the verificationist knowability commitment in this paper.

4.5 Objections and Discussions

Having presented our account concerning Fitch's paradox, we will now discuss some of the possible objections against our account.

4.5.1 The axiom R3: perfect recall and no miracle

Objection. First, in the deductive system of TAPAL, how can the axiom **R3**,

$$\langle !\varphi\rangle K\psi \leftrightarrow \langle !\varphi\rangle \top \wedge K(\langle !\varphi\rangle \top \rightarrow \langle !\varphi\rangle \psi)$$

be justified in the proposed verificationist framework? Unless it is well-motivated, verificationists will not accept the deductive system even if they accept the proposed framework.

Reply. As mentioned in Section 4.3.3, accepting the axiom **R3** requires us to assume the well-known properties, *perfect recall* and *no miracles*. In our context, these properties concern agents and successful executions. The main idea of *perfect recall* is that agents do not forget about epistemic states in the past. The main idea of *no miracles* is that each successful execution of a fixed statement produces the same effects on the agents epistemic state, that is, one successful execution of a statement does not miraculously produce the information that other successful executions of the same statement do not produce.

One of these two properties justifies each direction of the biconditional. Perfect recall justifies the right-to-left direction. Given that an agent knows what is claimed in the right side of the equivalence and that the agent does not forget that piece of knowledge, we can easily see that the agent *will* know that ψ after the successful execution of φ. No miracle justifies the left-to-right direction. Given that the successful execution of φ always has the same epistemic effect, we cannot explain the fact that the agent will know ψ after the successful execution of φ unless we accept that the agent has the piece of knowledge claimed in the right side of the equivalence.

As we can see, these properties idealize agents and successful executions. However we claim that these idealizations can be assumed for the purpose of our discussions.

The question concerning the knowability thesis, as it is discussed in the literature, is the following: Is there any essential fact about our epistemic capacity that, in principle, prevents us from knowing certain true statements? As we investigate this question, facts about our forgetfulness or contingent epistemic side effects of particular epistemic events may be, and have been, left out of the debate. Therefore, we claim that these properties may be assumed and thus that the axiom **R3** can be accepted in our verificationist framework.

4.5.2 In Some Sense Knowable

Objection. Some formulas, say, $\neg Kq$, seem to be in some sense knowable, but they do not count as knowable by our formulation, since they are not self-retaining. However, this seems counterintuitive, since we can easily think of some model in which $\Diamond K \neg Kq$ is true (See [5]).

Reply. Granted. Our formulation of the knowability thesis does not count, say, $\neg Kq$ as knowable. However, this is no objection against our formulation. The main reason is that, in such cases, until the meaning of propositions p and q in the formula are fully specified, our account does not tell whether the formula is knowable. In our semantic setting, the meaning of p and q will be "specified", so to speak, when the class of the relevant epistemic models is fixed. After this, we can talk about the knowability of the formulas.

On the other hand, our formulation of the knowability thesis is only dependent on the structural properties of the relevant notions. Therefore, it is no surprise that our structural formulation of the knowability thesis does not imply the knowability of the formula of the kind in question. Thus, NKT only states the structural limitations concerning knowledge and may not decide the knowability of some formula without a suitable specification of the meaning of propositions.

4.5.3 Logical Omniscience on Knowledge

Objection. The system presented in Section 4.3 assumes that agents are logically omniscient, since it is an extension of epistemic logic. However, isn't this assumption problematic?

Reply. Granted. TAPAL, as an extension of epistemic logic, assumes logical omniscience. However, we do not take this as problematic for the reason mentioned in Section 4.5.1. That is, our question concerning the knowability thesis is whether there is any essential fact about our epistemic capacity that, in principle, prevents us from knowing certain true statements. In the light of this, it does not seem problematic to assume that all logical consequences are knowable by (some finite extensions of) epistemic capacity like ours, given that we have an effective procedure for the deductive system.[4] Thus, although epistemic logic may not give the best representation for knowledge in all contexts, it can at least represent enough for the purpose of analysing the knowability thesis.

4.5.4 Why Do We Have to Buy the Semantics?

Objection. Setting aside the above problems, the epistemic models do not represent the correct notion of knowledge anyway. Why must we accept all the theoretical baggage that comes with these epistemic models?

Reply. Granted. We do not claim that these epistemic models correctly represent every aspect of the notion of knowledge. Rather we claim that the semantic device we invoke is a heuristic device to capture some relevant aspects of the notions in question.

[4] As we mentioned above, although we presented a system with an infinitary rule as an axiomatization of TAPAL, this infinitary rule can be replaced with some finitary one. Therefore our proof system is effective.

If modelling in epistemic logic is not palatable, then we can just forget about the intended semantics and accept the axiomatic system by convincing ourselves of the validity given the intended readings of the operators and our pre-theoretic intuitions about the relevant notions. We can still achieve the goals of this paper in the same manner without depending on this particular semantic story.

4.6 Conclusion

We have presented a verificationist framework that avoids Fitch's paradox and the idealism problem. Given a true statement, we distinguish its truth from the successful executability of its verification procedure and its self-retainingness. This distinction in our framework allows verificationism to hold that every statement has its verification procedure without being committed to the knowability thesis, *every truth is knowable*. We have also shown that, by appealing to the system TAPAL, we can clarify the logical relationship between the relevant notions and make explicit the presuppositions we use to draw distinctions between them. Moreover, the formalization in TAPAL allows us to formulate the verificationist commitment concerning knowability as a theorem in TAPAL: for every statement, if it is successful, then it is knowable provided that it is successfully executable. Also we can give a fine-grained comparison between alternative formulations of the knowability thesis.

We conclude the paper with some general remarks about our framework from the perspective of dynamic epistemic logic. The core of our framework hinges on two crucial demarcations. First, we make the notion of verification procedure independent of epistemic actions for actually executing them. This allows us to draw the line between verification procedures yielding the value *success* and their successful execution. As we diagnose, the idealism problem arises when successful executions of verification procedures are not distinguished from verification procedures yielding the value *success*. Second, we abstract away the possible dynamic changes to agents' epistemic states from the actions of executing verification procedures. This allows us to introduce the notion of self-retaining statements. As we diagnose, Fitch's paradox arises when these two concepts are not clearly separated.

Recent developments in dynamic epistemic logic have clarified this second aspect. In dynamic epistemic logic, the static epistemic states of agents are represented by *epistemic models*, and, independently from them, relevant informational events are represented by *event models*. The dynamic character of agents' epistemic states is represented by those model transformations on given epistemic models that are induced by given event models. In the light of this, the main contribution of our system is to introduce into dynamic epistemic logic a framework that can represent the first distinction. In our framework, the executability of verification procedures is represented by *protocols* and the information in the protocols is determined independently from truth in a given epistemic model. This additional structure gives us another dimension along which to describe epistemic phenomena and thereby disentangle the puzzlement in the knowability paradox. Therefore, protocol information highlights the important aspects of dynamic epistemic events.

Chapter 5

Logical Omniscience and Deductive Inference

The next philosophical application of our framework concerns what is called *the problem of logical omniscience*. Epistemic logic validates the principle *if* $[i]\varphi$ *and* φ *logically implies* ψ*, then* $[i]\psi$. When the epistemic operator $[i]$ is interpreted as "i knows...", what the principle amounts to saying is that *agents know whatever is logically implied by what they know* or simply *knowledge is closed under logical implication*. However, this principle does not fit the notion of knowledge at least in its ordinary sense. It is simply false to say that *I*, for instance, know every theorem of Peano Arithmetic, even though I know the axiomatic system. It seems that the principle can be satisfied only by highly idealized agents, *logically omniscient* agents, but does not properly represent knowledge of realistic agents like us with finite cognitive resources. This problem is known as *the problem of logical omniscience*.

One popular view on the problem is based on the distinction between *explicit knowledge* and *implicit knowledge*.[1] Explicit knowledge is often characterized as what an agent *concurrently* knows and implicit knowledge is whatever follows from explicit

[1] The distinction was first brought up by Levesque [47] on this topic.

knowledge. On the popular view, what epistemic logic describes by the operator $[i]$ is implicit knowledge and the principle is not problematic when interpreted accordingly. What needs to be done in epistemic logic is then to characterize the notion of explicit knowledge that is free from the problem of logical omniscience. Based on this perspective, various alternative systems of epistemic logic have been proposed in the literature.

On the other hand, Robert Stalnaker ([55, 56]) presents a rather negative view on the prospect of giving a reasonable formalization to the notion of explicit knowledge. He claims that what distinguishes what we concurrently know from what we know implicitly is the notion of *availability*. Knowledge is *available* if we have the capacity to make actions depend on it whenever we want to act that way. Therefore, to characterize the required notion of availability, we need to consider the notion of knowledge in relation to actions and motivations. The problem of logical omniscience consists in the fact that epistemic logic leaves these relevant factors out of the picture. For this reason, insofar as epistemic logic approaches the notion of knowledge as it does, it would not reasonably capture what we concurrently know in contrast to what we implicitly know. Furthermore, the three notions are so intimately connected and we might simply not be able to disentangle them to the extent that we can properly formalize them.

In the current chapter, we will challenge this pessimistic view and give a meaningful formalization of explicit knowledge. We will, as we should, grant Stalnaker that actions and motivations have an important connection to the notion of knowledge. Also we may further grant him that the formalization of knowledge that takes the factors into account may be very difficult to give. However, we will claim that the characterization of explicit knowledge can be given in an elucidating manner so that it does not depend on the notions of actions and motivations. The characterization that we propose is: *an agent explicitly knows φ if and only if she would not obtain* *any new information by observing φ.* With this characterization, we will be able to formalize the notion of explicit knowledge by suitably interpreting the system TPAL.

