FORMAL METHODS WITH DYNAMIC AGENT SAFETY LOGIC

A Thesis presented to

the Faculty of the Graduate School

at the University of Missouri

In Partial Fulfillment

of the Requirements for the Degree

Doctor of Philosophy

---

by

Seth Ahrenbach

Dr. Rohit Chadha, Thesis Supervisor

MAY 2019

The undersigned, appointed by the Dean of the Graduate School, have examined the dissertation entitled:

FORMAL METHODS WITH
DYNAMIC AGENT SAFETY LOGIC

presented by Seth Ahrenbach,

a candidate for the degree of Doctor of Philosophy and hereby certify that, in their opinion, it is worthy of acceptance.

Dr. Rohit Chadha

Dr. Guilherme DeSouza

Dr. Alwyn Goodloe

Dr. William Harrison

Dr. Paul Weirich

ACKNOWLEDGMENTS

This page is where you would acknowledge all those who helped you with your academic research.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Figure Page

ABSTRACT

This is the abstract of your dissertation project. It should not exceed one page.

Chapter 1

Introduction

In this doctoral thesis, we present a logic for reasoning about the connection among four aspects of human agency: knowledge, belief, mere action, and safe action. A mere action is an action that has either been performed or not; its evaluation as a good or bad action, relative to some normative domain, is not considered. A safe action, on the other hand, is a mere action that is evaluated in a safety domain. Other normative domains could be considered without much change to the logic being required, but in this dissertation I confine myself to the domain of aviation safety.

Knowledge and belief are aspects of human agency that have been well-studied by philosophers, logicians, and computer scientists. Epistemology is a core field of philosophy, studying what knowledge is, and logicians have formalized these concepts. Computer scientists have taken these formalized systems and applied them to computational domains ranging from artificial intelligence to information security. The intersection of these fields at the knowledge nexus provides a rich variety of topics to study.

Similarly, the study of action has a history of richness at the intersection of philosophy, logic, and computer science. The relationship between actions and knowledge in particular finds rich application in the fields of economics and game theory. The logics for reasoning about these things have been actively developed over the past several decades, with a species of dynamic modal logic coming to the forefront as particularly well-suited to the task.

Dynamic logic itself originated in computer science as a way to formally study the executions of programs. Philosophers and logicians then borrowed it back and married it with epistemic logic in order to generate logics of knowledge and action. The exchange of ideas between fields often occurs through the logics they use to study their subjects.

This dissertation picks up the thread from philosophical logic, specifically the efforts to use dynamic epistemic logic to model behavior in the rarefied world of games and toy examples. There researchers have made progress formalizing simple scenarios where one agent announces to the other something, and this causes the other agent's knowledge to increase. We extend that logic here in two ways. First, we experiment with its application to real humans piloting aircrafts. The environment is more complex, and challenges the logic to become a more realistic representation of human knowledge. To address this, we weaken the axioms of knowledge, and include axioms of belief and the relationship between knowledge and belief. The second extension concerns the normative aspect of actions. Dynamic epistemic logics impose preconditions on actions that determine whether they can be performed. We extend this idea with another precondition governing whether the action can be safely performed. It is a simple idea that opens up a wide new world of expressive power for the logics, as the underlying normative precondition function can be almost any normative aspect one cares to consider. One could swap out our safety precondition function with an ethical precondition function, or a legal precondition function, and develop similar dynamic logics of ethical or legal agency. Those applications are beyond the scope of this dissertation, however.

In order to validate the above extensions, we rely on rigorous tools to check our math, so to speak. Many other researchers have made great strides in mechanizing the proof and model theories of modal logics in the interactive theorem proving tool Coq. We follow their lead and similarly extend their efforts to bear out the logic we develop in the target application domain. We modify a deep embedding of dynamic epistemic logic in Cog to create a deep embedding of our logic. We verify the theorems of the object language in the target domain in Coq, as well as the soundness and completeness proofs in the metalanguage in Coq. While this is not a novel approach to logic research, it represents a more rigorous and computational approach to the methods of logic research that help move the dissertation from the merely philosophical domain and into the computer science domain. We do not develop any new theorem proving tools or techniques in the thesis, but our use of the interactive theorem prover itself is a feature that distinguishes this dissertation as one of computer science.

We have chosen the target domain of aviation safety, and it has a nice property that makes it a good candidate domain for extending the logic. Rather than modelling a wide array of behaviors and information that apply to human behavior generally, we need only model the behavior and information that relates to flying an airplane. We need to model the instrument readings and the pilot's inputs to the flight controls. This significantly cuts down the expressive space we need to handle. However, it is still a real human in a real world situation, requiring the logic to relax certain assumptions about the reasoning power of the agents and the information they have about the state of the world. Instrument readings can be conflicting, and pilots can be unaware of this conflict. This is not typically a situation logicians have tried to model in the past. Additionally, aviation safety is already a domain familiar with the use of formal methods to increase assurance, so the approach is not so novel as to be completely foreign to the target domain's researchers. This gives us related work to compare and contrast the present efforts to.

In particular, various logics exist for modelling safety properties of avionics systems, and tools for verifying that systems satisfy those properties. Aviation historically has been one of the few domains where the cost, perceived or actual, of formal verification is justified by the risks associated with system failure. We have engaged with this research community to some extent through collaborative research, conference presentations, and journal articles, and have received very helpful feedback. In some ways the approach adopted in this dissertation is too heavy handed for the particular needs of formal methods researchers in aviation safety. Rushby makes this case particularly well in his response to Ahrenbach and Goodloe [6]. Many of the benefits achieved by the approach pursued here can be had through more straight forward integrations of modal logic with existing formal verification frameworks. Rushby achieves analogous results with merely a modal operator for a pilot's belief sufficing, with the belief treated as an input variable to the verification system just like any other.

However, our purposes here differ slightly from those of researchers focused on the aviation domain. An aviation-focused researcher asks 'what is the best way to use this formal method in order to make aviation safer?', while we ask, 'What domains are suitable proving grounds for this new general formal method?' We seek a level of generality beyond aviation, and so in some cases are unwilling to sacrifice the heavy machinery for the benefit of having a simpler pilot monitoring logic. We seek to develop a general logic of realistic agency, and we validate it in the proving ground of aviation safety. Just as a geneticist is interested in *Drosophila melanogaster* (fruit flies), or an artificial intelligence researcher is interested in chess, because these subject (or target) domains are areas to apply new techniques and test new approaches [29, 34][1].

The new approach pursued here, as mentioned earlier, is to relax the assumptions

[1]In his book about the building of *Deep Blue*, Hsu decries being called an AI researcher, and prefers his work to be seen as Very Large Scale Integration (VLSI) research, but the point stands that chess to him was a target domain useful as a proving ground for his primary research interest.

about agent rationality that other philosophical logics make, while augmenting the dynamic modality of action to include considerations of safety. If all goes well, this yields a logic for reasoning about realistic agents performing safe or unsafe actions, and about their beliefs and knowledge about those actions.

The logic, which I call Dynamic Agent Safety Logic ($\mathcal{DASL}$), is based on the logical foundations of game theory, in which models of agency formally capture how knowledge, rationality, and action relate to each other. Game theory presents a model that, given a description of a scenario, allows one to deduce what actions are dictated by a given theory of rationality. The standard game-theoretic inference works as follows:

$$\textit{Knowledge\_of\_Situation} \wedge \textit{Rationality} \Rightarrow \textit{Good\_Action}.$$

One can read this as, "if an agent has knowledge of a situation (*e.g.* a game), and the agent is rational, then the agent executes a good action. In game theory, the important terms are suitably formalized for mathematical treatment. Knowledge is assumed to be perfect, rationality is defined as the maximization of some measure of utility, and good actions are those that bring about the outcomes with the most possible utility payoffs. The definitions make the inference an analytic truth.

Empirically, however, humans frequently deviate from the prescribed behavior. Looking at the above formula, we can ask a question: what can we infer when an agent fails to execute the prescribed action, as when pilots provide unsafe control inputs to their aircrafts? We can answer this question by examining the contrapositive of the above game-theoretic inference:

$$\neg \textit{Good\_Action} \Rightarrow \neg(\textit{Knowledge\_of\_Situation} \wedge \textit{Rationality}),$$

or equivalently,

$$\neg Good\_Action \Rightarrow \neg Knowledge\_of\_Situation \vee \neg Rationality.$$

With a bit more Boolean manipulation, we have the following:

$$\neg Good\_Action \wedge Rationality \Rightarrow \neg Knowledge\_of\_Situation.$$

This can be read, "If an agent is rational but executes a bad action, then the agent lacked knowledge of the situation." Thus, embedded in the classical game-theoretic model of agency is a logical inference from bad action to missing knowledge. This makes intuitive sense upon reflection. If someone is rational, yet they commit an irrational (read: "bad") action, then it must be the case that they didn't know some crucial information. With this insight in hand, I identify a logic in which the above inference is sound, with details about which particular pieces of information are missing from an agent's knowledge base when she executes a bad action. Again, it should not be surprising that such a logic exists, because classical game theory already posits a *logical* relationship between knowledge of particular propositions and particular actions.

I have formally captured such inferences with $\mathcal{DASL}$, where a (mostly) rational agent executes a bad action, and from this we can infer which safety-critical information they are missing. This can be done at run-time, as demonstrated herein by an encoding of the problem space for use by the Z3 theorem prover, which computes the missing information. We do not propose that a future avionics monitor actually run the Z3 tool on the aircraft, but rather use it here as a proof of concept, with implementation details left for future work.

By developing a logic that can model the information flow in these situations, we advance the project of formally reasoning about human agency. Credit for initiating this project belongs to many researchers over the years, especially philosophers in the analytic tradition concerned with analyzing the epistemology and metaphysics of agency in a rigorous fashion. I have been most influenced by the works of the Amsterdam school of modal logic, led by Professor Johan van Benthem, where the efforts center around rich combinations of modal logics in order to model human agency. In [27], they develop a modal logic for reasoning about the knowledge and decision-making of agents in games, and in [47], van Benthem explores similar themes around information flow and interaction.

The history of research in this area dates back to the 1950s and '60s in the development of temporal logic (or tense logic) by Arthur Prior [28], a graph theoretic semantics by Saul Kripke [31], and logics for belief and knowledge due to work by Rescher [68], von Wright [73], and Hintikka [52]. These logics formalize reasoning about what was true, what will be true at some point, what is always going to be true, what is believed to be true, and what is known to be true. Each of these modifiers is a truth modality, and hence they each constitute a modal logic. The Amsterdam school and others built on these methods, especially the work by Patrick Blackburn [15], which illustrates the general features of any modal logic as a tool for reasoning about systems that can be modeled as graphs from an *internal* perspective, whereas first order logic reasons about such systems from an external perspective. A first order formula might say what is true of the object x, from a global perspective, but modal logic allows us to formalize truth from x's perspective. If the first order formula is $\forall x, \exists y : \ R(x, y) \wedge P(y) \Rightarrow P(x)$, the corresponding modal formula would simply be $\Diamond_R p \Rightarrow p$. They both say the same thing: For any node x in the graph, if it can reach a node y by relation R, and y is a node where the proposition "y is P" holds, then "x is P" holds. The former explicitly quantifies over the nodes, and the latter does so implicitly through the semantics, to be described later.

While these developments occurred in what might be called the philosophical branch of modal logic research, researchers in economics explored the mathematical foundations of game theory. Aumann, in [66], showed the axioms of epistemic logic that must be assumed in order for classical game theoretic results to hold. The agents in classical games, otherwise known as *homo economicus*, are ideally rational, with perfect knowledge of their situations. We will meet these axioms and modify them for our purposes later.

The final foundational school of modal logic comes from theoretical computer science, where computer programs are modeled as state transition diagrams. As a program executes, the computer transitions from state to state, where each state is a collection of values assigned to variables, and each transition is a simple action executed. As this formalization lends itself to graph theoretic representation, it lends itself to formalization in modal logic, per Blackburn's insight. There are two main approaches to applying modal logic to the analysis of programs. The first is a static approach, where the entirety of the program's execution tree is modeled at once. Transitions from each state are captured by temporal logic. A program might be formalized in temporal logic, and the following theorem might be proven of it: *At the source of the execution graph, it will always be true that bad event B does not occur* [39]. The second approach is dynamic, where each simple transition action A or B gets a modal operator, which allows us to reason about what happens after every execution of subprocess A, or after some executions of subprocess B, etc [33].

One thing that is easy to do with dynamic modal logic but somewhat complicated in the static approach is to model actions and knowledge. An epistemic logic is modeled by a static Kripke structure, and in the dynamic case this structure changes as the agents act and learn different things. The static approach requires a grand two dimensional Kripke structure with one dimension capturing the epistemic relations at a moment, and one dimension capturing the temporal relation as actions move forward through time. For an example of this approach, see John Horty [53]. For the dynamic approach, see van Ditmarsch *et al.* [55].

Van Benthem and the Amsterdam school identified these various threads dispersed around campuses and saw how they related to each other, and how they might be fruitfully combined for various ends. As philosophers, they were mostly concerned with using the rich tools from economics and computer science to analyze human agency robustly and accurately. Modal logics offer tremendous expressive power at often a lower cost than first- or second order logic, because modal logics take an internal view of the graphs they reason about and are defined by. This often means that a powerful, useful modal logic can be defined that is also sound, complete, and *decidable*. So, by carefully defining the modal operators for, say, knowledge, preference, and action, a modal logic of game theory can be developed; not just the epistemic aspect of the agents, but the games themselves, which van Benthem calls a Theory of Play.

Applications of the Theory of Play have thus far been limited to relatively simple, artificial examples, in the same way methods in genetics research are often developed on fruit flies. In this thesis, I continue this work by extending application of the methods to richer real-world cases of humans in cockpits, which for reasons mentioned earlier make good cases for early forays into the formal modeling of human behavior. Just as genetics methods mature and eventually apply to humans, so must modal logic methods mature and apply to real humans in the world.

Dynamic Agent Safety Logic allows us to reason about which critical information is not being delivered to the human's brain. Because we can deduce which safety-critical information is missing from her knowledge base, we can automatically act to correct this failure of delivery, and therefore build systems that have a high assurance that the information will be delivered. I apply this technique to aviation safety as a formal method, but in principle it could be applied to other domains of human agency that meet certain conditions. Some examples that strike me as plausible include doctors and nurses in emergency rooms, cybersecurity analysts monitoring network traffic alerts, power plant operators, and of course motor vehicle drivers. During crises, these environments can quickly become saturated with alarms, and humans quickly suffer from information overload. If actions can be properly related to instruments such that unsafe actions can be detected, then the agent's missing knowledge can be deduced and rectified.

In order to adequately model human agency, we must depart from the familiar and comfortable logic that serves as the static base for most dynamic approaches: $\mathcal{S}5$. This departure represents an innovation, which some in the formal methods community disagree with, so we must devote some time to motivating the move. We make the case that departing from $\mathcal{S}5$ as a static base representing human knowledge and belief makes sense for philosophical and technical reasons.

In what follows, we describe the relevant background material in Chapter 2, including the foundations of game theory and the logical models of agency informing our new developments. Chapter 3 motivates departure from $\mathcal{S}5$ as a static base and present $\mathcal{DASL}$, then proves that it is sound and complete using the Coq Proof Assistant. Chapter 4 applies $\mathcal{DASL}$ to three aviation mishaps, again formalized in Coq. Chapter 5 uses the Z3 Theorem Prover's Satisfiability Modulo Theory (SMT) solving capabilities to encode the aviation mishaps and demonstrate the inference of the missing safety-critical information.

Chapter 2

Background

This chapter describes the context in which this dissertation makes advances. Section 2.1 lays out the basics of game theory, which provides a model of agency that we target and make more realistic. Section 2.2 introduces Dynamic Epistemic Logic, upon which the logic presented in this thesis is based, with axioms corresponding to the more realistic model of agency we develop. Section 2.4 describes the formal tools we use to mechanically check that $\mathcal{DASL}$ is sound and complete, that its application to aviation safety is without errors, and to encode the inference of safety-critical information. Coq is both a programming language and a tool for verifying proofs, based on the Calculus of Inductive Constructions. We explain the basics of each of these to the reader in this section, since Coq is the tool we use for proof and application checking. Z3 is a theorem prover and model checker, which interract with solely for its SMT solving abilities. We encode the actual instrument readings as a first order formula, and we encode the safety precondition of an action as formulas comprising the constraining theory. We then check whether the formula is satisfiable modulo that theory. This tells us whether the intrument readings are consistent with the safety preconditions for an action, and if not, where the conflict lies. There is some additional complexity here, but we describe this in more detail in Chapter 5.

2.1 Game Theory

This section presents the basics of game theory in order to motivate the model of agency that $\mathcal{DASL}$ seeks to formally capture. However, this dissertation does not seek to make advances in game theory itself. Future research could take the resulting logic and use it as a new foundation for game theoretic reasoning among more realistic agents, but that task is not taken on in the present work. We begin this section with a brief overview and an example game, and then present the model of agency required for the standard solution to the game.

Game theory is a mathematical model for strategic reasoning. Strategic reasoning refers to the way an agent reasons in situations where her payoffs depend on the actions of other agents in addition to her own, and in which she knows about these dependencies. For turn-based games, the mathematical structure employed is a *game tree*, where each node represents a player's turn, and each edge the transition via a player's action. The leaves of the tree represent the payoffs each player receives at the end of the game. This paper is not concerned with the games themselves, but rather with the underlying assumptions about agency that entail their solutions. We briefly illustrate these underlying assumptions with the following example.

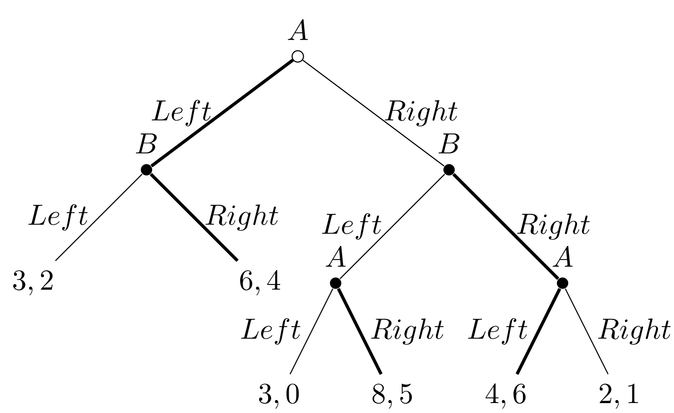


Figure 2.1: A game between players A and B

We can see in the figure below that the first player to act is Player A, at the root node. Her choices are to move *Left* or *Right*. Player B faces similar choices at the resulting nodes, and the players alternate turns until the game ends and they receive their payoffs, listed *(A,B)*. Briefly glancing through the outcomes, it looks like A should aim for the node with payoff *(8,5)*, because 8 is the highest payoff available to A. However, the official solution to the game is for A to first go *Left*, and then for B to go *Right*, resulting in a payoff of *(6,4)*, where both get less than the intuitively appealing outcome! Why is this so?

Game theory makes strong assumptions about agent knowledge and rationality. The solution to this game is reached through an algorithm called *backward induction* [27]. The players reason by starting at each end node and looking to the immediate parent node, and asking what the deciding player will do at that node, assuming she will choose the path with the highest payoff. So, at the bottom right of the figure, player A is to act, and she can go *Left* for a payoff of 4, or *Right* for a payoff of 2. So, she will go *Left*, illustrated by the bold line. The end nodes not selected are subsequently eliminated. This process is repeated at each end node. Then it is recursively applied up the tree. So along the right branch, player B decides between *Left* for a payoff of 5, or *Right* for a payoff of 6, because B knows that A is rational, and he knows how she will act at each end node. A, at the root, then must choose between *Left* for a payoff of 6, or *Right* for a payoff of 4, because she knows that B is rational, and knows that B knows that she is rational. The explanation begins to illustrate the assumptions game theory makes about each player's knowledge. In fact, this only scratches the surface.

Game theory, and classical economics in general, makes the following assumptions about agent knowledge, formalized in epistemic logic [66].

Agency Model in Classical Game Theory.

(1) $\mathbf{K_i}\,(\varphi \Rightarrow \psi) \Rightarrow (\mathbf{K_i}\,\varphi \Rightarrow \mathbf{K_i}\,\psi)$

(2) $\mathbf{K_i}\,\varphi \Rightarrow \varphi$

(3) $\mathbf{K_i}\,\varphi \Rightarrow \mathbf{K_i}\,\mathbf{K_i}\,\varphi$

(4) $\neg\mathbf{K_i}\,\varphi \Rightarrow \mathbf{K_i}\,\neg\mathbf{K_i}\,\varphi$

(5) $\mathbf{C}_G((1) \wedge (2) \wedge (3) \wedge (4) \wedge (5))$.

This forms an idealized model of the knowledge component of classical game theory's agents. $\mathbf{K_i}$ is a modal operator for knowledge, and $\mathbf{K_i}\,\varphi$ reads, "agent i knows that φ." $\mathbf{C}_G$ is a modal operator for common knowledge, the fixpoint for "everyone in group G knows that everyone knows that..." The agents are logically omniscient due to (1), knowledge implies truth with (2), agents have *positive introspection* with (3), and *negative introspection* with (4). Assumptions (1), (3), (4), and (5) are somewhat dubious. The model also fails to formally represent other aspects of agency, like action and evaluation of outcomes. The model we propose makes weaker, more realistic assumptions about knowledge, includes a modal operator for belief, and formally represents action and the evaluation of actions as either safe or unsafe.

Recent work at the intersection of game theory and logic focuses on the information flow that occurs during games. Van Ditmarsch identifies a class of games called *knowledge games*, in which players have diverging information [49]. This slightly relaxes the assumption of classical game theory that players have common knowledge about each other's perfect information. Similarly, it invites logicians to study the information conveyed by the fact that an action is executed. For example, if the action is that agent 1 asks agent 2 the question, "p?", the information conveyed is that 1 does not know whether p, believes that 2 knows whether p, and after the action occurs, this information becomes publicly known. The logic modeling games of this kind is of particular interest to us, as we are concerned with identifying the knowledge and belief state of human pilots based on their actions.

The proceeding sections introduce the various logical systems that form a foundation for the work of this thesis, starting with modal logic in its traditional philosophical interpretation, and expanding to epistemic and doxastic logic. Then, we introduce dynamic logic, and its expansion into Dynamic Epistemic Logic and Public Announcement Logic.

2.2 Modal Logic

Aristotle noted a distinction between contingent truth and necessary truth, and some Medieval philosophers continued this line of inquiry. Necessary truths could not have been otherwise. The definitions of natural numbers and the addition function guarantee that in all possible worlds, *2 + 2 = 4*. On a plane, the truths of Euclidean geometry are necessarily true. "There is life on Saturn's moon Enceladus" is a possible truth. "Enceladus has water on it" is a contingent truth. What about "Water is H20"? Modal logic was formalized in the early 1900's by C. I. Lewis and has recently had a resurgence in multidisciplinary interest [48]. At its core, modal logic allows us to reason about necessary and possible truths through the use of modal operators. What follows is a brief illustration of modal logic's concepts and formalisms.

When philosophers talk about necessity they usually mean *metaphysical* necessity. In addition to formalizing this notion for systematic reasoning, modal logic is used for clarifying what exactly this means [74]. Simple and intuitive examples are those from mathematics and notions of identity. Suppose p is the arithmetic expression "2 + 2 = 4". Obviously, p is true in the actual world. We can make the stronger claim that p is necessarily true, but what does this mean? As informal shorthands, initial attempts to define necessary truths might appeal, as I did above, to the claim that they could not possibly have been false. But then what do we mean by "possible"? The formal semantics for dealing with these questions comes from Arthur Prior and Saul Kripke, and we introduce that machinery in the next section [28, 31]. For now, we say a statement is necessarily true if and only if it is true in all worlds we consider possible. The modal operator for necessity makes the formal statement: $\Box p$. Consider the following inference, where $\Rightarrow$ means "implies": $\Box p \Rightarrow p$. This reads, "Necessarily p implies p," or equivalently, "If p is necessarily true, then p is true." If p is necessarily true, is p true? Intuitively, the answer is 'yes', and indeed a modal logic of metaphysical necessity includes this axiom for all formulas φ: $\Box\varphi \Rightarrow \varphi$.

What other inferences can we make, based on our intuitive notion of necessity and possibility? What about "if p is true, then p is possibly true"? To formalize this, we need the modal operator for possibility: $\Diamond$. So, the modal formula would be $p \Rightarrow \Diamond p$. This seems true as well, and indeed it is a theorem for metaphysical modal logic: $\varphi \Rightarrow \Diamond\varphi$.

It is obvious then that $\Box\varphi \Rightarrow \Diamond\varphi$ is a theorem. This states that if something is necessarily true then it is possibly true. What about the other direction: $\Diamond\varphi \Rightarrow \Box\varphi$? It turns out this is not a theorem under the typical notions of necessity and possibility. But this raises a question about how we would present a counterexample that disproves it. To do this, we need a semantics for the logic. The semantics we use are called *possible world semantics*, and they are usually attributed to Saul Kripke [31]. One would be hard-pressed to find a species of modal logic, whether in economics, computer science, or philosophy, that does not use possible world semantics in some form or another. Sometimes they are referred to as *Kripke semantics*.

In possible world semantics, a graph structure is created with worlds as nodes and accessibility relations among worlds as the edges in the graph [15]. Propositional formulas are true or false at each world. These graph structures are typically called Kripke structures. We can define the following Kripke structure, $\mathcal{M} = \{W, R, V\}$, where W is a finite set of worlds, $\{w, v\}$, R is a binary accessibility relation defined on those worlds $\{(w, v), (v, w)\}$, meaning w has access to v and v has access to w, and V is a *valuation* function, which maps propositions to sets of worlds at which they are true. For example, if p is true at w, then $w \in V(p)$. In our model we only care about the proposition p, which now stands for some contingent proposition, like "all swans are white". Formally, we say $w \notin V(p)$ if not all swans are white in world w, denoted by $\neg p$, while $w \in V(p)$ if they are, denoted by p. The following figure illustrates $\mathcal{M}$:

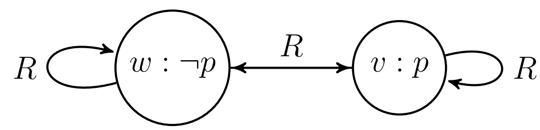


Figure 2.2: $\mathcal{M}$: A simple counterexample using possible world semantics.

According to possible world semantics, $\Box\varphi$ is true at a world w, written $w \models \Box\varphi$, if and only if for all worlds v such that $R(w, v)$ (v is related to w by the R relation), $v \models \varphi$. This says a formula is necessarily true at a world if and only if it is true at all worlds accessible by that world according to the underlying R relation. Similarly, $w \models \Diamond\varphi$ if and only if there is some world v such that $R(w, v)$ and $v \models \varphi$. In $\mathcal{M}$, w is R-accessible to itself, so there is a world accessible to w where p is false, and thus $w \models \neg\Box p$. However, since v is R-accessible to w, and $v \models p$, it is true that $w \models \Diamond p$. Thus, we have $w \models \Diamond p$ and $w \models \neg\Box p$, a negation of $\Diamond p \Rightarrow \Box p$, so it cannot be the case that $\Diamond\varphi \Rightarrow \Box\varphi$ is a theorem, for arbitrary formula φ.

There are many systems of modal logic. The way any is distinguishedfrom another is based entirely on the definition of R. For the popular $\mathcal{S}5$, which we will examine later, R is a Euclidean and eflexive binary relation on worlds. Thus, that is how "possibility" is formally defined, and by extension, "necessarily". That is also what determines the axioms and theorems comprising the logic. We formally define these notions in the next section, along with the syntax and semantics of propositional modal logic.

2.2.1 Modal Logic Syntax and Semantics

This section formally defines the syntax, what the logic looks like, and the semantics, what the truth conditions are for the logic.

Recall that Boolean logic is a simple logic for reasoning about basic propositions using the logical connectives 'and', 'or', 'not', and 'if...then'. It forms the foundation of most logics and has applications ranging from philosophy to circuit design. Propositions are represented as constants p, q and well-formed formulas of the language are constants and any proper combination of constants using the above logical connectives, represented symbolically as,

$$\varphi \stackrel{def}{=} p \mid \varphi \wedge \varphi \mid \varphi \vee \varphi \mid \neg\varphi \mid \varphi \Rightarrow \varphi.$$

As illustrated in the previous section, modal logic adds to propositional logic with modal operators for necessary and possible truths. The syntax for propositional logic is extended in the following way to make modal logic:

The semantics for Boolean logic are simply truth tables for each connective, which I will not reproduce here. However, the operators in modal logic are not truth functional, and require more complex semantics, which we earlier mentioned are called possible world semantics. They are as follows.

