On the Projection Problem in Active Knowledge Bases with Incomplete Information

Von der Fakultät für Mathematik, Informatik und Naturwissenschaften der Rheinisch-Westfälischen Technischen Hochschule Aachen zur Erlangung des akademischen Grades eines Doktors der Naturwissenschaften genehmigte Dissertation

vorgelegt von

Vaishak Belle, M. Sc.

aus

Mangalore, Indien

Berichter: Prof. Gerhard Lakemeyer, Ph.D.
Prof. Hector J. Levesque, Ph.D.

Tag der mündlichen Prüfung: 8. Juni 2012

Abstract

The problem of projection has been identified as a fundamental reasoning concern in dynamical domains, where we are to determine whether or not some conditions will hold after a sequence of actions has been performed starting in some initial state. Solving the problem requires, at the very least, effectively reasoning about how actions transform the world, and inferring the logical consequences of the initial knowledge base (KB). For various reasons, tractability one of them, applications often make the closed-world assumption, thereby limiting the scope of these systems for the real world.

In this thesis, using the language of the situation calculus, we investigate the computational properties of a number of unsolved reasoning tasks in the context of projection with incomplete information. We first look at inherently incomplete KBs, where the information provided to the agent may not determine every fact about the world. Projection, then, may involve reasoning about what is believed and also, about what is not believed. We then look at physical agents with unreliable hardware, as a result of which actions lead to certain kinds of incomplete knowledge. Intuitively, beliefs should be (periodically) synchronized with this noise. Finally, we consider the presence of other agents in the environment, whose beliefs may differ arbitrarily, and the formalism should incorporate what others sense and learn during actions.

To enable a precise mathematical treatment of incomplete KBs, we appeal to a seminal proposal by Levesque, called only knowing. Building on existing work, we investigate projection wrt extensions to the situation calculus for only knowing, noisy hardware and multiple agents. Our central contribution will be to show that, in spite of the additional expressivity, reasoning about knowledge and action reduces to non-epistemic non-dynamic reasoning about the initial KB. More precisely, we show that when the initial KB is an arbitrary first-order theory, we are able to identify conditions under which projection can be solved by progressing the KB to a sentence reflecting the changes due to actions that have already occurred. Moreover, when effectors are unreliable, we allow the system to maintain probabilistic beliefs and then show how projection can be addressed by means of updating these beliefs. Finally, when there are many agents in the picture, we show that queries about the future can be resolved by regressing the query backwards to a formula about the initial KB. Only knowing comes with a significant result that allows us to reduce queries about knowledge to first-order theorem-proving tasks, which is then made use of when solving projection.

Zusammenfassung

Das Problem der Projektion wurde als fundamentale Eigenschaft in dynamischen Domains erkannt, wobei die Aufgabe darin besteht, zu bestimmen, ob einige Bedingungen nach der Ausführung einer Aktionssequenz, ausgehend von einem initialen Zustand, anschließend weiterhin gelten oder nicht. Um das Problem zu lösen ist wenigstens effektives Folgern darüber nötig, wie Aktionen die Welt verändern, sowie das Ziehen von Rückschlüssen ausgehend von der initialen Wissensbasis. Aus verschiedenen Gründen, wie unter anderem der Berechenbarkeit, wird in Anwendungen oft die Closed-World Assumption angenommen, wodurch die Einsetzbarkeit dieser Systeme in der realen Welt eingeschränkt wird.

In dieser Arbeit verwenden wir die Sprache des Situationskalküls um die rechenbetonten Eigenschaften einer Reihe von ungelösten Schlussfolgerungsaufgaben im Kontext der Projektion mit unvollständigen Wissensbasen zu untersuchen. Zuerst betrachten wir inhärent unvollständige Wissensbasen, wobei die dem Agenten zur Verfügung gestellten Informationen nicht jeden Fakt über die Welt abbilden können. Projektion kann dann beinhalten zu folgern, was und was nicht angenommen wird. Dann schauen wir auf physische Agenten mit unzuverlässiger Hardware, durch die Aktionen zu bestimmten Arten von unvollständigem Wissen führen können. Intuitiv sollten Annahmen periodisch synchronisiert sein mit diesem Rauschen. Schließlich betrachten wir die Anwesenheit anderer Agenten in der Umgebung, deren Annahmen sich willkürlich unterscheiden können. Der Formalismus sollte Sensorwahrnehmungen von anderen Agenten mit einbeziehen und von ihnen während der Aktionsausführung lernen.

Um die exakte mathematische Behandlung von unvollständigen Wissensbasen zu ermöglichen knüpfen wir an die wegweisende Arbeit von Levesque an, welche bekannt ist als “Only Knowing”. Aufbauend auf dieser bestehenden Arbeit untersuchen wir Projektion in Bezug auf Erweiterungen des Situationskalküls für Only Knowing, verrauschte Hardware und multiple Agenten. Unser zentraler Beitrag wird es sein zu zeigen, dass – ungeachtet der gesteigerten Ausdrucksstärke – das Schlussfolgern über Wissen und Aktionen sich auf nicht-epistemisches, nicht-dynamisches Folgern über die initiale Wissensbasis zurückführen lässt. Genauer gesagt zeigen wir, dass wir, wenn die Wissensbasis eine beliebige Logiktheorie erster Ordnung ist, Bedingungen bestimmen können unter denen Projektion durch Progression der Wissensbasis lösbar ist, welche die änderungen durch bereits geschehene Aktionen darstellt. Ferner erlauben wir dem System wahrscheinlichkeitstheoretische Annahmen wenn Effektoren unzuverlässig sind und zeigen wie Projektion als Aktualisieren der Annahmen verstanden werden kann. Schließlich zeigen wir für den Fall mehrerer Agenten im Szenario, dass Anfragen über die Zukunft durch Regression der Anfrage zurück auf eine Formel über die initiale Wissensbasis beantwortet werden können. Only Knowing beinhaltet ein bedeutendes Resultat, dass es uns erlaubt, Anfragen über Wissen auf das Beweisen von Logiktheoremen erster Ordnung zurückzuführen, welche dann Projektion zur Lösung verwenden können.

To Family

Acknowledgements

Completing this Ph.D. has been an enriching journey. What follows is my humble attempt to thank those who have been a part of this ride, through its many potholes.

Let me begin by expressing my sincere gratitude to Professor Gerhard Lakemeyer, for first encouraging me to find my own path, and then offering valuable suggestions to realize my goals. My previous background was in machine learning, and I imagine my initial intuitions about epistemic logic must have seemed peculiar. In spite of that, Gerhard patiently helped me fix my technical knowledge, allowed me to explore tangential ideas freely, and more importantly, shared his thoughts on what makes a problem interesting. He has overseen my academic growth for the past four years, and his views on mathematical logic and artificial intelligence have greatly influenced my own. For all of this, for showing me how to write scientifically, and for your continued guidance and support: thank you.

A special thanks also goes to my external examiner Professor Hector Levesque for painstakingly reading this thesis, and bringing significant bugs and typographical errors to my attention.

Carrying out my work at the Knowledge-based Systems Group has been a lot of fun. I thank its members for great corridor and coffee conversations. Participating in soccer matches, and doctoral seminar escapes to picturesque countrysides, with the remainder of Informatik 5 was very exciting, and I hope these traditions are here to stay. I also thank Jens Claßen for technical discussions, and Tim Niemüller for offering to translate the abstract of this thesis.

A number of people visited the department over the years, and among them, I would especially like to thank Yongmei Liu for insightful conversations on research. She also had shared her intuitions about "forgetting" with me, which prompted me to explore that topic. I thank her for this.

Financial support for the years 2008–2010 was gratefully received by the Deutsche Forschungsgemeinschaft (German Research Foundation), which funded the graduate school *Software for mobile communication systems*, of which I was a stipendiary. During my time at the graduate school, Prof. Otto Spaniol provided constant encouragement, and supported our participation in Dagstuhl seminars. For this, I am thankful. Further, Prof. Wolfgang Thomas encouraged me to present my results to the *AlgoSyn* graduate school, and for this, I thank Wolfgang. For much of the year 2011, financial support was received by the *B-IT* research school, and for this I express my gratitude to Gerhard and Prof. Matthias Jarke. For the remainder of 2011, financial support was provided by the Knowledge-based Systems Group, and for this, I thank Gerhard.

Besides the papers, the conferences and the university, my life has been made that much more enjoyable thanks to a group of friends, spread across the globe, many of whom I visited and many of whom visited me during my life at Aachen. To all of them, I wish to say a big heartfelt thanks. My mom has been a powerhouse, understanding my choice of career, and understanding what gets me excited about research. I am eternally grateful for all this support.

This leaves one last thing: for the last years, I have chosen to share my life with one special person, a person of wit, of kindness, and more importantly, of love. A person who has delighted me with her bright smile during the good days, and shouldered my sadness on the not-so-good days. A person who shares with me the most seminal of human qualities: curiosity. We have traveled extensively over the last years, and there was not a single place, not a single time, when I did not feel complete connectedness in what we are looking for. I express my sincere love to my wife Sukanya.

Contents

Chapter 1

Introduction

Broadly speaking, artificial intelligence (AI) is concerned with building *agents* that are capable of *intelligent behavior*: an agent is any entity that perceives and acts in its environment, and intelligent behavior is the choosing of actions that are appropriate as a function of some current set of beliefs about the world. To achieve general-purpose and open-ended intelligent behavior, however, conventional programming seems restrictive. That is, there is a need to make behavior depend on explicitly represented propositions, which describe features about the world in an abstract way, together with the (computational) ability to reason about these propositions. Therefore, knowledge representation and reasoning (KR), which is the field of AI that investigates the modeling and manipulation of knowledge that an agent or a system exhibits, plays a fundamental role. The problem we examine in this thesis is described as follows.

1.1 The Problem

In this thesis our efforts are directed towards the knowledge representation and reasoning problems faced by an autonomous agent, such as a robot, operating in a dynamic and incompletely known environment [Levesque and Reiter, 1998]. At the outset, the agents are assumed to be *cognitive* in the sense of having cognitive capabilities such as memory and perception, and they are *long-lived* in the sense of functioning autonomously for extended periods of time. We are not, however, interested in engineering robot controllers that solve a specific class of problems. Therefore, central to this effort is:

1. a clear understanding of the relationship between the beliefs, the action and the perception of the agent; and

2. the ability to represent the current state of the world and its dynamics as a formal system, which allows us, among other things, to *query* the system about properties that are *true* in the world presently, as well as in the future.

To a first approximation, these characteristics encourage a *high-level* control of the system behavior, differing somewhat from traditional robotics (and such), in that it emphasizes decision making as an outcome of

the agent's cognition.[1] This requires, for example, determining how much of the knowledge[2] of the agent is compatible with reality, and describing the way in which actions change the world. Wanting to deal with truth-preserving operations over formal representations puts us in the domain of mathematical logic. By representing the agent's beliefs about the world, together with the inherent physics of the domain, as sentences in a language with a truth theory, which constitute the so-called *knowledge base* (KB), properties about the domain can, then, be logically deduced. Thus, the general idea will be to provide a theoretical and computational account for a kind of deliberation that involves reasoning about action and change which has formal logic as the underlying mathematical representation for dynamical worlds and the agent's beliefs about them.[3]

Modeling a domain in this way has a number of advantages, of course. For instance, the problem of *classical planning* is that of finding a sequence of actions that, after execution, will transform the world and enable properties so as to reach a state satisfying articulated goal conditions. Think of having a robot and wanting it to achieve some goal. Instead of programming it directly, we allow the robot to use what it believes about the world initially and the actions at its disposal to figure out what to do to achieve the goal. This, then, has the desirable effect that if something changes about the world, as a result of the agent's own actions or some exogenous event, it will not be necessary to reprogram the robot.[4] We reap similar benefits when considering high-level control programs by means of agent programming proposals for complex applications, such as a mail delivery robot or non-player characters in computer games [Levesque et al., 1997]. The paradigm in this case is to allow a modeler to write very powerful programs, with usual constructs such as recursion and concurrency, but whose *primitive statements* are actions that an agent can perform. This, then, provides a way to control (and filter) the kinds of plans generated, while also allowing us to address applications whose complexity goes well beyond the range of automated planning [Levesque and Reiter, 1998].

Both tasks, among other reasoning concerns in dynamic domains, can be interpreted in terms of a more fundamental problem: the problem of *projection.* Simply put, the (temporal) projection problem, as generally encountered in AI, is the following: given a set of logical sentences modeling the domain (or a KB) Σ, a formula ϕ and sequence of actions σ, decide whether or not

$$\Sigma \textit{ entails } \phi \text{ after } \textit{performing } \sigma. \tag{1.1}$$

There are at least two sources of difficulty with this entailment. Besides the fact that agents have to reason about how actions transform the world, the initial knowledge base, which describes what the agent knows initially, is usually as expressive as a general first-order theory where the entailment problem is undecidable and comes with many intractability results [Boerger et al., 1997]. Owing to the very basic nature of the problem, however, projection has received significant attention in the literature. To get around the first difficulty, two conceptually different ways to solve projection have been pursued; either by transforming the query ϕ wrt to the action sequence σ to a "static" query about the KB, which is referred to as the *regression* methodology, or

[1] The assumption, then, is that a tight coupling of the high-level control program and other parts of the system's software will be achieved.

[2] We use the term "knowledge" and "belief" interchangeably, as we do not treat them as distinct propositional attitudes. In other words, we do not insist that knowledge is necessarily true.

[3] This view corresponds to the research agenda of *cognitive robotics* [Reiter, 2001; Lakemeyer and Levesque, 2007].

[4] Having an architecture where the behavior of the system can be altered by changing its beliefs is termed *cognitive penetrability* [Pylyshyn, 1986; Levesque and Lakemeyer, 2001].

by transforming the KB itself wrt σ to obtain a new theory that reflects the current situation, which is referred to as the *progression* methodology. To get around the second difficulty, it is quite common for applications to assume that the initial KB satisfies additional constraints such as domain closure, unique names and the closed world assumption [Reiter, 1977] in which case the theory behaves as a relational database [Abiteboul et al., 1995]. Intuitively, this amounts to assuming that the agent can infer everything about the domain and so, has *complete* knowledge about it.

In this thesis we are interested in a variety of more difficult and unsolved reasoning tasks in the context of projection when the knowledge of the agent is *incomplete*. To this end, we identify three major sources of uncertainty in beliefs:

Inherent Incompleteness: The information given to the agent may be inherently incomplete, in the sense that the sentences in its KB do not determine every fact about the world. This is quite a natural occurrence, such as the world that people live in.

Usually in the literature, when an agent has a model of the world, in the form of a KB, then it is often assumed that the KB represents what the agent believes about the world. That is, there is no explicit notion of knowledge at all, and this is sufficient, when we are only interested in the logical consequences of the KB, as in, say, (1.1). But the instant we allow knowledge bases to be incomplete, the logical language should include an explicit notion of knowledge. To see why, think of having a robot that is attempting to call a person. The person has a telephone number by assumption. If the robot does not know the number, it must attempt to look it up in the telephone directory. That is, only by knowing that it does not know the number the robot should attempt the look up. This is a general principle with *perception*, where the robot decides whether or not to sense based on what it does *not* know. Sensing actions are also primarily different from other kinds of actions, of course, since they affect the mental state of the agent, and do not affect the world in any way.

But having a notion of knowledge or belief raises questions of its own. One major concern is what properties truly characterize knowledge. Within the field of AI, it is most common to find reasoners capable of at least *positive* and *negative* introspection.[5] Full introspection, in fact, is so important because it allows the agents to have beliefs about their own incomplete picture of the world, which then allows them to take appropriate actions, as in the case of a robot figuring out when sensing is necessary. Be that as it may, it is clear that an explicit representation of the entire set of beliefs and non-beliefs about a domain is not a very concise specification. It should be possible to say that a given set of logical sentences is *all that is believed*, which would then allow the system to generate its meta-beliefs purely by deduction or introspection.

Presence of Other Agents: When there are multiple agents in the picture, which perhaps also function autonomously, then there is uncertainty regarding what the others believe and what actions they will do next. With a few exceptions, most of the work in the area of action and perception is restricted to the single agent case. But at the heart of any analysis of a conversation, a protocol, or a friendly card game, is the interaction of multiple agents. When many agents are involved, of course, an agent has to not

[5] While arguments have been provided for the appropriateness and inappropriateness of full introspection [Lenzen, 1978; Fagin et al., 1995], it turns out that full introspection can often be given a simple formal treatment. For that reason, among others, fully introspective reasoners are used most often in the kind of applications we have in mind [Lakemeyer and Levesque, 2011].

only consider the state of the world, but also reason about the mental life of the other agents.

Unreliable Hardware: In theory, a model of the world that accounts for the agent's actions is all that is needed to effectively reason about the current and future states. In practice, things are often different, where both the effects of the action and the information returned from the sensors are subject to error. Without a principled way to reason about this noise, the agent will not be able to operate in its environment, or manipulate it, in any purposeful manner. Consider, for example, a robot moving towards a wall. Suppose it executes an action representing a forward motion. Even if, due to inaccuracies in its effectors, the robot is unable to precisely determine by how much distance it has moved ahead, its belief that it is closer to the wall should, nevertheless, increase.

Each source of incompleteness[6] extends the class of projection queries considered. When there is inherent incompleteness in the KB, we must be able to ask introspective queries about what is believed and also, about what is not believed. When there are multiple agents in the domain, we must be able to ask what others believe, as well as how these beliefs evolve as a result of sensing actions. When there is noise in the executability of effectors, beliefs must be synchronized with the unreliability of actions.

1.2 The Approach

Our approach for providing effective solutions to projection in incomplete and *active*[7] knowledge bases is based on the *situation calculus* [McCarthy and Hayes, 1969; Reiter, 2001], which is one of the most influential formalisms to reason about action and change. We will be proposing extensions for knowledge, multiple agents and noise, and our central contribution will be to show that, in spite of the additional expressivity, reasoning about knowledge and action reduces to non-epistemic non-dynamic reasoning about the initial knowledge base. So to solve the projection problem in the presence of the various sources of uncertainty listed earlier, first-order reasoning will be sufficient.

The usage of the situation calculus is in terms of a special kind of logical theory called a *basic action theory*, built on the situation calculus vocabulary that includes the initial knowledge base and a set of sentences that describes the dynamics of the world. In the framework of the situation calculus, and in the context of basic action theories, Reiter [1991] proved that that the projection problem can be solved by regressing the query, and later Lin and Reiter [1997] proved that alternatively, one can progress the initial knowledge base. The methods are natural duals. On the one hand, progression has a number of obvious advantages over regression, particularly in the case of long-lived agents that has performed thousands of actions in its lifetime where processing goal conditions back through this entire sequence is not practically viable. On the other, progression is geared to answer questions about the current situation only, which means that any historical information about what held in the past is essentially lost. More importantly, progression comes with a strong negative result [Lin and Reiter, 1997; Vassos and Levesque, 2008] that it is not computationally feasible in general, in the sense that sometimes progression requires second-order logic. Therefore, when considering a

[6] There is yet another kind of incompleteness known as *deductive* incompleteness, which arises when an agent is unable to infer a fact even though it is a logical consequence of the KB. This category of incompleteness is not the focus of this thesis.

[7] By "active" we mean dynamical systems that are capable of performing world-transforming actions and capable of sensing (by way of which the system obtains more information about the world).

progression-based solution to the projection problem, it is important to identify the conditions under which progression is both first-order definable and computable.

While there have been proposals to expand the vocabulary of the situation calculus for representing knowledge [Moore, 1985a; Scherl and Levesque, 2003], even in the multiagent case [Shapiro et al., 2002], we argued that simply dealing with a set of sentences that an agent supposedly believes does not quite capture what we intuitively understand by a *knowledge base.* A knowledge base, in our view, should be all that an agent knows. This not only implies believing certain sentences, but it also implies not believing others. By introspection, then, it should come out that the agent believes that the others are not believed. This idea can be traced back to a seminal proposal by Levesque [1990], who was among the first to identify and formalize the concept of *only knowing.* In contrast to a number of other proposals attempting to characterize a similar notion [Halpern and Moses, 1984; Moore, 1985b], by the use of various meta-logical notions including fixed-point operators, the logic of *only knowing* $\mathcal{OL}$ as considered by Levesque is a very intuitive extension to classical modal logic [Chellas, 1980]. Moreover, one desirable feature of only knowing is that reasoning about beliefs and non-beliefs can be *reduced* to first-order reasoning by means of an important result known as the *representation theorem* [Levesque and Lakemeyer, 2001]. For this reason, we appeal to a modal fragment of the situation calculus, the logic $\mathcal{ES}$ [Lakemeyer and Levesque, 2011], which amalgamates the model theory of $\mathcal{OL}$ and the situation calculus in a clean and natural way.

In the context of knowledge, and only knowing in particular, projection tasks can be extended in terms of the following entailment: given a basic action theory Σ that represents all that an agent knows, a sequence of actions σ and a query ϕ decide whether or not

$$\boldsymbol{O}\Sigma \textit{ entails } \boldsymbol{K}\phi \text{ after } \textit{performing } \sigma.$$

Regression, then, Lakemeyer and Levesque [2011] show, corresponds to transforming $\boldsymbol{K}\phi$ wrt σ to $\boldsymbol{K}\phi'$, which does not mention any actions, and evaluating that against $\boldsymbol{O}\Sigma$. That is, one only reasons about what is believed initially, a much simpler entailment. Progression, in a similar fashion, Lakemeyer and Levesque [2009] show, corresponds to transforming $\boldsymbol{O}\Sigma$ wrt σ to $\boldsymbol{O}\Sigma'$ against which $\boldsymbol{K}\phi$ can be evaluated. That is, one reasons about the updated knowledge base, again, a simpler entailment. Finally, by leveraging the representation theorem, reasoning about knowledge in the absence of actions can be done using standard theorem-proving techniques.

In this thesis, we continue this line of work. Concerning the three sources of incomplete information that we identified earlier, we strengthen results regarding projection where required, and propose new ones where none exist.

1.3 Contributions

The contributions of this thesis are as follows:

1. Regarding progression in $\mathcal{ES}$, as mentioned above, the new knowledge base may contain second-order sentences in general. In order to have a well-defined projection operator, we investigate cases where the new knowledge base is first-order definable and computable. In particular, we show that when the initial knowledge base is a first-order sentence, mentioning both *predicate* and *function* symbols, the following hold:

(a) If the basic action theory is *local-effect* [Liu and Levesque, 2005a], which constrains the action theory in the manner that the effects of every action is determined exclusively by the arguments of the action, then progression is first-order definable and computable. This generalizes an earlier result by Liu and Lakemeyer [2009] who show that the progression of a *function-free* finite theory wrt local-effects is first-order definable and computable.

(b) If the action theory is *normal* [Liu and Lakemeyer, 2009], which relaxes the local-effect assumption by allowing actions to have non-local effects provided that these effects always depend on facts about the world that are uniquely determined by the action itself, then progression is first-order definable and computable. This generalizes an earlier result by Liu and Lakemeyer [2009] who show that the progression of a *function-free* finite theory wrt normal actions (under similar constraints) is first-order definable and computable.

(c) If the theory is further restricted to a certain kind of disjunctive information called *proper*$^+$ *knowledge bases*, then progression wrt local-effects and normal actions is efficiently computable. This generalizes an earlier result by [Liu and Lakemeyer, 2009] who show that the progression of a *predicate-only* version of proper$^+$ KBs is first-order definable and efficiently computable wrt local-effects and normal actions.

Since deductive reasoning for this fragment is undecidable in general, we also propose a sound and complete query evaluation mechanism for a large class of queries.

(d) Under certain assumptions, progression of proper$^+$ KBs wrt *range-restricted theories* [Vassos et al., 2009], which relax the local-effect assumption by allowing the effects of actions to not necessarily depend on the action type but also be specified using information from the initial knowledge base, is first-order definable and efficient. This presents a variant of an earlier result by Vassos et al. [2009] who proved that the progression of a *database of possible closures* wrt range-restricted theories is first-order definable and efficient. In terms of relative merits between the two results, we propose progression for proper$^+$ KBs which are much more expressive than the knowledge bases considered by Vassos et al., but we make stronger assumptions about the conditions under which progression can occur. More significantly, our progression account is a very simple one, using the same techniques identified in (*a*) and (*c*).

2. In order to extend $\mathcal{ES}$ to the many agent case, we need to be clear about how only knowing works with multiple agents. Unfortunately, previous accounts to extend only knowing to the many agent case have problems [Halpern and Lakemeyer, 2001]. More importantly, it is not clear how that semantics is to be extended to a first-order language.

   In this thesis, we propose a new semantics for multiagent only knowing for a quantified language. We show that it generalizes the model theory of $\mathcal{OL}$ in a natural way. For the propositional fragment, we also characterize the semantics with an axiomatization that faithfully lifts Levesque's axiomatization for the propositional fragment of $\mathcal{OL}$.[8] We also establish the precise relationship between our approach and previous proposals.

[8] Levesque's axiomatization for the full language has been shown to be incomplete [Halpern and Lakemeyer, 1995]. In fact, it is also shown that any complete axiomatization cannot be recursive.

3. Based on these results, we propose an extension of $\mathcal{ES}$ to the many agent case. More importantly, we prove a regression property by means of which multiagent beliefs after actions can be reduced to multiagent beliefs about the initial state. We then generalize the representation theorem, which when coupled with regression allows us to solve projection tasks strictly using a first-order theorem prover.

4. We then expand the language of $\mathcal{ES}$ to allow for the modeling of faulty hardware. The main feature is that the nondeterminism in the effects of actions can be quantified with probabilities. To synchronize the agent's mental state with this representation, we allow the agent to maintain probabilistic beliefs. A semantics to capture this uncertainty is proposed, and we approach the projection problem by providing a computational account based on progression.

The regression results in this thesis are very general, in the sense that no restrictions on basic action theories will be necessary. However, when considering progression, it is important to identify the precise conditions under which it is computationally well-behaved. We conclude the introduction with an outline of the rest of the thesis.

1.4 Outline

The rest of the sequel is organized as below. In Chapter 2 we review the relevant background literature, and provide brief surveys of existing results. In Chapter 3, we begin by reviewing the logic $\mathcal{OL}$ (and the representation theorem) since it serves as the basis both for $\mathcal{ES}$ and the other logics considered in this thesis. This, then, allows us to identify some salient properties of its model theory, which leads to our work on multiagent only knowing. We then present an axiomatization for the propositional case, and elucidate on the relationship between our approach and the earlier ones. This concludes our work for the non-dynamic setting.

In Chapter 4, we review the logic $\mathcal{ES}$, its regression property, and its leveraging of the representation theorem. Since we consider a regression-based solution to the projection problem in the multiagent case, we continue the chapter by extending $\mathcal{ES}$ to the many agent case. We accompany that extension with a regression property and a generalization of the representation theorem.

In Chapter 5, we review the changes to the semantics of $\mathcal{ES}$ to capture the notion of progression. After considering the general second-order definition, we consider the first-order definability results for local-effects and normal actions. Next, we prove that the progression of proper$^+$ knowledge bases wrt local-effects and normal actions is not only first-order definable but also efficiently computable. After that, we consider the progression of proper$^+$ knowledge bases wrt range-restricted theories. Finally, at the end of that chapter, we propose a query evaluation mechanism for a fragment of the language.

In Chapter 6, we consider a formal theory for noisy effectors. We extend the language to include a notion of probabilistic beliefs, and then present a semantics which is inspired by the progression semantics for $\mathcal{ES}$. We then cover the foundations of progression in this new setting. With that in hand, we turn to a practical case, where we are able to define the progressed knowledge base after deterministic and nondeterministic actions.

We conclude with a summary of the thesis and a discussion of future work.

The results on multiagent only knowing from Chapter 3 appeared in [Belle and Lakemeyer, 2010a], and then reappeared in [Belle and Lakemeyer, 2011a]. Its extension to a theory of actions, including the

regression result, in Chapter 4 were published in [Belle and Lakemeyer, 2010b]. The generalization of the representation theorem from that chapter is unpublished. Preliminary versions of the computability results for the progression of knowledge bases from Chapter 5 appeared in [Belle and Lakemeyer, 2011b]. A semantical account for progression in the presence of noise from Chapter 6 was investigated in [Belle and Lakemeyer, 2011c].

Chapter 2

Relevant Literature

In this chapter, we review the relevant background literature. We begin with an overview of logics of knowledge. We then turn to the problems that arise in reasoning about action, including the projection problem. Finally, we survey various knowledge representation formalisms to reason about action and change, and in process also illustrate how projection is addressed in these formalisms.

2.1 Logics of Knowledge

As mentioned in Chapter 1, in this thesis we are interested in reasoning about knowledge in incomplete knowledge bases, which means knowing what you know and also not believing what is not known. To prepare for that, in this section, we review logics of knowledge. We begin with a brief history of the study of knowledge.

Epistemology, which is the study of knowledge, has a long tradition in philosophy, dating to the early Greek philosophers. The idea of a formal logical analysis is much more recent, however, going back at least to [Von Wright, 1951]. The first book-length treatment of *epistemic (modal) logic* is Hintikka's seminal work *Knowledge and Belief* [1962]. A model theory for modal logic was also developed independently by Kripke [1959; 1963], at about the same time.

The initial interests of philosophers was mostly restricted to settling questions such as "what is knowledge?" and "what can be known?". Over the years, researchers from a number of fields, such as artificial intelligence, distributed systems, and game theory, have found modal logic to be tremendously useful in capturing formal properties of systems exhibiting dynamic or temporal behavior. Consequently, the focus of the attention has shifted to more pragmatic concerns such as computational requirements, interactions between multiple agents, dealing with incomplete information, and so on. A comprehensive coverage of the various applications of epistemic logic and variant formal systems can be found in [Fagin et al., 1995].

The essential idea behind the semantics to modal logic, often called the *possible-world semantics*, is that besides the true state of affairs, there are a number of other possible states of affairs. Agents, by and large, may not be able to distinguish between these possibilities. An agent is said to *know* α if α is true at all the possible states or "worlds". For instance, an agent in a brightly lit hallway may or may not be aware if the coffee machine is switched on. Therefore, in all the worlds considered possible, the hallway is lit. But in a

subset of these worlds, the coffee machine is switched on, and in another subset, it is switched off.

Formally, propositional modal logic is propositional logic[1] enriched with a modal operator for knowledge, say $\boldsymbol{K}$. If α is a formula, then so is $\boldsymbol{K}\alpha$. A semantics is specified using *Kripke structures*. A Kripke structure M is a tuple $(W, \pi, \mathcal{K})$, where W is a set of worlds, π is a function that associates worlds with a truth assignment to the set of propositions in the language, and $\mathcal{K}$ is a binary relation on W, intuitively capturing the epistemic possibilities between worlds, and referred to as an *accessibility relation*. A simple Kripke structure in shown in Figure 2.1, consisting of three worlds defined over the proposition p. An arrow from the world colored white to the world colored black indicates that the latter is consider possible when at the former.

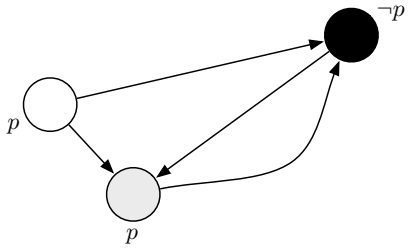

Figure 2.1: A simple Kripke structure.

Using $\models$ as the satisfaction relation, given a Kripke structure $M = (W, \pi, \mathcal{K})$, a world $w \in W$ and a formula α we define a semantics inductively as follows:

1. $M, w \models p$ iff $\pi(w)(p) = \text{TRUE}$ where p is a proposition;

2. $M, w \models \neg\alpha$ iff $M, w \not\models \alpha$;

3. $M, w \models \alpha \vee \beta$ iff $M, w \models \alpha$ or $M, w \models \beta$;

4. $M, w \models \boldsymbol{K}\alpha$ iff $M, w' \models \alpha$ for all w' such that $w' \in \mathcal{K}(w)$.

We say that α is valid ($\models \alpha$) if $M, w \models \alpha$ for every Kripke structure M and world w.[2]

It turns out that the accessibility relation has a special role to play regarding the properties of knowledge. When $\mathcal{K}$ is unrestricted, we obtain the simplest variant of epistemic logic, called **K**, which is characterized by means of the following schemas:

Axioms:

PL. All instances of axioms of propositional logic;

K. $\boldsymbol{K}\alpha \wedge \boldsymbol{K}(\alpha \supset \beta) \supset \boldsymbol{K}\beta$.

[1] See [Enderton, 1972] for an introduction to propositional logic.

[2] Note that the first three clauses in this definition correspond to the standard clauses in the definition of truth for propositional logic. The last clause appeals to the intuition examined above.

Inference Rules:

MP. From α and $\alpha \supset \beta$ infer β.

NEC. From α infer $\boldsymbol{K}\alpha$.

In the philosophical literature, a great deal of attention has been devoted to what other properties truly characterize knowledge [Lenzen, 1978]. The main contenders are **T**, **D**, **4** and **5** given in the table below. For example, axiom **T** stipulates that knowledge is necessarily true in the real world, and axioms **4** and **5** stipulate an agent who is capable of *positive* and *negative* introspection respectively. In terms of the model theory, each axiom corresponds in a precise sense to a constraint on $\mathcal{K}$. The axiom **T**, for example, together with **PL**, **K**, **MP** and **NEC** is a sound and complete for Kripke structures where the accessibility relation is *reflexive*. Going back to Figure 2.1, for example, we observe that it is a Kripke structure whose accessibility relation is clearly unrestricted, hence it is a model only for the logic **K**.

We obtain other modal logics by combining any combination of $\{\mathbf{T}, \mathbf{D}, \mathbf{4}, \mathbf{5}\}$ with the logic **K**. For example, the logic **K45** is the modal logic characterized by the axioms of the logic **K** together with **4** and **5**. (For brevity, it is sometimes called *weak* **S5**. *Strong* **S5**, or just **S5**, is an abbreviation for the addition of **T** to weak **S5**.) Typically, in the AI literature, formal systems characterized at least by **K**, **4** and **5** are most common. Interestingly, when $\mathcal{K}$ is transitive and Euclidean, as in the case of a model for **K45**, it is easy to see that $\mathcal{K}$ functions as a globally accessible set of worlds. We then imagine $\mathcal{K}$ simply as a set of worlds, thereby simplifying much of the presentation. We will appeal to this insight in the next chapter.

Axiom	Schema	Constraint on $\mathcal{K}$
K	$\boldsymbol{K}\alpha \wedge \boldsymbol{K}(\alpha \supset \beta) \supset \boldsymbol{K}\beta$	*No restriction*
T	$\boldsymbol{K}\alpha \supset \alpha$	*Reflexive*: for all worlds w, $w \in \mathcal{K}(w)$
D	$\boldsymbol{K}\alpha \supset \neg\boldsymbol{K}\neg\alpha$	*Serial:* for all worlds w, there is a w' such that $w' \in \mathcal{K}(w)$
4	$\boldsymbol{K}\alpha \supset \boldsymbol{K}\boldsymbol{K}\alpha$	*Transitive:* if $w' \in \mathcal{K}(w)$ and $w'' \in \mathcal{K}(w')$ then $w'' \in \mathcal{K}(w)$
5	$\neg\boldsymbol{K}\alpha \supset \boldsymbol{K}\neg\boldsymbol{K}\alpha$	*Euclidean:* if $w' \in \mathcal{K}(w)$ and $w'' \in \mathcal{K}(w)$ then $w' \in \mathcal{K}(w'')$

Extensions to Multiple Agents

Kripke structures have a natural extension in the many agent case. The idea is to simply entertain an accessibility relation for each agent. That is, define a *Kripke structure for n agents* as the tuple $(W, \pi, \mathcal{K}_1, \ldots, \mathcal{K}_n)$ where W and π are as before, and $\mathcal{K}_i$ is the accessibility relation reflecting i's epistemic state. A semantics is proposed in an analogous manner:

1.-3. as before;

4. $M, w \models \boldsymbol{K}_i\alpha$ iff for all $w' \in \boldsymbol{K}_i(w)$, $M, w' \models \alpha$.

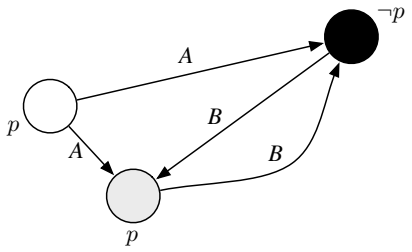

Figure 2.2: A simple multiagent Kripke structure.

For example, Figure 2.2 is a simple Kripke structure for two agents A and B, consisting of three worlds defined over the proposition p.

The axioms are generalized to the many agent case in an obvious way. For instance, the generalization of **T** to the n agent case is:

$$\mathbf{T}_n.\ \boldsymbol{K}_i\alpha \supset \alpha$$

which, in terms of a model theory constraints $\mathcal{K}_i$, for every i, to be reflexive. By extension, the n agent generalization of a modal logic, say **K45**, is obtained by considering the n agent generalization of $\mathbf{K}, \mathbf{4}, \mathbf{5}$, along with **PL**, **NEC** and **MP**, and is referred to as $\mathbf{K45}_n$.

It is worth noting that, unlike the single agent case, when considering structures for $\mathbf{K45}_n$ we cannot simplify $\mathcal{K}_i$ to a set of worlds because this leads to mutual introspection [Lakemeyer, 1993].

First-Order Modal Logic

Just as the syntax of propositional modal logic is obtained by enriching propositional logic with modal operators, we get the syntax of first-order modal logic by enriching a first-order language with modal operators. The semantics is typically specified using *relational Kripke structures*. The idea is associate each world with a relational structure.[3] By and large, this is the result of combining the semantics of first-order logic and modal logic in a straightforward way.

First-order modal logic is significantly more expressive than either first-order logic or propositional modal logic. One powerful feature is the ability to make distinctions between *knowing who* and *knowing that*. For instance, I may not know who John's teacher is, but I may know that someone teaches John.

A number of additional complexities arise in the first-order case. One of the major arguments is insisting that the domains vary across the worlds, which intuitively corresponds to what does and what does not exists may vary from world to world. However, this intuition seems to lead to considerable problems [Kaplan, 1968]. This problem is alleviated by insisting on a fixed domain for all the relational structures considered in the model, and allowing for the properties of objects to change between worlds. We do not give the details here, but note that the technical material presented in the subsequent chapters are based on a first-order modal logic appealing to the latter idea.

[3] We assume that the reader is familiar with the semantics for first-order logic. Both [Smullyan, 1995] and [Enderton, 1972] offer excellent introductions.

2.2 Problems in Reasoning about Action

To prepare for our work on dynamical systems, in this chapter we present various representational and computational issues that arise in reasoning about action.

One of the earliest examples of using a logical formalism to facilitate the axiomatization of action and their effects is the *situation calculus* (to be introduced shortly), as conceived by McCarthy [1968]. Since then a number of different knowledge representation formalisms have been suggested for the representation and reasoning of actions. They all, however, share a number of fundamental problems, the main ones being:

1. the *frame* problem [McCarthy and Hayes, 1969];

2. the *projection* problem [Reiter, 2001].

The frame problem arises in a theory of action when formulating the changes to the world after executing an action. An action results in a few changes, but a number of properties remain unaffected. For instance, moving an object causes its location to change but it does not, however, change its color. The problem is that actions typically have fewer effects than *non-effects*. Moreover, for realistic domains, the axioms that characterize the invariants in the domain, referred to as the *frame axioms*, may be colossal in number. So explicitly representing all the non-effects is both cumbersome and unintuitive. Thus, the frame problem captures the *requirement* that a representational formalism models the dynamics of the world, in the sense of the effects and non-effects of actions, in a *concise* way.

Now, the intended use of a theory of action is that a domain expert axiomatizes the dynamic aspects of an application, and then a reasoning mechanism is employed to decide which properties of the formal system hold, both initially and after a sequence of actions. The *projection problem*, then, captures the *requirement* that the formal system gives *correct* answers regarding what the world looks like after any sequence of actions. For example, in a blocks world scenario where A is on B and B is on C, we may want to determine whether moving A and then B frees C so that it can be moved to another location. In the context of logical theories, projection is cast in terms of the logical entailment relation.

Major AI applications can be interpreted in terms of the projection problem. For example, the area of planning is concerned with producing a sequence of actions in the manner that the initial theory entails a goal after the action sequence is performed. The task of verifying if the goal is satisfied is essentially projection. In the case of agent programming [Levesque et al., 1997], we are often interested in the legality of a program, which may involve tests and loop constructs, after some sequence of actions. Then evaluating test conditions is essentially projection. Finally, by formulating a database as an initial theory and database transactions as actions, asking a query wrt a database transaction is essentially projection [Reiter, 1992].

Early work on logical formalisms to address action and change focussed on the *representational* concerns about having a concise logical theory with the correct properties. However, recently, the focus has shifted to the *computational* aspect of it. That is, we are interested in practical procedures that allow us, given a formal system, to reason about the state of the world after performing actions in a manner as efficiently as possible. To that end, two important techniques are known in the literature: *regression* and *progression*.

2.2.1 Projection by Regression and Progression

In the simplest case, a projection task asks about what is true initially, and in this sense, the projection problem reduces to the query evaluation problem about the initial knowledge base. *Regression* is a computational idea that aims to reduce queries about the future also to a query about the initial situation. While the technical nature of regression varies among formalisms, the general idea is to reason backwards, that is, the property about the future is reduced in an iterative manner beginning with the last action performed. Regression is not a new concept [Waldinger, 1977], and forms the basis of a number of planning algorithms [Waldinger, 1977; Reiter, 2001].

Nevertheless, it has been argued elsewhere [Lakemeyer and Levesque, 2007] that regression is not an effective tool with long-lived agents. For instance, imagine an agent who has performed a million actions to date. At some point, if it needs to determine if a certain property currently holds then projection by regression, which involves the transformation of the property wrt all the million actions, does not seem practically viable.

The natural dual of regression is *progression* which refers to updating the logical theory after actions to one that reflects the current state of the world. Therefore, a query about the future is recast as a query about the updated initial theory. At first glance, there are two obvious benefits with progression. First, multiple queries about the future can be answered without any extra overhead. Second, one imagines that an agent, during its idle time, can compute progression while doing other physical activities. But progression has its problems as well. For one thing, it is geared to answering queries about the *current* situation only, which means what held in the past is essentially *forgotten*. But more importantly, there are some negative results regarding the computation of progression in expressive formalisms [Lin and Reiter, 1997]. Progression is also not a new concept, and lies at the heart of STRIPS [Fikes and Nilsson, 1971]. Even database updates, under certain assumptions, can be viewed as progression [Reiter, 2001].

In Section 2.3, where we examine representation formalisms for reasoning about action, we provide concrete illustrations of these two techniques, wherever applicable.

2.2.2 Projection in Open and Closed Worlds

In a nutshell, both the methodologies resort to converting the problem of answering queries after actions to one without by processing the effects of actions, either by transforming the query or by transforming the initial knowledge base. Therefore, the effectiveness of these methodologies also rests on being able to evaluate queries about the initial knowledge base in an efficient manner. But this problem, by itself, is a computationally hard one. For instance, in many formalisms, the initial knowledge base is as expressive as a general first-order theory, where the logical entailment problem is undecidable and comes with many intractability results [Boerger et al., 1997].

Of course in practice, we would like to reduce reasoning about the initial KB to a much more tractable problem than ordinary logical entailment. Therefore, it is quite common for applications to assume that the initial KB satisfies additional constraints such as domain closure, unique names and the *closed world assumption* [Reiter, 1984], in which case the initial KB behaves like a relational database [Abiteboul et al., 1995]. The query evaluation problem is well-studied for relational databases; it is decidable and even tractable in many cases [Vardi, 1982, 1986, 1995; Abiteboul et al., 1995]. De Giacomo and Mancini [2004], for example, study how relational database technology can be exploited, query and update services in particular,

to reason about projection queries.

Even without relational database technology, reasoning over closed databases is easier. For example, Prolog technology [Lloyd, 1987; Reiter, 2001] allows us to infer $\neg\phi$ when ϕ does not hold, using negation as failure. Similarly, to infer $\phi \vee \varphi$ it is sufficient to infer either ϕ or φ.

But we are not always justified in assuming a closed initial database, especially in applications involving automated agents such as robots. Therefore, dealing with projection tasks in *open worlds* is an important concern. We now briefly summarize the various kinds of techniques that developed to perform projection over open worlds. Detailed discussions on some of these will be taken up in later chapters.

One interesting direction that couples the advantages of closed databases in open worlds is pursued in [De Giacomo and Levesque, 1999], where they consider a property called *just-in-time*. The idea is to allow open initial databases but at the point where query evaluation needs to be performed, they fill in required information by means of sensing so that it is *locally complete.* The assumption, then, is that there is sufficient information available, accessible to the sensors, so as to evaluate queries. De Giacomo and Levesque investigate how that assumption can lead to tractable regression-based reasoning.

Another independent proposal for reasoning about dynamical systems over certain kinds of open initial databases is that of Liu and Levesque [2005a]. They consider what they call *proper knowledge bases*, initially proposed by Levesque [1998] as an extension to databases, which can not only allow atoms to be true or false, but also allow some atoms to be unknown. They define a form of progression for these knowledge bases which is efficient and always logically sound, and under certain circumstances, also logically complete. More recently, however, logically sound and complete progression is proven to be efficient for a much larger class of logical theories in [Liu and Lakemeyer, 2009].

A few approaches for efficient projection over open wolds rely on existing literature on databases with incomplete information. This line of work was originally pursued by Vassos and Levesque [2007] and later by Vassos et al. [2009]. The idea is allow the values of instances of functions to range over a finite number of possibilities, so that, roughly speaking, an instance of the theory is then equivalent to a finite number of *possible databases.* For instance, we are allowed to express that a robot may be holding, say either block A or B. Since database technology, then, can be exploited Vassos et al. explored the efficiency of progression, under certain restrictions, for these kinds of open databases.

Besides the above results on first-order initial knowledge bases, there are a number of propositional approaches that solve projection by a kind of progression [*e.g.* Amir and Russell, 2003; Son and Baral, 2001]. Note that, propositional languages, while not expressive, have the advantage that they can consider arbitrary incomplete knowledge. For this reason, progression-based propositional approaches are quite popular in the planning community [*e.g.* Cimatti and Roveri, 2000], although they do not always provide theoretical guarantees of tractability.

2.2.3 Other Problems

Besides the frame and the projection problems, the *qualification* and the *ramification* problems have also received considerable attention.

When axiomatizing actions, one of the tasks of the domain expert is to specify the preconditions that must hold for the action to be executable. But in the most general setting, we may need to specify an impractical

number of preconditions. For instance, an agent may move forward provided that his motors are switched on. But in addition to his motors being on, one needs to ensure that there is sufficient fuel, no obstacles in front, the ground is not slippery *etc.* Thus, the number of such qualifiers, which accommodate abnormal situations, is potentially infinite. Solutions based on a nonmonotonic treatment of these abnormalities have been proposed in literature [Morgenstern and McIlraith, 2011].

The ramification problem is regarding the various indirect effects that an action may have, and nonmonotonic treatments, among others, have also been proposed as solutions [Morgenstern and McIlraith, 2011]. These problems arise in theories that have *state constraints*, which essentially relate two or more predicates in the same state. For example, putting the sprinkler on causes the lawn, and every object on the lawn, to get wet. So an axiom characterizing this dependency conveys causal information about the action of wetting the lawn, and how that results in the action of wetting the objects. Solving the ramification problem amounts to interpreting the indirect effects of actions, as formalized by the state constraints, in a manner that captures the intended interpretation while minimizing change.

As a closing remark, and as is convention, we do not insist that the robot has true beliefs, that is, its beliefs about the world may be mistaken. This, then, raises the question as to what the agent is to do when it notices discrepancies between her mental state and the external reality, especially when the agent senses a fact that is logically inconsistent with its beliefs. Obviously, some mechanism of *belief revision* [Alchourròn et al., 1985] has to be incorporated. However, addressing the revision of beliefs is beyond the scope of this thesis.

2.3 Knowledge Representation Formalisms

In this section, we review five prominent approaches to reasoning about action. Finally, we present various design principles that illustrate how action formalisms are used in the architecture of agent systems. Wherever applicable, we also illustrate how the frame problem and the projection problem are addressed.

2.3.1 The Situation Calculus

The *situation calculus* is one of the most influential formalisms for representing action and change. Originally proposed by McCarthy [1968], the version we review below is a second-order refinement developed by Reiter and his colleagues [Reiter, 2001].

The situation calculus is a many-sorted first-order language (with some second-order features), with sorts for *actions*, *situations* and a catch-all sort *objects* for everything else. Here are its main features:

1. *Actions* and *Objects*: Changes in the world are a result of executing actions. For example, *forward* may denote moving ahead by one unit and $drop(x)$ may denote the dropping of x. In addition to such actions, we include *sensing actions* that provide new information to the agent. For example, *sonar* may denote an action that senses the actual distance between the agent and some fixed location by means of a sonar or any other hardware equipment.

2. *Situations*: Situations are viewed as a possible history of actions. A special constant S_0 denotes the *initial world state*, where no actions have occurred yet. Subsequent situations are a result of executing

actions at a previous situation. In particular, a distinguished binary function *do* is used for this purpose in the manner that $do(a, s)$ denotes a situation that is a result of executing a at the situation s.

3. *Relational and Functional Fluents*: Fluent predicates are predicates whose truth value vary as a result of performing actions. That is, their values vary from situation to situation. Similarly, fluent functions are functions whose denotations vary from situation to situation. For example, a relation that denotes the status of an object in terms of whether it is broken or not, is of the fluent type. On the other hand, the function that returns the title and author of a book is not of the fluent type. Syntactically, if a formula mentions a situation term then we can take the predicate and function symbols mentioned to be of the fluent type.

In order to formally capture an application domain, a special set of axioms are formulated as follows:

1. *Action precondition axioms*: these tells us the conditions under which an action is executable, characterized using a distinguished fluent *Poss*. For example, we may have the following sentence saying that moving forwards is only possible if the robot is not already at its goal.

$$Poss(forward, s) \equiv distance(s) > 0.$$

2. *Action sensing axioms*: these capture the results of sensing, characterized using a distinguished fluent function *SF*. For example, we may have the following sentence that says on executing *sonar*, the robot learns the actual distance to the location.

$$SF(sonar, s) = r \equiv distance(s) = r.$$

3. *Effect axioms*: these characterize the changes brought about by actions. For example, we may have:

$$\begin{aligned} &distance(do(a, s)) = x \equiv \\ &\quad a = forward \wedge x = distance(s) - 1 \ \vee \\ &\quad a \neq forward \wedge x = distance(s). \end{aligned} \tag{2.1}$$

   This says that moving forward causes the robot to be closer by unit. The second clause mainly states a consistency property that ensures that if the distance is x at situation s then the action a does not change the value of the fluent *distance* unless as specified by the first clause.

   This is formulated so to incorporate Reiter's *monotonic* solution to the frame problem [2001]. Reiter's idea consists of specifying the positive and negative effects as the necessary and sufficient conditions that change the value of a fluent. In particular, it covers an aspect of the qualification problem by making the *causal completeness assumption* where it is supposed that the effect axioms characterize all the conditions under which an action can affect a fluent and these are relatively few in number. Effect axioms formulated in this manner are called as *successor state axioms* (SSAs).

The above axioms together with the initial KB, the *unique name assumption* for actions which asserts that primitive action terms are distinct from one another, and a set of domain-independent *foundational axioms* are referred to as a *basic action theory* (BAT). The details can be found in [Reiter, 2001]. The foundational

axioms essentially ensure that situations are all and only those reachable from S_0, by the execution of a finite number of actions. This is formulated using a second-order induction axiom.

In a sense, a Tarskian model for the foundational axioms (and hence, a basic action theory) is a *tree* where the root is the initial situation S_0, the branches correspond to actions, and the subsequent nodes are a result of $do(a, s)$ where s is the situation corresponding to the parent node. One such tree is depicted in Figure 2.3.

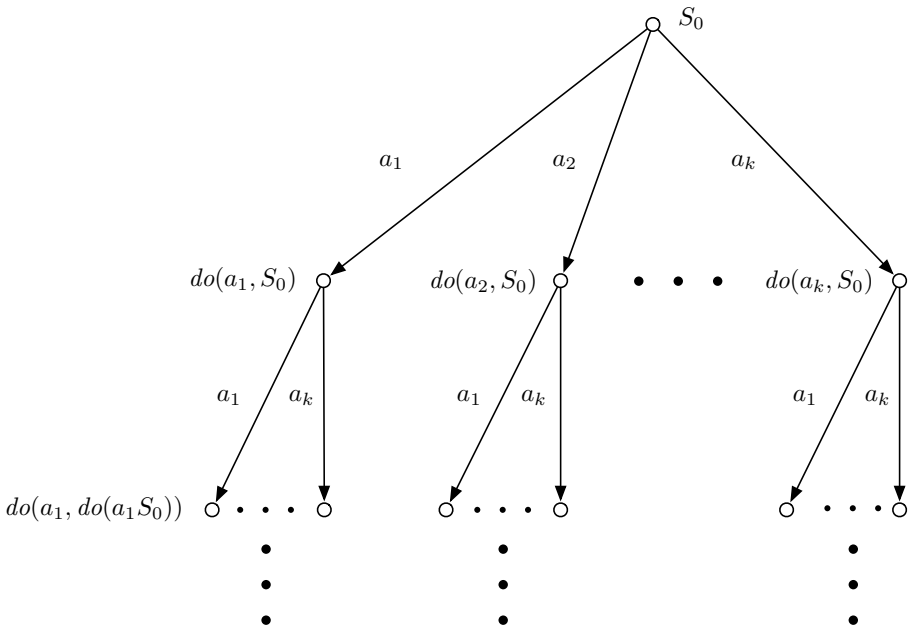

Figure 2.3: A tree of situations for a model with k actions.

The expressiveness and the generality of the situation calculus as a formalism to reason about action has led to the synthesis of a programming language called GOLOG [Reiter, 2001]. The idea behind GOLOG is to define powerful programming constructs, such as *iteration*, *tests*, *nondeterminism*, and *(while) loops*, as *macros*, which ultimately expand into situation calculus formulas. So, a GOLOG program is essentially a basic action theory.

In the context of a basic action theory, the projection problem is deciding if a formula (representing the goal) is a logical consequence (in classical FOL sense) of the basic action theory. Because of the second-order nature of the foundational axioms, simply applying a theorem prover is not optimal. To that end, one of the other main results from Reiter's [1991] paper on the solution to the frame problem is that for a large class of formulas, which Reiter calls *regressable formulas*, formulas containing action terms can be iteratively reduced by a regression property to a first-order formula about the initial theory [Reiter, 2001].

Alternatively, Lin and Reiter [1997] showed that basic action thoeries can be progressed. The problem of progression corresponds to updating the initial theory to one that reflect the situation after an action has been performed, and so the new theory and the old theory describe the future in the very same way. However, the definition of Lin and Reiter uses second-order logic. They identified two simple cases where progression is first-order definable. This was, then, used in the same paper to provide a logical basis for one of the oldest

yet still widely-used planning formalisms: STRIPS [Fikes and Nilsson, 1971]. In this formalism, the initial KB is simply an ordinary database (a set of literals for this discussion) and actions are formulated as *operators* on that database in the sense that executing actions result in the deletion of some literals and the addition of others. Thus, STRIPS can be modeled as a simple kind of situation calculus theory, and the operational nature of actions is nothing but a simple kind of progression, Lin and Reiter show.

Despite first-order definability in these cases, Lin and Reiter conjectured that no alternate first-order definition exists in general, even allowing infinite theories if necessary. Vassos and Levesque [2008] proved that the conjecture was indeed true. Since then, a number of papers have investigated a larger class of basic action theories where progression is first-order definable and (sometimes) even efficient [Vassos and Levesque, 2007; Vassos et al., 2008; Liu and Lakemeyer, 2009; Vassos et al., 2009].

An Epistemic Extension

To incorporate knowledge, Moore [1985a] observed that standard ideas from modal logic could be imported to the situation calculus by viewing situations as worlds. Of course, in contrast to modal logic, situations are *reified* in the situation calculus whereas worlds are not in modal logic.

However, if one assumes a single initial situation as above, then this is equivalent to assuming that the epistemic state only contains a single world, which is also the real world. Therefore, when incorporating knowledge, the existence of a number of initial situations is assumed. Of course, the foundational axioms need an alteration to accommodate this idea. Consequently, a model of the foundational axioms is now a set of trees. A distinguished binary fluent *Know* captures the intuitive notion of an accessibility relation in the sense that $Know(s, s')$ denotes that the situation s' is epistemically accessible from the situation s. The details can be found in Reiter [2001].

Scherl and Levesque [2003] extended Moore's formulation so as to incorporate Reiter's solution to the frame problem for the epistemic situation calculus. They also extended Reiter's regression operator to account for formulas of the form $Know(\alpha)$. Recently, Liu and Wen [2011] have proposed a notion of progression for a restricted fragment of the epistemic situation calculus.

However, since the situation calculus is characterized axiomatically, Lakemeyer and Levesque [2004] rightly observe that when we are interested in analyzing more than just the entailments of an action theory, such as properties about knowledge, the formalism is unworkable semantically. Moreover, there does not seem a straightforward way to capture only knowing in this flavor of the situation calculus [Lakemeyer, 1996; Lakemeyer and Levesque, 1998]. This led to their work on a modal reconstruction of the epistemic situation calculus called $\mathcal{ES}$ [Lakemeyer and Levesque, 2004].

An Extension for Noisy Actions and Sensors

In order to model noise in hardware effectors, Bacchus et al. [1995] introduced a variant of the situation calculus for formalizing degrees of belief and noisy actions. Their approach can be thought of as two important extensions to the epistemic situation calculus. First, the background theory includes a methodology to capture nondeterminism in actions, but which still falls back on Reiter's solution to the frame problem. Second, they introduce a companion fluent to *Know* that captures a subjective assessment of uncertainty. Formally, situations are associated with a *weight*. The weight of a situation s' is always measured relative to another sit-

uation s by means of the fluent function p. In this sense, p is analogous to *Know* because it also represents an accessibility relation. The probabilistic belief that an agent in situation s has about a certain formula ϕ is then obtained by the ratio of the weight of all p-accessible situations (relative to s) where ϕ holds, to the weight of all p-accessible situations relative to s. However, this estimation resorts to second-order logic, making the practical relevance of the formalism unclear. To remedy that shortcoming, [Gabaldon and Lakemeyer, 2007] propose an amalgamation of $\mathcal{ES}$ and uncertainty, where this calculation is instead evaluated semantically.

Unfortunately, both proposals do not consider solutions to the projection problem. That is, neither regression nor progression is extended to these proposals.

Finally, the situation calculus has also been extended to deal with issues such as time, concurrency and natural actions [Reiter, 2001], among others.

2.3.2 The Fluent Calculus

The fluent calculus [Thielscher, 1999] is a first-order formalism based on the logic programming approach of [Hölldobler and Schneeberger, 1990]. It takes much of the basic ontology of the situation calculus, that is, it also stipulates a branching time structure. The formalism inherits the action, the object and the situation sorts from the situation calculus, but also includes a new sort of terms called *states*. A state is defined as a *set* of fluent atoms and intuitively, it can be seen as a "snapshot" of the world. Formally, in addition to S_0 of the situation calculus, the fluent calculus includes an empty state $\emptyset$, a constructor $\circ$, and a function *State* that maps situations to states. States are closed under $\circ$-composition.

Interestingly, the formalism uses a STRIPS-like solution to the frame problem and a progression-based solution to the projection problem. To illustrate the idea, consider our earlier example about a robot moving towards the wall. Let *Distance* be a function that maps real numbers into a fluent: $Distance(x)$ denotes that the robot is x units away.[4] Since one needs to explicitly reason about states, which are sets of fluents as mentioned earlier, in the fluent calculus, fluents are *reified*. For example, to represent that the distance to the wall is 4 units initially, we write:

$$Holds(Distance(4), S_0) \tag{2.2}$$

where $Holds(f, s)$ is a distinguished macro to specify that a (reified) fluent atom f is true in situation s. The expression $Holds(f, s)$ expands to:

$$\exists z.State(s) = f \circ z.$$

where z is a variable of the state sort. Thus, the sentence (2.2) expands to $\exists z.\ State(S_0) = Distance(4) \circ z$, which determines the initial specification.

Unlike the situation calculus, where effects axioms are formulated as successor state axioms one per fluent, here effect conditions are encoded on the basis of the action. This is then coupled with states to obtain a dual representation of Reiter-style successor state axioms called as *state update axioms*. For instance, we may formulate the effects of the action *forward*, incorporating a solution to the frame problem, as follows:

$$\exists x, y.\ Holds(Distance(x), s) \wedge y = x - 1 \wedge$$

$$State(do(forward, s)) = (State(s) - Distance(x)) \circ Distance(y).$$

[4] Our axiomatization is adapted from [Thielscher, 2001].

That is, on moving forward when in situation s, the new state is obtained by deleting $Distance(x)$ from the state corresponding to s *viz.* $State(s)$, and adding $Distance(x-1)$. This can be compared to the successor state axioms as formulated in (2.1). It should also be clear that the intuitions regarding the dynamic world are very close in spirit to the situation calculus in the sense that if the current state corresponds to the situation s, then the new state corresponds to the situation $do(forward, s)$.

Given a fluent calculus axiomatization, which also includes foundational axioms identifying legal states and situations, the projection problem is cast in terms of an entailment after a given sequence of named actions. In a sense, the difficulty regarding projection is not much easier in the fluent calculus than in the situation calculus. However, the update rules from above lead naturally to a notion of progression of the states. This is because when an action occurs, there is precisely one state update axiom that is applicable and this axiom specifies what the next state looks given the current state and an action. But it is also worth noting that while the idea of additions and deletions to obtain a specification of the new state is conceptually intuitive, this is so only when the differences between the states amount to a finite set of literals. If the current state involves arbitrary incomplete knowledge, a representation of the new state is far more complex.