Furthermore, the formalization of explicit knowledge, which does not presuppose logical omniscience, makes it meaningful to think about the representation of deductive inference in the framework of epistemic logic. For realistic agents without logical omniscience, explicit knowledge is not closed under logical implication and, for this reason, such agents can extend knowledge by making deductive inferences. Thus, we will propose a characterization of logical inferences based on the notion of explicit knowledge. The characterization that we will propose is that logical inference is a process of observing what follows from explicit knowledge. This characterization will imply that *by deductively inferring φ, an agent obtains the information that she was able to observe φ.*

Bringing together the notions of explicit knowledge and deductive inference in a single formal framework of epistemic logic, we will be able to provide a ground on which to compare two different perspectives on the *epistemic closure principle*. In epistemic logic, the principle, formulated as above, is considered as a problem, since it presupposes logically omniscient agents, for whom the notion of deductive inference would be meaningless. In epistemology, the principle has often been discussed as a principle that guarantees that we can always extend our knowledge by deduction. By having our formal characterizations, we can view both of the perspectives together in one system. By using our framework, we will be able to give a formal analysis about the difference between the two perspectives.

Finally the purpose of the current chapter can be also motivated by the recent literature in epistemic logic. As mentioned above, various types of systems have been developed to block logical omniscience. (See e.g. a survey on the problem of logical omniscience in [54].) In addition, representations of deductive inference is a topic of increasing interests (e.g. [16, 17, 18, 66, 65, 43, 44, 75], etc.) Our approach takes a perspective from dynamic epistemic logic on those existing research directions.

We proceed as follows. We will start by reviewing Stalnaker's view on the problem of logical omniscience. (Section 5.1) We will then present our characterization of explicit knowledge in the way that avoids the problem that Stalnaker raises. Having the characterization of explicit knowledge, we will also propose a characterization of deductive inference. (Section 5.2) After this, we will formalize those characterizations by giving suitable interpretation of TPAL. (Section 5.3 and 5.4) By using the formalization of explicit knowledge and deductive inference, we will discuss perspectives in epistemic logic and in epistemology on epistemic closure principle. (Section 5.5)

5.1 Stalnaker on the Problem of Logical Omniscience

In epistemic logic, we have the following principle:

LO If $[i]\varphi$ and φ logically implies ψ, then $[i]\psi$.

This is validated by the two basic principles of epistemic logic, the *K-axiom* and the *necessitation rule*:

K $[i]\varphi \wedge [i](\varphi \rightarrow \psi) \rightarrow [i]\psi)$

Nec If $\vdash \varphi$, then $\vdash [i]\varphi)$.

Suppose $[i]\varphi$. If φ logically implies ψ, i.e. $\vdash \varphi \rightarrow \psi$, then we obtain $\vdash [i](\varphi \rightarrow \psi)$ by necessitation. By K-axiom, we obtain $[i]\psi$. The principles are validated in any standard interpretation of the modal operator $[i]$ with respect to Kripke models.[2]

[2] If φ and $\varphi \rightarrow \psi$ are true at each world accessible from a given world w ($[i]\varphi$ and $[i](\varphi \rightarrow \psi)$ are true at w), then ψ must be true at each world accessible from w ($[i]\psi$ is true at w). Also if φ is logically true ($\vdash \varphi$), then it is true at every world. This implies that $[i]\varphi$ is also true at every world ($\vdash [i]\varphi$).

Therefore, **LO** is validated in any version of epistemic logic based on the standard interpretation of modal operators. For this reason, the previous attempts to represent explicit knowledge without validating **LO** have been made by revising the standard semantic framework of epistemic models and introducing a new modality with non-standard interpretations.

However, Stalnaker [55, 56] presents a rather negative view about the project of characterizing the notion of explicit knowledge in epistemic logic. For the rest of this section, we will give a reconstruction of his argument.

Reconstruction of Stalnaker's Argument

According to Stalnaker, the reason that **LO** is problematic when applied to more realistic agents like us is based on our intuition that, even when some information is implicit in our knowledge, we may not have that information *available* in order to make decisions about our actions. Suppose my friend presents me a card with the number 1571 written on it. This number is in fact a prime number. Here, it seems quite possible, in an ordinary sense of knowledge, that I *do not* know whether the presented number is prime, even though my knowledge about prime numbers—e.g. the concept of prime number, effective procedures to determine whether a given number is prime, etc.—implies that 1571 is prime. Indeed, I may not be able to reasonably make up my mind based on the implicit information in order to answer 'yes', at the time that it is asked whether the number is prime. I have to take some time and must see what my careful calculation will turn out to be. In this sense, the information that the number is prime is implicit in my knowledge but not *available* to me to make my actions depend on it. What this suggests is that, in general, available knowledge must be characterized in relation to actions that an agent can make. Stalnaker says:

> [T]he problem is that we need to understand knowledge and belief as capacities and dispositions—states in order to distinguish what we actually know and believe, in the ordinary sense, from what we know and believe only implicitly. We can do this only by bringing the uses to which knowledge and belief are put into the concepts of knowledge and belief themselves, but, on the face of it, it does not seem that when we attribute knowledge or belief to someone we are making any claims about what the agent plans to do with that information. ([55] p.253)

Based on the consideration, Stalnaker gives the following rough characterization of available knowledge: *an agent know φ, if the agent have the capacity to make her action depend on whether φ.*

However, this is not the end of the story. Suppose that, in the above scenario, my friend has the only two cards, one with 1571 (prime) and the other with 1591 (non-prime). Therefore, I can be presented only with either of the two number cards, 1571 and 1591. Now assume that I have the capacity to make my action depend on whether the presented number is 1571 and I have the capacity to make my action depend on whether 1591. (I have a good vision on the card that is presented and can recognize these numbers.) In that situation, we can say, I would have the capacity to make my action depend on whether 1571 is prime. (If the presented number is 1571, it is automatically a prime number; if the presented number is 1591, it is automatically not a prime number) However, this situation could happen, even when I do not know 1571 is prime. Therefore the above characterization does not seem to fully capture the notion of available knowledge.

What this example illustrates is that, in general, an agents may *happen* to have the capacity to make actions depend on certain information in the way that the capacity does not represents her internal state as much as we want it to. In view of this problem, Stalnaker claims that such situations can be marked out by bringing agents' motivational states into the picture. What he proposes as the characterization of available knowledge can be cashed out as: *an agent know φ, if the agent have the capacity to make her action depend on whether φ and she is disposed to make her action depend on whether φ whenever she wants her action to depend on whether φ.*[3]

With the above conception of knowledge, a part of Stalnaker's view about the problem of logical omniscience can be characterized as follows. Availability of knowledge must be characterized in relation to action and motivation. As far as epistemic logic leaves these elements out in representing knowledge, as its standard framework does, it cannot reasonably capture availability of knowledge or any notion of knowledge that supports our intuition that we have when resisting the validity of the principle **LO**.

However, this seems to be only a part of Stalnaker's view. His negative view seems to go further. He seems to suggest that the relevant notions here, available knowledge, action and motivation, are so intimately related to each other that we cannot formally characterize what knowledge an agent has available based on action and motivation of an agent. Here is what Stalnaker says:

> But the problem is that we have no independent way to assign content to the motivational states. If we are talking just about machines and systems that we build and program to serve our needs, then it will be easy to see how to interpret the content of the "wants" of the processors, but we want our theory to apply also to organisms and systems that we find, and want to understand as autonomous agents. . . the information-bearing states of participants in the system have the (implicit) content they do because of the structure of the system—the constraints imposed by that structure on the relation between the internal states of the participants and the global

[3] The material discussed in this paragraph is from pp. 265-266 in [56], where Stalnaker discuss available knowledge that φ under condition ψ. We reconstructed the version without the relativization in the text above to make explicit what we think is Stalnaker's main argument.

> states of the system of which it is a part. If our theory is to contribute to an explanation of intentionality, then the decision rules and motivational states should also get their content from the structure of the system, and not be imposed from outside by the intentions and desires of the users. Motivational states should derive their contents from the dispositions of the participants to make its actions depend on the information it has. But it will be the dispositions to use the *available* knowledge, not the implicit knowledge, that will determine the content of decision rules and motivational states. (*ibid.* p.266)

Thus, when we would like to characterize available knowledge, we need to know actions and motivation of an agent. However, to know actions and motivation, we now in turn have to know what knowledge the agent has available. This interdependence between the notions makes the project of characterizing available knowledge look even intractable.

Then, what should we do? Should we abandon the project of formalizing the notion of knowledge that does not satisfy **LO** for the reasons that Stalnaker presents? Our answer to this question is negative. We will agree with Stalnaker on the importance of action and motivation to the characterization of knowledge. We will also grant him that the intimate interdependence between them makes it difficult to characterize available knowledge based on action and motivation. However, we claim that there is still a way of characterizing the desired notion of knowledge before we commit ourselves to the notions of action and motivation. This is the project that we will undertake for the rest of the chapter.

5.2 Explicit Knowledge and Deductive Inference

Our basic strategy is to characterize the notion of explicit knowledge by addressing the dynamics of information when an agent makes *observations*. The information we have about the world gets updated when we make observations. For instance, I may not have information about whether it is raining outside right now, but I will come to know it by going outside and observe the current weather outside. Or I may know the procedure to determine whether a given number is prime but may not tell whether 1571 is prime when the number is presented to me. After careful calculation, I will come to know whether the number is prime. Thus, observations, in the broad sense of the term, bring our informational states to new informational states.

Depending on what informational states we are in, a given observation can impact on our informational states in different ways. If I do not know whether it is raining now, observing the current weather outside will update my informational state by the new information. If I already know whether it is raining, then observing the current weather outside will not give me any new information. Similarly, if I do not concurrently know whether 1571 is prime though the information that the number is prime is implicit in my knowledge about arithmetic, my careful calculation will give me new information. However, once I concurrently come to know whether the number is prime, repeating calculation will not give me new information.

Based on these considerations, we articulate the standard intuition of explicit knowledge as follows: *An agent i explicitly knows φ iff i would not obtain new information after observing the information that φ.* If we concurrently know a certain fact, then observing the fact will not give us new information. If we do not concurrently know a certain fact, then observing the fact will give us new information. As the above examples suggest, the notion of explicit knowledge characterized this way accords with the intuition that we have against the principle **LO** and avoid the problem of logical omniscience.