Let $\mathcal{M} = \{W, R, V\}$ be a model such that W is a set of possible worlds, R is a binary relation on those worlds, and V is a valuation function mapping atomic propositions to the sets worlds satisfying them,

$$w \models p \textbf{ iff } w \in V(p)$$

$$w \models \neg\varphi \textbf{ iff } w \not\models \varphi$$

$$w \models \varphi \wedge \psi \textbf{ iff } w \models \varphi \textit{ and } w \models \psi$$

$$w \models \Box\varphi \textbf{ iff } \forall v,\ R(w,v) \textit{ implies } v \models \varphi.$$

$$w \models \Diamond\varphi \textbf{ iff } \exists v,\ R(w,v) \textit{ and } v \models \varphi.$$

The character of a modal logic is determined by the binary relation on worlds underlying the modal operators. We mentioned $\mathcal{S}5$ in the previous section as having a R relation in which every world is accessible to itself by the binary relation. It is also one in which the R relation is Euclidean. These conditions are called a *frame* conditions, and the models that satisfy these conditions belong to said frame. All reflexive frames have the following frame conditions:

$$\forall x,\ R(x,x) \tag{2.1}$$

Likewise, all reflexive frames have the following axiom:

$$\Box\varphi \Rightarrow \varphi \tag{2.2}$$

It is not just a stipulation that all reflexive frames must have that axiom: They have that axiom *because* they have that frame condition. This is due to correspondence theory, which we explain later on.

Relaxing a frame condition changes the R relation, and in doing so changes the logic. If we remove the reflexivity condition, the above axiom is no longer an axiom for the logic defined on $\mathcal{M}$. This means that we can specify the axioms we want by specifying the frame condition on the accessibility relation. Each frame condition corresponds to a modal logic axiom. In addition to reflexivity, the other common frame conditions are as follows, with their corresponding modal logic axiom:

- Transitivity

$$\forall x, y, z\ R(x,y) \wedge R(y,z) \Rightarrow R(x,z) \tag{2.3}$$

$$\Box\varphi \Rightarrow \Box\Box\varphi \tag{2.4}$$

- Symmetry

$$\forall x, y\ R(x,y) \Rightarrow R(y,x) \tag{2.5}$$

$$\varphi \Rightarrow \Box\Diamond\varphi \tag{2.6}$$

- Euclidean

$$\forall x, y, z\ R(x,y) \wedge R(y,z) \Rightarrow R(x,z) \tag{2.7}$$

$$\Diamond\varphi \Rightarrow \Box\Diamond\varphi \tag{2.8}$$

- Seriality

$$\forall x \exists y, R(x,y) \tag{2.9}$$

$$\Box\varphi \Rightarrow \Diamond\varphi \tag{2.10}$$

These conditions are not exhaustive, but they represent some of the commonly combined conditions used to define axioms of different modal logics. By combining frame conditions, a modal operator is defined with the properties desired. For example, if we wish to define a modal operator for reasoning about the knowledge of ideally rational agents, as is done in Fagin *et. al.*[50], we impose the frame conditions of reflexivity and Euclidicity (Euclideanness), or one that is transitive and reflexive, as in Hintikka [52]. However, if we wish to develop a logic for belief, as the previously cited works also do, we must impose transitivity, Euclidicity, and seriality. The reasons for this are explored in the following sections.

2.2.2 Epistemic Logic

Epistemic logic began with philosophical concerns about knowledge [50, 52, 68]. Presently, it is likely to be studied in computer science as the logic for reasoning about ideally rational agents in well-structured environments. The agents are ideally rational in the sense that there is no bound on how much they can be said to know, nor on what propositions they are aware of at any one time. Thus, they have no problem conceiving of every possible sequence of moves a game may consist of, evaluating all possible outcomes, and deducing the optimal sequence of moves in order to maximize their own utilities. They are even stronger than contemporary computers in this way, which are bounded by time and space, and so are unable to actually compute the ideal strategy for an otherwise solvable game like Go. An ideally rational agent can solve Go and compute the game tree all the way to its end. These are the agents of game theory, which always select the optimal move when they have perfect information about the game and the other players.

The axioms that specify this level of knowledge are those introduced in Section **??**, and recounted here without reference to common knowledge, which complicates things too much for our purposes:

Agency Model in Classical Game Theory.

(1) $\mathbf{K_i}\,(\varphi \Rightarrow \psi) \Rightarrow (\mathbf{K_i}\,\varphi \Rightarrow \mathbf{K_i}\,\psi)$

(2) $\mathbf{K_i}\,\varphi \Rightarrow \varphi$

(3) $\mathbf{K_i}\,\varphi \Rightarrow \mathbf{K_i}\,\mathbf{K_i}\,\varphi$

(4) $\neg\mathbf{K_i}\,\varphi \Rightarrow \mathbf{K_i}\,\neg\mathbf{K_i}\,\varphi$.

These axioms make the knowledge operator an *S5* modal operator. Strictly speaking, including (3) is unnecessary because it follows as a theorem from (2) and (4), but we include it so that the reader is not surprised when we refer to it.

The first axiom holds for all *normal* modal logics, and under the epistemic interpretation, it states that if an agent knows that φ implies ψ, and she knows φ, then she knows ψ. This is intuitive enough on the first pass, but can lead to a problem of global skepticism. If i knows that she is an embodied agent in the external world only if she knows that she is not a brain in a vat perceiving a simulated reality, then she knows she is an embodied agent only if she knows she is not a brain in a vat. It seems clear that she cannot know that she is not a brain in a vat, because her sensory experience is compatible with either the embodied scenario of the brain in a vat scenario. Therefore, she does not know she is an embodied agent in the external world. Thus, many philosophers question whether knowledge for human-like agents is closed under logical implication like this.

The second axiom states that a known proposition must be true, and finds ample support from philosophers devoted to studying the nature of knowledge. It can be considered the property of knowledge that most distinguishes it from belief, because beliefs can be false.

The third axiom, *positive introspection*, states that an agent knows something only if she knows *that* she knows it. This imposes a high standard on knowledge, under the assumption that an agent can identify the conditions that guarantee the truth of a known proposition, and in being able to do so, are justified to a sufficient degree. However, this property is largely rejected by philosophers, as this requirement is thought to be largely unattainable.

Finally, *negative introspection* states that an agent does not know something only if she knows that she does not know it. This has an undesirable effect of creating knowledge out of ignorance, for if i does not know that she does not know φ, this is sufficient for inferring that she knows φ, according to negative introspection. This is fine for ideally rational agents, and indeed it is a noble aspirational goal to know what one is ignorant about, but for human-like agents it is entirely unrealistic as an axiom.

These axioms are firmly rooted in the literature of formal epistemology and epistemic logic, and relaxing them in order to model more realistic agents requires a word or two. This section introduces and defends our relaxation of the classical axiomatization.

2.2.3 Why S5 Is Appealing

Before presenting our relaxed axiom schema for knowledge, we acknowledge the appealing properties of the S5 knowledge operator. Logicians who adopt the S5 knowledge operator have good reasons for doing so, both technical and intuitive. The technical appeal lies in the fact that the S5 operator is well-behaved from a logical point of view. Epistemic logics with S5 operators are decidable because S5 operators have semantics that satisfy the finite model property, meaning that every invalid proposition has a finite counter-model. The S5 knowledge operator allows a formula prefixed by any arbitrary combination of $\mathbf{K_i}$ and $\langle \mathbf{K}_i \rangle$ symbols to be reduced to a formula prefixed by at most two. This keeps the models small.

On the intuitive side, the relations that define the semantics of an S5 knowledge operator are equivalence relations. They are defined as reflexive and Euclidean, which implies that they are also transitive and symmetric. Equivalence relations capture a notion of indistinguishability because each world in the equivalence class is indistinguishable from each other. All the modal formulas true at one of the worlds in an equivalence relation are true at each of the other worlds in the relation, so from a modal perspective, they are indistinguishable. When the modality is knowledge, this means that relative to the agent's knowledge, the worlds are indistinguishable.

These properties of $\mathcal{S5}$ epistemic logic are strong points in its favor. We suppose that any formal system seeking to model human-like knowledge must capture the intuitive idea that an agent does not know *whether* a proposition φ is true or false just in case she cannot distinguish the world she is in from the possible worlds in which either might be the case. Distinguishability seems to imply that there is some evidence possessed by the agent that allows her to rule out possibilities, and therefore gain knowledge.

The technical benefit of the finite model property makes $\mathcal{S5}$ a friendly logic to work with, but it is not a reason to believe that the theory of knowledge described by $\mathcal{S5}$ is true of human agents. Therefore, we do not see the need to impose this condition as a requirement for realistic epistemic logics. However, it would be desireable to achieve the finite model property while maintaining realism.

Next we turn to a brief summary of doxastic logic, which shall play a role in $\mathcal{DASL}$'s static foundation.

2.2.4 Doxastic Logic

This section presents doxastic logic, the logic for reasoning about belief. Hintikka's formalization in [52] of the belief operator is standard, and we adopt it here. The operator $\mathbf{B_i}$ for 'agent i believes ℓ' is defined by a binary relation on worlds that is serial and Euclidean. It is a normal modal operator, and the property of transitivity follows from seriality and Euclidicity. So, the model of an agent's belief is defined by the following axioms,

Hintikka Belief Model.

(1) $\mathbf{B_i}\,(\varphi \Rightarrow \psi) \Rightarrow (\mathbf{B_i}\,\varphi \Rightarrow \mathbf{B_i}\,\psi)$

(2) $\mathbf{B_i}\,\varphi \Rightarrow \langle \mathbf{B}_i \rangle\,\varphi$

(3) $\mathbf{B_i}\,\varphi \Rightarrow \mathbf{B_i}\,\mathbf{B_i}\,\varphi$

(4) $\neg\mathbf{B_i}\,\varphi \Rightarrow \mathbf{B_i}\,\neg\mathbf{B_i}\,\varphi$.

Clearly the primary difference between this model of belief and the model of knowledge from classical game theory is axiom (2). Rather than guaranteeing truth, beliefs guarantee consistency. One could correctly argue that this is too strong of an assumption for humans, but we leave relaxations of the belief model to future work, and are content to relax the idealizations about knowledge in this dissertation. Similarly, reasonable cases could be made against axioms (3) and (4).

The logic associated with these axioms is called $\mathcal{KD}45$. On the doxastic interpretation, it represents a slight idealization of how humans actually adopt beliefs. Clearly, humans can have contradictory beliefs, and they may believe something without believing that they believe it. Cases of cognitive dissonance like that are idealized away, not only because modeling such a psychology is complicated, but also because we seek a logic that maintains some level of normativity. If someone believes two contradictory propositions, she should be able to give one up if this contradiction is pointed out to her, and furthermore she is rational to do so. This reflects Hintikka's goal of developing logics of knowledge and belief that allow us to consistently identify when attributions of knowledge and belief are in some sense criticizable. Maintaining a notion of normativity in a logic is important to us, and underlies some the idealizations we accept in $\mathcal{DASL}$'s static base logic.

Knowledge involves a condition that is external to the agent's direct perception: the truth or falsity of the proposition. Belief, on the other hand, is entirely internal to the agent's perception. She can introspect on her evidence and deliberate on a proposition, and this is sufficient to see that she believes or disbelieves it, and this attitude is immediately available to her. For this reason, the positive and negative introspection axioms are more appropriate for belief than they are for knowledge.

2.2.5 Knowledge and Belief Combined

In addition to developing a stand-alone axiomatization of the belief operator, Hintikka combined the knowledge and belief operators into a single multi-modal system. Other philosophers and logicians have constructed similar systems. Hintikka considers the following combined axioms, which he calls conditions, as options:

Hintikka Combinations Considered.

(1) $\mathbf{K_i}\,\varphi \Rightarrow \mathbf{B_i}\,\mathbf{K_i}\,\varphi$

(2) $\mathbf{K_i}\,\varphi \Rightarrow \mathbf{B_i}\,\varphi$

(3) $\mathbf{B_i}\,\varphi \Rightarrow \mathbf{K_i}\,\mathbf{B_i}\,\varphi$

(4) $\mathbf{B_i}\,\varphi \Rightarrow \mathbf{K_i}\,\varphi$.

He subsequently rejects (4) immediately, and through deliberation (3), while he accepts (1) and (2). Thus, Hintikka's combined logic of knowledge and belief is,

$\mathbf{K_i}\,(\varphi \Rightarrow \psi) \Rightarrow (\mathbf{K_i}\,\varphi \Rightarrow \mathbf{K_i}\,\psi)$	Distribution of $\mathbf{K_i}$
$\mathbf{K_i}\,\varphi \Rightarrow \varphi$	Truth Axiom
$\mathbf{K_i}\,\varphi \Rightarrow \mathbf{K_i}\,\mathbf{K_i}\,\varphi$	Positive Knowledge Introspection
$\mathbf{B_i}\,(\varphi \Rightarrow \psi) \Rightarrow (\mathbf{B_i}\,\varphi \Rightarrow \mathbf{B_i}\,\psi)$	Distribution of $\mathbf{B_i}$
$\mathbf{B_i}\,\varphi \Rightarrow \langle \mathbf{B}_i \rangle\,\varphi$	Belief Consistency Axiom
$\neg\mathbf{B_i}\,\varphi \Rightarrow \mathbf{B_i}\,\neg\mathbf{B_i}\,\varphi$	Negative Belief Introspection
$\mathbf{K_i}\,\varphi \Rightarrow \mathbf{B_i}\,\varphi$	Knowledge implies Belief
$\mathbf{K_i}\,\varphi \Rightarrow \mathbf{B_i}\,\mathbf{K_i}\,\varphi$	Undeniable Knowledge
From $\vdash \varphi$ and $\vdash \varphi \Rightarrow \psi$, infer $\vdash \psi$	Modus Ponens
From $\vdash \varphi$, infer $\vdash \mathbf{K_i}\,\varphi$	Necessitation of $\mathbf{K_i}$

Table 2.1: Hintikka's Combined Logic of Knowledge and Belief

We omit his defense of positive knowledge introspection, but point out that his defense of undeniable knowledge depends on his interpretation of the operators. Hintikka is concerned in his project with identifying statements about knowledge and belief that utterers cannot defensibly deny. Furthermore, his interpretation of knowledge and belief are not those propositions actively held in one's head, but those to which one could be led through a series of applications of *Modus Ponens* from what one already knows or believes. We can think of knowledge, for Hintikka, as those propositions implied by what one already knows, and beliefs as those propositions that one is logically committed to by one's beliefs. This avoids the problem of logical omnicience for both modal operators. And it sets up an argument like the following,

1. φ follows logically from what i knows

2. If (1) is pointed out to i, it is indefensible for i to deny that she believes that she knows φ.

3. Therefore, $\mathbf{K_i}\,\varphi \Rightarrow \mathbf{B_i}\,\mathbf{K_i}\,\varphi$

For Hintikka, this suffices to accept the formula as valid, as well as Knowledge implies Belief, for which a similar argument can be presented.

Conversely, Hintikka rejects (3) because he does not find it plausible that $\mathbf{B_i}\,\varphi \Rightarrow \mathbf{K_i}\,\langle\mathbf{K}_i\rangle\,\varphi$, which (3) and the Truth Axiom jointly imply. He furthermore rejects arguments from introspection, which might ground a defense in considered formula (3), as fallacious.

Another potential combined logic of knowledge and belief is due to Kraus and Lehmann in [30].

$\mathbf{K_i}\,(\varphi \Rightarrow \psi) \Rightarrow (\mathbf{K_i}\,\varphi \Rightarrow \mathbf{K_i}\,\psi)$	Distribution of $\mathbf{K_i}$
$\mathbf{K_i}\,\varphi \Rightarrow \varphi$	Truth Axiom
$\mathbf{K_i}\,\varphi \Rightarrow \mathbf{K_i}\,\mathbf{K_i}\,\varphi$	Positive Knowledge Introspection
$\neg\mathbf{K_i}\,\varphi \Rightarrow \mathbf{K_i}\,\neg\mathbf{K_i}\,\varphi$	Negative Knowledge Introspection
$\mathbf{B_i}\,(\varphi \Rightarrow \psi) \Rightarrow (\mathbf{B_i}\,\varphi \Rightarrow \mathbf{B_i}\,\psi)$	Distribution of $\mathbf{B_i}$
$\mathbf{B_i}\,\varphi \Rightarrow \langle\mathbf{B}_i\rangle\,\varphi$	Belief Consistency Axiom
$\mathbf{K_i}\,\varphi \Rightarrow \mathbf{B_i}\,\varphi$	Knowledge implies Belief
$\mathbf{B_i}\,\varphi \Rightarrow \mathbf{K_i}\,\mathbf{B_i}\,\varphi$	Known Belief
From $\vdash \varphi$ and $\vdash \varphi \Rightarrow \psi$, infer $\vdash \psi$	Modus Ponens
From $\vdash \varphi$, infer $\vdash \mathbf{K_i}\,\varphi$	Necessitation of $\mathbf{K_i}$

Table 2.2: Kraus-Lehmann Combined Logic of Knowledge and Belief

Knowledge is $\mathcal{S}5$ and belief is based on a serial relation. However, from this system the remaining *KD45* formulas are derivable. Additionally, they have the interesting theorem $\mathbf{K_i}\,\varphi \equiv \mathbf{B_i}\,\mathbf{K_i}\,\varphi$.

Kraus and Lehmann identify the formula $\mathbf{B_i}\,\varphi \Rightarrow \mathbf{B_i}\,\mathbf{K_i}\,\varphi$ as interesting but exclude it, as it would cause a collapse of knowledge and belief into equivalence: $\mathbf{K_i}\,\varphi \equiv \mathbf{B_i}\,\varphi$. Obviously, a logic combining knowledge and belief should treat them as distinct operators, and we treat the avoidance of this collapse as a desirable feature of our own static base for $\mathcal{DASL}$. We also include the interesting formula above as an axiom, which imposes a condition on agents' beliefs that we consider a desireable idealization. To say i believes φ only if she believes that she knows φ is to say she believes only those things that she has very good evidence for. This helps prevent agents from believing things without evidence, which we want to exclude from our model. The challenge with this axiom, as Kraus and Lehmann note, is to avoid the knowledge-belief collapse. We give this further treatment in a later chapter.

This concludes our discussion of static epistemic and doxastic logics, including their combination. We pick this thread back up later when we present the static base of $\mathcal{DASL}$. The next section presents the relevant background of dynamic extensions to epistemic logic.

2.3 Dynamic Epistemic Logic

This section presents the so-called *dynamic turn* in modal logic, which incorporates elements of propositional dynamic logic into epistemic logic in order to model knowledge and action. See van Benthem [47, 48, 27], and van Ditmarsch, van der Hoek, and Kooi [55] for detailed examinations. The related Public Announcement Logic (PAL) uses similar techniques to model information flow among agents. See Baltag, Moss, and Solecki in [44] for more on this.

Dynamic Epistemic Logic (DEL) formalizes situations in which agents' epistemic states change over time, due to announcements or other informational events[55]. For example, if Alice truthfully and trustworthily communicates to Bob that φ, then after this informational even it is true that Bob knows φ. This situation cannot be modeled by the epistemic logic introduced in the previous section. To model it, we introduce the following formal machinery.

To capture informational events, we introduce the idea of relativizing a Kripke structure. In the previous example, if we model the Alice and Bob situation prior to Alice's communication, we can have a world w from which Bob considers φ- as well as $\neg\varphi$-worlds possible. However, after the informational event, Bob knows φ, so the model is *relativized* to a submodel in which only φ-worlds are accessible by Bob's epistemic possibility relation. Thus, after the informational event, the model transitions to a submodel with fewer edge relations.

The logic for reasoning about information flow in knowledge games is called Dynamic Epistemic Logic (DEL). As its name suggests, it combines elements of epistemic logic and dynamic logic. Epistemic logic is the static logic for reasoning about knowledge, and dynamic logic is used to reason about actions. In dynamic logic semantics, nodes are states of the system or the world, and relations on nodes are transitions via programs or actions from node to node. If we think of each node in dynamic logic as being a model of epistemic logic, then actions become relations on models, representing transitions from one multi-agent epistemic model to another. For example, if we have a static epistemic model $M1$ representing the knowledge states of agents Alice and Bob at a moment, then the action "φ" is a relation between $M1$ and $M2$, a new static epistemic model of Alice's and Bob's knowledge after the question is asked. All of this is captured by DEL.

$$M1 \xrightarrow{\text{“}\varphi\text{”}} M2$$

The above figure illustrates the relationship between static epistemic models and dynamic logic models. As a purely dynamic model, the figure shows the action "p?" transitioning between nodes $M1$ and $M2$. If we zoom in on the nodes, we see their structure as epistemic models, with their own nodes and edges, representing possible worlds and epistemic relations.

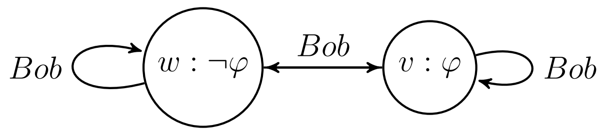


Figure 2.3: $M1$: The model before Alice announces "φ".

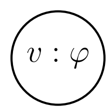


Figure 2.4: $M2$: The model after Alice announces "φ".

We are concerned with an additional element: the *safety* status of an action, and an agent's knowledge and belief about that. To capture this, we extend DEL and call the new logic Dynamic Agent Safety Logic ($\mathcal{DASL}$), which we introduce in the next chapter. The next section lays out the state of formal methods involving human-machine systems.

2.4 Formal Methods Tools

This section describes the Coq Proof Assistant [11] and the Z3 Theorem Prover [60] that this dissertation makes use of. We use Coq to mechanically check our metatheory proofs of soundness and completeness, and our object theory proofs about safety-critical information flow. We use the Z3 Theorem Prover's capacity as a SMT solver to automatically infer missing safety-critical information based on an unsafe input action. This serves as a proof of concept for potential safety-critical information monitors that make use of $\mathcal{DASL}$ as their foundation.

The Coq Proof Assistant is a combination of a dependently-typed functional programming language and a language of tactics for partially automating proof verification. Coq supports compilation to Haskell and OCaml, which is convenient because one can write a program and verify its correctness in Coq, then compile the code to a nicer implementation language. This dissertation does not take advantage of that feature. Instead, we embed $\mathcal{DASL}$ in Coq and use it as a proof checker that allows us to write recursive functions and proof-automating tactics. This represents a second use case of formal verification tools that one does not often see in the philosophy, mathematics, or logic departments of academia. Coq becomes a tool for us to use to increase the rigor of our research. Many who have ventured into the realm of mechanical theorem proving have been surprised initially by how complicated proofs 'by routine induction' can become once all of the details must be spelled out.

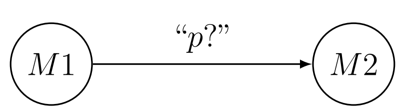
We use Z3 in the same way. As a tool, it is a collection of symbolic reasoning engines, including simplex, rewriting, DPLL, superposition, Euclidean solver, and others.[1] We use it to validate that one can encode an instrument reading configuration and a safety precondition for an input action, and automatically detect whether the safety precondition is satisfied by the instrument readings. This validates an important requirement for implementing tools based on $\mathcal{DASL}$. Thus, rather than using Z3 to verify correctness properties of a prototype, we use it as an inference engine that demonstrates real time inference of safety-critical information from unsafe input actions.

[1] Presentation by Dr. Leonardo DeMoura, "From Z3 to Lean: Efficient Verification."

Chapter 3

Dynamic Agent Safety Logic

The logic for reasoning about information flow in knowledge games is called Dynamic Epistemic Logic (DEL). As its name suggests, it combines elements of epistemic logic and dynamic logic. Epistemic logic is the static logic for reasoning about knowledge, and dynamic logic is used to reason about actions. In dynamic logic semantics, nodes are states of the system (or of the world), and relations on nodes are transitions via programs or actions from node to node. If we think of each node in dynamic logic as being a model of epistemic logic, then actions become relations on models, representing transitions from one multi-agent epistemic model to another. For example, if we have a static epistemic model $M1$ representing the knowledge states of agents 1 and 2 at a moment, then the action "p?" is a relation between $M1$ and $M2$, a new static epistemic model of 1's and 2's knowledge after the question is asked. All of this is captured by DEL.



The above figure illustrates the relationship between static epistemic models and dynamic logic models. As a purely dynamic model, the figure shows the action "p?"

transitioning between nodes $M1$ and $M2$. If we were to zoom in on the nodes, we would see their structure as epistemic models, with their own nodes and edges, representing possible worlds and epistemic relations.

We are concerned with an additional element: the *safety* status of an action, and an agent's knowledge and belief about that. To capture this, we extend DEL and call the new logic Dynamic Agent Safety Logic ($\mathcal{DASL}$). The remainder of this section presents $\mathcal{DASL}$'s syntax, semantics, and proves its soundness.

3.1 Syntax and Semantics

3.1.1 Syntax

$\mathcal{DASL}$ has the following syntax.

$$\varphi \ ::= \ p \mid \neg\varphi \mid \varphi \wedge \varphi \mid \mathbf{K_i}\,\varphi \mid \mathbf{B_i}\,\varphi \mid [\mathbf{A}, a]_i\varphi \mid [\mathbf{A}, a]_i^{\mathcal{S}}\varphi,$$

where $p \in AtProp$ is an atomic proposition letter drawn from a finite set of such letters , $\mathbf{i}$ refers to $i \in Agents$, a is the name of an action, called an action token, belong to a set of such tokens, *Actions*, and $\mathbf{A}$ refers to an action model. The knowledge operator $\mathbf{K_i}$ indicates that "agent i knows that ..." Similarly, the operator for belief, $\mathbf{B_i}$ can be read, "agent i believes that..." The notion of action tokens and structures will be defined in the semantics. The operators $[\mathbf{A}, a]_i$ and $[\mathbf{A}, a]_i^{\mathcal{S}}$ are the dynamic operators for action a from action structure A occurring in the former case, and happening safely in the latter case. One can read the action operators as "after a from A occurs (*safely*), φ holds in all resulting worlds.' We define the dual modal operators $\langle\mathbf{K}_i\rangle$, $\langle\mathbf{B}_i\rangle$, $\langle\mathbf{A}, a\rangle_i$, and $\langle\mathbf{A}, a\rangle_i^{\mathcal{S}}$ in the usual way.

The semantics of $\mathcal{DASL}$ involve two structures that are defined simultaneously, one for epistemic models, and one for action structures capturing the transition relation among epistemic models. Additionally, we define numerous helper functions that straddle the division between metalanguage and object language.

3.1.2 Metalanguage

The metalanguage defining $\mathcal{DASL}$ consists of traditional Kripke models, and action models, with functions for action pre- and post-conditions and the product update of Kripke and action models [55].

Kripke (Relational) Structure

A Kripke structure, called model M, is a tuple $\langle W, R_k^i, R_b^i, w, V\rangle$. It is a set of worlds W, epistemic and doxastic relations on worlds for agents, a world denoting the actual world, and a valuation function V mapping atomic propositions to the set of worlds satisfying them.

Action Model

An action structure $\mathbf{A}$ is a tuple $\langle \mathit{Actions}, \chi_k^i, \chi_b^i, \mathit{pre}, \mathit{post}\rangle$. It is a set of action tokens, sets of epistemic and doxastic relations on action tokens for agents, action pre- and post-condition functions, and safe action pre-condition functions.

An action model captures the agents' subjective perspectives of an event's occurrence. For example, consider a situation in which Alice flips a coin and Bob calls whether it is *heads* or *tails*. The action model imposes no precondition for occurring, represented by the value $\top$, and each action token's post-condition maps $\top$ to *heads* or *tails* respectively. Before Alice peeks to see what it is, the Action Model $\mathbf{A}$ looks like (omitting the safety pre-conditions and doxastic relations, for simplicity):

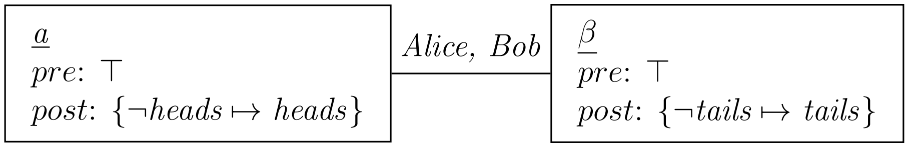


Figure 3.1: Action Model for “Alice flips a coin”

We use rectangles for action tokens to distinguish them from possible worlds in Kripke Structures. Each action token is marked by a token name, a, β, etc. The labels *Alice, Bob* indicate the agents' epistemic (——) relations. When the relations lack direction, it indicates bi-directionality, and every action token has a looping epistemic relation to itself that we omit in order to keep the pictures simpler. In Figure 3.1, neither Alice nor Bob sees whether the coin lands *heads* or *tails*, but the post-condition of the action is that the proposition expressing the state of the coin becomes *heads* or *tails*. Note that Action Models themselves do not have propositions true and false at them, but rather have pre-conditions and post-conditions for execution of their tokens, and modal relations over the token pairs. The action model for “Alice peeks” is:

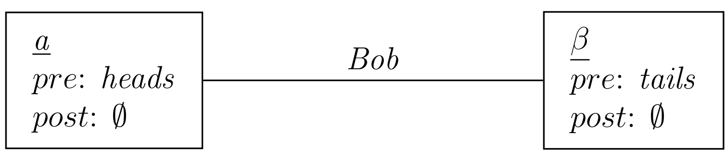


Figure 3.2: Action Model for “Alice peeks”

In Figure 3.2, the epistemic and doxastic relations for Alice have vanished. In the event that Alice peeks and sees *heads*, she sees that the coin landed *heads*, and similarly for *tails*. Bob, however, still considers it possible that Alice peeked and saw *heads* or that she peeked and saw *tails*. However, Bob knows that Alice knows the coin's state. To represent Alice surreptitiously peeking, we use the following action model:

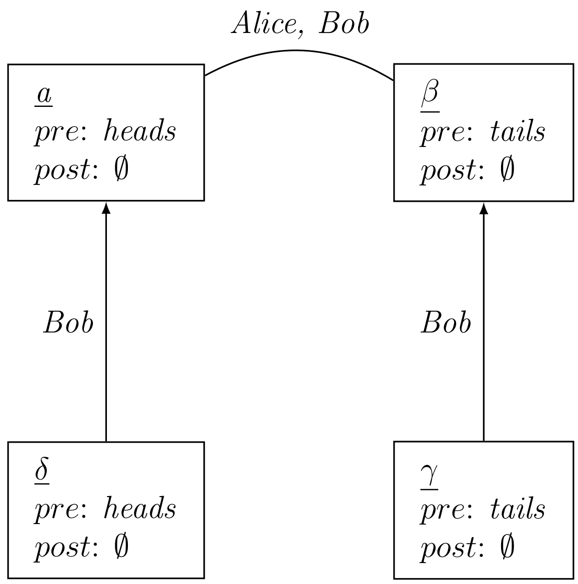


Figure 3.3: Action Model for “Alice secretly peeks”

In Figure 3.3, Alice knows the state of the coin, Bob has the false impression that Alice is, like him, totally ignorant about the coin's state, and Alice knows this about him. But again, the propositions are not true or false in the Action Model itself; it just describes the pre- and post-conditions and the subjective perspectives of the action. Once the Action Model is used during the application of the *Update Function* (described below), in conjunction with a Kripke Structure, it produces worlds with true and false propositions.