The fluent calculus has lead to the synthesis of the logic programming language FLUX, bearing somewhat the same motivations for the development of GOLOG from the situation calculus, and extensions to account for knowledge and noisy effectors have also been proposed [Thielscher, 2001].

2.3.3 The Event Calculus

In contrast to the branching time ontology of the fluent calculus and the situation calculus, the event calculus stipulates a linear time structure [Kowalski and Sergot, 1986]. This time structure is defined in terms of *time points*. There are a number of different versions of the event calculus: the original version [Kowalski and Sergot, 1986] and a simplified version [Shanahan, 1999], among others. We review the simplified version below.

Similar to the fluent calculus, fluents are also reified so as to be able to express that a fluent atom f holds at a specific time point t by means of the predicate $HoldsAt(f,t)$. Unlike the fluent calculus and the situation calculus, where a successor situation determines what the worlds may look like on performing an action, there is no distinction of between an actual event and an hypothetical event in the event calculus. That is to say, a predicate $Happens(a,t)$ specifies that action a actually occurred at t. For this reason, it is often described as a *narrative-based* formalism.

Besides the distinguished predicates $HoldsAt(f,t)$ and $Happens(a,t)$, the event calculus also includes $Initiates(f,t)$ and $Terminates(f,t)$ which denote that a fluent atom f begins and ceases to hold at time point t. The initial specification, which describes what is true at time point 0, is provided via $Initially(f)$. Finally, the predicate $Clipped(t_1,f,t_2)$ asserts that the fluent atom ceases to hold between time points t_1 and t_2.

To illustrate the idea, we first consider what the foundational axioms look like that stipulate the purpose of the distinguished predicates:

$$Initially(f) \wedge \neg Clipped(0,f,t) \supset HoldsAt(f,t).$$

$$Happens(a,t_1) \wedge Initiates(a,f,t_1) \wedge t_1 < t_2 \wedge \neg Clipped(t_1,f,t_2) \supset HoldsAt(f,t_2).$$

That is, f holds at time point t if it held at time point 0 and was not made false between 0 and t. The second expression asserts that f holds at t_2 if it was initiated at some point before, say t_1, and it has not been terminated between then and t_2. See [Shanahan, 1999] for details.

Consider, yet again, the example about a robot moving towards the wall. Representing its distance in terms of a relational fluent *Distance* here also, we may have the following:

$$\mathit{Initially}(\mathit{Distance}(4)).$$

$$\mathit{HoldsAt}(\mathit{Distance}(x), t) \supset \\ \mathit{Terminates}(\mathit{forward}, \mathit{Distance}(x), t) \wedge \mathit{Initiates}(\mathit{forward}, \mathit{Distance}(x-1), t).$$

This says that if the robot is x units away, moving forward makes $\mathit{Distance}(x)$ false while making $\mathit{Distance}(x-1)$ true. More concretely, if *forward* is performed at some time point, say 5, then if a reverse does not occur at later points t, $\neg\mathit{HoldsAt}(\mathit{Distance}(4), t)$ will continue to be true.

Arguably, the specification of the effect conditions in this manner is quite simple. Moreover, it does not seem to take the frame problem into account. This problem is solved by restricting entailments wrt Tarskian models that minimize the extension of the distinguished predicates. The entailment relation, therefore, is *nonmonotonic* and the preferred models chosen in this manner is based on the notion of *circumscription* [Reiter, 1987]. (The entailment relation may also be characterized using other nonmonotonic methodologies [Shanahan, 1999].)

The projection problem is essentially cast in terms of the nonmonotonic entailment relation wrt the domain axiomatization and the narratives (actions that have occurred). Over the years, a number of techniques have been used for automated reasoning in the event calculus, including logic programming and satisfiability solving, among others [Mueller, 2008].

We remark that regression-based and progression-based solutions are not well-explored in the event calculus. Nevertheless, certain applications of the formalism, such as database updates and planning, in addition to extensions that account for concurrency and nondeterministic effects have been explored previously (see [Mueller, 2008] and references therein).

2.3.4 The $\mathcal{A}$ Family of Languages

The action language $\mathcal{A}$, and its descendants [Gelfond and Lifschitz, 1993, 1998], are propositional languages to reason about effects via a domain specification using simple signatures of the form:

$$a \textbf{ causes } p_1, \ldots, p_k.$$

That is, when a is performed, all of p_i is made true. For example, dropping a container of fragile objects causes all of the objects to be broken.

The semantics for action theories in $\mathcal{A}$ are based on a transition system (a finite labeled directed graph), which is conceptually quite close in spirit to a situation calculus tree: edges are actions and states are action histories (provided that actions are deterministic). Typically, a distinction is made between the *description language*, which describes the transition system, and the *query language*, which queries the system for propositional assertions.

The projection problem in $\mathcal{A}$ is cast in terms of a temporal property, such as

$$p \textbf{ after } a_1, \ldots, a_k$$

or even a stronger global property, such as

$$\textbf{necessarily } p \textbf{ after } a_1, \ldots, a_k$$

that query whether p holds (and necessarily holds respectively) after the specified action sequence. The query is evaluated wrt a transition system determined from the domain description. The entailment relation is nonmonotonic here, but practical solutions do exist [Baral and Gelfond, 2005]. Moreover, the frame problem is also solved nonmonotonically.

The complexity of the projection problem in $\mathcal{A}$ is investigated in [Liberatore, 1997]. It is shown to be co-NP-complete. But when there is complete information in the initial specification, it is shown that projection is tractable. $\mathcal{A}$ has also been extended to account for knowledge and uncertainty [Son and Baral, 2001; Iocchi et al., 2009]. In particular, Son and Baral propose a kind of approximate progression to compute the belief states of the agent after actions.

2.3.5 Approaches Based on Dynamic Logic

Dynamic logic [Harel et al., 2000], and its extensions, are a family of modal representation languages that have been used for reasoning about actions and programs. We review some approaches that have been pursued to bring the intuitions of the situation calculus into dynamic logic.

In early work, Castilho et al. [1999] address how to import Reiter's monotonic solution to the frame problem in propositional dynamic logic. Propositional dynamic logic is propositional logic augmented with action operators of the form $[a]$ such that one reads $[a]\alpha$ as "after action a, α is true". The idea in [Castilho et al., 1999] is to introduce a *dependence relation* between actions and fluents as part of the domain axiomatization, quite similar to the **causes** construct in $\mathcal{A}$. Projection is then the task of checking whether the goal is a logical consequence of the initial specification and the axioms characterizing the dependence relations.

Later proposals, such as [Demolombe et al., 2003], formalize regression, as considered by Reiter, to dynamic logic. In [Herzig et al., 2000], an epistemic extension to dynamic logic is proposed for reasoning about knowledge and action. Regression is extended for the epistemic variant in [Van Ditmarsch et al., 2007; de Lima, 2007]. In particular, due to some recent developments in propositional modal logic, Van Ditmarsch et al. are able to show that reasoning about knowledge and action is not harder than reasoning in classical epistemic logic [Fagin et al., 1995]. Validity checking is co-NP-complete for epistemic logic, and therefore the complexity of projection is co-NP-complete for epistemic dynamic logic. Their idea can be generalized to a multiagent extension of epistemic dynamic logic [de Lima, 2007], and here again projection is not harder than reasoning in multiagent epistemic logic. When $n > 1$, validity checking is co-PSPACE-complete for multiagent epistemic logic, and therefore projection in the dynamic variant is also co-PSPACE-complete.

While the above mentioned approaches are propositional, there are also first-order treatments such as [Blackburn et al., 2001] and [Demolombe, 2003]. Blackburn et al. construct a version of the situation calculus in a formalism inspired by tense logic [Prior, 1967]. However, they do not consider epistemic notions. Demolombe, on the other hand, considers knowledge, and in fact, he even considers a form of only

knowing. However, he does not consider the quantification of actions which is necessary to capture Reiter-style successor state axioms even though his language is first-order. More importantly, he does not propose the equivalent of regression.

2.3.6 Final Remark: Design Principles

In this section, we briefly review the various ways in which action formalisms contribute to the architecture of automated agents. As we shall see, a number of issues arise in realistic domains.

Planning Agents

Perhaps the earliest application of action formalisms is in *classical planning* [Ghallab et al., 2004] which, as mentioned earlier, is concerned with producing action sequences that satisfy goal conditions. Plans need not always be sequential, and even early algorithms [Fikes et al., 1972] considered annotating plans with conditions that can be checked at execution time to confirm its validity as a fruitful operation. Since then, planning approaches have matured considerably. Uncertainty in the domain is often tackled by formulating *universal plans* [Schoppers, 1987], which represent all possible plans that can enable the goal, or by developing more robust techniques to deal with discrepancies that arise between the assumed and observed states of the world [Fritz, 2009].

Another independent proposal is by Levesque [1996], who considers a implementation-agnostic methodology of conditional planning that relies on the agent's sensing. The idea is interpret planning tasks as special kinds of *programs*, perhaps containing conditions and loops, that are *believed* by the agent to lead to the goal. By proposing a simple yet powerful programming language based on the theory of the situation calculus, Levesque demonstrates how these epistemic notions allow him to address problems outside the reach of classical planning, and further shows that the framework synthesizes intuitive non-sequential plans. Synthesizing non-sequential plans is also taken up in [Levesque, 2005] and [Srivastava et al., 2010], among others.

In practice, however, many planning approaches are formulated using the more restrictive notation of STRIPS [Fikes and Nilsson, 1971]. Since its expressiveness is quite limited, numerous extensions have been considered [*e.g.* Pednault, 1989; McDermott et al., 1998] which have also been found to be quite practical.

High-level Agent Programming

The paradigm of high-level agent programming is one which allows a modeler to write very powerful programs, with usual constructs such as concurrency and recursion, but whose primitive statements are actions that the agent can perform [Levesque and Reiter, 1998]. One influential proposal along this effort is the GOLOG programming language [Levesque et al., 1997], mentioned earlier in Section 2.3.1. In contrast to planning against a goal, the idea is to consider the execution of programs given a high-level program. Such programs are considerably more general than plans, involving loops and nondeterminism, where the modeler tells the agents what needs to be done in a high-level way for incompletely known worlds.[5] By this, we mean that the agent (and the modeler) may not necessarily know what the world looks like, and therefore sensing and online reactivity will be necessary to complete tasks successfully. For one thing, these programs provide

[5] Going further, in [Boutilier et al., 2000], a methodology to enable a preference over nondeterministic choices by providing *rewards* to successor situations is proposed.

a way to control (and filter) the kinds of plans that are considered. For another, the general framework has been shown to tackle applications that would be infeasible if formulated as a planning problem [Lakemeyer and Levesque, 2007]. Implementations that are *offline*, where the plan is generated before the agent begins to operate in its environment, as well as *online* ones [De Giacomo et al., 2001], where the agent senses at every step in an incremental fashion which is perhaps necessary in many incompletely known domains, are pursued.

Agent-Oriented Programming

This is another agent programming paradigm that focuses on ascribing certain mental qualities, such as desires and capabilities, to agents and so supports a *societal view of computation* [Shoham, 1993]. Thus, this line of research extends standard epistemic logic with temporal notions, among others. Shoham argues that with a precise logical theory that describes the mental states of the agents, a mechanism should be proposed that determines how these mental states lead to acting in the world. Agents are typically described in terms of what they believe, their *desires* (tasks that they would like to achieve), and *intentions* (desires that the agents are committed towards). Besides Shoham's original proposal AGENT-0, a number of other proposals have appeared in the recent years [*e.g.* Hindriks et al., 1999].

Chapter 3

Multiagent Only Knowing

Levesque is among the first to precisely capture the beliefs of a knowledge base that is capable of introspection. His proposal, the logic of only knowing $\mathcal{OL}$ [Levesque, 1990], is motivated by the observation that when one gives a formal specification of a KB in terms of a collection of first-order sentences Σ, then, intuitively, Σ is understood as all that the agent knows. Beliefs of the KB are then captured by reasoning about valid sentences of the form:

$$\boldsymbol{O}\Sigma \supset \boldsymbol{K}\alpha$$

which can be read as "if Σ is all that is known then α is also known". What is particularly interesting about the operator $\boldsymbol{O}$ is that it does not only allow us to draw conclusions about what is known but also about what is not known. For example, if p and q are distinct propositions, then both $\boldsymbol{O}p \supset \neg\boldsymbol{K}q$ and, by introspection, $\boldsymbol{O}p \supset \boldsymbol{K}\neg\boldsymbol{K}q$ come out valid. This is quite different from classical epistemic logics, as considered in the previous chapter, in the sense that if we replace $\boldsymbol{O}$ by $\boldsymbol{K}$, then neither of the two is valid. Further, $\mathcal{OL}$ is given a simple possible-worlds style semantics for a first-order language with equality, without complications of Tarski structures (variable maps, domains of interpretation *etc.*).

When the KB itself refers to the agent's beliefs, only knowing also exhibits a form of *nonmonotonicity* [Reiter, 1987]. For example, suppose we wish to tell a delivery robot that big blocks are *typically* found in the storage room. One way to capture this formally is in terms of the following assertion δ:

$$\forall x.\, \mathit{Big}(x) \wedge \neg\boldsymbol{K}\neg \mathit{At}(x, \mathit{storage}) \supset \mathit{At}(x, \mathit{storage})$$

which says that unless the robot has explicit reasons to believe that some block is not located in the storage, it is there. Now given any big block, say A, the following sentence is valid in $\mathcal{OL}$:

$$\boldsymbol{O}(\delta \wedge \mathit{Big}(A)) \supset \boldsymbol{K}\mathit{At}(A, \mathit{storage}).$$

The nonmonotonicity arises here because if we add anomalies about A, saying that it located in the mail room but not the storage, then the belief is retracted. This sort of introspective reasoning is very much in the spirit of a prominent approach to nonmonotonicity known as *autoepistemic logic* (AEL) [Moore, 1985b], and it demonstrates the ability to reason about prototypical assertions while exhibiting a certain flavor of commonsense reasoning in the presence of incomplete information.

Another category of assertions useful in this context is the *closed world assumption* [Reiter, 1987], as commonly encountered in AI and database theory [Reiter, 1984]. For instance, imagine telling the robot that it knows of every big block in the domain, which can be expressed by means of the following sentence δ':

$$\forall x.\, Big(x) \supset \boldsymbol{K} Big(x).$$

Then, $\boldsymbol{O}(\delta' \wedge Big(A)) \supset \boldsymbol{K}(\forall x(Big(x) \equiv x = A))$ is valid in $\mathcal{OL}$: the agent believes that A is the only big block. Thus, this family of assertions are useful in enabling complete knowledge about the domain. The propositional fragment of $\mathcal{OL}$ also has a sound and complete axiomatization which, among other things, allows us to formally derive such conclusions.

$\mathcal{OL}$, however, deals with a single agent. Given that in numerous applications there are several agents in the picture, it seems natural to ask whether these ideas can be extended to the many agent case. It turns out, however, that existing approaches to formalize multiagent only knowing are surprisingly complex compared to Levesque's intuitive and simple model theory. So proposing an appropriate generalization of $\mathcal{OL}$ will be the main focus of the chapter.

This chapter consists of two parts. The first part, which sets the stage for the second, introduces Levesque's logic of only knowing $\mathcal{OL}$. We then present an idea regarding how $\mathcal{OL}$ can be used to implement a KR service by means of the *representation theorem* [Levesque, 1984; Levesque and Lakemeyer, 2001]. We then present the axiomatization for the propositional sublanguage of $\mathcal{OL}$. We also quickly touch upon the relationship between $\mathcal{OL}$ and AEL.

In the second part, we present results on the many agent case. Existing approaches, besides having a number of undesirable properties, are restricted to propositional languages. We first show that a natural semantics can be given for a quantified language with equality. Next, we show that for the propositional case, a sound and complete axiomatization can be provided that faithfully lifts Levesque's proof to the many agent case. Along the way, we revisit some of the existing approaches and discuss their problems.

3.1 The Logic of Only Knowing $\mathcal{OL}$

Let $\mathcal{L}$ be a first-order language consisting of standard FOL with =. More precisely, $\mathcal{L}$ has the logical connectives $\vee$, $\forall$ and $\neg$. Other connectives are taken for their usual syntactic abbreviations. $\mathcal{L}$ also includes a countably infinite set of *standard names* (or simply *names*): $\mathcal{N} = \{{}^{\#}0, {}^{\#}1, {}^{\#}2, \ldots\}$, which is the domain of discourse. This will allow us to interpret quantifiers substitutionally, and previous arguments against substitutional interpretation notwithstanding [Kripke, 1976], greatly simplifies the technical treatment.

We now describe the expressions of $\mathcal{L}$. There are two types: *terms* and *formulas*, where terms describe the individuals of the domain, and *formulas* describe propositions, relations and properties.

All variables and names are terms. If $t_1, \ldots, t_k$ are terms, and f is a k-ary function, then $f(t_1, \ldots, t_k)$ is also a term. If a term mentions no variables, then we say that it is a *ground term*. If a ground term mentions only one function symbol, then we call it a *primitive term.* Put differently, primitive terms are of the form $f(n_1, \ldots, n_k)$, where n_i are names.

Moving on to formulas of $\mathcal{L}$, an *atomic formula* (or *atom*) is of the form $P(t_1, \ldots, t_k)$ where t_i are terms. A *ground atom* is an atom not mentioning variables and a *primitive atom* is a ground atom where every t_i is a name. But more generally, a formula is any of the following:

1. an atom;
2. $t_1 = t_2$, where t_i is a term;
3. $\neg\alpha$, where α is a formula;
4. $\alpha \vee \beta$, where both α and β are formulas;
5. $\forall x.\ \alpha$.[1]

We write α_n^x to mean that the variable x is substituted in α by the standard name n. Finally, by a *primitive equality* we mean a formula of the form $f(n_1, \ldots, n_k) = n_0$, where n_i are names. (In sum, primitive expressions mention only a single non-logical symbol.)

Levesque's logic $\mathcal{OL}$ is $\mathcal{L}$ with two epistemic (modal) operators: $\boldsymbol{K}$ and $\boldsymbol{O}$. As far as well-formed expressions of $\mathcal{OL}$ are concerned, this essentially means that we only need to include one extra formation rule for formulas: if α is a formula, then $\boldsymbol{K}\alpha$ and $\boldsymbol{O}\alpha$ are formulas too.

We now turn to a semantics. $\mathcal{OL}$ is given a possible-world semantics, where a world:

- is a set of primitive atoms, and
- is a function from primitive terms to names.

Let $\mathcal{W}$ be the set of all worlds. An epistemic state $e \subseteq \mathcal{W}$ is simply any set of possible worlds.

The possible-worlds framework of $\mathcal{OL}$ for a first-order language can be compared to Kripke structures. In particular, in contrast to Figure 2.1, we instead have a much simpler notion of an epistemic state as a set of worlds, as shown in Figure 3.1.

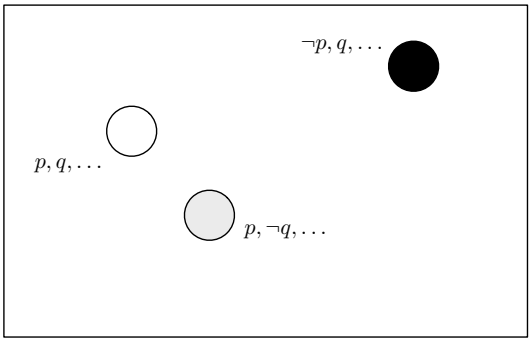

Figure 3.1: Viewing an epistemic state simply as a set of worlds. Here, p and q denote atoms.

Now, the standard names are essentially *rigid designators*, and denote precisely the same entities in all worlds. The evaluation of an arbitrary term, such as $f(g(n))$ is simply obtained wrt some world w by first obtaining the entity that $g(n)$ denotes, say n', and then obtaining the entity that $f(n')$ denotes. More formally, using $|t|_w$ to mean the "coreferring name" for the term t, let us inductively define $|\cdot|_w$ by

[1] In this thesis, we use the "dot" notation to indicate that the quantifier preceding the dot has maximum scope. For instance $\forall x.\ P(x) \supset Q(x)$ is to be understood as $\forall x[P(x) \supset Q(x)]$. Sometimes, we omit leading universal quantifiers altogether in writing sentences.

- $|t|_w = t$ if t is a name, and
- $|f(t_1, \ldots, t_k)|_w = w[f(n_1, \ldots, n_k)]$ where $n_i = |t_i|_w$.

We are ready to give a semantics. Defining a model to be the pair (e, w) with $w \in \mathcal{W}$, the truth of sentences is defined as follows:

1. $e, w \models P(t_1, \ldots, t_k)$ iff $P(n_1, \ldots, n_k) \in w$, where $|t_i|_w = n_i$;
2. $e, w \models t_1 = t_2$ iff $|t_1|_w$ is the same name as $|t_2|_w$;
3. $e, w \models \neg\alpha$ iff $e, w \not\models \alpha$;
4. $e, w \models \alpha \vee \beta$ iff $e, w \models \alpha$ or $e, w \models \beta$;
5. $e, w \models \forall x.\, \alpha$ iff $e, w \models \alpha^x_n$ for all standard names n;
6. $e, w \models \boldsymbol{K}\alpha$ iff for all $w' \in e$, $e, w' \models \alpha$;
7. $e, w \models \boldsymbol{O}\alpha$ iff for all w', $w' \in e$ iff $e, w' \models \alpha$.

Note that the only difference to the semantics for $\boldsymbol{K}$ is that an "if" becomes an "iff". We read $\boldsymbol{K}\alpha$ as "at least α is known" since $\boldsymbol{K}\alpha$ certainly does not preclude $\boldsymbol{K}(\alpha \wedge \beta)$ from holding, in general. It is easy to see that $\boldsymbol{K}$ is indeed the classical knowledge operator, and (at least) α is believed iff it is true at the worlds that the agent considers possible. We read $\boldsymbol{O}\alpha$ as "all that is known is α" and this denotes that precisely those worlds where α is true are in the epistemic state.

We use the following terminology to refer to certain classes of formulas:

- A formula not mentioning any modalities is called *objective*.
- A formula is called *subjective* if every atom is in the scope of a modality.
- A formula is called *basic* if it does not mention $\boldsymbol{O}$.

We say that $\alpha \in \mathcal{OL}$ is satisfiable iff there is a model (e, w) such that $e, w \models \alpha$. When α is objective, we often write $w \models \alpha$ instead of $e, w \models \alpha$. When α is subjective, we often write $e \models \alpha$ instead of $e, w \models \alpha$. Given a set of sentences Σ, we write $\Sigma \models \alpha$ (read: "Σ entails α") iff for every (e, w) such that $e, w \models \alpha'$ for every $\alpha' \in \Sigma$, $e, w \models \alpha$. We write $\models \alpha$ (read: "α is valid") to mean $\{\} \models \alpha$.

In the sequel, we will find it convenient to refer to always-true and always-false objective sentences. So let TRUE denote a sentence that is always-true, say $\forall x(x = x)$, and let FALSE denote the negation of TRUE.

3.1.1 Properties

Before embarking on the epistemic properties of $\mathcal{OL}$, it is worthwhile to first be clear on how precisely $\mathcal{L}$ is related to standard FOL. We review a few essentials briefly, especially those features that will prove useful for later chapters. A more comprehensive presentation, along with proofs for all statements dealt in this section, appears in [Levesque and Lakemeyer, 2001].

Properties of $\mathcal{L}$

The semantical aspects, in regards to $\mathcal{L}$, that differ from FOL is its treatment of equality, and the logical symbols we call standard names. It turns out for sentences not mentioning equalities and names, $\mathcal{L}$ and FOL behave identically.

Theorem 3.1.1. *[Levesque and Lakemeyer, 2001]*
Suppose $\alpha \in \mathcal{L}$ does not mention names and equality. Then $\models \alpha$ iff α is a valid sentence of FOL.

Further, the treatment of equality in $\mathcal{L}$ is intuitive in the sense that it allows for a equivalence relation for a substitution of arguments. More precisely, let $\mathcal{A}$ denote the following sentences.

- *reflexivity*: $\forall x(x = x)$;
- *symmetry*: $\forall x \forall y(x = y \supset y = x)$;
- *transitivity*: $\forall x \forall y \forall z((x = y) \wedge (y = z) \supset x = z)$;
- substitution of equals for functions: for any function symbol f,

$$\forall x_1, \ldots, x_k \forall y_1, \ldots, y_k((x_1 = y_1) \wedge \ldots \wedge (x_k = y_k)) \supset$$
$$f(x_1, \ldots, x_k) = f(y_1, \ldots, y_k);$$

- substitution of equals for predicates: for any predicate symbol P,

$$\forall x_1, \ldots, x_k \forall y_1, \ldots, y_k((x_1 = y_1) \wedge \ldots \wedge (x_k = y_k)) \supset$$
$$P(x_1, \ldots, x_k) \equiv P(y_1, \ldots, y_k).$$

Then,

Theorem 3.1.2. *[Levesque and Lakemeyer, 2001]*
Suppose $\alpha \in \mathcal{L}$ is a sentence not mentioning standard names. Then α is valid iff in the classical account of FOL, $\mathcal{A} \cup \mathcal{B}$ implies α, where $\mathcal{B}$ is the following set of sentences:

$$\{\neg \exists x_1, \ldots, x_k \forall y(y = x_1 \vee \ldots \vee y = x_k) \mid k \in \mathbb{N}\}.$$

The set $\mathcal{B}$ essentially says that for every k, there are more than k individuals in the domain. This essentially forces us to consider infinite domains.

Standard names are interesting in their own right. They can be understood as an infinitary variant of the unique name assumption. For instance, given two names n_1 and n_2, $n_1 = n_2$ is valid iff n_1 and n_2 are the same and $n_1 \neq n_2$ is valid iff n_1 and n_2 are distinct. To see where this pays off, let $*$ be a bijection from names to names. Given any term t or formula α, let t^* and α^* denote the expression obtained on replacing every name in t and α with their corresponding mappings under $*$. Then,

Theorem 3.1.3. *[Levesque and Lakemeyer, 2001]*
Let $*$ be a bijection from names to names. Then $\models \alpha$ iff $\models \alpha^*$.

As a useful corollary, we get:

Corollary 3.1.4. *Let $\alpha \in \mathcal{L}$ have a single free variable x and let n be a name not appearing in α. Let $n_1, \ldots, n_k$ be all the names mentioned in α. Then, $\models \forall x\alpha$ iff $\models \alpha^x_n$ and $\models \alpha^x_{n_j}$ for all $j \in \{n_1, \ldots, n_k\}$.*

That is, to evaluate $\forall x\alpha$, we only need to decide a finite $(k + 1)$ number of substitutions.

One last difference to take note is that the *compactness property* does not hold for $\mathcal{L}$. That is, FOL has the property that a (possibly infinite) set of sentences is satisfiable iff all of its finite subsets are. This does not hold for $\mathcal{L}$, which the following unsatisfiable set illustrates because all of its proper subsets are indeed satisfiable:

$$\{\exists x P(x), \neg P(^{\#}0), \neg P(^{\#}1), \neg P(^{\#}2), \ldots\}.$$

The reason for this is simply that since the domain is countably infinite, we can use an infinite set of sentences to name every element of the domain.

Properties of $\mathcal{OL}$

Let us begin with a remark about $\boldsymbol{O}$. From the semantics, it is clear that only knowing a formula implies knowing it as well. That is,

$$\models (\boldsymbol{O}\alpha \supset \boldsymbol{K}\alpha).$$

Also worth noting is that since we do not force the real world to be included in the epistemic state, the agent can indeed believe false facts:

Theorem 3.1.5. *[Levesque and Lakemeyer, 2001]*
There are sentences α such that

- *$\alpha \wedge \neg\boldsymbol{K}\alpha$ is satisfiable;*
- *$\neg\alpha \wedge \boldsymbol{K}\alpha$ is satisfiable.*

Turning to the usual properties about knowledge, it is perhaps not too surprising to note that $\mathcal{OL}$ exhibits **K45** properties.

1. $\models (\boldsymbol{K}\alpha \wedge \boldsymbol{K}(\alpha \supset \beta) \supset \boldsymbol{K}\beta)$;
2. $\models (\boldsymbol{K}\alpha \supset \boldsymbol{K}(\boldsymbol{K}\alpha))$;
3. $\models (\neg\boldsymbol{K}\alpha \supset \boldsymbol{K}(\neg\boldsymbol{K}\alpha))$.

Since we have a language with quantifiers, two other properties are of interest. The first is called the *Barcan property*, and it is about the closure of belief under universal generalization:

4. $\models (\forall x\boldsymbol{K}\alpha \supset \boldsymbol{K}\forall x\alpha)$,

which says that if an agent believes every instance of α, then he also believes $\forall x\alpha$. This can be seen as a consequence of keeping a fixed discourse. The other property is about existential quantifiers. Consider the difference between saying "There is someone I know who is a spy" and "I know that someone is a spy", that is, $\exists x\boldsymbol{K}Spy(x)$ *vs.* $\boldsymbol{K}\exists xSpy(x)$. One would say that the latter is more or less obvious, but the former expresses a concrete assertion, one not known to everybody. In philosophical circles, the latter is termed *de*

dicto (literally, "knowledge of words") knowledge and the former is called *de re* (literally, "knowledge of things") knowledge [Hintikka, 1962]. In $\mathcal{OL}$, the existential version of the Barcan property holds, which is intuitive, and states that *de re* knowledge implies *de dicto* knowledge. However, the converse property does not, as should be the case.

5. $\models (\exists x \boldsymbol{K}\alpha \supset \boldsymbol{K}\exists x\alpha)$;

6. But $\not\models (\boldsymbol{K}\exists x\alpha \supset \exists x\boldsymbol{K}\alpha)$ for arbitrary α. For example, if n and n' are distinct names, then $\not\models \boldsymbol{K}(P(n) \vee P(n')) \supset \exists x\boldsymbol{K}P(x)$.

3.1.2 Representations of Epistemic States

Knowledge in a knowledge-based system can be thought of in two closely related ways: as characterized by an (abstract) epistemic state, where knowledge is interpreted over a set of world states, and in a symbolic form, *i.e.* a collection of propositions about the world, and knowledge is what can be logically deduced from that collection. These two perspectives is what Newell [1993] differentiates as the *knowledge level* and the *symbol level* respectively. We briefly discuss this relationship below.

The first property to note about epistemic states is that there are many equivalent states in the sense that they satisfy the same set of *basic beliefs* [Levesque, 1990]. More precisely, define the *basic belief set* wrt an epistemic state e as the set of the basic formulas believed at e, *i.e.* $\{\alpha \mid \alpha \text{ is basic, and for any } w, e, w \models \boldsymbol{K}\alpha\}$. But if two epistemic states, say e and e', have the same basic belief set, we would also want them to agree on what they only know. To realize this feature, Levesque proposes a simple solution, that of restricting ourselves to certain *maximal* epistemic states. Formally, given e, we define

$$e^+ = \{w \mid \text{ for all objective formulas } \alpha, \text{ if } e, w \models \boldsymbol{K}\alpha \text{ then } e, w \models \alpha\}$$

and say that e is *maximal* iff $e = e^+$. Then,

Theorem 3.1.6. *[Levesque and Lakemeyer, 2001]*
For any epistemic state e, there is a unique maximal state e^+ such that their basic belief set is identical.

It then follows that we can relate every basic belief set with a maximal epistemic state:

Theorem 3.1.7. *[Levesque and Lakemeyer, 2001]*
There is a bijection between basic belief sets and maximal epistemic states.

So suppose we restrict our attention to maximal epistemic states. There are two questions we want to now ask:

1. *Given a symbolic KB, how does one characterize the corresponding maximal epistemic state?*

2. *Given an epistemic state e, how does one find a representation for it in terms of a symbolic KB?*

The answer to the first question turns out to be a simple one if we are to restrict ourselves to objective KBs. We make this assumption for now, and defer discussions on subjective KBs to Section 3.1.4. We define the *epistemic state represented by the KB* Σ, where Σ is any (possibly infinite) set of objective sentences, as:

$$\Re(\Sigma) = \{w \mid w \models \alpha, \text{ for every } \alpha \in \Sigma\}.$$

It is not hard to see that $\Re(\Sigma)$ is an epistemic state where Σ is all that is known.

The answer to the second question, however, is more elaborate. Let us begin by calling an epistemic state e *representable* when there is a (possibly infinite) set of objective sentences Σ such that $e = \Re(\Sigma)$. We say e is *finitely representable* if there is a *finite* set of objective sentences Σ such that $e = \Re(\Sigma)$. Then the second question can be reformulated to one that asks whether every maximal epistemic state is representable. This would then allow, among other things, to have a precise correspondence between symbolic representations and epistemic states. Preferably, we are further interested in finitely representable epistemic states. This is because our typical intuitions about the representation of knowledge is some collection of finite structures, which can be manipulated to query stored facts and learn new ones.

Unfortunately, Lakemeyer and Levesque [2001] show that there is an infinite set of satisfiable basic sentences such that no representable epistemic state satisfies the set. Practically speaking, however, in any KR system one only considers a finite set of sentences. Fortunately, every finite set of satisfiable basic sentences is indeed satisfiable in a representable epistemic state.

So, we now finally ask: can we restrict ourselves to finitely representable ones? The answer, it turns out, is *no*. That is, there is a satisfiable basic sentence which is false at every finitely representable epistemic state. By above, of course, it is satisfiable in some representable epistemic state corresponding to an infinite set of objective sentences. One unfortunate consequence of this is that when dealing with *validity*, we cannot restrict ourselves to finitely representable epistemic states.

Nevertheless, for practical purposes, Lakemeyer and Levesque show that it suffices to consider finitely representable epistemic states. To give a cursory introduction to their idea, let us first formalize the notion of adding information to an epistemic state. We define **TELL** as an operator that accepts an epistemic state e and an objective sentence α to result in a new epistemic state as follows:

$$\mathbf{TELL}[\alpha, e] = e \cap \{w \mid w \models \alpha\}.$$

Now, imagine starting with an empty KB, where the epistemic state is simply $\mathcal{W}$. Of course, $\mathcal{W}$ is finitely representable, *i.e.* $\mathcal{W} = \Re(\text{TRUE})$. Suppose we have a set of objective sentences $\alpha_1, \ldots, \alpha_k$ to add to the KB. Lakemeyer and Levesque show that finitely representable states are closed under **TELL**. So $\mathbf{TELL}[\alpha_j, \mathcal{W}]$ is finitely representable. Therefore, under the practical consideration that we will only add a finite number of objective facts to the KB, epistemic states of interest will always be finitely representable.

Of course, there may be instances where we would like to add subjective sentences to the KB. Natural examples are sentences of the form $\forall x.(P(x) \supset \boldsymbol{K}P(x))$ that make *closed world assertions*, as touched upon earlier. Roughly, it says that every instance of P in the real world is currently known. In order to deal with the more general **TELL** operation, they make use of the *representation theorem* to be introduced next, which allows us to reduce subjective sentences to purely objective ones.

3.1.3 The Representation Theorem

One way to imagine a practical KR system is to think of two major operations. On the one hand, we devise a mechanism to absorb information as it becomes available, such as **TELL**, and on the other, we propose a decision procedure for the evaluation of queries. These correspond to a *functional approach* to KR [Levesque,

1984], where instead of viewing a knowledge base as a set of structures that represent knowledge, we imagine interacting with a system in terms of what it can be asked and told.

The intuitive usage of $\mathcal{OL}$ is to begin with a KB in the scope of $\boldsymbol{O}$, and ask queries of the form $\boldsymbol{K}\alpha$. Roughly speaking, this means that we require a first-order modal prover, and in particular, one that is faithful to $\mathcal{OL}$'s truth theory. The representation theorem is a fundamental result to eliminate the $\boldsymbol{K}$ operators in the query (wrt the KB), so that a first-order theorem prover can be used to evaluate the modified version of the query against the KB. Thus, no modal reasoning will be necessary.

Example 3.1.8. To illustrate the idea, suppose a KB Σ is the following:

$$\Sigma = \{Smaller(B, A), Smaller(C, A) \vee Smaller(D, A)\}.$$

That is, in a blocks world domain: B is smaller than A, and C is smaller than A or D is smaller than A.[2] Supposing we ask:

$$\boldsymbol{K}\exists x.\, (Smaller(x, A) \wedge \neg \boldsymbol{K} Smaller(x, A))$$

That is, does Σ know of a block that is smaller than A, but does not know which one? The answer is certainly *yes* because the list of smaller blocks known is incomplete, except for B. The main step is to replace $\boldsymbol{K}(x, A)$ with $x = B$. Then, it can be shown that the query reduces to verifying if $\exists x.\, (Smaller(x, A) \wedge x \neq B)$ is entailed by Σ. ∎

To make this intuition precise, we define $\|\alpha\|_\Sigma$ that eliminates every $\boldsymbol{K}$ operator appearing in α using equality expressions derived from Σ, provided Σ is all that is known. But first, since expressions such $\boldsymbol{K} Smaller(x, A)$ contain free variables, a procedure $\textsc{Res}[\alpha, \Sigma]$ is defined to return the known instances of the objective formula α from an objective sentence Σ.

Definition 3.1.9. [Levesque and Lakemeyer, 2001]
Let α be an objective formula, and Σ is an objective sentence. Let $n_1, \ldots, n_k$ be all the names occurring in Σ and α and n' is a name not occurring in Σ or α. Then, $\textsc{Res}[\alpha, \Sigma]$ is defined as:

1. If α has no free variables, then $\textsc{Res}[\alpha, \Sigma]$ is $\textsc{True}$ if $\Sigma \models \alpha$ and $\textsc{False}$ otherwise.[3]
2. If x is a free variable in α, then $\textsc{Res}[\alpha, \Sigma]$ is defined as:

$$((x = n_1) \wedge \textsc{Res}[\alpha^x_{n_1}, \Sigma]) \vee \ldots \vee ((x = n_k) \wedge \textsc{Res}[\alpha^x_{n_k}, \Sigma]) \vee$$
$$((x \neq n_1) \wedge \ldots \wedge (x \neq n_k) \wedge \textsc{Res}[\alpha^x_{n'}, \Sigma]^{n'}_x). \text{ ∎}$$

Definition 3.1.10. [Levesque and Lakemeyer, 2001]
Given an objective sentence Σ and a basic formula α, $\|\alpha\|_\Sigma$ is the objective formula defined by

1. $\|\alpha\|_\Sigma = \alpha$, when α is objective;
2. $\|\neg\alpha\|_\Sigma = \neg\|\alpha\|_\Sigma$;

[2] We implicitly assume that all proper nouns dealt in the sequel are names from $\mathcal{N}$.
[3] Because of item 1, note that $\textsc{Res}[*, *]$ is not *recursively enumerable* [Rogers Jr., 1987] since it appeals to validity, when returning $\textsc{True}$, and appeals to falsifiability, when returning $\textsc{False}$ [Levesque and Lakemeyer, 2001].

3. $\|\alpha \vee \beta\|_\Sigma = \|\alpha\|_\Sigma \vee \|\beta\|_\Sigma$;

4. $\|\forall x\alpha\|_\Sigma = \forall x\|\alpha\|_\Sigma$;

5. $\|\boldsymbol{K}\alpha\|_\Sigma = \text{RES}[\|\alpha\|_\Sigma, \Sigma]$. ∎

The representation theorem is applied by means of the following main result:

Theorem 3.1.11. *[Levesque and Lakemeyer, 2001]*
Suppose α is a basic sentence. Then, $\boldsymbol{O}\Sigma \supset \alpha$ is valid iff $\|\alpha\|_\Sigma$ is valid.

Example 3.1.8 Continued. Revisiting the example query, we have as follows:

$$\text{RES}[\boldsymbol{K}\exists x.\ (\mathit{Smaller}(x, A) \wedge \neg\boldsymbol{K}\mathit{Smaller}(x, A)), \Sigma]$$

$$= \text{RES}[\|\exists x.\ (\mathit{Smaller}(x, A) \wedge \neg\boldsymbol{K}\mathit{Smaller}(x, A)\|_\Sigma, \Sigma].$$

Now, pursue

$$\|\exists x.\ (\mathit{Smaller}(x, A) \wedge \neg\boldsymbol{K}\mathit{Smaller}(x, A)\|_\Sigma$$

$$= \exists x.\ (\|\mathit{Smaller}(x, A)\|_\Sigma \wedge \|\neg\boldsymbol{K}\mathit{Smaller}(x, A)\|_\Sigma).$$

Now pursue

$$\|\neg\boldsymbol{K}\mathit{Smaller}(x, A)\|_\Sigma$$

$$= x \neq B.$$

In sum, we are to decide if $\Sigma \models \exists x.\ (\mathit{Smaller}(x, A) \wedge x \neq B)$, which is indeed the case and so the procedure returns TRUE. ∎

Returning to our discussion on **TELL** from Section 3.1.2, in order to add an arbitrary basic sentence α to Σ, Lakemeyer and Levesque show that this is equivalent to **TELL**$[\|\alpha\|_\Sigma, \Sigma]$. Thus, for most applications, it often suffices to consider only objective KBs.

3.1.4 Nonmonotonicity

While the reduction of subjective formulas to objective ones has conceptual clarity with regards to a knowledge base in question, they do not, however, allow us to reason about default rules, such as the one about big blocks being in the storage room considered earlier. With that example, it is often beneficial to imagine subjective (in particular, basic) KBs.

However, when reasoning about default rules, or about KBs that refer to their own beliefs in general, a number of additional difficulties arise. For example, if we let Σ denote the following KB

$$(\neg\boldsymbol{K}p \supset q) \wedge (\neg\boldsymbol{K}q \supset p)$$

then it turns out that only knowing that KB results in two distinct maximal epistemic states:

$$\boldsymbol{O}\Sigma \equiv \boldsymbol{O}p \vee \boldsymbol{O}q.$$

For another example, there is no epistemic states satisfying $\boldsymbol{O}(\boldsymbol{K}p \vee \boldsymbol{K}q)$, or for that matter $\boldsymbol{O}\boldsymbol{K}p$. All of this, of course, complicates the treatment of knowledge bases at the knowledge level.

We mentioned earlier that the nonmonotonicity sanctioned by $\mathcal{OL}$ is similar in spirit to AEL [Moore, 1985b]. However, AEL is defined using meta-logical properties; fixed-point operators on the beliefs of a KB in particular. In contrast, as we have already observed, inferences in $\mathcal{OL}$ are understood in terms of the logical consequences of only knowing the KB. Moreover, Levesque [1990] showed that there is a precise correspondence between the inferences sanctioned by AEL and $\mathcal{OL}$. Thus, $\mathcal{OL}$ allows us to semantically reconstruct a major proposal in nonmonotonic reasoning entirely within a classical *monotonic* logic. We will not go into more details on these topics and defer interested readers to [Levesque and Lakemeyer, 2001]. As a consolation, we will also not make use of nonmonotonicity in this thesis, except for illustrative purposes in this chapter.

We should also point out that there have also been other approaches to capture the intuition of only knowing, which are also motivated in terms of providing a semantical rationalization of nonmonotonicity wrt beliefs. In particular, Halpern and Moses [1984] propose a notion called *minimal knowledge*, which is similar in spirit to Levesque's only knowing. While initially proposed in the context of (propositional) **S5** *i.e.* knowledge is true, it was shown to accommodate **K45**, among others, without any modifications [Halpern, 1997]. The idea is that α is all that is known iff the epistemic state is the maximal subset of worlds where α is known. Equivalently, this epistemic state is the union of all worlds states that know α. They call a formula *honest* if it is known in the epistemic state. A formula is *dishonest* if it not honest. For example, $\boldsymbol{K}p \vee \boldsymbol{K}q$ is dishonest since there is no maximal epistemic state where that formula is known.

Levesque's only knowing coincides with Halpern and Moses's notions for objective formulas, which are trivially honest. At first glance, the difference between the two approaches is made obvious by the fact that Halpern and Moses discuss their version of only knowing as a meta-logical concept, *i.e.* only knowing is not expressed in the object language. But there are other differences. For one, it turns that the only knowing modality has a subtle relationship with the belief modality in the Levesque framework, which we will take up in Section 3.2.1, that is quite different from how the notion of "all I know" behaves in the Halpern and Moses framework [Halpern, 1997]. For another, Rosati [2000] demonstrates that while satisfiability in propositional $\mathcal{OL}$ is at the second level of polynomial hierarchy,[4] the satisfiability in the logic of minimal knowledge is at the third level. Therefore, unless the polynomial hierarchy collapses, reasoning in minimal knowledge is computationally harder than reasoning about only knowing.

A proposal by Pratt-Hartmann [2000] is also closely related to only knowing. Pratt-Hartmann introduces the concept of *total knowledge*, where α is total knowledge if α is known and every formula not entailed as a result of knowing α is not known. For objective theories, his ideas are closely related to $\mathcal{OL}$. However, his formalism insists that knowledge is true. For that reason a relationship to AEL is not immediate, and requiring certain formulas such as $\neg\boldsymbol{K}p \supset \neg p$ to be total knowledge leads to an inconsistency.

Of course, there are also other approaches to nonmonotonic reasoning that are not based on only knowing.

[4] Intuitively, the second level contains problems that can be solved in polynomial time by a nondeterministic Turing machine using an NP-oracle. (An NP-oracle solves NP problems in constant time.) See [Johnson, 1990] for an overview of the polynomial hierarchy.

Readers may want to consider Reiter's [1987] survey for an overview. Over the years, extensive work has been carried out to establish the relationship between AEL and other approaches [*e.g.* Gottlob, 1993; Konolige, 1989], and by way of Levesque's result, these comparisons can be imported to $\mathcal{OL}$.

3.1.5 Proof Theory

For a clearer view of only knowing, it is convenient for this section and the next to not consider $\boldsymbol{O}$ as a primitive notion but in terms of $\boldsymbol{K}$ and a new modal operator $\boldsymbol{N}$ such that $\boldsymbol{O}\alpha$ is understood as syntactically denoting $\boldsymbol{K}\alpha \wedge \boldsymbol{N}\neg\alpha$. Intuitively, in light of reading $\boldsymbol{K}\alpha$ as "at least α is believed", the new operator is a natural dual in the sense that $\boldsymbol{N}\neg\alpha$ is to be read as "at most α is believed". The justification for this reading will be clear in a moment.

Turning now to a semantics for $\boldsymbol{N}\alpha$, the idea is to interpret α at all the worlds that the agent does not consider epistemically possible. So clearly we have to consider the worlds $w \notin e$. But if $\boldsymbol{N}\alpha$ is a complicated formula involved nested modalities, we would still need an epistemic state to interpret subformulas of the form $\boldsymbol{K}\alpha'$. Levesque takes the intuitive approach by still using the same e for the subformulas:

- $e, w \models \boldsymbol{N}\alpha$ iff for every $w' \notin e$, $e, w' \models \alpha$.

Thus, we have good reasons to read $\boldsymbol{N}\neg\alpha$ as *at most* α is believed because if the agent knew more, then he would not consider impossible all those worlds where α is true.

Levesque's proof theory for propositional $\mathcal{OL}$ is as follows. First, since $\mathcal{OL}$ exhibits **K45** properties, it is not surprising that the axiomatization includes axioms **K**, **4** and **5**, and that inference rules include Modus Ponens and knowledge generalization. In fact, these schemas are applicable to both the modal operators $\boldsymbol{K}$ and $\boldsymbol{N}$.

Axioms:

A1. All instances of axioms of propositional logic,

A2. $\boldsymbol{K}(\alpha \supset \beta) \supset (\boldsymbol{K}\alpha \supset \boldsymbol{K}\beta)$,

A3. $\boldsymbol{N}(\alpha \supset \beta) \supset (\boldsymbol{N}\alpha \supset \boldsymbol{N}\beta)$,

A4. $\sigma \supset \boldsymbol{K}\alpha \wedge \boldsymbol{N}\alpha$ for every subjective σ,

A5. $\boldsymbol{N}\alpha \supset \neg\boldsymbol{K}\alpha$ if $\neg\alpha$ is a propositionally consistent objective formula.

Inference rules:

MP. From α and $\alpha \supset \beta$ infer β.

NEC. From α infer $\boldsymbol{K}\alpha$ and $\boldsymbol{N}\alpha$.

Axiom schemas **A1** − **A3** justify **K45** properties for both $\boldsymbol{K}$ and $\boldsymbol{N}$ separately. Axiom **A4** tells us that the operators are mutually introspective, *i.e.* $\boldsymbol{K}\alpha \supset \boldsymbol{N}\boldsymbol{K}\alpha$ is valid. One way to look at this is in terms of two agents that are mutually introspective. Arguably, the most interesting and novel axiom here is **A5**.[5] It is this

[5]Note that the axioms are recursive. In particular, **A5** appeals to consistency in classical propositional logic, which is decidable.

axiom that makes precise the relationship between $\boldsymbol{N}$ and $\boldsymbol{K}$. It is possible to see that its soundness rests of the fact that $\boldsymbol{K}$ and $\boldsymbol{N}$ together range over all possible worlds.

We call this proof theory **AX**, and we write $\mathbf{AX} \vdash \alpha$ (or simply $\vdash \alpha$) to denote that α is provable using the schemas in **AX**.

Example 3.1.12. Let us revisit the big blocks default once more. We now consider a propositional default. Let p denote a big block. Let q denote that this block is found in the storage. Then, let the default δ be

$$p \wedge \neg \boldsymbol{K} \neg q \supset q.$$

That is, if the agent believes p and if the agent does not believe $\neg q$, then q can be inferred. We intend to reason as follows:

$$\vdash \boldsymbol{O}(p \wedge \delta) \supset \boldsymbol{K} q$$

That is, if all the agent knows is p and the default δ, he must come to believe q. A formal derivation is given below. We write **Def** to mean the equivalence $\boldsymbol{O}\alpha \equiv \boldsymbol{K}\alpha \wedge \boldsymbol{N}\neg\alpha$. We freely reason with propositional logic (**PL**) and **K45**.

1.	$\boldsymbol{O}(p \wedge \delta) \supset \boldsymbol{O}(p \wedge (\neg \boldsymbol{K} \neg q \supset q))$	**PL**
2.	$\boldsymbol{O}(p \wedge (\neg \boldsymbol{K} \neg q \supset q)) \supset \boldsymbol{K}(p \wedge (\neg \boldsymbol{K} \neg q \supset q))$	1, **Def**
3.	$\boldsymbol{K}(p \wedge (\neg \boldsymbol{K} \neg q \supset q)) \supset (\boldsymbol{K} \neg \boldsymbol{K} \neg q \supset \boldsymbol{K} q)$	2, **K45**
4.	$\boldsymbol{O}(p \wedge (\neg \boldsymbol{K} \neg q \supset q)) \supset \boldsymbol{N} \neg (p \wedge (\neg \boldsymbol{K} \neg q \supset q))$	1, **Def**
5.	$\boldsymbol{N} \neg (p \wedge (\neg \boldsymbol{K} \neg q \supset q)) \supset \boldsymbol{N}(p \supset \neg q)$	4, **K45**
6.	$\boldsymbol{N}(p \supset \neg q) \supset \neg \boldsymbol{K}(p \supset \neg q)$	5, **A5**
7.	$\neg \boldsymbol{K}(p \supset \neg q) \supset \neg \boldsymbol{K} \neg q$	6, **K45**
8.	$\neg \boldsymbol{K} \neg q \supset \boldsymbol{K} \neg \boldsymbol{K} \neg q$	7, **A4**
9.	$\boldsymbol{O}(p \wedge \delta) \supset (\boldsymbol{K} \neg \boldsymbol{K} \neg q \wedge (\boldsymbol{K} \neg \boldsymbol{K} \neg q \supset \boldsymbol{K} q))$	$1-8$, **PL**
10.	$\boldsymbol{O}(p \wedge \delta) \supset \boldsymbol{K} q$	9, **PL**

For most parts of the proof, we make use of standard **K45** and propositional reasoning. The only place we need to invoke the relationship between $\boldsymbol{N}$ and $\boldsymbol{K}$ is in line 6, where we apply it to a propositional subformula obtained in line 5. Note again that the applicability of the axiom **A5** is limited to non-valid formulas, which is the case with $p \supset \neg q$. Line 8 makes use of **A4**. Equivalently, we could have used **K45** reasoning to reduce $\boldsymbol{K} \neg \boldsymbol{K} \neg q$ to $\neg \boldsymbol{K} \neg q$ in line 3 and omitted this step. ∎

Levesque obtained the following result regarding the axiomatization wrt propositional $\mathcal{OL}$:

Theorem 3.1.13. *[Levesque, 1990]*
The above axiomatization is sound and complete for (propositional) $\mathcal{OL}$.

As a last remark, the completeness result crucially depends on the assumption that there are an infinite set of propositions Φ. Halpern and Lakemeyer [2001] show that if Φ is finite, then the axioms are incomplete. Nevertheless, a completeness result does hold with the addition of an axiom, the details of which does not concern us here.

3.2 The Logic of Only Knowing with Many Agents $\mathcal{OL}_n$

In this section, we are interested in generalizing $\mathcal{OL}$ to the many agent case. As noted in Halpern and Lakemeyer [2001], the semantical framework of $\mathcal{OL}$ has some interesting features, which we expect to guide us when proposing the generalization. For one thing, note that the union of the worlds wrt which $\boldsymbol{K}\alpha$ is evaluated, *i.e.* the epistemically possible worlds, and the worlds wrt which $\boldsymbol{N}\alpha$ is evaluated, *i.e.* the worlds considered impossible, is the set of all conceivable worlds $\mathcal{W}$. Roughly speaking, the operator $\boldsymbol{K}$ has a subtle relationship to belief operator $\boldsymbol{N}$ which makes extensions to the many agent case non-trivial. In fact, existing approaches that extend only knowing to many agents have undesirable properties. For example, Lakemeyer [1993] proposes a Kripke structure approach where, among other problems, it can be shown that certain types of epistemic states cannot be constructed. In a Kripke approach by Halpern [1993], the at most and at least modalities do not seem to interact in a natural manner. An approach by Halpern and Lakemeyer [2001] does successfully model multiagent only knowing, but it axiomatizes the semantical notion of validity. Precisely for this reason that approach is not natural. Finally, an axiomatization by Waaler [2004] does not resort to such ideas, but here the model theory has problems [Waaler and Solhaug, 2005].

In general, among the main problems are that these approaches make use of arbitrary Kripke structures which already unwittingly discard the simplicity of Levesque's semantics. Moreover, the approaches are mostly propositional and it is not obvious how one can extend these ideas to a first-order language.

Thus, in order to prepare for the results presented in this section, we begin by reviewing the features of $\mathcal{OL}$'s model theory, as outlined in [Halpern and Lakemeyer, 2001]. Then we propose a semantics for the first-order case, and return to analyze these features. After that, we turn to a proof theory that characterizes the semantics for the propositional fragment, and illustrate some examples. Finally, we compare our work to the approach in [Halpern and Lakemeyer, 2001] and show that we capture the same properties but without their problems.

3.2.1 Features of $\mathcal{OL}$

In the framework of $\mathcal{OL}$, we associate a world with the set of formulas that are true at the world. Intuitively speaking, a world is a possible state of affairs, or a *possibility* for short. The intuitive reading of an epistemic state then is that it a *set of possibilities*, which captures the idea that an agent entertains a number of possible ways in which the world can be.

The first property to note with regards to $\mathcal{OL}$ is the semantics of $\boldsymbol{N}$. We observed that the epistemic state is not affected when evaluating $\boldsymbol{N}\alpha$:

P1. *The set of possibilities remains fixed when evaluating formulas of the form* $\boldsymbol{N}\alpha$.

A closer look informs us that this idea is essential for the soundness of **A4** in Levesque's proof theory. Indeed,

since this axiom makes the formula $\boldsymbol{K}\alpha \supset \boldsymbol{N}\boldsymbol{K}\alpha$ valid, it follows that an epistemic state that interprets $\boldsymbol{K}\alpha$ is also one where $\boldsymbol{N}\boldsymbol{K}\alpha$ is true.

The second property to note is that the set of *impossible worlds* that interpret $\boldsymbol{N}\alpha$ is always $\mathcal{W} - e$. Equivalently, the worlds that evaluate $\boldsymbol{K}\alpha$ and $\boldsymbol{N}\alpha$ is clearly $\mathcal{W}$.

P2. *The union of the set of possible worlds, which evaluate $\boldsymbol{K}\alpha$, and the set of impossible worlds, which evaluate $\boldsymbol{N}\alpha$, results in the set of all conceivable worlds $\mathcal{W}$. The set of conceivable worlds is absolute and independent of the model (e, w). Further, it is always the set of all truth assignments.*

This property is essential for the soundness of **A5** in Levesque's proof theory. This can be reasoned as follows. Suppose α is any non-valid formula and $\boldsymbol{N}\alpha$ is satisfied in a model (e, w). By axiom **A5**, $\neg\boldsymbol{K}\alpha$ is also true at (e, w). That is, since α is true in all the impossible worlds, and since $\neg\alpha$ is satisfiable by assumption, there must be a possible world that satisfies $\neg\alpha$. Therefore, α is not believed at (e, w).

The intent of the last property is to allow an agent to only know any objective formula:

P3. *For every set of conceivable worlds, there is a model where precisely this set constitutes as the epistemic state.*

Arguably, these properties seem straightforward in the single agent case. However, generalizing these properties to the many agent case is non-trivial. The difficulty seems to be that, in the multiagent case, we no longer identify a possible state of affairs with objective formulas. This is because a formula such as p is just as objective from i's point of view as $\boldsymbol{K}_j p$ (read: "j believes p"). Therefore, the appropriate generalization of a possibility in the many agent case is a set of *i-objective* formulas, by which we mean formulas such as p and $\boldsymbol{K}_j p$ for $j \neq i$ (defined formally below).

In the next section, we present a semantics for multiagent only knowing for a quantified language with equality, that is, a faithful extension to $\mathcal{OL}$. We then present the earlier propositional treatments, and return to analyze and generalize the above properties.

As a concluding remark to this section, it is interesting to note that the formulation of $\mathcal{OL}$'s features, especially **P3**, as analyzed by Halpern and Lakemeyer is only faithful to the propositional fragment. That is, it is assumed in their analysis that epistemic possibilities are completely captured by objective formulas in the single agent case. But for a quantified language, we mentioned in Section 3.1.2 that there are epistemic states that are not representable, *i.e.* they can not be characterized using only objective formulas. Be that as it may, we will continue to be interested in **P1**, **P2** and **P3**. It is an open question as to how precisely one is to generalize the features of first-order $\mathcal{OL}$.

3.2.2 A Semantics for Multiagent Only Knowing

Let us begin by extending the language. Let $\mathcal{OL}_n$ be a first-order modal language that enriches the non-modal fragment of $\mathcal{OL}$ with modal operators $\boldsymbol{K}_i, \boldsymbol{N}_i$ and $\boldsymbol{O}_i$ for $i \in \{A, B\}$. For ease of exposition, we will only consider two agents: A denoting the agent Alice, and B denoting the agent Bob. Extensions to more agents is straightforward.

By analogy to the single agent case, we freely use $\boldsymbol{O}_i$ such that $\boldsymbol{O}_i\alpha$ syntactically denotes $\boldsymbol{K}_i\alpha \wedge \boldsymbol{N}_i\neg\alpha$ and is to be read as "all that i knows is α". The first step is to be clear on what *objective* and *subjective* formulas

mean in the many agent case. As hinted in the previous discussion, they are now understood relative to the agent. A formula is called *i-objective* if all epistemic operators which do not occur within the scope of another epistemic operator are of the form $\boldsymbol{M}_j$, $j \neq i$, where $\boldsymbol{M}_i$ denotes $\boldsymbol{K}_i$ or $\boldsymbol{N}_i$. A formula is called *i-subjective* if every atom is in the scope of an epistemic operator and all epistemic operators which do not occur within the scope of another epistemic operator are of the form $\boldsymbol{M}_i$. Intuitively, *i*-subjective formulas represent *i*'s beliefs about the world whereas *i*-objective formulas determine what is true about the world from *i*'s perspective, which may include beliefs of agents other than *i*.

In what follows, it will be useful to refer to the degree of nesting of modal operators within a formula, and we lump together consecutive nesting of operators for the same agent. More precisely, we define the notion of the *i*-depth of a formula:

Definition 3.2.1. (*i*-depth.) The *i*-depth of a formula α, denoted $|\alpha|_i$, is defined inductively as ($\boldsymbol{M}_i$ denotes $\boldsymbol{K}_i$ or $\boldsymbol{N}_i$):

1. $|\alpha|_i = 1$ for atoms,
2. $|\neg\alpha|_i = |\alpha|_i$,
3. $|\forall x.\, \alpha|_i = |\alpha|_i$,
4. $|\alpha \vee \beta|_i = max(|\alpha|_i, |\beta|_i)$,
5. $|\boldsymbol{M}_i\alpha|_i = |\alpha|_i$,
6. $|\boldsymbol{M}_j\alpha|_i = |\alpha|_j + 1$, for $j \neq i$.