We may further illustrate the idea in relation to the notions of action and motivation. If I concurrently know whether 1571 is prime, then I would not obtain new information by observing that 1571 is prime. The idea that I have the same information before and after observing that 1571 can be explained by noting I would answer 'yes' to the question whether the number is prime before and after the observation. On the other hand, if the information that 1571 is prime is only implicit in my knowledge about arithmetic, I would not answer 'yes' to the question or at least would not be able to make my answer depend on whether the number is prime, as I want it to. However, once I observe that the number is prime, I would obtain new information. This can be explained by noting that I would now be able to say 'yes' to the question and make the action depend on whether the number is prime, as I want it to, even though I was not be able to do so before I made the observation. We explain this difference in my disposition to answer the question in terms of my informational after deducing that 1751.

Note that the last paragraph appeals to the notions of actions and motivations only for the purpose of illustrating our characterization of explicit knowledge. Indeed, our characterization itself does not involve the notions. However, it would be appropriate to articulate the nature of our project further here. As mentioned above, we agree with Stalnaker that action and motivation are important to the characterization of available knowledge and that they are indeed so intimately related in the way that they do not seem to allow formalizations of available knowledge. This is why we had to appeal to action and motivation in order to explain our characterization. What we intend to do by the above characterization, however, is to show that we can still reveal interesting aspects of explicit knowledge without invoking the notions of action and motivation as our theoretical primitives to be used in characterizations of knowledge. Thus, we are with Stalnaker, if he thinks that the full-fledged theory of knowledge would have to invoke the notion of action and motivation, but we are not, if he thinks that there is nothing that can be said about the desired notion of knowledge without considering action and motivation.

Once we characterize the notion of knowledge that does not suffer from the problem of logical omniscience, we can now meaningfully consider the situations where an agent makes deductive inferences. To describe the intuition that we appeal to in characterizing deductive inferences, let us review our prime number example. First, the fact that I can deduce the conclusion that 1571 is prime can be explained by my explicit knowledge about the relevant part of arithmetic. In order to make deductive inferences, I have to appeal to my knowledge about arithmetic. The parts of my knowledge about arithmetic that I use should be my explicit knowledge, when I indeed make the deductive inferences. Otherwise, it is not clear how we can explain the fact that I perform the calculations I do in the course of maing the deductive inferences. Second, by deducing the conclusion, I observe the conclusion of my deductive inference. My careful calculations reveal relevant arithmetical facts about the number 1571 and will present me the information that the number is prime. I observe, in the broad sense of the word, that 1571 is prime through the process of making deductive inferences.

Based on these considerations, we propose the following characterization of deductive inferences. When an agent makes a deductive inference,

1. the agent must explicitly know the premises of the inference and

2. by making a deductive inference, the agent observes the true information that the conclusion is the case.

Thus, we characterize deductive inferences as a process of observing the conclusions of the inferences based on explicit knowledge.

5.3 Formalizing Explicit Knowledge

We now formalize the notion of explicit knowledge basd on our characterization above.

5.3.1 Reinterpretation of TPAL

TPAL extends propositional logic with the epistemic operator $[i]$ and the public announcement operator $\langle !\theta \rangle$ (and its dual $[!\theta]$). For our current purpose, we will restrict our attention to the single agent case and denote the epistemic operator by K^i.[4] We start by giving interpretations of the operators in TPAL for our application.

1. $K^i\varphi$: "An agent implicitly knows φ.".

2. $\langle !\theta \rangle\varphi$: "An agent can observe θ after which φ is true."

3. $[!\theta]\varphi$: "After observing θ, φ is true.".

Also we assume that propositional letters refer to atomic propositions about the world, whose truth values are determined independently of the epistemic state of agents. Given that our purpose is to analyze the relevant epistemic concepts, such as information, explicit knowledge, observation, we take this familiar assumption to mark off the objects of our analysis from unanalyzable atomic propositions.

Some remarks are in order concerning the intended reading of the operator K^i. First, as the standard modal operator, the operator K^i validates the principle, *if* $K^i\varphi$ *and* φ *logically implies* ψ*, then* $K^i\psi$. This accords the way in which we conceive the notion of implicit knowledge. What we will do below is to define another notion of knowledge that does not validate the principle. Second, we interpret $K^i\varphi$ also as "an agent has the information that φ" or "an agent is informationally committed to φ". We consider these readings and the above reading equivalent for our current purpose.

[4]We will discuss the multi-agent case in Section 5.6.2

The models of TPAL are ETL-tree structures that represent temporal evolutions of initial epistemic models over informational updates by sequences of public announcements based on protocols assigned to worlds in the epistemic models. For the complete definitions of the models, we refer readers to Chapter 2. Here, we only describe how we interpret the models in TPAL to reflect the intended readings that we listed above.

First, epistemic models represent informational states of an agent by domains of points, indistinguishability relations, and valuation functions. Points in the domain of an epistemic model represent epistemic possibilities for an agent and the valuation function characterizes each epistemic possibility by determining truth values of atomic propositions. The indistinguishability relation represents what epistemic possibilities an agent considers at a given world.

Public announcements $!\varphi$ update the informational state of an agent by relativizing the epistemic possibilities in epistemic models to epistemic possibilities in which φ is true. By this mechanism, we represent the informational event of observations. When observing φ, an agent eliminates the possibility of non-φ. At this level of abstraction, our use of the word, "observation", should be taken in a broad sense. Not only is it intended to capture physical observations such as observing that it is raining, but also to capture other types of observations in a broad sense, such as observing things by reflection, etc., as we suggested in our previous discussions.

Protocols are sets of sequences of public announcements assigned to points in a given epistemic model. In general, they represent the information about what public announcements can happen in given points of the epistemic model. With our interpretation of public announcements, the protocol at a point represents what can possibly be *observed*. If $!p$ is in the protocol at a point, then p can be observed provided that p is true at the point. Similarly if $!p$ is not in the protocol at a point, then p cannot be observed even when p is true at the point.

Finally, based on our current interpretations, the models in TPAL represent temporal evolutions of agents' informational states over sequences of observations. A model $\mathcal{H}$ in TPAL is a triple of the form $(H, \sim, V)$. H is a set of *histories*, sequences of the form $w!\varphi_1 \ldots !\varphi_n$, where $w!\varphi_1 \ldots !\varphi_n$ represents the state at w after the sequence of observations $!\varphi_1 \ldots !\varphi_n$ have been made. $\sim$ represents the indistinguishability of those temporal states for an agent and V determines the truth of atomic propositions at each state. Truth of formulas in TPAL are defined with respect to a model $\mathcal{H} = (H, \sim, V)$ and a history h in $\mathcal{H}$. Here we give the truth definitions for the operators of our interest (For the complete list, readers are referred to Chapter 2.):

$$\mathsf{Forest}(\mathcal{M}, \mathsf{p}), h \models K^i\varphi \quad \text{iff} \quad \forall h' : h \sim h' \Rightarrow \mathsf{Forest}(\mathcal{M}, \mathsf{p}), h' \models \varphi.$$

$$\mathsf{Forest}(\mathcal{M}, \mathsf{p}), h \models \langle !\theta \rangle \varphi \quad \text{iff} \quad h!\theta \in H \text{ and } \mathsf{Forest}(\mathcal{M}, \mathsf{p}), h!\theta \models \varphi.$$

Having this interpretation of TPAL, we introduce the new operator for explicit knowledge, K, in the framework of TPAL, and define it by the following definition:

Definition 5.3.1 (Explicit Knowledge) Let $\mathcal{H} \in \mathbb{F}(\mathbb{PAL})$ and h, a history in $\mathcal{H}$. Then

$$\mathsf{Forest}(\mathcal{M}, \mathsf{p}), h \models K\varphi \quad \text{iff} \quad \forall h' : h \sim h' \text{ implies } \mathsf{Forest}(\mathcal{M}, \mathsf{p}), h' \models \langle !\varphi \rangle \top$$

◁

We read $K\varphi$ as "an agent explicitly knows φ". Given that $\langle !\varphi \rangle \top$ reads as "φ is observable", our definition of explicit knowledge says that an agent explicit knows that φ when the true information that φ is observable by the agent in all indistinguishable worlds.

This formulation of K^e has two immediate consequences. First, as is clear in the definition, the operator can be expressed by the language of TPAL. The following equivalence is an immediate consequence of Definition 5.3.1 and the truth definition of TPAL:

$$K^e\varphi \leftrightarrow K^i\langle !\varphi\rangle\top.$$

Therefore, we can obtain the completeness and decidability of the extension of TPAL with K^e by adding this equivalence to the axiomatization of TPAL (Section 2.3), based on the completeness and decidability results we derived in Chapter 2 (Section 2.3.3 and 2.3.2. Second, the formula $K^e\varphi \rightarrow K^i\varphi$ (explicit knowledge implies implicit knowledge) is valid in the semantics of TPAL, because $\langle !\varphi\rangle\top \rightarrow \varphi$ (the observability of the true information that φ implies the truth of φ). This coheres with the characterization of implicit knowledge as anything that follows from explicit knowledge. Explicit knowledge trivially follows from itself.

5.3.2 Dynamic Characterization of Explicit Knowledge

How does the formalization above capture our intuitive characterization of explicit knowledge in the previous section? To see how it does, let us consider the three models in Figure 5.1. The three models are constructed from an epistemic model that has two indistinguishable worlds w, v. In the model on the left, p and q are true only at w and v respectively, and the protocols at both worlds allow $!p$. Since w and v are indistinguishable, an agent does not know that p is true even in an implicit sense. However, after observing p, the world v is "eliminated" in the sense that $v!p$ is absent. This is simply because p is false at v and thus $v!p$ is not created even though $!p$ is in the protocol. In this sense, this model represents the case where the agent obtain *new* information by observing p.