Model Relation

Just as R_k^i denotes a relation on worlds, $[\![(A, a)_i]\!]$ denotes a relation on Kripke model-world pairs. It represents the relation that holds between M, w and M', w' when agent i engages in action (A, a) at M, w and causes the world to transition to M', w'.

Precondition Function

The Precondition function, $pre :: Actions \mapsto \varphi$, maps an action to the formula capturing the conditions under which the action can occur. For example, if we assume agents tell the truth, then an announcement action has as a precondition that the announced proposition is true, as with regular Public Announcement Logic [16]. Compare this function with the weakest precondition from Hoare logic [20]. It returns a formula that must be true prior to an action's execution, just as the weakest precondition of Hoare logic is a formula that must be true prior to a program's execution. We impose the condition on the precondition function that it output formulas that are composed entirely of literals in disjunctive normal form.

Postcondition Function

The Postcondition function, $post :: Actions \times \varphi \mapsto (AtProp \mapsto \mathcal{P}(W))$, takes an action token and a formula in disjunctive normal form, and maps to a new valuation function from an atomic proposition to sets of worlds satisfying the proposition. It is similarly a partial function, undefined on worlds where the input DNF formula is false.

This functions is a substitution function, as in the logic of DEL with assignment [46]. A similarity is again seen in the strongest postcondition of Hoare logic. If a consequence of a coin flip action is that $h \equiv$ *the coin shows heads*, then first the precondition function filters out worlds at which the coin flip is not possible, and then the postcondition function assigns all remaining worlds to $V(h)$.

Update Function

The Update function, $update :: (Model \times ActionStruct \times W \times Actions \times Agents) \mapsto (Model \times W)$, takes a Kripke model M, an action structure A, a world from the Kripke model, an action token from the action structure, and an agent i from *Agents*, and returns a new Kripke model-world pair. It represents the effect actions have on models, and is more complicated than other DEL semantics in that actions can change the facts on the ground in addition to the knowledge and belief relations. It is a partial function that is defined iff a model-world pair satisfies the action's preconditions.

$$update(M, A, w, a, i) = (M', w')\ where:$$

1. $M = \langle W, \{R_k^i\}, \{R_b^i\}, w, V\rangle$
2. $A = \langle Actions, \{\chi_k^i\}, \{\chi_b^i\}, a, pre, post\rangle$
3. $M' = \langle W', \{R_k'^i\}, \{R_b'^i\}, w', V'\rangle$
4. $W' = \{(w, a) | w \in W, a \in Actions, \quad \text{and} \quad w \models pre(a)\}$
5. $R_k'^i = \{((w, a), (v, b)) | wR_k^i v \text{ and } \ a\chi_k^i b\}$
6. $R_b'^i = \{((w, a), (v, b)) | wR_b^i v \text{ and } \ a\chi_b^i b\}$
7. $w' = (w, a)$
8. $V'(p) = post(a, pre(a))(p)$

Safety Precondition Function

The Safety Precondition Function, $pre_s :: Actions \mapsto \varphi$, is a more restrictive function than pre. Where pre returns the conditions that dictate whether the action is possible, pre_s returns the conditions that dictate whether the action is safely permissible. This function is the key reason the dynamic approach allows for easy inference from action to safety-critical information. It represents an innovation in dynamic logic that this thesis introduces, because it introduces modality to the transition relation among models. Just as in static modal logics two worlds can be related by an epistemic relation but not by a doxastic relation, one static model can dynamically transition to another via a *mere* action even if it cannot do so *safely*. One could extend the analogy to Hoare logic and introduce a weakest secure precondition in addition to a weakest precondition. The weakest secure precondition is stronger than the weakest precondition, because it imposes more restrictions on the state of the system prior to a program's execution.

3.1.3 Semantics

$\mathcal{DASL}$ has the following relational semantics.

$$
\begin{array}{c}
M, w \models p \Leftrightarrow w \in V(p) \\
M, w \models \neg\varphi \Leftrightarrow M, w \not\models \varphi \\
M, w \models \varphi \wedge \psi \Leftrightarrow M, w \models \varphi \text{ and } M, w \models \psi \\
M, w \models \mathbf{K_i}\,\varphi \Leftrightarrow \forall v,\ wR_k^i v \text{ implies } M, v \models \varphi \\
M, w \models \mathbf{B_i}\,\varphi \Leftrightarrow \forall v,\ wR_b^i v \text{ implies } M, v \models \varphi \\
M, w \models [\mathbf{A}, a]_i\varphi \Leftrightarrow \forall M', w',\ (M, w)[\![(A, a)_i]\!](M', w') \\
\text{implies } M', w' \models \varphi \\
M, w \models [\mathbf{A}, a]_i^S\varphi \Leftrightarrow \forall M', w',\ (M, w)[\![(A, a)_i]\!]^S(M', w') \\
\text{implies } M', w' \models \varphi
\end{array}
$$

Table 3.1: $\mathcal{DASL}$ semantics

The definitions of the dynamic modalities make use of a relation between two model-world pairs, which we now define.

$$
\begin{array}{c}
(M, w)[\![(A, a)_i]\!](M', w') \Leftrightarrow M, w \models \mathit{pre}(a) \\
\text{and } \mathit{update}(M, A, w, a, i) = (M', w') \\
\\
(M, w)[\![(A, a)_i]\!]^S(M', w') \Leftrightarrow M, w \models \mathit{pre}_s(a) \\
\text{and } \mathit{update}(M, A, w, a, i) = (M', w')
\end{array}
$$

Table 3.2: $\mathcal{DASL}$'s dynamic model relations.

3.1.4 Hilbert System

$\mathcal{DASL}$ is axiomatized by the following Hilbert system.

$\mathbf{K_i}\,(\varphi \Rightarrow \psi) \Rightarrow (\mathbf{K_i}\,\varphi \Rightarrow \mathbf{K_i}\,\psi)$	Distribution of $\mathbf{K_i}$
$\mathbf{K_i}\,\varphi \Rightarrow \varphi$	Truth Axiom
$\mathbf{B_i}\,(\varphi \Rightarrow \psi) \Rightarrow (\mathbf{B_i}\,\varphi \Rightarrow \mathbf{B_i}\,\psi)$	Distribution of $\mathbf{B_i}$
$\mathbf{B_i}\,\varphi \Rightarrow \langle \mathbf{B}_i \rangle\,\varphi$	Belief Consistency Axiom
$\neg\mathbf{B_i}\,\varphi \Rightarrow \mathbf{B_i}\,\neg\mathbf{B_i}\,\varphi$	Negative Belief Introspection
$\mathbf{K_i}\,\varphi \Rightarrow \mathbf{B_i}\,\varphi$	Knowledge implies Belief
$\mathbf{B_i}\,\varphi \Rightarrow \mathbf{B_i}\,\mathbf{K_i}\,\varphi$	Evidential Restraintf
$[\mathbf{A}, a]_i p \Leftrightarrow post(a, pre(a))(p)$	Atomic Consequence
$[\mathbf{A}, a]_i \neg\varphi \Leftrightarrow (pre(a) \Rightarrow \neg[\mathbf{A}, a]_i \varphi)$	Action Negation
$[\mathbf{A}, a]_i (\varphi \wedge \psi) \Leftrightarrow ([\mathbf{A}, a]_i \varphi \wedge [\mathbf{A}, a]_i \psi)$	Action Conjunction
$[\mathbf{A}, a]_i \mathbf{K_i}\,\varphi \Leftrightarrow (pre(a) \Rightarrow \bigwedge_{a\chi_k^i\beta} \mathbf{K_i}\,[\mathbf{A}, \beta]_i \varphi)$	Action and Knowledge
$[\mathbf{A}, a]_i \mathbf{B_i}\,\varphi \Leftrightarrow (pre(a) \Rightarrow \bigwedge_{a\chi_b^i\beta} \mathbf{B_i}\,[\mathbf{A}, \beta]_i \varphi)$	Action and Belief
$[\mathbf{A}, a]_i^{\mathcal{S}} p \Leftrightarrow post(a, pre_s(a))(p)$	Safe Atomic Consequence
$[\mathbf{A}, a]_i^{\mathcal{S}} \neg\varphi \Leftrightarrow (pre_s(a) \Rightarrow \neg[\mathbf{A}, a]_i^{\mathcal{S}} \varphi)$	Safe Action Negation
$[\mathbf{A}, a]_i^{\mathcal{S}} (\varphi \wedge \psi) \Leftrightarrow ([\mathbf{A}, a]_i^{\mathcal{S}} \varphi \wedge [\mathbf{A}, a]_i^{\mathcal{S}} \psi)$	Safe Action Conjunction
$[\mathbf{A}, a]_i^{\mathcal{S}} \mathbf{K_i}\,\varphi \Leftrightarrow (pre_s(a) \Rightarrow \bigwedge_{a\chi_k^i\beta} \mathbf{K_i}\,[\mathbf{A}, \beta]_i^{\mathcal{S}} \varphi)$	Safe Action and Knowledge
$[\mathbf{A}, a]_i^{\mathcal{S}} \mathbf{B_i}\,\varphi \Leftrightarrow (pre_s(a) \Rightarrow \bigwedge_{a\chi_b^i\beta} \mathbf{B_i}\,[\mathbf{A}, \beta]_i \varphi)$	Safe Action and Belief
$[\mathbf{A}, a]_i \varphi \Rightarrow [\mathbf{A}, a]_i^{\mathcal{S}} \varphi$	Inevitability
$\langle \mathbf{A}, a \rangle_i \varphi \Rightarrow \mathbf{B_i}\,\langle \mathbf{A}, a \rangle_i^{\mathcal{S}} \varphi$	Minimum Rationality
From $\vdash \varphi$ and $\vdash \varphi \Rightarrow \psi$, infer $\vdash \psi$	Modus Ponens
From $\vdash \varphi$, infer $\vdash \mathbf{K_i}\,\varphi$	Necessitation of $\mathbf{K_i}$
From $\vdash \varphi$, infer $\vdash [\mathbf{A}, a]_i \varphi$	Necessitation of $[\mathbf{A}, a]_i$

Table 3.3: Hilbert System of $\mathcal{DASL}$

Above are the axioms characterizing the logic, including the reduction axioms translating formulas with dynamic modalities into purely static formulas.

The reduction axioms are recursively defined on φ, terminating with propositional atoms. When a dynamic operator, either $[\mathbf{A}, a]_i$ or $[\mathbf{A}, a]_i^{\mathcal{S}}$, applied to a propositional atom p, is translated to a static formula, first the *pre* (or pre_s) function is called on the action a, which if defined, is then passed through the *post* function to substitute the atoms that change as a result of the action. If this translation process results in an implication that p, then the dynamic formula is true; otherwise, false.

We proceed by defending the axiom schema for the static foundation, then continue with the dynamic extensions.

3.1.5 Static Base

Like other dynamic logics for knowledge and belief, $\mathcal{DASL}$ consists of a static base of axiom schemas for the knowledge and belief operators. This section explains and defends $\mathcal{DASL}$'s static base, as they represent a novel axiomatization of formal epistemology and a contribution to the literature by this thesis.

The standard axiom schema for the knowledge operator, whether for a wholly static logic or for the static base of a dynamic system, is that of the $\mathcal{S}5$ operator, described earlier in section 2.2.2. This axiom schema for the knowledge operator's role in classical economics was first formalized by Aumann in [66]. In this model, the belief operator is not necessary, because agents do not form false beliefs. They either know a proposition, or they know that they do not know it. This is not the case for humans in most epistemic situations.

The static base of $\mathcal{DASL}$ logic attends to the formalization of agents who can make decisions based on false beliefs, but whose beliefs are all justified. It is a weaker model than $\mathcal{S}5$, while still making an idealization about its agents.

Recall the static base of $\mathcal{DASL}$.

$\mathbf{K_i}\,(\varphi \Rightarrow \psi) \Rightarrow (\mathbf{K_i}\,\varphi \Rightarrow \mathbf{K_i}\,\psi)$	Distribution of $\mathbf{K_i}$
$\mathbf{K_i}\,\varphi \Rightarrow \varphi$	Truth Axiom
$\mathbf{B_i}\,(\varphi \Rightarrow \psi) \Rightarrow (\mathbf{B_i}\,\varphi \Rightarrow \mathbf{B_i}\,\psi)$	Distribution of $\mathbf{B_i}$
$\mathbf{B_i}\,\varphi \Rightarrow \langle \mathbf{B}_i \rangle\,\varphi$	Belief Consistency Axiom
$\neg\mathbf{B_i}\,\varphi \Rightarrow \mathbf{B_i}\,\neg\mathbf{B_i}\,\varphi$	Negative Belief Introspection
$\mathbf{K_i}\,\varphi \Rightarrow \mathbf{B_i}\,\varphi$	Knowledge implies Belief
$\mathbf{B_i}\,\varphi \Rightarrow \mathbf{B_i}\,\mathbf{K_i}\,\varphi$	Evidential Restraint
From $\vdash \varphi$ and $\vdash \varphi \Rightarrow \psi$, infer $\vdash \psi$	Modus Ponens
From $\vdash \varphi$, infer $\vdash \mathbf{K_i}\,\varphi$	Necessitation of $\mathbf{K_i}$

Table 3.4: Static Base of $\mathcal{DASL}$

As far as we can tell, this axiom scheme is novel, and disagrees with other attempts to model the interaction of knowledge and belief. The primary point of departure is that the knowledge operator is no longer and $\mathcal{S}5$ operator, but merely a T operator. Thus, it is severely weakened as a modal operator, but the corollary to this fact is that knowledge no longer has the imposing necessary conditions of positive and negative introspection.

This section presents the desirable features that a static logic should have, and along the way argues against $\mathcal{S}5$'s appropriateness as an epistemic base. Before beginning, we briefly establish a consistent informal interpretation of knowledge and belief.

Defense of $\mathcal{S}5$ for knowledge rests on an interpretation of the knowledge operators not as capturing what agents *actively* know at a moment, but rather what their knowledge allows them to potentially infer. Thus, the Distribution axiom is not a license for logical omniscience at a moment, but rather describes what is logically inferrable from what is known by an agent. Similarly for positive and negative introspection. However, this defense does not quite provide positive reasons for accepting the $\mathcal{S}5$ axiom schemas. We present arguments against the positive and negative introspection axiom schemas that show them to be inappropriate even on an *inferrable* or *subjunctive* interpretation of knowledge, where an agent knows φ if and only if her present knowledge allows her to come to know φ after sufficient reflection or inference power.

Doxastic logics similarly tend to adopt a subjunctive interpretation, where an agent believes φ if and only if her current beliefs would allow her eventually infer φ. Both of these interpretations come straight out of Hintikka in [52]. These explications suffer an obvious problem of circularity that we do not wish to suffer here. We wish to start from a more fundamental concept, and use it to build the notions of knowledge and belief. The more fundamental concept we use is *evidence*.

We shall say that an agent knows that φ when all of her objectively observable evidence entails φ. Thus, we are still friendly to the subjunctive interpretations, because it reduces to an inferrability condition. However, we lack the problem of circularity. Similarly, for belief. We shall say that an agent believes that φ when all of her subjectively available evidence entails that φ. We shall spell out the details to these components below.

The formal theory of epistemology espoused by this dissertation departs from those typical of the field. We reject $\mathcal{S}5$, we require the mixing of knowledge and belief operators, and we depend on more fundamental notions of evidence. Now we proceed to lay out what we take to be the desirable features of any epistemic logic.

3.1.6 Desiderata of a Static Base

In constructing our static base, we must identify the desirable features to be achieved. A model of knowledge and belief for human-like agents can be assessed by how well it satisfies these desirable features.

First, the model should be well-balanced in the trade off of normativity *vs.* realism so that it can be used to analyze real-world cases as well as idealized formal cases. It scores well on realism to the extent that it represents the reasoning of human-like agents in environments of relative messiness, like the real world. It scores well on normativity to the extent that its departures from realism are improvements over reality to be strived for. Not all idealizations in models are normative in character, as we shall see.

Second, the model should make sense from an epistemological standpoint, *viz.* the philosophical considerations of epistemologists. To construct a system of epistemic logic is to construct a formal theory of epistemology for *some* type of agent, whether it is a perfect deductive reasoner like a computer over a relatively small database ontology or *homo economicus* in well-structured strategic situations, or an imperfect reasoner like a human in the real world. So far, the latter type of agent has been largely neglected by formal efforts, because attention has been on properties of formal systems, see below, rather than primarily on their philosophical foundations.[1]

Third, the model should be a sound and complete logical system to serve as a static foundation for an extension that treats actions. Just because attention must be paid to the philosophical foundations and motivations for a formal system does not mean that the formal system is off the hook for the standard desiderata of a logic. The logic must avert a collapse of belief and knowledge. The logic must abide by the principle of indistinguishability, where an agent cannot distinguish between possible worlds it cannot rule out.

1. Normative-Realism Trade-Off A model of rationality is situated in a spectrum trading off realism for normative evalutation and prescription. A totally idealized model like that of game theory or rational choice theory is entirely normative. There is no doubt that the agents being modeled are not anything like human

[1] One significant exception to this claim is the study of bounded rationality, which imposes limitations on the number of iterations of i knows that k knows that i knows ... We regard this approach as valuable, but taking a different approach than ours. Typically, bounded rationality approaches still assume that knowledge is $\mathcal{S}5$, while we relax this assumption because we regard it is philosophically untenable.

beings. The model presumes normativity by showing that acting like an ideally rational agent results in optimal payoffs possible. So, for humans who want to optimize their decisions for payoffs, acting like ideally rational agents is best. However, ideally rational agents have properties that real humans lack, so unless a human can run a simulated agent on a correctly formalized model of a situation, she is unlikely to identify the correct action. After the fact, with a correctly formalized model of the situation, her decision can be analyzed in a post-mortem, but post-mortems never do any good for their subjects.

Theories of bounded rationality achieve more realism, but the trade off of realism and normativity is not clearly balanced. By keeping the $\mathcal{S}5$ knowledge operator and limiting the agents' epistemic depth[2], theories of bounded rationality seek to model actual human behavior in the rarefied world of games of perfect information, where the agents know the payoff structure and the game tree or game matrix. This is fine for modeling humans in behavioral economics experiments, where they are sitting at a table with a formal description of the game in front of them. But for the messier world of cockpits, emergency rooms, power plants, and automobiles, this assumption of $\mathcal{S}5$ knowledge fails. The difference is between *closed* worlds and *open* worlds. A formal game environment is a closed world, meaning the human is aware of all of the relevant propositions and their truth values, *i.e.,* the game structure. The real world's environment is an open one, where the set of relevant propositions is a mystery, and the truth values of the propositions known to be relevant is a matter of uncertainty.

Likewise, theories of bounded rationality with $\mathcal{S}5$ knowledge do not offer clear normative prescriptions or evaluative guidelines. Rather, it appears to be an attempt to makes sense of the following finding of behavioral economics: In some situations, human agents depart from the Nash equilibrium, and they both end up with higher payoffs. The centipede game is a good example. The Nash equilibrium states that

[2] i knows that j know that i knows that... depth

player A should immediately take the money on the table, whereas experiments show that humans regularly let the pot build up until a few steps before the end, one of the players realizes what will happen, and takes the majority of the pot. From the perspective of maximizing expected payoffs, it is hard to criticize either player, as each achieved higher payoff from this outcome than they would have in the Nash equilibrium. A theory of bounded rationality takes this at face value, and seeks to build a model for it. With such a model, the hope is to offer general normative guidance for agents with bounded rationality. However, to examine the particular outcome of a centipede game, supposing player B is the one with the minority payoff, we can truly say that B will regret not having taken the majority pot one step earlier, and properly that he *should* have, because then he would have had a higher payoff, and so this outcome is not in subgame perfect equilibrium. If the game is analyzed as a sequence of decisions, the final decision the minority-payoff-receiving player makes is the wrong one, normatively. Bounded rationality with S5 knowledge explains this as bad luck determined by the structure of the game and the first (or second) mover advantage, depending on the players' bounds on epistemic depth. But it is nonetheless true that B could have and should have taken the majority pot one step earlier, and this is not a feature of games with outcomes determined by first (or second) mover advantage, like the game of nim with a 3-4-5 setup (or nim with a 1-3-5-7 setup).

Other attempts to relax the assumptions of classical game theory abandon the assumption of $\mathcal{S}5$ knowledge and turn toward probability theory in order to capture uncertainty. Uncertainty of various sorts can be adequately modeled in this fashion, including of the game's structure, and the type of the other agents, *e.g.* whether they are a type of agent capable of reasoning 3 epistemic levels deep or merely 2 levels deep. This is realistic for translation to the real world, and normative in that the probabilistic reasoning must abide by the axioms of probability theory. However, humans are not perfect Bayesian reasoners, so this model is unable to formally represent the mistakes humans frequently make when reasoning probabilistically without additional adjustments. A realistic model should be able to model mistakes systematically in such a way that valid inferences can be made from mistakes in belief or action.

The model presented here abandons the $\mathcal{S}5$ knowledge assumption in order to model how beliefs and knowledge logically relate for a new type of agent, only slightly improved over a normal human agent. This agent knows some things, believes more than she knows, and has false beliefs. She is different from normal humans in that she believes a proposition only if she has a good justification to do so, like the observation of evidence that supports the proposition, or because she believes she has made a valid inference to that proposition. The agent is realistic to the extent that a very careful human reasoner can achieve this level some of the time, when it really matters to do so, and this model is normative to the extent that an agent with more true beliefs about the environment will make better decisions after the dynamic extension of actions is layered on top.

Along similar lines but in pursuit of different theoretical interests is the so-called Belief-Desire-Intention (BDI) model of action. This model seeks to explain the modules responsible for actions among humans and presumably other potential agents. The model holds that beliefs about the world, desires for how the world could be, and intentions to realize those desired states, are jointly sufficient and individually necessary conditions for actions. This model is related to the model defended here in that its agents can make mistakes due to false beliefs. However, the model is primarily a theory of action and motivation, not a theory of rational action, which we seek to establish.

2. Philosophical Foundations A formal epistemic theory should strive for adequate philosophical grounding in good epistemology. Hintikka took this very seriously in his 1967 book on *Knowledge and Belief* [52], which serves as a seminal work in the field of formal epistemology. Hintikka dedicates a great portion of the book to exploring how his formal system handles the intuitive judgments of philosophers regarding ordinary language statements, which was the primary method at the time. An epistemic logic divorced from a philosophical foundation is no longer an epistemic logic for reasoning about human-like knowledge. Some idealized epistemic logics can serve as logics for reasoning about machine knowledge in closed worlds, but as machines come to interact with larger and more complex parts of the world, these idealized logics diminish in adequacy, as computational limitations prevent the machine from implementing even a perfect Bayesian reasoning system. Thus, careful consideration of an epistemic logic's philosophical foundation is in order.

Epistemologists primarily seek a theory of justification that can avoid Gettier-style counterexamples [14] and a theory of knowledge that can solve the problem of skepticism. A system of formal epistemology should respect these questions by coherently addressing them. We consider it required for a system of epistemic logic to be evaluated as a theory of epistemology. The epistemic logic of $\mathcal{S}5$ with common knowledge does not strive for this. We briefly recount some broad positions of interest from epistemology that our model should take a considered position on.

The Space of Justification Theories Epistemology is the study of knowledge, and knowledge is widely held to require truth and justified beliefs in some way, with perhaps additional conditions being required for a jointly sufficient set of conditions. Our model defines truth through the semantic $\models$ relation between worlds and formulas, and knowledge explicitly requires belief via the axiom *Knowledge implies Belief*. Justification remains for us to elaborate, although we do not model it explicitly with a justification modal operator. A theory of epistemic justification must respond to the infinite regress argument which concludes that justified beliefs are impossible. The infinite regress argument against justification runs as follows:

1. A belief that φ is justified if and only if the belief that φ is justified by other justified beliefs.
2. A justification chain of beliefs either: (a) terminates with unjustified base beliefs, or (b) is circular, or (c) is infinite and non-circular.
3. If (a), then no beliefs are justified.
4. If (b), then no beliefs are justified.
5. If (c), then no beliefs are justified.
6. Therefore, justification is impossible for beliefs.

A theory of justification is shaped by how it responds to this regress argument. Denying premise (1) requires a theory to explain what, other than beliefs, justifies beliefs. Popular candidates include experiential mental states. This involves justification flowing from something that is not true or false to something true or false, and thus is not easily formalized in a logic. Denying premise (3) pushes one toward a *foundationalist* theory, where the task then is to describe how a foundation of basic beliefs can be justified, and how the principle of premise (1) does not properly apply to them. Denying premise (4) pushes one toward *coherentism*, where the task is to justify how a circular chain of beliefs is not viciously circular. Denying premise (5) pushes one toward *infinitism*, where the task is to justify how an incomplete infinite chain can have justified constituent beliefs.

In addition to responding to the regress argument, a theory of epistemic justification must present an analysis of what constitutes justification. In addressing the regress argument, a theory takes a position about the structure of the justification relation and whether believes are of a single type or of multiple types. A theory must then take a position about the definition of the justification relation so as to avoid the problem of skepticism.

Skepticism We take it as a basic assumption that human knowledge is possible, and therefore that the epistemic position of general skepticism is false. Therefore, in constructing a formal model of epistemic logic, that model should not lead to general skepticism when applied to humans.

In addition to addressing the infinite regress argument, theories of justification must address an argument from logical closure. The logical closure argument runs as follows:

1. i knows that φ only if i knows $\neg$*skeptical hypothesis*, where *skeptical hypothesis* is some proposition that entails $\neg\varphi$.

2. i does not know $\neg$*skeptical hypothesis*.

3. Therefore, i does not know φ.

Denying (1) denies that knowledge is closed under logical entailment, and denying (2) requires care so that the theory will generalize and avoid all potential skeptical scenarios. Denying (1) presents a significant problem for formal epistemologists hoping to create a normal modal logic of knowledge. Common attempts to deny (2) include placing knowledge in a direct justification relation with the real world via causation, or defending a position that some hypotheses can be justifiably ignored from consideration.

A theory of justification is internalist if it holds that the believer is justified in believing some proposition φ if and only if she has awareness or access to the basis for believing φ. Externalist theories deny this, and hold that external facts, *e.g.* the causal process that produced the belief, are what justify beliefs and produce knowledge. Theories of justification sometimes apply to both knowledge and justified belief, and sometimes to just one or the other.

An example internalist view is that, through reflection, one can know that she knows a propostion, and know that she believes but does not know a proposition [?]. This corresponds to including both introspection axioms for the knowledge operator. A weaker internalist view that we endorse is *evidentialism*, which holds that a belief is justified if and only if the believer has evidence supporting the belief. The task of an *evidentialist* theory is to spell out what evidence is.

An example externalist position is that the process that produced the belief that φ be a *reliable* one, in some sense to be defined by a given *reliablist* theory. This avoids the problem of skepticism by placing the justification relation between the world and the knowledge of the agent, with or without the agent's awareness of this justification being present. Thus, if φ is true, and i's belief that φ is produced by a reliable process in the real world, then i knows that φ and therefore knows that the *skeptical hypothesis* is false, because inferring that from their being inconsistent is also a reliable process. This approach clearly rejects both positive and negative introspection axioms, as introspection plays no role in the justification of knowledge.