A formula α has a depth k if $max(|\alpha|_A, |\alpha|_B) = k$. ∎

Example 3.2.2. To illustrate the notion of depth, consider the formula $\boldsymbol{K}_A\boldsymbol{K}_B\boldsymbol{K}_Ap \vee \boldsymbol{K}_Bq$. Here:

1. $|\boldsymbol{K}_A\boldsymbol{K}_B\boldsymbol{K}_Ap \vee \boldsymbol{K}_Bq|_A = max(|\boldsymbol{K}_A\boldsymbol{K}_B\boldsymbol{K}_Ap|_A, |\boldsymbol{K}_Bq|_A) = 3$ because
   (a) $|\boldsymbol{K}_A\boldsymbol{K}_B\boldsymbol{K}_Ap|_A = |\boldsymbol{K}_B\boldsymbol{K}_Ap|_A = 1 + |\boldsymbol{K}_Ap|_B = 2 + |p|_A = 3$,
   (b) $|\boldsymbol{K}_Bq|_A = 1 + |q|_B = 2$.
2. $|\boldsymbol{K}_A\boldsymbol{K}_B\boldsymbol{K}_Ap \vee \boldsymbol{K}_Bq|_B = max(|\boldsymbol{K}_A\boldsymbol{K}_B\boldsymbol{K}_Ap|_B, |\boldsymbol{K}_Bq|_B) = 4$ because
   (a) $|\boldsymbol{K}_A\boldsymbol{K}_B\boldsymbol{K}_Ap|_B = 1 + |\boldsymbol{K}_B\boldsymbol{K}_Ap|_A = 1 + 3$ (as shown above) $= 4$,
   (b) $|\boldsymbol{K}_Bq|_B = |q|_B = 1$.
3. Therefore, the depth of the formula is 4.

Consider each of the disjuncts. $\boldsymbol{K}_A\boldsymbol{K}_B\boldsymbol{K}_Ap$ is both *A*-subjective as well as *B*-objective. On the other hand, $\boldsymbol{K}_Bq$ is both *B*-subjective as well as *A*-objective. Moreover, $\boldsymbol{K}_A\boldsymbol{K}_B\boldsymbol{K}_Ap \vee \boldsymbol{K}_Bq$ is neither *A*-subjective nor *B*-subjective. For that matter, it is neither *A*-objective nor *B*-objective. ∎

By analogy to the single agent case, an *objective* formula is one that does not mention any modal operators. A *basic* formula is one that does not mention any $\boldsymbol{N}_i$ (or equivalently, $\boldsymbol{O}_i$).

We now turn to the semantics. Typically, a semantics for multiagent systems is specified with Kripke structures generalized to n agents. That is, Kripke structures are models that interpret i-subjective and i-objective formulas. However, they do not share the simplicity of Levesque's semantical framework. Moreover, it seems that there are no natural ways to capture all the necessary features of multiagent only knowing with Kripke structures, as demonstrated by the involved treatments in [Halpern and Lakemeyer, 2001]. More on this in Section 3.2.6.

Our approach will instead be based on what we call as k-structures. The idea is to keep separate the worlds A believes from the worlds she considers B to believe, to some depth k. At the lowest level, we imagine an agent who has no beliefs whatsoever about the other agent, and reasons only about the state of the world. The next level involves an agent who not only reasons about the world but also has beliefs about what the other agent believes about the world. And so on, to some depth k.

Definition 3.2.3. A k-structure, with $k \geq 1$, say e^k, for an agent is defined inductively as:

- $e^1 \subseteq \mathcal{W} \times \{\{\}\}$,
- $e^k \subseteq \mathcal{W} \times \mathbb{E}^{k-1}$, where $\mathbb{E}^m$ is the set of all m-structures. ∎

That is, a e^1 is simply a set of worlds. A e^2 is a set of the form $\{(w, e^1), (w', e'^1), \ldots\}$ which states that at w an agent, say A, believes B to consider worlds from e^1 possible, and at w' she believes B to consider worlds from e'^1 possible. This captures the intuition that A has partial information about B, and so her beliefs about B differs at different worlds.

When modeling a k-structure, say e^k, for A we denote it as e^k_A. Analogously, when modeling a j-structure, say e^j, for B we denote it as e^j_B.

We define a e^k for A, a e^j for B and a world w as a (k, j)-model (e^k_A, e^j_B, w). The idea is that only formulas with a maximal A-depth of k and with a maximal B-depth of j are to be interpreted wrt (k, j)-models. A definition of truth is as follows:

1. $e^k_A, e^j_B, w \models P(t_1, \ldots, t_k)$ iff $P(n_1, \ldots, n_k) \in w$, where $|t_i|_w = n_i$;
2. $e^k_A, e^j_B, w \models t_1 = t_2$ iff $|t_1|_w$ is the same name as $|t_2|_w$;
3. $e^k_A, e^j_B, w \models \neg\alpha$ iff $e^k_A, e^j_B, w \not\models \alpha$;
4. $e^k_A, e^j_B, w \models \alpha \vee \beta$ iff $e^k_A, e^j_B, w \models \alpha$ or $e^k_A, e^j_B, w \models \beta$;
5. $e^k_A, e^j_B, w \models \forall x.\, \alpha$ iff $e^k_A, e^j_B, w \models \alpha^x_n$ for all $n \in \mathcal{N}$;
6. $e^k_A, e^j_B, w \models \boldsymbol{K}_A\alpha$ iff for all $(w', e^{k-1}_B) \in e^k_A$, $e^k_A, e^{k-1}_B, w' \models \alpha$;
7. $e^k_A, e^j_B, w \models \boldsymbol{N}_A\alpha$ iff for all $(w', e^{k-1}_B) \notin e^k_A$, $e^k_A, e^{k-1}_B, w' \models \alpha$.

Since $\boldsymbol{O}_A\alpha$ is syntactically understood as $\boldsymbol{K}_A\alpha \wedge \boldsymbol{N}_A\neg\alpha$, it follows that

- $e^k_A, e^j_B, w \models \boldsymbol{O}_A\alpha$ iff for all w', for all e^{k-1} for B, $(w', e^{k-1}_B) \in e^k_A$ iff $e^k_A, e^{k-1}_B, w' \models \alpha$.

A semantics for formulas of the form $\boldsymbol{K}_B\alpha$, $\boldsymbol{N}_B\alpha$ (and hence, $\boldsymbol{O}_B\alpha$) is given analogously based on items 6 and 7.

We remark that if only a single agent is involved, one only needs 1-structures. Then it is easy to see that the semantics for $\mathcal{OL}_n$ coincides with that of $\mathcal{OL}$ when $n = 1$.

When a formula has a maximal A-depth of k and a maximal B-depth of j, we say that the formula has a maximal A, B-depth of k, j for brevity. We say that a formula α of maximal A, B-depth of k, j is *satisfiable* iff there is a (k, j)-model (e_A^k, e_B^j, w) such that $e_A^k, e_B^j, w \models \alpha$. The formula is *valid* (written $\models \alpha$) iff α is true at all (k, j)-models. Satisfiability is extended to a set of formulas Σ (of maximal A, B-depth of k, j) in the manner that there is a (k, j)-model (e_A^k, e_B^j, w) such that $e_A^k, e_B^j, w \models \alpha'$ for every $\alpha' \in \Sigma$. We write $\Sigma \models \alpha$ to mean that for every (k, j)-model (e_A^k, e_B^j, w), if $e_A^k, e_B^j, w \models \alpha'$ for all $\alpha' \in \Sigma$, then $e_A^k, e_B^j, w \models \alpha$. As before, $\models \alpha$ denotes $\{\} \models \alpha$.

We often write $\{\}, e_B^j, w \models \alpha$ when α is A-objective because the k-structure for A is irrelevant. Analogously, for B-objective formulas, we often write $e_A^k, \{\}, w \models \alpha$. When the formula α is objective, we omit the structures for A and B altogether and simply write $w \models \alpha$.

Before moving on, let us consider examples of reasoning about only knowing with k-structures.

Example 3.2.4. Let α be an atom. Then the following sentences are valid.

1. $\boldsymbol{O}_A(\text{TRUE}) \supset \neg\boldsymbol{K}_A\neg\boldsymbol{K}_B\alpha$.

   The sentence is A-subjective and of A-depth 2. So consider any $(2, j)$-model that satisfies $\boldsymbol{O}_A(\text{TRUE})$. Here is one: let $e_A^2 = \mathcal{W} \times 2^{\mathcal{W}}$. Clearly $e_A^2, \{\}, w \models \boldsymbol{O}_A(\text{TRUE})$. It is easy to verify that no other e^2 satisfies $\boldsymbol{O}_A(\text{TRUE})$.

   Now $e_A^2, \{\}, w \models \neg\boldsymbol{K}_A\neg\boldsymbol{K}_B\alpha$ iff there is some $(w', e_B^1) \in e_A^2$ such that $e_A^2, e_B^1, w' \models \boldsymbol{K}_B\alpha$. By construction, there is $(w, e^{*1}_B) \in e_A^2$ where $e^{*1}_B = \{w \mid w \models \alpha\}$ and $e_A^2, e^{*1}_B, w \models \boldsymbol{K}_B\alpha$.

2. $\boldsymbol{O}_A(\text{TRUE}) \supset \neg\boldsymbol{K}_A\boldsymbol{K}_B\alpha$.

   Construct e_A^2 as in item 1. Then $e_A^2, \{\}, w \models \neg\boldsymbol{K}_A\boldsymbol{K}_B\alpha$ iff there is some $(w', e_B^1) \in e_A^2$, such that $e_A^2, e_B^1, w' \models \neg\boldsymbol{K}_B\alpha$. By construction, $(w, e^{*1}_B) \in e_A^2$ where $e^{*1}_B = \{w \mid w \not\models \alpha\}$ and, $e_A^2, e^{*1}_B, w \models \neg\boldsymbol{K}_B\alpha$.

3. $\boldsymbol{O}_A(\alpha \wedge \boldsymbol{O}_B\alpha) \supset \boldsymbol{K}_A\alpha$.

   We will consider any 2-structure satisfying $\boldsymbol{O}_A(\alpha \wedge \boldsymbol{O}_B\alpha)$ and prove that $\boldsymbol{K}_A\alpha$ is also satisfied at the structure.

   Let $\mathcal{W}_\alpha = \{w \mid w \models \alpha\}$. Clearly $e_B^1 = \mathcal{W}_\alpha$ is the only 1-structure for B that satisfies $\boldsymbol{O}_B\alpha$. Similarly, the 2-structure $e_A^2 = \mathcal{W}_\alpha \times \{e_B^1\}$ is the only 2-structure for A that satisfies $\boldsymbol{O}_A(\alpha \wedge \boldsymbol{O}_B\alpha)$. It follows that $e_A^2, \{\}, w \models \boldsymbol{K}\alpha$ since all w' in $(w', e_B^1) \in e_A^2$ satisfy α by construction.

4. $\boldsymbol{O}_A(\alpha \wedge \boldsymbol{O}_B\alpha) \supset \boldsymbol{K}_A\boldsymbol{K}_B\alpha$.

   A 2-structure e_A^2 is constructed as in item 3. Then it follows that $e_A^2, \{\}, w \models \boldsymbol{K}_A\boldsymbol{K}_B\alpha$ since all worlds $\{w'' \mid (w'', \{\}) \in e_B^1$ and $(w', e_B^1) \in e_A^2$ for some $w'\}$ satisfy α by construction.

5. $\boldsymbol{O}_A(\alpha \wedge \boldsymbol{O}_B\alpha) \supset (\boldsymbol{K}_A\neg\boldsymbol{K}_B\boldsymbol{K}_A\alpha \wedge \boldsymbol{K}_A\neg\boldsymbol{K}_B\neg\boldsymbol{K}_A\alpha)$.

Using ideas from item 1 and item 2, it follows that $\boldsymbol{O}_B\alpha \supset \neg\boldsymbol{K}_B\boldsymbol{K}_A\alpha \wedge \neg\boldsymbol{K}_B\neg\boldsymbol{K}_A\alpha$ is valid. Let e^3_A be any structure that satisfies $\boldsymbol{O}_A(\alpha \wedge \boldsymbol{O}_B\alpha)$. Since for all $(w', e^2_B) \in e^3_A$, $e^3_A, e^2_B, w' \models \alpha \wedge \boldsymbol{O}_B\alpha$, it follows that $e^3_A, e^2_B, w' \models \neg\boldsymbol{K}_B\boldsymbol{K}_A\alpha \wedge \neg\boldsymbol{K}_B\neg\boldsymbol{K}_A\alpha$. Therefore $e^3_A, \{\}, w \models \boldsymbol{K}_A(\neg\boldsymbol{K}_B\boldsymbol{K}_A\alpha \wedge \neg\boldsymbol{K}_B\neg\boldsymbol{K}_A\alpha)$. ∎

3.2.3 On Validity

While it seems perfectly reasonable to expect models of a certain depth to interpret formulas of a corresponding depth, it is also the case that the validity of formulas is not affected when models of a higher depth than that required are considered. That is, if a formula of maximal A,B-depth k, j is true at all (k, j)-models, then the formula is also true at all (k', j')-models, for $k' \geq k$ and $j' \geq j$. To demonstrate this property, we construct for every $e^{k'}_A$, a k-structure $e_A{\downarrow}^{k'}_k$, such that they agree on all formulas of maximal A-depth k. Analogously, a j-structure that agrees on all formulas of maximal B-depth j can be constructed for every $e^{j'}_B$.

Definition 3.2.5. Given $e^{k'}_A$, we define a k-structure $e_A{\downarrow}^{k'}_k$ for $k' \geq k \geq 1$:

1. $e_A{\downarrow}^1_1 = e^1_A$,
2. $e_A{\downarrow}^{k'}_1 = \{(w, \{\}) \mid (w, e^{k'-1}_B) \in e^{k'}_A\}$,
3. $e_A{\downarrow}^{k'}_k = \{(w, e_B{\downarrow}^{k'-1}_{k-1}) \mid (w, e^{k'-1}_B) \in e^{k'}_A\}$. ∎

We now establish the relationship between k'-structures $e^{k'}_A$ and j'-structures $e^{j'}_B$, and their corresponding k-structures $e_A{\downarrow}^{k'}_k$ and j-structures $e_B{\downarrow}^{j'}_j$ respectively. We begin by showing that the A-depth and B-depth of a formula are closely related.

Lemma 3.2.6. *Suppose α is a formula and $|\alpha|_i = k$. Then $|\alpha|_j \in \{k-1, k, k+1\}$.*

Proof: Let i be A. The argument is symmetric if i is B. The proof is by induction on α.

case *atoms*: The A-depth and B-depth of atoms is 1, and so the claim holds.

case $\neg\alpha$: Suppose $|\neg\alpha|_A = k$. Then, by definition, $|\alpha|_A = k$. By induction hypothesis, $|\alpha|_B \in \{k-1, k, k+1\}$. To prove the claim we need to show that B-depth of $\neg\alpha$ is one of $\{k-1, k, k+1\}$. Since $|\neg\alpha|_B = |\alpha|_B$, the case is proved.

case $\forall x\alpha$: The holds in an analogous manner to the case of negations.

case $\alpha \vee \beta$: Suppose $|\alpha \vee \beta|_A = k$. That is, $max(|\alpha|_A, |\beta|_A) = k$. By induction, if the A-depth of γ is k, where $\gamma \in \{\alpha, \beta\}$, then its B-depth is in the range $\{k-1, k, k+1\}$. By assumption, one of α or β has A-depth k.

So consider $|\alpha \vee \beta|_B = max(|\alpha|_B, |\beta|_B)$ which, by above, is in the range $\{k-1, k, k+1\}$.

case $\boldsymbol{M}_i\alpha$, where $\boldsymbol{M}_i \in \{\boldsymbol{K}_A, \boldsymbol{K}_B, \boldsymbol{N}_A, \boldsymbol{N}_B\}$: Suppose $|\boldsymbol{K}_A\alpha|_A = k$. By definition of i-depth, $|\alpha|_A = k$. Now consider $|\boldsymbol{K}_A\alpha|_B$ which, by definition, equals $1 + |\alpha|_A = k + 1$. The treatment of $\boldsymbol{N}_A\alpha$ is analogous.

Suppose $|\boldsymbol{K}_B\alpha|_A = k$. By definition of i-depth, $|\alpha|_B = k-1$. Now consider $|\boldsymbol{K}_B\alpha|_B$ which, by definition, equals $|\alpha|_B = k - 1$. The treatment of $\boldsymbol{N}_B\alpha$ is analogous. ∎

Lemma 3.2.7. *Suppose α is a formula and $|\alpha|_i = k$. Then for every subformula $\boldsymbol{M}_i\beta$ in α, $|\boldsymbol{M}_i\beta|_i \leq k$ and for every subformula $\boldsymbol{M}_j\gamma$ in α, for $j \neq i$, $|\boldsymbol{M}_j\gamma|_j < k$.*

Proof: Let i be A. The argument is symmetric if i is B. The proof is by induction on α.

case *atoms*: since atoms do not mention modalities, the lemma is vacuously true.

case $\neg\alpha, \forall x\alpha, \alpha \vee \beta$: easy to show these cases by induction.

case $\boldsymbol{M}_i\alpha$: Suppose $|\boldsymbol{M}_i\alpha|_A = k$. We now prove that the lemma holds for $\boldsymbol{M}_i\alpha$ assuming it holds for α.

Suppose $\boldsymbol{M}_i$ is $\boldsymbol{K}_A$. Since $|\boldsymbol{K}_A\alpha|_A = k$, by definition, $|\alpha|_A = k$. By induction, for every subformula $\boldsymbol{M}_A\beta$ in α, we have $|\boldsymbol{M}_A\beta|_A \leq k$ and for every subformula $\boldsymbol{M}_B\gamma$ in α, we have $|\boldsymbol{M}_B\gamma|_B < k$. To prove the claim, we look at lengthier subformulas from $\boldsymbol{K}_A\alpha$, *i.e.* $\boldsymbol{K}_A\boldsymbol{M}_A\beta$ and $\boldsymbol{K}_A\boldsymbol{M}_B\gamma$. More precisely, to prove the claim we need to show that the A-depth of $\boldsymbol{K}_A\boldsymbol{M}_A\beta$ and $\boldsymbol{K}_A\boldsymbol{M}_B\gamma$ is $\leq k$. So consider $|\boldsymbol{K}_A\boldsymbol{M}_A\beta|_A$ which equals, by definition, $|\boldsymbol{M}_A\beta|_A \leq k$ by induction. Moving on, consider $|\boldsymbol{K}_A\boldsymbol{M}_B\gamma|_A$ which equals, by definition, $|\boldsymbol{M}_B\gamma|_A$ which equals, by definition, $1 + |\gamma|_B$. By induction, $|\boldsymbol{M}_B\gamma|_B < k$, say $k-1$, which implies that $|\gamma|_B = k-1$ by definition. Therefore $|\boldsymbol{K}_A\boldsymbol{M}_B\gamma|_A = 1 + (k-1) = k$. This proves the case of $\boldsymbol{M}_i$ being $\boldsymbol{K}_A$. The case of $\boldsymbol{M}_i$ being $\boldsymbol{N}_A$ is analogous.

Suppose $\boldsymbol{M}_i$ is $\boldsymbol{K}_B$. Since $|\boldsymbol{K}_B\alpha|_A = k$, by definition $|\alpha|_B = k-1$. By induction, for every subformula $\boldsymbol{M}_B\gamma$ in α we have $|\boldsymbol{M}_B\gamma|_B \leq k-1$ and for every subformula $\boldsymbol{M}_A\beta$ in α we have $|\boldsymbol{M}_A\beta|_A < k-1$. To prove the claim, we look at lengthier subformulas in $\boldsymbol{K}_B\alpha$, *i.e.* $\boldsymbol{K}_B\boldsymbol{M}_B\gamma$ and $\boldsymbol{K}_B\boldsymbol{M}_A\beta$. More precisely, to prove the claim we need to show that the B-depth of $\boldsymbol{K}_B\boldsymbol{M}_B\gamma$ and $\boldsymbol{K}_B\boldsymbol{M}_A\beta$ is $\leq k-1$. So consider $|\boldsymbol{K}_B\boldsymbol{M}_B\gamma|_B$ which equals, by definition, $|\boldsymbol{M}_B\gamma|_B \leq k-1$ by induction. Moving on, consider $|\boldsymbol{K}_B\boldsymbol{M}_A\beta|_B$ which equals, by definition, $|\boldsymbol{M}_A\beta|_B$ which equals, by definition, $1 + |\beta|_A$. By induction, $|\boldsymbol{M}_A\beta|_A < k-1$, say $k-2$. This implies that $|\beta|_A = k-2$ by definition. Therefore $|\boldsymbol{K}_B\boldsymbol{M}_A\beta|_B = 1 + (k-2) = k-1$. This proves the case of $\boldsymbol{M}_i$ being $\boldsymbol{K}_B$. The case of $\boldsymbol{M}_i$ being $\boldsymbol{N}_B$ is analogous. ∎

Lemma 3.2.8. *Suppose α is a formula such that all subformulas $\boldsymbol{M}_A\beta$ in α are of maximal A-depth k and all subformulas $\boldsymbol{M}_B\gamma$ in α are of maximal B-depth j. Then for $k' \geq k$, $j' \geq j$:*

$$e_A^{k'}, e_B^{j'}, w \models \alpha \text{ iff } e_A{\downarrow_k^{k'}}, e_B{\downarrow_j^{j'}}, w \models \alpha.$$

Proof: The proof is by induction on α.

case *atoms*: since we have the same world in both the models, the argument is trivial.

case $\neg\alpha$: We have $e_A^{k'}, e_B^{j'}, w \models \neg\alpha$

iff $e_A^{k'}, e_B^{j'}, w \not\models \alpha$ by definition

iff $e_A{\downarrow_k^{k'}}, e_B{\downarrow_j^{j'}}, w \not\models \alpha$ by induction

iff $e_A{\downarrow_k^{k'}}, e_B{\downarrow_j^{j'}}, w \models \neg\alpha$ by definition.

case $\forall x\alpha$: Straightforward.

case $\alpha \vee \beta$: We have $e_A^{k'}, e_B^{j'}, w \models \alpha \vee \beta$

iff $e_A^{k'}, e_B^{j'}, w \models \alpha$ or $e_A^{k'}, e_B^{j'}, w \models \beta$

iff $e_A{\downarrow_k^{k'}}, e_B{\downarrow_j^{j'}}, w \models \alpha$ or $e_A{\downarrow_k^{k'}}, e_B{\downarrow_j^{j'}}, w \models \beta$ by induction

iff $e_A{\downarrow_k^{k'}}, e_B{\downarrow_j^{j'}}, w \models \alpha \vee \beta$.

case $\boldsymbol{K}_A\alpha$: Suppose $|\boldsymbol{K}_A\alpha|_A = k$. Then, by definition, $|\alpha|_A = k$. We have $e_A^{k'}, e_B^{j'}, w \models \boldsymbol{K}_A\alpha$

iff $e_A^{k'}, e_B^{k'-1}, w' \models \alpha$ for all $(w', e_B^{k'-1}) \in e_A^{k'}$ by definition

iff $e_A{\downarrow_k^{k'}}, e_B{\downarrow_{k-1}^{k'-1}}, w' \models \alpha$ for all $(w', e_B{\downarrow_{k-1}^{k'-1}}) \in e_A{\downarrow_k^{k'}}$ by induction (since from Lemma 3.2.7 subformulas $\boldsymbol{M}_A\beta$ in α have maximal A-depth k and subformulas $\boldsymbol{M}_B\gamma$ in α have maximal B-depth $k-1$)

iff $e_A{\downarrow_k^{k'}}, \{\}, w \models \boldsymbol{K}_A\alpha$ by definition

iff $e_A{\downarrow_k^{k'}}, e_B{\downarrow_{k-1}^{j'}}, w \models \boldsymbol{K}_A\alpha$ since B's epistemic state is irrelevant.

The cases of $\boldsymbol{N}_A\alpha$, $\boldsymbol{K}_B\alpha$ and $\boldsymbol{N}_B\alpha$ are analogous. ∎

Lemma 3.2.9. *Let $k' \geq k$, $j' \geq j$. For all formulas α of A-depth of k and B-depth j:*

$$e_A^{k'}, e_B^{j'}, w \models \alpha \quad \textit{iff} \quad e_A{\downarrow_k^{k'}}, e_B{\downarrow_j^{j'}}, w \models \alpha.$$

Proof: Suppose we are given a formula α of A-depth k. By Lemma 3.2.6, $|\alpha|_B \in \{k-1, k, k+1\}$. That is, $j \in \{k-1, k, k+1\}$. From Lemma 3.2.7, we know that all subformulas $\boldsymbol{M}_A\beta$ in α have maximal A-depth k and all subformulas $\boldsymbol{M}_B\gamma$ in α have maximal A-depth $< k$, *i.e.* $\leq k-1$. Then $e_A^{k'}, e_B^{j'}, w \models \alpha$

iff $e_A{\downarrow_k^{k'}}, e_B{\downarrow_{k-1}^{j'}}, w \models \alpha$ by Lemma 3.2.8

iff $e_A{\downarrow_k^{k'}}, e_B{\downarrow_j^{j'}}, w \models \alpha$ by Lemma 3.2.8, since $(e_B{\downarrow_j^{j'}}){\downarrow_{k-1}^{j}}$ is $e_B{\downarrow_{k-1}^{j'}}$ given $j' \geq j \geq k-1$. ∎

Theorem 3.2.10. *For all formulas α of A, B-depth of k, j, if α is true at all (k, j)-models, then α is true at all (k', j')-models, where $k' \geq k$ and $j' \geq j$.*

Proof: Suppose α is true at all (k, j)-models. Given any (k', j')-model by assumption $e_A{\downarrow_k^{k'}}, e_B{\downarrow_j^{j'}}, w \models \alpha$. Then by way of Lemma 3.2.9 we have $e_A^{k'}, e_B^{j'}, w \models \alpha$. ∎

3.2.4 A Limitation

Let us briefly reflect on the fact that k-structures have finite depth. So suppose A only knows Σ, of depth k. Using k-structures alone allows us to reason about what is believed and what is not believed, up to depth k. Moreover, as already observed in Example 3.2.4, the logic correctly captures that A is ignorant about beliefs at depth greater than k. That is, using the simple example of an agent who only knows TRUE, which of depth 1, we saw that both the sentences $\boldsymbol{O}_A(\text{TRUE}) \supset \neg\boldsymbol{K}_A\neg\boldsymbol{K}_B\alpha$ and $\boldsymbol{O}_A(\text{TRUE}) \supset \neg\boldsymbol{K}_A\boldsymbol{K}_B\alpha$ are valid.

So, although the KB has finite depth, we are able to ask queries α of any depth in the sense of determining whether $\boldsymbol{O}_i\Sigma \supset \boldsymbol{K}_i\alpha$ is valid.

For most purposes, this restriction of having a parameter k seems harmless in the sense that agents usually have a finite knowledge base with sentences of some maximal depth k and they should not be able to conclude anything about what is known at depths higher than k. But there is one aspect which previous approaches to multiagent only knowing can handle, but we cannot: the property of simultaneously satisfying an infinite set of sentences of unbounded depth. Indeed, k-structures cannot be used for this purpose simply because, for a fixed k, the satisfaction relation is undefined for formulas beyond depth k.

One prominent application of such a property is the notion of *common knowledge*. To illustrate the idea briefly, let us write $\boldsymbol{E}\alpha$ to syntactically mean $\boldsymbol{K}_A\alpha \wedge \boldsymbol{K}_B\alpha$. The intuitive reading of $\boldsymbol{E}\alpha$ is that both A and B know α, that is, "*everybody* knows α". Now, let $\boldsymbol{E}^0\alpha$ be an abbreviation for α, and let $\boldsymbol{E}^{k+1}$ be an abbreviation for $\boldsymbol{E}\boldsymbol{E}^k\alpha$. Then α is said to be common knowledge, written $\boldsymbol{C}\alpha$, if $\boldsymbol{E}^k\alpha$ for $k = 1, 2\ldots$ While the nature of $\boldsymbol{C}$ is infinitary, in the sense that it essentially corresponds to an infinite conjunction, it can nonetheless be given a finite axiomatic characterization [Fagin et al., 1995], making it a useful operator for reasoning in distributed systems and games.

Thus, if we were to include the notion of common knowledge in a logic, then we would get entailments about what is believed at arbitrary depths. With our current model, however, this cannot be captured. While this is certainly a restriction, we are willing to pay that price because in return we get, for the first time, a very simple possible-world style account of only knowing for many agents.

3.2.5 Properties of Knowledge

Knowledge with k-structures satisfy $\mathbf{K45}_n$ properties as well as the Barcan formula. That is, k-structures exhibit the same properties as $\mathcal{OL}$, generalized to many agents.

Lemma 3.2.11. *If α is a formula, the following are valid wrt models of appropriate depth ($\boldsymbol{M}_i$ denotes $\boldsymbol{K}_i$ or $\boldsymbol{N}_i$):*

1. $\boldsymbol{M}_i\alpha \wedge \boldsymbol{M}_i(\alpha \supset \beta) \supset \boldsymbol{M}_i\beta$,
2. $\boldsymbol{M}_i\alpha \supset \boldsymbol{M}_i\boldsymbol{M}_i\alpha$,
3. $\neg\boldsymbol{M}_i\alpha \supset \boldsymbol{M}_i\neg\boldsymbol{M}_i\alpha$,
4. $\forall\vec{x}\,\boldsymbol{M}_i\alpha \supset \boldsymbol{M}_i\forall\vec{x}\,\alpha$,
5. $\exists\vec{x}\,\boldsymbol{M}_i\alpha \supset \boldsymbol{M}_i\exists\vec{x}\,\alpha$.

Proof: The proofs are very similar. We demonstrate item 3 and item 4. Let $\boldsymbol{M}_i = \boldsymbol{K}_A$.

3. Suppose $e_A^k, e_B^j, w \models \neg\boldsymbol{K}_A\alpha$. Then there is some $(w', e_B^{k-1}) \in e_A^k$ such that $e_A^k, e_B^{k-1}, w' \models \neg\alpha$. Let w'' be any world such that $(w'', {e'}_B^{k-1}) \in e_A^k$. Then $e_A^k, {e'}_B^{k-1}, w'' \models \neg\boldsymbol{K}_A\alpha$. Therefore $e_A^k, e_B^j, w \models \boldsymbol{K}_A\neg\boldsymbol{K}_A\alpha$.
4. Suppose $e_A^k, e_B^j, w \models \forall x\boldsymbol{K}_A\alpha$. Then $e_A^k, e_B^j, w \models (\boldsymbol{K}_A\alpha)_n^x$ for every name n. That is, $e_A^k, e_B^j, w \models \boldsymbol{K}_A\alpha_n^x$ for every n. Then for all $(w', e_B^{k-1}) \in e_A^k$, $e_A^k, e_B^{k-1}, w' \models \alpha_n^x$ for every name n iff $e_A^k, e_B^{k-1}, w' \models \forall x\alpha$ by definition. Therefore $e_A^k, e_B^j, w \models \boldsymbol{K}_A\forall x\alpha$.

The case of $\boldsymbol{N}_A$ is analogous, where the argument ranges over all k-structures not in e_A^k. ∎

This concludes our presentation of a semantics for $\mathcal{OL}_n$. We now return to the features of $\mathcal{OL}$ emphasized in Section 3.2.1, and talk about their generalization.

3.2.6 Generalizing the Features of $\mathcal{OL}$

In this section, we are concerned with arguing that the semantics for $\mathcal{OL}_n$ generalizes the features of $\mathcal{OL}$ in an suitable manner. Perhaps the most appropriate way to begin is by reviewing how earlier approaches attempted to capture and generalize $\mathcal{OL}$'s features. Lakemeyer [1993] and Halpern [1993] independently attempted to extend $\mathcal{OL}$ to the many agent case. Both approaches provide a semantics by means of multiagent Kripke structures.[6] There are some subtle differences in these proposals, but the main restriction is that they only allow a propositional language. Henceforth, to make the comparison feasible, we shall also speak of the propositional subset of $\mathcal{OL}_n$ with the understanding that the semantical framework is now defined over an infinite number of propositions rather than ground atoms.[7]

Lakemeyer's approach is based on a $\mathbf{K45}_n$ canonical model, which is a Kripke structure whose worlds are all maximally consistent (wrt the axioms of the modal logic $\mathbf{K45}_n$) subsets of basic formulas (*cf.* Definition 3.3.2 and Definition 3.3.3). Among its main criticisms is that canonical models cannot be used in a practical way. Not only are there an infinite number of worlds, but each world is characterized by an infinite set of formulas and so cannot be described easily. Therefore, this approach is only of theoretical interest, that is, to clarify if a reasonable semantics for only knowing can be given in the multiagent case. Moreover, since the semantics is based on proof-theoretic machinery, in the sense of being based on maximally consistent sets of formulas, the approach is also not natural in the usual sense where a semantics independently justifies truth in a logic. Following [Halpern and Lakemeyer, 2001], we refer to this approach as the *canonical model* approach. Independently, Halpern proposed another Kripke structure approach. Although he did not restrict his attention to canonical models, arguments can be provided as to why this approach also cannot be used in a practical way [Halpern and Lakemeyer, 2001]. Therefore, yet again, the approach is only of theoretical interest. Following [Halpern and Lakemeyer, 2001], we refer to Halpern's approach as the *i-set* approach.

In both proposals, the fundamental concern is about the notion of an *epistemic possibility*. As discussed earlier, the appropriate generalization of a possibility in the many agent case is a set of *i-objective formulas*. The question, then, is which set of i-objective formulas represent the epistemic possibilities of the agent? To answer that, Halpern and Lakemeyer proceed as follows. Given a Kripke structure $M = (W, \pi, \mathcal{K}_A, \mathcal{K}_B)$ and a world $w \in W$, epistemic possibilities are obtained as the following set of formulas:

$$Obj_i(M, w) = \{obj_i(M, w') \mid w' \in \mathcal{K}_i(w)\}$$

where $obj_i(M, w')$ is the set of i-objective formulas that are satisfied at (M, w').[8] With this in hand, Halpern and Lakemeyer examine the faithfulness of their approaches wrt $\mathbf{P1} - \mathbf{P3}$ in [Halpern and Lakemeyer, 2001].

[6] In his original formulation, Halpern [1993] uses a different but equivalent model theory. Our presentation is based on discussions in [Halpern and Lakemeyer, 2001].

[7] Other than [Lakemeyer, 1993] and [Halpern, 1993], generalizations of $\mathcal{OL}$ are considered in [Halpern and Lakemeyer, 2001] and [Waaler and Solhaug, 2005]. They are also propositional and based on Kripke structures, but since they are motivated by the proof theory discussions on these approaches are deferred to after we review the axiomatization.

[8] Note that in the single agent case i-objective formulas are simply *objective formulas*, and so in this sense, $Obj_i(M, w)$ generalizes the single agent case in the context of Kripke structures.

We do not discuss this in detail here, but they find that the canonical model approach does not satisfy **P3** and that the i-set approach does not satisfy **P2**. Consequently, these approaches show some peculiar properties, which we will look at shortly.

Let us remark that while at first glance the definition of epistemic possibilities as represented by the set $Obj_i(M, w)$ certainly seems intuitive, even for propositional $\mathcal{OL}_n$, a Kripke model is a completely different entity from what Levesque supposes. Perhaps, one consequence is that the semantic proofs in earlier approaches are very involved. In contrast, our underlying possible worlds framework follows Levesque. Here is how we capture the notion of a possibility (and epistemic possibilities):

Definition 3.2.12. Suppose $M = (e_A^k, e_B^j, w)$ is a (k, j)-model. Let

1. $obj_A(M) = \{A\text{-objective } \phi \text{ of } B\text{-depth} \leq j \mid M \models \phi\}$;
2. $obj_B(M) = \{B\text{-objective } \phi \text{ of } A\text{-depth} \leq k \mid M \models \phi\}$;
3. $Obj_A(e_A^k) = \{obj_A(\{\}, e_B^{k-1}, w) \mid (w, e_B^{k-1}) \in e_A^k\}$;
4. $Obj_B(e_B^j) = \{obj_B(e_A^{j-1}, \{\}, w) \mid (w, e_A^{j-1}) \in e_B^j\}$. ∎

To see the intuition behind the definition, suppose that we are interested in the set of A-objective formulas true at the model M. Clearly, the objective formulas true at w are to be included in this set, as are the B-subjective formulas that hold wrt the j-structure e_B^j in M. Note that these formulas do not actually capture A's possibilities, which is, in fact, determined strictly by the k-structure e_A^k. Hence, we define $Obj_A(e_A^k)$ as the set of all A-objectives formulas that A considers possible, which is obtained from the set of all k-structures $(w, e_B^{k-1}) \in e_A^k$. With this cleared up, we now argue that an appropriate generalization of $\mathcal{OL}$ is satisfied in our semantical framework by means of features **P1**, **P2** and **P3** from Section 3.2.1.

Generalizing P1:

When a single agent is involved, this property ensures that epistemic possibilities are not affected when evaluating $\boldsymbol{N}$. To see why this idea holds for $\mathcal{OL}_n$ in a straightforward manner, consider any model $M = (e_A^k, e_B^j, w)$. Then, A's possibilities is given as $Obj_A(e_A^k)$. Now, to evaluate formulas of the form $\boldsymbol{N}_A\alpha$, we are interested in the set of models

$$\mathcal{M} = \{(e_A^k, e_B^{k-1}, w') \mid (w', e_B^{k-1}) \notin e_A^k\}.$$

Observe that A's possibilities in every $M' \in \mathcal{M}$ is given also as $Obj_A(e_A^k)$. Therefore, epistemic possibilities are not affected on evaluating $\boldsymbol{N}_i$.

Generalizing P2:

Here we are concerned with the property that the evaluation of $\boldsymbol{K}_i\alpha$ and $\boldsymbol{N}_i\alpha$ is wrt the set of all possibilities. Moreover, the set of all possibilities is independent of the epistemic state.

To demonstrate this property, let α be any A-objective formula, say of maximal B-depth k, which represents a possibility in A's view. Then let e_A^{k+1} be any epistemic state satisfying $\boldsymbol{K}_A\alpha$. By analogy to the single

agent case, the set of models used to interpret $\boldsymbol{N}_A\alpha$ must be the exact complement of e_A^{k+1}, in the sense that these models together with e_A^{k+1} is the full set of conceivable states. But since epistemic notions are defined with regards to the depth of formulas, the set of conceivable states of depth $k+1$ is $\mathbb{E}^{k+1}$. This is precisely what we establish with the following proposition, whose proof is a rather direct one by the definition of the semantics. Moreover, $\mathbb{E}^{k+1}$ is independent of e_A^{k+1}.

Proposition 3.2.13. *Let α be A-objective of B-depth k. Then, the set of $(k+1)$-structures that evaluate $\boldsymbol{K}_A\alpha$ and $\boldsymbol{N}_A\alpha$ is $\mathbb{E}^{k+1}$. (Analogously stated for B.)*

Generalizing P3:

The third property, by analogy to single agent case, must allow us to characterize epistemic states from any set of i-objective formulas. Intuitively, given such a set of formulas, we must have a model where precisely this set represents the beliefs of an epistemic state. Since every set of formulas can be extended to a maximally consistent set of formulas, it suffices to show that there is an epistemic state corresponding to every set of maximally consistent sets of formulas.

There are two problems, however. The first is regarding the depth of formulas in a maximally consistent set. Usually, the notion of a maximally consistent set (see Definition 3.3.2) does not place restrictions on the nesting of modal operators in a formula, *i.e.* the depth of the formula is unbounded. In the case of **P2** above, we restricted our notion of possibility to A-objective formulas of a certain depth. In similar fashion, it seems reasonable to restrict ourselves to maximally consistent sets of a certain depth. The second issue is more significant, however. Consistency is implicitly coupled with an axiom system. The approach taken by Lakemeyer [1993] is to appeal to the axiom system of $\mathbf{K45}_n$, and he basically shows that epistemic states corresponding to any set of basic maximally consistent sets of formulas can be constructed in his formalism. But defining possibilities via $\mathbf{K45}_n$ proof-theoretic machinery inevitably leads to some limitations, as we shall see.

Instead of unnecessarily complicating matters at this point as to what the right notion of consistency should look like, we define an equivalent notion of *maximally satisfiable set* of formulas. This is a purely semantical notion, and should be seen a semantically characterized *complete* description of a possibility, analogous to the proof-theoretically characterized notion of a maximally consistent set of formulas. The idea is this: Let Σ be a satisfiable set of A-objective (not necessarily basic) formulas, say of maximal B-depth k. Let γ be a A-objective formula of maximal B-depth k. If $\Sigma \cup \{\gamma\}$ is satisfiable, then let $\Sigma_1 = \Sigma \cup \{\gamma\}$. Otherwise, let $\Sigma_1 = \Sigma$. By considering all A-objective formulas of maximal B-depth k, construct $\Sigma_2, \ldots$, and let Σ^* be the limit. We term a set of formulas constructed in this fashion as a *maximally satisfiable A-objective set* of formulas. (There may be many maximally satisfiable A-objective sets corresponding to Σ depending on our choice of γ and the subsequent formulas added.)

We now show that given *any* set of maximally satisfiable i-objective sets, there is a model where precisely this set characterizes the epistemic state.

Theorem 3.2.14. *Let S_i be a set of maximally satisfiable sets of i-objective formulas, and σ a satisfiable objective formula. Suppose S_A is of maximal B-depth k and S_B is of maximal A-depth j. Then there is a model $M^* = (e^{*k+1}_A, e^{*j+1}_B, w^*)$ such that $M^* \models \sigma$, $S_A = Obj_A(e^{*k+1}_A)$ and $S_B = Obj_B(e^{*j+1}_B)$.*

Proof: Consider S_A. Each $S' \in S_A$ is a maximally satisfiable A-objective set and thus by definition, there is a $(k+1)$-structure (w', e_B^k) such that $\{\}, e_B^k, w' \models S'$. Let

$$e_A^{*k+1} = \{(w', e_B^k) \mid \{\}, e_B^k, w' \models S' \text{ and } S' \in S_A\}.$$

It is immediate to verify that $Obj_A(e_A^{*k+1}) = S_A$. In an analogous fashion construct e_B^{*j+1} from S_B. Finally by assumption σ is satisfiable in some world, say w^*. ∎

To summarize, arguably, a generalized variant of Levesque's properties are satisfied in the semantical framework. To add further support to this claim, we now present the unintuitive properties exhibited by the canonical model and the i-set approaches. Meanwhile, we show that our approach does not suffer from these problems.

Lakemeyer [1993] noted that certain types of epistemic states cannot be constructed in his approach. This is a consequence of the approach not satisfying **P3**. More precisely,

Proposition 3.2.15. *[Lakemeyer, 1993]*
For any proposition p and $i \neq j$, $\neg \boldsymbol{O}_i \neg \boldsymbol{O}_j p$ is valid in the canonical model approach.

Intuitively, for $i = A$ and $j = B$, it says that all that Alice knows is that Bob does not only know p, and as Lakemeyer admits, the validity of $\neg \boldsymbol{O}_A \neg \boldsymbol{O}_B p$ is unintuitive. After all, Bob could *honestly* tell Alice that he does not only know p.

In contrast, we first prove that the formula $\boldsymbol{O}_i \neg \boldsymbol{O}_j p$, which was not satisfiable in Lakemeyer's approach, is indeed satisfiable in our approach.[9]

Proposition 3.2.16. *For any proposition p and $i \neq j$, $\boldsymbol{O}_i \neg \boldsymbol{O}_j p$ is satisfiable (in a model of appropriate depth).*

Proof: Let i be A. (The argument is symmetric if i is B.) Let $\mathcal{W}_p = \{w \mid w \models p\}$ and let $E = 2^{\mathcal{W}} - \{\mathcal{W}_p\}$. It is easy to see that if $e_B^1 \in E$, then $\{\}, e_B^1, w \not\models \boldsymbol{O}_B p$ for any world w. Now define $e_A^2 = \mathcal{W} \times E$. Then $e_A^2, \{\}, w \models \boldsymbol{O}_A \neg \boldsymbol{O}_B p$. ∎

From a technical viewpoint, as we noted when discussing our generalization of **P3**, Lakemeyer restricts the notion of a i-objective possibility to a maximally $\mathbf{K45}_n$-consistent set of *basic* i-objective formulas. Unfortunately, there is more to agent i's possibility than just basic formulas. The restriction to basic formulas is an artifact of a semantics based on the canonical model. This suggests that Lakemeyer makes an unavoidable technical commitment. In contrast, Theorem 3.2.14 shows that we allow non-basic formulas and by using a strictly semantic notion, we avoid problems that arise from $\mathbf{K45}_n$ proof-theoretic restrictions.

Let us turn to the problem with the i-set approach. We mentioned earlier that this approach does not satisfy **P2**. In fact, it can be shown that:

Proposition 3.2.17. *[Halpern and Lakemeyer, 2001]*
For any proposition p and $i \neq j$, $\boldsymbol{N}_i \neg \boldsymbol{O}_j p \wedge \boldsymbol{K}_i \neg \boldsymbol{O}_j p$ is satisfiable in the i-set approach.

[9] We remark that because Halpern's approach satisfies **P3**, Proposition 3.2.16 is also provable in his framework.

Recall that this property requires the evaluation of $\boldsymbol{K}_i\alpha$ and $\boldsymbol{N}_i\alpha$ to consider all conceivable states. So the satisfiability of the above sentence leaves open the question as to why $\boldsymbol{O}_j p$ is not considered since $\neg\boldsymbol{O}_j p$ is true at all conceivable states.

We now prove that the formula $\boldsymbol{N}_i\neg\boldsymbol{O}_j p \wedge \boldsymbol{K}_i\neg\boldsymbol{O}_j p$ is *not satisfiable* wrt our semantics, in contrast to Proposition 3.2.17, as should be the case.

Proposition 3.2.18. *For any proposition p and $i \neq j$, $\boldsymbol{N}_i\neg\boldsymbol{O}_j p \wedge \boldsymbol{K}_i\neg\boldsymbol{O}_j p$ is not satisfiable.*

Proof: Let i be A, with the other case being symmetric. Suppose $e^k_A, \{\}, w \models \boldsymbol{K}_A\neg\boldsymbol{O}_B p$ for any w. Then for all $(w', e^{k-1}_B) \in e^k_A$, $e^k_A, e^{k-1}_B, w' \models \neg\boldsymbol{O}_B p$. Since $\boldsymbol{O}_B p$ is satisfiable, there is a $e^{*\,k-1}_B$ such that $\{\}, e^{*\,k-1}_B, w'' \models \boldsymbol{O}_B p$ for any w''. By assumption $(w'', e^{*\,k-1}_B) \notin e^k_A$. Therefore $e^k_A, \{\}, w \models \neg\boldsymbol{N}_A\neg\boldsymbol{O}_B p$. ∎

With the i-set approach the problem seems to be that $\boldsymbol{K}_i$ and $\boldsymbol{N}_i$ do not interact naturally, and that the full complement of epistemic possibilities is not considered in interpreting $\boldsymbol{N}_i$. In our case, however, since the semantics faithfully complies with **P2**, the sentence $\boldsymbol{N}_A\neg\boldsymbol{O}_B p \wedge \boldsymbol{K}_A\neg\boldsymbol{O}_B p$ is not satisfiable. Thus, it seems that a semantics with k-structures satisfies our intuitions about only knowing.

3.3 Proof Theory

Naturally, the next question is if there are axioms that characterize the semantics. We show that the answer is affirmative. However, obtaining that axiomatization is not entirely straightforward. We proceed as follows. We begin with an axiomatization proposed by Lakemeyer [1993], which is sound and complete for both the canonical model and the i-set approach, but for a restricted language. We then use Lakemeyer's proof theory to devise a new one for the complete language.

3.3.1 Lakemeyer's Proof Theory for a fragment of $\mathcal{OL}_n$

Recall Levesque's axiomatization from Section 3.1.5. It was clear that the most interesting axiom in his proof theory is **A5**, which discusses the relationship between the at most and the at least belief operator. From a technical perspective, however, it appeals to falsifiability in propositional logic. The idea is that the axiom is applicable on any consistent propositional formula. But in the many agent case, since we go beyond propositional formulas, *i.e.* since we have to establish the consistency of i-objective formulas, generalizing **A5** is non-trivial, and even circular. To this end, Lakemeyer proposes to resolve this consistency by relying on the existing logic $\mathbf{K45}_n$. As a consequence his proof theoretic formulation appropriately generalizes all of Levesque's axioms, except for **A5** where its application is restricted to basic i-objective consistent formulas only.

Axioms:

$\mathbf{A1}_n$. All instances of propositional logic

$\mathbf{A2}_n$. $\boldsymbol{K}_i(\alpha \supset \beta) \supset (\boldsymbol{K}_i\alpha \supset \boldsymbol{K}_i\beta)$,

$\mathbf{A3}_n$. $\boldsymbol{N}_i(\alpha \supset \beta) \supset (\boldsymbol{N}_i\alpha \supset \boldsymbol{N}_i\beta)$,

A4$_n$. $\sigma \supset \boldsymbol{K}_i\sigma \wedge \boldsymbol{N}_i\sigma$ for all i-subjective σ,

A5$_n$. $\boldsymbol{N}_i\alpha \supset \neg\boldsymbol{K}_i\alpha$ if $\neg\alpha$ is a **K45**$_n$-consistent i-objective basic formula.

Inference Rules:

MP. From α and $\alpha \supset \beta$ infer β.

NEC. From α infer $\boldsymbol{K}_i\alpha$ and $\boldsymbol{N}_i\alpha$.

We refer to the above set of schemas as **AX**$_n$. Lakemeyer proves that **AX**$_n$ is sound and complete for the canonical model approach when formulas are restricted to $\mathcal{OL}_n^- \subsetneq \mathcal{OL}_n$.

Definition 3.3.1. $\mathcal{OL}_n^-$ consists of all formulas α in $\mathcal{OL}_n$ such that no $\boldsymbol{N}_j$ may occur in the scope of a $\boldsymbol{K}_i$ or a $\boldsymbol{N}_i$, for $i \neq j$. ∎

Halpern shows that **AX**$_n$ is also sound and complete for formulas in $\mathcal{OL}_n^-$ wrt the semantics of the i-set approach. We now prove that this is the case wrt our semantics as well.

AX$_n$ **is sound and complete for** $\mathcal{OL}_n^-$

The soundness of **A5**$_n$ appeals to consistency wrt **K45**$_n$. But precisely because our semantics is not formulated using Kripke structures, stating that **K45**$_n$-consistent formulas are satisfiable is not immediate. Therefore we propose a construction known as a *correspondence model*. Intuitively, a correspondence model is a (k, j)-model obtained from a given Kripke structure. Since we will need to deal with worlds as considered by Levesque on one hand, and Kripke worlds on the other we refer to the former as *propositional valuations* in this section.

It is a well-known property in modal logic that every consistent formula is satisfiable at least in the canonical model [Chellas, 1980]. Therefore our idea will be to show that if a formula (of a certain depth) is satisfiable in the canonical model then it is also satisfiable in an appropriate correspondence model, and vice versa.

To review the construction of a canonical model, we first need the notion of a maximally consistent set of formulas.

Definition 3.3.2. (Consistency and Maximal Consistency.) Given an axiom system $\mathcal{X}$, we say that a formula α is consistent wrt $\mathcal{X}$ if it not the case that $\mathcal{X} \vdash \neg\alpha$. A finite set of formulas $\alpha_1, \ldots, \alpha_k$ is consistent wrt $\mathcal{X}$ if its conjunction is consistent wrt $\mathcal{X}$. An infinite set of formulas is consistent wrt $\mathcal{X}$ if every finite subset of formulas is consistent wrt $\mathcal{X}$.

Given a set of formulas S, a *maximally consistent* subset (wrt $\mathcal{X}$) of S is a subset S' which is consistent wrt $\mathcal{X}$, and any superset of S' is not consistent wrt $\mathcal{X}$. ∎

A **K45**$_n$ canonical model is a Kripke structure whose worlds are all the maximally **K45**$_n$-consistent sets of basic formulas. Formally,

Definition 3.3.3. (Canonical models.) The **K45**$_n$ canonical model $M^c = (\mathcal{W}^c, \pi^c, \mathcal{K}_A^c, \mathcal{K}_B^c)$ is defined as follows:

1. $\mathcal{W}^c = \{w \mid w \text{ is a maximally consistent set of basic formulas wrt } \mathbf{K45}_n\}$;
2. for all propositions p and worlds w, $\pi^c(w)(p) = \textsc{True}$ iff $p \in w$;
3. $(w, w') \in \mathcal{K}_i^c$ iff $w\backslash\boldsymbol{K}_i \subseteq w'$, where $w\backslash\boldsymbol{K}_i = \{\alpha \mid \boldsymbol{K}_i\alpha \in w\}$. ∎

Suppose that we have defined a canonical model M^c as in Definition 3.3.3. We obtain propositional valuations as follows:

Definition 3.3.4. Given M^c, define a set of propositional valuations $\mathcal{W}$ such that for each world $w \in \mathcal{W}^c$, there is a valuation $\|w\| \in \mathcal{W}$ where $\|w\| = \{p \text{ is a proposition} \mid p \in w\}$. ∎

That is, a valuation $\|w\|$ is the set of all propositions that are true at the Kripke world w.

Definition 3.3.5. (Correspondence (k, j)-model.) Given M^c and a world $w \in \mathcal{W}^c$, construct a (k, j)-model $(e_{\|w\|}{}_A^k, e_{\|w\|}{}_B^j, \|w\|)$ from valuations $\mathcal{W}$ inductively:

1. $e_{\|w\|}{}_A^1 = \{(\|w'\|, \{\}) \mid w' \in \mathcal{K}_A^c(w)\}$.
2. $e_{\|w\|}{}_B^1 = \{(\|w'\|, \{\}) \mid w' \in \mathcal{K}_B^c(w)\}$.
3. $e_{\|w\|}{}_A^k = \{(\|w'\|, e_{\|w'\|}{}_B^{k-1}) \mid w' \in \mathcal{K}_A^c(w)\}$ for $k > 1$.
4. $e_{\|w\|}{}_B^j = \{(\|w'\|, e_{\|w'\|}{}_A^{j-1}) \mid w' \in \mathcal{K}_B^c(w)\}$ for $j > 1$.

We refer to this model as the *correspondence* (k, j)*-model* of (M^c, w). ∎

That is, a correspondence model constructs epistemic states by appealing to the accessibility relations in the canonical Kripke structure. For instance, a 1-structure for A has precisely those propositional valuations corresponding to the worlds $\{w' \mid w' \in \mathcal{K}_A^c(w)\}$. Analogous, a k-structure is the set of all $(\|w'\|, e^{k-1})$, where $w' \in \mathcal{K}_A^c(w)$ as before and e^{k-1} is constructed inductively (to the appropriate depth) from all $w'' \in \mathcal{K}_B^c(w')$.

By an induction on the depth of a *basic* formula α, we obtain a theorem that α of maximal A, B-depth k, j is satisfiable at (M^c, w) iff the correspondence (k, j)-model satisfies the formula.

Theorem 3.3.6. *For all basic formulas α in $\mathcal{OL}_n^-$ and of maximal A, B-depth of k, j:*

$$M^c, w \models \alpha \text{ iff } e_{\|w\|}{}_A^k, e_{\|w\|}{}_B^j, \|w\| \models \alpha.$$

Proof: The proof is by induction on α. By the definition of propositional valuations the proof is trivial for atoms. The case of disjunctions and negations is also easy. So let us consider modalities, say $\boldsymbol{K}_A\alpha$. The case of $\boldsymbol{K}_B\alpha$ is analogous. Suppose $|\boldsymbol{K}_A\alpha|_A = k$, which, by definition, implies $|\alpha|_A = k$.

We have $M^c, w \models \boldsymbol{K}_A\alpha$

iff for all $w' \in \mathcal{K}_A^c(w)$, $M^c, w' \models \alpha$

iff for all $w' \in \mathcal{K}_A^c(w)$, $e_{\|w'\|}{}_A^k, e_{\|w'\|}{}_B^j, \|w'\| \models \alpha$ by induction, where $j \in \{k-1, k, k+1\}$ by Lemma 3.2.6

iff for all $w' \in \mathcal{K}_A^c(w)$, $e_{\|w'\|}{}_A^k, e_{\|w'\|}{}_B{\downarrow}_{k-1}^j, \|w'\| \models \alpha$ by Lemma 3.2.7 and Lemma 3.2.8

iff for all $w' \in \mathcal{K}_A^c(w)$, $e_{\|w'\|}^k{}_A, e_{\|w'\|}^{k-1}{}_B, \|w'\| \models \alpha$ since $e_{\|w'\|_B}{\downarrow}_{k-1}^{j}$ is $e_{\|w'\|}^{k-1}{}_B$

iff for all $w' \in \mathcal{K}_A^c(w)$, $e_{\|w\|}^k{}_A, e_{\|w'\|}^{k-1}{}_B, \|w'\| \models \alpha$ since $w^* \in \mathcal{K}_A^c(w)$ iff $w^* \in \mathcal{K}_A^c(w')$ by the transitive and Euclidean property in accessibility relations of $\mathbf{K45}_n$ structures, it follows that $e_{\|w\|}^k{}_A = e_{\|w'\|}^k{}_A$

iff for all $(\|w'\|, e_{\|w'\|}^{k-1}{}_B) \in e_{\|w\|}^k{}_A, e_{\|w\|}^k{}_A, e_{\|w'\|}^{k-1}{}_B, \|w'\| \models \alpha$ by construction

iff $e_{\|w\|}^k{}_A, \{\}, \|w\| \models \boldsymbol{K}_A\alpha$ by definition

iff $e_{\|w\|}^k{}_A, e_{\|w\|}^j{}_B, \|w\| \models \boldsymbol{K}_A\alpha$ since B's epistemic state is irrelevant. ∎

Corollary 3.3.7. *Every* $\mathbf{K45}_n$*-consistent basic formula* α *is satisfiable wrt some* (k, j)*-model.*

Proof: It is a property of the canonical model that every $\mathbf{K45}_n$-consistent basic formula is satisfiable wrt the canonical model [Chellas, 1980]. Supposing that the formula has a A, B-depth of k, j then from Theorem 3.3.6, there is a (k, j)-model that also satisfies the formula. ∎

With Corollary 3.3.7 in hand, establishing the soundness of $\mathbf{AX}_n$ is relatively straightforward.

Theorem 3.3.8. *For all* $\alpha \in \mathcal{OL}_n^-$*, if* $\mathbf{AX}_n \vdash \alpha$ *then* $\models \alpha$.

Proof: The soundness is easily shown to hold for $\mathbf{A1}_n - \mathbf{A4}_n$. To demonstrate the soundness of $\mathbf{A5}_n$, let $\neg\alpha$ be any $\mathbf{K45}_n$-consistent basic A-objective formula. Suppose that α has a maximal B-depth of k. By Corollary 3.3.7 there is $(w^*, e^{*k}{}_B)$, such that $\{\}, e^{*k}{}_B, w^* \models \neg\alpha$. Given an arbitrary e_A^{k+1}, if $(w^*, e^{*k}{}_B) \in e_A^{k+1}$ then $e_A^{k+1}, \{\}, w \models \neg\boldsymbol{K}_A\alpha$ for any w. Otherwise $e_A^{k+1}, \{\}, w \models \neg\boldsymbol{N}_A\alpha$. Therefore $e_A^{k+1}, \{\}, w \models \boldsymbol{N}_A\alpha \supset \neg\boldsymbol{K}_A\alpha$. ∎

To show that $\mathbf{AX}_n$ is also complete for formulas in $\mathcal{OL}_n^-$, it is sufficient to show that every $\mathbf{AX}_n$-consistent formula is satisfiable. To see the argument suppose that α is $\mathbf{AX}_n$-consistent but not satisfiable. This must mean that $\neg\alpha$ is valid. But by assumption $\mathbf{AX}_n \nvdash \neg\alpha$. Therefore $\mathbf{AX}_n$ is not complete wrt the given semantics. Thus, it suffices to show that every $\mathbf{AX}_n$-consistent formula is satisfiable.

Halpern and Lakemeyer [2001] show that every formula $\alpha \in \mathcal{OL}_n$ can be reduced to a certain normal form, which we call ONF. Our idea, then, will be to reduce every $\mathbf{AX}_n$-consistent formula to one in ONF and use its simple structure to prove that it is indeed satisfiable.[10]

Lemma 3.3.9. (ONF.) *[Halpern and Lakemeyer, 2001]*
Every $\alpha \in \mathcal{OL}_n$ *is provably equivalent to disjunctions of formulas of the form:*

$$\sigma \wedge \boldsymbol{K}_A\phi_{A0} \wedge \bigwedge \neg\boldsymbol{K}_A\phi_{Az_A} \wedge \boldsymbol{K}_B\phi_{B0} \wedge \bigwedge \neg\boldsymbol{K}_B\phi_{Bz_B} \wedge$$

$$\boldsymbol{N}_A\psi_{A0} \wedge \bigwedge \neg\boldsymbol{N}_A\psi_{Az'_A} \wedge \boldsymbol{N}_B\psi_{B0} \wedge \bigwedge \neg\boldsymbol{N}_B\psi_{Bz'_B}$$

where σ *is a propositional formula,* ϕ_{iz_h} *and* ψ_{iz_h} *are* i*-objective. If* $\alpha \in \mathcal{OL}_n^-$ *then* ϕ_{iz_h} *and* ψ_{iz_h} *are basic.*

[10] Lakemeyer uses a similar technique to demonstrate that $\mathbf{AX}_n$ is sound and complete for formulas in $\mathcal{OL}_n^-$ wrt his approach. See [Halpern and Lakemeyer, 2001] for a proof.

The normal form is significant because it allows us to simplify i-subjective formulas to ones of the form $\boldsymbol{K}_i\alpha$, where α is strictly i-objective. The proof involves equivalences based on $\mathbf{K45}_n$ and $\mathbf{AX}_n$ such as $\boldsymbol{K}_A(\alpha \vee \neg\boldsymbol{K}_A\beta) \equiv \boldsymbol{K}_A\alpha \vee \neg\boldsymbol{K}_A\beta$. Before proceeding to the completeness result, the following notions from [Halpern and Lakemeyer, 2001] will prove useful.