The situation is different in the model in the middle. Here p is true at both w and

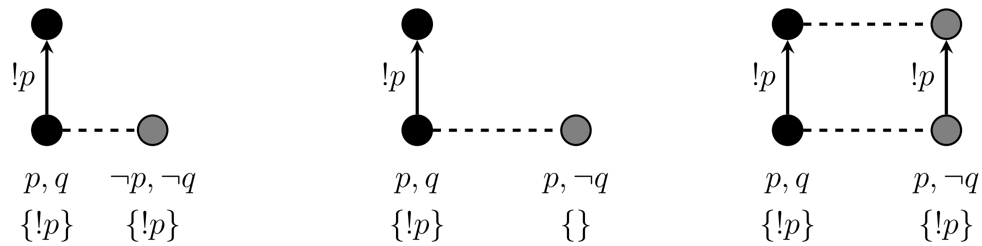

Figure 5.1: Dynamic characterization of explicit knowledge

v, but q is true only at w. Only the protocol at w allows $!p$. In this model, at w, it is already the case that an agent implicitly knows that p is the case. However, what is interesting about this model is that, even though the agent implicitly knows that p is true, she obtains new information by observing p. Indeed, at w, the agent does not implicitly know that q is true, but after observing p, i.e. at $w!p$, she does implicitly know that q is the case. The reason for this is that, in the model, $!p$ is not allowed in the protocol at v even though p is true. That is, at v, p is true but the fact that p is true is not observable. Thus, we can recapitulate what this example illustrate by saying: given a formula φ, *even when an agent implicitly know that* φ*, she may still gain information since she may be able to eliminate some possibilities based on the observability of the information.*

This situation should be contrasted with the model on the right. This model has the same propositional valuation as the model in the middle: p is true at both worlds; q is true only at w. However, in this model, the protocol allows $!p$ at both w and v. Thus, at w, an agent implicitly knows that p is true. Nonetheless, even after observing p, the agent still cannot distinguish w and v (more precisely $w!p$ and $v!p$), since in this model p is observable also at v. In general, for every φ, *if* φ *is observable at all indistinguishable worlds, then an agent does not gain new information in the sense that no world will be eliminated.*

This point explains the intuition of our definition of explicit knowledge. If an agent explicitly knows φ, then she does not obtain new information by observing φ

(the right model in Figure 5.1). If an agent does not explicitly know φ, then she obtain new information (the models on the right and the middle in Figure 5.1). By this dynamic characterization, our model goes beyond mere syntactic manipulations in formalizing the notion of explicit knowledge.

5.3.3 Epistemic Information and Protocol Information

The above explanation appeals to a specific notion of information that an agent has. We need to clarify what the notion is. To do so, we point to two kinds of information that are represented in models of TPAL. One kind of information in the models is represented by sets of points (or nodes), indistinguishability relations, and valuation functions. As we mentioned above, models in TPAL start with epistemic models and describe temporal evolutions of the models over informational updates—observations in our interpretation. Each observation creates a new set of points with the corresponding indistinguishability relation and valuation function. Thus, each level of tree structures in models of TPAL can be considered as an epistemic model and it represents an aspect of the information that an agent has. Pictorially, this is the aspect of the information of an agent that is represented 'horizontally' at each level of tree structures in the models discussed above. Let us call the kind of information *epistemic information*. The other kind of information that an agent has is the information about what information can be observed at worlds (or nodes). Protocols represent the information about what can be observable and an agent has information about them in addition to the epistemic information that she has. This aspect of the information of an agent is represented 'vertically' by the arrows coming out of each world in the models discussed above. Let us call this kind of information *protocol information*.

Given the distinction, we can now articulate our characterization of explicit knowledge. Our characterization was that an agent explicitly knows φ iff she would not obtain new information by observing φ. When we explained the 'gain' of information above, the sense of information that we appealed to was the epistemic information that an agent has. An agent gains epistemic information when the set of indistinguishable epistemic possibilities gets reduced. An agent does not gain new epistemic information when the set of indistinguishable epistemic possibilities stays the same. Therefore, we now need to restate the characterization of explicit knowledge as follows based on the distinction: *An agent i explicitly knows* φ *iff i would not obtain new* epistemic *information after observing* φ.

The following example also illustrates the importance of the distinction in our framework. Consider the following two schemas:

(Pos) $K^i\psi \wedge K^e\varphi \rightarrow [!\varphi]K^i\psi$

(Neg) $\neg K^i\psi \wedge K^e\varphi \rightarrow [!\varphi]\neg K^i\psi$

These two schemas together represent that, if an agent explicitly knows that φ, the information that she has, epistemic or protocol, stays the same after observing φ. These schemas are not valid in TPAL and counterexamples are given by Figure 5.2. At w, the instantiation of **Pos** with $\varphi := p$ and $\psi := p$ is false. (The antecedent is true since p is observable at w and v, but the consequent is false, since Kp is not true at $w!p$ by the non-observability of p at $w!p$ or $v!p$.) Similarly the instantiation of **Neg** with $\varphi := p$ and $\psi := q$ is false at w. These counterexamples are possible, since the protocol information may still change over the event of observing φ, even though her epistemic information is preserved. Thus, in general, an agent may obtain new explicit knowledge by obtaining new protocol information, or even lose his explicit knowledge by losing protocol information, even when her epistemic information kept preserved.

To ensure the validity of the above schemas, we need to make sure that the protocol information stays the same over events of making observations. The following schemas

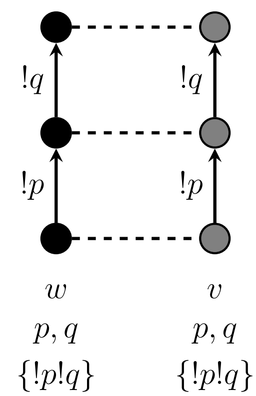

Figure 5.2: Counterexample

express such a condition.

(MT) $\langle !\varphi \rangle \top \to \langle !\psi \rangle \langle !\varphi \rangle \top$

(AMT) $\langle !\psi \rangle \langle !\varphi \rangle \top \to \langle !\varphi \rangle \top$

Indeed, we can prove the following:

Proposition 5.3.2 *Let $\mathcal{H}$ be a TPAL-model. If* **MT** *is valid in $\mathcal{H}$, then* **Pos** *is valid in $\mathcal{H}$. Also if* **AMT** *is valid in $\mathcal{H}$, then* **Neg** *is true at every h in $\mathcal{H}$.*

MT guarantees that, if it is true at every node in $\mathcal{H}$, $\mathcal{H}$ is *monotonic* in terms of observable information, that is, if φ is observable, then φ will be observable again after any information is observed. **AMT** guarantees that, if it is true at every node in $\mathcal{H}$, $\mathcal{H}$ is *anti-monotonic* in terms of observable information, that is, if φ is observable after any information is observed, then φ is observable before the information is observed.

5.3.4 Avoiding the Problem of Logical Omniscience

We shall also note that our formal definition of explicit knowledge extracts from the fact that the truth of TPAL-formulas containing public announcement operators depends on protocols, which are specified syntactically by TPAL-formulas (See

Chapter 2. As mentioned above, even if a formula φ is true at a world, $\langle !\varphi\rangle\top$ ($!\varphi$ is observable) will not be true unless $!\varphi$ is stipulated to be observable by the protocol assigned to the world. Also even if the true information that φ is observable, its logical consequences may not be, since protocols in general do not have to be logically closed (Even if φ is in a protocol and φ logically implies ψ, ψ may not be in the protocol) or closed by any specified condition.

Technically, this is exactly the feature that allows our K^e-operator to resolve the problem of logical omniscience. As mentioned above, the problem of logical omniscience arises, because the standard modal operators validate the K-axiom and necessitation rule. In our framework, the principles are validated with respect to the implicit knowledge operator K^i.[5] However, the principles are *not* valid with respect to the explicit knowledge operator K^e. It is straightforward to show the following.

Proposition 5.3.3 *The necessitation rule with respect to K^e,* if $\vdash \varphi$, then $\vdash K^e\varphi$ *is not sound in TPAL.*

Proposition 5.3.4 *The K-axiom with respect to K^e, $K^e\varphi \land K^e(\varphi \to \psi) \to K^e\psi$, is not valid in TPAL.*

The first proposition reflects the fact that, even if φ is true, the information that φ may not be observable. The second proposition follows from the fact that protocols are generally not closed under implication, as stated above. The readers are invited to generate counterexamples to the principles.

However, demanding that explicit knowledge have no closure property whatsoever may not sound too plausible in some contexts. For instance, we may think that, if an agent explicitly know a conjunction, then the agent also explicitly know each conjunct. Fortunately TPAL is flexible enough to characterize the closure properties of explicit knowledge (more precisely, closure of protocols, hence closure of explicit

[5] The schema $K^i\varphi \land K^i(\varphi \to \psi) \to K^i\psi$ is valid and the rule, *if* $\vdash \varphi$, *then* $\vdash K^i\varphi$, is sound.

knowledge). The validity of the following schema is equivalent respectively to the above closure properties:

$$\langle !(\varphi \wedge \psi)\rangle \top \rightarrow \langle !\varphi\rangle \top \wedge \langle !\psi\rangle \top$$

What the schema ensures may be pictorially described by saying that, in a given TPAL-model, if an arrow labeled by $!(\varphi \wedge \psi)$ comes out of a node, then two arrows labeled by $!\varphi$ and $!\psi$ respectively must come out of the node.

We can express more natural closure conditions, which may be properly assumed depending on the situations we describe. The following list are the pair of closure conditions and corresponding epistemic principles. The validity of a schema in a pair is equivalent to the validity of the other.

RF $\langle !\varphi\rangle \top \rightarrow \langle !K^e\varphi\rangle \top$.

- $K^e\varphi \rightarrow K^eK^e\varphi$.

EI $\langle !K^i\varphi\rangle \top \rightarrow \langle !\varphi\rangle \top$.

- $K^eK^i\varphi \rightarrow K^e\varphi$.