The problem for an epistemic logician constructing a formal theory is to select an appropriate target for formalization. Most recent work avoids this problem by instead focusing on technical innovations applied to toy problems. We deny ourselves that route, and in doing so much make decisions regarding the above problems in a way that we can philosophically defend.

As with Hintikka's work, the test of an epistemic logic's adequacy is its ability to correctly handle touchstone cases from the philosophical literature, or better yet, to adequately explain problems that currently lack convincing explanations. For example, the so-called Moore's sentence is a problem pervasive in epistemology and the philosophy of language, attributed to G.E. Moore [37] by Wittgenstein [75]. The standard Moore's paradox to be solved is a statement of the form,

$$\varphi, \textit{ but I don't believe that } \varphi.$$

The alleged paradox of Moorean sentences is that they are not formal contradictions, but they nonetheless seem like they should be. Hintikka solves this problem by first asserting that his logic is meant for actual communicated statements, not merely sentences, and he distinguishes three new types of implication under study: virtual, epistemic, and doxastic. All implications are established by checking for a model where the antecedent is satisfiable while the consequent is falsifiable. If no such model is possible, then the implication is established. Virtual implication refers to checking for satisfiability of the negated implication in a model. Epistemic implication refers to checking for satisfiability of the negated implication with a knowledge operator distributed over it, and similarly for doxastic implication. Thus, Hintikka solves Moore's paradox by searching for a model that satisfies the above statement. Such a statement is falsifiable, and so virtual implication from φ to $\mathbf{B_i}\,\varphi$ fails. Hintikka turns then to doxastic implication, which involves distributing $\mathbf{B_i}$ over the implication under inspection:

$$\mathbf{B_i}\,\varphi \wedge \neg \mathbf{B_i}\,\mathbf{B_i}\,\varphi.$$

This is justified by the principle that people normally believe what they assert as statements. Hintikka's doxastic logic has a transitive doxastic relation, so it takes positive introspection about belief as an axiom, and therefore the above formalization is not satisfiable in a model (and so is doxastically implied).

3. Formal Properties A system of logic ought to be sound and complete. An epistemic logic with both belief and knowledge operators must avoid an equivalence collapse of the knowledge and belief operator, which can occur if one is not careful in choosing axioms. However, the logic must satisfy the intuition that for v to be epistemically possible from w, v and w should in some sense be indistinguishable to the agent. Finally, for a logic of sufficient expressive power, namely one with agents who can reason about their own reasoning, it must avoid a special obstacle to an agent trusting its own conclusion, dubbed the Löbian Obstacle by researchers in the foundations of artificial intelligence. Modal logics with reflective reasoners must address this obstacle.

Soundness and Completeness Soundness and completeness are familiar properties for students of logic. They are briefly mentioned here as a reminder.

Definition 3.1.0.1 (Soundness)**.** *Logic $\mathcal{L}$ is* sound *if and only if* $\models \varphi$ *implies* $\vdash \varphi$,

where $\models \varphi$ is semantic validity of φ and $\vdash \varphi$ is axiomatic derivability.

Definition 3.1.0.2 (Completeness)**.** *Logic $\mathcal{L}$ is* complete *if and only if* $\vdash \varphi$ *implies* $\models \varphi$.

Avoid KB Collapse Knowledge and belief are distinct. There are two distinguishing characteristics between them. The first is that belief and knowledge sit in a part-whole relationship to each other. i knows that φ only if i believes that φ, with other necessary conditions that are jointly sufficient. The second is that knowledge of a proposition requires that the known proposition be true, while belief requires no such truth condition. If an epistemic logic with operators for belief and knowledge collapses them into equivalence, such that $\mathbf{B_i}\,\varphi \Leftrightarrow \mathbf{K_i}\,\varphi$, it has gone wrong, and is not acceptable.

An offending theorem flagged in the literature as responsible for equivalence collapse is $\mathbf{B_i}\,\varphi \Rightarrow \mathbf{B_i}\,\mathbf{K_i}\,\varphi$, which is our axiom *Evidential Restraint*. We must show that our system avoids $\mathbf{K}\,\mathbf{B}$ collapse.

Indistinguishability Many find it intuitive to speak of the epistemic relation as an equivalence or indistinguishability relation, which requires it to be symmetric, transitive, and reflexive. The epistemic relation in S5 is Euclidean and reflexive, which together yield symmetry and transitivity, and therefore an equivalence relation. Equivalence relations satisfy and indistinguishability relation because each world in the equivalence class is indistinguishable from each other relative to the relation defining the class. We must show that our axiom schema can satisfy the intuition behind epistemic indistinguishability.

We proceed now to our arguments that $\mathcal{DASL}$'s static base fares well on these counts.

3.2 Interpreting $\mathbf{K_i}$ and $\mathbf{B_i}$

Epistemic logic can produce some counterintuitive results if the knowledge operator is interpreted inappropriately for the system. For example, the standard Distribution Axiom of knowledge, $\mathbf{K_i}\,(\varphi \Rightarrow \psi) \Rightarrow (\mathbf{K_i}\,\varphi \Rightarrow \mathbf{K_i}\,\psi)$ states that knowledge is closed under implication. This presents several problems when the operator is interpreted as "agent i actively knows right now that φ". For empirical knowledge, the threat of global skepticism looms, because presumably i knows that she exists as an embodied agent in the external world only if she is not a mind in a simulated reality. Therefore, she knows she is an embodied agent in the external world only if she knows she is not a mind in a simulated reality. By hypothesis, the skeptic would say, she does not know she is not a mind in a simulated reality. Therefore, she does not know she is an embodied agent in the external world. For statements of pure logic, the Distribution Axiom states that agents know all logical truths.

The standard interpretation in the literature for an epistemic logic with just a knowledge operator is that "agent i *implicitly* knows that φ," meaning something like the claim that φ is a live candidate for knowledge for i, should i ever examine the proposition. For doxastic logic, the same concern arises for belief, as it has the Distribution Axiom as well. The interpretation here is usually something like "agent i's beliefs *commit* i to φ," or "i is disposed to believe that φ. In both cases, the interpretation moves away from an active, explicit occurrence of the proposition in the agent's mind in order to avoid the weirdness of closure. We adopt this interpretation, but further analyze the notions of epistemic and doxastic possibility in order to avoid circularity of our definitions and to avoid the problem of logical omniscience for our agents.

Our interpretation is comparable to a debate that occurs concerning the interpretation of probabilities among frequentists and Bayesians. Frequentists hold that probabilities represent objective frequencies that occur in the world, which can be counted through controlled experiment, and which should directly map to the probability for an event. For Bayesians, probabilities represent subjective uncertainty in the mind of a reasoner. Rather than constructing a model based on one or the other of these notions, we construct a model that includes both the notion of objectively available evidence and subjectively available evidence.

Definition 3.2.0.1 (Objectively Available Evidence)**.** *Evidence is* objectively available *for proposition φ if and only if some source of information exists in the actual world that supports the proposition φ.*

What is it for a source of information to support a proposition? We take it to be a primitive notion of an event with high mutual information with the proposition, and we leave it at that. Future work can pursue the relationship between information theory and epistemic possibility relations, but we are satisfied to leave things at this level of abstraction. We do, however, provide some brief examples.

Definition 3.2.0.2 (Subjectively Available Evidence)**.** *Evidence is* subjectively available *to agent i for proposition* φ *if and only if i has direct mental access to some information, accurate or inaccurate, that supports the proposition* φ*.*

When a coin lands with *heads* facing up, the light reflecting off the coin's surface is a source of information that supports the propositions that "the coin landed *heads* up," and "the coin is unfair and has two *heads*," and "the coin landed *tails* down." This is because evidence underdetermines the theories it supports, and so multiple propositions are always supported by the same source of information. The light reflecting off the surface is objectively available evidence for these propositions. If anyone notices the light reflecting off the coin, then this evidence becomes subjectively available to them. From the actual world, doxastic relations spider-web outward to possible worlds at which those propositions are true, and she considers those worlds possible. The event itself caused those worlds to become epistemically possible.

Thus, from the actual world, epistemically possible worlds are created by objectively available evidence, and doxastically possible worlds are created by subjectively available evidence. Seeing a coin land *heads* is subjectively available evidence to the agent that sees it, so doxastically possible *heads*-worlds proliferate and possibly overtake all *tails*-worlds, in which case the agent believes that the coin is *heads*.

In the case of false beliefs, suppose Bob is an early bird in 1948 America, and he reads the headlines that *DEWEY DEFEATS TRUMAN*. Here we have objectively available evidence, a source of information in the actual worlds that supports a proposition. One of the propositions supported is that Dewey defeated Truman in the 1948 United States Presidential election. This proposition happens to be false, but it nonetheless has objectively available evidence supporting it. Thus, objectively available evidence does not necessarily guarantee truth for the propositions it supports. It merely lends support to them. When Bob reads the newspaper, he may or may not form the belief that Dewey won the election, depending on the other evidence he has perceived.

Epistemically possible worlds are tied to the actual world in some sense, and doxastically possible worlds are tied to an agent's mental consideration. When objective evidence is perceived and interpreted, it becomes subjective evidence, and when an agent reflects on the evidence itself, it becomes subjectively objective. This is the intuitive gloss of the model-theoretic notion of composing the doxastic and epistemic relations. When world wR^i_bv and vR^i_ku, agent i is reflecting on the objectively available evidence at w, but because it is mediated by her reflection, it must be behind a subjective layer.

Thus, we have provided an interpretation of the possibility relations underlying knowledge and belief, which is not often done in the literature. We now present our theory's notion of justification and its response to skepticism.

3.2.1 Justification

Justification in our logic is not represented explicitly with a justification operator, as in Artemov and Nogina [1]. There, they formalize justification and knowledge with a formula like $t : \varphi$, where t is a justification (a proof-like object) for φ polynomial in its length. They adopt $\mathcal{S}4$ as the static base epistemic logic, and interpret knowledge in the usual implicit or subjunctive way. They include a connective axiom schema $t : \varphi \Rightarrow \mathbf{K}\,\varphi$, read "if φ is justified then φ is knowable". We seek a weaker notion of justification here, under the notion that false propositions can nonetheless be justified by evidence. Rather than representing justification explicitly in the object language, we bury it in the underlying model theory, relating evidence to the possibility relations over worlds. We capture the notion of justification in the object language with the axiom schema *dence in Belief*.

Recall *Evidential Restraint*: $\mathbf{B_i}\,\varphi \Rightarrow \mathbf{B_i}\,\mathbf{K_i}\,\varphi$. An agent believes φ only if *her reflection* on all the objectively available evidence yields only possible worlds where φ is true. It places a constraint on the beliefs agents form such that they have direct awareness of solid evidence for the proposition. Yet they can still be mistaken, because it is their reflection on the evidence that supports the proposition, not the objectively available evidence itself. Unpacking this, we accomplish several desiderata simultaneously. First, we make a normative idealizing assumption about agents, that all their beliefs are constrained by subjectively available evidence. Second, we capture an internalist theory of justification, where an agent's belief is justified by the evidence available to her[3]. This theory is philosophically defensible, even if it is not unanimously held (few philosphical positions are). Finally, the distinction this theory draws between objective and subject perspectives helps account for powerful intuitions about knowledge.

As a balance between the normative and descriptive, this theory achieves a plausible and desireable trade off. If an agent forms a belief that φ while having defeating evidence directly accessible to her, we are right to criticize her for forming and continuing to hold the belief. This follows the tradition set by Hintikka that the intuitive standard for evaluating formulas of epistemic logic is rational defensibility. On this ground, the axiom is intuitively plausible as a normative claim.

Recall the sentences used to test intuitive validity, called Moorean sentences, of the form, "φ, but I do not know that φ." The apparent absurdity of such a sentence is belied by the lack of a formal contradiction. Hintikka resolves this issue by defending that claim that when honest agents proclaim sentences, minimally, they believe them to be true. So when such a sentence is uttered, it is equivalent to a formula $\mathbf{B_i}\,\varphi \wedge \mathbf{B_i}\,\neg\mathbf{K_i}\,\varphi$. From this, we can deduce a contradiction if we assume *Evidential Restraint*.

$\mathbf{B_i}\,\neg\mathbf{K_i}\,\varphi$ implies $\langle\mathbf{B}_i\rangle\,\neg\mathbf{K_i}\,\varphi$, which is equivalent to $\neg\mathbf{B_i}\,\mathbf{K_i}\,\varphi$. Thus, we have $\mathbf{B_i}\,\varphi$ and $\neg\mathbf{B_i}\,\mathbf{K_i}\,\varphi$. However, from the former, and the axiom of *Evidential Restraint*, $\mathbf{B_i}\,\mathbf{K_i}\,\varphi$ holds. Thus, we have a contradiction. The fact that the axiom schema helps

[3]The specific internalist theory we adopt is called evidentialism; cf Feldman and Conee in [13].

explain the absurdity of the Moore sentence lends philosophical support to its being a normative constraint on belief. The fact that the constraint is a subjective one, *i.e.* relative to an agent's subjectively available evidence, makes it a realistically achievable idealization, unlike positive and negative introspection of knowledge.

Recall that the problem of logical omniscience is that all propositional tautologies are theorems of the logic, and by the rule of $\mathbf{K_i}$ Necessitation, all theorems are known. The Distribution Axiom of $\mathbf{K_i}$ over implication says if a theorem is known, and i knows that the theorem φ implies ψ, then i knows ψ. Since knowledge implies belief, $\mathbf{B_i}\,\psi$. Thus, i knows and believes every mathematical theorem. Our solution depends on a function that partitions the set of atomic propositions into an *active* set and an *inactive* set. This activation function constructs a model with atomic propositions that the agent considers relevant. It has the effect of constraining $AtProp$ to a finite subset of its original self.

Definition 3.2.0.3 (Active Set)**.** For a model M with set of atomic propositions $AtProp$, the active set $active \stackrel{def}{=} \{p \mid p \in AtProp \textsf{ and } activate(p) = True\}$.

Definition 3.2.0.4 (Inactive Set)**.** For a model M with set of atomic propositions $AtProp$, the inactive set $inactive \stackrel{def}{=} \{p \mid p \in AtProp \textsf{ and } p \notin active\}$.

These sets formalize the notion that a human-like agent can actively consider only a limited number of atomic propositions at a time. When an atomic proposition is *inactive*, it will be excluded from the model. Just as we build simple Kripke structures with only a few atomic propositions included, so do humans actually reason this way, with only a few atomic propositions under active consideration at a time.

Now we can ameliorate the problem of skepticism by appealing to our notion of evidence as the source of possible worlds, and the fact that models include only active atomic propositions. For logic and mathematics, evidence consists in proofs. A proof is a set of sentences realizing the sequence of deductions from axioms and theorems to the proved theorem. A proof is tokenized when it is instantiated in the form of writing, speech, or via mechanization software. We say a tokenized proof constitutes objectively available evidence for a theorem, and perceiving that tokenized proof constitutes subjectively available evidence. Thus, even though there may be a Platonic proof that P!=NP out there waiting to be instantiated, until it is instantiated, no such objective evidence is available to human-like reasoners. That's why humans truly and without contradiction say that they do not know whether P=NP.

Furthermore, we can deny the problematic part of the problem of logical omniscience by denying that agents know all tautological logical implications. They know the ones with tokenized proofs. The ones without tokenized proofs are actually unknown to us. In order to identify an inconsistency, a critic would have to instantiate a proof of a previously unknown theorem, and in so doing, the logic under consideration here would update to include it as objectively available evidence for formulas consisting of atomic propositions that are active.

3.3 Indistinguishibility Satisfied

Many find it intuitive to speak of the epistemic relation as an equivalence or indistinguishability relation, which requires it be an equivalence relation. Euclidicity, combined with the reflexivity underlying the Truth Axiom and transitivity of positive introspection, yields such an equivalence relation. It makes sense to say that world v is epistemically possible from world w for agent i if i cannot tell the two worlds apart, *i.e.* they are indistinguishable. The power of this intuitive appeal presents a challenge for us. We must show that our axiom schema can satisfy the intuition behind epistemic indistinguishability without imposing symmetry on the epistemic relation.

The key to achieving this is through the composition of the doxastic relation with the epistemic relation. From a first person perspective, our model does collapse belief and knowledge together. That means that an agent cannot distinguish her mere beliefs from her genuine knowledge. Let $R^i_{k/b}(w)$ denote the extension of the epistemic/doxastic relation of w. We also define the following mathematical notion on the relations.

Definition 3.3.0.1 (Doxastic Capture)**.** *For all $w, v, u \in W$, the doxastic capture of w, $dc(w)$ is defined recursively as follows:*

1. $v \in dc(w)$ iff $wR^i_b v$

We say world w is *path connected* to v if there is some number of relational steps, R^i_b and R^i_k interchangeably, from w to v. This yields the following theorem.

Theorem 3.3.1 (Doxastic Path Capture)**.** *For all $w, v, u \in W$, if $wR^i_b v$ and v is path connected to u, then $v, u \in dc(w)$.*

Proof. $v \in dc(w)$ trivially from the base case of the definition of $dc(w)$. For u, we iteratively simplify the path between v and u in the following way. Since u is path connected to v, there is some number n of relational steps from v to u. If it is by n R^i_b steps, then immediately $vR^i_b u$, since R^i_b is transitive. Then $wR^i_b u$ as well, by transitivity through v.

If there are n steps from v to u interchanging R^i_b and R^i_k, then let $x \in W$ be the first world path connected to v by a R^i_k step, with $m < n$ R^i_b steps between v and x. If $m = 1$, then $w(R^i_k \circ R^i_b)x$, and therefore $wR^i_b x$, yielding $x \in dc(w)$. Otherwise, there are $m-1$ R^i_b steps between v and x. Examine the world at step $m-1$, call it v'. We have that $vR^i_b v'$ by transitivity, and therefore $wR^i_b v'$. So $v' \in dc(w)$. Since $v'R^i_k x$, $w(R^i_k \circ R^i_b)x$, and so $wR^i_b x$ and $x \in dc(w)$. This process repeats, simplifying R^i_b steps by transitivity, and simplifying $(R^i_k \circ R^i_b)$ steps by the subset relation to R^i_b, until reaching u. □

We also have the following.

Theorem 3.3.2 (Doxastic Capture is Equivalence Relation)**.** *An equivalence relation is reflexive, transitive, and Euclidean. The doxastic capture of w is such a relation for both R^i_b and R^i_k.*

Proof. It suffices to show that R^i_b is reflexive and R^i_k is transitive and Euclidean under $dc(w)$. Suppose $v \in dc(w)$.

R^i_b-Reflexivity. From the supposition, wR^i_bv. Since R^i_b is serial, there is some u such that vR^i_bu. By transitivity, wR^i_bu. By Euclidean, uR^i_bv. Since R^i_k is reflexive, vR^i_kv. So, through u, $v(R^i_k \circ R^i_b)v$, and therefore vR^i_bv.

R^i_k- Transitivity. With wR^i_bv, let $x, y \in dc(w)$ be such that vR^i_kx and xR^i_ky. From the definition of $dc(w)$, wR^i_by. The Euclidean property yields vR^i_by, and since $R^i_b \subseteq R^i_k$, it follows that vR^i_ky.

R^i_k-Euclidean. We must show vR^i_kx and vR^i_ky implies xR^i_ky. Assume $v, x, y \in dc(w)$, vR^i_kx, and vR^i_ky. By the definition of $dc(w)$, wR^i_bv and wR^i_by, so vR^i_by by the Euclidean property of R^i_b, and by $R^i_b \subseteq R^i_k$, it follows that vR^i_ky. □

Since R^i_k and R^i_b are part of the same bimodal equivalence relation under $dc(w)$, for all $w \in W$, the following theorems follow in a straightforward manner, establishing notions of weak and strong indistinguishability, which we call *BK Indistinguishability*, highlighting the fact that behind the belief operator's doxastic relation, belief and knowledge are indistinguishable from each other, and in a mutual equivalence relation with each other.

Theorem 3.3.3 (Weak BK Indistinguishability)**.** *For all $w, v, w' \in W$, $w(R^i_k \circ R^i_b)v$ and wR^i_bw' implies $R^i_b(w') = R^i_k(w') = R^i_k(v) = R^i_b(v)$.*

Proof. Suppose $w(R^i_k \circ R^i_b)v$ and wR^i_bw' for some arbitrary $w, w', v \in W$. It suffices to show that $w', v \in dc(w)$. Since wR^i_bw', $w' \in dc(w)$. Since $w(R^i_k \circ R^i_b)v$, the composed relation being a subset of R^i_b, it follows that wR^i_bv and therefore $v \in dc(w)$. □

Theorem 3.3.3 does not guarantee that the actual world w is in this bimodal composed equivalence relation, because it may not be the case that $wR_b^i w$. This fails when i has false beliefs, which is frequently the case with humans. However, the following is still the case: any branch departing w via the R_b^i relation will be a bimodal composed equivalence class of epistemic and doxastic possibility, and thus indistinguishable from each other. Any path that departs from w solely by the R_k^i relation will not partake in the equivalence class. The way to interpret this is as follows. For some reason, the epistemically possible worlds outside the bimodal composed equivalence class are not registering with her cognizance. It could be that she is overlooking some information. It could be that she is motivated in her skepticism for some psychological reason. It could be that she is under the cloud of some cognitive bias. Regardless, when φ is true in the bimodal composed equivalence class but false in some epistemically possible world (*e.g.* the actual world), her belief that φ overpowers the lone epistemic relation, and she instead believes that she knows φ.

If the actual world is in its own doxastic relation, we have the following.

Theorem 3.3.4 (Strong BK Indistinguishability)**.** *For all $w, v \in W$, if $wR_b^i w$ then for all $v \in W$, $wR_b^i v$ implies $R_b^i(w) = R_k^i(w) = R_k^i(v) = R_b^i(v)$.*

Proof. Suppose for an arbitrary $w, v \in W$ that $wR_b^i w$ and $wR_b^i v$. By definition, both are in $dc(w)$, which establishes the bimodal equivalence relation. □

So, when i has no false beliefs, her belief and knowledge collapse into an $\mathcal{S}5$ modal operator. When this condition holds, she makes decisions on par with those of an ideally rational agent who does not make mistakes. However, she is not guaranteed to remain in this state, as the model (world) might change in such a way that the doxastic relation comes to exclude the actual world, in which case i will have false beliefs again.

Thus, in the static epistemic-doxastic logic of $\mathcal{DASL}$, the intuitively appealing property of indistinguishability is maintained from a subjective, internal standpoint, *i.e.,* from i's own perspective. This should be all that is needed, because the motivating intuition behind the indistinguishability requirement is that an agent should not be able to tell which of her possible worlds she is in, and that is the case here with respect to her subjective evidence.

3.4 Avoiding The Collapse

It is important that we maintain the distiction between knowledge and belief if we aim to model humans. To show that our logic maintains this distinction, we present the following counterexample.

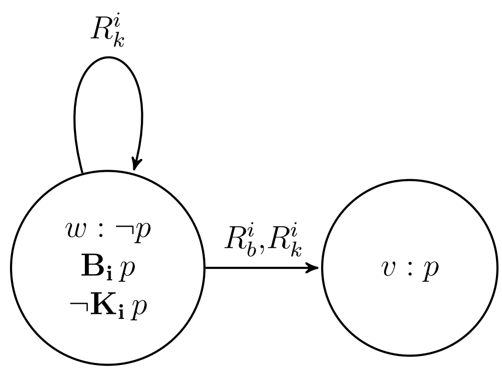


Figure 3.4: A counterexample to $\mathbf{B_i}\,\varphi \Rightarrow \mathbf{K_i}\,\varphi$.

The counterexample is based on the key difference between belief and knowledge, namely that knowledge must be true, while belief does not need to be true. Here, at w, the proposition p is not true. However, because i only considers p possible, and neglects the possibility that $\neg p$, she believes i at w. Since p is not true at w, and knowledge entails truth (via the reflexivity frame condition), it is not true at w that i knows p. Therefore, at w, i believes p but she does not know it.

This suffices to show that the two modalities do not collapse. We have thus far defended the static base in terms of indistinguishibility and collapse avoidance. We turn now to soundness and completeness.

3.5 Soundness and Completeness

This section establishes the soundness and completeness of the logic. We present mechanized formalizations of each proof in the Coq Proof Assistant. The proof of soundness shows that axioms of $\mathcal{DASL}$ are each valid for all $\mathcal{DASL}$ frames. The proof of completeness shows that each axiom schema is a Sahlqvist formula, and therefore forms a complete logic built by adding them to the axiom K of the minimal modal logic, by Sahlqvist's Theorem [69].

Both proofs make use of a special property of modal logics. Every modal logic formula is equivalent to a first- or second-order logic formula, in the sense that it is valid on the same frames as the *corresponding* first- or second order formula. The frame conditions of $\mathcal{DASL}$ are first order logic formulas, and they each correspond to an axiom schema of the modal logic. This correspondence suffices for soundness. When the modal formulas correspond to first order formulas, and the formulas have a certain special structure, the resulting logic is guaranteed to be complete with respect to the first order frame relations.

We begin this section with the soundness proof, and follow up with the completeness proof.

3.5.1 Soundness

A logic has the property of soundness *if and only if* all of its derivable theorems are valid in its model theory. $\mathcal{DASL}$'s model theory is the class of relational frames with two binary relations for each agent i, R_k^i and R_b^i, defined over possible worlds. The R_k^i relations are reflexive, and the R_b^i relations are serial, euclidean, and supersets of R_k^i, and of $(R_k^i \circ R_b^i)$. Any formula satisfied by all models of this class of frames, no matter the valuation assignment of propositional variables, is a valid formula. Thus, to show that $\mathcal{DASL}$is sound is to show that all of its theorems are satisfied by all such models.

Formally, this amounts to:

Theorem 3.5.1 (Soundness). *Dynamic Agent Safety Logic is sound for Kripke structures with*

(1) reflexive R_k^i relations,

(2) serial, Euclidean R_b^i relations,

(3) which are partially ordered $(R_k^i \circ R_b^i) \subseteq R_b^i$, and $R_b^i \subseteq R_k^i$.

In order to formalize this in Coq for mechanical proof checking, we must define the notions of a frame, model, proposition, and theorem. We follow the Master's thesis of Paulien De Wind [12] for the model theory mechanization and the works of Lescanne and Puisségur [59, 65] and Maliković and Čubrilo [61, 62] in setting up the proof theory.

A frame is a record in Coq that consists of a set of possible worlds and relations on agent-world-world tuples to capture the fact that each binary relation is parameterized by each agent in the model.

```
Record frame : Type := {
  W : Set;
  Rk : DASL.Agents → W → W → Prop;
  Rb : DASL.Agents → W → W → Prop
}.
```

A model is a frame combined with a valuation function assigning worlds to atoms. We likewise create a type of agents here, imported from a library $\mathcal{DASL}$, defining them as,

```
Inductive Agents: Type := Pilot | CoPilot | AutoPilot.
```

but any target domain's agents will suffice here.

```
Record model : Type := {
  F : frame;
  Val : (W F) → Atoms → Prop;
  Agents: DASL.Agents
}.
```

We say a proposition is an `Inductive` type of atoms, implications, negations, falsum, and the doxastic and epistemic modal operators.

```
Inductive prop : Type :=
  | atm : Atoms → prop
  | imp : prop → prop → prop
  | negp : prop → prop
  | falsum : prop
  | K : Agents → prop → prop
  | B : Agents → prop → prop.
```

The type of `Atoms` is domain specific for the ground truth facts on which propositions in the system are built. For $\mathcal{DASL}$these are instrument readings in the cockpit, but for other domains they will be different.