Definition 3.3.10. A formula ψ is said to be *independent* of the formula ϕ wrt an axiom system $\mathcal{X}$, if neither $\mathcal{X} \vdash \phi \supset \psi$ nor $\mathcal{X} \vdash \phi \supset \neg\psi$. ∎

Lemma 3.3.11. *[Halpern and Lakemeyer, 2001]*
If $\phi_1, \ldots, \phi_l$ are $\mathbf{K45}_n$-consistent basic i-objective formulas then there exists a basic i-objective formula ψ of the form $\boldsymbol{K}_j\psi'$ for $j \neq i$ that is independent of $\phi_1, \ldots, \phi_l$ wrt $\mathbf{K45}_n$.

Proof Sketch: Let i be B, and the other case is symmetric. Suppose ϕ_z, $z \geq 1$, are B-objective formulas of maximal A-depth k. Construct a formula ψ of the form $(\boldsymbol{K}_A\boldsymbol{K}_B)^{k+1}p$ for any proposition p, that is, p is in the scope of $k+1$ sequences of $\boldsymbol{K}_A\boldsymbol{K}_B$. It is easy to show that the A-depth of ψ is $2k+2$. The formula ψ can be shown to be independent of ϕ_z via model-theoretic arguments [Halpern and Lakemeyer, 2001]. Briefly, the idea is to show that a Kripke structure can be constructed that satisfies ϕ_z but falsifies ψ and another that satisfies both ϕ_z and ψ. Thus ψ is independent of ϕ_z. ∎

Lemma 3.3.12. *[Halpern and Lakemeyer, 2001]*
If ϕ and ψ are i-objective basic formulas, and if $\boldsymbol{K}_i\phi \wedge \boldsymbol{N}_i\psi$ is $\mathbf{AX}_n$-consistent, then $\models \phi \vee \psi$.

We are now ready to prove the main result for $\mathcal{OL}_n^-$.

Theorem 3.3.13. *For all formulas $\alpha \in \mathcal{OL}_n^-$, if $\models \alpha$ then $\mathbf{AX}_n \vdash \alpha$.*

Proof: It is sufficient to prove that every $\mathbf{AX}_n$-consistent formula α is satisfiable. Let us reduce α to one in ONF, as in Lemma 3.3.9, that is, to a disjunction of formulas of the form:

$$\sigma \wedge \boldsymbol{K}_A\phi_{A0} \wedge \bigwedge \neg\boldsymbol{K}_A\phi_{Az_A} \wedge \boldsymbol{K}_B\phi_{B0} \wedge \bigwedge \neg\boldsymbol{K}_B\phi_{Bz_B} \wedge$$

$$\boldsymbol{N}_A\psi_{A0} \wedge \bigwedge \neg\boldsymbol{N}_A\psi_{Az'_A} \wedge \boldsymbol{N}_B\psi_{B0} \wedge \bigwedge \neg\boldsymbol{N}_B\psi_{Bz'_B}$$

where σ is a propositional formula, and ϕ_{iz}, ψ_{iz} are *basic* i-objective formulas (since $\alpha \in \mathcal{OL}_n^-$). The idea will be to show that α in the normal form is satisfiable in some (k', j')-model.

Suppose $\{\phi_{A0}, \psi_{A0}, \phi_{Az}, \psi_{Az}\}$, $z \geq 1$ are of maximal B-depth k. Let Γ_A be the set of all consistent formulas of the form $\phi_{A0} \wedge \psi_{A0} \wedge \neg\phi_{Az}$ or $\phi_{A0} \wedge \psi_{A0} \wedge \neg\psi_{Az}$, $z \geq 1$. Let γ be a formula independent of all formulas in Γ_A, which exists by way of Lemma 3.3.11. Since the formulas in Γ_A are of maximal B-depth k, then note that the independent formula constructed in Lemma 3.3.11 is of B-depth $2k+2$. By only considering A-objective basic formulas of maximal B-depth k, let Σ_A be the set of all consistent sets of formulas containing $\phi_{A0} \wedge (\neg\psi_{A0} \vee (\psi_{A0} \wedge \gamma))$.

Since each $\Sigma' \in \Sigma_A$ is basic and A-objective, they are satisfiable by Corollary 3.3.7. Let

$$e_A^{2k+3} = \{(w, e_B^{2k+2}) \mid \{\}, e_B^{2k+2}, w \models \Sigma' \text{ and } \Sigma' \in \Sigma_A\}.$$

We now show that the A-subjective formulas in the normal form are satisfiable wrt e_A^{2k+3}. In an analogous fashion, an epistemic state for B can be constructed that satisfies the B-subjective formulas in α. Finally, since σ is a propositionally consistent formula, there is a world that satisfies σ. Clearly the resulting model satisfies α.

case $\boldsymbol{K}_A\phi_{A0}$: For all $\Sigma' \in \Sigma_A$, we have $\phi_{A0} \in \Sigma'$ by assumption. Therefore $e_A^{2k+3}, \{\}, w \models \boldsymbol{K}_A\phi_{A0}$ for any w.

case $\neg\boldsymbol{K}_A\phi_{Az}$, $z \geq 1$: Since $\boldsymbol{K}_A\phi_{A0} \wedge \neg\boldsymbol{K}_A\phi_{Az}$ is consistent it follows that $\phi_{A0} \wedge \neg\phi_{Az}$ is consistent. For suppose not. Then $\neg\phi_{A0} \vee \phi_{Az}$ is provable. Therefore $\boldsymbol{K}_A\phi_{A0} \supset \boldsymbol{K}_A\phi_{Az}$ is provable, which contradicts the consistency of $\boldsymbol{K}_A\phi_{A0} \wedge \neg\boldsymbol{K}_A\phi_{Az}$.

Given that $\phi_{A0} \wedge \neg\phi_{Az}$ is consistent, it follows that either $\phi_{A0} \wedge \neg\phi_{Az} \wedge \psi_{A0}$ or $\phi_{A0} \wedge \neg\phi_{Az} \wedge \neg\psi_{A0}$ is consistent. When the former, by the choice of γ, we also have that $\phi_{A0} \wedge \neg\phi_{Az} \wedge \psi_{A0} \wedge \gamma$ is consistent. Since Σ_A consists of all consistent sets containing $\phi_{A0} \wedge (\neg\psi_{A0} \vee (\psi_{A0} \wedge \gamma))$, it follows that there is a $\Sigma' \in \Sigma_A$ containing $\neg\phi_{Az}$. Therefore $e_A^{2k+3}, \{\}, w \models \neg\boldsymbol{K}_A\phi_{Az}$, $z \geq 1$.

case $\boldsymbol{N}_A\psi_{A0}$: Consider any arbitrary $(w', e_B^{2k+2}) \notin e_A^{2k+3}$. One of the following must hold wrt the structure: (1) $\phi_{A0} \wedge \psi_{A0}$; (2) $\neg\phi_{A0} \wedge \psi_{A0}$; (3) $\phi_{A0} \wedge \neg\psi_{A0}$; (4) $\neg\phi_{A0} \wedge \neg\psi_{A0}$.

It cannot be (4), because $\boldsymbol{K}_A\phi_{A0} \wedge \boldsymbol{N}_A\psi_{A0}$ is consistent implying that $\models \phi_{A0} \vee \psi_{A0}$ by way of Lemma 3.3.12. It cannot be (3), for it would be in some $\Sigma' \in \Sigma_A$. This leaves us with (1) and (2), both of which contain ψ_{A0}. Therefore $e_A^{2k+3}, \{\}, w \models \boldsymbol{N}_A\psi_{A0}$.

case $\neg\boldsymbol{N}_A\psi_{Az}$, $z \geq 1$: Given that $\boldsymbol{N}_A\psi_{A0} \wedge \neg\boldsymbol{N}_A\psi_{Az}$ is consistent. This implies that $\psi_{A0} \wedge \neg\psi_{Az}$ is consistent, by the same argument made for the case $\neg\boldsymbol{K}_A\phi_{Az}$. In addition, either $\psi_{A0} \wedge \neg\psi_{Az} \wedge \phi_{A0}$ or $\psi_{A0} \wedge \neg\psi_{Az} \wedge \neg\phi_{A0}$ must be consistent. If the former, then by the choice of γ, $\psi_{A0} \wedge \neg\psi_{Az} \wedge \phi_{A0} \wedge \neg\gamma$ is consistent. Let ζ be $\psi_{A0} \wedge \neg\psi_{Az} \wedge \neg\phi_{A0}$ if it is consistent. Otherwise let ζ be $\psi_{A0} \wedge \neg\psi_{Az} \wedge \phi_{A0} \wedge \neg\gamma$. Clearly $e_A^{2k+3}, \{\}, w \models \boldsymbol{K}_A\neg\zeta$. But ζ is consistent by construction and basic by assumption, and therefore it is satisfiable. This implies that there is some $(w', e_B^{2k+2}) \notin e_A^{2k+3}$, such that $\{\}, e_B^{2k+2}, w' \models \zeta$. Since ζ contains $\neg\psi_{Az}$ it follows that $\{\}, e_B^{2k+2}, w' \models \neg\psi_{Az}$. Therefore $e_A^{2k+3}, \{\}, w \models \neg\boldsymbol{N}_A\psi_{Az}$, for $z \geq 1$.

This completes the proof for A-subjective formulas. ∎

$\mathbf{AX}_n$ is not complete for $\mathcal{OL}_n$

The soundness of $\mathbf{AX}_n$ in Theorem 3.3.8 is easily lifted for all formulas $\alpha \in \mathcal{OL}_n$. The only interesting case is with regards to the $\mathbf{A5}_n$ schema. But because the application of the schema is restricted to basic formulas, the argument given holds immediately also for $\mathcal{OL}_n$. To show that $\mathbf{AX}_n$ is not a complete axiomatization, we simply need to a formula in $\mathcal{OL}_n$ such that it is valid but not provable using $\mathbf{AX}_n$.

Proposition 3.3.14. $\models \boldsymbol{K}_i(\text{FALSE}) \supset \neg\boldsymbol{N}_i\neg\boldsymbol{O}_j\neg\boldsymbol{O}_i p$.

Proof: Let i be A, with the other case being symmetric. Suppose $e_A^k, \{\}, w \models \boldsymbol{K}_A(\text{FALSE})$ for any $w \in \mathcal{W}$. Then for all $(w', e_B^{k-1}) \in e_A^k$, $e_A^k, e_B^{k-1}, w' \models \text{FALSE}$, and this implies that e_A^k is empty. Assume now, contrary to the proposition, $e_A^k, \{\}, w \models \boldsymbol{N}_A\neg\boldsymbol{O}_B\neg\boldsymbol{O}_A p$. Then wrt all of $(w', e_B^{k-1}) \notin e_A^k$, *i.e.* all of $\mathbb{B}^k$, $\neg\boldsymbol{O}_B\neg\boldsymbol{O}_A p$ is satisfied. Equivalently $\neg\boldsymbol{O}_B\neg\boldsymbol{O}_A p$ should be valid, contradicting Proposition 3.2.16. ∎

However, it is not possible to prove this formula using $\mathbf{AX}_n$ [Halpern and Lakemeyer, 2001, see Theorem 4.7]. Therefore $\mathbf{AX}_n$ is not complete for formulas in $\mathcal{OL}_n$. The validity of the non-provable formula $\neg \boldsymbol{O}_i \neg \boldsymbol{O}_j p \in \mathcal{OL}_n$ wrt the canonical model approach demonstrates in a similar spirit that $\mathbf{AX}_n$ is also not a complete axiomatization in that approach.

Part of the problem is $\mathbf{A5}_n$. It has to somehow go beyond basic formulas. But this is a problem of circularity. On the one hand, we would $\boldsymbol{N}_i\alpha \supset \neg \boldsymbol{K}_i\alpha$ to hold for any consistent i-objective α. On the other, to deal with consistency we have to clarify and define an axiom system.

The approach taken by Halpern and Lakemeyer [2001] is introduce the semantic notion of validity, and the dual notion of satisfiability, into the language as modal operators. The motivation is that is by syntactically representing satisfiability, the notion of consistency can be inductively defined from propositional formulas to modal ones. Not surprisingly, a new set of axioms are needed to characterize this feature, by way of which the axiomatization is significantly different from Levesque's formulation for the single agent case. We give the details in the next section but for now let us refer to their axiomatic proposal as $\mathbf{AX}'_n$.

Halpern and Lakemeyer show that the axioms $\mathbf{AX}'_n$ characterize an infinite model, very much in the spirit of the canonical model from Definition 3.3.3. There are differences to Definition 3.3.3, however. First, the worlds are defined as maximally consistent $\mathbf{AX}'_n$-consistent formulas. Second, $\boldsymbol{K}_i$ and $\boldsymbol{N}_i$ are treated as separate modal operators, which results in two separate accessibility relations. Owing to this difference, the model is referred to as the *extended canonical model*.

One approach towards an axiomatization for our semantics is to perhaps show that valid formulas wrt our approach coincides with the extended canonical model. But axiomatizing validity is not natural. One of the principal reasons for axiomatic formulations is to have an insightful view on valid formulas in a logic, independent of semantic notions. Further, the proof theory is difficult to use. Lastly, we would still understand the axioms to characterize a semantics bridged on proof-theoretic elements.

3.3.2 A Proof Theory for $\mathcal{OL}_n$

What is desired is a generalization of Levesque's axiom **A5**, and nothing more. To this end, we propose a new axiom system that is subtly related to the structure of formulas. Formally, we define a sequence of languages:

Definition 3.3.15. Let $\mathcal{OL}_n^1 = \mathcal{OL}_n^-$. Let $\mathcal{OL}_n^t$ be all Boolean combinations of formulas of $\mathcal{OL}_n^{t-1}$ and formulas of the form $\boldsymbol{K}_i\alpha$ and $\boldsymbol{N}_i\alpha$ for $\alpha \in \mathcal{OL}_n^{t-1}$. ∎

Clearly $\mathcal{OL}_n^t \supsetneq \mathcal{OL}_n^{t-1}$. Intuitively, each language adds another level of nesting of only knowing with varying agent indices.

We remark that when $t = 1$ we have already established that $\mathbf{AX}_n$ characterize formulas in $\mathcal{OL}_n^1$. The axiom system that characterizes $\mathcal{OL}_n^t$ is defined as:

Axioms:

$\mathbf{A1}_n - \mathbf{A4}_n$ from $\mathbf{AX}_n$,

$\mathbf{A5}_n^1$. $\boldsymbol{N}_i\alpha \supset \neg \boldsymbol{K}_i\alpha$ if $\neg\alpha$ is a $\mathbf{K45}_n$-consistent i-objective basic formula.

$\mathbf{A5}^t_n$. $\boldsymbol{N}_i\alpha \supset \neg \boldsymbol{K}_i\alpha$ if $\neg\alpha \in \mathcal{OL}^{t-1}_n$ is i-objective,
and consistent wrt $\mathbf{A1}_n - \mathbf{A4}_n$, $\mathbf{A5}^1_n - \mathbf{A5}^{t-1}_n$.

Inference Rules:

MP and **NEC**.

We denote this set of schemas as $\mathbf{AX}^t_n$. That is, for a given t, there are t axioms in addition to $\mathbf{A1}_n - \mathbf{A4}_n$. Intuitively, we address the circularity of the consistency issue inductively:[11]

- at the base level we appeal to consistency wrt $\mathbf{K45}_n$,
- at the next level we appeal to consistency wrt $\mathbf{AX}_n$,
- at the t level, we appeal to consistency wrt $\mathbf{AX}^{t-1}_n$.

Let us illustrate this idea with an example:

Example 3.3.16. Suppose p is a proposition. Then,

1. $\mathbf{A5}^t_n \vdash \boldsymbol{N}_A\boldsymbol{K}_Bp \supset \neg\boldsymbol{K}_A\boldsymbol{K}_Bp$, for $t \geq 1$.
2. $\mathbf{A5}^t_n \vdash \boldsymbol{N}_A\boldsymbol{O}_Bp \supset \neg\boldsymbol{K}_A\boldsymbol{O}_Bp$, for $t \geq 2$.

Consider that $\neg\boldsymbol{K}_Bp \in \mathcal{OL}^-_n$ is a A-objective basic formula that is consistent wrt $\mathbf{K45}_n$, which means that we may apply $\mathbf{A5}^1_n$ to prove item 1. Naturally, then, for this example Lakemeyer's proof theory is sufficient.

To see where we may need the full power of $\mathbf{AX}^t_n$ consider item 2. Let us denote $\boldsymbol{N}_A\boldsymbol{O}_Bp \supset \neg\boldsymbol{K}_A\boldsymbol{O}_Bp$ with γ. The formula $\boldsymbol{O}_Bp$ is not basic. As a result $\mathbf{A5}^1_n$ is not applicable. However, $\boldsymbol{O}_Bp$ is A-objective and $\boldsymbol{K}_Bp \wedge \boldsymbol{N}_B\neg p \in \mathcal{OL}^-_n$. Clearly $\neg\boldsymbol{O}_Bp$ is $\mathbf{AX}_n$-consistent since it is not the case that $\mathbf{AX}_n \vdash \neg\neg\boldsymbol{O}_Bp$. It follows from this that γ is provable from $\mathbf{A5}^2_n$. Further, if $\neg\boldsymbol{O}_Bp$ is $\mathbf{AX}^1_n$-consistent then clearly $\neg\boldsymbol{O}_Bp$ is $\mathbf{AX}^t_n$-consistent for $t \geq 1$. So for any $t \geq 2$, $\mathbf{A5}^t_n$ allows us to prove γ. Thus, the use of $\mathbf{A5}^t_n$ is straightforward and requires us to inspect the belief operators occurring in the scope of the outermost $\boldsymbol{N}_i$. ∎

We now turn to a soundness result:

Theorem 3.3.17. *For all $\alpha \in \mathcal{OL}^t_n$, if $\mathbf{AX}^t_n \vdash \alpha$ then $\models \alpha$.*

Proof: The soundness of $\mathbf{A1}_n - \mathbf{A4}_n$ is straightforward. The proof for $\mathbf{A5}^t_n$ is by an induction on t. Theorem 3.3.8 proves the case for $t = 1$. Therefore assume that for all $\alpha \in \mathcal{OL}^{t-1}_n$, if $\mathbf{AX}^{t-1}_n \vdash \alpha$ then $\models \alpha$.

Let $\neg\alpha \in \mathcal{OL}^{t-1}_n$ be a A-objective formula that is consistent wrt $\mathbf{AX}^{t-1}_n$. Suppose that α has a maximal B-depth of k. By induction, there is some (w^*, e^{*k}_B) such that $\{\}, e^{*k}_B, w^* \models \neg\alpha$. Now if $e^{k+1}_A, \{\}, w \models \boldsymbol{N}_A\alpha$ then for every $(w', e^k_B) \notin e^{k+1}_A$, $\{\}, e^k_B, w' \models \alpha$. Of course this means $(w^*, e^{*k}_B) \in e^{k+1}_A$. Therefore $e^{k+1}_A, \{\}, w \models \neg\boldsymbol{K}_A\alpha$ implying $e^{k+1}_A, \{\}, w \models \boldsymbol{N}_A\alpha \supset \neg\boldsymbol{K}_A\alpha$. ∎

[11] The idea was also suggested by a reviewer in [Halpern and Lakemeyer, 2001] for an axiomatic characterization of the extended canonical model, although its completeness was left open.

To illustrate that $\mathbf{AX}_n^t$ is also complete for formulas in $\mathcal{OL}_n^t$, the proof uses ideas similar to that considered in Theorem 3.3.13. Therefore it is necessary to ensure that Lemma 3.3.11 and Lemma 3.3.12 also hold for non-basic formulas.

Lemma 3.3.18. *Let $\phi_1, \ldots, \phi_l$ be i-objective formulas that are consistent wrt $\mathbf{AX}_n^t$. Then there is a basic formula ψ of the form $\boldsymbol{K}_j\psi'$ for $j \neq i$ that is independent of $\phi_1, \ldots, \phi_l$ wrt $\mathbf{AX}_n^t$.*

Proof: Let i be B, with the other case being symmetric. Suppose that ϕ_z, $z \geq 1$, are B-objective and of maximal A-depth k. We will show that a formula ψ of the form $(\boldsymbol{K}_A\boldsymbol{K}_B)^{k+1}p$, where p is any proposition and p is in the scope of $k+1$ occurrences of $\boldsymbol{K}_A\boldsymbol{K}_B$, is independent of $\phi_1, \ldots, \phi_l$.

From Theorem 3.3.17, the axiom system is sound. So if $\mathbf{AX}_n^t \vdash \phi_z \supset \psi$ then $\models \phi_z \supset \psi$. Likewise if $\mathbf{AX}_n^t \vdash \phi_z \supset \neg\psi$ then $\models \phi_z \supset \neg\psi$. Therefore to prove the independence result, we only need to show there is a model that satisfies ϕ_z and ψ, and one that satisfies ϕ_z but falsifies ψ.

Let us begin with the observation that ϕ_z has a maximal A-depth of k whereas ψ has a A-depth of $2k+2$. Given a e_A^k (analogously for a e_B^j), it is possible to construct a k'-structure, say $e_A{\uparrow_k^{k'}}$, for $k' > k \geq 1$ such that it satisfies a formula α of the form $\boldsymbol{K}_A\boldsymbol{K}_B\boldsymbol{K}_A\boldsymbol{K}_B \ldots p$ of A-depth k':

- $e_A{\uparrow_1^{k'}} = \{(w, e_B^{*k'-1}) \mid w \in e_A^1\}$, where $e_B^{*k'-1}$ is an epistemic state that satisfies $\boldsymbol{K}_B\boldsymbol{K}_A \ldots p$ of B-depth $k'-1$;
- $e_A{\uparrow_k^{k'}} = \{(w, e_B{\uparrow_{k-1}^{k'-1}}) \mid (w, e_B^{k-1}) \in e_A^k\}$.

Now, since ϕ_z is consistent, B-objective and is of maximal A-depth k, it is satisfiable by the soundness property, *viz.* Theorem 3.3.17. Let e_A^k be a k-structure that satisfies ϕ_z. Next, construct $e_A{\uparrow_k^{2k+2}}$. By construction, $e_A{\uparrow_k^{2k+2}}$ satisfies ψ. By Lemma 3.2.9, $e_A{\uparrow_k^{2k+2}}$ satisfies ϕ_z iff $(e_A{\uparrow_k^{2k+2}}){\downarrow_k^{2k+2}}$ satisfies ϕ_z, *i.e.* iff e_A^k satisfies ϕ_z, which it does by assumption.

Analogously, given a e_A^k, we can construct a k'-structure, $k' > k \geq 1$, that satisfies $\neg\boldsymbol{K}_A\boldsymbol{K}_B \ldots p$ of A-depth k'. By analogous arguments, then, it is possible to prove that there is e_A^{2k+2} that satisfies ϕ_z and $\neg(\boldsymbol{K}_A\boldsymbol{K}_B)^{k+1}p$, *i.e.* $\neg\psi$. ∎

Lemma 3.3.19. *Suppose $\phi \in \mathcal{OL}_n^{t-1}$ and $\psi \in \mathcal{OL}_n^{t-1}$ are i-objective formulas. If $\boldsymbol{K}_i\phi \wedge \boldsymbol{N}_i\psi$ is consistent wrt $\mathbf{AX}_n^t$ then $\models \phi \vee \psi$.*

Proof: Assume to the contrary. Then $\neg\phi \wedge \neg\psi$ is consistent wrt $\mathbf{AX}_n^{t-1}$. Let i be A. By $\mathbf{A5}_n^t$ we prove $\boldsymbol{N}_A(\phi \vee \psi) \supset \neg\boldsymbol{K}_A(\phi \vee \psi)$. Therefore $\boldsymbol{N}_A\psi \supset \neg\boldsymbol{K}_A\phi$ is provable, contradicting the consistency of $\boldsymbol{K}_A\phi \wedge \boldsymbol{N}_A\psi$. ∎

Theorem 3.3.20. *For all $\alpha \in \mathcal{OL}_n^t$, if $\models \alpha$ then $\mathbf{AX}_n^t \vdash \alpha$.*

Proof: Proof by an induction on t. It is sufficient to prove that every $\mathbf{AX}_n^t$-consistent formula α is satisfiable. The base case, *i.e.* when t is 1, is already established in Theorem 3.3.13. For the induction hypothesis, assume that every $\mathbf{AX}_n^{t-1}$-consistent formula $\alpha \in \mathcal{OL}_n^{t-1}$ is satisfiable.

Suppose $\alpha \in \mathcal{OL}_n^t$ is $\mathbf{AX}_n^t$-consistent. Without any loss of generality, let us suppose that it is in ONF, as in Lemma 3.3.9.[12] That is, α is equivalent to a disjunction of formulas of the form

[12] Recall that a reduction of a formula in $\mathcal{OL}_n$ to ONF only uses the axioms from $\mathbf{K45}_n$ and $\mathbf{AX}_n$ [Halpern and Lakemeyer, 2001].

$$\sigma \wedge \boldsymbol{K}_A\phi_{A0} \wedge \bigwedge \neg\boldsymbol{K}_A\phi_{Az_A} \wedge \boldsymbol{K}_B\phi_{B0} \wedge \bigwedge \neg\boldsymbol{K}_B\phi_{Bz_B} \wedge$$

$$\boldsymbol{N}_A\psi_{A0} \wedge \bigwedge \neg\boldsymbol{N}_A\psi_{Az'_A} \wedge \boldsymbol{N}_B\psi_{B0} \wedge \bigwedge \neg\boldsymbol{N}_B\psi_{Bz'_B}$$

where σ is a propositional formula and ϕ_{iz}, ψ_{iz} for $z \geq 1$ are i-objective (not necessarily basic) formulas. Further, by the definition of $\mathcal{OL}_n^t$, $\phi_{iz}, \psi_{iz} \in \mathcal{OL}_n^{t-1}$.

The remainder of the proof uses formal ideas similar to the ones used in Theorem 3.3.13, the only exception being that we do not restrict our attention to basic formulas. Suppose now $\{\phi_{A0}, \psi_{A0}, \phi_{Az}, \psi_{Az}\}$ are of maximal B-depth k. Then, let Γ_A be the set of all $\mathbf{AX}_n^{t-1}$-consistent formulas of the form $\phi_{A0} \wedge \psi_{A0} \wedge \neg\phi_{Az}$ or $\phi_{A0} \wedge \psi_{A0} \wedge \neg\psi_{Az}$, $z \geq 1$. Let γ be a formula that is independent of all formulas in Γ_A, which exists by way of Lemma 3.3.18. As constructed in that lemma, there is a formula γ whose maximal B-depth is $2k + 2$.

Now, only using A-objective formulas of maximal B-depth k, let Σ_A be the set of all $\mathbf{AX}_n^{t-1}$-consistent sets of formulas containing $\phi_{A0} \wedge (\neg\psi_{A0} \vee (\psi_{A0} \wedge \gamma))$. By the induction hypothesis, each $\Sigma' \in \Sigma_A$ is satisfiable. Then let

$$e_A^{2k+3} = \{(w, e_B^{2k+2}) \mid \{\}, e_B^{2k+2}, w \models \Sigma' \text{ and } \Sigma' \in \Sigma_A\}.$$

We claim that all the A-subjective formulas in α (in ONF) are satisfied wrt e_A^{2k+3}. An analogous construction of an epistemic state for B satisfies the B-subjective formulas in α. Finally, a world w^* satisfies the consistent propositional formula σ by definition. Thus a model for α is found. We only prove the case of $\neg\boldsymbol{K}_A\phi_{Az}$ below, since the argument for the other cases is pursued in the same fashion as done in Theorem 3.3.13.

case $\neg\boldsymbol{K}_A\phi_{Az}, z \geq 1$: Since $\boldsymbol{K}_A\phi_{A0} \wedge \neg\boldsymbol{K}_A\phi_{Az}$ is consistent wrt $\mathbf{AX}_n^t$, it follows that $\phi_{A0} \wedge \neg\phi_{Az}$ is consistent wrt $\mathbf{AX}_n^t$. Further, since $\phi_{A0}, \phi_{Az} \in \mathcal{OL}_n^{t-1}$, the formula must be consistent wrt $\mathbf{AX}_n^{t-1}$. For if not, they cannot by definition be consistent wrt $\mathbf{AX}_n^t$. This means that either $\phi_{A0} \wedge \neg\phi_{Az} \wedge \psi_{A0}$ or $\phi_{A0} \wedge \neg\phi_{Az} \wedge \neg\psi_{A0}$ is consistent. If the former is consistent, then so is $\phi_{A0} \wedge \neg\phi_{Az} \wedge \psi_{A0} \wedge \gamma$. Since Σ_A consists of all $\mathbf{AX}_n^{t-1}$-consistent sets of formulas containing $\phi_{A0} \wedge (\neg\psi_{A0} \vee (\psi_{A0} \wedge \gamma))$, there is clearly a $\Sigma' \in \Sigma_A$ such that $\neg\phi_{Az} \in \Sigma'$. Therefore $e_A^{2k+3}, \{\}, w \models \neg\boldsymbol{K}_A\phi_{Az}$. ∎

Thus, we have a sound and complete axiomatization for propositional $\mathcal{OL}_n$. In comparison to Lakemeyer's proof theory $\mathbf{AX}_n$, the current axiomatization goes beyond a language that restricts the nesting of $\boldsymbol{N}_i$. In contrast to Halpern and Lakemeyer [2001], the axiomatization does not necessitate the use of semantic notions in the proof theory. In the next section, we consider examples of formal derivations with our proof theory.

As a closing remark, let us draw comparisons to one other attempt to capture multiagent only knowing. An axiomatization by Waaler [2004] considers an interesting alternative to deal with the circularity in a generalized **A5**. The idea is to first define consistency by formulating a fragment of the axiom system in the sequent calculus. Quite analogous to having t-axioms, they allow us to apply the $\boldsymbol{N}_i$ *vs.* $\boldsymbol{K}_i$ relationship on i-objective formulas of a lower depth, thereby avoiding circularity without the need to appeal to satisfiability as in [Halpern and Lakemeyer, 2001]. Waaler and Solhaug [2005] also define a semantics for multiagent only knowing which does not appeal to canonical models. Instead, they define a class of Kripke structures which need to satisfy certain constraints. Unfortunately, these constraints are quite involved and, as the authors admit, the nature of these models "is complex and hard to penetrate".

3.3.3 Formal Derivations of Multiagent Reasoning

In this section, we provide examples of how the proof theory can be used for reasoning about multiagent beliefs. Let q denote that D is a big block and p denote that D is located in the storage. We consider variant examples involving Alice's beliefs about Bob's knowledge regarding p and q.

In what follows, we give justifications when outlining the proofs on the *rhs* often referring to a previous line or axioms that are used to obtain the current line. We write **PL** to mean propositional reasoning. We write **Def** to refer to the equivalence $\boldsymbol{O}_i\alpha \equiv \boldsymbol{K}_i\alpha \wedge \boldsymbol{N}_i\neg\alpha$. We also freely reason with $\mathbf{K45}_n$.

Example 3.3.21. While Alice knows both p and q, she assumes, however, that all that Bob knows is that D is a big block. She, then, must clearly believe that Bob does not know where D is located. Formally, we show:

$$\vdash \boldsymbol{K}_A(p \wedge q \wedge \boldsymbol{O}_B q) \supset \boldsymbol{K}_A\neg\boldsymbol{K}_B p.$$

Let us begin by observing that by using Levesque's proof theory for $\mathcal{OL}$, it is not hard to show

$$\vdash \boldsymbol{O}q \supset \neg\boldsymbol{K}p.$$

We then prove our claim as follows:

1. $(p \wedge q \wedge \boldsymbol{O}_B q) \supset \neg\boldsymbol{K}_B p$ — see above
2. $\boldsymbol{K}_A((p \wedge q \wedge \boldsymbol{O}_B q) \supset \neg\boldsymbol{K}_B p)$ — 1, **NEC**
3. $\boldsymbol{K}_A(p \wedge q \wedge \boldsymbol{O}_B q) \supset \boldsymbol{K}_A\neg\boldsymbol{K}_B p$ — 2, $\mathbf{A2}_n$.

Thus, this example shows that Alice is able to reason about Bob's non-beliefs when she makes assumptions about all that he knows. ∎

Example 3.3.22. We can also capture Alice's assumptions about Bob's ignorance regarding D's location by means of the following remark:

> *Unless I know that Bob knows p assume that he does not know it.*

We can prove that if this assumption is all that Alice knows, then she believes that Bob does not know D's location. Formally, we can show:

$$\vdash \boldsymbol{O}_A(\neg\boldsymbol{K}_A\boldsymbol{K}_B p \supset \neg\boldsymbol{K}_B p) \supset \boldsymbol{K}_A\neg\boldsymbol{K}_B p.$$

Let α denote the default $\neg\boldsymbol{K}_A\boldsymbol{K}_B p \supset \neg\boldsymbol{K}_B p$.

1. $\boldsymbol{O}_A\alpha \supset (\boldsymbol{K}_A\neg\boldsymbol{K}_A\boldsymbol{K}_B p \supset \boldsymbol{K}_A\neg\boldsymbol{K}_B p)$ — **Def**, **PL**, $\mathbf{A2}_n$
2. $\boldsymbol{O}_A\alpha \supset (\boldsymbol{N}_A\neg\boldsymbol{K}_A\boldsymbol{K}_B p \wedge \boldsymbol{N}_A\boldsymbol{K}_B p)$ — **Def**, **PL**, $\mathbf{K45}_n$
3. $\boldsymbol{N}_A\boldsymbol{K}_B p \supset \neg\boldsymbol{K}_A\boldsymbol{K}_B p$ — $\mathbf{A5}_n^1$
4. $\neg\boldsymbol{K}_A\boldsymbol{K}_B p \supset \boldsymbol{K}_A\neg\boldsymbol{K}_A\boldsymbol{K}_B p$ — $\mathbf{A4}_n$

5. $\boldsymbol{O}_A\alpha \supset \boldsymbol{K}_A\neg\boldsymbol{K}_A\boldsymbol{K}_B p$ 2, 3, 4, **PL**

6. $\boldsymbol{O}_A\alpha \supset \boldsymbol{K}_A\neg\boldsymbol{K}_B p$ 1, 5, **PL**

Most of the steps involve standard propositional or $\mathbf{K45}_n$ reasoning. In line 3, we invoke the relationship between $\boldsymbol{N}_A$ and $\boldsymbol{K}_A$ with the axiom $\mathbf{A5}^1_n$. This axiom is applicable since $\neg\boldsymbol{K}_B p \in \mathcal{OL}^1_n$ is A-objective and $\mathbf{K45}_n$-consistent. In fact, this example is provable using $\mathbf{AX}_n$, that is, the proof theory proposed in [Lakemeyer, 1993]. ∎

Example 3.3.23. In contrast to both Example 3.3.21 and Example 3.3.22, let us suppose that Alice makes more modest assumptions about Bob. In particular, she considers that

If I do not believe Bob to only know q, then q is not all that he knows.

Then we are interested in proving:

$$\vdash \boldsymbol{O}_A(\neg\boldsymbol{K}_A\boldsymbol{O}_B q \supset \neg\boldsymbol{O}_B q) \supset \boldsymbol{K}_A\neg\boldsymbol{O}_B q.$$

Let β denote the default $\neg\boldsymbol{K}_A\boldsymbol{O}_B q \supset \neg\boldsymbol{O}_B q$. Note that as far as the default goes, it differs from the one from Example 3.3.22 in containing $\boldsymbol{O}_B q$ instead of $\boldsymbol{K}_B q$. For this reason, we are not able to apply $\mathbf{A5}^1_n$ from $\mathbf{AX}_n$ because $\boldsymbol{O}_B q$ is not a basic formula. Nevertheless, it is provable using $\mathbf{AX}^t_n$, and we see below that the proof is quite similar:

1. $\boldsymbol{O}_A\beta \supset (\boldsymbol{K}_A\neg\boldsymbol{K}_A\boldsymbol{O}_B q \supset \boldsymbol{K}_A\neg\boldsymbol{O}_B q)$ **Def**, **PL**, $\mathbf{A2}_n$

2. $\boldsymbol{O}_A\beta \supset (\boldsymbol{N}_A\neg\boldsymbol{K}_A\boldsymbol{O}_B q \wedge \boldsymbol{N}_A\boldsymbol{O}_B q)$ **Def**, **PL**, $\mathbf{K45}_n$

3. $\boldsymbol{N}_A\boldsymbol{O}_B q \supset \neg\boldsymbol{K}_A\boldsymbol{O}_B q$ $\mathbf{A5}^2_n$

4. $\neg\boldsymbol{K}_A\boldsymbol{O}_B q \supset \boldsymbol{K}_A\neg\boldsymbol{K}_A\boldsymbol{O}_B q$ $\mathbf{A4}_n$

5. $\boldsymbol{O}_A\beta \supset \boldsymbol{K}_A\neg\boldsymbol{K}_A\boldsymbol{O}_B q$ 2, 3, 4, **PL**

6. $\boldsymbol{O}_A\beta \supset \boldsymbol{K}_A\neg\boldsymbol{O}_B q$ 1, 5, **PL**

The only difference is in line 3, where we make use of $\mathbf{A5}^2_n$. This axiom is applicable because $\neg\boldsymbol{O}_B q \in \mathcal{OL}^1_n$ is A-objective and consistent wrt $\mathbf{AX}^1_n$. We remark that this proof requires reasoning with the *satisfiability* modal operator in [Halpern and Lakemeyer, 2001, see Example 6.1]. ∎

Example 3.3.24. For our last example, we suppose that Alice believes that Bob also knows of the big blocks default. Formally, suppose Alice assumes the following:

$$\boldsymbol{O}_B(p \wedge \neg\boldsymbol{K}_B\neg q \supset q) \tag{3.1}$$

We are now interested in showing that if A believes (3.1), then A believes that B believes that D is located in the storage. That is, if we let γ denote (3.1) then we prove:

$$\vdash \boldsymbol{K}_A\gamma \supset \boldsymbol{K}_A\boldsymbol{K}_B q.$$

Observe that by adding the agent index B in Example 3.1.12, we have shown $\gamma \supset \boldsymbol{K}_B q$. Then:

1. $\boldsymbol{O}_B(p \wedge \neg \boldsymbol{K}_B \neg q \supset q) \supset \boldsymbol{K}_B q$ see Example 3.1.12

2. $\boldsymbol{K}_A(\boldsymbol{O}_B(p \wedge \neg \boldsymbol{K}_B \neg q \supset q) \supset \boldsymbol{K}_B q)$ 1, **NEC**

3. $\boldsymbol{K}_A(\boldsymbol{O}_B(p \wedge \neg \boldsymbol{K}_B \neg q \supset q)) \supset \boldsymbol{K}_A \boldsymbol{K}_B q$ 2, $\mathbf{A2}_n$

Intuitively, A attributes nonmonotonic reasoning abilities to B and therefore draws conclusions based on B's reasoning patterns. ∎

3.3.4 Axiomatizing Validity

We already discussed that both [Lakemeyer, 1993] and [Halpern, 1993] fail to capture the intuitions of multiagent only knowing. Besides, Lakemeyer's proof theory is restricted to formulas in $\mathcal{OL}_n^-$. Extending this work, Halpern and Lakemeyer [2001] proposed a multiagent only knowing logic that does handle the nesting of $\boldsymbol{N}_i$ operators. However, as discussed, there are two undesirable features. The first is a semantics based on canonical models, and the second is a proof theory that axiomatizes validity. Although such a construction is far from natural, we show in this section that they do indeed capture the desired properties of only knowing. More precisely, we show that the approaches agree on provable formulas. This mainly instructs us that our axiomatization avoids such problems in a reasonable manner.

We begin by presenting the main formal features of their approach. More precisely, this involves enriching $\mathcal{OL}_n$ with a new modal operator $\boldsymbol{V}$, for validity. We read $\boldsymbol{V}\alpha$ as "α is valid". A modal operator $\boldsymbol{S}$, for satisfiability, is freely used such that $\boldsymbol{S}(\alpha)$ syntactically denotes $\neg \boldsymbol{V}(\neg\alpha)$. Let $\mathcal{OL}_n^+$ be the addition of $\boldsymbol{V}$ to $\mathcal{OL}_n$.

To handle the circularity of consistency, Halpern and Lakemeyer propose an axiom system, which we will denote as $\mathbf{AX}'_n$, that has the following set of schemas:

Axioms:

$\mathbf{A1}_n - \mathbf{A4}_n$,

$\mathbf{A5}'_n$. $\boldsymbol{S}(\neg\alpha) \supset (\boldsymbol{N}_i\alpha \supset \neg \boldsymbol{K}_i\alpha)$ if α is i-objective,

V1. $\boldsymbol{V}(\alpha) \wedge \boldsymbol{V}(\alpha \supset \beta) \supset \boldsymbol{V}(\beta)$,

V2. $\boldsymbol{S}(\alpha)$ if α is a satisfiable propositional formula,

V3. $\bigwedge \boldsymbol{S}(\alpha \wedge \beta_z) \wedge \bigwedge \boldsymbol{S}(\gamma \wedge \delta_{z'}) \wedge \boldsymbol{V}(\alpha \vee \gamma) \supset$

$$\boldsymbol{S}(\boldsymbol{K}_i\alpha \wedge \bigwedge \neg \boldsymbol{K}_i \neg \beta_z \wedge \boldsymbol{N}_i\gamma \wedge \bigwedge \neg \boldsymbol{N}_i \neg \delta_{z'})$$

if $\alpha, \beta_z, \gamma, \delta_{z'}$ are i-objective formulas,

V4. $\boldsymbol{S}(\alpha) \wedge \boldsymbol{S}(\beta) \supset \boldsymbol{S}(\alpha \wedge \beta)$ if α is i-objective and β is i-subjective.

Inference Rules:

MP and **NEC**,

NEC$_{\boldsymbol{V}}$. From α infer $\boldsymbol{V}(\alpha)$.

The axioms seem somewhat mysterious but intuitively, they allow us to extend the notion of consistency from propositional formulas to modal ones. To see this, consider the axiom **V2** which allows us to include satisfiable propositional formulas in the scope of $\boldsymbol{S}$. **V3** then allows us to construct consistent i-subjective formulas from i-objective ones. Note also that the generalized **A5** in this proof theory is **A5**$'_n$. This then allows us to invoke the $\boldsymbol{N}_i$ vs. $\boldsymbol{K}_i$ relationship on all falsifiable i-objective formulas, that is, the formulas α for which $\boldsymbol{S}(\neg\alpha)$ is provable from the given premises. The axiom **V1** and the inference rule **NEC**$_{\boldsymbol{V}}$ make the modality $\boldsymbol{V}$ a *normal modal operator* [Chellas, 1980].

We mentioned in Section 3.3.2 that $\mathbf{AX}'_n$ characterizes the extended canonical model. We do not present the details of this structure here since we will not directly make use of it.

In order to enable comparisons with the $\mathbf{AX}'_n$ approach, it seems obvious that we need to extend our logic to handle the enriched language $\mathcal{OL}^+_n$.[13] However, it turns out that there is a much simpler alternative, which rests on the following result proved by Halpern and Lakemeyer:

Theorem 3.3.25. *[Halpern and Lakemeyer, 2001]*
Every formula $\alpha \in \mathcal{OL}^+_n$ is provably equivalent (wrt $\mathbf{AX}'_n$) to some formula $\alpha' \in \mathcal{OL}_n$.

The theorem essentially tells us that as far as we are concerned regarding derivations of $\mathbf{AX}'_n$, it suffices to restrict our attention to the language $\mathcal{OL}_n$. Now, to obtain a correspondence between the two axiom systems, we first demonstrate that $\boldsymbol{S}(\alpha)$, where $\alpha \in \mathcal{OL}^t_n$, is provable by $\mathbf{AX}'_n$ iff α is consistent wrt $\mathbf{AX}^t_n$. This will then allow us to use Theorem 3.3.25 to establish the agreement on provable formulas.

We begin by proving the following variant of Lemma 3.3.19, and a corollary thereof.

Lemma 3.3.26. *Suppose $\phi, \psi \in \mathcal{OL}^{t-1}_n$ are i-objective formulas. If they are consistent wrt $\mathbf{AX}^t_n$ and $\models \phi \vee \psi$ then $\boldsymbol{K}_i\phi \wedge \boldsymbol{N}_i\psi$ is consistent wrt $\mathbf{AX}^t_n$.*

Proof: Let i be A. Assume the contrary. That is, $\mathbf{AX}^t_n \vdash \neg\boldsymbol{K}_A\phi \vee \neg\boldsymbol{N}_A\psi$. Then by Theorem 3.3.17, $\models \neg\boldsymbol{K}_A\phi \vee \neg\boldsymbol{N}_A\psi$. Further, $\models \neg\boldsymbol{K}_A(\phi \vee \psi) \vee \neg\boldsymbol{N}_A(\phi \vee \psi)$.

Suppose that $|\phi \vee \psi|_A \leq k$. Let e^k_A be any structure, and clearly e^k_A must satisfy $\neg\boldsymbol{K}_A(\phi \vee \psi) \vee \neg\boldsymbol{N}_A(\phi \vee \psi)$. But $e^k_A, \{\}, w \not\models \neg\boldsymbol{K}_A(\phi \vee \psi)$ because there cannot be any $(w', e^{k-1}_B) \in e^k_A$ such that $e^k_A, e^{k-1}_B, w' \models \neg(\phi \vee \psi)$. For similar reasons, $e^k_A, \{\}, w \not\models \neg\boldsymbol{N}_A(\phi \vee \psi)$. Therefore $\neg\boldsymbol{K}_A(\phi \vee \psi) \vee \neg\boldsymbol{N}_A(\phi \vee \psi)$ is not valid. ∎

Corollary 3.3.27. *Suppose $\phi_0, \phi_1, \ldots, \phi_l, \psi_0, \psi_1, \ldots, \psi_{l'}$ are i-objective formulas from $\mathcal{OL}^{t-1}_n$. Suppose that $\phi_0 \wedge \phi_z$ is consistent wrt $\mathbf{AX}^t_n$ for every $z \geq 1$. Similarly, suppose that $\psi_0 \wedge \psi_z$ is consistent wrt $\mathbf{AX}^t_n$ for every $z \geq 1$. If $\models \phi_0 \vee \psi_0$ then the following formula is also consistent wrt $\mathbf{AX}^t_n$:*

$$\boldsymbol{K}_i\phi_0 \wedge \bigwedge \neg\boldsymbol{K}_i\neg\phi_z \wedge \boldsymbol{N}_i\psi_0 \wedge \bigwedge \neg\boldsymbol{N}_i\neg\psi_{z'}.$$

[13] This is indeed the direction pursued in [Belle and Lakemeyer, 2010a]. But the methodology considered in the sequel is cleaner and more direct.

Proof: We claim that $\boldsymbol{K}_i\phi_0 \wedge \neg\boldsymbol{K}_i\neg\phi_z$, $z \geq 1$, is consistent. Suppose not. Then $\neg(\boldsymbol{K}_i\phi_0 \wedge \neg\boldsymbol{K}_i\neg\phi_z)$ is provable. That is, $\boldsymbol{K}_i\phi_0 \supset \boldsymbol{K}_i\neg\phi_z$ is provable. By standard modal reasoning, it follows that $\boldsymbol{K}_i(\phi_0 \supset \neg\phi_z)$ is provable. By way of the soundness result from Theorem 3.3.17, it follows that $\boldsymbol{K}_i(\phi_0 \supset \neg\phi_z)$ is valid, which also means that $\phi_0 \supset \neg\phi_z$ is valid. This contradicts the consistency of $\phi_0 \wedge \phi_z$. A similar argument establishes the consistency of $\boldsymbol{N}_i\psi_0 \wedge \neg\boldsymbol{N}_i\neg\psi_z$, $z \geq 1$. Lemma 3.3.26 establishes the consistency of $\boldsymbol{K}_i\phi_0 \wedge \boldsymbol{N}_i\psi_0$. It then follows that the formula of interest is also consistent. ∎

Theorem 3.3.28. *For all $\alpha \in \mathcal{OL}_n^t$, $\mathbf{AX}'_n \vdash \boldsymbol{S}(\alpha)$ iff α is consistent wrt $\mathbf{AX}_n^t$.*

Proof: The proof is by induction on the structure of the formula for both the directions.

For the only-if direction, suppose α is a satisfiable propositional formula. Then, by an application of **V2**, $\mathbf{AX}'_n \vdash \boldsymbol{S}(\alpha)$. But α is clearly $\mathbf{AX}_n^t$-consistent, since it is a propositional consistent formula. Assume that the hypothesis holds for i-objective formulas.

Suppose that we have proved $\boldsymbol{S}(\phi_0 \wedge \phi_z)$ for $z \geq 1$ and $\boldsymbol{S}(\psi_0 \wedge \psi_{z'})$ for $z \geq 1$, where $\phi_i \in \mathcal{OL}_n^{t-1}$ and $\psi_i \in \mathcal{OL}_n^{t-1}$ are i-objective formulas. Suppose also that we have proven $\neg\boldsymbol{S}(\neg\phi_0 \wedge \neg\psi_0)$. Using **V3**, $\mathbf{AX}'_n \vdash \boldsymbol{S}(\zeta)$ where ζ is the formula:

$$\boldsymbol{K}_i\phi_0 \wedge \bigwedge \neg\boldsymbol{K}_i\neg\phi_z \wedge \boldsymbol{N}_i\psi_0 \wedge \bigwedge \neg\boldsymbol{N}_i\neg\psi_{z'}. \tag{3.2}$$

Using the hypothesis, both $\phi_0 \wedge \phi_z$ and $\psi_0 \wedge \psi_{z'}$ are consistent wrt $\mathbf{AX}_n^t$. Further, $\neg\phi_0 \wedge \neg\psi_0$ is not $\mathbf{AX}_n^t$-consistent allowing us to prove $\vdash \phi_0 \vee \psi_0$. By Theorem 3.3.17, $\models \phi_0 \vee \psi_0$. Therefore, by way of Corollary 3.3.27, the consistency of ζ is implied.

Finally, suppose that we have proved $\boldsymbol{S}(\alpha)$ and $\boldsymbol{S}(\beta)$ where α is i-objective and β is i-subjective. By an application of **V4**, $\mathbf{AX}'_n \vdash \boldsymbol{S}(\alpha \wedge \beta)$. By the induction hypothesis, α is $\mathbf{AX}_n^t$-consistent and therefore it is satisfiable. Without any loss of generality, let i be A. So suppose that $(\{\}, e_B^j, w)$ is a model for α. By the induction hypothesis, β is satisfiable. Then let $(e_A^k, \{\}, w)$ be a model for β. Then clearly $e_A^k, e_B^j, w \models \alpha \wedge \beta$.

Conversely, suppose α is a propositionally consistent formula. Then it is also consistent wrt $\mathbf{AX}_n^t$. This implies that α is satisfiable, allowing us to use **V2** to derive $\boldsymbol{S}(\alpha)$. Assume that the hypothesis holds for i-objective formulas.

Now given an arbitrary formula α that is consistent wrt $\mathbf{AX}_n^t$, by means of Lemma 3.3.9, it suffices to consider it in ONF. That is, α is a disjunction of formulas α' of the form: $\sigma \wedge \zeta \wedge \xi$, where

ζ is $\boldsymbol{K}_A\phi_{A0} \wedge \bigwedge \neg\boldsymbol{K}_A\phi_{Az} \wedge \boldsymbol{N}_A\varphi_{A0} \wedge \bigwedge \boldsymbol{N}_A\varphi_{Az'}$, and

ξ is $\boldsymbol{K}_B\phi_{B0} \wedge \bigwedge \neg\boldsymbol{K}_B\phi_{Bh} \wedge \boldsymbol{N}_B\varphi_{B0} \wedge \bigwedge \boldsymbol{N}_B\varphi_{Bh'}$

and ϕ_i, φ_i are i-objective formulas and σ is a propositional formula. If α is consistent wrt $\mathbf{AX}_n^t$, then there is some α' which is consistent.

We proceed as follows. In Proposition 5.1 in [Halpern and Lakemeyer, 2001] it is shown that $\mathbf{AX}'_n \vdash \boldsymbol{S}(\gamma \vee \delta) \equiv \boldsymbol{S}(\gamma) \vee \boldsymbol{S}(\delta)$. Therefore, given any disjunct α' that is consistent wrt $\mathbf{AX}_n^t$ if we are able to show that $\mathbf{AX}'_n \vdash \boldsymbol{S}(\alpha')$ then it follows that $\mathbf{AX}'_n \vdash \boldsymbol{S}(\alpha)$.

So consider α' that is is consistent wrt $\mathbf{AX}'_n$. By the induction hypothesis, $\mathbf{AX}'_n \vdash \boldsymbol{S}(\sigma)$. We proceed to show that $\mathbf{AX}'_n \vdash \boldsymbol{S}(\zeta)$. Analogously, $\mathbf{AX}'_n \vdash \boldsymbol{S}(\xi)$. Then by the use of **V4**, we first establish $\mathbf{AX}'_n \vdash \boldsymbol{S}(\sigma \wedge \xi)$, since σ is B-objective and ξ is B-subjective. Next, by applying the same axiom, we prove $\mathbf{AX}'_n \vdash \boldsymbol{S}(\sigma \wedge \xi \wedge \zeta)$,

since $\sigma \wedge \xi$ is A-objective and ζ is A-subjective, which allows us to prove $\mathbf{AX}'_n \vdash \boldsymbol{S}(\alpha')$. Thus, $\mathbf{AX}'_n \vdash \boldsymbol{S}(\alpha)$ by the argument above.

So to prove $\mathbf{AX}'_n \vdash \boldsymbol{S}(\zeta)$, we can assume that $\phi_{A0} \wedge \neg\phi_{Az}$, $z \geq 1$ is consistent. For suppose not. Then $\models \neg\phi_{A0} \vee \phi_{Az}$. This means that $\boldsymbol{K}_A\phi_{A0} \wedge \neg\boldsymbol{K}_A\phi_{Az}$ is not satisfiable, contradicting the consistency of $\boldsymbol{K}_A\phi_{A0} \wedge \neg\boldsymbol{K}_A\phi_{Az}$ due to the soundness result, *viz.* Theorem 3.3.17. On the same lines, it is easy to show that $\varphi_{A0} \wedge \neg\varphi_{Az}$, $z \geq 1$, is also consistent. Lastly, owing to the consistency of $\boldsymbol{K}_A\phi_{A0} \wedge \boldsymbol{N}_A\varphi_{A0}$, by means of Lemma 3.3.19 it follows that $\models \phi_{A0} \vee \varphi_{A0}$. Thus, by means of the induction hypothesis, we have $\mathbf{AX}'_n \vdash \boldsymbol{S}(\phi_{A0} \wedge \neg\phi_{Az})$, $\mathbf{AX}'_n \vdash \boldsymbol{S}(\varphi_{A0} \wedge \neg\varphi_{Az'})$ and $\mathbf{AX}'_n \vdash \neg\boldsymbol{S}(\neg\phi_{A0} \wedge \neg\varphi_{A0})$ and so by **V3**, $\mathbf{AX}'_n \vdash \boldsymbol{S}(\zeta)$. ∎

This allows us to obtain the main result of the section:

Theorem 3.3.29. *For all* $\alpha \in \mathcal{OL}^t_n$, $\mathbf{AX}'_n \vdash \alpha$ *iff* $\mathbf{AX}^t_n \vdash \alpha$.

Proof: For the if direction, we prove by contradiction. Suppose $\mathbf{AX}^t_n \vdash \alpha$ and $\mathbf{AX}'_n \nvdash \alpha$. Proposition 5.1 in [Halpern and Lakemeyer, 2001] states that if α is not provable from $\mathbf{AX}'_n$, then $\mathbf{AX}'_n \vdash \neg\boldsymbol{V}(\alpha)$, *i.e.* $\mathbf{AX}'_n \vdash \boldsymbol{S}(\neg\alpha)$. But by assumption, $\neg\alpha$ is not consistent wrt $\mathbf{AX}^t_n$. Then $\mathbf{AX}'_n \nvdash \boldsymbol{S}(\neg\alpha)$ because of Theorem 3.3.28. This is a contradiction.

Conversely, suppose $\mathbf{AX}'_n \vdash \alpha$. Then by $\mathbf{NEC}_{\boldsymbol{V}}$, it follows that $\mathbf{AX}'_n \vdash \boldsymbol{V}(\alpha)$, or $\mathbf{AX}'_n \vdash \neg\boldsymbol{S}(\neg\alpha)$, or $\mathbf{AX}'_n \nvdash \boldsymbol{S}(\neg\alpha)$. Thus, $\neg\alpha$ is not consistent wrt $\mathbf{AX}^t_n$ by Theorem 3.3.28. Therefore, $\mathbf{AX}^t_n \vdash \alpha$. ∎

Recall the main message of Theorem 3.3.25: it is sufficient to restrict ourselves to formulas from $\mathcal{OL}_n$ with regards to the Halpern and Lakemeyer approach. Therefore, Theorem 3.3.29 establishes an exact correspondence between the Halpern and Lakemeyer approach and our approach.

3.4 Concluding Remarks

In this chapter, we were concerned with multiagent only knowing. We reviewed the logic of only knowing $\mathcal{OL}$, and then presented a semantics for multiagent only knowing. Unlike previous attempts, our semantics was proposed for the full first-order language. We also showed, by means of various arguments, to have appropriately generalized $\mathcal{OL}$ to the many agent case. We then proved that for the propositional fragment, the semantics is characterized by an axiomatization with which we are able to derive certain kinds of nonmonotonic conclusions. In the process, we drew comparisons to earlier attempts that reason about only knowing with many agents. We also briefly mentioned Levesque's result regarding the relationship between AEL and $\mathcal{OL}$. For those interested, a generalization of this characterization to the many agent case appears in [Belle and Lakemeyer, 2011a]. This concludes our work for the static case.

In the next chapter, we review an amalgamation of $\mathcal{OL}$ and actions by [Lakemeyer and Levesque, 2004]. Then based on the results established in this chapter, we consider an amalgamation of $\mathcal{OL}_n$ and a theory of actions.

Chapter 4

Projection with Many Agents by Regression

In the previous chapter, we were concerned with obtaining a suitable semantical basis for multiagent only knowing, but our attention was limited to the static case. In the remainder, we will be concerned with reasoning about action and the problem of projection.

The first approach to projection we consider in this thesis is based on the idea of transforming queries about the future to a query about the initial KB. We begin by reviewing the logic $\mathcal{ES}$ which is an amalgamation of $\mathcal{OL}$ and the situation calculus by Lakemeyer and Levesque [2004; 2011]. Briefly, $\mathcal{ES}$ is a situation-suppressed reconstruction of the situation calculus as considered in Section 2.3.1 which captures much of the expressive power of the original but while being amenable to more straightforward semantic proofs. More importantly, $\mathcal{ES}$ allows for the same style of semantic arguments and analysis that we are familiar with from the previous chapter. The regression property, as considered by Reiter [2001] and Scherl and Levesque [2003], is also obtained for the logic, allowing us to reduce both subjective and objective queries about the future to a query about the initial situation. By an application of the representation theorem (Section 3.1.3), epistemic queries can be reduced further to objective ones and therefore, no modal reasoning is necessary.

Based on this formal theory and the results on multiagent only knowing from the previous chapter, we extend $\mathcal{ES}$ (and its features) to the multiagent case. It is worth noting that a number of conceptual difficulties arise when multiple agents are involved. For instance, the beliefs that agents hold about the dynamics of the world may differ arbitrarily. Nevertheless, we show that a regression property is still provable in a generalized framework which allows us to reduce multiagent beliefs after actions to what is believed initially. Finally, we also extend the representation theorem to the many agent case, which allows us to reduce multiagent beliefs about the initial situation to pure first-order reasoning.

4.1 The Logic $\mathcal{ES}$ = $\mathcal{OL}$ + Actions

An early proposal that integrates the situation calculus and only knowing was considered in [Lakemeyer, 1996] and later refined in [Lakemeyer and Levesque, 1998]. We remark that only knowing cannot be inte-

grated in a simple fashion into the epistemic situation calculus of Scherl and Levesque [2003], say by means of a companion fluent to *Know*, mainly because the notion of only knowing requires the existence of "enough" worlds, as noted in the previous chapter. Thus, Lakemeyer and Levesque appeal to a possible-worlds treatment, where a world is a *tree of situations*. In a sense, this closely follows the models for the foundational axioms. Similar to the situation calculus, the early proposals reify situations in the object language allowing us to quantify over situations, among other things. However, unlike $\mathcal{OL}$, they are not able interpret quantifiers substitutionally. Part of this problem is the reification of situations, which is to say that the foundational axioms required them to consider an uncountable number of situations. The definition of knowledge turned out to be also complex, involving second-order quantifiers.

The logic $\mathcal{ES}$ is a much cleaner amalgamation of actions with $\mathcal{OL}$. Situations are no longer included in the language, but the main features of the situation calculus such as the successor state axioms (incorporating the solution to the frame problem) and a regression property are a part of the formalism. More importantly, as in $\mathcal{OL}$, first-order variables are understood substitutionally in $\mathcal{ES}$.

To reason about actions, a *dynamic logic* like syntax is used [Harel et al., 2000]. For example,

$$[forward]distance \neq 1$$

is a well-formed formula in $\mathcal{ES}$ and says that performing a *forward* action results in the robot not being 1 unit away. Moreover, in order to recast a notion of basic action theories in $\mathcal{ES}$, the quantification of actions is an essential prerequisite to formulate Reiter-style successor state axioms. To this end, the language includes *action variables* and *action names*, which then also allows a substitutional interpretation of quantifiers over actions.

Before presenting the formal details, here are the main ingredients:

1. *Fluents and Rigids*: Like in the situation calculus, we will have *rigids*, whose value does not change over actions, and *fluents* whose values do change over actions. However, since situations are not present in the object language, fluents and rigids will have to be differentiated both syntactically and semantically.

2. *Standard names*: As hinted above, we will have two (countably infinite) sets of standard names, one of the action sort and the other of the object sort. In other words, we could imagine the language of $\mathcal{OL}$ augmented with a new set of names for actions.