LI $\langle !\varphi\rangle \top \wedge \langle !(\varphi \rightarrow \psi)\rangle \top \rightarrow \langle !\psi\rangle \top$.

- $K^e\varphi \wedge K^e(\varphi \rightarrow \psi) \rightarrow K^e\psi$

VF If $\vdash \varphi$, then $\vdash \langle !\varphi\rangle \top$.

- *if* $\vdash \varphi$, *then* $\vdash K^e\varphi$

It is straightforward to show that the principles **LI** and **VF** are independent (in the sense that there are models in which only either one of the two is valid). Also they together imply **EI**. **RF** is independent from the three principles.

5.4 Formalizing Deductive Inference

Having defined explicit knowledge, let us now model situations where an agent makes a deductive inference. First, to express such situations, we extend our language with the operator $\langle \Gamma \vdash \varphi \rangle$, where Γ logically implies φ. The intended reading of $\langle \Gamma \vdash \varphi \rangle \psi$ is "an agent can deductively infer φ from Γ after which ψ". The dual of $\langle \Gamma \vdash \varphi \rangle$ is denoted by $[\Gamma \vdash \varphi]$, and $[\Gamma \vdash \varphi]\psi$ is read as "After deductively inferring φ from Γ, ψ." "After an agent makes a logical deductive from Γ to φ, ψ."

We will give the truth definition of this new operator based on the observation we made in Section 5.2. That is, when an agent makes a logical inference from Γ to φ,

1. the agent must explicitly know all the formulas in Γ, and
2. by making a deductive inference, the agent observes that the conclusion is the case.

The corresponding truth definition is as follows. Let $\mathcal{H}$ be a TPAL-model ($\mathcal{H} \in \mathbb{F}(\mathbb{PAL})$) and h a history in $\mathcal{H}$:

$$\mathcal{H}, h \models \langle \Gamma \vdash \varphi \rangle \psi \quad \text{iff} \quad \begin{array}{l} (1)\ \mathcal{H}, h \models K^e \chi \text{ for all } \chi \in \Gamma \\ (2)\ \mathcal{H}, h!\varphi \models \psi. \end{array}$$

Given that we consider deductive inferences made by realistic agents with limited resources, we may assume that Γ is finite. By this assumption, we can define the deductive inference operator in TPAL. (K^e is definable in TPAL, as we saw above):

$$\langle \Gamma \vdash \varphi \rangle \psi \leftrightarrow \bigwedge_{\chi \in \Gamma} K^e \chi \wedge \langle !\varphi \rangle \psi.$$

The semantics of the deductive inference operator can be visualized as in Figure 5.3. In the figure, we have two indistinguishable nodes, black and gray, and, at

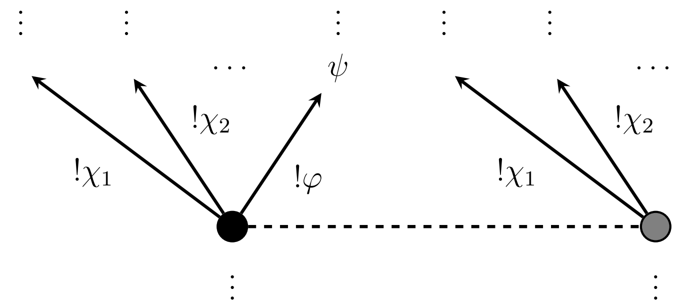

Figure 5.3: Making a deductive inference

the nodes, the formulas $\chi_1, \chi_2, \dots$ are observable. Therefore, we have $K^e\chi_i$ for each i. In addition, at the black node, φ is observable, and, after φ is observed, ψ becomes true. In this model at the black node, we have $\langle \chi_1, \chi_2 \dots \models \varphi \rangle \psi$ true.

This illustrates that our deductive inference operator represents a process of making a deductive inference as eliminating worlds where the conclusion is *not* observable. In the example, $\chi_1, \dots \chi_n$ and φ are all true at both the black and gray nodes. Nevertheless, an agent, observing that the conclusion φ is the case, gains further epistemic information (but not protocol information.) by eliminating the gray node where φ is not observable. This feature of our definition corresponds to the following intuition: *by deductive inference, an agent gains the information that the conclusion was observable.* The points in an epistemic model that will be left uneliminated after an agent makes a deductive inference are the ones where the conclusion is observable before an agent makes the inference.

5.5 Logical Omniscience vs Epistemic Closure

What is called *the epistemic closure principle* is the principle that states that knowledge is closed under logical implication. As we have discussed above, in epistemic logic, a version of this principle, i.e. *if φ is known and φ logically implies ψ, then ψ is* *known*, raises the problem of logical omniscience. We have investigated how to model the notion of knowledge, which we call explicit knowledge, that does not validate the version of the principle. The formalization of such a notion of knowledge makes it meaningful to pursue representations of deductive inference in epistemic logic and we have introduced a new operator $\langle \Gamma \vdash \varphi \rangle$ to represent a situation that an agent makes a deductive inference from Γ to φ.

Now our representation of deductive inferences allows us to address another perspective on the epistemic closure principle in the epistemology literature. In epistemology, the epistemic closure principle has been discussed as a principle expressing that we can *extend* our knowledge by logical inference. For instance, Williamson [81] presents the following formulation of the principle, which he calls *intuitive closure*:

> ... knowing $p_1, \ldots, p_n$, competently deducing q, and thereby coming to believe q is in general a way of coming to know q. (p.117)

He then says:

> We should in any case be very reluctant to reject intuitive closure, for it *is* intuitive. If we reject it, in what circumstances can we gain knowledge by deduction? (Williamson, pp.118, [81])

Those who defend the principle attempt to save this intuition about deduction as a good epistemic method, while those who attack the principle articulate why and how the intuition is sometimes betrayed.

This perspective on the closure principle is very contrastive to the perspective on the epistemic closure principle in epistemic logic. As we have discussed above, the epistemic closure principle is considered as the principle that assumes logically omniscient agents for whom deductive inferences are not meaningful. In [55], Stalnaker, discussing the problem of logical omniscience, says:

> Any context where an agent engages in reasoning is a context that is distorted by the assumption of deductive omniscience, since reasoning (at least deductive reasoning) is an activity that deductively omniscient agents have no use for... In fact any kind of information processing or computation is unintelligible as an activity of a deductively omniscient agent. (pp.428-9)

Thus, to highlight the contrast, one could say: on the interpretation in the philosophical literature on epistemology, the closure principle is something that would guarantee the unshakable epistemic value of deduction; on the interpretation in epistemic logic, it is something that would take away the epistemic value of deduction.

The reason that the situation has arisen may be characterized as follows. First, the problem of logical omniscience consists in the particular feature of the standard framework in epistemic logic. Therefore it is a task in epistemic logic, but not in epistemology as a whole, to resolve the problem. In the general epistemological context, the formulation of the closure principle, *if an agent knows φ and φ logically imply ψ, then she knows ψ*, must be simply abandoned for its immediate implausibility as a description of knowledge held by realistic agents like us. Second, while the nature of deductive inference have been investigated in philosophical discussions on the closure principle, the standard framework of epistemic logic does not represent deductive inference. In that case, there is no way to address the relevant aspect of the closure principle in the formal representation of knowledge in epistemic logic.

5.5.1 Formalizing the Epistemic Closure Principle

Having the representation of deductive inferences, we can now address the perspective on the epistemic closure principle in epistemology. We formulate the epistemic closure principle as discussed in the epistemology literature by:

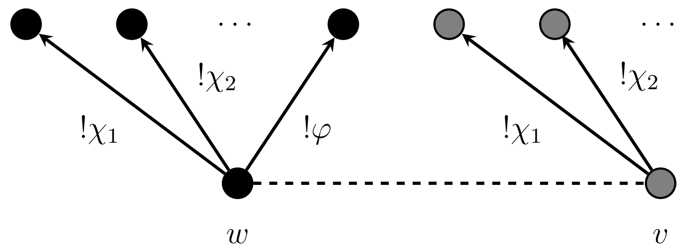

Figure 5.4: Counterexample to EC

EC

$$\bigwedge_{\chi \in \Gamma} K^e\chi \rightarrow [\Gamma \vdash \varphi]K^e\varphi$$

which reads "If an agent explicitly knows every formula χ in Γ, then, after making the deductive inference $\Gamma \vdash \varphi$, she explicitly knows φ."[6]

Despite the initial plausibility of the formulation, **EC** is not a valid formula in TPAL. A counterexample is visualized in Figure 5.4. (The dashed line between w and v represents the the indistinguishability relation between them as usual. The dashed lines are omitted between pairs of nodes, $w!\chi_i$ and $v!\chi_i$, for simplicity, even though they are indistinguishable.) Let $\Gamma := \{\chi_1, \dots, \chi_n\}$. At the node w of the model, $\bigwedge_{i=1}^{n} K^e\chi_i$ is true at w, since, for all i, $\langle \chi_i \rangle \top$ is true at w by the presence of the $!\chi_i$-arrow. However $[\Gamma \vdash \varphi]K^e\varphi$ is false. For $K^e\varphi$ is false at $w!\varphi$ given that $\langle !\varphi \rangle \top$ is false at $w!\varphi$ (by the absence of a $!\varphi$-arrow coming out of $w!\varphi$).

The reason of the failure of **EC** can be explained by the change of the protocol information over the events of making observation. (*cf.* **Pos** and **Neg** in Section 5.3.2) The cause of the change in protocol information may vary depending on situations. Here we give a possible interpretation of the model by one example from the literature in epistemology, given by Lawlor in [46]. The example can be described as follows:

Edward, being raised by a believer in homeopathy, believes in homeopathic

[6]Given the definition of the logical inference operator, the formulation is equivalent to $[\Gamma \vdash \varphi]K^e\varphi$.