A theorem type is defined in Coq so as to include all propositional tautologies, the modal axioms at the static base of $\mathcal{DASL}$, and closure through Modus Ponens and Necessitation.

```
Inductive theorem : prop → Prop :=
  | Hilbert_K: forall p q : prop, theorem (p ⟹ q ⟹ p)
  | Hilbert_S: forall p q r : prop,
     theorem ((p⟹ q⟹ r)⟹ (p⟹ q)⟹ (p⟹ r))
  | Classic_NOTNOT : forall p : prop, theorem ((NOT (NOT p)) ⟹ p)
  | MP : forall p q : prop, theorem (p ⟹ q) → theorem p → theorem q
  | K_Nec : forall (a : DASL.Agents) (p : prop),
     theorem p → theorem (K a p)
  | K_K : forall (a : DASL.Agents) (p q : prop),
     theorem (K a p ⟹ K a (p ⟹ q) ⟹ K a q)
  | K_T : forall (a : DASL.Agents) (p : prop), theorem (K a p ⟹ p)
  | B_K : forall (a : DASL.Agents) (p q : prop),
      theorem (B a p ⟹ B a (p ⟹ q) ⟹ B a q)
  |B_Serial : forall (a : DASL.Agents) (p : prop),
     theorem (B a p ⟹ NOT (B a (NOT p)))
  | B_5 : forall (a : DASL.Agents) (p : prop),
     theorem (NOT (B a p) ⟹ B a (NOT (B a p)))
  | K_B : forall (a : DASL.Agents) (p : prop), theorem (K a p ⟹ B a p)
  | B_BK : forall (a : DASL.Agents) (p : prop),
     theorem (B a p ⟹ B a (K a p)).
```

We denote the `theorem` type judgment with $|--$. Our mechanical proof strategy is to follow closely what a pen-and-paper proof of soundness consists in. We will define the frame conditions of $\mathcal{DASL}$'s relations, and we will prove that each is sufficient for deriving that its corresponding modal formula is valid for frames with that condition. An example will illustrate this.

```
Definition reflexive_Rk_frame (F : frame) : Prop :=
  forall (w : (W F)) (ags : DASL.Agents), (Rk F ags w w).
```

First we define the property of epistemic reflexivity on a frame. We then prove a lemma showing it is sufficient for the Truth Axiom's being valid on reflexive frames.

```
Lemma K_is_refl : forall (phi : prop) (F : frame) (a : DASL.Agents),
  (reflexive_Rk_frame F) →
  F ||= ((K a phi) ⟹ phi).
Proof.
  intros.
    unfold reflexive_Rk_frame in H.
      unfold Frame_validity.
      intros.
        unfold Model_satisfies.
        intros. pose proof H w; clear H. pose proof H0 a; clear H0.
          unfold satisfies.
          intros. pose proof H0 w; clear H0.
          simpl in H1. pose proof H1 H; clear H1.
  auto.
Qed.
```

We add this lemma to our `hints` so that Coq's engine will automatically try to use it on calls to `auto`.

```
Hint Resolve K_is_refl.
```

We define a $\mathcal{DASL}$ frame as a simple conjunction over the above properties.

```
Definition DASL_Frame (F : frame) : Prop :=
  reflexive_Rk_frame F ∧
  serial_Rb_frame F ∧
  euclidean_Rb_frame F ∧
  Rb_subset_Rk F ∧
  Rb_subset_Rb_compose_Rk F.
```

Through similar definitions and lemmas of the remaining frame conditions and axiom schemas, the soundness theorem is stated and proven as follows.

```
Theorem DASL_Soundness : forall (phi : prop) (F : frame) (a : DASL.Agents),
DASL_Frame F →
  |-- phi →
  F ||= phi.
Proof.
  intros phi F.
  unfold DASL_Frame.
  intros. destruct H; destruct H1; destruct H2; destruct H3.
  induction H0; eauto.
Qed.
```

The proof proceeds by instantiating `phi`, `F`, `a`, unfolding the definition of `DASL_Frame` into its constituent frame conditions, and running and induction over $|--$`phi`. The call to `eauto` suffices due to each of the helper lemmas added to the `hints`. We turn now to mechanizing completeness.

3.5.2 Completeness

For completeness, our proof depends on the notion of a Sahlqvist formula, which itself depends on syntactical properties of modal formulas, which we must also define. The completeness proof is only *mostly* mechanized. The overall theme of the proof is to rely on the *Sahlqvist Completeness Theorem* (SCT) of modal logic.

Theorem 3.5.2 (Sahlqvist Completeness Theorem[15][4]). *Every Sahlqvist formula is canonical for the first-order property it defines. Hence, given a set of Sahlqvist axioms Σ, the logic $\mathbf{K}\Sigma$ is complete with respect to the class of frames $\mathtt{F}_\Sigma$ (that is, the first-order class of frames defined by Σ.)*

This states that every modal formula with a syntactic structure, defined by Sahlqvist, is complete for the class of frames defined by its corresponding first order formula.

A modal formula φ and a first order formula χ *correspond* when they define the same class of frames, C, meaning a frame F belongs to C defined by φ and χ if and only if φ is valid on all F-frames and χ holds in the metalanguage of the semantics for all F-frames.

An example might help. Consider the class of reflexive frames. It is defined by the first order property of *reflexivity*, with the first order formula $\forall x,\ Rxx$. A frame for this class consists of a relational structure with a set of worlds W and an accessibility relation R. We see that the reflexivity formula holds in the metalanguage when x is drawn from W, and R is such that each world accesses itself, but the proof of this requires translation algorithms due to Sahlqvist and van Benthem[15]. Similarly, we know (from the soundness proof) that the modal formula $\Box\varphi \Rightarrow \varphi$ is true for all frames with such an accessibility condition. The formulas $\Box\varphi \Rightarrow \varphi$ and $\forall x,\ Rxx$ correspond to each other.

The Sahlqvist Completeness Theorem states that if $\Box\varphi \Rightarrow \varphi$ has certain syntactic properties to be defined shortly, then the basic modal logic $\mathbf{K}$ plus the axiom

[4]Theorem 4.42 of Chapter 4 in Blackburn *et. al.*.

scheme $\Box\varphi \Rightarrow \varphi$ is complete with respect to the class of reflexive frames, due to their correspondence.

The SCT is so powerful that it suffices to prove that each of our axiom schemas of $\mathcal{DASL}$'s static base is a Sahlqvist formula, and that the associated frame conditions used in the proof of soundness are their first order correspondents. We have mechanized the proof that establishes that the axiom schemas of $\mathcal{DASL}$ are such *Sahlqvist formulas*. The proof of correspondence between the axiom schemes and their first order counterparts is partially mechanized during the proof of soundness. We prove the remaining correspondence direction by hand below. We begin this section by defining a Sahlqvist formula's syntatic characteristics, and provide our mechanization of them in Coq.

A *Sahlqvist implication* is a formula with a *Sahlqvist antecedent* and a *positive consequent*.

A formula is *positive* if all of its propositional letters are in the scope of an even number of negation signs. A formula is *negative* if one or more of its propositional letters are in the scope of an odd number of negation signs.

A *Sahlqvist antecedent* is a formula built from *a*) propositional letters prefixed by $n \geq 0$ $\Box$'s and *b*) negative formulas, *c*) using only $\vee$, $\wedge$, and $\Diamond$ to build the antecedent from (*a*) and (*b*) components.

To define these notions and reason about axiom schemas rather than actual well-formed formulas of the object language, we define an inductive type in Coq that more closely resembles the formula schemas one typically uses when discussing a logic, as in $\mathbf{K_i}\,\varphi \Rightarrow \varphi$, where φ is a metalanguage variable standing in for any well-formed formula of the language. Because our inductive Coq type for well-formed formulas of the language is `prop`, we must create an inductive type that takes `prop` as basic.

```
Inductive schema : Type :=
  | SProp : prop → schema
  | SAnd : schema → schema → schema
  | SOr : schema → schema → schema
  | SImp : schema → schema → schema
  | SNeg : schema → schema
  | SK : DASL.Agents → schema → schema
  | SB : DASL.Agents → schema → schema.
```

We use the following notation symbols.

```
Notation "\ p" := (SNeg p) (at level 70, right associativity ).
Infix "=s⇒ " := SImp (right associativity, at level 85).
Infix "| s|" := SOr (right associativity , at level 75).
Infix "&s&" := SAnd (right associativity, at level 75).
```

We define a `negative` formula as one whose proposition letters are all under the scope of an odd number of negation signs.

```
Fixpoint negative_formula (phi : schema) : Prop :=
  match phi with
    | SProp p => False
    | SAnd phi1 phi2 => (negative_formula phi1)
      /\ (negative_formula phi2)
    | SOr phi1 phi2 => (negative_formula phi1)
      /\ (negative_formula phi2)
    | SImp phi1 phi2 => (negative_formula phi1)
      /\ (negative_formula phi2)
    | SNeg phi' => not (negative_formula phi')
    | SK a phi' => negative_formula phi'
    | SB a phi' => negative_formula phi'
  end.
```

We define a `positive` formula appropriately as one whose proposition letters are all under the scope of an even number of negation signs..

```
Fixpoint positive_formula (phi : schema) : Prop :=
  match phi with
    | SProp p => True
    | SAnd phi1 phi2 => (positive_formula phi1)
       /\ (positive_formula phi2)
    | SOr phi1 phi2 => (positive_formula phi1)
        /\ (positive_formula phi2)
    | SImp phi1 phi2 => (positive_formula phi1)
        /\ (positive_formula phi2)
    | SNeg phi' => not (positive_formula phi')
    | SK a phi' => positive_formula phi'
    | SB a phi' => positive_formula phi'
  end.
```

To construct Sahlqvist antecedents, we define the notions of a `boxed` formula and a `s_a_component` (for Sahlqvist antecedent component).

```
Fixpoint boxed_formula (phi : schema) : Prop :=
  match phi with
    | SProp p => True
    | SAnd phi1 phi2 => False
    | SOr phi1 phi2 => False
    | SImp phi1 phi2 => False
    | SNeg phi' => False
    | SK a phi' => boxed_formula phi'
    | SB a phi' => boxed_formula phi'
  end.
```

This states that a `boxed_formula` can be a formula variable on its own, or any number of modal boxes prefixing a formula variable, but any other structure imposed on the formula schema means it is not boxed.

A `s_a_component` is a formula schema built up of `boxed_formula`'s and `negative` formulas connected by conjunction, disjunction, and $\langle \mathbf{K}_i \rangle$, $\langle \mathbf{B}_i \rangle$.

```
Fixpoint s_a_component (phi : schema) : Prop :=
  match phi with
    | SProp p ⇒ True
    | SAnd phi1 phi2 ⇒ (s_a_component phi1) ∧ (s_a_component phi2)
    | SOr phi1 phi2 ⇒ (s_a_component phi1) ∧ (s_a_component phi2)
    | SImp phi1 phi2 ⇒ not (s_a_component phi1) ∧ (s_a_component phi2)
    | SNeg phi' ⇒ match phi' with
      | SProp p ⇒ True
      | SAnd p1 p2 ⇒ positive_formula p1 ∧ positive_formula p2
      | SOr p1 p2 ⇒ positive_formula p1 ∧ positive_formula p2
      | SImp p1 p2 ⇒ negative_formula p1 ∧ positive_formula p2
      | SNeg p' ⇒ s_a_component p'
      | SK a p' ⇒ not (s_a_component p')
      | SB a p' ⇒ not (s_a_component p')
    end
    | SK a phi' ⇒ boxed_formula phi'
    | SB a phi' ⇒ boxed_formula phi'
  end.
```

Next, we define what it is for a subformula to be a Sahlqvist antecedent.

```
Fixpoint sahlqvist_antecedent (phi : schema) : Prop :=
  s_a_component phi.
```

We combine the above components to define what it is for a formula to be a Sahlqvist implication.

```
Definition sahlqvist_implication (phi psi : schema) : Prop :=
  sahlqvist_antecedent (phi) ∧ positive_formula (psi).
```

According to Blackburn *et. al.*[15], a Sahlqvist formula is built up from "Sahlqvist implications by freely applying boxes and conjunctions, and by applying disjunctions only between formulas that do not share any proposition letters." So we must define a function that parses implications to identify when a proposition letter appears in both antecedent and consequent. We do that with the following in Coq.

```
Fixpoint share_prop_letter (phi psi : schema) {struct phi} : Prop :=
  match phi with
  | SProp phi' ⇒ match psi with
    | SProp psi' ⇒ phi' = psi'
    | SAnd psi1 psi2 ⇒ (prop_in_schema phi' psi1)
      ∨ (prop_in_schema phi' psi2)
    | SOr psi1 psi2 ⇒ (prop_in_schema phi' psi1)
      ∨ (prop_in_schema phi' psi2)
    | SImp psi1 psi2 ⇒ (prop_in_schema phi' psi1)
      ∨ (prop_in_schema phi' psi2)
    | SNeg psi' ⇒ (prop_in_schema phi' psi')
    | SK a psi' ⇒ (prop_in_schema phi' psi')
    | SB a psi' ⇒ (prop_in_schema phi' psi')
    end
  | SAnd phi1 phi2 ⇒ (share_prop_letter phi1 psi)
    ∨ (share_prop_letter phi2 psi)
  | SOr phi1 phi2 ⇒ (share_prop_letter phi1 psi)
    ∨ (share_prop_letter phi2 psi)
  | SImp phi1 phi2 ⇒ (share_prop_letter phi1 psi)
    ∨ (share_prop_letter phi2 psi)
  | SNeg phi' ⇒ (share_prop_letter phi' psi)
  | SK a phi' ⇒ (share_prop_letter phi' psi)
  | SB a phi' ⇒ (share_prop_letter phi' psi)
  end.
```

With these components, we define a Sahlqvist formula as follows.

```
Fixpoint sahlqvist_formula (phi : schema) : Prop :=
  match phi with
    | SProp phi'⇒ True
    | SAnd phi1 phi2 ⇒ (sahlqvist_formula phi1)
       ∧ (sahlqvist_formula phi2)
    | SOr phi1 phi2 ⇒ not (share_prop_letter phi1 phi2)
       ∧ (sahlqvist_formula phi1) ∧ (sahlqvist_formula phi2)
    | SImp phi1 phi2 ⇒ (sahlqvist_implication phi1 phi2)
    | SNeg phi' ⇒ (positive_formula phi')
    | SK a phi' ⇒ sahlqvist_formula phi'
    | SB a phi' ⇒ sahlqvist_formula phi'
  end.
```

For completeness, it suffices to show that each static $\mathcal{DASL}$ axiom schema is a Sahlqvist formula. Because in our soundess proof we mechanically proved that the $\mathcal{DASL}$ frame conditions correspond to the axiom schemas of $\mathcal{DASL}$, and because this correspondence consists of unique pairs per the theorem, it follows that those frame conditions are the ones for which the axioms schemas are complete.

We can represent a theory of modal logic as a list of axiom schemas, and represent $\mathcal{DASL}$ in particular as the following list.

```
Definition DASL_Axioms (p q r : prop) (a : DASL.Agents) :=
(SK a (SProp p) =s⇒ SK a ((SProp p) =s⇒ (SProp q)) =s⇒ SK a (SProp q))
:: (SK a (SProp p) =s⇒ (SProp p))
:: (SB a (SProp p) =s⇒ SB a ((SProp p) =s⇒ (SProp q)) =s⇒ SB a (SProp q))
:: (SB a (SProp p) =s⇒ \ (SB a (\ (SProp p))))
:: (\ (SB a (\ (SB a (SProp p)))) =s⇒ SB a (SProp p))
:: (SK a (SProp p) =s⇒ SB a (SProp p))
:: (SB a (SProp p) =s⇒ SB a (SK a (SProp p)))
:: nil.
```

We define a function over lists of axiom schemas to determine whether each schema is a Sahlqvist implication.

```
Fixpoint Complete_via_Sahlqvist (l : list formula) : Prop :=
match l with
| nil ⇒ True
| (l' :: els) ⇒ sahlqvist_formula (l') ∧
Complete_via_Sahlqvist (els)
end.
```

Then we prove that $\mathcal{DASL}$ is complete by calling `Complete_via_Sahlqvist` on its representative list of axiom schemas.

```
Theorem DASL_Axioms_Complete : forall (p q r : prop) (a : DASL.Agents),
    Complete_via_Sahlqvist (DASL_Axioms p q r a).
```

Of course, this theorem is not in the expected form of $\models \varphi$ implies $\vdash \varphi$, and in fact we have only mechanized the proof that $\mathcal{DASL}$ is complete with respect to *some* set of frame relations. To show that $\mathcal{DASL}$ is complete with respect to the very same relations as those for which we proved soundness, we refer back to the definition of *correspondence*.

In our proof of soundness, we defined a $\mathcal{DASL}$ frame as one consisting of reflexivity of $\mathbf{K_i}$, seriality, transitivity, and euclidicity of $\mathbf{B_i}$, partial ordering of $\mathbf{K_i}\,\varphi \Rightarrow \mathbf{B_i}\,\varphi$ (Knowledge implies Belief), and $\mathbf{B_i}\,\varphi \Rightarrow \mathbf{B_i}\,\mathbf{K_i}\,\varphi$ (Evidential Restraint). The soundness proof appealed to lemmas, which we mechanically proved, showing that each frame condition makes one of the axiom schemas valid for that class of frames. Thus, the soundness proof establishes that the first order frame conditions and their associated modal axioms establish one direction of the correspondence proof. What remains is to show that each axiom scheme in $\mathcal{DASL}$ implies its associated frame condition. This, combined with the soundness proof's condition to axiom scheme derivation, establishes their model equivalence. Therefore they define the same class of frames, and hence they correspond.

Our proof of this direction applies van Benthem's standard translation algorithm, as presented by Blackburn *et. al.*in [15] in order to yield a second order translation of each $\mathcal{DASL}$ axiom schema, and then applies the Sahlqvist-van Benthem algorithm, presented also by [15], to derive the corresponding first order formulas. The transformation takes us from a modal formula through monadic second order logic, and applies lambda calculus to transform second order formulas into first order formulas. We do not mechanize all of this machinery, but do provide the important details here.

The Standard Translation algorithm consists of the following recursive rewrite rules.

Definition 3.5.2.1 (Standard Translation)**.** . *For binary modal operators* $\mathbf{B_i}$ *and*

$\mathbf{K_i}$, *we define the function ST on a given variable x and proposition letter p:*

$$\begin{aligned} ST(x,p) &= P(x) \\ ST(x,\bot) &= (x \neq x) \\ ST(x,\neg\varphi) &= \sim (ST(x,\varphi)) \\ ST(x,\varphi \vee \psi) &= ST(x,\varphi) + ST(x,\psi) \\ ST(x,\mathbf{K_i}\,\varphi) &= \forall y,\ R^i_k(x,y) \Longrightarrow ST(y,\varphi) \end{aligned}$$

The resulting predicate and objects corresponding to modal propositions and variables are arbitrary, and so there are infinitely many standard translations from a modal formula to a first order formula.[5] By the van Benthem Characterization Theorem, each such translation is equivalent to the other. An intuitive interpretation is that the translated formula in classical logic represents the satisfaction predicate for the proposition semantics of the modal logic, with the *current* world w assigned to the free variable x. However, because we quantify over predicates, the resulting formula is second order, and we can instantiate the predicates to minimal valuations. We restrict our quantification over monadic predicates, so that we are limited to monadic second order logic[6].

When modal formulas are uniform in their proposition letters (each letter is only positive or only negative), one can universally quantify over translated predicates and instantiate them in the following way, for every P in $ST(x,\varphi)$:

$$P \rightsquigarrow \begin{cases} \lambda u.\bot, & \text{if } ST(x,\varphi) \text{ positive in } P \\ \lambda u.\top, & \text{if } ST(x,\varphi) \text{ negative in } P \end{cases}$$

[5]See chapter 2, example 2.46 in [15].

[6]monadic second order logic is restricted to predicates over sets.

This produces the smallest valuation of positive propositions and the largest valuation of negative propositions, which by monotonicity of uniform formulas, allow for upward and downward truth preservation respectively. Thus, from a modal formula, a first order formula is derived.

Sahlqvist formulas are not necessarily uniform in their propositions, but a similar technique is used to generalize translated predicates into a monadic second order formula, and instantiate them to minimal valuations. The monadic second order standard translation is fed into the Sahlqvist-van Benthem algorithm for producing a corresponding frame condition.

The Sahlqvist-van Benthem algorithm is described in the following way.

Step 1. Apply the following equivalences, from left to right, to the antecedent (in this order):

$$(\exists x, \Phi(x) \cdot \psi) \Longleftrightarrow \exists x, (\Phi(x) \cdot \psi) \tag{3.1}$$

$$(\exists x, \Phi(x) \Longrightarrow \psi) \Longleftrightarrow \forall x, (\Phi(x) \Longrightarrow \psi), \tag{3.2}$$

where $\Phi(x)$ is a sentence with one free variable x.

Step 2. Construct minimal valuations. For each proposition letter instance in the formula's antecedent, select a minimal valuation that makes the antecedent true, then combine them as a lambda function over a disjunction of them. Proposition letter instances are converted to predicates over variables. The resulting formula will quantify over these variables, so they must be distinct from the free variable in the standard translation. Our intuitive interpretation is that the variables are worlds from the frame. An appropriate lambda function can be created for each predicate such that the smallest frame possible validates it. For a proposition letter p appearing once in a formula, either as an atom or behind a diamond (*e.g.* $\langle \mathbf{B}_i \rangle\, p$), $\lambda u.u = x_1$ is created, assuming x_1 is the bound variable appearing in second order translation

$\forall P, \forall x_1, (R_b^i(x, x_1) \cdot P(x_1))$.

If p appears twice in the antecedent, either as an atom or behind a diamond, each instance appears in the translated sentence in some form like $\exists x_1 (R_b^i(w, x_1) \cdot P(x_1))$ (for diamonds) ... $\cdot P(x_2)$. The lambda function to serve as P's instance is $\lambda u.(u = x_1 + u = x_2)$. For more than two appearances as atoms or behind diamonds, this pattern is repeated.

For proposition letters behind boxes (*e.g.* $\mathbf{B_i}\, p$), lambda functions of the form $\lambda u.(\bigvee_{j=0}^{n}(R_b^i(x_j, u)))$ are used, for the number of proposition letters n in the formula. When a proposition letter appears behind boxes and atomically, the atomic instances use the $=$ relation in the lambda function.

Step 3. Perform the instantiations. This step substitutes the predicate variables for the minimal valuations constructed in *Step 2*. For example, if $P(x_1)$ appears in the second order formula's antecedent as a singular instance of P, with the minimal valuation substitution $\lambda u.u = x_1$, the instantiated predicate instance yields the sentence $(\lambda u.u = x_1)(x_1)$, which beta reduces to $x + 1 = x_1$. If the proposition letter instance appears behind a box in the modal formula, *e.g.* $\mathbf{B_i}\, p$, the second order formula is $\forall y(R_b^i(x_1, y) \implies P(y))$ with the lambda function $\lambda u.(R_b^i(x_1, u))$, which yields the formula $\forall y(R_b^i(x_1, y) \implies \lambda u.(R_b^i(y, x_1)))(y)$. This beta reduces to $\forall y(R_b^i(x_1, y) \implies R_b^i(x_1, y))$, which is equivalent to $\forall y, R_b^i(x_1, y)$. For examples, see Blackburn *et. al.*[15] chapter 3.

The result is a first order logic frame condition corresponding to the axiom schema. We apply this algorithm to each $\mathcal{DASL}$ axiom schema and show that it is equivalent to the frame condition used to prove soundness.

Lemma 3.5.3 (Schema to First Order Frame Condition Correspondence)**.** *For all frames* $\mathtt{F}$*, if each* $\mathcal{DASL}$ *axiom schema and its associated first order frame condition hold for* $\mathtt{F}$*, then* $\mathtt{F}$ *is a* $\mathcal{DASL}$ *frame,*

where *hold* refers to frame validity and metalanguage semantics, respectively.

Proof. We must show a translation from the following axiom schemas to the appropriate first order frame conditions:

1. $\mathbf{K_i}\,\varphi \Rightarrow \varphi$
2. $\mathbf{B_i}\,\varphi \Rightarrow \langle \mathbf{B}_i \rangle\,\varphi$
3. $\mathbf{K_i}\,\varphi \Rightarrow \mathbf{B_i}\,\varphi$
4. $\mathbf{B_i}\,\varphi \Rightarrow \mathbf{B_i}\,\mathbf{K_i}\,\varphi$

$\mathbf{K_i}\,p \Rightarrow p$. Applying the Standard Translation rules yields $\forall P(\forall y,\ R_k^i(x,y) \Longrightarrow P(y)) \Longrightarrow P(x)$. We have a single predicate with one instance in the antecedent behind a box. This yields the lambda function $\lambda u.(R_k^i(x,u))$. We substitute this instance of P into the formula to yield:
$\forall y, (R_k^i(x,y) \Longrightarrow \lambda u.(R_k^i(x,u))(y)) \Longrightarrow \lambda u.(R_k^i(x,u))(x)$, which beta reduces and simplifies to:
$\forall y, R_k^i(x,y) \Longrightarrow R_k^i(x,x)$, which is equivalent to $R_k^i(x,x)$[7].

$\mathbf{B_i}\,\varphi \Rightarrow \mathbf{B_i}\,\mathbf{K_i}\,\varphi$. We skip ahead to this more interesting example.

$$\rightsquigarrow \forall P, (\forall y, R_b^i(x,y) \Longrightarrow P(y)) \Longrightarrow (\forall z, (R_k^i \circ R_b^k)(x,z) \Longrightarrow P(z))$$
$$\rightsquigarrow \forall P, \forall y, z, (R_k^i \circ R_b^k)(x,z) \cdot (R_b^i(x,y) \Longrightarrow P(y)) \Longrightarrow P(z)$$
$$\rightsquigarrow \forall y, z, (R_k^i \circ R_b^k)(x,z) \cdot (R_b^i(x,y) \Longrightarrow \lambda u.(R_b^i(x,u))(y)) \Longrightarrow \lambda u.(R_b^i(x,u)(z)$$
$$\rightsquigarrow \forall y, z, (R_k^i \circ R_b^k)(x,z) \cdot R_b^i(x,y) \Longrightarrow R_b^i(x,z)$$

$\forall z, (R_k^i \circ R_b^k)(x,z)$ is equivalent to $\forall z, z', R_b^i(x,z') \cdot R_k^i(z',z)$. Thus, the above translation simplifies to $\forall z, (R_k^i \circ R_b^k)(x,z) \Longrightarrow R_b^i(x,z)$, which is the subset relation of the frame condition.[8]

[7]Proof: Assume $R_k^i(x,x)$. Then by monotonicity, $\forall y, (R_k^i(x,y) \Longrightarrow R_k^i(x,x))$. For the other direction, by contraposition, assume $\sim R_k^i(x,x)$. Then $\exists y, \sim R_k^i(x,y)$, which is equivalent to $\sim \forall y, R_k^i(x,y)$

[8]Because $\forall y, z, z', R_b^i(x,y) \cdot R_b(x,z') \cdot R_k(z',z)$ can rewrite z' with y and simplify to $R_b^i(x,y) \cdot R_k^i(y,z)$.

$\mathbf{B_i}\,\varphi \Rightarrow \langle \mathbf{B}_i \rangle\, \varphi$.

$$\rightsquigarrow \forall P, y, (R_b^i(x,y) \Longrightarrow P(y)) \Longrightarrow \exists z, (R_b^i(x,z) \cdot P(z))$$

$$\rightsquigarrow \forall y, (R_b^i(x,y) \Longrightarrow \lambda u.(R_b^i(x,u))(y)) \Longrightarrow \exists z, (R_b^i(x,z) \cdot \lambda u.(R_b^i(x,u))(z))$$

$$\rightsquigarrow \forall y, (R_b^i(x,y)) \Longrightarrow \exists z, (R_b^i(x,z))$$

which is equivalent to frame condition for seriality.[9]

$\mathbf{K_i}\,\varphi \Rightarrow \mathbf{B_i}\,\varphi$.

$$\rightsquigarrow \forall P, y, (R_k^i(x,y) \Longrightarrow P(y)) \Longrightarrow \forall z, (R_b^i(x,z) \Longrightarrow P(z))$$

$$\rightsquigarrow \forall y, (R_k^i(x,y) \Longrightarrow \lambda u.(R_k^i(x,u))(y)) \Longrightarrow \forall z, (R_b^i(x,z) \Longrightarrow \lambda u.(R_b^i(x,u))(z))$$

$$\rightsquigarrow \forall y, (R_k^i(x,y)) \Longrightarrow \forall z, (R_b^i(x,z) \Longrightarrow R_k^i(x,z))$$

$$\equiv \forall y, z, (R_k^i(x,y) \Longrightarrow R_b^i(x,z) \Longrightarrow R_k^i(x,z))$$

$$\equiv \forall y, z, (R_k^i(x,y) \cdot R_b^i(x,z) \Longrightarrow R_k^i(x,z))$$

$$\equiv \forall z, (R_b^i(x,z) \Longrightarrow R_k^i(x,z))^{10}$$

which is the frame condition for Knowledge implies Belief. □

Lemma 3.5.3, in conjunction with the proof of soundness, prove that the modal axiom schemas of $\mathcal{DASL}$ correspond to the first order frame conditions of $\mathcal{DASL}$ frames. By the SCT, this suffices to show that $\mathcal{DASL}$ is complete for $\mathcal{DASL}$ frames as we have defined them.

Theorem 3.5.4 ($\mathcal{DASL}$ complete on $\mathcal{DASL}$ frames)**.** *If* $\mathtt{F}$ *is a* $\mathcal{DASL}$ *frame, then* $\mathtt{F} \models \varphi$ *implies* $\vdash \varphi$ *in* $\mathcal{DASL}$*.*

Proof. By the SCT, $\mathcal{DASL}$ is complete with respect to the class of frames its axiom schemas define. By the proof of soundness, $\mathcal{DASL}$ frame conditions each imply

[9]In one direction by monotonicity, in the other by instantiating the consequent and then the antecedent, and then existentially generalizing.

an axiom schema validity, and by the Sahlqvist-van Benthem algorith, each axiom schema translates to the corresponding frame condition. Therefore, the first order frame conditions and axiom schemas correspond. Therefore, the axiom schemas define the $\mathcal{DASL}$ frame conditions. Thus, $\mathcal{DASL}$ is complete with respect to the $\mathcal{DASL}$ class of frames. $\square$

To help validate our mechanization of the `sahlqvist_formula` function in Coq, we run it against some modal formulas whose status as Sahlqvist formulas, in the positive and negative cases, is known.