3. *Knowledge*: Unlike the situation calculus, we will not represent knowledge in terms of a fluent. We will instead include the familiar modal operators $\boldsymbol{K}$ and $\boldsymbol{O}$. The agents can obtain more knowledge by means of *sensing*.

Let us be clear on the fact that although at first glance $\mathcal{ES}$ seems somewhat different from the situation calculus considered in Section 2.3.1, a result in [Lakemeyer and Levesque, 2011] establishes that when restricted to the projection problem in the context of basic action theories, $\mathcal{ES}$ is only a *notational variant*. That is, under reasonable assumptions, the valid sentences of $\mathcal{ES}$ can be reformulated as entailments in the original one. It is also worth noting that since the language is defined semantically a number of mathematical proofs about the formalism, such as Reiter's regression property and properties about knowledge, are considerably simpler to establish in $\mathcal{ES}$ owing to its simple model theory [Lakemeyer and Levesque, 2004].

The Language

Symbols are taken from the following vocabulary:

- first-order variables of the object sort: $x, y, h, \ldots$;
- first-order variables of the action sort: $v_1, v_2, \ldots$;
- fluent function symbols of arity k: $f_1^k, f_2^k, \ldots$; *e.g. distance*, *teacher*;
- rigid function symbols of arity k: $g_1^k, g_2^k, \ldots$; *e.g. forward*, *bestAction*;
- connectives and other symbols: $=, \vee, \neg, \forall, \boldsymbol{K}, \boldsymbol{O}, [v], \Box$, parenthesis, period and comma;
- a countable infinite set of standard names $\mathcal{N}$, of the object sort: ${}^{\#}0, {}^{\#}1, {}^{\#}2, \ldots, obj5, \ldots$[1]
- standard names of the action sort, which are constructed from $\mathcal{N}$:

$$\mathcal{A} = \{A(n_1, \ldots, n_k) \mid n_i \in \mathcal{N}, \text{ and } A \text{ is a (rigid) function of the action sort}\};^{[2]}$$

e.g. drop(*obj5*),*forward*;

We let $\mathcal{Q} = \mathcal{N} \cup \mathcal{A}$ be the set of all standard names. We let n and m (possibly decorated with subscripts and superscripts) schematically denote elements of $\mathcal{Q}$. We let r (possibly decorated with subscripts and superscripts) schematically denote elements of $\mathcal{A}$.

Two Simplifying Assumptions

In the interest of simplifying the presentation, we have made the following two inessential assumptions about the underlying language:

1. The language does not include predicates. This is not a definitive restriction, since it is possible to model predicates using functions. The approach we take in this thesis is by letting the name 1 denote truth, while every other assignment denotes falsity. More precisely, if we are capturing a predicate $P(\vec{x})$ by means of a function $f(\vec{x})$, then $f(\vec{x}) = 1$ denotes that the predicate is TRUE and $f(\vec{x}) \neq 1$ (as well as $f(\vec{x}) = n$ for every name n other than 1) denotes that the predicate is FALSE.[3] In this sense, we are allowing full first-order expressivity.
2. There are no rigids of the object sort. In Section 4.1.1, we show how fluents can be used to capture properties that remain unchanged over any sequence of actions.

We reiterate that none of the technical results hinge on these assumptions. Now, since we are not including predicates, *Poss* and *SF* will be distinguished functional fluents assumed to be part of our vocabulary.

[1] We assume that the set of rational numbers $\mathbb{Q}$, closed under standard mathematical operators $\times, +, -, \div$, is included in the set of names $\mathcal{N}$.

[2] That is, as is standard in the situation calculus, we assume that all action function symbols are rigid.

[3] This is similar in spirit to how predicates are captured in COMMON LISP [Steele, 1984], except that there is a single false value `nil`, and every non-`nil` value is evaluated as *true*.

Terms

Terms are of the sort action or object, and they are the least set of expressions such that:

1. Every name and first-order variable is a term of the corresponding sort. We follow the notational convention of writing $\vec{t}$ to denote a vector of terms and t_i to denote any term from $\vec{t}$. We follow the same notation for variables and names also.

2. If $\vec{t}$ are terms of the object sort and A is a k-ary function of the action sort, then $A(\vec{t})$ is a term.

3. If $\vec{t}$ are terms (of any sort) and f is a k-ary function of the object sort, then $f(\vec{t})$ is a term.

By *primitive action term*, we mean any element of $\mathcal{A}$. By *primitive object term*, we mean one of the form $f(\vec{n})$, where f is a function of the object sort and $n_i \in \mathcal{Q}$.

We remark that we are restricting the parameters of action functions to be of the object sort. On the one hand, this greatly simplifies the readability of technical results to follow. On the other, our treatment of primitive actions as standard names has the added feature that the unique name assumption for actions is built into the logic.[4] Moreover, it is hard to imagine applications where this restriction leads to any conceptual difficulties.

Formulas

The *well-formed formulas* of $\mathcal{ES}$ form the least set such that

1. If t and t' are terms, then $t = t'$ is an (atomic) formula.

2. If t is an action term and α is a formula, then $[t]\alpha$ is a formula.

3. If α and β are formulas, and x is a first-order variable then the following are also formulas: $\neg\alpha, \alpha \vee \beta, \forall x\alpha, \Box\alpha, \boldsymbol{K}\alpha, \boldsymbol{O}\alpha$.

We read $[t]\alpha$ as "α holds after the action t". We read $\Box\alpha$ as "α holds after any sequence of actions".

A formula without any free variables is called a *sentence*. We also refer to certain kinds of formulas with the following terminology:

- A formula with no $\Box$ operators is called *bounded*.
- A formula with no $[t]$ or $\Box$ operators is called *static*.
- As in $\mathcal{OL}$, a formula with no $\boldsymbol{K}$ or $\boldsymbol{O}$, *i.e.* no epistemic operators, is called *objective*.
- A formula with no $\boldsymbol{K}, \boldsymbol{O}, [t]$, *Poss* or *SF* is called *fluent*.

[4] The variant of $\mathcal{ES}$ presented in the sequel differs in precisely this manner from the ones appearing in literature [Lakemeyer and Levesque, 2004, 2011]. That is to say, in [Lakemeyer and Levesque, 2011] primitive actions are handled in a more general manner, but then the unique name assumption needs to be additionally axiomatized as part of the background theory.

The Semantics

In $\mathcal{OL}$ we were concerned with providing meaning to static formulas. In $\mathcal{ES}$, the semantics has to clarify how fluents are to be handled after any given sequence of actions. We extend our idea of a model, which consisted earlier of an epistemic state and a world, to the tuple (e, w, z) where z denotes some (finite) sequence of actions. The idea is that the world determines the truth of objective formulas, both initially and after any sequence of actions.

More precisely, let $\mathcal{Z}$ denote all finite sequences of action names from $\mathcal{A}$, including $\langle\rangle$ which is the empty sequence (corresponding to the initial situation). Then,

- a world $w \in \mathcal{W}$ is a function from primitive object terms and $\mathcal{Z}$ to $\mathcal{N}$;
- an epistemic state e is any set of worlds.

The intuition is that each world is a *tree of situations*, very much in the spirit of the Tarskian models for the foundational axioms of the situation calculus. Figure 4.1 depicts a world.

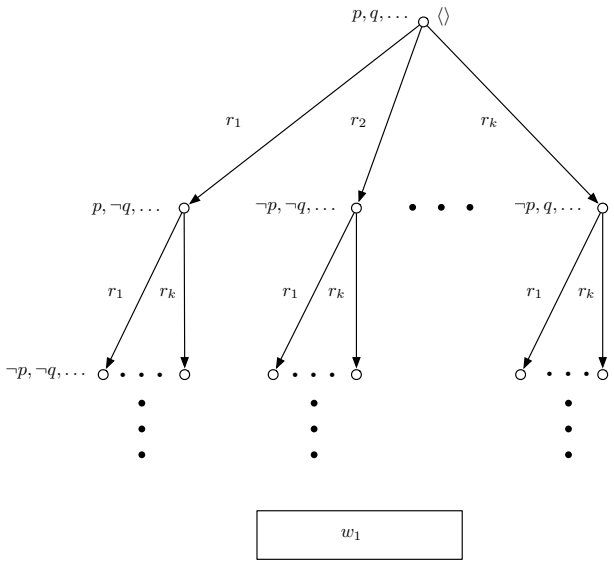

Figure 4.1: This figure depicts a world, where as many as k actions are depicted. Here, p and q denote atomic formulas.

The epistemic state e essentially represents the agent's initial beliefs, that is, the initial state of knowledge. But when actions occur, perhaps an agent acquires new information and as a result of this, some of the worlds present in e will no longer be considered possible. We capture this feature by means of a compatibility relation $\simeq_z$. The idea is to verify the truth (wrt the sensing results) for all the worlds in e against the real world, and discard those that disagree on the sensing results. In this way, the agent learns more over the course of actions and can supplement its knowledge with what is true in the real world. Formally,

- $w' \simeq_{\langle\rangle} w$ for all worlds w and w';
- $w' \simeq_{z \cdot r} w$ iff $w' \simeq_z w$, and $w'[SF(r), z] = w[SF(r), z]$.

Note that $\simeq$ is an equivalence relation.[5]

Finally, we interpret arbitrary terms in the language as follows. As in $\mathcal{OL}$, names are rigid designators. Now, given a term t without variables, a world w, and an action sequence z, we define $|t|^z_w$ (to be read as "the coreferring standard name for t given w and z") by:

1. $|t|^z_w = t$ if $t \in \mathcal{Q}$;
2. $|f(\vec{t})|^z_w = w[f(\vec{n}), z]$, where $|t_i|^z_w = n_i$ and f is a function of the object sort;
3. $|A(\vec{t})|^z_w = A(\vec{n})$, where $|t_i|^z_w = n_i$ and A is a function of the action sort.

We are now ready to define the meaning of truth. Given a sentence $\alpha \in \mathcal{ES}$, an epistemic e, a world $w \in \mathcal{W}$ and an action sequence z, a semantics is as follows:

1. $e, w, z \models t_1 = t_2$ iff n_1 and n_2 are identical, where $|t_i|^z_w = n_i$;
2. $e, w, z \models \neg\alpha$ iff $e, w, z \not\models \alpha$;
3. $e, w, z \models \alpha \vee \beta$ iff $e, w, z \models \alpha$ or $e, w, z \models \beta$;
4. $e, w, z \models \forall x\alpha$ iff $e, w, z \models \alpha^x_n$ for every name n of the appropriate sort;
5. $e, w, z \models [t]\alpha$ iff $e, w, z \cdot r \models \alpha$, where $r = |t|^z_w$;
6. $e, w, z \models \Box\alpha$ iff $e, w, z \cdot z' \models \alpha$ for every $z' \in \mathcal{Z}$;
7. $e, w, z \models \boldsymbol{K}\alpha$ iff for all $w' \simeq_z w$, if $w' \in e$ then $e, w', z \models \alpha$;
8. $e, w, z \models \boldsymbol{O}\alpha$ iff for all $w' \simeq_z w$, $w' \in e$ iff $e, w', z \models \alpha$.[6]

Note that first-order variables are understood substitutionally as before. We say that α is true wrt a model (e, w, z) if $e, w, z \models \alpha$. We write $e, w \models \alpha$ to mean $e, w, \langle\rangle \models \alpha$. Given a set of sentences Σ and a sentence α, we write $\Sigma \models \alpha$ to mean that for every e and w, if $e, w \models \alpha'$ for every $\alpha' \in \Sigma$, then $e, w \models \alpha$. Finally, we write $\models \alpha$ to mean $\{\} \models \alpha$.

Properties

Given the similarity to the semantical framework of $\mathcal{OL}$, it is perhaps not surprising that knowledge satisfies **K45** properties and the Barcan property (see Section 3.1.1). Observe that these properties are provable for all sequence of actions, that is, they hold in the scope of the $\Box$ operator. Formally,

Theorem 4.1.1. *[Lakemeyer and Levesque, 2004]*

1. $\models \Box(\boldsymbol{K}\alpha \wedge \boldsymbol{K}(\alpha \supset \beta) \supset \boldsymbol{K}\beta)$,

[5] Our definition of world-compatibility follows Lakemeyer and Levesque [2004]. In later versions of $\mathcal{ES}$ [Lakemeyer and Levesque, 2011], and as originally intended by Scherl and Levesque [2003], the compatibility relation is strengthened by enforcing that r is additionally *executable* in w (by means of *Poss*(r) holding at w). This has the unintended effect of making $\simeq$ a non-equivalence relation. But as pointed out by Lakemeyer and Levesque, an alternate account that would state that the agent learns the value of *Poss* (analogous to *SF*) would allow $\simeq$ to be a full equivalence relation. We ignore these issues for simplicity.

[6] As in $\mathcal{OL}$, the semantics for the $\boldsymbol{O}$ operator differs from the one for $\boldsymbol{K}$ in containing an "iff" instead of an "if".

2. $\models \Box(\boldsymbol{K}\alpha \supset \boldsymbol{K}\boldsymbol{K}\alpha)$,
3. $\models \Box(\neg\boldsymbol{K}\alpha \supset \boldsymbol{K}\neg\boldsymbol{K}\alpha)$,
4. $\models \Box(\forall\vec{x}\boldsymbol{K}\alpha \supset \boldsymbol{K}(\forall\vec{x}\alpha))$,
5. $\models \Box(\exists\vec{x}\boldsymbol{K}\alpha \supset \boldsymbol{K}(\exists\vec{x}\alpha))$.

It is also possible to show that the converse of the Barcan existential property is not valid, as in $\mathcal{OL}$.

4.1.1 Basic Action Theories

We now turn to the equivalent of basic action theories of the situation calculus. Since $\mathcal{ES}$ does not mention situations explicitly, it turns out that basic action theories in $\mathcal{ES}$ do not require the foundational axioms, such as the second-order induction axiom for situations (Section 2.3.1; [Reiter, 2001]).[7]

Definition 4.1.2. (Basic action theories.) Given a set of fluents $\mathcal{F}$, a set $\Sigma \subseteq \mathcal{ES}$ of sentences is called a *basic action theory* (BAT) over $\mathcal{F}$ iff $\Sigma = \Sigma_0 \cup \Sigma_{pre} \cup \Sigma_{post} \cup \Sigma_{sense}$ where Σ only mentions fluents from $\mathcal{F}$ and[8]

1. Σ_0 is any set of fluent sentences;
2. Σ_{pre} is a singleton sentence of the form $\Box Poss(v) = 1 \equiv \pi$, where π is a fluent formula;[9]
3. Σ_{post} is a set of sentences of the form

   $$\Box[v]f(\vec{x}) = y \equiv \gamma_f(\vec{x}, y, v) \vee f(\vec{x}) = y \wedge (\neg\exists h)\, \gamma_f(\vec{x}, h, v),$$

   one for each fluent f, where γ_f is a fluent formula;[10]
4. Σ_{sense} is a sentence similar to the one for *Poss* of the form $\Box SF(v) = x \equiv \varphi$, where φ is a fluent formula. ∎

The idea is that Σ_0 expresses what is true initially, Σ_{pre} is one large precondition axiom, Σ_{post} are the successor state axioms, one per fluent, which are formulated so as to incorporate Reiter's solution the frame problem. Σ_{sense}, like Σ_{pre}, is one large sensing axiom, and we follow the convention [Scherl and Levesque, 2003] that every action returns a sensing result. For actions such as *forward*, which do not return any sensing information, *SF* is defined to return a special standard name 1.

Knowledge about the initial situation may be incomplete, of course. In order to account for false beliefs, the simplest way perhaps is to have two basic action theories Σ and Υ: we let Σ denote the agent's beliefs, and let Υ denote what is true in the real world. The two may differ arbitrarily.

Example 4.1.3. (The simple robot domain.) To illustrate the idea of an action theory, we adapt an example from [Lakemeyer and Levesque, 2011]. Imagine a robot navigating itself in a 1-dimensional world, as shown in Figure 4.2.

[7] Indeed, as readers may have noticed, no second-order features were considered in the presentation of $\mathcal{ES}$.

[8] We follow the usual situation calculus convention that free variables are universally quantified from the outside.

[9] We assume that $\Box$ has lower syntactic precedence than the logical connectives, so that $\Box Poss(v) = 1 \equiv \pi$ stands for $\forall v.\Box(Poss(v) = 1 \equiv \pi)$. Also, let us again clarify that *Poss* is a function, and throughout we interpret $Poss(r) = 1$ to mean that r is executable.

[10] The $[v]$ construct has higher precedence than the logical connectives. So $\Box[v]f(\vec{x}) = y \equiv \gamma_f$ abbreviates $\forall v.\Box([v]f(\vec{x}) = y \equiv \gamma_f)$.

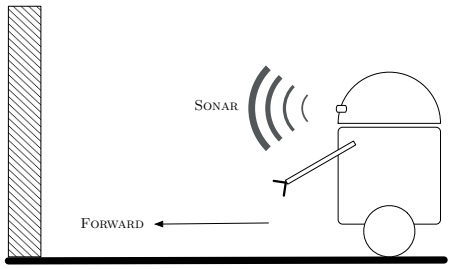

Figure 4.2: A simple robot.

We imagine that the robot can move towards a fixed location and is equipped with a sonar sensor that tells the robot its actual distance to the door. So we might imagine two actions *forward* and *sonar* that does the moving and sensing respectively. We have a single fluent function *distance* which gives the actual distance between the robot and the door. Figure 4.3 axiomatizes the successor state axiom for the fluent.[11]

We imagine that moving forward is only possible when the robot is not already at the fixed location. We will also assume that *sonar* is always executable. Figure 4.3 then illustrates the formalization of the precondition and sensing conditions. The formalization also stipulates that sensing on *forward* simply returns 1 which, as mentioned earlier, is a trivial sensing result.

Suppose that the robot is 4 units away from the door. Suppose also that the robot is uncertain about the distance and believes that it is either 4 or 5 meters away. This uncertainty is captured by differentiating between Υ_0 and Σ_0 in Figure 4.3. Putting this together, we let Υ and Σ be:

- $\Sigma = \Sigma_0 \cup \Sigma_{pre} \cup \Sigma_{post} \cup \Sigma_{sense}$.
- $\Upsilon = \Sigma \cup \Upsilon_0$. ∎

Before concluding our presentation on basic action theories, we show how not allowing rigids of the object sort does not lead to any loss of expressivity, as claimed earlier. In order to capture properties that remain unchanged over any sequence of actions, such as $title(x)$, which returns the title of a book, the modeler is required to use an axiom of the following form as part of the background theory:

$$\Box(\forall x, y, v.\ ([v]title(x) = y) \equiv title(x) = y).$$

This can be understood as saying that the title of x is the same for all sequences of actions.

4.1.2 The Problem of Projection

As discussed earlier, in the context of basic action theories, projection is the task of determining what holds after a number of actions have occurred. For example, after moving forward twice, does it follow that the robot is 4 units away? Formally, this corresponds to the following entailment:

$$\Sigma \models [forward][forward]distance = 4.$$

[11] It is perhaps of interest to the readers to contrast this formulation of the successor state axiom with the one for Reiter's version of the situation calculus from Section 2.3.1.

$\Sigma_{post} = \{\Box[v]distance = x \equiv$

$v = forward \wedge x = distance - 1 \vee$

$v \neq forward \wedge x = distance\}.$

$\Sigma_{pre} = \{\Box Poss(v) = 1 \equiv$

$v = forward \wedge distance > 0\}.$

$\Sigma_{sense} = \{\Box SF(v) = r \equiv$

$v = forward \wedge r = 1 \vee$

$v = sonar \wedge r = distance\}.$

$\Sigma_0 = \{distance = 4 \vee distance = 5\}.$

$\Upsilon_0 = \{distance = 4\}.$

Figure 4.3: The simple robot domain.

In general, the projection task involves determining if

$$\Sigma \models [r_1] \ldots [r_k]\alpha$$

where $r_1, \ldots, r_k$ are primitive actions and α is an arbitrary *objective* sentence.

In the context of knowledge or beliefs, projection can be extended in the sense of determining what *beliefs* hold after a number of actions have occurred. In particular, assuming that all that an agent believes is a basic action theory Σ, and letting Υ denote the action theory that is true in the world, we are interested in determining whether

$$\Upsilon \wedge \boldsymbol{O}\Sigma \models [r_1] \ldots [r_k]\alpha$$

where $r_1, \ldots, r_k$ are as above and α is an arbitrary sentence.

We illustrate projection queries using Example 4.1.3.

Example 4.1.3 continued. Let Υ and Σ be basic action theories from the example. Then the projection queries in Figure 4.4 are logical consequences of $\Upsilon \wedge \boldsymbol{O}(\Sigma)$.

Proof: The proofs are similar, and so we consider item 3. Let $z = \langle forward \cdot sonar \rangle$. Suppose that $e, w \models \Upsilon \wedge \boldsymbol{O}(\Sigma)$. We need to show that $e, w, z \models \boldsymbol{K}(distance = 3)$.

First, consider that $e, w \models [forward](distance = 3)$. That is, since w satisfies $distance = 4$ initially and the basic action theory updates this to $distance = 3$, it follows that $e', w, forward \models (distance = 3)$ for any e'. Note that $w[SF(forward), \langle\rangle] = 1$.

1. $\forall x.\, \neg \boldsymbol{K}(\mathit{distance} \neq x)$.
2. $[\mathit{forward}]\boldsymbol{K}(\mathit{distance} = 3 \vee \mathit{distance} = 4)$.
3. $[\mathit{forward}][\mathit{sonar}]\boldsymbol{K}(\mathit{distance} = 3)$.

Figure 4.4: Sample projection queries in $\mathcal{ES}$.

Second, consider that $e, w \models [\mathit{forward}]\boldsymbol{K}(\mathit{distance} = 3 \vee \mathit{distance} = 4)$. This is because all worlds in e satisfy either $\mathit{distance} = 4$ or $\mathit{distance} = 5$ initially, and the basic action theory updates these values to $\mathit{distance} = 3$ and $\mathit{distance} = 4$ respectively. Further, since $w'[SF(\mathit{forward}), \langle\rangle] = 1$ for all worlds $w' \in e$, it follows that $w' \simeq_{\mathit{forward}} w$.

However, observe that $w[SF(\mathit{sonar}), \mathit{forward}] = 3$. So it follows that only the worlds

$$\{w' \mid w' \models [\mathit{forward}](\mathit{distance} = 3), w' \in e\}$$

remain compatible with w after z. Therefore $e, w, z \models \boldsymbol{K}(\mathit{distance} = 3)$. ∎

4.1.3 Regression

Reiter [2001] developed an important solution to the projection problem in the situation calculus called *regression*. The idea is to reduce a query α about the future to a query α' about the initial situation by successively replacing fluents in α by the *rhs* of the successor state axioms until the resulting sentence α' contains no more actions. We then need to only verify whether α' is entailed by the initial theory. The class of formulas which are amenable to regression are called *regressable formulas*. These roughly correspond to bounded objective sentences in $\mathcal{ES}$. Regression was later defined for the epistemic situation calculus in [Scherl and Levesque, 2003]. The class of formulas that Scherl and Levesque consider roughly correspond to bounded basic sentences. Lakemeyer and Levesque [2011] define the regression of bounded basic sentences in $\mathcal{ES}$, which we summarize below.

Regressing Objective Formulas

Suppose $\Upsilon \models \alpha$, where α is a bounded objective sentence and Υ is a basic action theory. In order to evaluate this entailment via regression, let us assume without any loss of generality that the query α is syntactically reformulated as follows:

1. quantifiers use distinct variables, and we say such formulas are *rectified*;
2. formulas are in a certain normal form called FNF (defined below).

After applying these transformations, the query becomes amenable to regression. The first syntactic manipulation is required because of the way regression handles quantifiers, which can lead to incorrect transformations if the variables are not distinct. The second is required for giving a simple formulation of regression.

Definition 4.1.4. A formula α is in FNF if every function symbol f in α occurs only in equality expressions of the form $f(\vec{t}) = t'$, where t_i and t' are either variables or names. ∎

It is immediate to verify that every formula can be rewritten to one in FNF, and this transformation is linear in the size of the formula. For instance, $f(g(x)) = f'(x)$ is equivalent to $\exists y, h.\ f(y) = h \wedge f'(x) = h \wedge g(x) = y$. Further, by this definition, if a term t appears either as an argument for a function or as an action operator $[t]$, then it follows that it is either an (action) name or a variable. In the following we will use σ to denote sequences that consist of action variables or action names.

Lakemeyer and Levesque [2011] define the regression operator $\mathcal{R}$, which is applicable to any bounded objective formula. If such a formula is not rectified or not in FNF, it is transformed to a formula satisfying these conditions.

Definition 4.1.5. (Regression.) Define $\mathcal{R}[\alpha]$, the regression of α wrt Υ, to be the fluent formula $\mathcal{R}[\langle\rangle, \alpha]$. For any sequence of action names or variables σ, $\mathcal{R}[\sigma, \alpha]$ is defined inductively:

1. $\mathcal{R}[\sigma, t_1 = t_2] = (t_1 = t_2)$ if t_1 and t_2 do not mention fluents;
2. $\mathcal{R}[\sigma, \forall x \alpha] = \forall x \mathcal{R}[\sigma, \alpha]$;
3. $\mathcal{R}[\sigma, \alpha \vee \beta] = \mathcal{R}[\sigma, \alpha] \vee \mathcal{R}[\sigma, \beta]$;
4. $\mathcal{R}[\sigma, \neg\alpha] = \neg\mathcal{R}[\sigma, \alpha]$;
5. $\mathcal{R}[\sigma, [t]\alpha] = \mathcal{R}[\sigma \cdot t, \alpha]$;
6. $\mathcal{R}[\sigma, Poss(t) = 1] = \mathcal{R}[\sigma, \pi_t^v]$;
7. $\mathcal{R}[\sigma, f(\vec{t}) = t']$ for fluents is defined inductively by:
   (a) $\mathcal{R}[\langle\rangle, f(\vec{t}) = t] = f(\vec{t}) = t$;
   (b) $\mathcal{R}[\sigma \cdot t^*, f(\vec{t}) = t'] = \mathcal{R}[\sigma, \exists y.\ (\gamma_f)^{v\ \vec{x}}_{t^*\ \vec{t}} \wedge y = t']$;
8. $\mathcal{R}[\sigma, SF(t) = t'] = \mathcal{R}[\sigma, \varphi^{v\ x}_{t\ t'}]$. ∎

Note that this definition includes π, φ and γ_f which are the *rhs* of the precondition axioms, sensing axioms and the successor state axioms from Υ.

The main result regarding Definition 4.1.5 is that the evaluation of objective bounded sentences reduces to a query about the initial theory.

Theorem 4.1.6. *[Lakemeyer and Levesque, 2004]*
Let Υ be a basic action theory, whose initial theory is Υ_0, and let α be an objective bounded sentence. Then $\mathcal{R}[\alpha]$ is a fluent sentence and satisfies:

$$\Upsilon \models \alpha \quad \textit{iff} \quad \Upsilon_0 \models \mathcal{R}[\alpha].$$

Regressing Subjective Formulas

We now turn to the more general case of regressing bounded sentences that may refer to the agent's knowledge. This needs an equivalent of a successor state axiom for knowledge, which roughly tells us how knowledge can be regressed wrt an action [Scherl and Levesque, 2003]. Lakemeyer and Levesque prove the following theorem:[12]

Theorem 4.1.7. *[Lakemeyer and Levesque, 2004]*

$$\models \Box[v]\boldsymbol{K}(\alpha) \equiv \exists x.\, SF(v) = x \wedge \boldsymbol{K}(SF(v) = x \supset [v]\alpha).$$

This theorem essentially says that knowledge after an action depends on what was known before, and what the future would look like contingent of the sensing result. Note that this theorem is not a stipulation of the action theory, but a theorem of the logic.

We mentioned earlier that in the general case, we will need two basic action theories Υ and Σ. The idea behind regression is to transform objective formulas wrt Υ, while subjective ones are regressed wrt Σ. Consequently, $\mathcal{R}$ is defined wrt both Υ and Σ.

Definition 4.1.5 continued. $\mathcal{R}[\Upsilon, \Sigma, \sigma, \alpha]$ is defined inductively as:

1.-8. as before, except for the arguments Υ and Σ;

9. $\mathcal{R}[\Upsilon, \Sigma, \sigma, \boldsymbol{K}\alpha]$ is defined inductively on σ by:

   (a) $\mathcal{R}[\Upsilon, \Sigma, \langle\rangle, \boldsymbol{K}\alpha] = \boldsymbol{K}(\mathcal{R}[\Sigma, \Sigma, \langle\rangle, \alpha])$;

   (b) $\mathcal{R}[\Upsilon, \Sigma, \sigma \cdot t, \boldsymbol{K}\alpha] = \mathcal{R}[\Upsilon, \Sigma, \sigma, \beta_t^v]$, where β is the *rhs* of Theorem 4.1.7. ∎

That is, regressing $\boldsymbol{K}\alpha$ wrt the initial situation is equivalent to believing the regression of α wrt the theory believed by the agent, *viz.* Σ. More generally, regressing $\boldsymbol{K}\alpha$ after an action t is equivalent to regressing the *rhs* of Theorem 4.1.7.

As an analogue to Theorem 4.1.6, Lakemeyer and Levesque prove the following:

Theorem 4.1.8. *[Lakemeyer and Levesque, 2004]*
Let α be a bounded basic sentence. Then $\mathcal{R}[\Upsilon, \Sigma, \langle\rangle, \alpha]$ is a static sentence and satisfies:

$$\Upsilon \wedge \boldsymbol{O}\Sigma \models \alpha \quad \textit{iff} \quad \Upsilon_0 \wedge \boldsymbol{O}\Sigma_0 \models \mathcal{R}[\Upsilon, \Sigma, \langle\rangle, \alpha].$$

That is, we solve projection which is the task of verifying whether α is entailed by regressing α and verifying that it is an entailment of the conjunction of what is true initially and the agent only knowing its initial beliefs.

Example 4.1.9. Suppose Υ and Σ are basic action theories from Example 4.1.3. Consider the projection query (3) from Figure 4.4 wrt $\Upsilon \wedge \boldsymbol{O}\Sigma$. We now verify this entailment by means of regression. Pursue as follows

$$\mathcal{R}[\Upsilon, \Sigma, \mathit{forward} \cdot \mathit{sonar}, \boldsymbol{K}(\mathit{distance} = 3)]$$

[12] Lakemeyer and Levesque [2004] consider a binary sensing function, which means that they only sense truth values. We generalize their account to sense arbitrary values based on [Scherl and Levesque, 2003].

$$= \mathcal{R}[\Upsilon, \Sigma, forward, \exists x\, distance = x \wedge \boldsymbol{K}(distance = x \supset [sonar]distance = 3)]$$
$$= \exists x.\, distance = x + 1 \wedge \mathcal{R}[\Upsilon, \Sigma, forward, \boldsymbol{K}(distance = x \supset [sonar]distance = 3)]$$
$$= \exists x.\, distance = x + 1 \wedge \boldsymbol{K}(distance = x + 1 \supset distance = 4).$$

The regressed query is easily shown to be entailed by $\Upsilon_0 \wedge \boldsymbol{O}\Sigma_0$, and so we are done. ∎

Readers will have noticed that we are restricting the regression operator to bounded basic sentences. There are at least two reasons for this limitation. First, note that the language is not expressive enough to refer to only knowing in non-initial situations. So if we begin by only knowing a basic action theory, one presumes that after an action the agent only knows another basic action theory. Regressing the latter should intuitively lead to a sentence that talks about what was only known before the action was executed, and this currently cannot be expressed in the language. Second, note that a basic action theory contains sentences such as the successor state axioms which are not bounded. So, if after an action we are left with a formula of the form $\boldsymbol{O}(\alpha)$, where α by the above argument would contain sentences that are not bounded, then this α would not be regressable. This is because Theorem 4.1.6 is limited to regressing bounded formulas.

Nevertheless, as we mentioned earlier, the regression operator covers the same class of formulas as considered by [Scherl and Levesque, 2003], and seems sufficient for most practical purposes. However, be that as it may, investigating what is only known after an action seems interesting in its own right. This is addressed in the next chapter.

4.1.4 Applying the Representation Theorem

If we restrict ourselves to static formulas that do not mention *Poss* or *SF* then we are essentially left with the language of $\mathcal{OL}$. For this fragment, Lakemeyer and Levesque show that $\mathcal{ES}$ and $\mathcal{OL}$ agree precisely in terms of valid sentences.

Theorem 4.1.10. *[Lakemeyer and Levesque, 2004]*
Suppose $\alpha \in \mathcal{OL}$. Then α is valid in $\mathcal{OL}$ iff α is valid in $\mathcal{ES}$.

Where this pays off is that results proved in $\mathcal{OL}$, such as the representation theorem (Section 3.1.3), can be imported when doing non-dynamic analysis in $\mathcal{ES}$. That is, by way of the regression property, we are able to reduce bounded basic sentences after actions to static basic fluent formulas about the initial theory, which clearly do not mention *Poss* and *SF*. Then, by way of the representation theorem, we can further restrict our attention to first-order reasoning. Formally, we can couple the representation theorem with the regression property as follows:

Theorem 4.1.11. *[Lakemeyer and Levesque, 2004]*
Given basic action theories Υ and Σ, and a basic bounded sentence α,

$$\Upsilon \wedge \boldsymbol{O}\Sigma \models \alpha \quad \textit{iff} \quad \models \Upsilon_0 \supset \|\mathcal{R}[\Upsilon, \Sigma, \langle\rangle, \alpha]\|_{\Sigma_0}.$$

Note that this is a well-defined statement only if $\mathcal{R}[\Upsilon, \Sigma, \langle\rangle, \alpha]$ is a sentence in $\mathcal{OL}$, which the following lemma establishes:

Lemma 4.1.12. *[Lakemeyer and Levesque, 2004]*
Suppose α is a bounded basic sentence. Then $\mathcal{R}[\Upsilon, \Sigma, \langle\rangle, \alpha]$ is a (basic) sentence in $\mathcal{OL}$.

Example 4.1.13. We illustrate the application of Theorem 4.1.11. Suppose Υ and Σ are basic action theories from Example 4.1.3. Consider the projection query (2) from Figure 4.4 wrt $\Upsilon \wedge \boldsymbol{O}\Sigma$. First pursue the regression of the query:

$$\mathcal{R}[\Upsilon, \Sigma, \mathit{forward}, \boldsymbol{K}(\mathit{distance} = 3 \vee \mathit{distance} = 4)]$$
$$= \boldsymbol{K}(\mathit{distance} = 4 \vee \mathit{distance} = 5).$$

Next, pursue resolving this epistemic query wrt the initial KB Σ_0

$$\|\boldsymbol{K}(\mathit{distance} = 4 \vee \mathit{distance} = 5)\|_{\Sigma_0}$$
$$= \textsc{Res}[\|\mathit{distance} = 4 \vee \mathit{distance} = 5\|_{\Sigma_0}, \Sigma_0] \quad \text{by (5) of Definition 3.1.10}$$
$$= \textsc{Res}[\Sigma_0, \Sigma_0] \quad \text{by (1) of Definition 3.1.10}$$
$$= \textsc{True} \quad \text{because } \models \Sigma_0 \supset \Sigma_0.$$

Since $\Upsilon_0 \models \|\boldsymbol{K}(\mathit{distance} = 4 \vee \mathit{distance} = 5)\|_{\Sigma_0}$, the query is indeed entailed by $\Upsilon \wedge \boldsymbol{O}\Sigma$. ∎

4.2 Multiagent Only Knowing in the Situation Calculus

We now propose an amalgamation of $\mathcal{OL}_n$ and the situation calculus. The idea will be to recast the semantical framework of multiagent only knowing against the action models of $\mathcal{ES}$. The resulting formalism, among other things, allows us to reason about *de re* and *de dicto* distinctions after actions in a multiagent setting. This is illustrated using a simple example involving two robots. In order to maintain the benefits of both the solution to the projection problem in $\mathcal{ES}$ as well as the reduction of basic queries, we will be proving some generalizations of those properties for the resulting multiagent formalism.

We first present a semantics and then turn to basic action theories. A number of conceptual differences arise in the many agent case. Earlier we accounted for incomplete knowledge by means of a basic action theory for what is true in the real world and another for what the agent believes to be true. But now, besides providing an account about the real world and the beliefs of A and B, one may need to also consider differences in the subsequent levels of belief, such as A's beliefs about B's knowledge of the world. We show that a regression property can indeed be obtained for certain epistemic states of that form, which allows us to reduce multiagent beliefs after actions to multiagent beliefs about the initial situation. In other words, we generalize Theorem 4.1.8 to the many agent case. Finally, we prove a representation theorem for $\mathcal{OL}_n$, which further allows us to reduce reasoning about multiagent beliefs about the initial situation to pure first-order reasoning in a manner similar to Theorem 4.1.11.

The Language

Following our convention for naming modal languages, we let $\mathcal{ES}_n$ be the non-modal fragment of $\mathcal{ES}$ enriched with modal operators $\boldsymbol{K}_i$ and $\boldsymbol{O}_i$, for $i \in \{A, B\}$. Terms and formulas are understood as in the previous

section, generalized to the many agent case in an obvious way. For example, if $\boldsymbol{M}\alpha$ is a well-formed formula of $\mathcal{ES}$ (where $\boldsymbol{M}$ is either $\boldsymbol{K}$ or $\boldsymbol{O}$) then $\boldsymbol{M}_i\alpha$ is a well-formed formula of $\mathcal{ES}_n$. Analogously, *objective*, *bounded*, *static* and *fluent* fluent formulas are defined.

The language also includes a distinguished fluent *Poss* that handles the conditions under which an action is executable, and sensing functions SF_i for i's sensing capabilities.

A Semantics

The semantical account is provided using the notion of k-structures. Following $\mathcal{OL}_n$, the satisfaction relation will be defined wrt the depth of formulas, where as before, our idea of depth will be to lump together consecutive nestings of epistemic operators for the same agent. Formally, the notion of i-depth gets extended for formulas in $\mathcal{ES}_n$ in the following way:

Definition 4.2.1. (i-depth.) The i-depth of a formula $\alpha \in \mathcal{ES}_n$, which is denoted $|\alpha|_i$, is defined inductively as:

1.-6. as in Definition 3.2.1;

7. $|[v]\alpha|_i = |\alpha|_i$;

8. $|\Box\alpha|_i = |\alpha|_i$.

A formula α has a depth k if $max(|\alpha|_A, |\alpha|_B) = k$. ∎

It is easy to see that the i-depth of a $\mathcal{ES}_n$ formula is simply the i-depth of the corresponding $\mathcal{OL}_n$ formula, obtained by ignoring all the mentioned action operators. To illustrate this, consider the following example in relation to Example 3.2.2.

Example 4.2.2. Consider the formula $\Box\boldsymbol{K}_A\boldsymbol{K}_B\boldsymbol{K}_A p \vee \boldsymbol{K}_B[t]q$. Here:

1. $|\Box\boldsymbol{K}_A\boldsymbol{K}_B\boldsymbol{K}_A p \vee \boldsymbol{K}_B[t]q|_A = max(|\Box\boldsymbol{K}_A\boldsymbol{K}_B\boldsymbol{K}_A p|_A, |\boldsymbol{K}_B[t]q|_A) = 3$ because
   (a) $|\Box\boldsymbol{K}_A\boldsymbol{K}_B\boldsymbol{K}_A p|_A = |\boldsymbol{K}_A\boldsymbol{K}_B\boldsymbol{K}_A p|_A = |\boldsymbol{K}_B\boldsymbol{K}_A p|_A = 1 + |\boldsymbol{K}_A p|_B = 2 + |p|_A = 3$,
   (b) $|\boldsymbol{K}_B[t]q|_A = 1 + |[t]q|_B = 1 + |q|_B = 2$.
2. $|\Box\boldsymbol{K}_A\boldsymbol{K}_B\boldsymbol{K}_A p \vee \boldsymbol{K}_B[t]q|_B = max(|\Box\boldsymbol{K}_A\boldsymbol{K}_B\boldsymbol{K}_A p|_B, |\boldsymbol{K}_B[t]q|_B) = 4$ because
   (a) $|\Box\boldsymbol{K}_A\boldsymbol{K}_B\boldsymbol{K}_A p|_B = |\boldsymbol{K}_A\boldsymbol{K}_B\boldsymbol{K}_A p|_B = 1 + |\boldsymbol{K}_B\boldsymbol{K}_A p|_A = 1 + 3$ (as shown above) $= 4$,
   (b) $|\boldsymbol{K}_B[t]q|_B = |[t]q|_B = |q|_B = 1$.
3. Therefore, the depth of the formula is 4.

The analysis is almost identical to Example 3.2.2, and the presence of action operators is just an extra step that does not complicate any calculation. ∎

Suppose now $\mathcal{W}$ is the set of *all possible worlds*.[13] The beliefs of an agent are captured by means of a k-structure (Definition 3.2.3) defined over the set $\mathcal{W}$. Such a structure essentially represents the initial beliefs of the agent, whose elements may be discarded over the course of executing actions. This is formalized by means of an equivalence relation $\simeq^i_z$ that looks for truth in the real world by means of sensing. We define $w' \simeq^i_z w$ inductively by the following:

- $w' \simeq^i_{\langle\rangle} w$ for all worlds w' and w;
- $w' \simeq^i_{z\cdot r} w$ iff $w' \simeq^i_z w$ and $w'[SF_i(r), z] = w[SF_i(r), z]$.

Now, by a (k, j)-model, we mean the tuple (e^k_A, e^j_B, w) where e^k_A is a k-structure for A, e^j_B is a j-structure for B and $w \in \mathcal{W}$ is a world. Only formulas of maximal A,B-depth of k, j are interpreted wrt (k, j)-models. To determine whether a formula of maximal A,B-depth of k, j is true or not after a sequence of actions z given a (k, j)-model, we write $e^k_A, e^j_B, w, z \models \alpha$. The complete definition is:

1. $e^k_A, e^j_B, w, z \models t_1 = t_2$ iff n_1 and n_2 are the same standard names, where $|t_i|^z_w = n_i$,
2. $e^k_A, e^j_B, w, z \models \neg\alpha$ iff $e^k_A, e^j_B, w \not\models \alpha$;
3. $e^k_A, e^j_B, w, z \models \alpha \vee \beta$ iff $e^k_A, e^j_B, w, z \models \alpha$ or $e^k_A, e^j_B, w, z \models \beta$;
4. $e^k_A, e^j_B, w, z \models \forall x\alpha$ iff $e^k_A, e^j_B, w, z \models \alpha^x_n$ for every name n of the appropriate sort;
5. $e^k_A, e^j_B, w, z \models [t]\alpha$ iff $e^k_A, e^j_B, w, z \cdot r \models \alpha$ where $|t|^z_w = r$;
6. $e^k_A, e^j_B, w, z \models \Box\alpha$ iff $e^k_A, e^j_B, w, z \cdot z' \models \alpha$ for every $z' \in \mathcal{Z}$;
7. $e^k_A, e^j_B, w, z \models \boldsymbol{K}_A\alpha$ iff for all $w' \simeq^A_z w$, for all e^{k-1} for B, if $(w', e^{k-1}_B) \in e^k_A$ then $e^k_A, e^{k-1}_B, w', z \models \alpha$;
8. $e^k_A, e^j_B, w, z \models \boldsymbol{O}_A\alpha$ iff for all $w' \simeq^A_z w$, for all e^{k-1} for B, $(w', e^{k-1}_B) \in e^k_A$ iff $e^k_A, e^{k-1}_B, w', z \models \alpha$.

In an analogous fashion, the semantics for $\boldsymbol{K}_B\alpha$ and $\boldsymbol{O}_B\alpha$ are specified.

Given a sentence of maximal A,B-depth k, j, we write $e^k_A, e^j_B, w \models \alpha$ to mean $e^k_A, e^j_B, w, \langle\rangle \models \alpha$. We say that a sentence α of maximal A,B-depth k, j is satisfiable if there is a (k, j)-model (e^k_A, e^j_B, w) such that $e^k_A, e^j_B, w \models \alpha$. If Σ is any set of sentences of maximal A,B-depth of k, j and α is as above, we write $\Sigma \models \alpha$ (read: "Σ entails α") iff for every (k, j)-model such that $e^k_A, e^j_B, w \models \alpha'$ for every $\alpha' \in \Sigma$ then $e^k_A, e^j_B, w \models \alpha$. We write $\models \alpha$ (read: "α is valid") to mean $\{\} \models \alpha$.

To assert that the validity of $\mathcal{ES}_n$ formulas is not affected when considering structures of a higher depth, as we have done so for $\mathcal{OL}_n$, we first establish the following lemma:

Lemma 4.2.3. *Let $k' \geq k$, $j' \geq j$. For all formulas $\alpha \in \mathcal{ES}_n$ of maximal A,B-depth k, j:*

$$e^k_A, e^j_B, w \models \alpha \qquad \textit{iff} \qquad e_A{\downarrow^{k'}_k}, e_B{\downarrow^{j'}_j}, w \models \alpha.$$

[13] Henceforth, unless specified otherwise, by worlds we will mean $\mathcal{ES}$-worlds, defined as in Section 4.1.

Proof: The proof is by induction on α. It follows the steps involved in proving Lemma 3.2.9. The only additional step in the base case are formulas of the form $[t]\alpha$. Note that for formulas of this form, $w \models [t]\alpha$ iff $w, r \models \alpha$ by definition where $|t|_w^{\langle\rangle} = r$. Since we have the same world in both models, the proof for the base case is immediate. ∎

From this, we get (see proof of Theorem 3.2.10):

Theorem 4.2.4. *For all $\alpha \in \mathcal{ES}_n$ of maximal A,B-depth of k, j, if α is true at all (k, j)-models then α is true at all (k', j')-models where $k' \geq k$, $j' \geq j$.*

As a closing remark to this section, let us note that the usual properties of knowledge regarding introspection and quantifying-in apply here as well (by analogy to $\mathcal{ES}$ and $\mathcal{OL}_n$):

Lemma 4.2.5. *The following are valid wrt models of appropriate depth:*

1. $\Box(\boldsymbol{K}_i\alpha \wedge \boldsymbol{K}_i(\alpha \supset \beta) \supset \boldsymbol{K}_i\beta)$,
2. $\Box(\boldsymbol{K}_i\alpha \supset \boldsymbol{K}_i\boldsymbol{K}_i\alpha)$,
3. $\Box(\neg\boldsymbol{K}_i\alpha \supset \boldsymbol{K}_i\neg\boldsymbol{K}_i\alpha)$,
4. $\Box(\forall\vec{x}\,\boldsymbol{K}_i\alpha \supset \boldsymbol{K}_i(\forall\vec{x}\,\alpha))$,
5. $\Box(\exists\vec{x}\,\boldsymbol{K}_i\alpha \supset \boldsymbol{K}_i(\exists\vec{x}\,\alpha))$.

Proof: The proofs are very similar and so, we only show item 3. Let i be A. Suppose $e_A^k, e_B^j, w, z \models \neg\boldsymbol{K}_A\alpha$. Then there is some $(w', e_B^{k-1}) \in e_A^k$ (where $w' \simeq_z^A w$) such that $e_A^k, e_B^{k-1}, w', z \models \neg\alpha$. Suppose $w'' \simeq_z^A w$ is any world and $(w'', {e'}_B^{k-1}) \in e_A^k$. Then clearly $e_A^k, {e'}_B^{k-1}, w'' \models \neg\boldsymbol{K}_A\alpha$. Therefore $e_A^k, e_B^j, w \models \boldsymbol{K}_A\neg\boldsymbol{K}_A\alpha$. ∎

4.2.1 Basic Action Theories and Projection

In the many agent case the notion of a basic action theory is essentially identical to what one defines in $\mathcal{ES}$. Except, one has to specify the sensing axioms for each agent that may differ. For example, when B senses that A is reading a letter, we would not expect B to learn the contents of that letter.

Definition 4.2.6. (Basic action theory.) Given a set of fluents $\mathcal{F}$, a set $\Sigma \subseteq \mathcal{ES}_n$ is called a basic action theory over $\mathcal{F}$ iff $\Sigma = \Sigma_0 \cup \Sigma_{pre} \cup \Sigma_{post} \cup \Sigma_{sense}$ where Σ only mentions fluents from $\mathcal{F}$ and

1. Σ_0, Σ_{pre} and Σ_{post} are specified as in Definition 4.1.2;
2. Σ_{sense} is a set of sentences of the form $\Box SF_i(v) = x \equiv \varphi_i$, one for each agent i and where φ_i is a fluent formula. ∎

Modeling Incomplete Information

When dealing with initial knowledge in the multiagent case, we have to distinguish between what is true in the real world and what the agents know or believe about the world. Of course, what A believes about the world may differ from B's knowledge. Moreover, what A believes B to know may differ from what B actually believes. Perhaps the simplest way to capture such generality is to have multiple basic action theories for subsequent levels of beliefs, as illustrated by (say) the following background theory:[14]

$$\Upsilon \wedge \boldsymbol{O}_A(\Sigma \wedge \boldsymbol{O}_B\Sigma^*) \wedge \boldsymbol{O}_B(\Sigma' \wedge \boldsymbol{O}_A\Sigma^{**}) \tag{4.1}$$

where Υ and Σ (with superscripts) are basic action theories that may differ arbitrarily. Here, Υ represents what is true in the real world, and Σ (with superscripts) represent the agent's beliefs. For example, Σ^* represents what A believes B to know.

More often than not, however, we imagine that if i has certain beliefs about the world, then she at least considers other agents to hold similar beliefs. Think of having agents play *a game with imperfect information*, such as Poker [Osborne and Rubinstein, 1994; Belle and Lakemeyer, 2010b]. It is safe to assume that, in a fair situation, all that i knows initially are the rules of the game. Reasonably enough, i also expects that all that j knows initially are these rules. Similarly, think of having two robots coordinate among themselves to deliver a heavy package. We imagine that both i and j are given an initial specification, which determines (say) where the package is located and is to be delivered. In this domain, it is reasonable to assume that both agents consider that the other robot also has access to the same domain knowledge. In sum, the assumption is that if i believes a basic action theory Σ, then i also believes that j believes Σ. A background theory, then, is a special case of (4.1), as illustrated by the following sentence:

$$\Upsilon \wedge \boldsymbol{O}_A(\Sigma \wedge \boldsymbol{O}_B\Sigma) \wedge \boldsymbol{O}_B(\Sigma' \wedge \boldsymbol{O}_A\Sigma') \tag{4.2}$$

where Υ, Σ and Σ' may differ arbitrarily. For ease of exposition, we prove properties for background theories of this form. Extensions to more general case of (4.1) is straightforward but tedious.

Be that as it may, notice a crucial stipulation made in (4.1) and (4.2). We are assuming that initial knowledge can be represented by sentences of the form $\boldsymbol{O}_i(\phi \wedge \boldsymbol{O}_j\psi)$. In a sense, we are generalizing the assumption we made about background theories in the single agent case, where we stipulated that the basic action theory was all that was known. There are other possibilities, of course. Examples include

- $\boldsymbol{O}_i(\phi \wedge (\boldsymbol{K}_j\psi \vee \boldsymbol{K}_j\psi'))$, where all that i knows besides ϕ is that j believes ψ or ψ';
- $\boldsymbol{O}_i(\phi \vee \boldsymbol{K}_j\psi)$, where all that i knows is that ϕ may be true or that j believes ψ.

While perhaps some of the results proved in the remainder of the chapter could be adapted to cover cases such as these, we believe that it would be at the cost of a considerably more complex formal theory. Moreover, initial knowledge of the form $\boldsymbol{O}_i(\phi \wedge \boldsymbol{O}_j\psi)$ is both reasonable and natural in many applications, as in the cases of the game and delivery domains taken up above. Of course, as actions occur, one imagines that the beliefs of the agents diverge, but at least initially we can imagine the agents starting off with the very same set of facts. Under this premise, we settle for a stipulation that takes the form of (4.2).

[14] For the sake of the following discussion, assume agents have beliefs of two levels.

The Problem of Projection

In what follows, we will prove properties for sentences such as (4.2). In order to prepare for agents that may have beliefs to some arbitrary (but finite) depth, we introduce the following inductive definition over a basic action theory Σ:

- let $OKnow_\Sigma[A, 1] = \boldsymbol{O}_A\Sigma$;
- let $OKnow_\Sigma[B, 1] = \boldsymbol{O}_B\Sigma$;
- for $k > 1$, let $OKnow_\Sigma[A, k] = \boldsymbol{O}_A(\Sigma \wedge OKnow_\Sigma[B, k-1])$;
- for $j > 1$, let $OKnow_\Sigma[B, j] = \boldsymbol{O}_B(\Sigma \wedge OKnow_\Sigma[A, j-1])$.

Given basic action theories Υ, Σ and Σ', in the remainder of the chapter we will be interested in theories of the form

$$\Upsilon \wedge OKnow_\Sigma[A, k] \wedge OKnow_{\Sigma'}[B, j] \tag{4.3}$$

which says that A believes the basic action theory Σ to k levels, *i.e.* he believes B to also believe Σ and so on, while B believes the basic action theory Σ' to j levels. The problem of *projection*, then, is that of effectively reasoning about entailments about (4.3). That is, we are interested in determining what follows after a number of actions have occurred:

$$(4.3) \models [r_1] \ldots [r_k]\alpha$$

where $r_1, \ldots, r_k$ are primitive actions and α is an arbitrary formula.

We now illustrate a basic action theory and projection queries in the multiagent case. We follow the convention that all actions, including sensing actions, are *publicly observable* while the information obtained via sensing is *private* [Kelly and Pearce, 2008]. For the kind of applications we have in mind, this assumption seems reasonable. We will shortly see examples that demonstrates this asymmetry in the information as actions occur.

Example 4.2.7. (The cooperating robots domain.) Imagine two simple robots, say A and B, coordinating among themselves to deliver a heavy package, say the block C, as illustrated in Figure 4.5. The destination of the delivery is also provided, which A reads first, and then B does the same. We assume that the two robots should start moving *only after* they believe that the other agent also knows the location. However, since the specification of the agent programs is orthogonal to the analysis of the entailments of a background theory, we do not go over the actual nature of the coordination protocol for the sake of simplicity.

Let us suppose that the fluent $goal(x)$ specifies the delivery location of package x. The agent i senses the location specified for x by means of the action $seegoal_i(x)$. Informally, sensing should work as follows. When $seegoal_A(x)$ is executed, A learns the location of x. In the meantime, since we assume that all actions are public, B also observes that $seegoal_A(x)$ is executed. However, B does not learn anything of interest from that observation. Symmetric arguments hold wrt B's sensing action $seegoal_B(x)$.

Let us now make this precise by formalizing the sensing axioms Σ_{sense} in Figure 4.6. There we stipulate that when B senses on $seegoal_A(x)$, irrespective of which x, the standard name 1 is always returned. This is another way of saying that B does not obtain any useful information when he senses on $seegoal_A(x)$.

For simplicity's sake, physical and other sensing actions that may be necessary for a coordinated delivery are ignored in the example. Now, assuming that block C is to be delivered to *roomC*, let

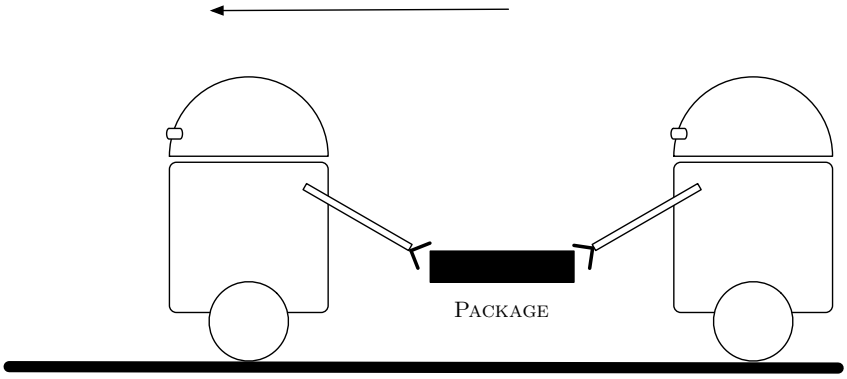

Figure 4.5: Two simple robots coordinating for a delivery.

$$\Sigma_{sense} = \{\Box SF_A(seegoal_A(x)) = y \equiv goal(x) = y$$

$$\Box SF_B(seegoal_A(x)) = y \equiv y = 1,$$

$$\Box SF_A(seegoal_B(x)) = y \equiv 1,$$

$$\Box SF_B(seegoal_B(x)) = y \equiv goal(x) = y\}.$$

$\Sigma_0 = \{\text{TRUE}\}$.

$\Upsilon_0 = \Sigma_0 \cup \{goal(C) = roomC\}$.

Figure 4.6: A basic action theory for two cooperating robots.

- $\Sigma = \Sigma_{sense} \wedge \Sigma_{pre} \wedge \Sigma_{post} \wedge \Sigma_0$ be the basic action theory that A and B believe to levels k and j respectively, and
- $\Upsilon = \Sigma \cup \Upsilon_0$ be what is true in the real world.

This leads to the following background theory:

$$\Upsilon \wedge OKnow_\Sigma[A, k] \wedge OKnow_\Sigma[B, j] \tag{4.4}$$

Before analyzing the entailments of (4.4), it is convenient to state a lemma regarding how a model of (4.4) can be constructed. Let us denote the set of worlds $\{w \mid w \models \Sigma\}$ as $\mathcal{W}_\Sigma$. Further, let ${e_\Sigma}^1 = \mathcal{W}_\Sigma \times \{\{\}\}$. Let ${e_\Sigma}^k = \{(w, e_\Sigma^{k-1}) \mid w \in \mathcal{W}_\Sigma\}$ be defined inductively. Then,

Lemma 4.2.8. *Suppose w is any world such that $w \models \Upsilon$, $e_{\Sigma_A}^k$ and $e_{\Sigma_B}^j$ are constructed as above. Then $e_{\Sigma_A}^k, e_{\Sigma_B}^j, w \models$ (4.4).*

Proof: Since $w \models \Upsilon$ by assumption, we can restrict ourselves to the subjective formulas in (4.4). Moreover, since $OKnow_\Sigma[i, *]$ is interpreted wrt i's epistemic state, we will prove the lemma by doing a simple induction on the *modal depth* of the background theory, where the modal depth of a formula α, denoted $modal(\alpha)$, is defined inductively:

- $modal(\alpha) = 0$ for primitive atoms;
- $modal(\alpha \vee \beta) = max(modal(\alpha), modal(\beta))$;
- $modal(\neg\alpha) = modal(\alpha)$;
- $modal(\boldsymbol{M}_i\alpha) = 1 + modal(\alpha)$ where $\boldsymbol{M}_i \in \{\boldsymbol{K}_i, \boldsymbol{O}_i\}$.

That is, when the modal depth of the background theory is l, then we have a sentence of the form $OKnow_\Sigma[A, k] \wedge OKnow_\Sigma[B, j]$ such that $k \leq l$, $j \leq l$ and k or j is l.

The base case is for theories of modal depth 1, where we are considering background theories of the form $\boldsymbol{O}_A\Sigma \wedge \boldsymbol{O}_B\Sigma$ (or $\boldsymbol{O}_A\Sigma$ or $\boldsymbol{O}_B\Sigma$.) To prove the base case, consider any world w'. Clearly $w' \simeq_{\langle\rangle} w$ by definition. By construction, $(w', \{\}) \in e^1_{\Sigma_A}$ iff $w' \models \Sigma$. Therefore $e^1_{\Sigma_A}, \{\}, w \models \boldsymbol{O}_A(\Sigma)$. Analogously for $e^1_{\Sigma_B}$.

Suppose that the lemma holds for background theories of modal depth $k-1$, that is, $e^{k-1}_{\Sigma_A}$ satisfies $OKnow_\Sigma[A, k-1]$. (This is analogously stated for B.) Let $(w', e^{k-1}_{\Sigma_B})$ be any k-structure in $e^k_{\Sigma_A}$. By construction $w' \models \Sigma$. By induction hypothesis, $\{\}, e^{k-1}_{\Sigma_B}, w' \models OKnow_\Sigma[B, k-1]$. That is, by construction, $(w', e^{k-1}_{\Sigma_B}) \in e^k_{\Sigma_A}$ iff $\{\}, e^{k-1}_{\Sigma_B}, w' \models \Sigma \wedge OKnow_\Sigma[B, k-1]$. Therefore $e^k_{\Sigma_A}, \{\}, w \models \boldsymbol{O}_A(\Sigma \wedge OKnow_\Sigma[B, k-1])$, *i.e.* $e^k_{\Sigma_A}, \{\}, w \models OKnow_\Sigma[A, k]$. ∎

Proposition 4.2.9. *Let M be any model of* (4.4)*. Then, the following holds:*

1. $M \models \neg\boldsymbol{K}_A(goal(C) = roomC)$.

   Initially A does not know the location.

2. $M \models [seegoal_A(C)]\boldsymbol{K}_A(goal(C) = roomC)$.

   After sensing, A knows where the package is to be delivered.

3. $M \not\models [seegoal_A(C)]\exists x\boldsymbol{K}_B(goal(C) = x)$.

   But it is not the case that B knows the location when he observes A sensing. That is, B does not have de re knowledge about the location.

4. $M \models [seegoal_A(C)]\boldsymbol{K}_A\neg\exists x\boldsymbol{K}_B(goal(C) = x)$.

   Moreover, A knows that B does not know the location as of now.

5. $M \models [seegoal_A(C)]\boldsymbol{K}_B\exists x\boldsymbol{K}_A(goal(C) = x)$.

   Nevertheless, B knows that A knows the location. That is, B has de dicto knowledge about A's beliefs.

6. $M \models [seegoal_A(C)][seegoal_B(C)]\boldsymbol{G}(goal(C) = roomC)$, *where* $\boldsymbol{G} \in \{\boldsymbol{K}_A\boldsymbol{K}_B, \boldsymbol{K}_B\boldsymbol{K}_A\}$.