> medicine. He has the strong conviction that he has seen illness cured by homeopathic treatments. Edward knows that the cold medicine he takes has a concentration of 1 part per 100^{200}. Recently he learned chemistry just recently, so he knows that, if a substance is diluted that much, it is unlikely that even one molecule of the substance remains. From this, he arrives through deduction at the belief that the cold medicine will very likely not work. Given the strength of his previous conviction in homeopathy, one could argue that he does not know that the cold medicine will very likely not work, until he goes through the process of questioning the efficacy of the homeopathic medicine. However, by epistemic closure, Edward necessarily has to know that the medicine will very likely not work.

In our model, let Γ be the set of propositions stating the relevant facts in chemistry, and φ, the proposition that the medicine will very likely not work. Initially, Edward explicitly knows the propositions in Γ, but not φ. After making the inference $\Gamma \vdash \varphi$, he still cannot possibly justify φ in the presence of the previous conviction. Thus in the sense that he cannot eliminate the possibility of non-φ, he cannot observe φ. Hence he does not (explicitly) know φ. There are other counterexamples raised in the literature (e.g. [28]), which involves the conflict between the belief arrived at by deduction and the previously held beliefs, as in this Edward example.

To block counterexamples to **EC**, we can add the schema **MT** to the axiomatization of TPAL:

$$\langle !\varphi \rangle \top \to \langle !\psi \rangle \langle !\varphi \rangle \top.$$

As stated in Section 5.3.2, this guarantees that, if a formula φ can be observed, then it will be observable after any information is observing. In the context of the Edward example, we may read this as saying “If some information is observable by an agent, there is nothing that prevents her from observing it after observing any information". This will block our counterexample, since information will stay observable once it becomes observable. Indeed it is straightforward to show the following:

Proposition 5.5.1 *The validity of* **MT** *implies the validity of* **EC**.

5.5.2 Independence

Discussions in the previous sections points out that the problem of logical omniscience in epistemic logic and debates about the epistemic closure principle in the philosophical literature deal with different aspects of the principle *knowledge is closed under logical implication*. By using our formulations of the problems, we can indeed show that principles that generate the problem of logical omniscience and the epistemic closure principles are logically independent.

First let us review the following principles and the corresponding facts:

MT $\langle !\varphi\rangle\top \rightarrow \langle !\psi\rangle\langle !\varphi\rangle\top$.

- The validity of **MT** implies the validity of **EC**.

LI $\langle !\varphi\rangle\top \wedge \langle !(\varphi \rightarrow \psi)\rangle\top \rightarrow \langle !\psi\rangle\top$.

- **LI** is equivalent to $K^e\varphi \wedge K^e(\varphi \rightarrow \psi) \rightarrow K^e\psi$

VF If $\vdash \varphi$, then $\vdash \langle !\varphi\rangle\top$.

- **VF** is equivalent to the principle *if* $\vdash \varphi$*, then* $\vdash K^e\varphi$

Given the above facts, the following simple results shows the independence between the principles that generate the problem of logical omniscience and the principle **EC**.

Proposition 5.5.2 (Consistency) **MT**, **LI** *and* **VF** *are consistent. That is, there is a model in which all the instances of the schemas are true at every node.*

Proposition 5.5.3 (Independence) **MT** *is independent from* **LI** *and from* **VF**, *i.e. the former pair does not imply the latter and that the latter does not imply the former.*

These facts can be shown by the models in Figure 5.5. First consider the model on the left. At w, φ and all of its logical consequences, $\tau_1, \tau_2 \ldots$, are observable (and nothing else). At $w!\varphi$, ψ and all of its logical consequences, $\tau'_1, \tau'_2, \ldots$ are observable (and nothing else). Also at every other nodes, all and only formulas provable in TPAL are observable. Now, suppose φ and ψ are identical with some provable formula. Then, at every node of the model, **MT**, **LI** and **VF** are all true, since all provable formulas are observable at every node and every node has the same observable formulas. Next, suppose φ and ψ are distinct propositional letter, say p and q. In this case, **LI** and **VF** are still true at every node, but **MT** are false at w.[7] Next consider the model on the right. In the model, p is observable at every node. This guarantees that **MT** is true at every node. However, no node satisfies **LI** or **VF**, since no node is closed under logical consequence. Therefore, **MT** does not imply logical omniscience and logical omniscience does not imply **MT**. (This fact will be mentioned in Section 5.5.)

By the two proposition, we can argue for the independence of the problem of logical omniscience and the validity of **EC**. First, by Proposition 5.5.2, there is a model in which **MT**, **LI** and **VF** are valid. In such a model, logical omniscience is present in the model and **EC** is valid. (**MT** guarantees that **EC**, and **LI** plus **VF** guarantees logical omniscience.) Second, by Proposition 5.5.3, there is a model in which **LI** and **VF** are valid, while **MT** is false at some node. In such a model, logical omniscience is present and **EC** is false. Third, by Proposition 5.5.3 again, there is a

[7] p is observable at w but not at $w!p$. Thus at w, $\langle !p\rangle\top$ but not $\langle !p\rangle\langle !p\rangle\top$. This means that **MT** is false at w.

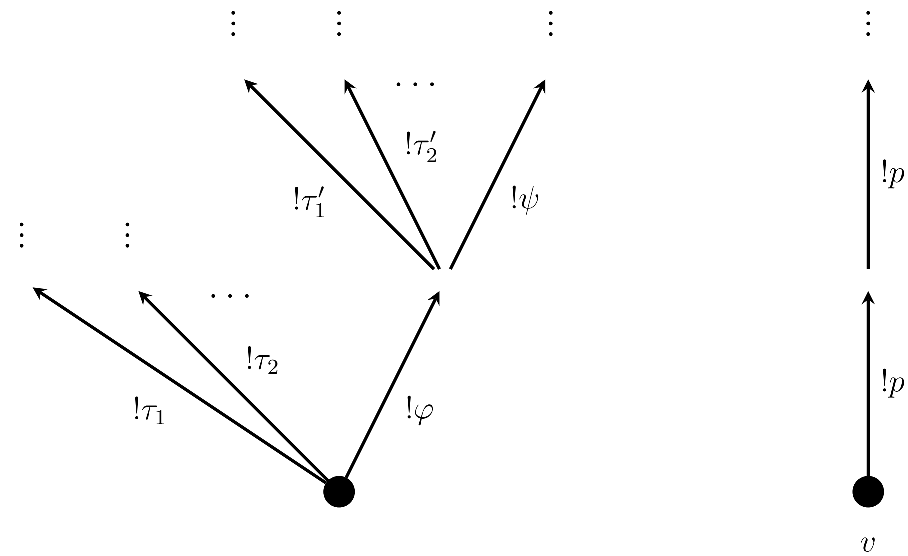

Figure 5.5: Independence of **MT** and **AMT** from **LI** and **VF**

model in which **MT** is valid, while **LI** and **VF** are false at a node. Finally, in general, **MT**, **LI** and **VF** are not valid in TPAL. Therefore, we claim that *the presence or absence of logical omniscience does not determine the question about* **EC***, and the truth or falsity of* **EC** *does not determine the question about logical omniscience.*

Here one may object that the formulation of **EC** does not fully capture the epistemic closure principle discussed in epistemology. In particular, consider a formulation of the epistemic closure principle in [81], mentioned in the introduction:

> ...knowing $p_1, \ldots, p_n$, competently deducing q, and thereby coming to believe q is in general a way of coming to know q.

At least, it is not clear how **EC** captures the part *thereby coming to believe* q. Other formulations, e.g. in [28, 35, 46], include similar conditions. In fact, the part in question is essential since it avoids counterexamples in which an agent does not arrive at the belief of the conclusion given her previously held beliefs and ones in which the belief of the conclusion is arrived at based on some unreliable methods independent from the deduction from known premises.

In fact, the objection rightly points out that **EC** does not completely capture standard formulations of the epistemic closure principle in the relevant literature. It is true that our framework does not capture the notion of belief in contrast with knowledge, let alone the ways by which beliefs are arrived at.

However our main point in the current paper is to highlight that the literature on the problem of logical omniscience and debates on the epistemic closure principle in epistemology take different perspectives on the principle *knowledge is closed under logical implication*. We did this by providing a single framework in dynamic epistemic logic that characterizes the notions of explicit knowledge and deductive inference. We defined explicit knowledge by the observability at all indistinguishable worlds, and showed that the principles that give rise to logical omniscience (the necessitation rule and the K-axiom) fail with respect to the defined notion. Also we defined the epistemic closure principle as a distinct dynamic principle, *if an agent explicitly knows the premises, then she will explicitly know the conclusion after observing that the conclusion is the case.* This way, we could show that the two problems are independent in the sense that the validity of one does not imply the other. These points would not be taken away by the objection in question. Indeed, as is illustrated in the Edward example, our claims do not depend on the fact that the notion in question *thereby coming to believe* is left out of the formulation. The example would work out even when the notion is considered.

5.6 Concluding Discussions

We have formalized the notions of explicit knowledge and deductive inference in the framework of TPAL. We defined explicit knowledge by the information observable at all indistinguishable worlds. The syntactic character of the notion avoids the problem of logical omniscience and the dynamic character allows us to formulate the principle that, when an agent explicitly knows φ, then she does not obtain new information after observing that φ. We also defined the notion of logical inference as a dynamic notion of observing, on suitable conditions, that the conclusion is the case. This enabled us to express the epistemic closure principle as a dynamic principle. We showed that this dynamic principle is independent from the principles that generate the problem of logical omniscience. Now we conclude the paper by discussing future research directions.

5.6.1 Comparison with Other Systems

Various systems have been developed in epistemic logic in order to represent the notions of explicit knowledge and deductive inference. Quite a few systems among those systems have some syntactic elements in their semantic models. In the sense that protocols are also syntactic entities, our system has similarities with those frameworks. Comparison between our system and those will be useful.