Lób's formula $\Box(\Box\varphi \Rightarrow \varphi) \Rightarrow \Box\varphi$, the characteristic axiom of provability logic, is not a Sahlqvist formula, and lacks a first order correspondent[10].

To remain in the mechanized axiom schema logic we've produced in Coq, we use the knowledge operator as the $\Box$ operator of the formula, and prove:

```
Example Lob_not_sahlqvist : forall (phi : prop) (a : DASL.Agents),
   not sahlqvist_formula (
      SK a (SK a (SProp phi) =s⇒ (SProp phi)) =s⇒ SK a (SProp phi)
   ).
Proof.
intros. unfold not. not_sahlqvist.
Qed.
```

It makes use of a custom tactic we use in the Coq file.

```
Ltac not_sahlqvist :=
   try (unfold sahlqvist_formula;
        unfold sahlqvist_implication;
        simpl; intuition).
```

We define a similar tactic for verifying that formulas are Sahlqvist formulas.

```
Ltac sahlqvist_reduce :=
simpl; try (unfold sahlqvist_implication; split);
try (unfold positive_formula; simpl; intuition);
try (unfold sahlqvist_antecedent; simpl; intuition; unfold normal_form).
```

We make use of this tactic in, for example,

```
Lemma B_5_is_sahlqvist : forall (phi : prop) (a : DASL.Agents),
sahlqvist_formula (\ (SB a (\ SB a (SProp phi))) =s⇒ (SB a (SProp phi))).
Proof.
  intros; sahlqvist_reduce.
Qed.
```

We prove that each of the modal axioms schemas of $\mathcal{DASL}$ are Sahlqvist formulas using the above techniques, which can be seen in the appendix. Another example Sahlqvist formula that we prove is the Church-Rosser formula, $\Diamond\Box\varphi \Rightarrow \Box\Diamond\varphi$, again using the knowledge operator:

```
Example Church_Rosser_is_sahlqvist : forall (phi : prop) (a : DASL.Agents),
  sahlqvist_formula (
    \ (SK a (\
      (SK a (SProp phi)))) =s⇒
      (SK a (\ (SK a (\ (SProp phi)))))).
Proof.
  intros; sahlqvist_reduce.
Qed.
```

However, if one rearranges the $\Diamond$ and $\Box$ in Church-Rosser, this yields its converse, the McKinsey formula, which is not a Sahlqvist formula: $\Box\Diamond\varphi \Rightarrow \Diamond\Box\varphi$, and we likewise can prove this.

```
Example McKinsey_not_sahlqvist : forall (phi : prop) (a : DASL.Agents),
  not sahlqvist_formula (
    SK a (\ (SK a (\ (SProp phi)))) =s⇒
    \ (SK a (\ (SK a (SProp phi))))).
Proof.
  intros; not_sahlqvist.
Qed.
```

The sharp-eyed reader might notice that we have not defined the `prop` type that appears in the above Coq formulas. Our mechanization of $\mathcal{DASL}$ consists in two levels. The top level is that of the axiom schema, which allows us to prove theorems about the logic using a metavariable for propositions, just as one sees logicians do in papers. The metavariable represents propositions in the object language, which bottoms out in atomic propositions about airplane instruments in this case. With this architecture, one could define alternative object languages in Coq for different domains, perhaps one for driving automobiles, one for power plant operators, and one for submarine operators. Each of these domains involves well-structured relationships between the instrument readings and the actions' safety status, but each concerns different atomic propositions. However, the top level metalanguage remains the same.

Our mechanization of `prop` is discussed in the following chapter, where we present aviation mishap case studies mechanized in the object language.

3.6 Dynamic Extension

The previous section established $\mathcal{DASL}$ as a sound and complete logic, which is a desirable formal property of the static base, since the dynamic extension is reducible to the static base in finitely many steps, it is also sound and complete.

Below are the axioms characterizing the reduction laws from the dynamic logic to a purely static logic through recursive application.

$[\mathbf{A}, a]_i p \Leftrightarrow (post(a, pre(a)) \Rightarrow p)$	Atomic Consequence
$[\mathbf{A}, a]_i \neg\varphi \Leftrightarrow (pre(a) \Rightarrow \neg[\mathbf{A}, a]_i \varphi)$	Action Negation
$[\mathbf{A}, a]_i (\varphi \wedge \psi) \Leftrightarrow ([\mathbf{A}, a]_i \varphi \wedge [\mathbf{A}, a]_i \psi)$	Action Conjunction
$[\mathbf{A}, a]_i \mathbf{K_i}\, \varphi \Leftrightarrow (pre(a) \Rightarrow \bigwedge_{a\chi^i_k\beta} \mathbf{K_i}\, [\mathbf{A}, \beta]_i \varphi)$	Action and Knowledge
$[\mathbf{A}, a]_i \mathbf{B_i}\, \varphi \Leftrightarrow (pre(a) \Rightarrow \bigwedge_{a\chi^i_b\beta} \mathbf{B_i}\, [\mathbf{A}, \beta]_i \varphi)$	Action and Belief
$[\mathbf{A}, a]^{\mathcal{S}}_i p \Leftrightarrow (post(a, pre_s(a)) \Rightarrow p)$	Safe Atomic Consequence
$[\mathbf{A}, a]^{\mathcal{S}}_i \neg\varphi \Leftrightarrow (pre_s(a) \Rightarrow \neg[\mathbf{A}, a]^{\mathcal{S}}_i \varphi)$	Safe Action Negation
$[\mathbf{A}, a]^{\mathcal{S}}_i (\varphi \wedge \psi) \Leftrightarrow ([\mathbf{A}, a]^{\mathcal{S}}_i \varphi \wedge [\mathbf{A}, a]^{\mathcal{S}}_i \psi)$	Safe Action Conjunction
$[\mathbf{A}, a]^{\mathcal{S}}_i \mathbf{K_i}\, \varphi \Leftrightarrow (pre_s(a) \Rightarrow \bigwedge_{a\chi^i_k\beta} \mathbf{K_i}\, [\mathbf{A}, \beta]^{\mathcal{S}}_i \varphi)$	Safe Action and Knowledge
$[\mathbf{A}, a]^{\mathcal{S}}_i \mathbf{B_i}\, \varphi \Leftrightarrow (pre_s(a) \Rightarrow \bigwedge_{a\chi^i_b\beta} \mathbf{B_i}\, [\mathbf{A}, \beta]^{\mathcal{S}}_i \varphi)$	Safe Action and Belief

Table 3.5: The reduction axioms of $\mathcal{DASL}$.

Theorem 3.6.1. *Translating a dynamic formula from the dynamic extension of $\mathcal{DASL}$ via the reduction axiom schemas terminates in finitely many steps with an equivalent static formula*

Proof. It is clear from observing each of the reduction axiom schemas that they result in formulas with smaller dynamic components, *i.e.* smaller parse trees for those components. Eventually an iterative application of the reduction axiom schemas results in a purely static formula that has preserved the truth of the original formula. □

The truth-preservation of the reduction axiom schemas has not yet been established, and we do so here.

Proof. **Atomic Consequence** : $\Rightarrow$. Assume $M, w \models [\mathbf{A}, a]_i p$. We must show that $M, w \models (post(a, pre(a)) \Rightarrow p)$. By the semantics of $[\mathbf{A}, a]_i$, for all (M', w'), if $M, w \models pre(a)$ and $update(M, A, w, a) = (M', w')$, then $M', w' \models p$. By definition of $post(a, p)$, if $update(M, A, w, a) = (M', w')$ and $M', w' \models p$, then $post(a, p) = p$. So, if $M, w \models pre(a)$ is defined, then $post(a, pre(a)) = p$, and thus $post(A, p) \Rightarrow p$.

$\Leftarrow$. Assume $M, w \models (post(a, pre(a)) \Rightarrow p)$. By the definition of $post(a, p)$, if $post(a, p) = p$ then $update(M, A, w, a) \models p$. So, if $M, w \models pre(a)$, then $update(M, A, w, a) \models p$. Therefore, $M, w \models [\mathbf{A}, a]_i p$.

Atomic Negation : $\Rightarrow$. Assume $M, w \models [\mathbf{A}, a]_i \neg\varphi$. It suffices to show that $M, w \models pre(a) \Rightarrow \langle \mathbf{A}, a\rangle_i \neg\varphi$. From the assumption and the semantics, for all (W', w'), if $(M, w)[\![(A, a)_i]\!](M', w')$ then $M', w' \models \neg\varphi$. So, if $M, w \models pre(a)$ and $update(M, A, w, a) = (M', w')$, then $M', w' \models \neg\varphi$. Assume $M, w \models pre(a)$, and it follows that $update(M, A, w, a)$ is defined, so there exists a M', w' such that $(M, w)[\![(A, a)_i]\!](M', w')$ and $update(M, Aw, a) = (M', w')$ and $M', w' \models \neg\varphi$. Therefore, $M, w \models pre(a) \Rightarrow \langle \mathbf{A}, a\rangle_i \neg\varphi$.

$\Leftarrow$. Assume $M, w \models pre(a) \Rightarrow \neg[\mathbf{A}, a]_i \varphi$. This is equivalent to $M, w \models pre(a) \Rightarrow \langle \mathbf{A}, a\rangle_i \neg\varphi$. By the semantics, if $M, w \models pre(a)$, then there exists a (M', w') such that $(M, w)[\![(A, a)]\!](M', w')$ and $\neg\varphi$. The relation $[\![(A, a)]\!]$ is functional, so $\exists$ implies $\forall$. So, for all (M', w'), if $(M, w)[\![(A, a)]\!](M', w')$, then $M', w' \models \neg\varphi$, and therefore $M, w \models [\mathbf{A}, a]_i \neg\varphi$.

Action Conjunction is obvious.

Action and Knowledge. For this proof, assume for simplicity, without loss of generality, that $Actions = \{a\}$.

$\Rightarrow$. Assume $M, w \models [\mathbf{A}, a]_i \mathbf{K_i}\,\varphi$. Unfolding the semantics, for all (M', w'), if $(M, w) \models pre(a)$ and $update(M, A, w, a) = (M', v')$, then $M', w' \models \mathbf{K_i}\,\varphi$. $M', w' \models \mathbf{K_i}\,\varphi$ iff for all $v \in W$, if $wR^i_k v$ and $M, v \models pre(a)$ and $update(M, A, v, a) = (M', v')$ and $a\chi^i_k a$, then $M', v' \models \varphi$. That is, $M, w \models \mathbf{K_i}\,[\mathbf{A}, a]_i \varphi$.

$\Leftarrow$. Assume $M, w \models pre(a) \Rightarrow \mathbf{K_i}\,[\mathbf{A}, a]_i\varphi$. We must show $M, w \models [\mathbf{A}, a]_i\mathbf{K_i}\,\varphi$. Thus, we must show $M, w \models pre(a)$ and $update(M, A, w, a) = (M', w')$ implies $M', w' \models \mathbf{K_i}\,\varphi$. So it suffices to show that if $update(M, A, w, a) = (M', w')$ and $M, w \models \mathbf{K_i}\,[\mathbf{A}, a]_i\varphi$, then $M', w' \models \mathbf{K_i}\,\varphi$. Assume $update(M, A, w, a) = (M', w')$ and $M, w \models \mathbf{K_i}\,[\mathbf{A}, a]_i\varphi$. Then for all v, if $wR^i_k v$, then $M, v \models [\mathbf{A}, a]_i\varphi$. It follows that $M, v \models pre(a)$ and $update(M, A, v, a) = (M', v')$ implies $M', v' \models \varphi$. Since $wR^i_k v$ and $a\chi^i_k a$, it holds that $w'R^{i'}_k v'$. Thus, $M', w' \models \mathbf{K_i}\,\varphi$.

Proofs for **Action and Belief** through **Safe Action and Belief** follow the above proofs exactly analogously.

□

Thus, the dynamic extension inherits soundness and completeness from the static base. We proceed now to the philosophical justification of the dynamic extension, specifically the axiom schemas of *Inevitability* and *Minimum Rationality*.

3.6.1 Inevitability and Minimum Rationality

Recall the two axiom schemas under consideration.

$[\mathbf{A}, a]_i\varphi \Rightarrow [\mathbf{A}, a]^{\mathcal{S}}_i\varphi$	Inevitability
$\langle\mathbf{A}, a\rangle_i\varphi \Rightarrow \mathbf{B_i}\,\langle\mathbf{A}, a\rangle^{\mathcal{S}}_i\varphi$	Minimum Rationality

Table 3.6: $\mathcal{DASL}$ axiom schemas of dynamic non-reductive character.

The axiom schema for *Inevitability* states that if engaging in action a from action structure A guarantees that φ is true as a result, then safely engaging in that action also guarantees that φ will result. One way to think about this is that the postcondition of a mere action is the same function as the postcondition for a safe action. The converse of *Inevitability* does not hold, however, because *pre* is defined on a subset of worlds on which pre_s is defined for a given action, representing the notion that it is *easier* to merely engage in an action than to do so safely. The subset relation here represents number of constraints. Thus there may be worlds on which the safety precondition for an action is undefined but on which the precondition is defined. When the safety precondition is undefined it is trivially true that engaging in the action safely guarantees φ, for all φ. However, φ may not be guaranteed given mere engagement in the action.

The axiom schema for *Minimum Rationality* states that if after engaging in a φ results, then i believes that after safely engaging in a φ results. It is more intuitive when the implication is read as an "only if". This axiom schema is clearly an idealization meant to contribute normative value to the logic rather than descriptive value. It can be interpreted as an assumption that the human agents being modeled do not engage in actions that they believe to be unsafe. For commercial pilots, this assumption is reasonably secure. For other domains, this assumption should be carefully noted and considered.

Chapter 4

Case Studies

4.1 Case Study and Mechanization

In this section we apply the logic just developed to the formal analysis of the Air France 447 aviation incident, then mechanize the formalization in the Coq Proof Assistant. Our mechanization follows similar work by Maliković and Čubrilo [61, 62], in which they mechanize an analysis of the game of Cluedo using Dynamic Epistemic Logic, based on van Ditmarsch's formalization of the game [49]. It is commonly assumed that games must be adversarial, but this is not the case. Games need only involve situations in which players' payoffs depend on the actions of other players. Similarly, knowledge games need not be adversarial, and must only involve diverging information. Thus, it is appropriate to model aviation incidents as knowledge games of sorts, where players' payoffs depend on what others do, specifically the way the players communicate information with each other. The goal is to achieve an accurate situational awareness and provide flight control inputs appropriate for the situation. Failures to achieve this goal result in disaster, and often result from imperfect information flow. A formal model of information flow in these situations provides insight and allows for the application of formal methods to improve information flow during emergency situations.

4.1.1 Air France 447

This case study is based on the authoritative investigative report into Air France 447 performed and released by France's Bureau d'Enquêtes et d'Analyses pour la Sécurité de l'Aviation Civile (BEA), responsible for investigating civil aviation incidents and issuing factual findings[26]. The case is mechanized by instantiating, in Coq, the above logic to reflect the facts of the case. One challenge associated with this is that the readings about inputs present in aviation are often real values on a continuum, whereas for our purposes we require discrete values. We accomplish this by dividing the continuum associated with inputs and readings into discrete chunks, similar to how fuzzy logic maps defines predicates with real values[58].

Air France flight 447 from Rio de Janeiro, Brazil to Paris, France, June 1, 2009. The Airbus A330 encountered adverse weather over the Atlantic ocean, resulting in a clogged Pitot-static system. Consequently, the airspeed indicators delivered unreliable data concerning airspeed to the pilot flying, resulting in confusion. A chain of events transpired in which the pilot overcorrected the plane's horizontal attitude again and again, and continued to input nose up pitch commands, all while losing airspeed. Perhaps most confusing to the pilot was the following situation: the aircraft's angle of attack (AOA) was so high it was considered invalid by the computer, so no stall warning sounded until the nose pitched down into the valid AOA range, at which point the stall warning would sound. When the pilot pulled up, the AOA would be considered invalid again, and the stall warning would cease. The aircraft entered a spin and crashed into the ocean. Palmer [64] argues that had the pilot merely taken no action, the Pitot tubes would have cleared in a matter of seconds, and the autopilot could have returned to Normal Mode.

This paper will formalize an excerpted instance from the beginning of the case, involving an initial inconsistency among airspeed indicators, and the subsequent dangerous input provided by the pilot. Formalized in the logic, the facts of the case allow us to infer that the pilot lacked negative introspection about the safety-critical data required for his action. This demonstrates that the logic allows information about the pilot's situational awareness to flow to the computer, via the pilot's actions. It likewise establishes a safety property to be enforced by the computer, namely that a pilot should maintain negative introspection about safety-critical data, and if he fails to do so, it should be re-established as quickly as possible.

According to the official report, at 2 hours and 10 minutes into the flight, a Pitot probe likely became clogged by ice, resulting in an inconsistency between airspeed indicators, and the autopilot disconnecting. This resulted in a change of mode from Normal Law to Alternate Law 2, in which certain stall and control protections ceased to exist. The pilot then made inappropriate control inputs, namely aggressive nose up commands, the only explanation for which is that he mistakenly believed that the aircraft was in Normal Law mode with protections in place to prevent a stall. This situation, and the inference regarding the pilot's mistaken belief, is modeled in the following application and mechanization of the logic.

We first introduce a lemma relating belief to knowledge, and then formalize the critical moment.

Lemma 4.1.1 (Belief is epistemically consistent)**.** $\mathbf{B_i}\,\varphi \Rightarrow \neg\mathbf{K_i}\,\neg\varphi$.

Proof. From the fact that the belief modality is serial, it holds that

$$\mathbf{B_i}\,\varphi \Rightarrow \langle\mathbf{B}_i\rangle\,\varphi,$$

which is equivalent to

$$\mathbf{B_i}\,\varphi \Rightarrow \neg\mathbf{B_i}\,\neg\varphi.$$

Due to axiom EP1, it follows that

$$\mathbf{B_i}\,\varphi \Rightarrow \neg\mathbf{K_i}\,\neg\varphi. \qquad \square$$

We now formalize the critical moment.

1. $\neg(Mode = Normal)\ldots$ ——————————— configuration.
2. $\langle pilot\rangle_i hardnoseuptrue$——————————- pilot input.
3. $\mathbf{B}_{pilot}\,(Mode = Normal)$——————— from axiom PR, pre_s.
4. $\neg\mathbf{K}_{pilot}\,(Mode = Normal)$————— from axiom K-Reflexive.
5. $\mathbf{B}_{pilot}\,\mathbf{K}_{pilot}\,(Mode = Normal)$————- from (3), axiom EP2.
6. $\neg\mathbf{K}_{pilot}\,\neg\mathbf{K}_{pilot}\,(Mode = Normal)$———- from (5), Lemma 4.1.1.
7. $\neg\mathbf{K}_{pilot}\,(Mode = Normal) \wedge \neg\mathbf{K}_{pilot}\,\neg\mathbf{K}_{pilot}\,(Mode = Normal)$—— from (4), (6).
8. $\neg(\neg\mathbf{K}_{pilot}\,(Mode = Normal) \Rightarrow \mathbf{K}_{pilot}\,\neg\mathbf{K}_{pilot}\,(Mode = Normal))$ —— from (7).

The crux of the case is that inconsistent information was being presented to the pilot, along with a cacophony of inconsistent alarms, and the pilot's control inputs indicated a lack of awareness of safety-critical information. A detailed analysis, using the Coq Proof Assistant and the logic developed by my research, will make explicit these failures of information flow, both from the computer to the pilots, between the pilot and co-pilot, and from the pilot to the computer. This will motivate the description of a prototype safety monitor that identifies and corrects information flow failures like those found in Air France 447.

4.1.2 Mechanization in Coq

Mechanical theorem proving divides into two categories, automated and interactive. Automated theorem proving combines a search algorithm with a proof checking algorithm to fully automate the process. The problem itself is undecidable in general, and human control is limited to the injection of hints prior to the algorithm's execution. Interactive theorem proving, however, combines human-directed search with a proof checking algorithm, allowing the human to have more control over the procedure. Coq is a tool that facilitates interactive theorem proving.

The underlying logic of Coq is called the Calculus of Inductive Constructions, a dependently-typed constructive logic. One uses Coq by formalizing the target logic and its semantics in Coq and using what are called *tactics* to manipulate proof objects. My project will implement the previously described Safe Dynamic Agency Logic in Coq and formally model the Air France 447 case, demonstrating the logic's ability to dynamically model safety-critical information flow in a real-world scenario. This will require translating the logic as it appears here and its metatheory into Coq, complete with tactics appropriate for the desired proofs and fully instantiated semantics, a process called *mechanization*.

The following mechanization demonstrates progress from the artificially simply toy examples normally analyzed in the literature to richer real-world examples. However, it does not represent the full richness of the approach. The actions and instrument readings mechanized in this paper are constrained to those most relevant to the case study. The approach is capable of capturing the full richness of all instrument reading configurations and actions available to a pilot. To do so, one needs to consult a flight safety manual and formally represent each action available to a pilot, and each potential instrument reading, according to the following scheme.

We first formalize the set of agents.

```
Inductive Agents: Type := Pilot | CoPilot | AutoPilot.
```

Next we formalize the set of available inputs. These themselves are not actions, but represent atomic propositions true or false of a configuration. When a pilot manipulates an actuator via an action, the actuator being manipulated changes its state. We assume the pilot changes only one control surface actuator at a time, and so an input consists of the new state of whichever actuator is being manipulated.

```
Inductive Inputs : Type :=
HardThrustPlus | ThrustPlus
| HardNoseUp    | NoseUp
| HardWingLeft  | WingLeft
| HardThrustMinus | ThrustMinus
| HardNoseDown  | NoseDown
| HardWingRight | WingRight.
```

We represent readings by indicating which *side* of the panel they are on. Typically, an instrument has a left-side version, a right-side version, and sometimes a middle version serving as backup. When one of these instruments conflicts with its siblings, the autopilot will disconnect and give control to the pilot.

```
Inductive Side : TYpe := Left | Middle | Right.
```

We divide the main instruments into chunks of values they can take, in order to provide them with a discrete representation in the logic. For example, the reading *VertUp1* may represent a nose up reading between 0° and 10°, while *VertUp2* represents a reading between 11° and 20°.

```
Inductive Readings (s : Side) : Type :=
VertUp1 | VertUp2 | VertUp3 | VertUp4
| VertDown1 | VertDown2 | VertDown3 | VertDown4
| VertLevel | HorLeft1 | HorLeft2 | HorLeft3
| HorRight1 | HorRight2 | HorRight3 | HorLevel
| AirspeedFast1 | AirspeedFast2 | AirspeedFast3
| AirspeedSlow1 | AirspeedSlow2 | AirspeedSlow3
| AirspeedCruise| AltCruise | AltClimb | AltDesc | AltLand.
```

The difference between an input and a reading is that the inputs are values of actuator states, while readings are values of instrument states. In a properly functioning aircraft, the input values and readings values would correspond. However, in emergency situations, these tend to diverge.

We define a set of potential modes the aircraft can be in.

```
Inductive Mode : Type := Normal | Alternate1 | Alternate2.
```

We define a set of global instrument readings representing the mode and all of the instrument readings, left, right, and middle, combined together. This represents the configuration of the instumentation.

```
Inductive GlobalReadings : Type := Global (m: Mode)
(rl : Readings Left)
(rm : Readings Middle)
(rr : Readings Right).
```

The set of atomic propositions we are concerned with are those representing facts about the instrumentation.

```
Inductive Atoms : Type :=
| M (m : Mode)
| Input (a : Inputs)
| InstrumentL (r : Readings Left)
| InstrumentM (r : Readings Middle)
| InstrumentR (r : Readings Right)
| InstrumentsG (g : GlobalReadings).
```

Next we again follow Lescanne and Puisségur [59, 65] and Maliković and Čubrilo [61, 62] in defining *prop* of propositions of type `Type`. The definition provides constructors for atomic propositions consisting of particular instrument reading predicate statements, implications, propositions beginning with a knowledge modality, and those beginning with a belief modality.

```
Inductive prop : Type :=
| atm : Atoms → prop
| imp: prop → prop → prop
| negp: prop → prop
| falsum : prop
| K : Agents → prop → prop
| B : Agents → prop → prop
```

These should be familiar, as they are similar to the Backus-Naur form for the static modal logic from 3,

$$\varphi \ ::= \ p \mid \varphi \Rightarrow \varphi \mid \neg\varphi \mid \bot \mid \mathbf{K_i}\,\varphi \mid \mathbf{B_i}\,\varphi$$

We use the following notation for implication.

```
Infix "==>" := imp (right associativity, at level 85).
```

We likewise follow Lescanne and Puisségur [59, 65] by defining an inductive type *theorem* representing a theorem of $\mathcal{DASL}$. Mechanized, it takes a previously defined *prop* and returns a built-in Coq type *Prop* for propositions in Coq's core logic. The constructors correspond to the Hilbert system, either as characteristic axioms, or inference rules. The first three represent axioms for propositional logic, then the rule Modus Ponens, then the axioms for the epistemic operator plus its Necessitation rule, then the doxastic operator and its Necessitation rule. Do not confuse the Necessitation rules with material implication in the object language. The final constructors capture the axioms relating belief and knowledge. The axioms for dynamic modal operators are defined separately, and are not included here.

```
Inductive theorem : prop → Prop :=
|Hilbert_K: forall p q : prop, theorem (p ⟹ q ⟹ p)
|Hilbert_S: forall p q r : prop,
theorem ((p⟹ q⟹ r)⟹ (p⟹ q)⟹ (p⟹ r))
|Classic_NOTNOT : forall p : prop, theorem ((NOT (NOT p)) ⟹ p)
|MP : forall p q : prop, theorem (p ⟹ q) → theorem p → theorem q
|K_Nec : forall (a : Agents) (p : prop), theorem p → theorem (K a p)
|K_K : forall (a : Agents) (p q : prop),
theorem (K a p ⟹ K a (p ⟹ q) ⟹ K a q)
|K_T : forall (a : Agents) (p : prop), theorem (K a p ⟹ p)
|B_Nec : forall (a : Agents) (p : prop), theorem p → theorem (B a p)
|B_K : forall (a : Agents) (p q : prop),
theorem (B a p ⟹ B a (p ⟹ q) ⟹ B a q)
|B_Serial : forall (a : Agents) (p : prop),
theorem (B a p ⟹ NOT (B a (NOT p)))
|B_4 : forall (a : Agents) (p : prop), theorem (B a p ⟹ B a (B a p))
|B_5 : forall (a : Agents) (p : prop),
theorem (NOT (B a p) ⟹ B a (NOT (B a p)))
|K_B : forall (a : Agents) (p : prop), theorem (K a p ⟹ B a p)
|B_BK : forall (a : Agents) (p : prop), theorem (B a p ⟹ B a (K a p)).
```

We use the following notation for *theorem*:

```
Notation "|-- p" := (theorem p) (at level 80).
```

We encode actions as records in Coq, which allow for the definition of an inductive type along with projection functions. An action consists of the acting pilot, the observability of the action (whether it is observed by other agents or not), the input provided by the pilot, and the preconditions for the action and the safety preconditions for the action, both represented as global atoms.

```
Record Action : Set := act {Ai : Agents; Aj : Agents; pi : PI;
input : Inputs; c : GlobalReadings;
c_s : GlobalReadings}.
```

The variable c holds the configuration representing the precondition for the action, while the variable c_s holds the configuration for the safety precondition. We encode the precondition and safety precondition functions as follows.

```
Function pre (a:Action) : prop := atm (InstrumentsG (c a)).
Function pre_s (a : Action) : prop := atm (InstrumentsG (c_s a)).
```

In the object language, the dynamic modalities of action and safe action are encoded as follows.

```
Parameter aft_ex_act : Action → prop → prop.
Parameter aft_ex_act_s : Action → prop → prop.
```

Many standard properties of logic, like the simplification of conjunctions, hypothetical syllogism, and contraposition, are encoded as Coq axioms. As an example, here is how we encode simplifying a conjunction into just its left conjunct.

```
Axiom simplifyL : forall p1 p2,
|-- p1 & p2 → |-- p1.
```

We formalize the configuration of the instruments at 2 hour 10 minutes into the flight as follows.

```
Definition Config_1 := (atm (M Alternate2)) &
(atm (InstrumentL (AirspeedSlow3 Left))) &
(atm (InstrumentM (AirspeedSlow3 Middle))) &
(atm (InstrumentR (AirspeedCruise Right))).
```

The mode is Alternate Law 2, and the left and central backup instruments falsely indicate that the airspeed is very slow, while the right side was not recorded, but because there was a conflict, we assume it remained correctly indicating a cruising airspeed.