   After both A and B do the sensing, each robot knows that the other knows where the package is to be delivered.

Proof: Let $M = (e_A^k, e_B^j, w)$ be a model of (4.4). Below, we abbreviate $goal(C) = roomC$ as α, $seegoal_A(C)$ as r and $seegoal_B(C)$ as r'.

1. Assume the contrary. Suppose that $e_A^k, e_B^j, w \models \boldsymbol{K}_A(\alpha)$. Then for all $(w', e_B^{k-1}) \in e_A^k$, $e_A^k, e_B^{k-1}, w' \models \alpha$.

   Now, note that Σ_0 leaves the value of $goal(C)$ unspecified. Thus, by construction (and definition of $\mathcal{W}$), there are worlds $w'' \in \mathcal{W}_\Sigma$ such that $w'' \simeq^A_{\langle\rangle} w$, $w'' \models goal(C) \neq roomC$ and where $(w'', e_B^{k-1}) \in e_A^k$. This is a contradiction.

2. After A executes r, it follows that only those worlds $w' \in \mathcal{W}_\Sigma$ such that $w'[SF_A(r), \langle\rangle] = w[SF_A(r), \langle\rangle] = roomC$ are considered when evaluating A-subjective formulas. (These are worlds that agree on the delivery location with the real world.) Therefore $e_A^k, e_B^j, w, r \models \boldsymbol{K}_A(\alpha)$ since for every $(w', e_B^{k-1}) \in e_A^k$ such that $w' \simeq^A_r w$, $e_A^k, e_B^{k-1}, w', r \models \alpha$ by the definition of the sensing axioms Σ_{sense}.

3. Observe that for every $w' \in \mathcal{W}_\Sigma$, $w'[SF_B(r), \langle\rangle] = 1 = w[SF_B(r), \langle\rangle]$. So, while evaluating B-subjective formulas every $(w', e_A^{j-1}) \in e_B^j$ is considered, including ones where $\neg\alpha$ holds at w'. Therefore $e_A^k, e_B^j, w, r \not\models \boldsymbol{K}_B(\alpha)$.

4. Consider any $(w', e_B^{k-1}) \in e_A^k$ such that $w' \simeq^A_r w$. By the same arguments from item 3, it follows that

$$e_A^k, e_B^{k-1}, w', r \not\models \exists x \boldsymbol{K}_B(goal(C) = x).$$

   Therefore, $e_A^k, e_B^j, w, r \models \boldsymbol{K}_A\neg\exists x \boldsymbol{K}_B(goal(C) = x)$.

5. Consider any $(w', e_A^{j-1}) \in e_B^j$. By the same arguments from item 2, it follows that

$$e_A^{j-1}, e_B^j, w', r \models \exists x \boldsymbol{K}_A(goal(C) = x).$$

   That is, if $w'[SF_A(r), \langle\rangle] = n$ then $e_A^{j-1}, e_B^j, w', r \models \boldsymbol{K}_A(goal(C) = n)$. Therefore $e_A^k, e_B^j, w, r \models \boldsymbol{K}_B\exists x \boldsymbol{K}_A(goal(C) = x)$.

   To see what is happening here, suppose that B only considered j-structures (w, e_A^{j-1}) possible, where w is the real world. Then he would know what A knows about the location. But since his epistemic state is $\{(w', e_A^{j-1}), (w'', e_A^{j-1}), \ldots\}$ he believes at each of the worlds w' that A knows the location as well as what this is, but he does not know of which of these is the real world.

6. We consider the case $\boldsymbol{G} = \boldsymbol{K}_A\boldsymbol{K}_B$. The other case is symmetric.

   Assume the contrary. Then $e_A^k, e_B^j, w, r \not\models [r']\boldsymbol{K}_A\boldsymbol{K}_B\alpha$. This implies that there is some $(w', e_B^{k-1}) \in e_A^k$ such that $w' \simeq^A_{r'.r} w$, *i.e.* such that $w' \models \alpha$, and $e_A^k, e_B^{k-1}, w' \not\models \boldsymbol{K}_B\alpha$. This then implies that there is some $(w'', e_A^{k-2}) \in e_B^{k-1}$ such that $w'' \simeq^B_{r'.r} w'$, and $e_A^{k-2}, e_B^{k-1}, w'' \models \neg\alpha$. That is, $w'' \models \neg\alpha$. However, $w'' \simeq^B_{r'.r} w'$ means that $w''[SF_B(r'), r] = w'[SF_B(r'), r] = roomC$, *i.e.* $w'' \models \alpha$, which is a contradiction. ∎

4.2.2 Regression

We now consider a solution to the projection problem in $\mathcal{ES}_n$ by means of regression. By analogy to the single agent case, the class of *regressable* formulas in $\mathcal{ES}_n$ consists of all bounded basic formulas. Naturally, the regression of bounded objective formulas receives an identical treatment as before, formalized in terms of Theorem 4.1.6. To handle the subjective case, let us consider the generalization of the successor state axiom for knowledge to the multiagent case. We remark, again, that this is a theorem of the logic and not a stipulation of the basic action theory.

Theorem 4.2.10. (Successor State Axiom for Knowledge.)

$$\models \Box[v]\boldsymbol{K}_i(\alpha) \equiv$$

$$\exists x.\, SF_i(v) = x \wedge \boldsymbol{K}_i(SF_i(v) = x \supset [v]\alpha).$$

Proof: Let i be A, with the other case being symmetric. For the only-if direction, suppose that $e_A^k, e_B^j, w, z \models [r]\boldsymbol{K}_A\alpha_r^v$ for an action name $r \in \mathcal{A}$. Abbreviate α_r^v as α'. Suppose that $e_A^k, e_B^j, w, z \models SF_A(r) = n$. It then suffices to show that $e_A^k, e_B^j, w, z \models \boldsymbol{K}_A(SF_A(r) = n \supset [r]\alpha')$.

So suppose $(w', e_B^{k-1}) \in e_A^k$ and $w'[SF_A(r), z] = n$. Since $w' \simeq_{z\cdot r}^A w$, it follows by assumption that $e_A^k, e_B^{k-1}, w', z \cdot r \models \alpha'$, *i.e.* $e_A^k, e_B^{k-1}, w', z \models [r]\alpha'$. Thus $e_A^k, e_B^{k-1}, w', z \models SF_A(r) = n \supset [r]\alpha'$, and it follows then that $e_A^k, e_B^j, w', z \models \boldsymbol{K}_A(SF_A(r) = n \supset [r]\alpha')$.

Conversely, suppose that $e_A^k, e_B^j, w, z \models SF_A(r) = n \wedge [r]\boldsymbol{K}_A(SF_A(r) = n \supset \alpha')$. We now need to show that $e_A^k, e_B^j, w, z \models \boldsymbol{K}_A([r]\alpha')$, *i.e.* for all $(w', e_B^{k-1}) \in e_A^k$ such that $w' \simeq_z^A w$, $e_A^k, e_B^{k-1}, w', z \cdot r \models \alpha'$.

Suppose $w' \simeq_{z\cdot r}^A w$, *i.e.* $w'[SF_A(r), z] = n$ and $(w', e_B^{k-1}) \in e_A^k$ for some e_B^{k-1}. Then by assumption, $e_A^k, e_B^{k-1}, w', z \cdot r \models \alpha'$. Therefore $e_A^k, e_B^{k-1}, w', z \models [r]\alpha$, from which it follows that $e_A^k, e_B^j, w, z \models \boldsymbol{K}_A([r]\alpha')$. ∎

Observe that this generalizes Theorem 4.1.7 in an obvious way. It roughly says that what i knows after an action depends on what was known before and what the future would look like contingent on i's sensing result. With this in hand, we are now ready to generalize regression to multiple agents.

We define a regression operator $\mathcal{R}[\Upsilon, \Sigma, \Sigma', \sigma, \alpha]$ wrt a basic action theory Υ for what is true in the real world, a basic action theory Σ for what A believes at all levels, and a basic action theory Σ' for what B believes at all levels, as considered in (4.3).

Definition 4.2.11. (Regression.) We define $\mathcal{R}[\Upsilon, \Sigma, \Sigma', \alpha]$, the *regression* of α wrt Υ, Σ and Σ', to be $\mathcal{R}[\Upsilon, \Sigma, \Sigma', \langle\rangle, \alpha]$. For a given sequence of action names or variables σ, we define $\mathcal{R}[\Upsilon, \Sigma, \Sigma', \sigma, \alpha]$ inductively by:

1.-7. See Definition 4.1.5. (Note that this definition uses the *rhs* of the precondition, successor state axiom, and sense axioms from Υ.)

8. $\mathcal{R}[\Upsilon, \Sigma, \Sigma', z, SF_i(t) = t'] = \mathcal{R}[\Upsilon, \Sigma, \Sigma', z, \varphi_{i_t\ t'}^{v\ x}]$;

9. $\mathcal{R}[\Upsilon, \Sigma, \Sigma', z, \boldsymbol{K}_A\alpha]$ is defined inductively on z by:

   (a) $\mathcal{R}[\Upsilon, \Sigma, \Sigma', \langle\rangle, \boldsymbol{K}_A\alpha] = \boldsymbol{K}_A(\mathcal{R}[\Sigma, \Sigma, \Sigma, \langle\rangle, \alpha])$;

(b) $\mathcal{R}[\Upsilon, \Sigma, \Sigma', z \cdot r, \boldsymbol{K}_A\alpha] = \mathcal{R}[\Upsilon, \Sigma, \Sigma', z, \beta_r^v]$, where β is *rhs* of the equivalence in Theorem 4.2.10 for the agent index A.

10. $\mathcal{R}[\Upsilon, \Sigma, \Sigma', z, \boldsymbol{K}_B\alpha]$ is defined inductively on z by:

(a) $\mathcal{R}[\Upsilon, \Sigma, \Sigma', \langle\rangle, \boldsymbol{K}_B\alpha] = \boldsymbol{K}_B(\mathcal{R}[\Sigma', \Sigma', \Sigma', \langle\rangle, \alpha])$;

(b) $\mathcal{R}[\Upsilon, \Sigma, \Sigma', z \cdot r, \boldsymbol{K}_B\alpha] = \mathcal{R}[\Upsilon, \Sigma, \Sigma', z, \beta_r^v]$, where β is *rhs* of the equivalence in Theorem 4.2.10 for the agent index B. ∎

The regression operator in the multiagent case works as follows. At the initial situation, regressing $\boldsymbol{K}_A\alpha$ is equivalent to regressing α wrt the basic action theory Σ that A believes at all levels. Similarly, at the initial situation, regressing $\boldsymbol{K}_B\alpha$ is equivalent to regressing α wrt the basic action theory Σ' that B believes at all levels. More generally, if we are regressing $\boldsymbol{K}_i\alpha$ wrt an action sequence $z \cdot r$, then this is equivalent to regressing the *rhs* of Theorem 4.2.10 wrt z by first substituting the ground action r. Readers may observe that this is analogous to the single agent case with the exception that depending on the index of the epistemic operator, we regress the remainder formula wrt the basic action theory that the corresponding agent believes.

For simplicity, we often write $\mathcal{R}[z, \alpha]$ instead of $\mathcal{R}[\Upsilon, \Sigma, \Sigma', z, \alpha]$. We are now ready to prove the main result of this section:

Theorem 4.2.12. *Suppose α is a bounded basic sentence of maximal A, B-depth k, j. Let Υ, Σ and Σ' be BATs. Then $\mathcal{R}[\langle\rangle, \alpha]$ is a static sentence and satisfies:*

$$\Upsilon \wedge \psi \models \alpha \quad \textit{iff} \quad \Upsilon_0 \wedge \psi_0 \models \mathcal{R}[\langle\rangle, \alpha]$$

where $\psi = OKnow_{\Sigma}[A, k] \wedge OKnow_{\Sigma'}[B, j]$

$\psi_0 = OKnow_{\Sigma_0}[A, k] \wedge OKnow_{\Sigma_0'}[B, j]$.

That is, we solve projection which is the task of verifying whether α is entailed by regressing α and verifying that is an entailment of the conjunction of what is true initially and each agent only knowing their initial beliefs. The proof for this theorem is provided in the appendix.

Readers will have noticed that the theorem assumes a background theory where A has beliefs to level k and B has beliefs to level j, given a query whose maximal A, B-depth is k, j. This syntactic restriction is essential for our relatively simple regression operator to be well-defined. To see that, suppose we are interested in verifying whether $\boldsymbol{K}_A\boldsymbol{K}_B[r]\alpha$ is entailed by $\boldsymbol{O}_A(\Sigma)$, where Σ is a basic action theory. By the definition of the regression operator given above, evaluating the query reduces to regressing $[r]\alpha$ wrt Σ, but this is not a correct transformation because A does not have any beliefs about B's knowledge of the world. In fact, the formula $\boldsymbol{K}_A\boldsymbol{K}_B[r]\alpha$ does not seem amenable to regression wrt $\boldsymbol{O}_A(\Sigma)$ since it is simply not clear how one should regress the subformula $\boldsymbol{K}_B[r]\alpha$. But now note that the formula $\boldsymbol{K}_A\boldsymbol{K}_B[r]\alpha$ is of depth 2 and that the transformation is indeed correct wrt initial knowledge for A of at least depth 2, such as $\boldsymbol{O}_A(\Sigma \wedge \boldsymbol{O}_B\Sigma)$.

Example 4.2.13. We illustrate regression by means of Example 4.2.7. Let Υ and Σ be basic action theories from Example 4.2.7. Pursue the sample projection query from Figure 4.7 wrt the background theory $\Sigma \wedge \boldsymbol{O}_A\Sigma \wedge \boldsymbol{O}_B\Sigma \wedge goal(C) = roomC$ via regression. We use Γ as an abbreviation for the background theory, α abbreviates $goal(C) = roomC$ and r abbreviates $seegoal_A(C)$.

Regressing the first conjunct in the query:

Let α denote $goal(C) = roomC$ and r denote $seegoal_A(C)$.

- Background theory Γ: $\Sigma \wedge \alpha \wedge \boldsymbol{O}_A\Sigma \wedge \boldsymbol{O}_B\Sigma$.
- Sample query: $[r]\boldsymbol{K}_A\alpha \wedge [r]\neg\boldsymbol{K}_B\alpha$.
- Regressed query: $\exists x.goal(C) = x \wedge \boldsymbol{K}_A(goal(C) = x \supset \alpha) \wedge \neg\boldsymbol{K}_B\alpha$.

Figure 4.7: Sample projection query and regression in $\mathcal{ES}_n$.

$$\mathcal{R}[\langle\rangle, \exists x.SF_A(r) = x \wedge \boldsymbol{K}_A(SF_A(r) = x \supset [r]\alpha)]$$

$$= \exists x.goal(C) = x \wedge \mathcal{R}[\langle\rangle, \boldsymbol{K}_A(SF_A(r) = x \supset [r]\alpha)]$$

$$= \exists x.goal(C) = x \wedge \boldsymbol{K}_A(goal(C) = x \supset \alpha).$$

In an analogous manner, regress the second conjunct

$$\neg\mathcal{R}[r, \boldsymbol{K}_B\alpha]$$

$$= \neg(\exists y.\ y = 1 \wedge \boldsymbol{K}_B(y = 1 \supset \alpha)), \text{ which is equivalent to } \neg\boldsymbol{K}_B\alpha.$$

The regressed query $\exists x.goal(C) = x \wedge \boldsymbol{K}_A(goal(C) = x \supset \alpha) \wedge \neg\boldsymbol{K}_B\alpha$ is easily shown to be a consequence of $\alpha \wedge \Sigma_0 \wedge \boldsymbol{O}_A\Sigma_0 \wedge \boldsymbol{O}_B\Sigma_0$. Therefore, we are allowed to conclude that the projection query is entailed as well. ∎

4.3 A Representation Theorem

In this section, we complement the regression theorem by generalizing the representation theorem to the multiagent case which reduces the evaluation of a basic query in $\mathcal{OL}_n$ to first-order reasoning.

Recall that the representation theorem in $\mathcal{OL}$ has two fundamental parts. The first is the operator $\textsc{Res}[\alpha, \phi]$ formulated in Definition 3.1.9 where α is an objective formula perhaps with free variables and ϕ is an objective sentence. The RES operator resolves to a formula not mentioning any function symbols such that on substituting its free variables with names, it is either valid or its negation is valid. Second, one defines a reduction operator $\|\alpha\|_\phi$, where $\alpha \in \mathcal{OL}$ is any basic sentence and ϕ is an objective sentence that is taken to be all that the agent knows, which transforms α to an objective formula and thereby appealing to RES to answer the query. When extending the result to the multiagent case, the RES operator is not modified since it is about asking an objective query to an objective KB. However, the reduction operator which now has to somehow consider the sentences that A and B believe to the appropriate depths is modified as follows.

Definition 4.3.1. Let ϕ and ϕ' be objective sentences, and α and β be basic $\mathcal{OL}_n$ sentences. Then we define the objective sentence $\|\alpha\|_{\phi,\phi'}$ by:

1. $\|\alpha\|_{\phi,\phi'} = \alpha$ if α is objective;

2. $\|\neg\alpha\|_{\phi,\phi'} = \neg\|\alpha\|_{\phi,\phi'}$;
3. $\|\alpha \vee \beta\|_{\phi,\phi'} = \|\alpha\|_{\phi,\phi'} \vee \|\beta\|_{\phi,\phi'}$;
4. $\|\forall x\alpha\|_{\phi,\phi'} = \forall x\|\alpha\|_{\phi,\phi'}$;
5. $\|\boldsymbol{K}_A\alpha\|_{\phi,\phi'} = \textsc{Res}[\|\alpha\|_{\phi,\phi}, \phi]$;
6. $\|\boldsymbol{K}_B\alpha\|_{\phi,\phi'} = \textsc{Res}[\|\alpha\|_{\phi',\phi'}, \phi']$. ∎

Intuitively, given an objective KB ϕ that A believes at all levels and an objective KB ϕ' that B believes at all levels, a conceptually simple reduction operator can be obtained. The reader may notice some similarity to the regression operator, *viz.* whenever $\boldsymbol{K}_A\alpha$ is encountered then the reduction is continued wrt the KB ϕ. Analogously, the reduction is continued wrt ϕ' whenever $\boldsymbol{K}_B\alpha$ is encountered.

By analogy to Theorem 4.1.11, the representation theorem is coupled with the regression operator as follows.

Theorem 4.3.2. *Let Υ, Σ and Σ' be basic action theories. Suppose α is a basic bounded sentence of maximal A,B-depth k, j, then*

$$\Upsilon \wedge \psi \models \alpha \quad \textit{iff} \quad \models \Upsilon_0 \supset \|\mathcal{R}[\langle\rangle, \alpha]\|_{\Sigma_0,\Sigma_0'}.$$

where $\psi = OKnow_{\Sigma}[A,k] \wedge OKnow_{\Sigma'}[B,j]$.

That is, a query α perhaps with action operators is entailed by the background theory iff the regressed query reduced by the representation theorem wrt Σ_0 and $\Sigma_0{}'$ is entailed by the set of sentences that are true initially. Thus, no modal reasoning is necessary.

The following example illustrates the representation theorem in action. We reconsider Example 4.2.13 where regression was applied to reduce a basic query after actions to a static basic query. Below, we resolve this static query containing epistemic operators.

Example 4.3.3. Let Υ and Σ be basic action theories from Example 4.2.7. Consider the background theory from Figure 4.7. We let Γ abbreviate $\Sigma \wedge goal(C) = roomC \wedge \boldsymbol{O}_A\Sigma \wedge \boldsymbol{O}_B\Sigma$, α abbreviates $goal(C) = roomC$ and r abbreviates $seegoal_A(C)$. For this example, we will verify that

$$\Gamma \models [r]\neg\boldsymbol{K}_B\alpha.$$

We first pursue the regression of the query. We know from Figure 4.7 that this is equivalent to $\neg\boldsymbol{K}_B\alpha$. We now resolve the regressed query by means of the representation theorem. That is:

$$\begin{aligned}
&\|\neg\boldsymbol{K}_B\alpha\|_{\Sigma_0,\Sigma_0} \\
&\quad = \neg\textsc{Res}[\|\alpha\|_{\Sigma_0,\Sigma_0}, \Sigma_0] && \text{by (2) and (6) of Definition 4.3.1} \\
&\quad = \neg\textsc{Res}[\alpha, \Sigma_0] && \text{by (1) of Definition 4.3.1} \\
&\quad = \neg(\textsc{False}) && \text{because } \Sigma_0 \not\models \alpha \\
&\quad = \textsc{True}.
\end{aligned}$$

Since, trivially, $\Upsilon_0 \models \textsc{True}$, Theorem 4.3.2 proves that the projection query is indeed entailed by the background theory. ∎

4.4 Concluding Remarks

This chapter proposes a solution to the projection problem for arbitrary BATs in multiagent knowledge bases with incomplete information by extending the notion of regression to the new framework. Moreover, by generalizing the representation theorem, we have shown that ordinary first-order reasoning is all that is required when reasoning about action.

In comparison to the early work of Reiter [1991], $\mathcal{ES}$ lifts the benefits of regression to the case of epistemic queries involving perhaps introspection and quantifying-in. However, while the representation theorem does reduce the problem to a first-order reasoning task, note that there is a price to pay. In contrast to classical theorem proving, the operator RES (Definition 3.1.9) is *not recursively enumerable* because it appeals to provability, when returning TRUE, and it appeals to *non-provability*, when returning FALSE. Nevertheless, in this chapter, by leveraging both the regression property and the representation theorem to the multiagent case, we have demonstrated that, under reasonable assumptions, reasoning in the multiagent case is not much harder than in the single agent case.

In practice, of course, we would like to reduce query evaluation (which also lies at the heart of RES) to a much more tractable problem that ordinary logical entailment. Therefore, it is quite common for applications to assume initial theories in the form of a closed database, for which query evaluation is tractable [Abiteboul et al., 1995], or propose ways to model and reason about incomplete information. One notable effort in this direction is the work of De Giacomo and Levesque [1999]. Although they do not consider knowledge and restrict themselves to a single agent, their work allows for open databases that are *locally complete*. Basically, they present an intuitively plausible requirement called the *just-in-time* property that rely on a robot's sensor data to fill in the gaps with incomplete knowledge. The idea is that whenever the truth of a sentence needs to be evaluated, they provide the conditions under which the suitable information necessary to evaluate the sentence is obtained by the means of sensing. This work has been extended further into a just-in-time regression algorithm in [De Giacomo et al., 2001]. In a nutshell, these results presents a general methodology that could be incorporated into regression-based formalisms such as ours.

Projection by regression in the presence of multiple agents has been addressed in the literature. For instance, the epistemic situation calculus [Scherl and Levesque, 2003] has been extended to the many agent case in earlier work [Shapiro et al., 2002]. Recently, Kelly and Pearce [2008] consider evaluating epistemic queries, including queries about common knowledge [Fagin et al., 1995], by means of a meta-level operator using regression. In contrast to these strands of work, we are mainly concerned with identifying how regression works in the presence of multiagent only knowing operators. As we have argued earlier, by being able to define initial knowledge in terms what is only known one obtains a natural means of reasoning about both beliefs and non-beliefs. Moreover, the epistemic situation calculus of Scherl and Levesque does not have an equivalent of the representation theorem. Therefore, the other approaches require a form of modal reasoning about the initial situation.

In some aspects, however, the formalisms are not comparable. On the one hand, in contrast to Kelly and Pearce we observed in Section 3.2.4 that common knowledge cannot be captured with our semantics. On the other hand, we mentioned that integrating only knowing in the situation calculus when the situation terms are explicit is very problematic [Lakemeyer, 1996; Lakemeyer and Levesque, 1998].

While regression is indeed a complete solution wrt arbitrary basic action theories, it is not without its prob-

lems. Complex projection tasks, especially open-ended ones involving a long sequence of actions, becomes unmanageable by regression. More precisely, reducing queries to the initial database becomes computationally expensive, at least linear in the number of actions and in the worst case, exponential [Reiter, 2001]. In the next chapter, we consider an important alternative called *progression*.

Chapter 5

Projection by Progression

Regression, which was investigated in the previous chapter, is not an effective choice for agents functioning autonomously for extended periods of time, involving, say, open-ended tasks. In these cases, it becomes essential to periodically update the initial knowledge base to one that reflects the changes due to actions that have already occurred. Equivalently, we are interested in updating what is only known after an action is performed. This has been identified as the problem of progression of basic action theories [Lin and Reiter, 1997].

As far as the computational feasibility of a progression mechanism is concerned, especially in practice, Liu and Levesque [2005a] observe that it must satisfy three main computational requirements. The first is that the progressed KB should be efficiently computable. The second requirement is that the new theory must be linear in the size of the initial one, so that progression can iterate. If this requirement is not satisfied, then the size of the KB grows after executing actions and it becomes unmanageable after many actions. The last requirement is that query evaluation, as in the case of regression, must be efficient against the KB.

Conceptually, regression and progression are natural duals of each other. One might except, therefore, that regression and progression have analogous logical foundations. But this is not the case. In the context of basic action theories, Lin and Reiter [1997] show that progression is not even computationally feasible in general, in the sense that it appeals to second-order logic. However, they identify two simple cases, based on syntactic restrictions on basic action theories, where progression is efficient. Based on that early work, Vassos et al. [2008] investigated first-order definable progression for so-called *local-effect* basic action theories [Liu and Levesque, 2005a], which are a generalization of one of the cases studied by Lin and Reiter. For a slightly smaller class of action theories, they also proved that progression is computable, in the sense that the resulting theory is finite. This work was generalized in [Liu and Lakemeyer, 2009], where it was proven that the progression of an arbitrary finite first-order theory wrt local-effect basic action theories is both first-order definable as well as computable. In addition, Liu and Lakemeyer [2009] proved that the progression of certain kinds of first-order theories wrt so-called *normal actions*, which are not local-effect, is first-order definable and computable. However, the size of the progressed knowledge base may blow-up exponentially. To that end, for certain kinds of first-order disjunctive information, called *proper*$^+$ *knowledge bases* [Lakemeyer and Levesque, 2002], Liu and Lakemeyer prove that progression wrt local-effect and normal actions is also efficient, under reasonable assumptions. Meanwhile, Vassos et al. [2009] considered another class of non-

local basic action theories, called *range-restricted* theories, and proved that progression for a certain kind of *possible-values* database is first-order definable and computable. All of these results are limited to the case of relational fluents.

In this chapter, we are concerned with identifying conditions under which progression becomes first-order definable, and proposing methodologies under which it can be efficiently computed. Note that, in contrast to earlier results, our representation language has functional fluents. To that end, we will be proving that in the presence of functional fluents, progression remains first-order definable for local-effect and normal action theories. Moreover, for a functional variant of proper$^+$ KBs, we will be proposing procedures for computing progression efficiently. Then, for a functional variant of proper$^+$ KBs, we will prove that progression wrt range-restricted action theories is both first-order definable as well as efficiently computable, under reasonable assumptions. Finally, to address the requirement that reasoning about the initial knowledge base should be efficient, we propose a *decidable* query evaluation mechanism for a large class of queries against our variant of proper$^+$ KBs.[1]

The remainder of the chapter is organized as follows. We first review a semantics by Lakemeyer and Levesque [2009] that explains how progression works in the context of only knowing. Then we turn to the computational requirements of progression, and present results in the order indicated above. Owing to the additional technical subtleties when dealing with an account of progression, we will restrict our attention to the single agent case. We leave the multiagent case for future work.

5.1 The Logic $\mathcal{ES}_o$

The notion of progression dealt in this chapter is very closely related to the work of Lin and Reiter.

5.1.1 Background

In a seminal paper, Lin and Reiter [1997] introduce the progression of situation calculus basic action theories. Given a basic action theory Σ, a one-step progression wrt a ground action r consists of replacing the initial theory Σ_0 in Σ by an appropriate set of sentences Σ_0' such that the original theory Σ and $(\Sigma - \Sigma_0) \cup \Sigma_0'$ agree on the future after doing r. The idea is that Σ_0' represents the successor to the initial situation, which intuitively means that those sentences that were true initially but no longer hold now are "forgotten".

Lin and Reiter define the concept of progression in terms of certain properties that the (Tarskian) models of the initial and the progressed theories show. The intuitive idea is this: Σ' is the progression of Σ iff for every (Tarskian) model M of Σ', there is a model M' of Σ such that they agree on all future situations. Equivalently, we say that Σ' is the progression of Σ wrt an action r if Σ' entails the same sentences about the future after r as Σ does. Lin and Reiter also show that when the initial theory is a finite one, then its progression is always representable using a second-order formula.[2] Since the two views are equivalent for finite theories, we will not present the formal aspects of the model-theoretic definition but only use a (variant)

[1] While decidability alone does not guarantee efficiency, it is a crucial step since in general the query evaluation problem for the considered fragment is undecidable.

[2] In recent work, Vassos and Levesque [2008] show that progression indeed cannot be captured by a first-order theory, even an infinite one.

of the syntactic representation in this chapter instead. But before going into that, let us intuitively see why $\mathcal{ES}$ cannot capture progression.

5.1.2 Why not $\mathcal{ES}$?

The semantics for the only knowing operator $\boldsymbol{O}$ as given in the logic $\mathcal{ES}$ does not have the desired properties. Roughly speaking, the problem is that the semantics for $\mathcal{ES}$ is in some sense "static". To see this, reconsider Example 4.1.3. If Υ and Σ denotes the basic action theories developed in the example corresponding to the real world and what the agent believes is true, then the following sentence in valid in $\mathcal{ES}$, as should be the case:

$$\Upsilon \wedge \boldsymbol{O}\Sigma \supset [forward][sonar]\boldsymbol{K}(distance = 3). \tag{5.1}$$

In other words, $\mathcal{ES}$ has reasonable properties regarding basic beliefs. But now, following the progression methodology, where the KB is updated after actions, consider what should be only known after *forward* and *sonar*. One expects that the distance fluent is now set to 3 units, but everything else remains the same. Formally,

$$\Upsilon \wedge \boldsymbol{O}(\Sigma) \supset [forward][sonar]\boldsymbol{O}(distance = 3 \wedge (\Sigma - \Sigma_0)) \tag{5.2}$$

should be valid. But it can be shown by means of the semantical definition of $\mathcal{ES}$ that this is not the case. Roughly speaking, the reason is that what is known initially is not forgotten, which contrasts with what is required for progression. Therefore, doing an action (sensing or otherwise) always leads to an *expansion* of belief, that is, more ends up being known. To this end, Lakemeyer and Levesque [2009] introduce the logic $\mathcal{ES}_o$, which differs from $\mathcal{ES}$ *only* in its treatment of the epistemic operators, which are handled by a notion of *progressing* the world states.

5.1.3 A Semantics

Syntactically, $\mathcal{ES}_o$ and $\mathcal{ES}$ are identical. Due to the second-order nature of progression, however, we extend the language in the following manner:

- Let us assume an infinite supply of rigid second-order function variables of arity k: $X_1^k, X_2^k, \ldots$
- We only need to include one extra formation rule for terms: if $\vec{t}$ are terms and X is a k-ary second-order function variable, then $X(\vec{t})$ is a term. By analogy to primitive (first-order) terms, by *primitive second-order term*, we mean one of the form $X(\vec{n})$ where X is a second-order function variable and $n_i \in \mathcal{Q}$.
- We only need to include one extra formation rule for formulas: if α is a formula and X is a second-order variable, then $\forall X\alpha$ is a formula.

For the purposes of this thesis, we will make the restriction (and assume henceforth) that second-order quantifiers are *only* applied to formulas that do not mention $\{\boldsymbol{K}, \boldsymbol{O}\}$. For instance, $\boldsymbol{K}\forall P\alpha$ and $\boldsymbol{O}\exists P\alpha$, where α does not mention $\{\boldsymbol{K}, \boldsymbol{O}\}$, are allowed, but $\forall P\boldsymbol{O}\alpha$ and $\exists P\boldsymbol{K}\alpha$ are not.

Now, we define a set of possible worlds $\mathcal{W}$ as before, except that in addition to interpreting primitive object terms, worlds will also interpret primitive second-order terms. That is:

- a world $w \in \mathcal{W}$ is a function
  - from primitive object terms and $\mathcal{Z}$ to $\mathcal{N}$;
  - from primitive second-order terms to $\mathcal{N}$.

When interpreting formulas with free variables, first-order variables are handled substitutionally, as before. To interpret second-order variables, we introduce the notation $w{\sim_X}w'$ to mean that w and w' agree on everything except assignments involving X.

Finally, we interpret arbitrary terms in the language as follows. As in $\mathcal{OL}$, names are rigid designators. Now, given a term t without variables, a world w, and an action sequence z, we define $|t|^z_w$ (to be read as "the coreferring standard name for t given w and z") by:

1. $|t|^z_w = t$ if $t \in \mathcal{Q}$;
2. $|f(\vec{t})|^z_w = w[f(\vec{n}), z]$, where $|t_i|^z_w = n_i$ and f is a function of the object sort;
3. $|A(\vec{t})|^z_w = A(\vec{n})$, where $|t_i|^z_w = n_i$ and A is a function of the action sort;
4. $|X(\vec{t})|^z_w = w[X(\vec{n})]$, where $|t_i|^z_w = n_i$ and X is a second-order variable.

We are now ready to give the meaning of truth. Given a sentence $\alpha \in \mathcal{ES}_o$, an epistemic state $e \subseteq \mathcal{W}$, and world w and an action sequence z, a semantics is given inductively:

1.-6. as in Section 4.1;

7. $e, w, z \models \forall X\alpha$ iff $e, w', z \models \alpha$ for every $w'{\sim_X}w$;[3]

But to give the meaning of epistemic operators, we define the progression of worlds and epistemic states as follows:

Definition 5.1.1. Let w be a world, z a sequence of actions and e any set of worlds. Then

1. w_z is a world such that $w_z[d, z'] = w[d, z \cdot z']$ for all primitive terms d and action sequences z';
2. $e^w_z = \{w'_z \mid w' \in e \text{ and } w' \simeq_z w\}$. ∎

Intuitively, one "clips" the world w after z and the resulting tree is the world w_z. Figure 5.1 illustrates the progression of the world w wrt a primitive action r_1.

Thus, w_z is exactly like w after z has occurred. In this sense, w_z is the *progression* of w after z. The epistemic state e^w_z then contains all the worlds in e which are progressed wrt z and are compatible with the real world w with regards to the sensing results until z. Of course when z is empty, $e^w_z = e$ by the definition of $\simeq_z$. With this in hand, we give a semantics for subjective formulas as follows:

8. $e, w, z \models \boldsymbol{K}\alpha$ iff for all $w' \in e^w_z$, $e^w_z, w', \langle\rangle \models \alpha$;

[3] In [Lakemeyer and Levesque, 2011], second-order variables are semantically evaluated by means of *variable maps*. Our (much simpler) account is due to Jens Claßen [personal communication, 2011]. But observe that our semantics for second-order quantifiers works as intended *only* when α is syntactically restricted to not mention $\boldsymbol{K}$ and $\boldsymbol{O}$, in which case it suffices to look at worlds w' differing from w wrt X. We thank Hector Levesque [personal communication, 2012] for bringing this our attention.

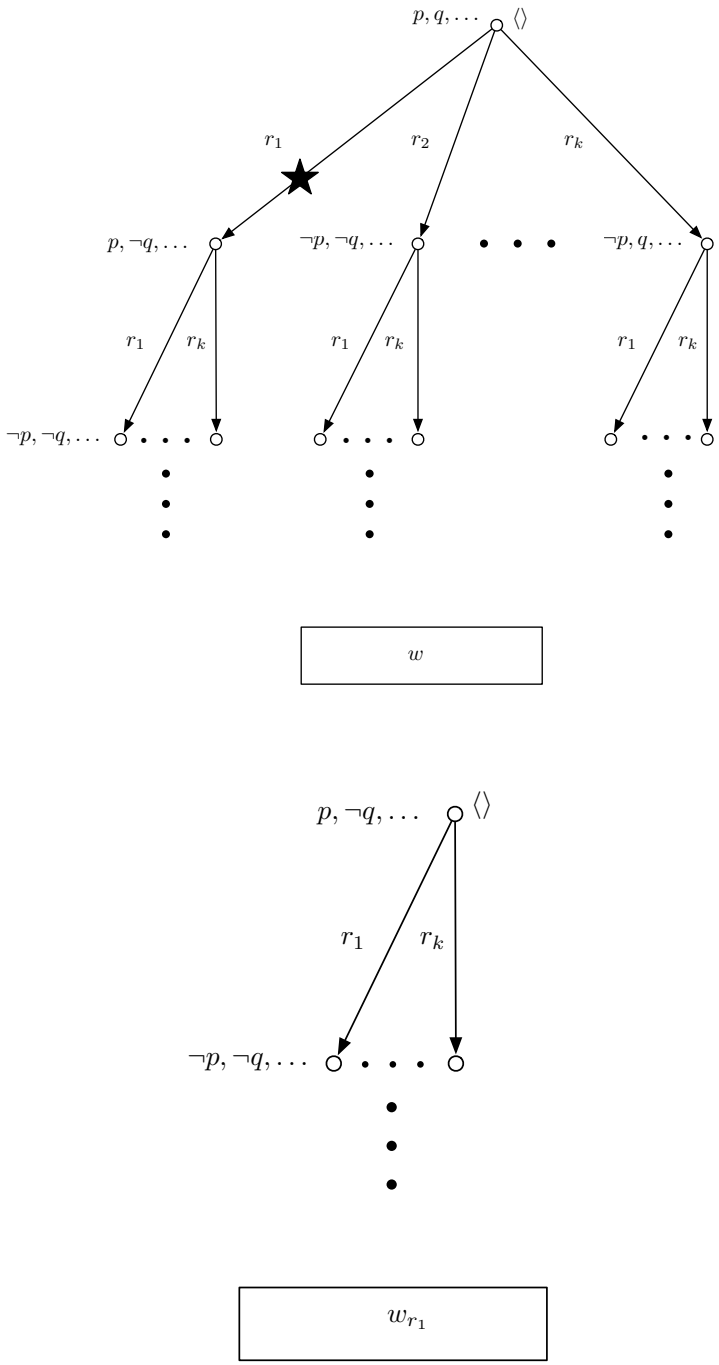

Figure 5.1: The progression of w wrt r_1.

9. $e, w, z \models \boldsymbol{O}\alpha$ iff for all w', $w' \in e_z^w$ iff $e_z^w, w', \langle\rangle \models \alpha$.

That is, knowing α in e and w after z means that α is true at all the progressed worlds of e that are compatible with w. Only knowing α has the same relationship to knowing as before, that is, the "if" is replaced by an "iff". In other words, e_z^w must contain every world that satisfies α.

Satisfiability and validity are then defined in an obvious way.

5.1.4 Properties

Let us first draw comparisons to the notion of only knowing in $\mathcal{ES}$. To disambiguate between the two semantics, let us denote the only knowing operator of $\mathcal{ES}$ as $\boldsymbol{O}'$. From the observation made earlier that $e_z^w = e$ when $z = \langle\rangle$, it follows that the two operators coincide when no actions have been performed:

Proposition 5.1.2. *[Lakemeyer and Levesque, 2009]*
$\models \boldsymbol{O}\alpha \equiv \boldsymbol{O}'\alpha$.

However, this is no longer the case after actions are performed. For instance, as we observed in (5.2), the older only knowing interpretation does *not* have desirable properties regarding what should be only known after actions. As we shall see shortly, the modified version, however, does capture the desiderata.

Moving to the usual **K45** properties, it is not hard to show that they also hold in $\mathcal{ES}_o$.

Proposition 5.1.3. *[Lakemeyer and Levesque, 2009]*

1. $\models \Box(\boldsymbol{O}\alpha \supset \boldsymbol{K}\alpha)$;
2. $\models \Box(\boldsymbol{K}\alpha \supset \boldsymbol{K}\boldsymbol{K}\alpha)$;
3. $\models \Box(\neg\boldsymbol{K}\alpha \supset \boldsymbol{K}\neg\boldsymbol{K}\alpha)$;
4. $\models \Box(\forall\vec{x}\boldsymbol{K}\alpha \supset \boldsymbol{K}(\forall\vec{x}\alpha))$.

Proof: Showing item 3, let $e, w, z \models \neg\boldsymbol{K}\alpha$. Then for some $w' \in e_z^w$, $e_z^w, w', \langle\rangle \not\models \alpha$. It then follows that for any $w'' \in e_z^w$, $e_z^w, w'', \langle\rangle \models \neg\boldsymbol{K}\alpha$. Therefore $e_z^w, w, \langle\rangle \models \boldsymbol{K}\neg\boldsymbol{K}\alpha$ as needed. ∎

5.1.5 Progression = Only Knowing after Actions

In this section, we show that the semantics of only knowing is compatible with Lin and Reiter's notion of progression. In the following, for a given basic action theory Σ, we often write ϕ for the initial theory Σ_0 and write $\Box\beta$ for the rest of the action theory $\Sigma_{pre} \cup \Sigma_{post} \cup \Sigma_{sense}$. We assume that φ refers to the *rhs* of the definition of SF in Σ, and γ_f is the *rhs* of the successor state axiom for the fluent f. Also, let $f_1, \ldots, f_k$, denoted $\vec{F}$, consist of all the fluent symbols appearing in Σ, and let $\vec{P}$ be corresponding second-order variables, where each P_i has the same arity as f_i. Then $\alpha^{\vec{F}}_{\vec{P}}$ denotes the formula α with every occurrence of f_i replaced by P_i.

The following result characterizes in general terms all that is known after performing an action:

Theorem 5.1.4. *[Lakemeyer and Levesque, 2009]*
Let r be a action standard name, then

$$\models \boldsymbol{O}(\phi \wedge \Box\beta) \wedge SF(r) = x \supset [r]\boldsymbol{O}(Prog(\phi) \wedge \Box\beta),$$

where $Prog(\phi) = \exists \vec{P}.\ [(\phi \wedge \varphi_r^v)_{\vec{P}}^{\vec{F}} \wedge \bigwedge \forall \vec{x}, y.\ f(\vec{x}) = y = \gamma_{f_r}^{v\vec{F}}{}_{\vec{P}}]$.

The intuition behind $Prog(\phi)$ is as follows. Observe that ϕ determines the initial values of fluents and the successor state axioms decide what these values look like after actions. Therefore, to obtain the progression of a given basic action theory wrt r we need to consider the union of the initial theory and the instantiations of the successor state axioms wrt r but while taking care to eliminate the initial values of fluent atoms that have changed on doing r. To accomplish this, the trick is to use second-order variables to represent the initial values of fluents which are then to be eliminated.

The above theorem essentially says that if all that is known initially is a basic action theory, then after doing an action r the agent knows another basic action theory but with the initial theory ϕ replaced by the progressed one $Prog(\phi)$. Given that the agent knows another basic action theory after an action, the progression procedure can iterate. The proof of this theorem relies on two intermediate lemmas.

Lemma 5.1.5. *[Lakemeyer and Levesque, 2009]*
Suppose $w' \models \phi \wedge \Box\beta$ *and* $w' \simeq_r w$. *Then* $w'_r \models Prog(\phi) \wedge \Box\beta$.

Lemma 5.1.6. *[Lakemeyer and Levesque, 2009]*
Suppose $w' \models Prog(\phi) \wedge \Box\beta$. *Then there exists a world* w'' *such that* $w''_r = w'$, $w'' \simeq_r w$ *and* $w'' \models \phi \wedge \Box\beta$.

The idea behind these lemmas is to show that worlds satisfy $\phi \wedge \Box\beta$ iff their progressed versions satisfy $Prog(\phi) \wedge \Box\beta$, provided that they are compatible with the real world. In later sections, when we turn to cases where progression is first-order definable, an important step will be to adapt these lemmas in the sense of proving that $Prog(\phi)$ is equivalent to a first-order formula.

Theorem 5.1.4 is very close to Lin and Reiter's concept in the sense that what is only known after an action is nothing but what Lin and Reiter define as the progression of an initial theory, with the exception of two differences. In Lin and Reiter's definition, the new theory is additionally conjoined with the unique name axioms for actions. We do not need that, mainly because, as we have remarked earlier, the unique name assumption is built into the logic. Secondly, their definition does not consider sensing, essentially, because they do not consider knowledge. On the other hand, with the above theorem, the basic action theory can include non-trivial sensing results which become part of the progressed theory after actions.

Example 5.1.7. Let us illustrate how the result captures our desiderata regarding progression. Consider Example 4.1.3 yet again. We now have

$$\models \Upsilon \wedge \boldsymbol{O}(\phi \wedge \Box\beta) \supset [forward]\boldsymbol{O}(Prog(\phi) \wedge \Box\beta)$$

where $Prog(\phi) = Prog(distance = 4 \vee distance = 5) =$

$$\exists P.\ [(P = 4 \vee P = 5) \wedge distance = y \equiv$$

$$forward = forward \wedge y = P - 1 \vee$$

$forward \neq forward \wedge y = P]$.

Using the fact that two primitive terms are equal only when they are identical, it is easy to see that $Prog(\phi) = [distance = 3 \vee distance = 4]$. By means of similar steps, after sensing it is not too hard to show

$$\models \Upsilon \wedge \boldsymbol{O}(\Sigma) \supset [forward][sonar]\boldsymbol{O}(distance = 3 \wedge \Box\beta). \blacksquare$$

We had remarked earlier that the progression of a basic action theory wrt a ground action, say r, and the original theory are equivalent in how they describe the future of r. This view is formally justified in $\mathcal{ES}_o$ as follows:

Theorem 5.1.8. *[Lakemeyer and Levesque, 2009]*
$\models \boldsymbol{O}(\phi \wedge \Box\beta) \wedge SF(r) = x \supset [r]\boldsymbol{K}\alpha$ *iff* $\models \boldsymbol{O}(Prog(\phi) \wedge \Box\beta) \supset \boldsymbol{K}\alpha$.

That is, it follows from the agent's initial knowledge base that α is known after executing the action r iff knowing α follows from the progressed knowledge base. In this way, progression addresses projection.

5.2 First-Order Definability of Progression

In this section, we examine two cases where progression is first-order definable. We extend previous results by Liu and Lakemeyer [2009], where first-order progression was proved for theories not mentioning function symbols. The results make use of the concept of *forgetting*, which we address first and review a few simple results.

5.2.1 Forgetting

In order to prepare for our results regarding first-order definable progression, we first consider *forgetting*. Lin and Reiter [1994] defined a notion of *forgetting* a ground atom or a predicate from a logical theory. The idea is to remove all the information that is no longer true. For example, if John is a student of physics and he is removed from the university, then a database maintaining the information needs to forget that fact about John. Now, if the university is not offering physics anymore, then the database may need to forget the relation denoting students of physics. Intuitively, after forgetting, the resultant theory should be weaker than the original, but nevertheless entail all the sentences as the original which are "irrelevant" to the atom or the predicate that is forgotten.

Lin and Reiter show that while forgetting a ground atom is first-order definable, forgetting a predicate requires second-order logic. We adapt their ideas below for a language with functions. Note that while their definitions are given for standard FOL, we consider analogous notions for our semantical framework.

We begin with a few preliminaries. In what follows, abusing notation somewhat, we write $w' \sim_d w$ to mean that w' and w agree on everything, except maybe the value to the primitive term d initially. Going further, we write $w' \sim_f w$ to mean that w' and w agree on everything, except maybe on all the primitive terms which are instances of f initially. Formally:

- We write $w' \sim_d w$ to mean that:

  - for all primitive terms d' except d, $w'[d', \langle\rangle] = w[d', \langle\rangle]$;
  - for all primitive terms d', $w'[d', z] = w[d', z]$ for all $z \in \mathcal{Z} - \{\langle\rangle\}$.

- We write $w' \sim_f w$ to mean that:
  - for all primitive terms d' which are not instances of f, $w'[d', \langle\rangle] = w[d', \langle\rangle]$;
  - for all primitive terms d', $w'[d', z] = w[d', z]$ for all $z \in \mathcal{Z} - \{\langle\rangle\}$.

We now define the notion of forgetting.

Definition 5.2.1. (Forgetting.) Let δ denote either a primitive term, or a function symbol. Given a fluent formula ϕ, we say any fluent formula ϕ' is the result of forgetting δ from ϕ, denoted $forget(\phi, \delta)$, if for any world w, $w \models \phi'$ iff there is a world w' such that $w' \models \phi$ and $w \sim_\delta w'$.

Inductively define $forget(\phi, \{\delta_1, \ldots, \delta_k\})$, *i.e.* the forgetting of $\delta_1, \ldots, \delta_k$ from ϕ, as $forget(forget(\phi, \delta_1), \ldots, \delta_k)$. ∎

We proceed to show that forgetting a primitive term d is first-order definable. It then follows that forgetting a finite set of primitive terms is also first-order definable. We will assume henceforth that every fluent formula is in FNF, as formulated in Definition 4.1.4, where functions appear only in the form $f(\vec{t}) = t'$ where t_i and t' are either variables of names.

Definition 5.2.2. Let ϕ be a fluent sentence, $f(\vec{m})$ a primitive term, and x a variable not appearing in ϕ. We write $\phi[f(\vec{m}) = x]$ to denote the result of replacing every occurrence of $f(\vec{t}) = t'$ in ϕ with

$$(\vec{t} = \vec{m} \wedge t' = x) \vee (\vec{t} \neq \vec{m} \wedge f(\vec{t}) = t'). \quad ∎$$

Proposition 5.2.3. *Let ϕ be a fluent formula, x a variable not appearing in ϕ and d a primitive term. Suppose $w \models d = n$ and $w \sim_d w'$. Then $w \models \phi$ iff $w' \models (\phi[d = x])^x_n$.*

Proof: Let d denote $f(\vec{m})$. By induction on ϕ. We only show the base case since the case of logical connectives is straightforward.

- Suppose ϕ is $d = n$. Then $\phi[d = x]$ is $(\vec{m} = \vec{m} \wedge x = n) \vee (\vec{m} \neq \vec{m} \wedge f(\vec{m}) = n)$. Therefore, $\phi[d = x]^x_n$ is equivalent to TRUE.[4] Since $w \models d = n$, it follows that $w \models \phi$. Thus $w \models \phi$ iff $w' \models \phi[d = x]^x_n$.
- The argument is similar if ϕ is $d \neq n'$ where n is distinct from n'. Here too $\phi[d = x]^x_n$ is equivalent to TRUE, and since $w \models d = n$, it follows that $w \models \phi$. Thus $w \models \phi$ iff $w \models \phi[d = x]^x_n$.
- Suppose ϕ is $d = n'$ or $d \neq n$, where n' is distinct from n. In both cases, $\phi[d = x]^x_n$ is equivalent to FALSE. Moreover, since $w \models d = n$, $w \not\models d = n'$ and $w \not\models d \neq n$. Therefore, $w \models \phi$ iff $w' \models \phi[d = x]^x_n$. ∎

Theorem 5.2.4. *Let ϕ be a fluent sentence, x a variable that is not mentioned in ϕ and d a primitive term. Then $\models forget(\phi, d) \equiv \exists x \phi[d = x]$.*

[4] Recall that FALSE is equivalent to any sentence β that is always-false, *i.e.* β does not hold at any world, such as $\neg\forall(x = x)$ or $n = n'$ where n and n' are distinct names. TRUE is the negation of that sentence.

Proof: Let w be a world and let us denote $\phi[d = x]$ with ϕ'. We need to show that $w \models \exists x\phi'$ iff there is a world w', $w' \models \phi$ and $w' \sim_d w$.

Suppose the latter. Suppose also that $w' \models d = n$. By Proposition 5.2.3, $w \models \phi'^x_n$, or $w \models \exists x\phi'$.

Now, suppose $w \models \exists x\phi'$, say $w \models \phi'^x_n$. Pick any world $w' \sim_d w$ such that $w' \models d = n$. By Proposition 5.2.3, $w' \models \phi$. ∎

Example 5.2.5. We illustrate forgetting of primitive terms with a few examples:

- Let D be a block,[5] and let $\phi_1 = [at(D) = roomC \vee at(D) = roomD]$. Suppose $d = \{at(D)\}$. Then $\phi_1[d = x]$ is

  $$(D = D \wedge x = roomC) \vee (D \neq D \wedge at(D) = roomC) \vee$$
  $$(D = D \wedge x = roomD) \vee (D \neq D \wedge at(D) = roomD).$$

  Then, $\exists x\phi_1[d = x]$ is equivalent to

  $$\exists x((D = D \wedge x = roomC) \vee (D = D \wedge x = roomD))$$

  i.e. $\exists x(x = roomC \vee x = roomD)$, which is equivalent to TRUE. Therefore forgetting d from ϕ_1 is equivalent to TRUE.

- Let C and D be blocks, and let $d = \{at(D)\}$ and ϕ_2 be the following theory.

  $$\{\forall y.\, y = roomC \supset at(D) \neq y, \exists y.\, in(y) \neq roomA, at(C) = roomD \supset at(D) \neq roomD\}.$$

  Then $\phi_2[d = x]$ simplifies to

  $$\{\forall y.\, y = roomC \supset x \neq y, \exists y.\, in(y) \neq roomA, at(C) = roomD \supset x \neq roomD\}.$$

  Then $forget(\phi_2, d)$ simplifies to $\{\exists y.\, in(y) \neq roomA\}$.

- Let ϕ_3 be $\forall x f(x) = 1$ and let d denote $f(1)$. Then, $\phi_3[d = y]$ simplifies to

  $$\forall x((x = 1 \wedge y = 1) \vee (x \neq 1 \wedge f(x) = 1)).$$

  Then $\exists y\phi_3[d = y]$ is equivalent to

  $$\exists y \forall x((x = 1 \wedge y = 1) \vee (x \neq 1 \wedge f(x) = 1))$$

  which is equivalent to

  $$\forall x(x = 1 \vee (x \neq 1 \wedge f(x) = 1)).$$

  Therefore $forget(\phi_3, d)$ is equivalent to $\forall x(x \neq 1 \supset f(x) = 1)$. ∎

We remark that forgetting can always be expressed using second-order logic.

Theorem 5.2.6. $\models forget(\phi, f) \equiv \exists P \phi^f_P$.

[5] Recall the convention we have of treating *proper names* as *object standard names*.

Proof: We are asked to show that forgetting f from ϕ is equivalent to $\exists P \phi^f_P$. That is, we have to prove that for any world w that satisfies $\exists P \phi^f_P$, there is a world w' such that $w' \sim_f w$ such that $w' \models \phi$.

Suppose the latter. Let w'' be a world $w'' \sim_P w'$ such that $w''[P(\vec{m}), \langle\rangle] = w'[f(\vec{m}), \langle\rangle]$. It is easy to show (by induction) that $w'' \models \phi^f_P$. Therefore, $w' \models \exists P \phi^f_P$. Since $w' \sim_f w$, $w \models \exists P \phi^f_P$.

Suppose $w \models \exists P \phi^f_P$, or $w \models \phi^f_P$. Construct a world $w' \sim_f w$ such that $w'[f(\vec{m}), \langle\rangle] = w[P(\vec{m}), \langle\rangle]$. It is easy to show (by induction) that $w' \models \phi$. ∎

Example 5.2.7. Consider ϕ_1 from Example 5.2.5. Forgetting *at* from ϕ_1 is, by Theorem 5.2.6, $\exists P.\ P(D) = roomC \vee P(D) = roomD$ and this is equivalent to TRUE. ∎

Lin and Reiter [1994] show that forgetting a function (or a relation) is not first-order definable in general. However, forgetting a function f from a fluent sentence ϕ is first-order definable in the following special case.[6]

Suppose ϕ entails that the values of two functions f and g of the same arity differ only at a finite number of *known* instances, then forgetting f from ϕ can be obtained by forgetting those instances where the two functions differ and replacing f everywhere in the resultant by g.

Let $\Omega = \{\vec{m}_1, \ldots, \vec{m}_k\}$ denote a finite set of name vectors, of the same arity as f and g. Now, define $\Delta = \{f(\vec{m}) \mid \vec{m} \in \Omega\}$, and let $f \approx_\Omega g$ denote the sentence $\forall \vec{x}. \vec{x} \notin \Omega \supset f(\vec{x}) = g(\vec{x})$. Then

Proposition 5.2.8. *Let ϕ, Ω and Δ be as above. Suppose $w \models f \approx_\Omega g$. Then $w \models forget(\phi, \Delta)$ iff $w \models forget(\phi, \Delta)^f_g$.*

Proof: For ease of exposition, let Δ be a singleton, say $\{f(\vec{m})\}$, which means that $\Omega = \{\vec{m}\}$. If Δ is not a singleton, say $\Delta = \{f(\vec{m}_1), \ldots, f(\vec{m}_k)\}$, then the argument is the same as the one given below except that instead of $forget(\phi, f(\vec{m}))$ we do $forget(\phi, \{f(\vec{m}_1), \ldots, f(\vec{m}_k)\})$.

The proof is by induction on ϕ. We only consider the base case since other cases are straightforward.

Suppose ϕ is $f(\vec{m}) \circ n$, where $\circ \in \{=, \neq\}$. It is easy to see, then, that $\exists x \phi[d = x]$ is equivalent to TRUE. Therefore $w \models forget(\phi, d)$ iff $w \models forget(\phi, d)^f_g$.

Suppose ϕ is $f(\vec{m}') \circ n$ where $\vec{m}' \neq \vec{m}$. Then $\phi[d = x]$ simplifies to $[(\vec{m}' = \vec{m} \wedge x = n) \vee (\vec{m}' \neq \vec{m} \wedge f(\vec{m}') = n)]$ if $\circ$ is $=$, or to $\neg[(\vec{m}' = \vec{m} \wedge x = n) \vee (\vec{m}' \neq \vec{m} \wedge f(\vec{m}') = n)]$ otherwise. Basically, $\exists x \phi[d = x]$ simplifies to $f(\vec{m}') \circ n$. Since $w[f(\vec{m}'), \langle\rangle] = w[g(\vec{m}'), \langle\rangle]$, we have that $w \models g(\vec{m}') \circ n$ iff $w \models f(\vec{m}') \circ n$. Therefore, $w \models forget(\phi, d)$ iff $w \models forget(\phi, d)^f_g$. ∎

Theorem 5.2.9. *Let ϕ, Ω and Δ be as above. Then $\models forget(\phi \wedge f \approx_\Omega g, f) \equiv forget(\phi, \Delta)^f_g$.*

Proof: For ease of exposition, let Δ be a singleton, say $\{f(\vec{m})\}$, denoted d, which means that $\Omega = \{\vec{m}\}$. If Δ is not a singleton, say $\Delta = \{f(\vec{m}_1), \ldots, f(\vec{m}_k)\}$, then the argument is the same as the one given below except that instead of $forget(\phi, f(\vec{m}))$ we do $forget(\phi, \{f(\vec{m}_1), \ldots, f(\vec{m}_k)\})$.

From Theorem 5.2.4, we know that $forget(\phi, d)$ is equivalent to $\exists x \phi[d = x]$. We show that $w \models \exists x \phi[d = x]^f_g$ iff there exist a world w' such that $w' \models \phi \wedge f \approx_\Omega g$ and $w' \sim_f w$.

[6] This is inspired by an analogous result on forgetting predicates for a special case from [Liu and Lakemeyer, 2009].

Suppose the latter. Suppose $w' \models d = n$. Since $w' \models \phi$ from Proposition 5.2.3, it follows that $w' \models \phi[d = x]^x_n$, *i.e.* $w' \models \exists x \phi[d = x]$, *i.e.* $w' \models forget(\phi, d)$. Since $w' \models f \approx_\Omega g$, from Proposition 5.2.8, $w' \models forget(\phi, d)^f_g$. Since $w' \sim_f w$, $w \models forget(\phi, d)^f_g$.

For the other direction, suppose $w \models (\exists x \phi[d = x])^f_g$, say $w \models (\phi[d = x]^x_n)^f_g$. Since that formula does not mention f, let w' be a world that agrees with w on everything except f, *i.e.* $w' \sim_f w$, such that for every vector of names $\vec{m}' \neq \vec{m}$, let $w'[f(\vec{m}'), \langle\rangle] = w[g(\vec{m}'), \langle\rangle]$, and let $w'[f(\vec{m}), \langle\rangle] = n$. Since $w' \sim_f w$, $w' \models (\phi[d = x]^x_n)^f_g$. Now since $w' \models f \approx_\Omega g$, by means of Proposition 5.2.8, $w' \models (\phi[d = x]^x_n)$. Finally since $w' \models d = n$, by Proposition 5.2.3 we have that $w' \models \phi$. ∎

5.2.2 Progression for Local-Effect Actions

In many domains, actions have a *locality* property in the sense that they only affect those objects that are explicitly mentioned as arguments of the action. For example, moving an object to some location, say the object C to the location *roomC* by means of the primitive action $move(C, roomC)$ clearly mentions the object itself. This class of action theories are, for that reason, called *local-effect* and were first introduced in [Liu and Levesque, 2005a]. In general, if a local-effect action $A(\vec{o})$ affects a fluent term $f(\vec{n})$, then $\vec{n}$ is contained in $\vec{o}$. This contrasts with actions such as an exploding bomb, say by means of the primitive action *explode*, that results in the destruction of all other objects in the vicinity of the bomb, none of which are mentioned in the action.

Definition 5.2.10. (Local-effect action theories.) Let the successor state axiom for the fluent f be of the form:

$$\Box[v] f(\vec{x}) = y \equiv \gamma_f(\vec{x}, y, v) \vee f(\vec{x}) = y \wedge \neg \exists h. \gamma_f(\vec{x}, h, v).$$

The successor state axiom is *local-effect* if $\gamma_f(\vec{x}, y, v)$ is a disjunction of formulas of the form:

$$\exists \vec{u}[v = A(\vec{z}) \wedge \mu(\vec{z})]$$

where $\vec{z}$ contains variables from $\vec{x}$ and y, $\vec{u}$ corresponds to the remaining variables in $\vec{z}$, and $\mu(\vec{z})$ is called the *context formula.* A basic action theory is local-effect iff each of the successor state axioms in Σ_{post} is local-effect. ∎

In their paper, Lin and Reiter [1997] identified a class of action theories they call *strictly context-free* where progression is first-order definable. Essentially, however, strictly context-free theories are a subset of local-effect action theories where the context formula is simply TRUE. Thus, local-effect action theories are a proper generalization of strictly context-free action theories.