On explicit knowledge, a list of similar systems includes van Benthem [66, 65], Fagin and Halpern [22] and Velazquez-Quesada [75]. The basic idea of these systems is to assign a set of formulas to each state in a given epistemic model. These sets, let us call accessibility sets, represent the information to which an agent have internal access. The general aspects of their models can be captured by a quadruple $\mathcal{M} = (W, \sim, V, I)$, where $(W, \sim, V)$ is an epistemic model and I is a function assigning a set of formulas to each point in W. With this setting, the notion of explicit knowledge, say $E\varphi$, is defined by something along the following line:

$$\mathcal{M}, w \models E\varphi \quad \text{iff} \varphi \in I(w) \text{ and } \forall v : w \sim v \Rightarrow \mathcal{M}, v \models \varphi.$$

Since our protocols are also sets of (sequences of) formulas assigned to each point in a given epistemic model, these approaches are very similar to our system.

Nonetheless there are differences. First, in those models, although accessibility sets distinguish what is explicitly known from what is implicitly known, they do not contribute to further characterizations of explicit knowledge. In contrast, in our model, assigned formulas receive dynamic interpretations. Each formula corresponds to the operation of eliminating the worlds where the formula is false. This character of our system allows us to add an intended aspect of explicit knowledge to our formalization, that is, if an agent explicit knows φ, she does not obtain further information by accessing φ.

Second, those systems, in particular those of van Benthem and Velazquez-Quesada, consider explicit ways of updating accessibility sets. One kind of operators they consider are the ones that add new formulas to initially assigned accessibility sets. On the other hand, in the current paper, we did not consider such operations. In our framework, there are two ways to model operations of the kind. One way is to appeal to the temporal evolution of accessible information along given protocols. What is accessible at a given point h and what is accessible after a given information φ is accessed at h are both determined by an assigned protocol. Addition of formulas Φ to accessible information at φ can be represented in a given model by setting the accessible information at $h!\varphi$ so that it extends the accessible information at h by formulas in Φ. Another way is to represent the operations by update operations to protocols. By the operations, we add specified formulas to given protocols in suitable ways and this gives the desired operations.

On logical inference, a list of similar systems include van Benthem [66, 65], Duc [16, 17, 18], Jago [43, 44], and Velazquez-Quesada [75]. By appealing to the models with certain forms of syntactic accessible sets, they represent logical inferences in two distinct ways. Duc and Jago represent logical inference as transitions from a given point, say w, to another, say v. The accessibility set at v expands that at w

with the conclusion of corresponding logical inferences. In contrast, van Benthem and Velazquez-Quesada represent logical inferences by updating accessibility sets by their conclusions. When an agent makes logical inferences, formulas expressing the conclusions are added to accessibility sets.

The basic intuition of our system goes in the middle of these two approaches. As the systems of Duc and Jago, logical inferences in our system correspond to temporal transitions from a node to another. By accessing the information that the conclusion is the case, we go along an arrow and move to another node where there is another set of accessible information. As the systems of van Benthem and Jago, these temporal transition in our system correspond to updates of initial epistemic states. By accessing the information, worlds in conflict with the information are eliminated.

Beyond these abstract considerations, more precise comparison between these systems and ours remains to be investigated. The presence of the mentioned differences seems to be relatively minor and it seems to be a promising project to try to imitate those systems in our TPAL framework and enrich our system by importing different essences from those systems.

5.6.2 Extension to Multi-agent with TDEL

In developing our system, we restricted ourselves to the single agent case. Thus it is natural to ask whether the system can be extended to the multi-agent case. One way that suggests itself here is to simply expand initial epistemic models with indistinguishability relations assigned to multiple agents. Such epistemic models will be of the form $(W, \sim_1, \ldots, \sim_n, V)$ where each $\sim_i$ corresponds to the indistinguishability relation for an agent i. However, this approach does not give intended results automatically. The reason is that, in the above models, we consider the operations of *public announcements* as the events of making observations. If we keep this setting for the multi-agent case, every agent must make the same observation when a single agent observes it. Public announcement operations eliminate worlds in conflict with the information from the model and thus such worlds are eliminated from indistinguishability relation of every agent.

Therefore we need to appeal to operations that allow agents to observe true information independently from the other agents. Fortunately DEL provides a way to represent such operations. In addition, ETL tree models can be generated, in a way similar to TPAL ([68]), with respect to the class of such operations and a logic over those models turn out to be manageable ([42]). Upon these developments in DEL, a precise form multi-agent version of our system remains to be seen.

Conclusion

In this dissertation, we have developed a formal framework that captures two important aspect of intelligent interaction, epistemic dynamics and protocol information. The framework merges the two major systems in the literature, Dynamic Epistemic Logic (DEL) and Epistemic Temporal Logic (ETL). The main idea of the framework is to construct time-branching tree structures by successively updating an epistemic model based on protocols assigned to its states. Product update and event models generate correct representations of agents' informational states at each moment and time-branching tree structures represent possible temporal evolutions of the states that are permitted by assigned protocols.

We have studied the framework from three perspectives. First, in generating ETL models from models in DEL, the framework can provide a 'bridge' between DEL and ETL. From this perspective, we have given a systematic comparison between DEL and ETL in Chapter 1. Our main result is the representation theorem that characterizes the class of DEL-generated ETL models as the class of ETL models with propositional stability, synchronicity, perfect recall, and uniform no miracle. Second, our framework provides a reinterpretation of the language of DEL. From this perspective, we have studied logics over classes of DEL-generated ETL models in Chapter 2. Furthermore, we have considered various extensions of the logics in Chapter 3. Third, by representing epistemic dynamics and protocol information in one system, our framework can work as a powerful modelling tool. From this perspective, we have made philosophical applications of our system in Chapter 4 and 5. In the former chapter, we have dealt with the knowability paradox and given a fine-grained logical analysis of the knowability thesis. In the latter chapter, we have represented explicit knowledge and deductive inference.

Our investigations of the framework in this dissertation suggest further research topics from each of these perspectives. First, our model constructions may be extended to the systems that describe beliefs. In this dissertation, we have mainly dealt with DEL, which is designed to describe knowledge. However, in the literature, systems that describe beliefs have been studied (e.g. [7, 64]). [67] takes this perspective and merge Dynamic Doxastic Logic and Doxastic Temporal Logic. Also, in introducing syntactic structures by protocols, our framework shares some similarities with other related systems, such as Justification Logic (e.g. [1, 3, 2, 24]), Logic of Awareness (e.g [21, 22]). The exact relation between our system and those systems remains to be seen.

Second, our study of logics in TDEL has left several open questions. Two major kinds of questions concern computational complexity and extensions of TDEL. On computational complexity, we have investigated the systems, TPAL and TDEL(X), and have shown that they are decidable. However, the exact complexities of these systems are not known. Furthermore, for the extended systems, including TAPAL and TDEL(X)+P, the decidability question is open. On extensions of TDEL, we have seen that some extensions of TPAL could not be extended in a straightforward way to TDEL(X). For instance, the axiomatization of TDEL with generalized event operators is left open. Also, the method to generalize TDEL in order to permit preconditions containing future operators remains to be developed.

Third, possible applications of our framework are not limited to the ones we have given in this dissertation. In epistemology, recent discussions have highlighted important features of various epistemic concepts, for which much remains to be done from a formal perspective. For instance, contextual aspects of knowledge have been widely discussed in the literature. The question of how to best represent such aspects in formal systems remains to be investigated. It is interesting to see if protocols in our framework may give a reasonable representation for contexts of knowledge attribution. Also our framework can provide further modeling tools for the fields of studies beyond philosophy. Artificial intelligence and game theory are prime examples of the kind of disciplines. Other related topics include cryptography ([72]), learning theory, etc.

Bibliography

[1] S. Artemov. Proceedings of the 11th european conference on logics in artificial intelligence. In *Lecture Notes in Artificial Intelligence*, volume 5293, pages 1–4. Springer-Verlag, 2008.

[2] S. Artemov and E. Nogina. Basic epistemic logic with justification. Technical Report TR-2005004, CUNY Ph.D. Program in Computer Science, 2005.

[3] S. Artemov and E. Nogina. On epistemic logic with justification. In R. van der Meyden, editor, *Theoretical Aspects of Rationality and Knowledge. Proceedings of the Tenth Conference*, pages 279–294, Singapore, June 10-12 2005.

[4] R. Aumann. Agreeing to disagree. *Annals of Statistics*, 4(6):1236–1239, 1976.

[5] P. Balbiani, A. Baltag, H. van Ditmarsch, A. Herzig, T. Hoshi, and de Lima. Arbitrary announcement logic. *Review of Symbolic Logic*, 2008. To appear.

[6] A. Baltag, L. Moss, and S. Solecki. The logic of public announcements, common knowledge and private suspicions. In I. Gilboa, editor, *Proceedings of the 7th Conference on Theoretical Aspects of Rationality and Knowledge*, pages 43–56, 1998.

[7] A. Baltag and S. Smets. Dynamic belief revision over multi-agent plausibility models. In G. Bonanno, W. van der Hoek, and M. Wooldridge, editors, *Logic*

and the Foundations of Game and Decision Theory: Proceedings of LOFT'06, Texts in Logic and Games, pages 11–24. Amsterdam University Press, 2006.

[8] J. C. Beall. Fitch's proof, verificationism, and the knower paradox. *Australasian Journal of Philosophy*, 78:241–247, 2000.

[9] N. Belnap and M. Perloff. Seeing to it that: a canonical form for agentives. In Jr. et al H. E. Kyburg, editor, *Knowledge Representation and Defeasible Reasoning*, pages 167–190. Kluwer, Dordrecht, 1990.

[10] P. Blackburn, M. de Rijke, and Y. Venema. *Modal Logic*. Cambridge University Press, Cambridge, 2001.

[11] G. Bonanno. Memory and perfect recall in extensive games. *Games and Economic Behaviour*, 47:237–256, 2004.

[12] Giacomo Bonanno. Belief revision in a temporal framework. In G. Bonanno, W. van der Hoek, and M. Wooldridge, editors, *Logic and the Foundations of Game and Decision Theory: Proceedings of LOFT'06, Texts in Logic and Games*, pages 43–50. Amsterdam University Press, 2006.