The pilot's dangerous input, a hard nose up command, is encoded as follows.

```
Definition Input1 := act Pilot Pilot Pri HardNoseUp
(Global Alternate2 (AirspeedSlow3 Left)
(AirspeedSlow3 Middle)
(AirspeedCruise Right))
(Global Normal (AirspeedCruise Left)
(AirspeedCruise Middle)
(AirspeedCruise Right)).
```

The action is represented in the object language by taking the dual of the dynamic modality, $\neg[\mathbf{i}, (\mathbf{A}, a)]_i \neg True$, equivalently $\langle \mathbf{i} \rangle_i (A, a) True$, indicating that the precondition is satisfied and the action token is executed.

```
Definition Act_1 := NOT (aft_ex_act Input1 (NOT TRUE)).
```

The actual configuration satisfies the precondition for the action, but it is inconsistent with the safety precondition. The safety precondition for the action indicates that the mode should be Normal and the readings should consistently indicate cruising airspeed. However, in Config_1, the conditions do not hold. Thus, the action is unsafe. From the configuration and the action, DASL allows us to deduce that the pilot lacks negative introspection of the action's safety preconditions.

Negative introspection is an agent's awareness of the current unknowns. To lack it is to be unaware of one's unknown variables, so lacking negative introspection about one's safety preconditions is to be unaware that they are unknown.

```
Theorem NegIntroFailMode :
|-- (Config_1 ==>
Act_1 ==>
((NOT (K Pilot (pre_s(Action1)))) &
(NOT (K Pilot (NOT (K Pilot (pre_s(Action1)))))))).
```

In fact, in general it holds that if the safety preconditions for an action are false, and the pilot executes that action, then the pilot lacks negative introspection of those conditions. We have proven both the above theorem, and the more general theorem, in Coq.

```
Theorem neg_intro_failure :
forall (A Ao : Agents) (pi : PI) (inp : Inputs)
(m : Mode)
(rl : Readings Left) (rm : Readings Middle) (rr : Readings Right)
(ms : Mode)
(rls : Readings Left) (rms : Readings Middle) (rrs : Readings Right)
phi,
|-- (NOT
(aft_ex_act
(act A Ao pi inp (Global m rl rm rr) (Global ms rls rms rrs))
(NOT phi)) ⟹
NOT (atm (InstrumentsG (Global ms rls rms rrs))) ⟹
(NOT (K A (atm (InstrumentsG (Global ms rls rms rrs)))) &
(NOT (K A (NOT (K A (atm (InstrumentsG (Global ms rls rms rrs))))))))).
```

This indicates that negative introspection about safety preconditions is a desirable safety property to maintain, consistent with the official report's criticism that the Airbus cockpit system did not clearly display the safety critical information. The logic described in this research accurately models the report's findings that the pilot's lack of awareness about safety-critical information played a key role in his decision to provide unsafe inputs. Furthermore, the logic supports efforts to automatically infer which safety-critical information the pilot is unaware of and effectively display it to him.

The next section formalizes additional case studies in DASL.

4.2 Additional Case Studies

To illustrate the flexibility of this approach, we now formalize additional case studies in the logic. We analyze Copa Airlines flight 201 and Asiana Airlines flight 214.

4.2.1 Copa 201 and Asiana 214

Copa flight 201 departed Panama City, Panama for Cali, Colombia in June, 1992. Due to faulty wiring in the captain's Attitude Indicator, he incorrectly believed he was in a left bank position. In response to this, he directed the plane into an 80 degree roll to the right, which caused the plane to enter a steep dive. A correctly functioning backup indicator was available to the captain, and investigators believe that the captain intended to direct the backup indicator's readings to his own, but due to an outdated training module, the flip he switched actually sent his own faulty readings to the co-pilot's indicator. Approximately 29 minutes after takeoff, the plane crashed into the jungle and all passengers and crew perished. We formalize the moment at which the pilot provides the hard right roll input.

We begin with the present configuration of the instruments, simplified.

```
Definition Config_2 := (atm (M Normal))
  & (atm (InstrumentL (HorLeft2 Left)))
  & (atm (InstrumentM (HorLevel Middle)))
  & (atm (InstrumentR (HorLevel Right))).
```

We define the pilot's input and corresponding action.

```
Definition Input2 := act Pilot Pilot Pri HardWingRight
   (Global Normal (HorLeft2 Left) (HorLevel Middle) (HorLevel Right))
   (Global Normal (HorLeft2 Left) (HorLeft2 Middle) (HorLeft2 Right)).


Definition Act_2 := NOT (aft_ex_act Input2 (NOT TRUE)).
```

Finally, we state the theorem but omit the proof for space. It is available on the $\mathcal{DASL}$ Github.

```
Theorem NegIntroFailHorLevel :
   |-- (Config_2 ==> Act_2 ==>
   ((NOT (K Pilot (pre_s(Input2)))) &
   (NOT (K Pilot (NOT (K Pilot (pre_s(Input2)))))))).
```

Asiana flight 214 from South Korea to San Francisco departed in the evening of July 6, 2013 and was schedule to land just before noon that morning [63]. The weather was good and air traffic control cleared the pilots to perform a visual approach to the runway. The plane came in short and crashed against an embankment in front of the runway, resulting in the deaths of three passengers and 187 injured. The National Transportation Safety Board (NTSB) investigation found that the captain had mismanaged the approach and monitoring of the airspeed, resulting in the plane being too high for a landing. Upon noticing this, the captain selected a flight mode (flight level change speed) which unexpectedly caused the plane to climb higher. In response to this, the captain disconnected the autopilot and pulled back on the thrust. This caused an autothrottle (A/T) protection to turn off, so when the captain pitched the nose down, the plane descended faster than was safe, causing it to come down too quickly and collide with the embankment in front of the runway. We will formalize the moment at which the pilot pitches the nose down.

```
Definition Config_3 := (atm (M Alternate1))
    & (atm (InstrumentL (AirspeedSlow3 Left)))
    & (atm (InstrumentM (AirspeedSlow3 Middle)))
    & (atm (InstrumentR (AirspeedSlow3 Right))).
```

```
Definition Input3 := act Pilot Pilot Pri HardThrustMinus
    (Global Alternate1 (AirspeedSlow3 Left)
                       (AirspeedSlow3 Middle)
                       (AirspeedSlow3 Right))
    (Global Normal (AirspeedSlow3 Left)
                   (AirspeedSlow3 Middle)
                   (AirspeedSlow3 Right)).

Definition Act_3 := NOT (aft_ex_act Input3 (NOT TRUE)).
```

The corresponding theorem states that the pilot engaging in the action under those conditions is unaware of that the safety precondition is false.

```
Theorem NegIntroFailATOff :
    |-- (Config_3 ==> Act_3 ==>
    ((NOT (K Pilot (pre_s(Input3))))
    & (NOT (K Pilot (NOT (K Pilot (pre_s(Input3)))))))).
```

The above formalizations follow the same format as that of Air France 447. A pilot provides an input whose safety precondition conflicts with one of the instruments in the configuration, and we infer that the pilot lacks negative introspection of the safety precondition. This is distinct from but related to the property that the pilots, in engaging in an unsafe action, are unaware of the unsafe instrument readings. We can capture this in the form of safety properties, which I turn to in the next section.

4.3 Safety Properties

Definition. Safety Negative Introspection (SNI). If a safety precondition does not hold, then agent knows that he does not know it to hold.

$$\neg pre_s(a) \Rightarrow \mathbf{K_i} \neg \mathbf{K_i}\, pre_s(a)$$

Definition. Safety-Critical Delivery (SCD). If a safety precondition is false, then agent knows that it is false.

$$\neg pre_s(a) \Rightarrow \mathbf{K_i} \neg pre_s(a)$$

Our above formalizations show that SNI is false when a pilot provides an unsafe input. Notice that SCD implies SNI.

Lemma 4.3.1 (SCD implies SNI). *Safety-Critical Delivery (SCD) implies Safety Negative Introspection (SNI).*

Proof. It suffices to show that $\mathbf{K_i} \neg\varphi \Rightarrow \mathbf{K_i} \neg \mathbf{K_i}\, \varphi$. Assume $\mathbf{K_i} \neg\varphi$ holds. From EP1, it follows that $\mathbf{B_i} \neg\varphi$, and because knowledge is a normal modality, it follows that $\mathbf{K_i}\, \mathbf{B_i} \neg\varphi$ holds. From lemma 1, and again the fact that knowledge is normal modality, it follows that $\mathbf{K_i} \neg \mathbf{K_i} \neg\neg\varphi$, or equivalently, that $\mathbf{K_i} \neg \mathbf{K_i}\, \varphi$. $\square$

However, the converse does not hold. We can satisfy SNI when the safety precondition is false, the agent knows that he doesn't know it, but doesn't know that it is false. A counterexample consists of a model with three worlds: {u,v}. Let φ be the safety precondition, with the following truth assignment: {False, True}. Let the epistemic relation include (u,v), (v,u), and the reflexive relations. Then at world u φ is false, and $\mathbf{K_i} \neg \mathbf{K_i} \varphi$ is true, but $\mathbf{K_i} \neg \varphi$ is false.

The formalizations show that from the pilot's unsafe action, it follows that he lacks negative introspection of the safety precondition.

$$\langle \mathbf{A}, a \rangle_i true \wedge \neg pre_s(a) \Rightarrow \neg \mathbf{K_i} \neg \mathbf{K_i}\, pre_s(a) \tag{4.1}$$

This situation violates SNI, because the pilot doesn't know that he doesn't know the safety precondition. Since SCD implies SNI, SCD is also violated.

$$\langle \mathbf{A}, a \rangle_i true \wedge \neg pre_s(a) \Rightarrow \neg \mathbf{K_i} \neg pre_s(a) \tag{4.2}$$

So, from an unsafe action, we can also infer that the pilot does not know that the safety precondition is false, a stronger conclusion.

Thus, by restoring knowledge that the safety precondition is false, it follows that either the safety precondition is true, or the unsafe action is not executed.

$$\mathbf{K_i} \neg pre_s(a) \Rightarrow \neg \langle \mathbf{A}, a \rangle_i true \vee pre_s(a) \tag{4.3}$$

The pilot's knowledge in the antecedent implies that the safety precondition is false, so this simplifies to:

$$\mathbf{K_i} \neg pre_s(a) \Rightarrow \neg \langle \mathbf{A}, a \rangle_i true. \tag{4.4}$$

This squares with the standard game theoretic inference, wherein a rational agent with knowledge of the situation executes a good action. Because our model of knowledge and rationality is weaker, we make the weaker claim that a minimally rational pilot with knowledge of the safety-critical information does not execute a bad action.

4.4 Decision Problem

An important extension of the foundational work provided by this paper is the construction of a system that takes advantage of the logic as a runtime safety monitor. It will monitor the pilot's control inputs and current flight configurations, and in the event that an action's safety preconditions do not hold, infer which instrument readings the pilot is unaware of and act to correct this. In order to avoid further information overload, the corrective action taken by the computer should be to temporarily remove or dim the non-safety-critical information from competition for the pilot's attention, until the pilot's unsafe control inputs are corrected, indicating awareness of the safety-critical information. Construction of a prototype of this system is underway. Here we define the decision problem and prove that it is NP-Complete. The decision problem we face is formalized as follows.

SD. Input: $a : \mathit{Action}$, $C : \mathit{Configuration}$, $\mathit{pre}_s : \mathit{Action} \mapsto Configuration$, $k : \mathit{Int}$.

Output: Is there a set of instruments I of size k from configuration C that falsifies $\mathit{pre}_s(a)$?

Theorem 4.4.1 (NP)**.** *SD is NP-Complete.*

Proof. First, we prove that SD is in NP by defining the decision problem SDV that takes input to SD and a *certificate* and verifies that the certificate falsifies $\mathit{pre}_s(a)$ in polynomial time. Let the certificate be a set of instruments I of size k from the configuration C. Note that $\mathit{pre}_s(a)$ has the form $(c_1 \wedge c_2 \wedge \ \ldots \ \wedge c_n) \vee (c'_1 \wedge c'_2 \wedge \ \ldots \ \wedge c'_n) \vee \ \ldots$ The negation of this has the form $(\neg c_1 \vee \neg c_2 \vee \ \ldots \ \vee \neg c_n) \wedge (\neg c'_1 \vee \neg c'_2 \vee \ \ldots \neg c'_n) \wedge \ \ldots$, which is in Conjunctive Normal Form. Computing the negation of pre_s can be done in linear time. Next we take I and treat it as an assignment of values to instruments of the form $(i_1 \wedge i_2 \wedge \ \ldots \ \wedge i_n)$ for each $i \in I$. Then we check whether I satisfies $\neg \mathit{pre}_s(a)$ in polynomial time, treating I as the certificate for the

SAT problem.

Second, we prove that SD is in NP-Hard. We do this by providing a polynomial time reduction from $nSAT$ to SD. Taking the input from $nSAT$, we let the size of the configuration $|C| = n$, that is, we let n be the number of instruments on the flight deck. The maximum size of any solution I to SD is therefore n. We let the $nSAT$ formula be the negation of $pre_s(a)$. We iterate over $k \in \{1..n\}$. Thus, for $nSAT$, the SD problem is run at most n times. Any input to $nSAT$ will be at least of size n, so the number of times SD is run is polynomial in the length of $nSAT$'s input. $\square$

4.4.1 Summary

This chapter mechanized an instance of $\mathcal{DASL}$appropriate for reasoning about safety-critical information flow and action in the context of aviation mishaps. It illustrates how $\mathcal{DASL}$is useful as a formal method for modeling human agents and the logical relationship between their actions and their mental states. Similar instances of $\mathcal{DASL}$ could occur in a variety of situations, where a human agent makes decisions and it is important to rapidly identify when unsafe actions occur, and which safety-critical information would improve the safety of those actions.

We proved that when a rational pilot engages in unsafe action, it is because she is unaware of some safety-critical information. We explored the implications of this insight by identifying the information assurance properties of *Safety Negative Introspection* (SNI) and *Safety-Critical Delivery* (SCD), which formalize the notions that an agent should be aware when an action's safety precondition is unknown, and that she should know when an action's safety precondition is false, respectively. We proved that the latter implies the former, and that the aviation mishaps involve violations of SNI (and therefore SCD as well). This shows that both negative introspection and direct knowledge of a safety precondition's being false are sufficient for preventing unsafe actions. This formalizes the standard game-theoretic inference, weakened for our correspondingly weaker agents, that knowledge implies the *avoidance* of bad actions.

Finally, we showed that capitalizing on this insight via an algorithm is an NP-Complete problem. The decision problem takes as input an action, configuration, safety precondition, and integer k, and returns true or false whether there is a set of instrument readings of size k that falsifies the safety precondition for the given action.

The next chapter presents a theoretic worry facing $\mathcal{DASL}$, and indeed most modal logics of belief and knowledge developed in the literature so far.

Chapter 5

Löb's Obstacle

Thus far, the dynamic modal logic we have constructed and applied has followed closely the standard formalizations in the literature, departing primarily in how it deals with an agent's static knowledge. The static operator for an agent's belief remains $KD45$, as it is normally formalized in the literature. Our dynamic operators follow closely those defined in [55] and [16], with the novel addition of a safety precondition to capture normative constraints on action in addition to logical constraints. The dissertation thus far, and the logics of knowledge and agency in the literature, ignore a problem facing agents of a certain reflective type, identified by Smullyan in [71].[1] This chapter confronts the problem of reflective reasoners facing modal logics of knowledge and agency, including $\mathcal{DASL}$. We respond with an adjustment to $\mathcal{DASL}$'s static foundation so that the problem can be avoided, if the agents being modeled are reflective reasoners.

In Smullyan's, "Logicians who reason about themselves," he considers epistemic problems related to undecidability results in mathematics. He identifies "a complete parallelism between logicians who believe propositions and mathematical systems that

[1] Smullyan refers to these as reflexive reasoners, but we use the term 'reflective' in order to avoid ambiguity with the reflexive frame condition on worlds.

prove propositions.” In provability logic, the formula $\varphi \Leftrightarrow \neg \mathbf{B_i}\, \varphi$ expresses the Gödel proposition, “This proposition is not provable in system i.” In a doxastic interpretation, the same formula expresses the reflective belief, “agent i does not believe this proposition.” This means that for any doxastic or epistemic system, if the agents it models are reflective, then it must not have the theorem $\mathbf{B_i}\, \varphi \Rightarrow \varphi$, under pain of contradiction.

A *reflective* agent is one that can form beliefs and knowledge about self-referential sentences and propositions. Such propositions are of the form $\varphi \Leftrightarrow (\mathbf{B_i}\, \varphi \Rightarrow \psi)$. The reflective proposition “agent i does not believe this proposition” is of that form, where ψ is replaced with $\perp$ in the generic form. More specifically, an agent is *reflective* just in case for every proposition ψ, there is a proposition φ such that $\varphi \Rightarrow (\mathbf{B_i}\, \varphi \Rightarrow \psi)$.

The above uses the $\mathbf{B_i}$ operator for belief, but just as easily we could have used $\mathbf{K_i}$. The problem for epistemic logic is immediately apparent. Most humans are capable of reasoning about self-referential propositions, so a logic for human knowledge ought to include such propositions. However, with the Truth Axiom of the knowledge operator, this seems to yield inconsistency. It seems to yield the very same inconsistency that mathematical system i would face if it could prove its own soundness.

In what follows, we examine the mathematical logic that underpins this issue. Just as Smullyan identified the problem as one facing reflective reasoners with Axioms K and 4 and the Rule of Necessitation in [71], Löb identified these same conditions as the ones that allow a mathematical system to derive his theorem in [22]. We highlight the obstacle this presents to epistemic and doxastic logics, and show how to avoid it. We present a relaxed $\mathcal{DASL}$ for reasoning about safety-critical information flow to reflective agents.

The potential problem is due to Löb in [22] from the mathematical logic perspective, and Smullyan in [71] from the epistemic/doxastic logic perspective. Contemporary research in artificial intelligence foundations addresses the problem as it pertains to agents being confident in their own conclusions. See for example research at the Machine Intelligence Research Institute[2], who kindly invited the author to a workshop introducing him to the problem. Some of their preliminary results on the problem are available on the footnoted website. They have named the problem Löb's Obstacle, or the Löbian Obstacle. We describe it, and the problem it presents to modal logic of agency, including our own, below.

5.1 Löb's Theorem

Löb's Theorem takes its name from Martin Hugo Löb, who tackled a question of mathematical logic posed by Leon Henkin in the years following the results of Gödel. Henkin asked what could be said of propositions asserting their own provability, as opposed to unprovability in the case of Gödel sentences.[72] Löb answered by showing that in a consistent system, proof that proof of a proposition implies that the proposition is true (soundness) only for propositions that are actually provable. This is surprising, because it generalizes Gödel's Second Incompleteness Theorem: no formal system can prove its own soundness (as opposed to consistency).

Löb's Theorem in provability logic is,

$$\Box(\Box\varphi \Rightarrow \varphi) \Rightarrow \Box\varphi. \tag{5.1}$$

The $\Box$ is interpreted as "provability" in some formal system, particularly one at least as powerful as Peano arithmetic. If a modal operator involves the reasoning abilities of a human-like agent, then *a fortiori* it is a formal system at least as powerful as Peano arithmetic. This presents the following problem. If that agent, in its own reasoning

[2]See their website at https://www.intelligence.org

system, can deduce the soundness of its own system, then its reasoning system is unsound. This is because φ can be any formula, including $\bot$. The particular modal logic at risk, in our mind, is epistemic logic, or any doxastic logic with accurate reasoners.

Löb identified the following conditions of a formal system that would allow derivation of his theorem (which we present in modal form).

1. $\Box(\varphi \Rightarrow \psi) \Rightarrow (\Box\varphi \Rightarrow \Box\psi)$ Axiom K
2. $\Box\varphi \Rightarrow \Box\Box\varphi$ Axiom 4
3. From $\vdash \varphi$, infer $\Box\varphi$ Rule of Necessitation

Items (1) and (3) are constants for all normal modal logics. There is a suppressed condition that Löb did not mention because it was unnecessary for the domain of arithmetic, but we must mention it here. The system must either admit of self-referential sentences or involve modal fixed points. Because systems of human-like reasoning must include self-referential sentences, at least in the form of Peano arithmetic, in order to remain human-like, this condition is satisfied for our concerns.

Here we give a template derivation of Löb's Theorem, which we shall refer to below when describing how Löb's Obstacle corrupts various epistemic logics.

Proof.

(1.1) $\Box(\Box\varphi \Rightarrow \varphi)$ .... Assumption

(1.2) $\Box(\psi \Leftrightarrow (\Box\psi \Rightarrow \varphi))$ .... Löb Sentence[3]

(1.3) $\Box(\Box\psi \Leftrightarrow \Box(\Box\psi \Rightarrow \varphi))$ .... Axiom K

(1.4) $\Box(\Box\psi \Rightarrow \Box(\Box\psi \Rightarrow \varphi)))$ .... (1.3) Simplification of $\Leftrightarrow$

[3]Sometimes referred as a Curry sentence after logician Haskell Curry.

(1.5) $\Box(\Box\psi \Rightarrow (\Box\Box\psi \Rightarrow \Box\varphi))$ .... (1.4) Axiom K

(1.6) $\Box(\Box\psi \Rightarrow \Box\Box\psi)$ .... Axiom 4

(1.7) $\Box(\Box\psi \Rightarrow \Box\varphi)$ .... (1.5), (1.6)

(1.8) $\Box(\Box\psi \Rightarrow \varphi)$ .... (1.7), (1.1)

(1.9) $\Box\psi$ .... (1.3), (1.8)

(1.10) $\Box\Box\psi$ .... (1.9), Axiom 4

(1.11) $\Box\Box\psi \Rightarrow \Box\varphi$ .... (1.8), Axiom K

(1.12) $\Box\varphi$ .... (1.10), (1.11)

$\mathcal{QED}$ $\Box$

Mathematical and Provability logicians refer to the key components of this proof as Löb Conditions[10]. Identifying them in the proof above helps us identify which epistemic logics collide with Löb's Obstacle. Conversely, understanding how the Löb Conditions interact helps us construct epistemic logics that avoid the Obstacle.

The Conditions are:

1. The Löb Sentence. A self-referential sentence, also formalizable as a modal fixed point.

2. Axiom K. The standard distribution axiom of normal modal logics.

3. Axiom 4. The axiom corresponding to a transitive frame relation.

4. The rule of necessitation. Likewise a standard feature of normal modal logics.

The Löb Sentence is sometimes not mentioned as a Condition, because Löb's Theorem is typically studied in the context of mathematical logic or provability logic, where such self-referential expressiveness is known to exist. We point out, however, that humans are capable of reasoning about self-referential sentences, as well, and any advanced artificial agent will be able to do so, as well.

Finally, we note the importance of Löb's Theorem's antecedent: $\Box(\Box\varphi \Rightarrow \varphi)$. Epistemic logics typically include the antecedent as a theorem, in which case Löb's Theorem will allow us to derive $\Box\varphi$ for all φ. This is why consistent mathematical systems at least as expressive as Peano arithmetic cannot prove their own consistency.

We identify some candidate epistemic logics and, on the assumption that they capture human-like reasoning, show how they crash into Löb's Obstacle.

5.2 Epistemic Logics that Crash

5.2.1 $\mathcal{S}5$ Epistemic Logic

The most prominent epistemic logic in the literature, by far, is $\mathcal{S}5$ epistemic logic. $\mathcal{S}5$ epistemic logic is routinely presented as the logic of knowledge, and often serves as a static base for dynamic extensions to epistemic logic involving action and communication. Its characteristic axioms are:

$$\mathbf{K_i}\,(\varphi \Rightarrow \psi) \Rightarrow (\mathbf{K_i}\,\varphi \Rightarrow \mathbf{K_i}\,\psi) \tag{5.2}$$

$$\mathbf{K_i}\,\varphi \Rightarrow \varphi \tag{5.3}$$

$$\neg\mathbf{K_i}\,\varphi \Rightarrow \mathbf{K_i}\,\neg\mathbf{K_i}\,\varphi \tag{5.4}$$

Clearly (2) is Axiom K, and (3), troublingly, is the antecedent of Löb's Theorem, known as Axiom T. (4) is called the Negative Introspection axiom, or sometimes in philosophy circles, the Wisdom Axiom. Logicians call it Axiom 5. It is read, "If i does not know that φ, then she knows that she doesn't know it". Other than being clearly invalid for humans, this axiom and (3) allows us to derive,

$$\mathbf{K_i}\,\varphi \Rightarrow \mathbf{K_i}\,\mathbf{K_i}\,\varphi$$

Proof.

(2.1) $\neg\mathbf{K_i}\,\neg\mathbf{K_i}\,\varphi \Rightarrow \mathbf{K_i}\,\varphi$	Contrapositive of Axiom 5
(2.2) $\mathbf{K_i}\,\neg\mathbf{K_i}\,\neg\mathbf{K_i}\,\varphi \Rightarrow \mathbf{K_i}\,\mathbf{K_i}\,\varphi$	Rule of Necessitation on (5.1), Axiom K
(2.3) $\varphi \Rightarrow \neg\mathbf{K_i}\,\neg\varphi$	Axiom T, Contrapositive
(2.4) $\neg\mathbf{K_i}\,\neg\varphi \Rightarrow \mathbf{K_i}\,\neg\mathbf{K_i}\,\neg\varphi$	Axiom 5
(2.5) $\varphi \Rightarrow \mathbf{K_i}\,\neg\mathbf{K_i}\,\neg\varphi$	(4.3), (4.4)
(2.6) $\mathbf{K_i}\,\varphi \Rightarrow \mathbf{K_i}\,\neg\mathbf{K_i}\,\neg\mathbf{K_i}\,\varphi$	$\mathbf{K_i}\,\varphi/\varphi$, (4.5)
(2.7) $\mathbf{K_i}\,\varphi \Rightarrow \mathbf{K_i}\,\mathbf{K_i}\,\varphi$	(4.2), (4.6)

$\mathcal{QED}$ $\square$

Thus, $\mathcal{S}5$ satisfies Löb's three conditions, if we assume the presence of self-referential sentences possible, which we should. Therefore, with $\mathbf{K_i}$ instead of $\square$, the proof of Löb's Theorem is possible in this brand of $\mathcal{S}5$. However, to make matters worse, the antecedent of Löb's Theorem is itself an axiom of S5. Therefore, $\mathbf{K_i}\,\varphi$ is a theorem, for all φ.

We take this as a *reductio ad absurdum* that $\mathcal{S}5$ epistemic logic cannot be a logic for reasoning about the knowledge of agents with expressive power beyond Peano arithmetic. Therefore, it cannot be a logic of knowledge for humans, or human-like agents.

5.2.2 Hintikka's S4 Epistemic Logic

In Hintikka's 1967 *Knowledge and Belief: A logic of the two notions*, he presented an epistemic logic for determining the validity and consistency of claims people make about knowledge and belief. He rejected out of hand the negative introspection axiom for knowledge, but chose to include positive introspection, which is formalized as $\mathbf{K_i}\,\varphi \Rightarrow \mathbf{K_i}\,\mathbf{K_i}\,\varphi$. Clearly then, if Hintikka's epistemic system is meant for human-like reasoners who can express sentences like, "If I know this sentence is true, then $1 + 1 = 2$," then it crashes into Löb's Obstacle, with the extra bite of having the antecedent of Löb's Theorem as a theorem itself, and therefore, $\mathbf{K_i}\,\varphi$ is also a theorem.

5.2.3 Kraus and Lehman System

In [**?**], Kraus and Lehman tackle the issue of how to combine knowledge and belief in a single system remarked on in the conclusion of Halpern and Moses in [32]. The project at the time was very similar to that addressed by this dissertation's static foundation: the development of a logic sufficiently realistic and expressive for reasoning about intelligent agents.

They axiomatize knowledge and belief as follows.

$\mathbf{K_i}\,(\varphi \Rightarrow \psi) \Rightarrow (\mathbf{K_i}\,\varphi \Rightarrow \mathbf{K_i}\,\psi)$	Distribution of $\mathbf{K_i}$
$\mathbf{K_i}\,\varphi \Rightarrow \varphi$	Truth
$\neg\mathbf{K_i}\,\varphi \Rightarrow \mathbf{K_i}\,\neg\mathbf{K_i}\,\varphi$	Negative Introspection
$\mathbf{B_i}\,(\varphi \Rightarrow \psi) \Rightarrow (\mathbf{B_i}\,\varphi \Rightarrow \mathbf{B_i}\,\psi)$	Distribution of $\mathbf{B_i}$
$\mathbf{B_i}\,\varphi \Rightarrow \langle\mathbf{B}_i\rangle\,\varphi$	Belief Consistency
$\mathbf{K_i}\,\varphi \Rightarrow \mathbf{B_i}\,\varphi$	Knowledge implies Belief
$\mathbf{B_i}\,\varphi \Rightarrow \mathbf{K_i}\,\mathbf{B_i}\,\varphi$	Conscious Belief
From $\vdash \varphi$ and $\vdash \varphi \Rightarrow \psi$, infer $\vdash \psi$	Modus Ponens
From $\vdash \varphi$, infer $\vdash \mathbf{K_i}\,\varphi$	Necessitation of $\mathbf{K_i}$

Table 5.1: Logic of Kraus and Lehman

The key differences with $\mathcal{DASL}$ are that knowledge is $\mathcal{S}5$ and the axiom of Conscious Belief, which reverses the composition of knowledge and belief under belief relative to $\mathcal{DASL}$'s Evidential Restraint axiom. From the above axiom schemas, it follows that belief is a regular $\mathcal{KD}45$ operator.