An illustration of a local-effect action theory follows.

Example 5.2.11. (The office robot domain.) For illustration, we consider the simple example of a robot moving blocks in an office environment. In particular, assume the delivery of blocks from *roomA* to other locations, and let the action $move(x, y)$ capture the moving of x from *roomA* to y. The basic action theory is presented in Figure 5.2, where $at(x)$ gives the current location of x. For simplicity, we assume throughout that actions are always executable and that they return trivial sensing results.

$\Sigma_0 = \{\forall x.\ x = C \vee x = D \supset at(x) = roomA\}$.

$\Sigma_{post} = \{\Box[v]at(x) = y \equiv$

$v = move(x, y) \wedge at(x) = roomA \vee$

$at(x) = y \wedge \neg\exists h.\ (v = move(x, h) \wedge at(x) = roomA)\}$.

$\Sigma_{pre} = \{\Box Poss(v) = 1 \equiv \text{TRUE}\}$;

$\Sigma_{sense} = \{\Box SF(v) = 1 \equiv \text{TRUE}\}$.

Figure 5.2: The office robot domain.

Observe that the successor state axiom is local-effect according to Definition 5.2.10, and therefore, the basic action theory of the office robot domain is local-effect. ∎

The instantiation of a local-effect successor state axiom on a primitive action can be significantly simplified, as the following proposition shows:

Proposition 5.2.12. *Let $A(\vec{o})$ be any primitive action. Suppose that the successor state axiom for fluent f is local-effect. Then there exists a formula $\delta(\vec{x}, y)$ of the form:*

$$\vec{x} = \vec{m}_1 \wedge y = n_1 \wedge \mu_1 \vee \ldots \vee \vec{x} = \vec{m}_k \wedge y = n_k \wedge \mu_k$$

where $\vec{m}$ and n are name vectors contained in $\vec{o}$ and μ_i do not contain free variables such that the following holds:

$$\models \forall \vec{x}, y.\ \gamma_f(\vec{x}, y, A(\vec{o})) \equiv \delta(\vec{x}, y).$$

Proof: Since the successor state axiom for f is local-effect, $\gamma_f(\vec{x}, y, v)$ is a disjunction of formulas of a certain form (Definition 5.2.10). Then, for any world w, $w \models \gamma_f(\vec{x}, y, A(\vec{o}))$ iff $w \models \exists \vec{u}.\ [A(\vec{o}) = A(\vec{z}) \wedge \mu(\vec{z})]$ for some disjunction iff (by uniqueness of actions) $w \models \exists \vec{u}.\ [\vec{o} = \vec{z} \wedge \mu(\vec{z})]$ iff $w \models \vec{x} = \vec{m} \wedge y = n \wedge \mu(\vec{o})$, where $\vec{x}$ and y are contained in $\vec{z}$ and corresponds to $\vec{m}$ and n in $\vec{o}$. ∎

Example 5.2.13. Let $\gamma_{at}(x, y, v)$ be $v = move(x, y) \wedge at(x) = roomA$. Then the instantiation of γ_{at} wrt $move(C, roomC)$ simplifies to $x = C \wedge y = roomC \wedge at(C) = roomA$. ∎

Without loss of generality, we assume henceforth that after performing an action, every successor state axiom is simplified to a form as indicated by Proposition 5.2.12. Of course, the simplified formula may look different for every primitive action. The motivation behind this simplification is that it now becomes possible to identify a finite number of fluent terms that are affected after a primitive action is performed. Following [Vassos et al., 2008; Liu and Lakemeyer, 2009], we make this precise:

Definition 5.2.14. Let Σ be a basic action theory, defined over the set of fluents $\mathcal{F}$, that is local-effect and suppose that $A(\vec{o})$ is a primitive action. Without loss of generality, let the instantiation of the successor state axioms for each $f \in \mathcal{F}$, *viz.* $\gamma_f(\vec{x}, y, A(\vec{o}))$, be simplified as indicated by Proposition 5.2.12, *i.e.* to the formula $\delta(\vec{x}, y)$. Now, define the *argument set* of f wrt $A(\vec{o})$ as the following set Ω_f of name vectors:

$$\Omega_f = \{\vec{m} \mid \vec{x} = \vec{m} \text{ appears in } \delta(\vec{x}, y)\}.$$

Then, define the *characteristic set* of $A(\vec{o})$ as the following set of primitive fluent terms:

$$\Delta = \{f(\vec{m}) \mid \vec{m} \in \Omega_f \text{ for fluent } f \in \mathcal{F}\}. \quad \blacksquare$$

It is worth noting that since $\mathcal{F}$ is finite, both Ω_f and Δ are also *finite*.

Example 5.2.15. Consider the basic action theory of the office robot domain from Example 5.2.11, and suppose $move(C, roomC)$ has occurred. Then, by way of Example 5.2.13, we have $\Omega_{at} = \{C\}$. Therefore the characteristic set of $move(C, roomC)$ is $\{at(C)\}$. ∎

The argument set Ω_f essentially identifies all primitive terms from $f(\vec{x})$ which are affected after the action. Equivalently, for every vector of names $\vec{n}$ not in Ω_f, it follows that the value of $f(\vec{n})$ remains the same after the action. The following proposition proves this property:

Proposition 5.2.16. *Let Σ be a basic action theory that is local-effect, let r denote the primitive action $A(\vec{o})$, and let Ω_f be the argument set of a fluent $f \in \mathcal{F}$ wrt r. Then*

$$\Sigma_{post} \models \forall \vec{x}.\ \vec{x} \notin \Omega_f \supset [r]f(\vec{x}) = y \equiv f(\vec{x}) = y.$$

Proof: Consider any $\vec{n} \notin \Omega_f$. Now, without loss of generality, assume that $\gamma_f(\vec{x}, y, r)$ is simplified to obtain disjunctions of the form $\vec{x} = \vec{m} \wedge y = n \wedge \mu$. It then follows that $\Sigma_{post} \models ([r]f(\vec{m}') = y) \equiv (f(\vec{m}') = y \wedge \neg\exists h \gamma_f(\vec{m}', h, r))$ for $\vec{m}' \notin \Omega_f$. By Proposition 5.2.12, $\gamma_f(\vec{m}', h, r)$ is logically equivalent to disjunctions of formulas of the form $\vec{x} = \vec{m} \wedge h = n \wedge \mu$. This implies that there is no world such that $w \models \neg\exists h.\gamma_f(\vec{m}', h, r)$. Therefore $\Sigma_{post} \models \forall y[([r]f(\vec{m}') = y) \equiv f(\vec{m}') = y]$. ∎

Example 5.2.17. Continuing the office robot domain from Example 5.2.11, let r denote $move(C, roomC)$. We identified in Example 5.2.15 that the argument set of *at* wrt r is the singleton $\{C\}$. So, now, consider the fluent term $at(D)$, where of course $\{D\} \notin \Omega_{at}$. Then, as Proposition 5.2.16 indicates, it is easy to verify

$$\Sigma_{post} \models \forall x[at(D) = x \equiv [r]at(D) = x].$$

That is, the location of D does not change after C is delivered. ∎

We now proceed to show that progression of a local-effect basic action theory wrt a primitive action essentially corresponds to forgetting all fluent terms in the characteristic set. By formulating this property in terms of Theorem 5.2.9, progression becomes first-order definable.

Theorem 5.2.18. *Let Σ be local-effect, and r any primitive action. Let $\vec{P}$ be a fresh set of functions. Then*

$$\models \boldsymbol{O}(\phi \wedge \Box\beta) \wedge SF(r) = x \supset [r]\boldsymbol{O}(Prog(\phi) \wedge \Box\beta)$$

where

$Prog(\phi) = forget((\phi \wedge \varphi_r^v)_{\vec{P}}^{\vec{F}} \wedge \Omega_{ss}, \Delta_{\vec{P}}^{\vec{F}})_{\vec{F}}^{\vec{P}}$, *and*

$\Omega_{ss} = \{f(\vec{m}) = y \equiv \gamma_f(\vec{m}, y, r)_{\vec{P}}^{\vec{F}} \mid \vec{m} \in \Omega_f \text{ for fluent } f \in \mathcal{F}\}$.

Intuitively, the set of sentences Ω_{ss} denotes the instantiations of the successor state axioms wrt the characteristic set. The proof is as follows. We first note the following property regarding progressed worlds.

Proposition 5.2.19. *[Lakemeyer and Levesque, 2009]*
If ϕ is a fluent sentence then $w, z \cdot z' \models \phi$ iff $w_z, z' \models \phi$.

We now prove that if a world satisfies a basic action theory then its progressed counterpart satisfies the progression of the basic action theory, which is first-order definable for local-effects. This result is an extension of Lemma 5.1.5, which formulates the general (second-order) definition of the progressed theory.

Let us use ψ to denote the sentence $(\phi \wedge \varphi_r^v)_{\vec{P}}^{\vec{F}} \wedge \Omega_{ss}$.

Lemma 5.2.20. *Suppose $w' \models \phi \wedge \Box\beta$ and $w' \simeq_r w$. Then $w'_r \models forget(\psi, \Delta_{\vec{P}}^{\vec{F}})_{\vec{F}}^{\vec{P}} \wedge \Box\beta$.*

Proof: Let us start by noting that if $w' \models \Box\beta$ then $w' \models [r]\Box\beta$, *i.e.* $w', r \models \Box\beta$. By Proposition 5.2.19, $w'_r \models \Box\beta$.

Next consider that $w' \models [r]f(\vec{x}) = y \equiv \gamma_f(\vec{x}, y, r) \vee f(\vec{x}) = y \wedge \neg\exists h\gamma_f(\vec{x}, h, r)$ for every $f \in \mathcal{F}$ because $w' \models \Box\beta$. Observe that for $\vec{m} \in \Omega_f$, it is easy to see that $w' \models [r]f(\vec{m}) = y \equiv \gamma_f(\vec{m}, y, r)$. In contrast, by means of Proposition 5.2.16, we have that $w' \models \forall\vec{x}.\vec{x} \notin \Omega_f \supset ([r]f(\vec{x}) = y) \equiv f(\vec{x}) = y$. Thus we obtain

$$w' \models [\forall\vec{x}.\vec{x} \in \Omega_f \supset ([r]f(\vec{x}) = y) \equiv \gamma_f(\vec{x}, y, r)] \wedge [\forall\vec{x}.\vec{x} \notin \Omega_f \supset ([r]f(\vec{x}) = y) \equiv f(\vec{x}) = y].$$

Now construct a world $w'' \sim_{\vec{P}} w'$ such that $w''[P_i(\vec{m})] = w'[f_i(\vec{m}), \langle\rangle]$. It is easy to show (by induction) that

$$w'', r \models [\forall\vec{x}.\vec{x} \in \Omega_f \supset (f(\vec{x}) = y \equiv \gamma_f(\vec{x}, y, r)_{\vec{P}}^{\vec{F}})] \wedge [\forall\vec{x}.\vec{x} \notin \Omega_f \supset (f(\vec{x}) = y \equiv P(\vec{x}) = y)],$$

for every $f \in \mathcal{F}$. Equivalently:

$$w'', r \models \Omega_{ss} \wedge \bigwedge_i [\forall\vec{x}.\vec{x} \notin \Omega_{f_i} \supset (f_i(\vec{x}) = y) \equiv P_i(\vec{x}) = y].$$

That is, $w'', r \models \Omega_{ss} \wedge \bigwedge_i f_i \approx_\Omega P_i$.

In an analogous manner, since $w' \models \phi \wedge \varphi_r^v$ by assumption (recall that $w' \simeq_r w$), $w'', r \models (\phi \wedge \varphi_r^v)_{\vec{P}}^{\vec{F}}$. Putting this together, $w'' \models [r](\psi \wedge \bigwedge_i f_i \approx_\Omega P_i)$. Since $w'' \sim_P w'$, $w' \models \exists\vec{P}[r](\psi \wedge \bigwedge_i f_i \approx_\Omega P_i)$, or $w', r \models \exists\vec{P}[\psi \wedge \bigwedge_i f_i \approx_\Omega P_i]$. By Proposition 5.2.19, $w'_r \models \exists\vec{P}[\psi \wedge \bigwedge_i f_i \approx_\Omega P_i]$.

Given that we have $w'_r \models forget(\psi \wedge \bigwedge_i f_i \approx_\Omega P_i, \vec{P})$, we simply apply Theorem 5.2.9 to obtain $w'_r \models forget(\psi, \Delta_{\vec{P}}^{\vec{F}})_{\vec{F}}^{\vec{P}}$. ∎

We now consider the reverse direction:

Lemma 5.2.21. *Suppose* $w' \models forget(\psi, \Delta^{\vec{F}}_{\vec{P}})^{\vec{P}}_{\vec{F}} \wedge \Box\beta$. *Then there exists a world* w'' *such that* $w''_r = w'$, $w'' \simeq_r w$ *and* $w'' \models \phi \wedge \Box\beta$.

Proof: From Theorem 5.2.9, we infer that $w' \models forget(\psi \wedge \bigwedge f \approx_\Omega P, \vec{P}) \wedge \Box\beta$. That is,

$$w' \models \exists\vec{P}[(\phi \wedge \varphi^v_r)^{\vec{F}}_{\vec{P}} \wedge \Omega_{ss} \wedge \bigwedge f \approx_\Omega P].$$

Expanding our abbreviations, we have

$$w' \models \exists\vec{P}[(\phi \wedge \varphi^v_r)^{\vec{F}}_{\vec{P}} \wedge \bigwedge f(\vec{m}) = y \equiv \gamma_f(\vec{m}, y, r)^{\vec{F}}_{\vec{P}} \wedge \bigwedge \forall\vec{x}.\vec{x} \notin \Omega_f \supset (f(\vec{x}) = y \equiv P(\vec{x}) = y)].$$

Equivalently,

$$w' \models \exists\vec{P}[(\phi \wedge \varphi^v_r)^{\vec{F}}_{\vec{P}} \wedge \bigwedge \forall\vec{x}, y.\ f(\vec{x}) = y \equiv \gamma_f{}^{\vec{F}}_{\vec{P}}].$$

That is, $w' \models Prog(\phi) \wedge \Box\beta$. It is now not hard to see by way of Lemma 5.1.6 that there is indeed a world w'' such that $w''_r = w'$, $w'' \simeq_r w$ and $w'' \models \phi \wedge \Box\beta$. ∎

The proof for Theorem 5.2.18 makes use of the above lemmas and goes as follows:

Proof: Suppose $e, w \models \boldsymbol{O}(\phi \wedge \Box\beta) \wedge SF(r) = n$. We will need to show that for all w', $w' \in e^w_r$ iff $w' \models Prog(\phi) \wedge \Box\beta$.

For the if direction, suppose $w' \models Prog(\phi) \wedge \Box\beta$. By Lemma 5.2.21, there is a world w'' such that $w''_r = w'$, $w'' \simeq_r w$ and $w'' \models \phi \wedge \Box\beta$. By assumption, $w'' \in e$ and therefore $w''_r \in e^w_r$, or $w' \in e^w_r$.

Conversely, suppose $w' \in e^w_r$. That is, by construction of e^w_r, there is a world $w'' \in e$ such that $w''_r = w'$ and $w'' \simeq_r w$. By assumption, $w'' \models \phi \wedge \Box\beta$ and therefore, by Lemma 5.2.20, $w' \models Prog(\phi) \wedge \Box\beta$. ∎

Example 5.2.22. We will pursue the progression of the basic action theory from Example 5.2.11 wrt the action $move(C, roomC)$. From Theorem 5.2.18, we first identify the components of the progressed theory, which is obtained by forgetting $\Delta^{\vec{F}}_{\vec{P}}$ from $(\phi \wedge \varphi^v_r)^{\vec{F}}_{\vec{P}} \wedge \Omega_{ss}$, and replacing $\vec{P}$ with $\vec{F}$ in the end. Let us use Q as a second-order function variable for *at*.

- The initial KB, the sensing results and the instantiated successor state axioms:
  1. $\phi^{\vec{F}}_{\vec{P}} = [\forall x.\ x = C \vee x = D \supset Q(x) = roomA]$.
  2. $\varphi^{v\,\vec{F}}_{r\,\vec{P}} = \{\text{TRUE}\}$.
  3. Recall from Example 5.2.15 that the characteristic set Δ is $\{at(C)\}$. Therefore, the instantiated successor state axioms wrt Δ is equivalent to $at(C) = roomC \equiv Q(C) = roomA$.
- The atom to be forgotten is $\Delta^{\vec{F}}_{\vec{P}} = \{Q(C)\}$.

Pursue forgetting $\{Q(C)\}$ from $\{(1), (2), (3)\}$. This can be shown to be equivalent to $\{Q(D) = roomA, at(C) = roomC\}$. Finally, replace Q with *at* to obtain the new knowledge base:

$$at(C) = roomC \wedge at(D) = roomA.$$

In short, the location of C changes to $roomC$. ∎

5.2.3 Progression for Normal Actions

The locality assumption made in the previous section covers a broad range of applications, but it is still quite limited. To that end, Liu and Lakemeyer [2009] first observed that in many cases, non-local actions seldom depend on the fluents on which they have non-local effects. Put differently, the actions generally have local-effects on the fluents appearing in the *rhs* of successor state axioms. For example, in the delivery domain, moving a container of objects not only changes the location of the container but also the location of the objects in the container. This observation led them to introduce the notion of *normal actions* that captures such examples. In this section, as an extension of their work to functional fluents, we show that if the initial theory is in a so-called *semi-Horn* form (see below), then progression wrt normal actions is first-order definable and computable. It is based on an early quantifier elimination technique by Ackermann [1935].

Quantifier elimination is an active area of research where one of the main objectives is to reduce higher-order logic formulas to equivalent first-order or propositional ones [Nonnengart et al., 1999]. This work is generally motivated by the fact that automated reasoning in higher-order logic, such as second-order or fixed-point logic, is much more difficult than reasoning in predicate or propositional logic. Of course, such reductions are not always possible. Gabbay and Ohlbach [1992] consider techniques that guarantee correctness whenever reductions from second-order logic formulas to predicate logic ones exist. In the sequel, we are concerned with one particular result where a first-order equivalent formula always exists.

We say a formula ϕ is *positive* wrt a predicate P if $\neg P$ does not occur in the negation normal form of ϕ. The idea behind Ackermann's quantifier elimination result is that if one can bring a sentence to the form:

$$\exists P.\, \forall \vec{x}.\, (\neg P(\vec{x}) \vee \phi(\vec{x})) \wedge \Delta[P]$$

where ϕ is a first-order formula not mentioning the predicate P and $\Delta[P]$ is a first-order formula that is positive wrt P, then the sentence is equivalent to $\Delta[\phi]$ which denotes the result of replacing $P(\vec{x})$ everywhere in Δ with $\phi(\vec{x})$. Note that by Theorem 5.2.6, this result is also applicable to forgetting predicates from a theory satisfying the above form.

We now consider a simple case of this result for our logical framework. We first introduce the notion of a *semi-Horn* formula, which intuitively means that the formula can be brought to the form required for Ackermann's result. Then we prove a theorem regarding the elimination of function symbols from semi-Horn formulas.

Definition 5.2.23. (Semi-Horn, SNC and WSC.) We say that a fluent sentence ϕ is semi-Horn wrt a function f if the only appearance of f in ϕ is

(a) in the form $f(\vec{x}) = y \supset N(\vec{x}, y)$, where we call N a *necessary condition* of f;

(b) or, in the form $S(\vec{x}, y) \supset f(\vec{x}) = y$, where we call S a *sufficient condition* of f.

We let SNC_f denote the conjunction of all $N(\vec{x}, y)$ such that $f \supset N$ is in ϕ. We call SNC_f the *strongest necessary condition* of f wrt ϕ. We let WSC_f denote the disjunction of all $S(\vec{x}, y)$ such that $S \supset f$ is in ϕ. We call WSC_f the *weakest sufficient condition* of f wrt ϕ. ∎

Theorem 5.2.24. *Let ϕ be a sentence that is semi-Horn wrt the function f. Let ϕ' be the set of sentences in ϕ that contain no occurrences of f. Then*

$$\models \mathit{forget}(\phi, f) \equiv \phi' \wedge \forall \vec{x}, y.\ \mathrm{Wsc}_f(\vec{x}, y) \supset \mathrm{Snc}_f(\vec{x}, y).$$

Proof: Let us first note that ϕ is equivalent to

$$\phi' \wedge f(\vec{x}) = y \supset \mathrm{Snc}_f(\vec{x}, y) \wedge \mathrm{Wsc}_f(\vec{x}, y) \supset f(\vec{x}) = y$$

by assumption.

Suppose w is any world such that $w \models \mathit{forget}(\phi, f)$, that is, $w \models \exists P \forall \vec{x}, y.(\phi' \wedge P(\vec{x}) = y \supset \mathrm{Snc}_f(\vec{x}, y) \wedge \mathrm{Wsc}_f(\vec{x}, y) \supset P(\vec{x}) = y)$. Let $w' \sim_P w$ and then $w' \models \forall \vec{x}, y.(\phi' \wedge P(\vec{x}) = y \supset \mathrm{Snc}_f(\vec{x}, y) \wedge \mathrm{Wsc}_f(\vec{x}, y) \supset P(\vec{x}) = y)$. Clearly then $w' \models \phi' \wedge \forall \vec{x}, y.\ \mathrm{Wsc}_f(\vec{x}, y) \supset \mathrm{Snc}_f(\vec{x}, y)$ which does not mention P. Since $w' \sim_P w$, $w \models \phi' \wedge \forall \vec{x}, y.\ \mathrm{Wsc}_f(\vec{x}, y) \supset \mathrm{Snc}_f(\vec{x}, y)$. Therefore, $\mathit{forget}(\phi, f) \models \phi' \wedge \forall \vec{x}, y.\ \mathrm{Wsc}_f(\vec{x}, y) \supset \mathrm{Snc}_f(\vec{x}, y)$.

Conversely, suppose $w \models \phi' \wedge \forall \vec{x}, y.\ \mathrm{Wsc}_f(\vec{x}, y) \supset \mathrm{Snc}_f(\vec{x}, y)$. Let $w' \sim_P w$ such that $w'[P(\vec{m})] = n$ iff $w' \models \mathrm{Wsc}_f(\vec{m}, n)$. Then $w' \models \forall \vec{x}, y.\ \mathrm{Wsc}_f(\vec{x}, y) \equiv P(\vec{x}) = y$. Moreover, $w' \models \forall \vec{x}, y.\ P(\vec{x}) = y \supset \mathrm{Snc}_f(\vec{x}, y)$ by assumption. Since $w' \sim_P w$, $w' \models \phi'$. Then we have $w \models \exists P(\phi' \wedge \forall \vec{x}, y.\ \mathrm{Wsc}_f(\vec{x}, y) \supset P(\vec{x}) = y \wedge P(\vec{x}) = y \supset \mathrm{Snc}_f(\vec{x}, y))$. In other words, we have $w \models \exists P \phi_P^f$ which is essentially the forgetting of f from ϕ by Theorem 5.2.6. ∎

Example 5.2.25. Suppose f is a 1-ary function and let g, q and g' be 0-ary functions. Then,

- Let $\phi_1 = \forall x, y.\ (g = x \supset f(x) = y \wedge f(x) = y \supset q = y)$.

  The sentence ϕ_1 is semi-Horn wrt f. Here $g = x$ is both a sufficient condition as well as the weakest sufficient condition of f. Similarly, $q = y$ is both a necessary condition as well as the strongest necessary condition of f. Now, by Theorem 5.2.24, forgetting f from ϕ_1 is equivalent to $\forall x, y.\ g = x \supset q = y$.

- Let ϕ_2 be $\forall x, y.\ [(g = x \supset f(x) = y) \wedge (f(x) = y \supset q = y) \wedge (f(x) = y \supset g' \neq 1)]$.

  The sentence ϕ_2 is semi-Horn wrt f. The weakest sufficient condition is as above. However, both $q = y$ and $g' \neq 1$ are necessary conditions of f, and thus, the strongest necessary condition is $q = y \wedge g' \neq 1$. By Theorem 5.2.24 the forgetting of f from ϕ_2 is $\forall x, y.\ g = x \supset (q = y \wedge g' \neq 1)$.

- Let $\phi_3 = \forall x, y.\ (f(1) = y \vee f(x) = 1 \vee g = x) \wedge (f(y) = 1 \vee f(x) \neq y \vee q = y)$.

  Then ϕ_3 is *not* semi-Horn wrt f. Therefore, the forgetting technique from Theorem 5.2.24 is not applicable. ∎

The intuition behind normal actions is that if it has a non-local effect on a fluent f', then all fluents appearing in $\gamma_{f'}$ must be local-effect. To capture this property, we first adapt the notion of local-effects from Definition 5.2.10 and the simplification pursued in Proposition 5.2.12, in terms of primitive actions:

Definition 5.2.26. Let the successor state axiom for the fluent $f \in \mathcal{F}$ be of the form:

$$\Box[v] f(\vec{x}) = y \equiv \gamma_f(\vec{x}, y, v) \vee f(\vec{x}) = y \wedge \neg \exists h.\gamma_f(\vec{x}, h, v).$$

$$
\begin{aligned}
\Sigma_0 = \{\ & in(C) = boxA, \\
& in(D) \neq boxB \vee in(D) \neq boxC, \\
& at(boxA) = roomA, \\
& \forall x, y.\ in(x) = boxA \supset at(x) = roomA\}.
\end{aligned}
$$

$$
\begin{aligned}
\Sigma_{post} = \{ & \Box[v]at(x) = y \equiv \\
& \exists b.\ v = move(b, y) \wedge (x = b \vee in(x) = b) \vee \\
& at(x) = y \wedge \neg \exists b, h.\ (v = move(b, h) \wedge (x = b \vee in(x) = b)), \\
& \Box[v]in(x) = y \equiv in(x) = y \wedge v \neq remove(x, y)\}.
\end{aligned}
$$

Figure 5.3: Delivering boxes.

We say that a primitive action $A(\vec{o})$, denoted r, has *local-effects* on f if $\gamma_f(\vec{x}, y, r)$ is equivalent to a disjunction of formulas of the form

$$\vec{x} = \vec{m} \wedge y = n \wedge \mu,$$

where $\vec{m}$ and n are vectors of names contained in $\vec{o}$, and μ does not contain free variables and is called the *context formula.* We denote by LE(r) the set of all fluents on which r has local-effects. ∎

Definition 5.2.27. (Normal action.) We say that a primitive action r is *normal* if for each fluent f, all fluents appearing in $\gamma_f(\vec{x}, y, v)$ are in LE(r). ∎

In other words, normal actions always have local-effects on every fluent appearing in the *rhs* of successor state axioms. We now illustrate normal actions by adapting a previous example:

Example 5.2.28. (The office robot domain with box delivery.) We reconsider the office robot domain from Example 5.2.11, with two modifications. First, we assume that the blocks from before are found in one of many boxes, labeled $boxA, boxB \ldots, boxK$. Second, the robot may also deliver boxes to desired locations instead of, say, individual blocks.

We formalize the domain in Figure 5.3. We suppose that initially, C is in the first box, and D is not in the second box or not in the third one. The first box is located in *roomA* and so, by extension, every object in the first box is also located in *roomA*. Moreover, if the box is relocated, then so are all the objects in it. The location of an object is represented by the fluent *at*. An object may also be removed from a box. For simplicity, (yet again) we assume that actions are always executable and that actions return trivial sensing results.

To enable an illustration of normal actions, consider the primitive action $move(boxA, roomC)$, which we will denote by r. Clearly, r has local-effects on the fluent *in*. Thus, $in \in$ LE(r). Regarding the only other

fluent *at* observe that γ_{at} mentions a single fluent *in*, on which r has local-effect, and thus, r is a normal action.

Observe also that γ_{at} is not a local-effect successor state axiom since the variables of the action $move(b, y)$ do not include the free variable x appearing in the context formula of γ_{at}. ∎

The intuition behind the concept of a normal action is that we can lump fluents into two categories wrt a primitive action r: fluents $f \in \text{LE}(r)$ and fluents $f \notin \text{LE}(r)$. Now, on executing the action r, forgetting fluents from $\text{LE}(r)$ is done by adopting the methodology of the previous section. That is, only a finite number of fluent terms of each $f \in \text{LE}(r)$ are affected after r is executed and therefore forgetting these fluents is first-order definable by an application of Theorem 5.2.9. For forgetting fluents not in $\text{LE}(r)$, we make use of Theorem 5.2.24. Therefore, for the latter set of fluents we insist that the initial theory is semi-Horn wrt all $f \notin \text{LE}(r)$.

Definition 5.2.29. We say that a fluent sentence ϕ is *normal* wrt a primitive normal action r if for each $f \notin \text{LE}(r)$, ϕ is semi-Horn wrt f.

We reiterate that this constraint applies only to fluents not in $\text{LE}(r)$. The fluents in $\text{LE}(r)$ can appear in an arbitrary way in the initial theory.

Theorem 5.2.30. *Let $\Sigma = \phi \wedge \Box\beta$ be a basic action theory. Let the initial theory ϕ and the sensing result φ_r^v be normal wrt a primitive normal action r. Then the progression of Σ wrt r is first-order definable and computable.*

Proof: Let us recall from Theorem 5.1.4 that progression wrt a primitive action r is $Prog(\phi)$ together with $\Box\beta$, where $Prog(\phi)$ is

$$\exists \vec{P}.\ (\phi \wedge \varphi_r^v)_{\vec{P}}^{\vec{F}} \wedge \bigwedge \forall \vec{x}, y.\ f(\vec{x}) = y \equiv {\gamma_f}_r^{v\ \vec{F}}_{\vec{P}}.$$

We now show that under the conditions of the theorem, $Prog(\phi)$ is a first-order sentence. The idea will be to eliminate all the predicates from $\vec{P}$ based on whether the corresponding fluents are in $\text{LE}(r)$.

Below, we write $P \in \text{LE}(r)$ to mean that the second-order variable P corresponds to a function $f \in \text{LE}(r)$. We understand $P \notin \text{LE}(r)$ analogously.

case $P \notin \text{LE}(r)$:

First, by assumption, $(\phi \wedge \varphi_r^v)_{\vec{P}}^{\vec{F}}$ is semi-Horn wrt P. We now note that the instantiation of the successor state axioms in $Prog(\phi)$:

$$\forall \vec{x}, y.\ f(\vec{x}) = y \equiv (\gamma_f(\vec{x}, y, r) \vee f(\vec{x}) = y \wedge \neg\exists h.\ \gamma_f(\vec{x}, h, r))_{\vec{P}}^{\vec{F}} \tag{5.3}$$

is also semi-Horn wrt P. This can be seen as follows. Consider that (5.3) can be rewritten in a logically equivalent form ξ, where ξ is a conjunction of

1. $\forall \vec{x}, y.\ f(\vec{x}) = y \wedge \neg\gamma_f(\vec{x}, y, r)_{\vec{P}}^{\vec{F}} \supset P(\vec{x}) = y$,
2. $P(\vec{x}) = y \supset \exists h.\ \gamma_f(\vec{x}, h, r)_{\vec{P}}^{\vec{F}} \vee f(\vec{x}) = y$,
3. $\gamma_f(\vec{x}, y, r)_{\vec{P}}^{\vec{F}} \supset f(\vec{x}) = y$,
4. $\neg\gamma_f(\vec{x}, y, r)_{\vec{P}}^{\vec{F}} \wedge \exists h.\ \gamma_f(\vec{x}, h, r)_{\vec{P}}^{\vec{F}} \supset f(\vec{x}) \neq y$.

(The rewriting is straightforward: we convert (5.3), which is of the form $\alpha \equiv \beta$, to $\alpha \supset \beta \wedge \beta \supset \alpha$ and then simplify the latter.) Given that all fluents appearing in $\gamma_{f\,\vec{P}}^{\vec{F}}$ are also in LE(r) by definition, both $\gamma_f(\vec{x}, y, r)_{\vec{P}}^{\vec{F}}$ and $\gamma_f(\vec{x}, h, r)_{\vec{P}}^{\vec{F}}$ do not mention P. Then each of the items from 1 to 4 are clearly semi-Horn wrt P. Thus, ξ is semi-Horn wrt P. Therefore eliminating the predicate P from $\xi \wedge (\phi \wedge \varphi_r^v)_{\vec{P}}^{\vec{F}}$ is a first-order sentence by Theorem 5.2.24.

case $P \in \text{LE}(r)$:

Since r has local-effect on P, we obtain the argument set and then apply Theorem 5.2.18 to eliminate P. ∎

Corollary 5.2.31. *Let $\Sigma = \phi \wedge \Box\beta$ and r be as in Theorem 5.2.30. Then,*

$$\models \boldsymbol{O}(\Sigma \wedge \Box\beta) \wedge SF(r) = x \supset [r]\boldsymbol{O}(Prog(\phi) \wedge \Box\beta)$$

where $Prog(\phi)$ is first-order definable and computable.

Proof: The formal arguments follow the proof of Theorem 5.2.18 closely with a simple extra step needed to forget all fluents not in LE(r), as pursued in Theorem 5.2.30. ∎

Example 5.2.32. We pursue the progression of the basic action theory of the box delivery domain, *i.e.* Example 5.2.28, wrt $move(boxA, roomC)$. From Theorem 5.2.30, we first identify the components of the progressed theory, which is obtained by forgetting $\vec{P}$ from $(\phi \wedge \varphi_r^v)_{\vec{P}}^{\vec{F}} \wedge \bigwedge f(\vec{x}) = y \equiv \gamma_{f\,r\vec{P}}^{v\,\vec{F}}$. Let us use P as a second-order function variable for the fluent *in* and Q as a variable for *at*.

- The initial KB, the sensing results and the instantiated successor state axioms:
  1. Noting that φ_r^v is equivalent to TRUE, $(\phi \wedge \varphi_r^v)_{\vec{P}}^{\vec{F}}$ is a conjunction of
     (a) $P(C) = boxA \wedge P(D) \neq boxB \vee P(D) \neq boxC$,
     (b) $Q(boxA) = roomA \wedge \forall x.\, P(x) = boxA \supset Q(x) = roomA$.
  2. The successor state axioms are instantiated as
     (a) $at(x) = y \equiv y = roomC \wedge (x = boxA \vee P(x) = boxA)$.
     (b) $in(x) = y \equiv P(x) = y$.
- P and Q are to be forgotten from $\{(1), (2)\}$.

Begin by eliminating Q from $\{(1), (2)\}$. Mainly, observe that $\{(1a), (2a), (2b)\}$ does not mention Q. Second, by converting $\{(1b)\}$ to a semi-Horn form wrt Q, it is easy to see that forgetting Q from $\{(1), (2)\}$ is simply $\{(1a), (2a), (2b)\}$.

Next, consider eliminating P from the resultant. Since $in(x) = y \equiv P(x) = y$, the result can be shown to be equivalent to

$$in(C) = boxA \wedge (in(D) \neq boxB \vee in(D) \neq boxC) \wedge$$

$$at(x) = y \equiv y = roomC \wedge (x = boxA \vee (in(x) = boxA)),$$

which is the progression of the basic action theory wrt $move(boxA, roomC)$. ∎

5.3 Computability Results

Earlier, we proved that the progression of first-order sentences for local-effect is first-order definable. However, depending on the size of the characteristic set, it may lead to an exponential blow-up in the size of the KB because of the number of new sentences added to the KB. Similarly, we proved that the progression of finite first-order theories wrt normal actions is first-order definable, but this does not imply that it is efficiently computable. In this section, we prove that the progression of certain kinds of first-order theories, which allow us to capture disjunctive information, wrt local-effect and normal actions is not only first-order definable but also efficient (throughout this thesis, by "efficient" we mean that progression is computable in linear time), under reasonable assumptions.

Our result regarding efficient progression wrt local-effects generalizes a previous result by Liu and Lakemeyer [2009] to a language with functions. Since the notion of forgetting a primitive term is quite different from forgetting a primitive atom, there are considerable differences between the two methodologies. Our result regarding efficient progression wrt normal actions also extends a previous result by Liu and Lakemeyer [2009] to a language with functions, but here the methodologies are similar because forgetting a predicate or a function from a semi-Horn theory works in the same fashion.

5.3.1 Proper$^+$ Knowledge Bases

In the sequel we consider progression wrt syntactically normalized first-order disjunctive information called *proper$^+$ knowledge bases.* Proper$^+$ KBs were introduced in [Lakemeyer and Levesque, 2002], and in general, they correspond to a (possibly) infinite set of function-free ground clauses. We now generalize the idea behind proper$^+$ KBs to include function symbols.

Let ε denote a *ewff* by which we mean Boolean combinations of formulas of the form $t = t'$, where t and t' are either variables or names. Let c denote a *clause* by which we mean disjunctions of equalities of the form $f(\vec{t}) \circ n$, where f is any function, $\vec{t}$ contains either variables or names, n is any name, and $\circ \in \{=, \neq\}$. Let $\forall\alpha$ denote the universal closure of α. We refer to formulas of the form $\forall(\varepsilon \supset c)$ as $\forall$-clauses.

Definition 5.3.1. (Proper$^+$ KBs.) A proper$^+$ KB is any finite and satisfiable set of $\forall$-clauses. ∎

Example 5.3.2. To illustrate the expressiveness of proper$^+$ KBs, we do some examples. Let us consider two functions *at* and *in*, such that $at(x)$ gives the location of x and $in(x)$ specifies the enclosing body for the object x. Then the conjunction of the following sentences is a proper$^+$ KB:

- $in(D) \neq roomC$
- $at(C) = roomC \vee at(C) \neq roomD$
- $\forall(x = D \wedge y = C \supset at(x) = roomD \vee at(y) \neq roomD)$
- $\forall(x \neq D \wedge at(x) = hallway \supset in(x) \neq boxE)$
- $\forall(x = boxD \wedge y \neq z \wedge z = boxB \supset at(x) \neq roomC \vee at(y) \neq hallway)$. ∎

Example 5.3.3. In contrast to the above example, the following sentences are not $\forall$-clauses.

- $\forall(f(\vec{x}) = y \supset g(\vec{x}) = y)$ is not a $\forall$-clause because the clauses that appear in $\forall$-clause are of the form $f(\vec{t}) \circ n$, where $\vec{t}$ may mention either variables or names, but the value of the function n has to be a name. But in this example, the value of the two functions is the variable y.

- $\forall\vec{x}\exists y[f(y) = 2 \supset g(\vec{x}) = 2]$ is not a $\forall$-clause because y is existentially quantified. ∎

Our idea of a proper$^+$ KB can be contrasted to the original one in [Lakemeyer and Levesque, 2002], where a proper$^+$ KB is a finite set of sentences of the form

$$\forall(\varepsilon \supset P_1(\vec{t_1}) \vee \ldots \vee P_k(\vec{t_k})),$$

where ε is a ewff, P_i is a predicate and $\vec{t_i}$ is a vector containing either names or variables.[7] To disambiguate our notion of proper$^+$ KBs from the one that appeared in [Lakemeyer and Levesque, 2002], we refer to the latter as *function-free* proper$^+$ KBs.

Note that while our logical language does not include predicates, expressing predicates is straightforward, as explained in Section 4.1. But on the other hand, it is not possible to express functions with function-free proper$^+$ KBs because it is not possible to express the existence of the value of a function. For instance, suppose that we are capturing a k-ary function, say f, with a $(k+1)$-ary predicate, say P. While it is possible to express the possible values of $f(\vec{x})$ by means of say $P(\vec{x}, n_1) \vee \ldots \vee P(\vec{x}, n_k)$, we cannot express that there must be a value to $f(\vec{x})$ as a $\forall$-clause, *i.e.* $\forall\vec{x}\exists y P(\vec{x}, y)$, because here y is existentially quantified. Thus, proper$^+$ KBs *strictly generalize* function-free proper$^+$ KBs.

Before moving on, let us reiterate that by allowing an infinite domain of discourse, a $\forall$-clause such as

$$\{\forall x.\ \mathit{smallObject}(x) = 1 \supset \mathit{object}(x) = 1\}$$

is essentially equivalent to an infinite set of primitive clauses:

$$\mathit{smallObject}(n_1) = 1 \supset \mathit{object}(n_1) = 1, \mathit{smallObject}(n_2) = 1 \supset \mathit{object}(n_2) = 1, \ldots$$

So reasoning with proper$^+$ KBs is not trivial. In fact, deductive reasoning with function-free proper$^+$ KBs is already undecidable [Lakemeyer and Levesque, 2002]. It is for this reason that we will also be proposing a query evaluation mechanism for proper$^+$ KBs for a restricted class of queries. But for now, we consider procedures for progressing proper$^+$ KBs.

5.3.2 Efficient Progression for Local-Effect Actions

Before proposing a computability procedure for the progression of proper$^+$ KBs, we cover a preliminary result regarding the conditions under which we are entitled to inferring the validity of an existential from the validity of a finite number of substitution instances. More precisely, we prove that if we are able to bring a sentence, which is existentially quantified wrt a variable x, to what we call the *definitional form* wrt x then the sentence is equivalent to another sentence without the quantifier.

[7] Strictly speaking, they do not consider standard names. Instead they consider an infinite set of distinct constants, where equality is interpreted as identity and behaves as an equivalence relation for a substitution of arguments (see Section 3.1.1). These are essentially what we mean by standard names [Levesque, 1998].

Definition 5.3.4. (Definitional wrt x.) We say that a quantifier-free fluent formula φ is *definitional wrt the variable x* if it is of the form $(\alpha \vee x \circ_1 n_1 \vee \ldots \vee x \circ_k n_k)$, where $\circ_i \in \{=, \neq\}$, such that α does not mention x.

We say that a fluent formula ϕ is *definitional wrt x* if it of the form $(\varphi_1 \wedge \ldots \wedge \varphi_k)$, where φ_i is a quantifier-free formula that is definitional wrt x. ∎

Given a fluent formula ϕ that is definitional wrt x, let $H_x(\phi)$ denote all the names such that $x \circ n$ appears in ϕ. Let $H_x^+(\phi)$ denote the union of the names in $H_x(\phi)$ plus an arbitrary extra one.

Proposition 5.3.5. *Let α denote the formula $x \circ_1 n_1 \vee \ldots \vee x \circ_k n_k$. Let n^* be any name apart from the ones in $\{n_1, \ldots, n_k\}$. Then either $\models \alpha^x_{n^*}$ or $\models \neg\alpha^x_{n^*}$.*

Proof: Proof by induction on α. The base case is a single literal of the form $x \circ n$. We consider only the base case here, since it is straightforward to prove the result for a disjunction of literals using the base case.

Suppose $x = n$. Then, because n^* and n are distinct we have $w \models \neg\alpha^x_{n^*}$ for any w. Now, suppose $x \neq n$. Then $w \models \alpha^x_{n^*}$ for any w since n^* and n are distinct and so $n^* \neq n$ is true at all worlds. ∎

Corollary 5.3.6. *Suppose φ is the quantifier-free fluent formula $(\alpha \vee x \circ_1 n_1 \vee \ldots \vee x \circ n_k)$, whose free variables are in $\vec{y}$ and where α does not mention x. Suppose n^* is any name apart from the ones in $H_x(\varphi)$. Then either $\models \forall\vec{y}\, \varphi^x_{n^*}$ or $\models \forall\vec{y}[\varphi^x_{n^*} \equiv \alpha]$.*

Proof: By Proposition 5.3.5, either $\models (\bigvee x \circ_i n_i)^x_{n^*}$ or $\models \neg(\bigvee x \circ_i n_i)^x_{n^*}$. Clearly, if the former then $\models \forall\vec{y}\varphi^x_{n^*}$ and otherwise, $\forall\vec{y}\varphi^x_{n^*}$ is equivalent to $\forall\vec{y}\alpha$. ∎

Corollary 5.3.7. *Suppose ϕ is a fluent formula that is definitional wrt x, and whose free variables are in $\vec{y}$. Then $\forall\vec{y}.\ \phi^x_n$ is logically equivalent to $\forall\vec{y}.\ \phi^x_m$ for every $n, m \notin H_x(\phi)$.*

Proof: By Definition 5.3.4, suppose ϕ is of the form $\{\varphi_1, \ldots, \varphi_k\}$ where φ_i is $\alpha_i \vee x \circ_1 n_1 \vee \ldots \vee x \circ_k n_k$. By Corollary 5.3.6, $\forall\vec{y}.\ \varphi^x_{n^*}$ is either valid or equivalent to $\forall\vec{y}.\ \alpha$ regardless of whether n^* is n or m since they are both not the names in $H_x(\phi)$. Clearly, then, $\forall\vec{y}.\ \bigwedge(\varphi_i)^x_n$ is equivalent to $\forall\vec{y}.\ \bigwedge(\varphi_i)^x_m$. ∎

Theorem 5.3.8. *Suppose ϕ is a fluent formula, whose free variables are in $\vec{y}$, that is definitional wrt x. Then $\exists x\forall\vec{y}\phi$ is logically equivalent to $\bigvee_{n \in H_x^+(\phi)} \forall\vec{y}\phi^x_n$.*

Proof: By Corollary 5.3.7, $\forall\phi^x_n$ is logically equivalent to $\forall\phi^x_m$ for every $n, m \notin H_x(\phi)$. In other words, if n^* is the name appearing in $H_x^+(\phi)$ but not in $H_x(\phi)$ then $\forall\phi^x_{n^*}$ is logically equivalent to $\forall\phi^x_m$ for every $m \notin H_x^+(\phi)$.

Suppose $H_x^+(\phi) = \{n_1, \ldots, n_k, n^*\}$. Now, by definition $w \models \exists x\forall\vec{y}\phi$ iff $w \models \forall\phi^x_m$ for some name m iff (by the above argument) $w \models \forall\phi^x_{n_1}$ or $w \models \forall\phi^x_{n_2}$ or … or $w \models \forall\phi^x_{n_k}$ or $w \models \forall\phi^x_{n^*}$. Thus, it follows that $w \models \exists x\forall\phi$ iff $w \models \bigvee_{n \in H_x^+(\phi)} \forall\phi^x_n$. ∎

We are now ready to consider the progression of proper$^+$ KBs wrt local-effect theories. We first formalize a property called *irrelevance* and discuss how this property can be readily identified in proper$^+$ KBs in a certain normal form. Converting an arbitrary proper$^+$ KB to the normal form is also efficiently computable.

Definition 5.3.9. (Irrelevance.) Let ϕ be a sentence and $f(\vec{m})$ a primitive term. We say that $f(\vec{m})$ is irrelevant to ϕ if $\models forget(\phi, f(\vec{m})) \equiv \phi$. ∎

Proposition 5.3.10. *Let $\phi = \forall(\varepsilon \supset c)$ be a $\forall$-clause and $f(\vec{m})$ a primitive term. Suppose that for any $f(\vec{t})$ appearing in c, $\forall(\varepsilon \wedge \vec{t} = \vec{m})$ is unsatisfiable. Then $f(\vec{m})$ is irrelevant to ϕ.*

Proof: For ease of exposition, let us suppose that there is only a single appearance of f in c. The argument if there are k appearances of f in c is straightforward but tedious. So suppose $f(\vec{t}) \circ n$, where $\circ \in \{=, \neq\}$, appears in c and $\varepsilon \wedge \vec{t} = \vec{m}$ is unsatisfiable. Let the free variables in ϕ be in $\vec{y}$. So let us write $\phi = \forall\vec{y}(\varepsilon \supset c' \vee f(\vec{t}) \circ n)$. We show that $forget(\phi, f(\vec{m}))$ is equivalent to ϕ.

By Theorem 5.2.4, $forget(\phi, f(\vec{m}))$ is logically equivalent to $\exists x\phi[f(\vec{m}) = x]$ assuming that x is a fresh variable not appearing in ϕ. Now $w \models \exists x\phi[f(\vec{m}) = x]$

iff $w \models \exists x\forall\vec{y}[\varepsilon \supset c' \vee (\vec{t} = \vec{m} \wedge n = x) \vee (\vec{t} \neq \vec{m} \wedge f(\vec{t}) \circ n)]$ by Definition 5.2.2

iff $w \models \exists x\forall\vec{y}[\neg\varepsilon \vee c' \vee (\vec{t} = \vec{m} \wedge n = x) \vee (\vec{t} \neq \vec{m} \wedge f(\vec{t}) \circ n)]$

iff $w \models \exists x\forall\vec{y}[\neg\varepsilon \vee c' \vee (\vec{t} \neq \vec{m} \wedge f(\vec{t}) \circ n)]$ because $\varepsilon \wedge \vec{t} = \vec{m}$ is unsatisfiable and this implies that $[\neg\varepsilon \vee (\vec{t} = \vec{m} \wedge n = x)]$ is equivalent to $\neg\varepsilon$

iff $w \models \exists x\forall\vec{y}[\neg\varepsilon \vee f(\vec{t}) \circ n \vee c']$ because $\varepsilon \wedge \vec{t} = \vec{m}$ is unsatisfiable, and this implies that $[\neg\varepsilon \vee (\vec{t} \neq \vec{m} \wedge f(\vec{t}) \circ n)]$ is equivalent to $\neg\varepsilon \vee f(\vec{t}) \circ n$

iff $w \models \phi$.

So for any world w, $w \models \phi$ iff $w \models forget(\phi, f(\vec{m}))$, which means that $f(\vec{m})$ is irrelevant to ϕ. ∎

Definition 5.3.11. (Normal form.) Let ϕ be a proper$^+$ KB and $f(\vec{m})$ a primitive term. We say that ϕ is in normal form wrt $f(\vec{m})$ if for any $\forall(\varepsilon \supset c) \in \phi$, and for any $f(\vec{t})$ appearing in c, either $\vec{t}$ is $\vec{m}$ or $\forall(\varepsilon \wedge \vec{t} = \vec{m})$ is unsatisfiable. ∎

Example 5.3.12. The following $\forall$-clauses are in normal form wrt $in(C)$:

- $\forall(x \neq roomC \supset in(C) \neq x)$ because for the appearance of *in* in the $\forall$-clause, its argument is indeed C;
- $\forall(x = roomC \wedge y \neq C \supset in(y) = x)$ because for the appearance of *in* in the $\forall$-clause, $y = C$ is trivially unsatisfiable with the ewff $x = roomC \wedge y \neq C$ at the head of the $\forall$-clause. ∎

Example 5.3.13. The $\forall$-clause $\forall(in(x) = roomC)$ is not in normal form wrt $in(C)$ because the head of the $\forall$-clause is empty, *i.e.* is equivalent to TRUE, and $\forall(x = C \wedge \text{TRUE})$ is *not* unsatisfiable.

Interestingly, we can equivalently write this $\forall$-clause as the conjunction of the following two $\forall$-clauses:

- $in(C) = roomC$,
- $\forall(x \neq C \supset in(C) = roomC)$.

which, in fact, is easily seen to be in normal form wrt $in(C)$. ∎

In Example 5.3.13, we were able to convert a $\forall$-clause to an equivalent form which was in normal form wrt the considered primitive term. It turns out that every $\forall$-clause, and by extension every proper$^+$ KB, can be converted into an equivalent one which is in normal form wrt a primitive term by means of this idea. We prove this formally now.

Proposition 5.3.14. *Let $f(\vec{m})$ be a primitive term. Then every proper$^+$ KB ϕ can be converted into an equivalent one which is in normal form wrt $f(\vec{m})$. This procedure has the time complexity $O(n + 2^k m)$, where n is the size of ϕ, m is the size of the $\forall$-clauses in ϕ where f appears, and k is the maximum number of appearances of f in a $\forall$-clause in ϕ.*

Proof: Let $\varphi = \forall(\varepsilon \supset c)$ be a $\forall$-clause. Let $f(\vec{t_1}) = n_1, \ldots, f(\vec{t_k}) = n_k$ be all the occurrences of f in φ. Now, define $\Theta = \{\bigwedge_{i=1}^{k} \vec{t_1} \circ_i \vec{m} \mid \circ_i \in \{=, \neq\}\}$. Let $\theta \in \Theta$. We let $c[\theta]$ denote c with each $f(\vec{t_i}) = n_i$, $1 \leq i \leq k$, replaced by $f(\vec{m}) = n_i$ if θ contains $\vec{t_i} = \vec{m}$. We use $\varphi[\theta]$ to denote $\forall(\varepsilon \wedge \theta \supset c[\theta])$. It is easy to verify that φ is logically equivalent to the theory $\{\varphi[\theta] \mid \theta \in \Theta\}$. We denote the latter theory as NF$(\varphi, f(\vec{m}))$. Clearly, φ is in normal form wrt $f(\vec{m})$ because for every appearance of $f(\vec{t_i})$ in c either $\vec{t_i} = \vec{m}$ (which denotes the step involving a replacement) or θ denotes $\vec{t_i} \neq \vec{m}$ which means that $\varepsilon \wedge \theta \wedge \vec{t_i} = \vec{m}$ is unsatisfiable. Given a proper$^+$ KB ϕ, convert it to the union of NF$(\varphi, f(\vec{m}))$, where $\varphi \in \phi$ is a $\forall$-clause.

The time of the procedure is calculated as follows. For each $\forall$-clause φ where f appears, and for each $f(\vec{t}) = n$ appearing in φ, the size of the theory $\{\varphi[\theta] \mid \theta \in \{\vec{t} = \vec{m}, \vec{t} \neq \vec{m}\}\}$ is twice that of the original. Since there are m such $\forall$-clauses and k mentions of f in them, the size of the resulting generated theory is $O(2^k m)$. ∎

Proposition 5.3.15. *Suppose ϕ is a proper$^+$ KB that is in normal form wrt $f(\vec{m})$. Suppose x is a variable not mentioned in ϕ. Let ϕ' denote the formula obtained by replacing every occurrence of $f(\vec{m})$ in ϕ with x. Then $\models forget(\phi, f(\vec{m})) \equiv \exists x \phi'$.*

Proof: For the sake of exposition, suppose that ϕ is a single $\forall$-clause $\forall(\varepsilon \supset c)$ and f appears once in c. The general case is tedious but otherwise holds without any changes. By Theorem 5.2.4, $forget(\phi, f(\vec{m}))$ is equivalent to $\exists x \phi[f(\vec{m}) = x]$. Now, by assumption, ϕ is in normal form, which means that for every $f(\vec{t}) = n$ appearing in c, either $\vec{t}$ is $\vec{m}$ or $\varepsilon \wedge \vec{t} \neq \vec{m}$ is unsatisfiable.

- Suppose $\vec{t}$ is $\vec{m}$. Then $\phi[f(\vec{m}) = x]$ is formulated as $\forall(\varepsilon \supset c' \vee (\vec{m} = \vec{m} \wedge n = x) \vee (\vec{m} \neq \vec{m} \wedge f(\vec{m}) = n))$ which means $\exists x \phi[f(\vec{m}) = x]$ is equivalent to $\exists x \forall(\varepsilon \supset c' \vee n = x)$. As stated in the proposition, $\forall(\varepsilon \supset c' \vee n = x)$ denotes the formula obtained by replacing $f(\vec{m})$ in ϕ with x.

- Suppose $\vec{t} = \vec{m} \wedge \varepsilon$ is unsatisfiable. Suppose the free variables of ϕ are in $\vec{y}$. Then $\phi[f(\vec{m}) = x]$ is formulated as $\forall \vec{y}[\varepsilon \supset c' \vee (\vec{m} = \vec{t} \wedge n = x) \vee (\vec{m} \neq \vec{t} \wedge f(\vec{t}) = n)]$. So $w \models \exists x \phi[f(\vec{m}) = x]$

  iff $w \models \exists x \forall \vec{y}[\neg\varepsilon \vee c' \vee (\vec{m} \neq \vec{t} \wedge f(\vec{t}) = n)]$ because $\varepsilon \wedge \vec{t} = \vec{m}$ is unsatisfiable and this implies that $[\neg\varepsilon \vee (\vec{t} = \vec{m} \wedge n = x)]$ is equivalent to $\neg\varepsilon$

  iff $w \models \exists x \forall \vec{y}[\neg\varepsilon \vee c' \vee f(\vec{t}) = n]$ because $\varepsilon \wedge \vec{t} = \vec{m}$ is unsatisfiable and this implies that $[\neg\varepsilon \vee (\vec{t} \neq \vec{m} \wedge f(\vec{t}) = n)]$ is equivalent to $\neg\varepsilon \vee f(\vec{t}) = n$

iff $w \models \exists x\phi$. That is, as the proposition implies since $f(\vec{m})$ does not appear in ϕ, we leave it unaltered. ∎

Corollary 5.3.16. *Suppose ϕ, $f(\vec{m})$, x and ϕ' are as above. Then $\models forget(\phi, f(\vec{m})) \equiv \bigvee_{n \in H_x^+(\phi')} {\phi'}_n^x$.*

Proof: By Proposition 5.3.15, $forget(\phi, f(\vec{m}))$ is equivalent to $\exists x\phi'$. In particular, note that after converting a $\forall$-clause $\varphi \in \phi$ to normal form wrt $f(\vec{m})$ and replacing the appearances of $f(\vec{m})$ with x, it is now of the form $\forall(\varepsilon \supset c \vee x \circ_1 n_1 \vee \ldots \vee x \circ_k n_k)$ where $\circ_i \in \{=, \neq\}$. Equivalently, we have $\forall(\alpha \vee \bigvee x \circ n)$ where α is a the quantifier-free formula $\neg\varepsilon \vee c$. In other words, φ is essentially a universally closed fluent formula that is definitional wrt x, and thus, ϕ is also definitional wrt x. We can now apply Theorem 5.3.8. ∎

Theorem 5.3.17. *Let ϕ, $f(\vec{m})$, x and ϕ' be as above. Then the result of forgetting $f(\vec{m})$ from ϕ is definable as a proper$^+$ KB. This can be done in $O(l(n + 2^k m))$ where n, k and m are as in Proposition 5.3.14, and l is the number of elements in $H_x^+(\phi')$.*

Proof: Let $\phi(\vec{y}) = \forall\vec{y}[\bigwedge_i(\varepsilon_i \supset c_i)]$. Now, first convert $\phi(\vec{y})$ to normal form wrt $f(\vec{m})$ and this results in a theory of the size $O(n + 2^k m)$ by Proposition 5.3.14. Then convert it to $\phi'(\vec{y}, x)$, where $\phi'(\vec{y}, x)$ has x as the free variable. By Theorem 5.3.8, forgetting $f(\vec{m})$ from $\phi(\vec{y})$ is equivalent to $\bigvee_{n \in H_x^+(\phi')} \forall\vec{y}.\phi(\vec{y})$. It is easy to see that $\bigvee_{n \in H_x^+(\phi')} \forall\vec{y}.\phi(\vec{y})$ is equivalent to

$$\forall\vec{y}_1, \ldots, \vec{y}_l[\phi'(\vec{y}_1)_{n_1}^x \vee \ldots \vee \phi'(\vec{y}_l)_{n_l}^x]$$

where $H_x^+(\phi') = \{n_1, \ldots, n_l\}$ and $\vec{y}_i$ does not share variables with $\vec{y}_j$ for $j \neq i$. The size of this theory is l times the size of ϕ in normal form. ∎

In the above theorem it is reasonable to assume that l, m and k are in $O(1)$ and thus, forgetting a finite number of atoms can be done in $O(n)$ time.

In order to present a result about efficient progression, we need to clarify one last step. Recall from Theorem 5.2.18 that the progressed theory also includes as a conjunct the instantiation of the successor state axioms wrt the characteristic set, which we denoted by Ω_{ss}. Now, for Ω_{ss} to be definable as a set of $\forall$-clauses, we will only need the following assumption.

Definition 5.3.18. A basic action theory is said to *essentially quantifier-free* if for each primitive action r and fluent f, $\gamma_f(\vec{x}, y, r)$, $\exists h\gamma_f(\vec{x}, h, r)$ and φ_r^v can be simplified to quantifier-free formulas. ∎

Example 5.3.19. The office robot basic action theory from Example 5.2.11 is essentially quantifier-free. For instance, looking at Example 5.2.22, we observe that the instantiation of the successor state axioms wrt the action $move(C, roomC)$ is equivalent to $\{[move(C, roomC)]at(C) = roomC \equiv at(C) = roomA\}$. ∎

In general, if $\gamma_f(\vec{x}, y, v)$ is a disjunction of formulas of the form $\exists\vec{u}.[v = A(\vec{z}) \wedge \mu(\vec{x}, \vec{u})]$ where $\vec{z}$ contains $\vec{u}$ and the context formula μ is quantifier-free, then the successor state axiom is essentially quantifier free.

Proposition 5.3.20. *Suppose a basic action theory is essentially quantifier-free. Then the instantiation of the successor state axioms wrt a primitive action is definable as a set of $\forall$-clauses.*

Proof: Recall the general syntactic form of a successor state axiom:

$$\Box[v]f(\vec{x}) = y \equiv \gamma_f(\vec{x}, y, v) \vee f(\vec{x}) = y \wedge \neg\exists h\gamma_f(\vec{x}, h, v)$$

Given a primitive action r, by assumption $\gamma_f(\vec{x}, y, r)$ and $\exists h\gamma_f(\vec{x}, h, r)$ simplify to quantifier-free formulas. In other words, the progressed theory includes one of the following sentences:

1. $\forall(f(\vec{x}) = y \equiv \gamma_f(\vec{x}, y, r)^{\vec{F}}_{\vec{P}})$, where $\gamma_f(\vec{x}, y, r)$ is quantifier-free; or
2. $\forall(f(\vec{x}) = y \equiv P(\vec{x}) = y \wedge \neg(\exists h\gamma_f(\vec{x}, h, r)^{\vec{F}}_{\vec{P}}))$, where $\exists h\gamma_f(\vec{x}, h, r)$ is quantifier-free;

which are both definable as a set of $\forall$-clauses. ∎

We now state the main result regarding efficient progression.