[13] J. Broersen. A complete STIT logic for knowledge and action and some of its application. In *Proceedings DALT@AAMAS08*, Lecture Notes in Artificial Intellligence. Springer, 2008. To appear.

[14] B. Brogaard and J. Salerno. Fitch's paradox of knowability. *Stanford Encyclopedia of Philosophy*, 2002. http://plato.stanford.edu/entries/fitch-paradox/.

[15] B. F. Chellas. Time and modality in the logic of agency. *Studia Logica*, 51:485–517, 1992.

[16] H. Duc. In *Proceedings of the 7th Portuguese Conference on Artificial Intelligence, volume 990 of Lecture Notes in Computer Science*, pages 237–248. Springer, 1995.

[17] H. Duc. Reasoning about rational, but not logically omniscient, agents. *Journal of Logic and Computation*, 7(5):633–648, 1997.

[18] H. Duc. *Resource-Bounded Reasoning about Knowledge*. PhD thesis, Institut für Informatik, Universtät Leipzig, 2001.

[19] M. Dummett. Victor's error. *Analysis*, 61:1–2, 2001.

[20] D. Edgington. The paradox of knowability. *Mind*, 94:557–568, 1985.

[21] R. Fagin, J. Halpern, Y. Moses, and M. Vardi. *Reasoning about Knowledge*. Synthese Library. MIT Press, Boston, 1995.

[22] R. Fagin and J. Y. Halpern. Belief, awareness and limited reasoning. *Artificial Intelligence*, 34:39–76, 1988.

[23] F. Fitch. A logical analysis of some value concepts. *Journal of Symbolic Logic*, 28:135–142, 1963.

[24] M. Fitting. The logic of proofs, semantically. *Annals of Pure and Applied Logic*, 132(1):1–25, 2005.

[25] T. French and H. van Ditmarsch. Undecidability for arbitrary announcement logic. AiML, 2008.

[26] J. Gerbrandy. *Bisimulations on Planet Kripke*. PhD thesis, ILLC, 1999.

[27] J. Gerbrandy and W. Groeneveld. Reasoning about information change. *Journal of Logic, Language, and Information*, 6:147–169, 1997.

[28] S. Hales. Epistemic closure principles. *Southern Journal of Philosophy*, 33:185–201, 1995.

[29] J. Halpern and M. Vardi. The complexity of reasoning about knowledge and time. *Journal of Computer and System Sciences*, 38:195–237, 1989.

[30] M. Hand. Knowability and epistemic truth. *Australasian Journal of Philosophy*, 81(2):216–228, 2003.

[31] M. Hand and J. Kvanvig. Tennant on knowability. *Australasian Journal of Philosophy*, 77:422–428, 1999.

[32] D. Harel. Dynamic logic. In D. Gabbay and F. Guenther, editors, *Handbook of Philosophical Logic: extensions of classical logic*, volume 2, pages 497–604. Reidel Publishing Company, 1984.

[33] D. Harel, D. Kozen, and J. Tiuryn. *Dynamic Logic, Foundations of Computing*. MIT Press, Cambridge, 2000.

[34] W. D. Hart and C. McGinn. Knowledge and necessity. *Journal of Philosophical Logic*, 5:205–208, 1976.

[35] J. Hawthorne. Is knowledge closed under known entailment? the case for closure. In M. Setup and E. Sosa, editors, *Contemporary Debates in Epistemology*, pages 26–43. Blackwell, 2005.

[36] V. Hendricks. *Mainstream and Formal Epistemology*. Cambridge University Press, New York, 2006.

[37] J. Hintikka. *Knowledge and Belief: An Introduction to the Logic of the Two Notions*. Cornell University Press, Cornell, 1962.

[38] I. Hodkinson and M. Reynolds. Temporal logic. In P. Blackburn, J. van Benthem, and F. Wolter, editors, *Handbook of Modal Logic*. Elsevier, 2006.

[39] T. Hoshi. The knowability paradox and dynamics of knowledge. Philosophy Department, Stanford University, 2008.

[40] T. Hoshi. Logical omniscience and epistemic closure. Philosophy Department, Stanford University, 2008.

[41] T. Hoshi. Logics of public announcement with constrained protocols. In *LOFT*, 2008.

[42] T. Hoshi and A. Yap. Etl, del and past operators. In J. van Benthem and E. Pacuit, editors, *Proceedings of the Workshop on Logic and Intelligent Interaction*, pages 132–142, 2008.

[43] M. Jago. *Logics for Resource-Bounded Agents*. PhD thesis, Department of Philosophy, University of Nottingham, 2006.

[44] M. Jago. Epistemic logic for rule-based agents. *Journal of Logic, Language and Information*, 18:131–158, 2009.

[45] J. Kvanvig. The knowability paradox and the prospects for anti-realism. *Nous*, 29:481–499, 1995.

[46] K. Lawlor. Living without closure. In M. Blaauw, editor, *Epistemological Contextualism*, pages 25–49. Rodopi, 2005.

[47] H. J. Levesque. A logic of implicit and explicit belief. In *Proceedings of the Seventh International Joint Conference on Artificial Intelligence (IJCAI 1981)*, pages 240–245, 1981.

[48] D. Lewis. *Convention: A Philosophical Study.* Harvard University Press, Cambridge, 1969.

[49] J. S. Miller and L. S. Moss. Undecidability of iterated modal relativization. *Studia Logica*, 79(3):373–407, 2005.

[50] E. Pacuit and R. Parikh. Reasoning about communication graph. In J. van Benthem, D. Gabbay, and B. Löwe, editors, *Proceedings of the 7th Augustus de Morgan Workshop*. King's College Press, 2003.

[51] R. Parikh and R. Ramanujam. A knowledge based semantics of messages. *Journal of Logic, Language, and Information*, 12:453–467, 2003.

[52] P. Percival. Fitch and intuitionistic knowability. *Analysis*, 50:182–187, 1990.

[53] J. Plaza. Logics of public announcements. In *Proceedings of the 4th International Symposium on Methodologies for Intelligent Systems: Poster Session Program*, pages 201–216, 1989.

[54] K. Sim. Epistemic logic and logical omniscience. a survey. *International Journal of Intelligent Systems*, 12(1):57–81, 1997.

[55] R. Stalnaker. The problem of logical omniscience, I. *Synthese*, 89:425–440, 1991.

[56] R. Stalnaker. The problem of logical omniscience II. In *Context and Contents*, pages 255–278. Oxford University Press, 1999.

[57] N. Tennant. *The Taming of the True.* Oxford University Press, USA, 1997.

[58] N. Tennant. Is every truth knowable? Reply to Hand and Kvanvig. *Australasian Journal of Philosophy*, 79:107–113, 2001.

[59] N. Tennant. Is every truth knowable? Reply to Williamson. *Ratio*, XIV:263–280, 2001.

[60] N. Tennant. Victor vanquished. *Analysis*, 62:135–142, 2002.

[61] J. van Benthem. Games in dynamic epistemic logic. *Bulletin of Economic Research*, 53:216 – 248, 2001.

[62] J. van Benthem. One is lonely number: On the logic of communication. Technical Report PP-2002-27, University of Amsterdam, 2002.

[63] J. van Benthem. What one may come to know. *Analysis*, 64(2):95–105, 2004.

[64] J. van Benthem. Dynamic logic for belief revision. *Journal of Applied Non-Classical Logics*, 17(2):129–155, 2007.

[65] J. van Benthem. Merging observation and access in dynamic logic. *Journal of Logic Studies*, 1(1):80–90, 2008.

[66] J. van Benthem. Tell it like it is. *Journal of Beking University*, Humanities and Social Science Edition(1):80–90, 2008.

[67] J. van Benthem and C. Dègremont. Building bridges between dynamic and temporal doxastic logics. Technical Report PP-2008-17, Institution for Logic, Language & Computation, 2008.

[68] J. van Benthem, J. Gerbrandy, T. Hoshi, and E. Pacuit. Merging frameworks for interaction. *Journal of Philosophical Logic*, 2008. To appear.

[69] J. van Benthem and F. Liu. Diversity of logical agents in games. *Philosophia Scientiae*, 8(2):163–178, 2004.

[70] J. van Benthem and E. Pacuit. The tree of knowledge in action: Towards a common perspective. In G. Governatori, I. Hodkinson, and Y. Venema, editors, *Proceedings of Advances in Modal Logic*, volume 6. King's College Press, 2006.

[71] J. van Benthem, J. van Eijck, and B. J. Kooi. Logic of communication and change. *Information and Computation*, 204(11):1620–1662, 2006.

[72] H. van Ditmarsch. The Russian cards problem. *Studia Logica*, 71:1–32, 2003.

[73] H. van Ditmarsch, J. Ruan, and L. Verbrugge. Sum and product in dynamic epistemic logic. *Journal of Logic and Computation*, 2007. to appear.

[74] H. van Ditmarsch, W. van der Hoek, and B. Kooi. *Dynamic Epistemic Logic*, volume 337 of *Synthese Library*. Springer, 2007.

[75] F. Velazquez-Quesada. Inference and update. Technical Report PP-2008-48, ILLC, 2008.

[76] T. Williamson. Intuitionism disproved? *Analysis*, 42:203–207, 1982.

[77] T. Williamson. On knowledge of the unknowable. *Analysis*, 47:154–8, 1987.

[78] T. Williamson. On the paradox of knowability. *Mind*, 96:256–61, 1987.

[79] T. Williamson. Knowability and constructivism. *Philosophical Quarterly*, 38:422–432, 1988.

[80] T. Williamson. On intuitionistic modal epistemic logic. *Journal of Philosophical Logic*, 21:63–89, 1992.

[81] T. Williamson. *Knowledge and Its Limits*. Oxford University Press, USA, 2000.

[82] T. Williamson. Tennant on knowable truth. *Ratio*, 13:99–114, 2000.

[83] A. Yap. Dynamic epistemic logic and temporal modality. University of Victoria, 2007.

Index