In their article, and in Meyer and van der Hoek's [36], they show that $\mathbf{B_i}\,(\mathbf{B_i}\,\varphi \Rightarrow \varphi)$ is a theorem. Therefore, this system suffers from a particularly bad collision with Löb: Both the knowledge and belief modalities lead to inconsistency for intelligent agenst capable of self-referential reasoning.

5.3 $\mathcal{DASL}$

$\mathbf{K_i}\,(\varphi \Rightarrow \psi) \Rightarrow (\mathbf{K_i}\,\varphi \Rightarrow \mathbf{K_i}\,\psi)$	Distribution of $\mathbf{K_i}$
$\mathbf{K_i}\,\varphi \Rightarrow \varphi$	Truth
$\mathbf{B_i}\,(\varphi \Rightarrow \psi) \Rightarrow (\mathbf{B_i}\,\varphi \Rightarrow \mathbf{B_i}\,\psi)$	Distribution of $\mathbf{B_i}$
$\mathbf{B_i}\,\varphi \Rightarrow \langle \mathbf{B}_i \rangle\,\varphi$	Belief Consistency
$\mathbf{K_i}\,\varphi \Rightarrow \mathbf{B_i}\,\varphi$	Knowledge implies Belief
$\mathbf{B_i}\,\varphi \Rightarrow \mathbf{B_i}\,\mathbf{K_i}\,\varphi$	Evidential Restraint
From $\vdash \varphi$ and $\vdash \varphi \Rightarrow \psi$, infer $\vdash \psi$	Modus Ponens
From $\vdash \varphi$, infer $\vdash \mathbf{K_i}\,\varphi$	Necessitation of $\mathbf{K_i}$

Table 5.2: Logic of $\mathcal{DASL}$

Theorem 5.3.1 (Positive Belief Introspection). $\mathbf{B_i}\,\varphi \Rightarrow \mathbf{B_i}\,\mathbf{B_i}\,\varphi$

Proof.

(3.1) $\mathbf{B_i}\,\varphi \Rightarrow \mathbf{B_i}\,\mathbf{K_i}\,\varphi$ — ER Axiom

(3.2) $\mathbf{B_i}\,\mathbf{K_i}\,\varphi \Rightarrow \mathbf{B_i}\,\mathbf{B_i}\,\varphi$ — KiB Axiom + Necessitation of $\mathbf{B_i}$ [4]

(3.3) $\mathbf{B_i}\,\varphi \Rightarrow \mathbf{B_i}\,\mathbf{B_i}\,\varphi$ — (3.1), (3.2)

□

[4] This rule can be derived from Necessitation of $\mathbf{K_i}$ and KiB Axiom.

$\mathcal{QED}$

The logic defined by these axiom schemas and inference rules avoids Löb's Obstacle for $\mathbf{K_i}$, as it is no longer has the positive introspection property.

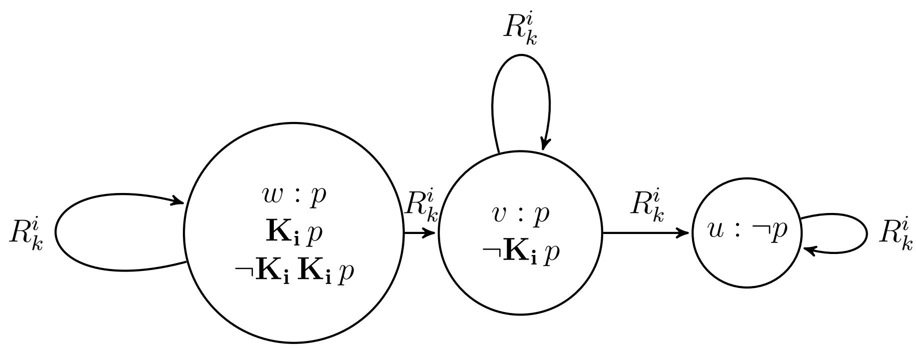


Figure 5.1: A counterexample to $\mathbf{K_i}\,\varphi \Rightarrow \mathbf{K_i}\,\mathbf{K_i}\,\varphi$.

Doxastic logic typically includes as an axiom of Belief Consistency, which corresponds to a serial doxastic possibility relation. Crucially, this axiom $\mathbf{B_i}\,\varphi \Rightarrow \langle\mathbf{B}_i\rangle\,\varphi$ is equivalent to $\neg(\mathbf{B_i}\,\varphi \wedge \mathbf{B_i}\,\neg\varphi)$, which is furthmore equivalent to $\neg\mathbf{B_i}\,(\varphi \wedge \neg\varphi)$, which results in a disaster.

Theorem 5.3.2 (Consistency Disaster). *If* $\neg\mathbf{B_i}\,(\varphi \wedge \neg\varphi)$ *and* $\mathbf{B_i}\,(\mathbf{B_i}\,\varphi \Rightarrow \varphi) \Rightarrow \mathbf{B_i}\,\varphi$ *are theorems, then* $\mathbf{B_i}\,(\varphi \wedge \neg\varphi)$.

Proof.

(3.1) $\neg\mathbf{B_i}\,(\varphi \wedge \neg\varphi)$	Belief is Consistent
(3.2) $\mathbf{B_i}\,(\varphi \wedge \neg\varphi) \Rightarrow (\varphi \wedge \neg\varphi)$	(3.1)
(3.3) $\mathbf{B_i}\,(\mathbf{B_i}\,(\varphi \wedge \neg\varphi) \Rightarrow (\varphi \wedge \neg\varphi))$	Necessitation of $\mathbf{B_i}$
(3.4) $\mathbf{B_i}\,(\mathbf{B_i}\,(\varphi \wedge \neg\varphi) \Rightarrow (\varphi \wedge \neg\varphi)) \Rightarrow \mathbf{B_i}\,(\varphi \wedge \neg\varphi)$	Löb's Theorem
(3.5) $\mathbf{B_i}\,(\varphi \wedge \neg\varphi)$	(3.3), (3.4)

□

Thus, with Belief Consistency, Löb's Theorem, and Theorem 5.3.2, it follows that:

Theorem 5.3.3 (Beliefs Inconsistent)**.** *For all* φ, $\mathbf{B_i}\,\varphi$.

Proof. This follows from Theorem 5.3.2 and $\neg\mathbf{B_i}\,(\varphi \wedge \neg\varphi)$. □

$\mathcal{QED}$

Therefore, the logic is inconsistent. Our only recourse, if we allow self-reference, is to severely relax the axiom schemas.

One might wonder whether it would be acceptable to abandon the Truth Axiom for knowledge and allow Löb's Theorem to hold for the knowledge operator. This would introduce more modesty to the notion of knowledge, where a human-like agent knows that her knowledge is true only for those propositions that she actually knows, but not in the general sense. What would this mean for epistemology? A false proposition would no longer imply a lack of knowledge. It is an unfamiliar notion, perhaps worth exploring. Relaxing the Truth Axiom allows positive and negative introspection to live harmoniously with self-reference. We leave this for future work to explore.

5.4 Avoiding Löb

We leave the reader with the following logic that avoids the Löbian Obstacle while retaining the inference of safety-critical information that is missing.

$\mathbf{K_i}\,(\varphi \Rightarrow \psi) \Rightarrow (\mathbf{K_i}\,\varphi \Rightarrow \mathbf{K_i}\,\psi)$	Distribution of $\mathbf{K_i}$
$\mathbf{K_i}\,\varphi \Rightarrow \varphi$	Truth
$\mathbf{B_i}\,(\varphi \Rightarrow \psi) \Rightarrow (\mathbf{B_i}\,\varphi \Rightarrow \mathbf{B_i}\,\psi)$	Distribution of $\mathbf{B_i}$
$\mathbf{B_i}\,\varphi \Rightarrow \langle \mathbf{B}_i \rangle\,\varphi$	Belief Consistency
$\mathbf{K_i}\,\varphi \Rightarrow \mathbf{B_i}\,\varphi$	Knowledge implies Belief
$\mathbf{B_i}\,\varphi \Rightarrow \langle \mathbf{K}_i \rangle\,\mathbf{K_i}\,\varphi$	Weak Evidential Restraint
From $\vdash \varphi$ and $\vdash \varphi \Rightarrow \psi$, infer $\vdash \psi$	Modus Ponens
From $\vdash \varphi$, infer $\vdash \mathbf{K_i}\,\varphi$	Necessitation of $\mathbf{K_i}$

Table 5.3: Logic of Grounded Coherent Epistemic Agents

We call it a logic of Grounded Coherent Epistemic Agents, because their beliefs are grounded by objectively available evidence, and their beliefs are coherent. Neither the belief operator nor the knowledge operator is susceptible to Löb's Obstacle, as Löb's Theorem is not derivable in the system.

In the counterexample below, we see that the belief operator lacks positive introspection.

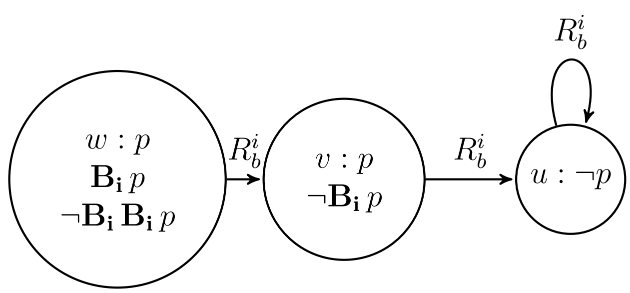


Figure 5.2: A counterexample to $\mathbf{B_i}\,\varphi \Rightarrow \mathbf{B_i}\,\mathbf{B_i}\,\varphi$.

Without positive belief introspection, Löb's Theorem for belief is no longer derivable, and therefore this (very weak) epistemic logic avoids Löb's Obstacle. The resulting logic is still in the Sahlqvist class, so it is sound and complete.

```
Theorem weak_evidential_restraint_is_sahlqvist:
  forall (phi : prop) (a : DASL.Agents),
    sahlqvist_formula
    (SB a (SProp phi) =s⇒ \ (SK a (\ (SK a (SProp phi))))).
Proof.
  intros; sahlqvist_reduce.
Qed.
```

We provide the following counterexample to Löb's Theorem.

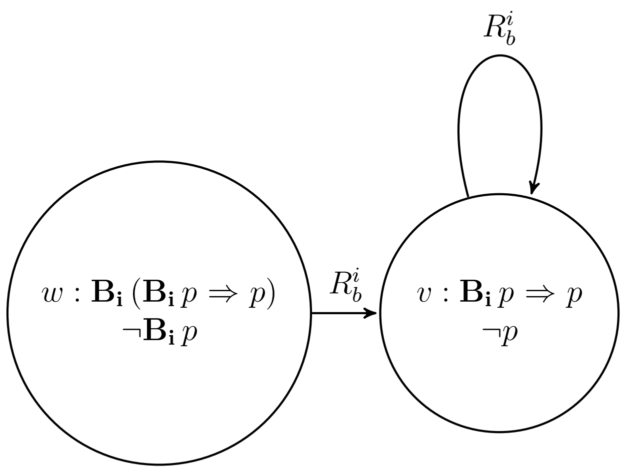


Figure 5.3: A counterexample to $\mathbf{B_i}\,(\mathbf{B_i}\,\varphi \Rightarrow \varphi) \Rightarrow \mathbf{B_i}\,\varphi$.

We assume in world w that $\mathbf{B_i}\,(\mathbf{B_i}\,p \Rightarrow p)$ holds. From this it follows that in all doxastically accessible worlds, *e.g.* v, $\mathbf{B_i}\,p \Rightarrow p$ holds. This holds for v when it has reflexive access only to itself, and $\neg p$ is the case. Because $\neg p$ is the case v, $\neg \mathbf{B_i}\,p$ is the case at w, concluding the counterexample.

The logic also includes a sort of weakened positive introspection theorem about knowledge, from the Knowledge implies Belief axiom and Weak Evidential Restraint. It is $\mathbf{K_i}\,\varphi \Rightarrow \langle \mathbf{K}_i \rangle\, \mathbf{K_i}\,\varphi$, which we read as "$i$ knows that φ only if she has objectively available evidence that she knows φ". This seems intuitive, and does not allow Löb's theorem to destroy the integrity of knowledge. It is perhaps a satisfying compromise for those who find positive introspection about knowledge to be intuitive. The contrapositive of this weak positive introspection formula, $\mathbf{K_i}\,\neg\mathbf{K_i}\,\varphi \Rightarrow \neg\mathbf{K_i}\,\varphi$ is an instance of the T axiom, so it turns out to have been a theorem all along anyway, for any epistemic logic with the Truth axiom for knowledge.

This logic allows us to make the key inference discussed in this thesis, when it serves as a static foundation for weakened $\mathcal{DASL}$.

Theorem 5.4.1 (Weakened $\mathcal{DASL}$ Inference of Safety-Critical Information)**.**

$$(\langle \mathbf{A}, a\rangle_i true \wedge \neg pre_s(a)) \Rightarrow \neg\mathbf{K_i}\,pre_s(a) \wedge \neg\mathbf{K_i}\,\neg\mathbf{K_i}\,pre_s(a).$$

Proof. Assuming $w \models \langle \mathbf{A}, a\rangle_i true$, it follows that $\mathbf{B_i}\,pre_s(a)$, from minimum rationality. From weak evidential restraint, it follows that $\langle \mathbf{K}_i\rangle\,\mathbf{K_i}\,pre_s(a)$, which is equivalent to $\neg\mathbf{K_i}\,\neg\mathbf{K_i}\,pre_s(a)$. From $\neg pre_s(a)$, it follows from the Truth axiom that $\neg\mathbf{K_i}\,pre_s(a)$. $\square$

Chapter 6

Summary and concluding remarks

In this work, we have presented a dynamic modal logic for reasoning about the relationship between an agent's beliefs, knowledge, and action, with a distinction between mere action and safe action expressible. We named this logic Dynamic Agent Safety Logic, or $\mathcal{DASL}$. Including this many modalities in one logic makes it very expressible and suitable for modelling a variety of realistic situations involving human-like agents. Additionally, the logic was developed with careful concern for realism. This departs from the typical thesis of modern lmodal ogic, which develops a logic with advanced formal properties without much regard for the realism of its foundational axiom schemas. By carefully considering the philosophical implications of our static base, the thesis made advances in reconnecting the worlds of epistemology and epistemic logic, and philosophy and modal logic more generally.

As a thesis of computer science and logic, we took care to use computational tools to validate our logical theses. We used the Coq Proof Assistant to mechanically check our proofs of soundness and completeness to a large degree, and similarly mechanized our case studies which validate $\mathcal{DASL}$'s application to aviation safety.

The proof of completeness mechanized a powerful aspect of modal logic from Sahlqvist and van Benthem. This dissertation, as far as we know, marks the first use of Coq for proving a modal logic's completeness via Sahlqvist theorem. To do this, we developed inductive predicates over the structure of axiom schemas corresponding to the definitions in Blackburn *et. al.*[15]. This two-leveled approach to mechanization allowed us to prove theorems about the logic, and then drop down into the object language and prove theorems in the logic. This approach is extensible to other domains, where an object language with different atoms could be defined in Coq, while the upper level schema mechanization remains fixed, along with its results.

We mechanically checked our formalization of three cases of aviation mishaps, zeroing in on particular moments from the events where $\mathcal{DASL}$ enables an inference from action to ignorance about safety-critical information. This inference was foreshadowed by the foundational models of agency in classical game theory, which infer good actions from assumptions of rationality and knowledge. A boolean manipulation of that classical inference yields the inference from bad action and assumption of rationality to an absence of knowledge. $\mathcal{DASL}$ provides the formal logic for capturing this inference in a rich and realistic fashion.

Further work must be done to explore this weakened $\mathcal{DASL}$. However, this dissertation is concluded.

Bibliography

[1] Artemov, S.; Nogina, E. ntroducing Justification into Epistemic Logic. *Journal of Logic and Computation* 15. 1059-1073. (2005).

[2] Federal Aviation Administration
website: https://www.faasafety.gov/files/gslac/library
/documents/2012/Nov/71574/DirtyDozenWeb3.pdf,
retrieved 10/5/2016.

[3] Rankin, William. "MEDA Investigation Process", Aero Quarterly.
Website: Boeing.com/commercial/aeromagazine/articles/qtr_2_07/aero_q207_article3.pdf,
retrieved 10/5/2016

[4] Aviation Maintenance Technician Handbook. Chapter 14, "Addendum/Human Factors".

[5] van Ditmarsch, H.; van der Hoek, W.; Kooi, B. Dynamic Epistemic Logic. Springer. Dordrecht, The Netherlands, 2008.

[6] Ahrenbach, S.; Goodloe, A. Formal Analysis of Pilot Error Using Agent Safety Logic. *Innovations in Systems and Software Engineering.* **Submitted**.

[7] Bolton, M.; Bass, E.; Siminiceanu, R. Using formal verification to evaluate human-automation interaction: A review. *Systems, Man, and Cybernetics: Systems, IEEE Transactions on* (Vol 43, No. 5, pp.488-503). 2013. IEEE.

[8] Bolton, M.; Bass, E. Formally verifying human–automation interaction as part of a system model: limitations and tradeoffs. *Innovations in systems and software engineering* (Vol 6, No. 3, pp.419-231). 2010. Springer-Verlag.

[9] Bolton, M.; Bass, E.; Siminiceanu, R. A systematic approach to model checking human–automation interaction using task-analytic models. *Systems, Man, and Cybernetics: Systems and Humans, IEEE Transactions on* (Vol 41, No. 5, pp.961-976). 2011. IEEE.

[10] Boolos, G. The Logic of Provability. Cambridge University Press. New York and Cambridge, 1993.

[11] Barras, B.; Boutin, S.; Cornes, C.; Courant, J.; Filliatre, J.C.; Gimenez, E.; Herbelin, H.; Huet, G.; Munoz, C.; Murthy, C. and Parent, C. The Coq proof assistant reference manual: Version 6.1 (Doctoral dissertation, Inria). 1997.

[12] De Wind, P. "Modal logic in Coq." M.Sc. University of Amsterdam. 2001.

[13] Feldman, R.; Conee, E. Evidentialism. *Philosopical Studies*. (Vol. 48 (1): pp 15-34). 1985.

[14] Gettier, E. Is Justified True Belief Knowledge? *Analysis* (Vol. 23, pp. 121-123). 1963.

[15] Blackburn, P.; de Rijke, M.; Venema, Y. Modal Logic. Cambridge University Press. New York, 2001.

[16] Baltag, A.; Moss, L.; Solecki, S. "The logic of public announcements, common knowledge, and private suspicions." Readings in Formal Epistemology. Springer International Publishing, 2016. 773-812.

[17] Fagin, R.; Halpern J.; Moses, Y.; Vardi, M. Reasoning about knowledge. MIT press, 2004.

[18] Basin, D., Radomirovic, S. and Schmid, L., Modeling Human Errors in Security Protocols.

[19] Butler, R.W., Miller, S.P., Potts, J.N. and Carreno, V.A., 1998. A formal methods approach to the analysis of mode confusion. In Digital Avionics Systems Conference, 1998. Proceedings., 17th DASC. The AIAA/IEEE/SAE (Vol. 1, pp. C41-1). IEEE.

[20] Hoare, C. A. R. "An axiomatic basis for computer programming." *Communications of the ACM* 12 (10): 576-583. (1969).

[21] Butler, R.; Johnson, S. Formal Methods For Life-Critical Software. *Computing in Aerospace 9 Conference*, (pp. 319-329), San Diego, CA. October 1993.

[22] Löb, M. Solution of a Problem of Leon Henkin. *Journal of Symbolic Logic* 20 (2), 115 - 118. (1955).

[23] Davis, M.; Putnam, H. "A Computing Procedure for Quantification Theory". J.ACM. 7 (3): 201–215. (1960).

[24] Rushby, J., 1993. Formal methods and the certification of critical systems. SRI International. Computer Science Laboratory.

[25] Aucher, G. and Schwarzentruber, F., 2013. On the complexity of dynamic epistemic logic. arXiv preprint arXiv:1310.6406.

[26] et d'Analyses, Bureau dEnqutes. "Final report on the accident on 1st June 2009 to the Airbus A330-203 registered F-GZCP operated by Air France flight AF 447 Rio de Janeiro–Paris." Paris: BEA (2012).

[27] van Benthem, J.; Pacuit, E; Roy, O. Toward a Theory of Play: A Logical Perspective on Games and Interaction. *GAMES* **2011**, *2*, 52-86.

[28] Prior, A. Time and Modality. Clarendon Press: Oxford, UK, 1957.

[29] Kohler, R. Lords of the Fly. The University of Chicago Press: Chicago, USA, 1994.

[30] Kraus, S.; Lehmann D. Knowledge, Belief and Time. *Theoretical Computer Science* **1988**, *58*: 155-174.

[31] Kripke, S. Semantical Considerations on Modal Logic. *Acta Philosophica Fennica* **1963**, *16*: 83-94.

[32] Halpern, J. Y.; Moses, Y. Towards a Theory of Knowledge and Ignorance: Preliminary Report. In: Apt K.R. (eds) Logics and Models of Concurrent Systems. NATO ASI Series (Series F: Computer and Systems Sciences), vol 13. Springer, Berlin, Heidelberg

[33] Harel, D.; Kozen, D; Tiuryn, J. Dynamic Logic. MIT Press: Cambridge, MA, USA, 2000.

[34] Hsu, F. Behind Deep Blue: Building the Computer that Defeated the World Chess Champion. Princeton University Press: Princeton, NJ, USA, 2004.

[35] Lenzen, W. Recent work in epistemic logic. *Acta Philosophica Fennica* **1972**, *30(1)*: 1-219.

[36] Meyer, J.J. Ch.; van der Hoek, W. Epistemic Logic for Artificial Intelligence. *Cambridge Tracts in Theoretical Computer Science.* Cambridge University Press: Cambridge, UK, 1995.

[37] Moore, G.E. "Moore's Paradox. In Baldwin, T. *G.E. Moore: Selected Writings.* London: Routledge **1993**.

[38] Negri, S. Kripke completeness revisited. *Acts of Knowledge: History, Philosophy and Logic: Essays Dedicated to Göran Sundholm* **2009**, 247-282.

[39] Pnueli, A. The Temporal Logic of Programs. *Proceedings of the 18th Annual Symposium on Foundations of Computer Science (SFCS 77)*. IEEE Computer Society.

[40] Rushby, J. Using model checking to help discover mode confusions and other automation surprises. *Reliability Engineering and System Safety* **2002**, *75*, 167-177.

[41] Rushby, J. On the Interpretation of Assurance Case Arguments. *Proceedings of the Second International Workshop on Argument for Agreement and Assurance (AAA 2015)*. Springer LNCS.

[42] Rushby, J. Trustworthy Self-Integrating Systems. *Springer LNCS*, **2016**, *9581*, 19-29.

[43] Gelman, G.; Feigh, K.; Rushby, J. Example of a Complementary use of Model Checking and Agent-based Simulation. *IEEE Int'l Conf. on Systems, Man, and Cybernetics* **2013**, 900-905.

[44] Baltag, A.; Moss, L.; Solecki, S. The Logic of Public Announcements, Common Knowledge, and Private Suspicions. *TARK '98*, **1998**, 43-56.

[45] Shoham, Y.; Leyton-Brown, K. *Multiagent Systems: Algorithmic, Game Theoretic and Logical Foundation*. Cambridge University Press: New York, USA, 2008.

[46] van Ditmarsch, H.P.; van der Hoek, W.; Kooi, B.P. Dynamic Epistemic Logic with Assignment. *Proceedings of the fourth internal joint conference on Autonomous agents and multiagent systems*, **2005**, 141-148.

[47] van Benthem, J. *Logical Dynamics of Information and Interaction*. Cambridge University Press: New York, USA, 2011.

[48] van Benthem, J. Modal Logic for Open Minds. Center for the Study of Language and Information, 2010.

[49] Van Ditmarsch, H. "Knowledge games." Bulletin of Economic Research 53.4 (2001): 249-273.

[50] Fagin, R.; Halpern J.; Moses, Y.; Vardi, M. *Reasoning About Knowledge*. The MIT Press: Cambridge, USA, 2003.

[51] Harel, D.; Kozen, D. and Jerzy Tiuryn. Dynamic logic. MIT press, 2000.

[52] Hintikka, J. *Knowledge and Belief*. Cornell University Press. Ithaca, USA, 1962.

[53] Horty, J. *Agency and Deontic Logic*. Oxford University Press. Oxford, UK, 2009.

[54] Barwise, J.; Seligman, J. *Information Flow: The Logic of Distributed Systems*. Cambridge University Press: New York, USA, 1997.

[55] van Ditmarsch, H.; van der Hoek, W.; Kooi, B. *Dynamic Epistemic Logic*. Springer: Dordrecht, The Netherlands, 2008.

[56] Bass, E.; Feigh K.; Gunter, E.; Rushby, J. Formal Modeling and Analysis for Interactive Hybrid Systems. In *Proceedings of the Fourth International Workshop on Formal Methods for Interactive Systems* (FMIS 2011). Electronic Communications of the EASST. Vol 45 (2011).

[57] Rushby, J. Formalism in Safety Cases. In *Making Systems Safer: Proceedings of the 18th Safety-Critical Systems Symposium*; Dale, C.; Anderson, T.; Eds. Bristol UK, 2010, 3-17.

[58] Klir, G.; Yuan, B. Fuzzy sets and fuzzy logic. Vol. 4. New Jersey: Prentice hall, 1995.

[59] Lescanne, P. ”Mechanizing common knowledge logic using COQ.” Annals of Mathematics and Artificial Intelligence 48.1-2 (2006): 15-43. APA

[60] De Moura, L.; Bjørner, N. Z3: An efficient SMT solver. *Tools and Algorithms for the Construction and Analysis of Systems*. 2008. Springer.

[61] Maliković, M.; Čubrilo, M. ”Modeling epistemic actions in dynamic epistemic logic using Coq.” CECIIS-2010. 2010.

[62] Maliković, M.; Čubrilo, M. ”Reasoning about Epistemic Actions and Knowledge in Multi-agent Systems using Coq.” Computer Technology and Application 2.8 (2011): 616-627.

[63] National Transportation Safety Board. “Descent below visual glidepath and impact with Seawall Asiana Flight 214, Boeing 777-200ER, HL 7742, San Francisco, California, July 6, 2013 (Aircraft Accident Report NTSB/AAR-14/01).” Washington, DC Author: NTSB (2014).

[64] Palmer, Bill. Understanding Air France 447. William Palmer, 2013.

[65] Lescanne, P.; Puiségur, J. ”Dynamic logic of common knowledge in a proof assistant.” arXiv preprint arXiv:0712.3146 (2007).

[66] Aumann, R. ”Interactive epistemology I: knowledge.” International Journal of Game Theory 28.3 (1999): 263-300.

[67] Hwang, M.; Lin, J. “Information dimension, information overload and decision quality.” Journal of Informatin Science 25 (1999): 213-218.

[68] Rescher, Nicholas. “Epistemic Logic.” University of Pittsburgh Press. 2005.

[69] Sahlqvist, Henrik. “Completeness and Correspondence in the First and Second Order Semantics for Modal Logic.” In *Proc. of the Third Scandinavian Logic Symposium*; Kanger, S.; Ed. Oslo, Norway, 1975, 110 - 143.

[70] Simpson, C.; Prusak, L. "Troubles with Information Overload = Moving from Quantity to Quality in Information Provision." International Journal of Information Management 15 (1995): 413-425.

[71] Smullyan, R. "Logicians who reason about themselves" In *Proceedings of the 1986 conference on Theoretical aspects of reasoning about knowledge*, Morgan Kaufmann Publishers Inc., San Francisco (CA), 1986, 341–352.

[72] Verbrugge, Rineke (L.C.) (5 April 2017). "Provability Logic". The Stanford Encyclopedia of Philosophy. Retrieved 18 June 2019.

[73] von Wright, G.H.. "An Essay on Modal Logic." Amsterdam: North-Holland Publishing Company. 1951.

[74] Williamson, T. Modal Logic as Metaphysics. Oxford Universty Press: New York, USA, 2013.

[75] Wittgeinstein, L. "Philosophical Investigations." Blackwell Publishers. 1953.

VITA

This is a summary of your *professional* life, and should be written appropriately. This can be written in the following order: where your where born, what undergraduate university you graduated from, if you received a masters, and which institution you graduated from with your PhD (University of Missouri). You can describe when you began research with your current advisor.

In another paragraph, you could say if/when you were married, what the name of your kids are, and what your plans are for after graduation if you choose. Take a look at other vita's from other dissertations for examples.