Theorem 5.3.21. *Suppose a basic action theory Σ is local-effect and essentially quantifier-free, and Σ_0 is a proper$^+$ KB. Then the progression of Σ wrt any primitive action is definable as a proper$^+$ KB and can be efficiently computed.*

Proof: By assumption, the initial theory, the sensing results and the instantiated successor state axioms are a set of $\forall$-clauses. Now, one needs to only forget the finite number of primitive terms in the characteristic set from the above set of $\forall$-clauses, which we argued in Theorem 5.3.17 to be efficient and definable as a proper$^+$ KB. Therefore, we are done. ∎

Recall our discussion earlier that under reasonable assumptions, forgetting a finite set of primitive terms from a proper$^+$ KB can be done in $O(n)$ time. Progression can now iterate.

Example 5.3.22. We investigated the progression of the basic action theory of the office domain robot, *i.e.* Example 5.2.11, wrt $move(C, roomC)$ in Example 5.2.22. To demonstrate the methodologies from this section, we reconsider that problem here. Foremost, observe that, as required, the initial KB of the basic action theory is definable as a proper$^+$ KB.

From Theorem 5.2.18, we first identify the components of the progressed theory, which is obtained by forgetting $\Delta^{\vec{F}}_{\vec{P}}$ from $(\phi \wedge \varphi^v_r)^{\vec{F}}_{\vec{P}} \wedge \Omega_{ss}$ and replacing $\vec{P}$ with $\vec{F}$ in the resultant. Let us use Q as a second-order variable for *at*.

- The initial KB, the sensing results and the instantiated successor state axioms:
  1. $\phi^{\vec{F}}_{\vec{P}} = \forall[x = C \vee x = D \supset Q(x) = roomA]$.
  2. $\varphi^{v\,\vec{F}}_{r\,\vec{P}} = \{\text{TRUE}\}$.
  3. The successor state axioms is instantiated as
     (a) $at(C) = roomC \supset Q(C) = roomA$,
     (b) $Q(C) = roomA \supset at(C) = roomC$,
- $\Delta^{\vec{F}}_{\vec{P}} = \{Q(C)\}$ is to be forgotten.

We begin by forgetting $Q(C)$ from $\{(1),(2),(3)\}$. We first convert $\{(1),(2),(3)\}$ to normal form wrt $Q(C)$. Then

- $\{(1)\}$ is converted to a conjunction of
  i. $\forall[(x = C \vee x = D) \wedge x = C \supset Q(C) = roomA]$
  ii. $\forall[(x = C \vee x = D) \wedge x \neq C \supset Q(x) = roomA]$.
- $\{(3a)\}$ is converted to a conjunction of
  i. $at(C) = roomC \wedge C = C \supset Q(C) = roomA$, and
  ii. $at(C) = roomC \wedge C \neq C \supset Q(C) = roomA$.

  Observe that $\{(3a.i)\}$ simplifies to $\{(3a)\}$ itself, and $\{(3a.ii)\}$ is equivalent to TRUE. Therefore, note that $\{(3a.i),(3a.ii)\}$ simplifies to $\{(3a)\}$ itself.
- In an analogous manner, on converting $\{(3b)\}$ to the normal form, it also simplifies to $\{(3b)\}$.

Let us denote this conversion by ϕ'. If u is a fresh variable not appearing in ϕ', then forgetting $Q(C)$ from ϕ' is equivalent to $\exists u\phi'[Q(C) = u]$ by Theorem 5.2.4. So let us now obtain $\phi'[Q(C) = u]$, which amounts to replacing every occurrence of $Q(C)$ with the variable u:

$$[Q(D) = roomA, x = C \supset u = roomA, at(C) = roomC \equiv u = roomA]$$

that is quantified from the outside for all variables except u, which is a free variable in $\phi'[Q(C) = u]$. Note that $H_u(\phi'[Q(C) = u]) = \{roomA\}$. Let $H_u^+(\phi'[Q(C) = u]) = \{roomA, roomC\}$. Then, by using Theorem 5.3.8, $\exists u.\phi'[Q(C) = u]$ is equivalent to

$\phi'[Q(C) = u]^u_{roomA} \vee \phi'[Q(C) = u]^u_{roomC}$ which is equivalent to

$\{Q(D) = roomA, at(C) = roomC\}$.

On substituting Q with at, the progressed KB is $\{at(C) = roomC, at(D) = roomA\}$. ∎

5.3.3 Efficient Progression for Normal Actions

Recall from Theorem 5.2.30 that progression wrt normal actions involves forgetting fluents on which the actions have local-effects and forgetting fluents on which the actions have non-local effects. As we shall shortly see, results from the previous section will enable forgetting the former set of fluents in proper$^+$ KBs. Then, provided a proper$^+$ KB is semi-Horn, an efficient resolution step can be proposed for forgetting the latter set of fluents.

Definition 5.3.23. ($\forall$-resolvents.) Let $\varphi_1 = \forall(\varepsilon_1 \supset c_1 \vee f(\vec{x}) = y)$ and $\varphi_2 = \forall(\varepsilon_2 \supset c_2 \vee f(\vec{x}) \neq y)$ be two $\forall$-clauses. Without any loss of generality, we assume that φ_1 and φ_2 do not share variables other than those contained in y and $\vec{x}$. We call the $\forall$-clause $\forall(\varepsilon_1 \wedge \varepsilon_2 \supset c_1 \vee c_2)$ the $\forall$-resolvent of the two input clauses wrt $f(\vec{x})$. ∎

Theorem 5.3.24. *Let ϕ be a proper$^+$ KB that is semi-Horn wrt f. Then the result of forgetting f in ϕ is definable as a proper$^+$ KB and can be computed in $O(n + m^2)$ time, where n is the size of ϕ and m is the size of the $\forall$-clauses in ϕ that mention f.*

Proof: We compute all $\forall$-resolvents wrt $f(\vec{x})$ and remove the $\forall$-clauses that mention f. The number of newly generated $\forall$-clauses is m^2. Then remove all $\forall$-clauses that mention f. This procedure results in a proper$^+$ KB which, by Theorem 5.2.24, is the result of forgetting f from ϕ. ∎

Example 5.3.25. To consider a simple example where $\forall$-resolvents are computed, let ϕ be the conjunction of the following $\forall$-clauses:

- $at(boxA) = roomA \supset status(roomA) = closed$,
- $status(roomC) = open \supset at(boxA) = roomA$.

Suppose we want to forget *at*. We can equivalently write ϕ as

- $\forall(x = boxA \wedge y = roomA \supset at(x) \neq y \vee status(roomA) = closed) \wedge$
- $\forall(x = boxA \wedge y = roomA \supset status(roomC) \neq open \vee at(x) = y)$.

Then we obtain the following generated $\forall$-resolvent φ.

- $\forall(x = boxA \wedge y = roomA \supset status(roomA) = closed \vee status(roomC) \neq open)$.

Now, forgetting *at* is obtained by considering ϕ with the generated $\forall$-resolvents and removing all those that mention the function. Clearly, this leads to φ which is the result of forgetting *at*. ∎

Under the reasonable assumption that m is $O(1)$, forgetting functions can be done in $O(n)$ time.

With this in hand, we present a computability result for progression based on Theorem 5.2.30. Since the progressed theory includes the instantiation of the successor state axioms, our assumption about essentially quantifier-free basic action theories from Definition 5.3.18 is also necessary here for the very same reasons.

Theorem 5.3.26. *Suppose that Σ is a basic action theory that is essentially quantifier free, r is a primitive normal action, φ_r^v is normal wrt r and Σ_0 is a proper$^+$ KB that is normal wrt r. Then the progression of Σ_0 wrt r is definable as a proper$^+$ KB and can be computed efficiently.*

Proof: By assumption and by Proposition 5.3.20, it is given that the conjunction of the initial theory, the sensing results and the instantiated successor state axioms are a set of $\forall$-clauses. First, forget the functions on which r has non-local effects. This can be computed efficiently and is definable as a proper$^+$ KB by way of Theorem 5.3.24, provided that the initial knowledge base is normal wrt r which it is by assumption. Next, obtain the characteristic set and forget these terms, which can computed efficiently and is definable as a proper$^+$ KB by way of Theorem 5.3.21. ∎

Example 5.3.27. To illustrate the computability procedure from this section, let us reconsider the progression of the basic action theory from Example 5.2.28 wrt $move(boxA, roomC)$ that we pursued earlier in Example 5.2.32. From Theorem 5.2.30, we first identify the components of the progressed theory which is obtained by eliminating $\vec{P}$ from $(\phi \wedge \varphi_r^v)_{\vec{P}}^{\vec{F}} \wedge \bigwedge f(\vec{x}) = y \equiv \gamma_{f_r}^{v\vec{F}}{}_{\vec{P}}$. Let us use P as a second-order function variable for *in* and Q as a second-order variable for *at*.

- The initial KB, the sensing results and the instantiated successor state axioms:
  1. $(\phi \wedge \varphi_r^v)_{\vec{P}}^{\vec{F}}$ is a conjunction of
     (a) $P(C) = boxA \wedge (P(D) \neq boxB \vee P(D) \neq boxC)$,
     (b) $Q(boxA) = roomA$
     (c) $\forall(P(x) = boxA \supset Q(x) = roomA)$.
  2. The instantiated successor state axioms are enumerated in Example 5.2.32.
- $\{P, Q\}$ is to be eliminated from $\{(1), (2)\}$.

Now, observe that $move(boxA, roomC)$ has a non-local effect on Q. Since (2) does not mention Q, it is easy to see that $\{(1), (2)\}$ is semi-Horn wrt Q. This allows us to eliminate Q from $\{(1), (2)\}$ by means of Theorem 5.2.24. So, now, consider that

1(b). $\forall(x = boxA \wedge y = roomA \supset Q(x) = y)$,

1(c). $\forall(y = roomA \supset P(x) \neq boxA \vee Q(x) = y)$

do not have any $\forall$-resolvents between them. Moreover, $\{(1a), (2)\}$ do not mention Q. Therefore Q is eliminated by simply removing all $\forall$-clauses mentioning Q from $\{(1), (2)\}$ which basically results in $\{(1a), 2\}$.

The second step involves eliminating P. The action $move(boxA, roomC)$ has a local-effect on P and so by eliminating P from $\{(1a), 2\}$ we obtain

$$in(C) = boxA \wedge (in(D) \neq boxB \vee in(D) \neq boxB) \wedge$$

$$\forall[at(x) = roomC \equiv x = boxA \vee in(x) = boxA]$$

which is the progression of the basic action theory wrt $move(boxA, roomC)$. ∎

5.4 Progression for Range-Restricted Theories

Observe that the simple robot domain from Example 5.2.11 is neither local-effect nor is the action of moving forward a normal one. To see this, let us recap the successor state axiom for the fluent *distance*:

$$\Box[v]distance = x \equiv$$

$$v = forward \wedge distance = x + 1 \vee$$

$$v \neq forward \wedge distance = x.$$

Now, the action of moving forward does not contain arguments while the context formula does indeed mention a variable, and therefore, the successor state axiom does not classify as local-effect. Moreover, the action has a non-local effect on the fluent *distance* and yet, the fluent itself is mentioned in the context formula and therefore, *forward* is not a normal action.

The argument is similar for actions such as an exploding bomb which destroys everything in the vicinity.

Example 5.4.1. (Exploding bomb.) Consider the following formulation of a successor state axiom for an exploding bomb:

$$\Box[v]status(x) = y \equiv$$

$$v = explode \wedge \forall x.\, near(bomb, x) = 1 \wedge y = destroyed \wedge status(bomb) = active \vee$$

$$status(x) = y \wedge v \neq explode.$$

That is, we assume that the fluent *status* represents the status of the objects, in the sense of whether it is destroyed, and the status of the bomb, in the sense of whether is active. This successor state axiom essentially states that every object near a bomb is destroyed once the bomb explodes. ∎

Observe that the exploding action is clearly non-local since it does not have any arguments while the context formula does mention the variables x and y. The successor state axiom is also not normal wrt *explode*, mainly because the fluent *status* is mentioned in the context formula.

In order to handle such domains, in this section we consider *range-restricted* basic action theories. Range-restricted basic action theories were first introduced by Vassos et al. [2009]. The idea is to capture domains that involve actions that may not be local-effect or normal, but whose effects are "bounded" in a certain sense. Vassos et al. consider first-order progression and computability for range-restricted basic action theories for a certain kind of first-order initial theory called a *database of possible closures* (DBPC). Roughly speaking, a DBPC corresponds to a finite consistent set of clauses. Part of the reason why a restriction on the initial theory is needed is because one computes a (necessarily) finite set of fluent terms that are affected when such non-local actions are performed by means of information that is available in the initial KB. However, the progression account is rather involved since Vassos et al. need to essentially ensure that the progression of a DBPC is also definable as a DBPC.

In this sequel, we instead consider the progression of the expressive proper$^+$ KBs wrt range-restricted theories. Moreover, in contrast to Vassos et al., progression is formulated in terms of forgetting: a much simpler account. But unlike Vassos et al., we will make a form of completeness assumption regarding the context formulas.

5.4.1 Just-in-Time Formulas

Since the locality assumption wrt action theories guarantees the first-order definability, non-local actions that have global effects such as an exploding bomb which destroys everything in the vicinity are believed to be the main reason why progression is second-order in general. Range-restricted theories, on the other hand, capture those cases where a finite number of objects are affected in a non-local way. So we first discuss a preliminary concept called *just-in-time* formulas that allow us to *bound* the number of affected objects. More

precisely, given a formula with free variables, we are interested in cases where the initial theory entails only a finite number of instances of the formula.

Given a proper$^+$ KB ϕ, we define $gnd(\phi)$ as

$$\{c\theta \mid \forall(e \supset c) \in \phi, \models e\theta \text{ where } \theta \text{ is a substitution of variables with names}\}.$$

Moreover, given a set of names H, we write $gnd(\phi)|H$ to mean that the substitutions are restricted to the names from H. We write H_k^+ to mean the names in H, plus k (arbitrary) names not appearing in H.

Definition 5.4.2. (Just-in-time formulas). Suppose ϕ is a proper$^+$ KB and let H be the names in ϕ. We say that a fluent formula $\alpha(\vec{x})$, whose free variables are in $\vec{x}$, is *just-in-time* (JIT) wrt ϕ iff there is a finite set of name vectors $\{\vec{m}_1, \ldots, \vec{m}_l\}$ such that $\vec{m}_i$ only contain names from H and $gnd(\phi)|H_k^+ \models \forall(\alpha(\vec{x}) \equiv \bigvee_i \vec{x} = \vec{m}_i)$ for every $k \geq 0$.

If $\alpha(\vec{x})$ is JIT wrt ϕ then the set of name vectors $\vec{m}$ such that $gnd(\phi)|H \models \alpha(\vec{m})$ is called the *set of answers* for the *query* $\alpha(\vec{x})$. ∎

That is, the JIT property is restricted to cases where it suffices to consider any finite representation of a proper$^+$ KB from H onwards.[8] We now illustrate how the JIT property keeps the answers both finite and known.

Example 5.4.3. Let $\phi = \{distance = 1\}$ be a proper$^+$ KB. Suppose $\alpha(x)$ denotes the fluent formula $distance \neq x$. Clearly $\phi \models \alpha(n)$ for every name n other than 1. Therefore $\alpha(x)$ is not JIT wrt ϕ since ϕ entails instances of α for names other than those mentioned in ϕ. ∎

Example 5.4.4. Let $\phi = \forall x.\, near(bomb, x) = 1 \equiv x = C \vee x = D$. This is another way of saying that there are precisely two objects near the bomb: C and D.

Let $\alpha(x)$ denote the formula $x \neq C \wedge near(bomb, x) = 1$. Clearly ϕ entails a single instance, *viz.* $\alpha(D)$. Thus $\alpha(x)$ is indeed JIT.

Let $\alpha(x)$ instead denote the formula $near(C, x) = 1$. Since nothing is specified in ϕ regarding the objects that are near to C, it follows that ϕ does not entail any instance of $\alpha(x)$. Therefore $\alpha(x)$ is not JIT. ∎

There is a simple syntactic way by which, given atoms that are JIT wrt a theory, we can construct complex formulas that remain JIT. Following Vassos et al. [2009], we introduce the notion of *range-restricted formulas*, which is based on the concept of *safe queries* from database theory [Abiteboul et al., 1995].[9]

Definition 5.4.5. (Range-restricted formulas.) We say that a fluent formula α is *safe-range* wrt a set of variables $\mathcal{X}$ according to the following rules:

1. If $\vec{m}$ is a vector of names, then:
   (a) $x = m$ is safe-range wrt $\{x\}$;
   (b) $f(\vec{m}) = n$ and $f(\vec{x}) = n$ is safe-range wrt $\{\}$;

[8] We remark that our notion of the JIT property is inspired by, but not the same as, the one appearing in [Vassos et al., 2009].

[9] We remark that our notion of range-restricted formulas is essentially the same as the one appearing in [Vassos et al., 2009], extended for a language with functional fluents.

(c) $f(\vec{m}) = y$ and $f(\vec{x}) = y$ is safe-range wrt $\{y\}$.

(d) If f represents a relation then:[10]

i. $f(\vec{m}, n) = 1$ and $f(\vec{x}, n) = 1$ is safe-range wrt $\{\}$;

ii. $f(\vec{m}, y) = 1$ and $f(\vec{x}, y) = 1$ is safe-range wrt $\{y\}$;

2. If α and α' are safe-range wrt $\mathcal{X}$ and $\mathcal{X}'$ respectively, then:

(a) $\alpha \wedge \alpha'$ is safe-range wrt $\mathcal{X} \cup \mathcal{X}'$;

(b) $\alpha \vee \alpha'$ is safe-range wrt $\mathcal{X} \cap \mathcal{X}'$;

(c) $\neg\alpha$ is safe-range wrt $\{\}$;

(d) $\exists x\alpha$ is safe-range wrt $\mathcal{X} - \{x\}$ provided that $x \in \mathcal{X}$.

3. No other formula is safe-range.

A formula is *range-restricted* iff it is safe-range wrt the set of its free variables. ∎

Example 5.4.6. We illustrate the notion of range-restricted formulas with simple examples.

1. $distance = x$ is safe-range wrt $\{x\}$ by clause 1(c) and since x is the only free variable here, it is also range-restricted.

2. $near(bomb, x) = 1$ is safe-range wrt $\{x\}$ by clause 1(d) and therefore, it is also range-restricted.

3. $near(bomb, x) \neq 1$ is safe-range wrt $\{\}$ by clause 2(c) and therefore it is not range-restricted.

4. $holding = x \wedge near(x, y) = 1$ is safe-range wrt $\{x, y\}$. This is because $holding = x$ is safe-range wrt $\{x\}$ by clause 1(c) and $near(x, y) = 1$ is safe-range wrt $\{y\}$ by clause 1(d) and therefore, the formula itself is safe-range wrt $\{x\} \cup \{y\}$ by clause 2(a). Clearly then it is range-restricted as well. On the other hand, $near(x, y) = 1$ by itself is not range-restricted since it has both x and y as free variables but it is safe-range wrt $\{y\}$ only.

5. $holding = x \vee near(x, y) = 1$ is safe-range wrt $\{\}$ by clause 2(b) and therefore not range-restricted.

6. $x \neq C \wedge near(bomb, x) = 1$ is safe-range wrt $\{x\}$ since the first conjunct is safe-range wrt $\{\}$, the second is safe-range wrt $\{x\}$ and thus, the formula itself is safe-range wrt $\{x\}$ by clause 2(a). Since x is the only free variable in the formula, it is also range-restricted. ∎

We now prove that the JIT property holds for complex formulas in the following way. For the theorem below, given an atom $\beta(\vec{u}, \vec{h})$ that is safe-range wrt variables in $\vec{h}$ (by an atom we mean formulas considered under clause 1 of Definition 5.4.5) and that is JIT for all substitutions of $\vec{u}$, we prove that complex formulas constructed from such atoms which are range-restricted also have the JIT property.

Theorem 5.4.7. *Suppose ϕ is a proper$^+$ KB. Let $\alpha(\vec{x}, \vec{y})$ be a fluent formula that is safe-range wrt variables in $\vec{y}$. Suppose for every atom $\beta(\vec{u}, \vec{h})$ mentioned in $\alpha(\vec{x}, \vec{y})$ which is safe-range wrt $\vec{h}$, $\beta(\vec{m}, \vec{h})$ is* JIT *wrt ϕ for every name vector $\vec{m}$. Then for every name vector $\vec{o}$, $\alpha(\vec{o}, \vec{y})$ is* JIT *wrt ϕ.*

[10] Recall (from Section 4.1) that this is captured by letting the name 1 denote truth, while every other name denotes falsity.

Proof: The proof is by induction on the construction of α. Since α is safe-range wrt the set of its free variables, we only have to consider the clauses identified in Definition 5.4.5. For the base case, we consider atoms, *i.e.* formulas appearing in clause 1 of Definition 5.4.5, say $\beta(\vec{u}, \vec{h})$ and for every name vector $\vec{m}$ that is the same size as $\vec{u}$, $\beta(\vec{m}, \vec{h})$ is JIT by assumption. So the base case holds trivially.

For the induction step, we have to consider the case identified in clause 2 of Definition 5.4.5. We only show the case of 2(b), and the other cases are similar. Suppose that we have the formula $\alpha_1(\vec{x}_1, \vec{u}, \vec{y}_1)$ which is, say, safe-range wrt the variables in $\{\vec{y}_1, \vec{u}\}$. Similarly, suppose that we have another formula $\alpha_2(\vec{x}_2, \vec{u}, \vec{y}_2)$ which is safe-range wrt $\{\vec{y}_2, \vec{u}\}$. Let us now suppose that $\vec{y}_1$ and $\vec{y}_2$ do not share variables. We now consider the formula $\alpha(\vec{x}_1, \vec{x}_2, \vec{u}, \vec{y}_1, \vec{y}_2) = \alpha_1(\vec{x}_1, \vec{u}, \vec{y}_1) \vee \alpha_2(\vec{x}_2, \vec{u}, \vec{y}_2)$. By Definition 5.4.5, α is only safe-range wrt the variables in $\vec{u}$.

By induction, for any vector of names $\vec{m}_1$ that is the same size as $\vec{x}_1$, we have

$$\phi \models \forall(\alpha_1(\vec{m}_1, \vec{u}, \vec{y}_1) \equiv \bigvee(\vec{u} = \vec{o}_1 \wedge \vec{y}_1 = \vec{n}_1)).$$

By induction for any vector of names $\vec{m}_2$ that is the same size as $\vec{x}_2$, we have

$$\phi \models \forall(\alpha_2(\vec{m}_2, \vec{u}, \vec{y}_2) \equiv \bigvee(\vec{u} = \vec{o}_2 \wedge \vec{y}_2 = \vec{n}_2)).$$

Now, let $\vec{m}_1^*$ and $\vec{m}_2^*$ be name vectors that are the same size as $\vec{y}_1$ and $\vec{y}_2$ respectively. Then we have

$$\phi \models \forall(\alpha(\vec{m}_1, \vec{m}_2, \vec{u}, \vec{m}_1^*, \vec{m}_2^*) \equiv \bigvee(\vec{u} = \vec{o}_1 \wedge \vec{m}_1^* = \vec{n}_1) \vee \bigvee(\vec{u} = \vec{o}_2 \wedge \vec{m}_2^* = \vec{n}_2)).$$

By the uniqueness of names, it follows that

$$\phi \models \forall(\alpha(\vec{m}_1, \vec{m}_2, \vec{u}, \vec{m}_1^*, \vec{m}_2^*) \equiv \bigvee \vec{u} = \vec{o}_j)$$

for some set of names $\vec{o}_j$. ∎

Example 5.4.4 continued. Let $\alpha(x)$ denote the formula $x \neq C \wedge near(bomb, x) = 1$. We observed in Example 5.4.4 that it is JIT wrt ϕ, and that $\phi \models \forall(\alpha(x) \equiv x = D)$. The JIT property can be independently justified using the above theorem based on the atoms in α as follows.

First, consider $near(bomb, x) = 1$, whose free variable is $\{x\}$. We noted in Example 5.4.6 that this atom is also safe-range wrt $\{x\}$. It is also trivially follows that this atom is JIT wrt ϕ.

Next, consider $x \neq C$. This atom, on the other hand, is not range-restricted since it is safe-range wrt $\{\}$ and yet it contains x as a free variable. However, on substituting x with any name, it no longer contains any free variables and therefore it is vacuously JIT wrt ϕ. Therefore $\alpha(\vec{x})$, which is safe-range wrt its free variables, is also JIT wrt ϕ. ∎

5.4.2 Just-in-time Progression

Based on the concepts developed in the previous section, we are now prepared to introduce the concept of range-restricted theories and the conditions under which progression becomes first-order definable.

Definition 5.4.8. (Range-restricted Action Theories.) Suppose the successor state axiom for a fluent $f \in \mathcal{F}$ is of the following form:

$$\Box[v]f(\vec{x}) = y \equiv \gamma_f(\vec{x}, y, v) \vee f(\vec{x}) = y \wedge \neg\exists h\gamma_f(\vec{x}, h, v).$$

The successor state axiom is *range-restricted* iff $\gamma_f(\vec{x}, y, v)$ is a disjunction of formulas of the form:

$$\exists\vec{u}[v = A(\vec{z}) \wedge \mu(\vec{z}, \vec{h})],$$

where $\vec{z}$ may contain $\vec{x} \cup \{y\}$, $\vec{u}$ are the variables in $\vec{z}$ but not in $\vec{x} \cup \{y\}$ and $\vec{h}$ are the variables in $\vec{x} \cup \{y\}$ but not in $\vec{z}$. The formula $\mu(\vec{z}, \vec{h})$ is called the *context formula* and it is safe-range wrt the variables in $\vec{h}$.

A basic action theory is range-restricted if all the successor state axioms are range-restricted in the above sense. ∎

Example 5.4.9. Consider the successor state axiom for the *distance* fluent from Example 4.1.3. Here the action *forward* does not have arguments, and according to the above definition the context formula $distance = x + 1$ must be safe-range wrt $\{x\}$. This is indeed the case, and so the successor state axiom for *distance* is range-restricted, and the simple robot domain is a range-restricted basic action theory.

Similarly, consider the successor state axiom for the exploding bomb from Example 5.4.1. Here too the action *explode* does not have arguments. According to the above definition, the context formula

$$\forall x.\, near(bomb, x) = 1 \wedge y = destroyed \wedge status(bomb) = active$$

should be safe-range wrt $\{x, y\}$. This is indeed the case, and so γ_{status} is a range-restricted successor state axiom. ∎

The intuition behind the syntactic form of range-restricted theories is that on instantiating the successor state axiom wrt a primitive action, the context formula simplifies to a range-restricted one. Then provided that this simplified formula is JIT wrt to the initial theory, we obtain a finite set of fluent terms that are affected after the action is performed. With this, the forgetting procedure identified earlier can be used to obtain a first-order progression.

For the first step, the JIT assumption that we are after is as follows:

Definition 5.4.10. A range-restricted basic action theory Σ is JIT wrt a primitive action r if for all fluents f, $\gamma_f(\vec{x}, y, r)$ is JIT wrt Σ_0. ∎

Next, consider the simplification of the successor state axioms wrt a primitive action to a range-restricted formula.

Proposition 5.4.11. *Let $A(\vec{o})$ be any primitive action. Suppose Σ is a range-restricted basic action theory that has a proper$^+$ KB as an initial theory and that is* JIT *wrt $A(\vec{o})$. Then for every fluent f there exists a formula $\delta(\vec{x}, y)$ of the form:*

$$\vec{x} = \vec{m}_1 \wedge y = n_1 \vee \ldots \vee \vec{x} = \vec{m}_k \wedge y = n_k$$

where $\vec{m}_i$ and n_i are names appearing in the initial KB such that the following holds:

$$\Sigma_0 \models \forall(\gamma_f(\vec{x}, y, r) \equiv \delta(\vec{x}, y)).$$

- $\phi_1 = \{distance = 4\}$.
- $\phi_2 = \{\forall(near(bomb, x) = 1 \equiv x = C \vee x = D),$

  $status(bomb) = active,$

  $\forall(x = C \vee x = D \supset status(x) \neq destroyed)\}$.

Figure 5.4: Initial knowledge bases for JIT progression.

Proof: Since the basic action theory is range-restricted, the successor state axiom for the fluent f is range-restricted. Consider Definition 5.4.8 where the variables $\vec{z}$ of the action perhaps includes some variables from $\vec{x} \cup \{y\}$. The remaining variables of $\vec{x} \cup \{y\}$ are $\vec{h}$.

Consider that $\gamma_f(\vec{x}, y, v)$ is a disjunction of formulas of the form $\exists \vec{u}.[v = A(\vec{z}) \wedge \mu(\vec{z}, \vec{h})]$. Then by the uniqueness of actions, $\gamma_f(\vec{x}, y, A(\vec{o}))$ simplifies to $\exists \vec{u}.[A(\vec{o}) = A(\vec{z}) \wedge \mu(\vec{z}, \vec{h})]$ which is equivalent to $\exists \vec{u}.[\vec{z} = \vec{o} \wedge \mu(\vec{z}, \vec{h})]$, *i.e.* $\exists \vec{u}.[\vec{z} = \vec{o} \wedge \mu(\vec{o}, \vec{h})]$.

Considering that $\mu(\vec{o}, \vec{h})$ is safe-range wrt $\vec{h}$ and by the JIT assumption, we have $\Sigma_0 \models \forall(\mu(\vec{o}, \vec{h}) \equiv \bigvee \vec{h} = \vec{n}_i)$ for some vectors $\vec{n}_i$. Supposing that the name vectors corresponding to $\vec{x}$ and y from $\{\vec{o}, \vec{n}_i\}$ are $\vec{m}_i$ and n_i respectively, it follows that $\Sigma_0 \models \forall(\gamma_f(\vec{x}, y, r) \equiv \bigvee \vec{x} = \vec{m}_i \wedge y = n_i)$. ∎

Notice an important difference to a similar simplification we obtained in the local-effect case in Proposition 5.2.12. In the local-effect case, we did not need the initial theory to proceed with the simplification. In contrast, as we observed in the proof of Proposition 5.4.11, the initial theory and the JIT assumption are crucial to resolve the context formula to a set of name vectors. The following examples also illustrate this property:

Example 5.4.12. Consider the successor state axiom for the *distance* fluent which we noted in Example 5.4.9 to be a range-restricted successor state axiom. Then the instantiation of $\gamma_{distance}$ wrt *forward* results in the range-restricted context formula $distance = x + 1$. Suppose that the initial theory is ϕ_1 from Figure 5.4. Clearly then the context formula is JIT wrt the initial theory. We also note that

$$\phi_1 \models \forall(\gamma_{distance}(x, forward) \equiv x = 3).$$

Similarly, consider the instantiation of the successor state axiom for the fluent *status* wrt the action *explode*. The context formula, as noted in Example 5.4.9, is equivalent to:

$$\forall x.\ near(bomb, x) = 1 \wedge y = destroyed \wedge status(bomb) = active.$$

Denote the context formula as $\mu(x, y)$. If the initial theory is ϕ_2 from Figure 5.4, then clearly

$$\phi_2 \models \forall(\mu(x, y) \equiv y = destroyed \wedge (x = C \vee x = D)),$$

as implied by Proposition 5.4.11. ∎

Note that the simplified form may look different for every primitive action. We will now identify a finite number of fluent terms that are affected after the execution of an action, as we did with local-effect actions.

Definition 5.4.13. Let Σ be a range-restricted basic action theory, where Σ_0 is a proper$^+$ KB, that is JIT wrt a primitive action $A(\vec{o})$. Without any loss of generality, let the instantiation of the successor state axioms for fluent f which is $\gamma_f(\vec{x}, y, A(\vec{o}))$ be simplified to the formula $\delta(\vec{x}, y)$ as indicated by Proposition 5.4.11. Then define the *argument set* of f wrt $A(\vec{o})$ as the following set Ω_f of name vectors:

$$\Omega_f = \{\vec{m} \mid \vec{x} = \vec{m} \text{ appears in } \delta(\vec{x}, y)\}.$$

Define the *characteristic set* of $A(\vec{o})$ as the following set of primitive terms:

$$\Delta = \{f(\vec{m}) \mid \vec{m} \in \Omega_f \text{ for some fluent } f \in \Omega_f\}. \blacksquare$$

Since $\mathcal{F}$ is finite, both the argument set and the characteristic set are finite.

Example 5.4.14. Note from Example 5.4.12 that the argument set of the fluent *distance* wrt the action *forward* is simply the empty set since this fluent does not have any argument. Thus, the characteristic set is simply $\{distance\}$.

On similar lines, from Example 5.4.12, consider the instantiation of the successor state axiom for the fluent *status* wrt *explode*. Observe that the argument set is $\Omega_{status} = \{C, D\}$ and thus the characteristic set is $\{status(C), status(D)\}$. $\blacksquare$

As with local-effects, the argument set Ω_f identifies all primitive terms from $f(\vec{x})$ that are affected after an action. Equivalently, for every vector of names $\vec{n} \notin \Omega_f$, it follows that the value of $f(\vec{n})$ remains the same after the action. The following proposition illustrates this:

Proposition 5.4.15. *Let Σ be a range-restricted basic action theory, where Σ_0 is a proper$^+$ KB that is* JIT *wrt a primitive action $A(\vec{o})$. Let r denote $A(\vec{o})$ and let Ω_f be the argument set of f wrt r. Then*

$$\Sigma_0 \wedge \Sigma_{post} \models \forall \vec{x}.\ \vec{x} \notin \Omega_f \supset [r]f(\vec{x}) = y \equiv f(\vec{x}) = y.$$

Proof: By definition, $\Sigma_{post} \models [r]f(\vec{x}) = y \equiv \gamma_f(\vec{x}, y, r) \vee f(\vec{x}) = y \wedge \neg\exists h \gamma_f(\vec{x}, h, r)$. Suppose $w \models \Sigma_0 \wedge \Sigma_{post}$. Now, as implied by Proposition 5.4.11, $w \models \forall(\gamma_f(\vec{x}, y, r) \equiv \bigvee \vec{x} = \vec{m} \wedge y = n)$. Then, for any $\vec{m} \notin \Omega_f$ it follows that $\Sigma_0 \wedge \Sigma_{post} \models [r]f(\vec{m}) = y \equiv f(\vec{m}) = y$. $\blacksquare$

Example 5.4.16. Consider the argument set for the fluent *status* wrt the action *explode* from Example 5.4.14. Here $\Omega_{status} = \{C, D\}$. Suppose the initial theory is ϕ_2 from Figure 5.4. Then it easy to to verify that for $\{E\} \notin \Omega_{status}$, we have the following property by way of Proposition 5.4.15:

$$\phi_2 \wedge \Sigma_{post} \models [r]status(E) = y \equiv status(E) = y.$$

That is, the block E is not destroyed after the bomb explodes because it is not in the vicinity of the bomb. $\blacksquare$

We are now ready to prove the main progression theorem. Due to the similarity of the arguments that we have made in this section to the local-effects case, the progression theorem takes after Theorem 5.2.18.

Theorem 5.4.17. *Let $\Sigma = \phi \wedge \Box\beta$ be a range-restricted basic action theory, where ϕ is a proper$^+$ KB that is* JIT *wrt primitive action r. Then*

$$\models \boldsymbol{O}(\phi \wedge \Box\beta) \wedge SF(r) = x \supset [r]\boldsymbol{O}(Prog(\phi) \wedge \Box\beta)$$

where

$$Prog(\phi) = forget((\phi \wedge \varphi_r^v)_{\vec{P}}^{\vec{F}} \wedge \Omega_{ss}, \Delta_{\vec{F}}^{\vec{P}}) \text{ and}$$

$$\Omega_{ss} = \{f(\vec{m}) = y \equiv \gamma_f(\vec{m}, y, r)_{\vec{P}}^{\vec{F}} \mid \vec{m} \in \Omega_f\}.$$

Proof: The formal argument follows the proof method for Theorem 5.2.18. The only difference is that the characteristic set is obtained with range-restricted theories by using the JIT property. But since we restricting ourselves to models of $\phi \wedge \Box\beta$, *i.e.* worlds that satisfy $\phi \wedge \Box\beta$, this does not complicate the ideas behind the proof. ∎

Example 5.4.18. We will pursue the progression of the simple robot basic action theory with ϕ_1 from Figure 5.4 as the initial theory, wrt *forward*. From Theorem 5.4.17 we first identify the components of the progressed theory, which is obtained by forgetting $\Delta_{\vec{P}}^{\vec{F}}$ from $(\phi \wedge \varphi_r^v)_{\vec{P}}^{\vec{F}} \wedge \Omega_{ss}$ and then replacing $\vec{P}$ with $\vec{F}$ in the resultant. Let us use P as a second-order function variable for the fluent *distance*.

- The initial theory, the sensing results and the instantiated successor state axioms:
  1. ${\Sigma_0}_{\vec{P}}^{\vec{F}} = \{P = 4\}$
  2. ${\varphi_r^v}_{\vec{P}}^{\vec{F}} = \{\text{TRUE}\}$.
  3. For the instantiated successor state axioms, recall from Example 5.4.14 that the characteristic set is simply $\{distance\}$. Now, by assumption, the *rhs* of the successor state axiom is JIT wrt ϕ_1. Thus, by simplifying the context formula as we have done so in Example 5.4.12, we obtain $distance = x \equiv x = 3$.
- P is to be forgotten.

Now, forgetting P results in

$$\exists u.\ u = 4 \wedge distance = x \equiv x = 3$$

which is equivalent to $distance = 3$, which is the progression of the action theory wrt *forward*. ∎

Example 5.4.19. We will pursue the progression of the basic action theory involving the bomb with ϕ_2 from Figure 5.4 as the initial theory, wrt *explode*. From Theorem 5.4.17, we first identify the components of the progressed theory, which consists of forgetting $\Delta_{\vec{P}}^{\vec{F}}$ from $(\phi \wedge \varphi_r^v)_{\vec{P}}^{\vec{F}} \wedge \Omega_{ss}$, and then replacing $\vec{P}$ with $\vec{F}$ in the resultant. Let use P and Q as second-order function variables for *near* and *status* respectively.

- The initial theory, the sensing results and the instantiated successor state axioms:
  1. ${\Sigma_0}_{\vec{P}}^{\vec{F}}$ is a conjunction of
     (a) $\forall(P(bomb, x) = 1 \equiv x = C \vee x = D)$,

(b) $Q(bomb) = active$,

(c) $\forall(x = C \vee x = D \supset Q(x) \neq destroyed)$.

2. $\varphi_{r\vec{P}}^{v\vec{F}} = \{\text{TRUE}\}$.

3. For the instantiated successor state axioms Ω_{ss}, recall from Example 5.4.14 that the characteristic set is $\{status(C), status(D)\}$. Then after simplifying the context formula by means of its JIT property as we have done so in Example 5.4.12, Ω_{ss} is shown to be equivalent to

$$\{status(C) = destroyed, status(D) = destroyed\}.$$

- $\Delta_{\vec{P}}^{\vec{F}} = \{Q(C), Q(D)\}$ is to be forgotten.

Forgetting $\Delta_{\vec{P}}^{\vec{F}}$ results in the removal of $\{1(c)\}$. Next, since the fluent *near* is not affected after any action, it is easy to show that its instantiated successor state axiom is of the form $near(x, y) = z \equiv P(x, y) = z$. Thus, P can be eliminated easily. Then, we get the following progressed KB:

$$\forall(near(bomb, x) = 1 \equiv x = C \vee x = D) \wedge$$

$$status(bomb) = active \wedge$$

$$status(C) = destroyed \wedge status(D) = destroyed.$$

That is, after the bomb explodes, both blocks C and D, which are in the bomb's vicinity, are destroyed. ∎

5.4.3 Computability Results for Range-Restricted Theories

There are two major steps involved when efficiently progressing range-restricted theories. The first is the computation of the possible answers for the context formula, that is, finding the set of name vectors for a formula $\delta(\vec{x}, y)$ such that Proposition 5.4.11 is true. By the definition of the JIT condition that is imposed in the progression result of Theorem 5.4.17, we only need to look at the finite set of names H mentioned in the initial theory which is a proper$^+$ KB. Moreover, we can restrict ourselves to $gnd(\phi)|H$ by Definition 5.4.2. In other words, given the context formula $\mu(\vec{x})$, in the worst case we need to check if $gnd(\phi)|H$ entails $\mu(\vec{m})$ for every permutation of names from the finite set H.

In the next section, we outline a procedure that shows that checking $\mu(\vec{m})$ against $gnd(\phi)|H$ is decidable. While this does not make it necessarily efficient, it nevertheless shows that resolving the possible answers for the context formula is at least computable. On the other hand, by storing this information as a database, the retrieval of possible answers can be done efficiently [Vassos et al., 2009; Abiteboul et al., 1995].

The second major step involved in computing the progression of range-restricted theories is the forgetting of the characteristic set from the conjunction of the initial theory and the instantiated successor state axioms. We argued in Section 5.3.2 that this is computable, and under reasonable assumptions it can be done in $O(n)$ time, when n is the size of the initial KB.

Theorem 5.4.20. *Suppose Σ is a range-restricted basic action theory, where Σ_0 is a proper$^+$ KB, that is* JIT *wrt a primitive action r. Suppose also that for every fluent f, we are given the possible answers for $\gamma_f(\vec{x}, y, r)$. Then progression of Σ wrt r is definable as a proper$^+$ KB, and can be efficiently computed.*

Proof: If we are given the possible answers to the context formula,[11] observe that the instantiated successor state axioms simply reduce to a finite set of primitive equalities. That is, since for every $f \in \mathcal{F}$ we have $\Sigma_0 \models \forall(\gamma_f(\vec{x}, y, r) \equiv \bigvee \vec{x} = \vec{m} \wedge y = n)$ from Proposition 5.4.11, it follows that the conjunction of the initial theory and the instantiated successor state axioms from which the characteristic set is to be forgotten is definable as a proper$^+$ KB. Finally, forgetting a finite set of primitive terms from a proper$^+$ KB is definable as a proper$^+$ KB and efficient by Theorem 5.3.17. ∎

Example 5.4.21. In order to illustrate the computability results from this section, we reconsider the progression of the bomb basic action theory from Example 5.4.19, wrt *explode*. From Theorem 5.4.17, we first identify the components of the progressed theory which consists of forgetting $\Delta_{\vec{P}}^{\vec{F}}$ from $(\phi \wedge \varphi_r^v)_{\vec{P}}^{\vec{F}} \wedge \Omega_{ss}$ and then replacing $\vec{P}$ with $\vec{F}$ in the resultant. Let us use P as a second-order function variable for *near* and Q as a second-order variable for *status*.

- The initial theory, the sensing axioms and the instantiated successor state axioms:
  1. ${\Sigma_0}_{\vec{P}}^{\vec{F}}$ is a conjunction of
     (a) $\forall(P(bomb, x) = 1 \equiv x = C \vee x = D) \wedge Q(bomb) = active$,
     (b) $\forall(x = C \vee x = D) \supset Q(x) \neq destroyed$.
  2. $\varphi_{\vec{P}}^{\vec{F}} = \{\text{TRUE}\}$.
  3. The instantiated successor state axioms simplify to (see Example 5.4.19):
     $\{status(C) = destroyed, status(D) = destroyed\}$.
- $\Delta_{\vec{P}}^{\vec{F}} = \{Q(C), Q(D)\}$ is to be forgotten.

First, consider forgetting $Q(C)$ from $\{(1), (2), (3)\}$ by means of the procedure outlined in Section 5.3.2. Since $Q(C)$ is irrelevant to $\{(1a), (2), (3)\}$, we convert $1(b)$ to the normal form wrt $Q(C)$:

- $\forall((x = C \vee x = D) \wedge x = C \supset Q(C) \neq destroyed)$,
- $\forall((x = C \vee x = D) \wedge x \neq C \supset Q(x) \neq destroyed)$.

From Theorem 5.2.4, forgetting $Q(C)$ from $\{(1b)\}$ is equivalent to $\exists u.\{(1b)\}[Q(C) = u]$, which is equivalent to

$$\forall((x = C \vee x = D) \wedge x \neq C \supset Q(x) \neq destroyed). \tag{5.4}$$

Similarly, $Q(D)$ is irrelevant to $\{(1a), (2), (3)\}$. So we now need to forget $Q(D)$ from (5.4) which can be shown to be equivalent to TRUE. Therefore, forgetting $\{Q(D), Q(C)\}$ from $\{(1), (2), (3)\}$ is equivalent to $\{(1a), (2), (3)\}$. Finally, on eliminating P, we obtain the progression as:

$\forall(x = C \vee x = D \equiv near(bomb, x) = 1) \wedge status(bomb) = active \wedge$

$status(D) = destroyed \wedge status(C) = destroyed.$ ∎

[11] Recall that by definition of the JIT property, the possible answers can only range over the set of names in Σ_0, which is necessarily finite.

5.5 Query Evaluation

The query evaluation procedure we have in mind is a logically sound and complete decision procedure for certain classes of queries on proper$^+$ KBs. To understand its basic principles, we begin with the simpler case where the KB ϕ and the query α are quantifier-free closed fluent formulas, and thus representable as *primitive clauses*. By a primitive clause we mean a disjunction of primitive equalities.

5.5.1 Reasoning with Quantifier-free Clauses

The decision procedure is essentially inspired by Boolean satisfiability, and in particular, the DPLL technique [Davis and Putnam, 1960; Davis et al., 1962]. The usual methodology is to transform the validity problem into a satisfiability problem by means of refutation. That is,

$$\phi \models \alpha \quad \text{iff} \quad \phi \wedge \neg\alpha \text{ is unsatisfiable.}$$

Clearly if $\phi \wedge \neg\alpha$ is satisfiable, then $\phi \not\models \alpha$.

We begin with the notion of an *assignment*, which is closely related to the notion of *partial assignments* in satisfiability solvers [Gomes et al., 2008].

Definition 5.5.1. (Assignments.) Let $p = \{f(\vec{m}) = n\}$ be a positive primitive equality, and ψ any set of primitive clauses. Then let $[\psi]_p$ denote replacing every $f(\vec{m}) = n'$ in ψ with $n' = n$, and simplifying ψ in the sense of removing all clauses that contain at least one TRUE literal, and deleting all occurrences of FALSE in the individual clauses.

Given a set of consistent positive primitive equalities $\{p_1, \ldots, p_k\}$, we define $[\psi]_{\{p_1,\ldots,p_k\}}$, $k > 1$, inductively as

$$[[\psi]_{p_1}]_{\{p_2,\ldots,p_k\}}. \quad \blacksquare$$

The idea is that $[\psi]_p$ reduces ψ to a simpler formula that is satisfiable provided $p \wedge \psi$ is. More precisely,

Proposition 5.5.2. *Let ψ and p be as above, and let w be any world such that $w \models p$. Then $w \models \psi$ iff $w \models [\psi]_p$.*

Proof: The argument is based on an induction of ψ. We consider only the base case, where ψ is a primitive equality. The other cases are straightforward.

Suppose p is $d = n$. Then there are three possibilities with the base case:

case ψ does not mention d. Then $[\psi]_p = \psi$, and the argument is trivial.

case ψ is $d = n$ or $d \neq n'$ for some $n' \neq n$. Then $[\psi]_p$ is simply TRUE. Clearly, $w \models [\psi]_p$. Since $w \models p$, $w \models \psi$.

case ψ is $d \neq n$ or $d = n'$ for some $n' \neq n$. Then $[\psi]_p$ is FALSE. Then $w \not\models [\psi]_p$. Also, $w \not\models \psi$ since $w \models p$. $\blacksquare$

Example 5.5.3. Let $\psi = \{distance = 4 \vee distance \neq 5\}$ and p denote $distance = 5$. Then $[\psi]_p$ is $5 = 4 \vee 5 \neq 5$. Therefore, $[\psi]_p$ is simply FALSE. $\blacksquare$

With the notion of assignment in hand, we can proceed to discuss how the satisfiability of a set of primitive clauses ψ can be established. As hinted earlier, the methodology is based on DPLL which given $\psi \doteq \phi \wedge \neg\alpha$ returns SAT and a set of literals $\mathcal{G}$ such that $[\psi]_{\mathcal{G}}$ is TRUE, if ψ is satisfiable. In other words, the procedure returns a satisfying assignment for the input theory. On the other hand, if ψ is unsatisfiable, then the procedure returns UNSAT.

A fundamental step in regular DPLL is that of recursively choosing a literal from the set of remaining clauses, and then pursuing two branches. In the first branch, the input set of clauses are simplified by setting the literal to true everywhere, and in the second branch, the simplification is by setting the literal to false. Roughly speaking, the proof procedure instantiated by DPLL is a binary tree, where the root is the set of input clauses and the successive children are the result of successive simplifications wrt truth assignments to literals.

In the method we will propose, the approach is similar. In contrast to propositions, which can only obtain values of TRUE or FALSE, we have to consider that primitive terms can obtain any one of the names from the domain. To this end, the branches in the procedure consists of *possible assignments of values* to the *primitive terms* in the remaining clauses. However, since the domain is not finite, this immediately leads to a proof tree where at each level the tree has an infinite number of branches. The following result instead proves that we are entitled to infer satisfiability by only considering the names mentioned in the input sentence, plus an arbitrary extra one.

Given a set of primitive clauses ψ and a primitive term d, let $H_d(\psi)$ denote the set of names $\{n_1, \ldots, n_k\}$ such that $d \circ_i n_i$ appears in d, where $\circ_i \in \{=, \neq\}$. Let $H_d^+(\psi)$ denote the union of $H_d(\psi)$ plus an arbitrary extra one. Now, the idea is to consider simplifications of ψ wrt $d = n_i$ for each i and verify if the remaining set of clauses are satisfiable. If SAT is not returned for any of these assignments, then the satisfiability of ψ can be confirmed with only one other assignment $d = n'$ for any $n' \in \mathcal{Q} - H_d(\psi)$.

Proposition 5.5.4. *Let ψ be a set of primitive clauses and d any primitive term. Then for any $n', n'' \notin H_d^+(\psi)$, $[\psi]_{d=n'}$ is logically equivalent to $[\psi]_{d=n''}$.*

Proof: By induction on ψ, and we only consider the base case since the other cases are straightforward.

- Suppose ψ is $d = n$. Since n' and n'' are both distinct from n by assumption, $[\psi]_{d=n'}$ and $[\psi]_{d=n''}$ are both FALSE.
- Suppose ψ is $d \neq n$. Since n' and n'' are both distinct from n by assumption, $[\psi]_{d=n'}$ and $[\psi]_{d=n''}$ are both TRUE.
- Suppose d is not mentioned in ψ, that is, ψ is $d' \circ n$, where $\circ \in \{=, \neq\}$ and d' is a different primitive term from d. Then $[\psi]_{d=n'}$ and $[\psi]_{d=n'}$ are both clearly equivalent to ψ itself. ∎

Theorem 5.5.5. *Let ψ be a set of primitive clauses and d any primitive term. Let $\mathcal{G} = \{d = n \mid n \in H_d^+(\psi)\}$. Then for any world w, $w \models \psi$ iff $w \models \bigvee_{p \in \mathcal{G}} [\psi]_p$.*

Proof: For every world w, $w \models d = n^*$ for some name n^*. Let p^* denote $d = n^*$. Suppose $p^* \in \mathcal{G}$, then by Proposition 5.5.2 the theorem clearly holds.

So suppose $p^* \notin \mathcal{G}$. By Proposition 5.5.2, $w \models \psi$ iff $w \models [\psi]_{p^*}$. Let p' denote $d = n' \in \mathcal{G}$ such that $n' \notin H_d(\varphi)$. By Proposition 5.5.4, $[\psi]_{p^*}$ is equivalent to $[\psi]_{p'}$. Therefore, $w \models \psi$ iff $w \models [\psi]_{p^*}$ iff $w \models [\psi]_{p'}$. ∎

Corollary 5.5.6. *Suppose ψ, d and $\mathcal{G}$ are as above. Then ψ is satisfiable iff $\bigvee_{p \in \mathcal{G}} [\psi]_p$ is satisfiable.*

We now give the DPLL procedure Algorithm 1 whose correctness is based on Theorem 5.5.5. Let us consider details of that procedure. The algorithm begins by *unit propagating*, where all clauses that contain a single positive literal are simplified wrt this literal. We outline this step in Algorithm 2. Next, we have two simple steps. If any of the clauses are FALSE, then the remaining theory cannot be satisfiable. Conversely, if the input theory is empty, *i.e.* simply TRUE, then clearly we are done. Then we arrive at the main component, where we construct DPLL branches by selecting a primitive term at random and considering assignments to it from $H^+_d(\varphi)$, as needed by Theorem 5.5.5.

Algorithm 1: DPLL($\psi, \mathcal{G}$)

Input: set of primitive clauses ψ with $\mathcal{G} = \{\}$ initially
Output: UNSAT, or a set of literals $\mathcal{G}$ such that $[\psi]_{\mathcal{G}}$ is TRUE
$(\psi, \mathcal{G}) \doteq$ UNIT-PROPOGATE($\psi, \mathcal{G}$);
if ψ *contains* FALSE **then**
  return UNSAT;
end
if ψ *has no* clauses *left* **then**
  output $\mathcal{G}$ and return SAT;
end
$d \doteq$ any primitive term not mentioned in $\mathcal{G}$ but appearing in ψ;
foreach $n \in H_d(\psi)$ **do**
  if DPLL($[\psi]_{d=n}, \mathcal{G} \cup \{d = n\}$) = SAT **then**
    return SAT;
  end
end
$n' \doteq$ any $n \in \mathcal{Q} - H_d(\psi)$;
return DPLL($[\psi]_{d=n'}, \mathcal{G} \cup \{d = n'\}$);

Algorithm 2: UNIT-PROPOGATE($\psi, \mathcal{G}$)

while ψ *does not contain* FALSE *but has unit clause* $d = n$ **do**
  $\psi \doteq [\psi]_{d=n}$;
  $\mathcal{G} \doteq \mathcal{G} \cup \{d = n\}$;
end

Finally, we now prove the correctness of the procedure in Theorem 5.5.7.

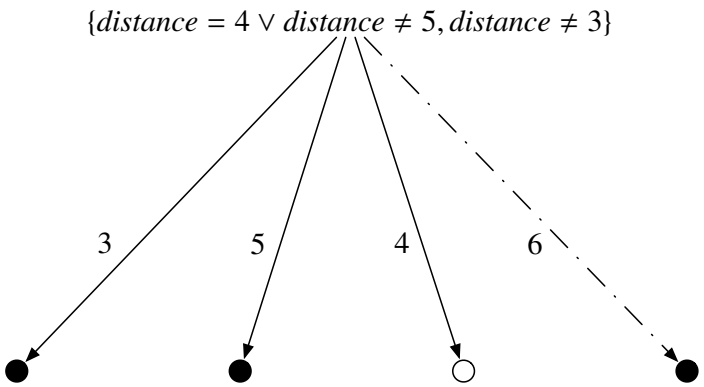

Figure 5.5: Sample DPLL proof tree. Black circles denote UNSAT, and white ones denote SAT.

Theorem 5.5.7. *Given any set of primitive clauses ψ, ψ is unsatisfiable (satisfiable) iff the* DPLL *procedure on ψ returns* UNSAT (SAT).

Proof: The proof is based on an induction on the number of distinct primitive equalities that are mentioned in ψ.

For the base case, suppose that ψ only mentions a single primitive equality. We prove by sub-induction on the length of ψ. We only show the base case where ψ is of the form $d = n$. The other cases are straightforward. For ψ as input, the algorithm pursues the assignment $d = n$, after which $[\psi]_{d=n}$ reduces to TRUE and the procedure returns SAT with $\{d = n\}$.

Suppose the result holds for k distinct equalities. Suppose now ψ mentions $k + 1$ distinct equalities. Then pursue the algorithm on ψ and the procedure picks a term d appearing in ψ. It then applies the DPLL procedure on $[\psi]_p$ for every $p \in \mathcal{G}$, where $\mathcal{G} = \{d = n \mid n \in H_d^+(\psi)\}$. Note that $[\psi]_p$ does not mention equalities with d, that is, it only mentions k distinct equalities. By induction then, since $[\psi]_p$ only mentions k distinct equalities, the DPLL procedure on $[\psi]_p$ returns SAT iff $[\psi]_p$ is satisfiable. From Corollary 5.5.6, we infer that ψ is satisfiable iff $\bigvee_{p\in\mathcal{G}}[\psi]_p$ is satisfiable iff one of the disjuncts in $\bigvee_{p\in\mathcal{G}}[\psi]_p$ is satisfiable iff (by induction) the DPLL procedure returns SAT for one of these disjuncts, *i.e.* the DPLL procedure returns SAT for ψ. ∎

Example 5.5.8. Suppose $\phi = \{distance = 4 \vee distance \neq 5\}$ and α denotes $\{distance = 3\}$.

- To verify if $\phi \models \alpha$, pursue finding whether $\psi \doteq \phi \wedge \neg\alpha$ is unsatisfiable. Let $H^+_{distance} = \{4, 5, 3, 6\}$. Then observe that ψ simplifies to TRUE wrt $distance = 4$ and therefore SAT is returned by the algorithm. This is also shown in Figure 5.5, where SAT is returned after only considering the names in $H_{distance}$. The procedure then does not consider the assignment $distance = 6$ which would have, in any case, returned UNSAT.

- To verify if $\phi \models \neg\alpha$, pursue finding whether $\psi \doteq \phi \wedge \alpha$ is unsatisfiable. Let $H^+_{distance}$ be as above. Then observe that ψ simplifies to TRUE wrt $distance = 3$. Therefore SAT is returned.

From this, we infer that neither $\phi \models \alpha$ nor $\phi \models \neg\alpha$. ∎

5.5.2 Handling Quantifiers

In this section, we extend the scope of the DPLL procedure to proper$^+$ KBs. At the same time, we allow for a more expressive class of queries. Both of these extensions rely on two main results. The first is about finding a finite quantifier-free representation for proper$^+$ KBs given a query.

The DPLL procedure is limited to quantifier-free clauses, while proper$^+$ KBs clearly mention quantifiers. Moreover, a proper$^+$ KB in general is equivalent to an infinite set of quantifier-free clauses, so computing $gnd(\phi)$ for a proper$^+$ KB ϕ and then applying the DPLL procedure is also not possible.

Instead, we proceed as follows. Given a proper$^+$ KB ϕ, let k be the maximum number of variables mentioned in a $\forall$-clause in ϕ. We now prove that given a quantifier-free query α, it is sufficient to restrict our attention to $gnd(\phi)|H_k^+$, where H_k^+ denotes the names in $\phi \cup \{\alpha\}$ and k new ones. To prove this, however, we will first need the following *Compactness* property:[12]

Theorem 5.5.9. *Suppose ϕ is a proper$^+$ KB and α is a closed quantifier-free formula. Then $S = gnd(\phi) \cup \{\neg\alpha\}$ is satisfiable iff every finite subset of S is satisfiable.*

The proof of this theorem appears in the Appendix. The argument is that a propositional encoding of $gnd(\phi) \cup \{\neg\alpha\}$ is possible such that $gnd(\phi) \cup \{\neg\alpha\}$ is satisfiable in $\mathcal{ES}$ iff its encoding it satisfiable in propositional logic.

Theorem 5.5.10. *Suppose ϕ is a proper$^+$ KB, and let k be the maximum number of variables mentioned in a $\forall$-clause in ϕ. Suppose α is a closed quantifier-free formula. Let H be all the names in $\phi \cup \{\alpha\}$, and H_k^+ be the union of H and k arbitrary new ones. Then*

$$\phi \models \alpha \;\; \textit{iff} \;\; gnd(\phi)|H_k^+ \models \alpha.$$

Proof: The if direction is immediate. For the only-if direction, we argue as follows. Suppose $\phi \wedge \neg\alpha$ is unsatisfiable, *i.e.* $S = gnd(\phi) \cup \{\neg\alpha\}$ is unsatisfiable. By Theorem 5.5.9, it cannot be that every finite subset S is satisfiable. Therefore, there is some finite subset of S, say S', that is unsatisfiable. Since by definition a proper$^+$ KB is a finite and satisfiable set of $\forall$-clauses, it follows that S' contains $\neg\alpha$. That is, S' is the conjunction of some finite subset of $gnd(\phi)$, say $gnd(\phi)|H_b^+$ for some finite b, and $\neg\alpha$.

We will now prove that if $gnd(\phi)|H_k^+ \wedge \neg\alpha$ is satisfiable then $gnd(\phi)|H_j^+ \wedge \neg\alpha$ is satisfiable for all $j \geq k$. Hence, $gnd(\phi)|H_b^+ \wedge \neg\alpha$ is satisfiable, showing, by contradiction, that $\phi \wedge \neg\alpha$ cannot be unsatisfiable.

Let us denote $gnd(\phi)|H_k^+ \wedge \neg\alpha$ by Γ and let us denote $gnd(\phi)|H_j^+ \wedge \neg\alpha$ by Γ'. We assume without loss of generality that H_j^+ is the union of the names in H_k^+ and $j - k$ new ones. Given a primitive clause φ, let us write $T(\varphi)$ to mean the set of all the primitive terms mentioned in φ and let us write $N(\varphi)$ to mean the set of all names appearing in φ.

So suppose Γ is satisfiable, and let $w \models \Gamma$. Let us construct w' based on the following rules:

1. for all $d \in T(\Gamma)$, let $w'[d, \langle\rangle] = w[d, \langle\rangle]$;

2. for all $d \notin T(\Gamma')$, let $w'[d, \langle\rangle] = w[d, \langle\rangle]$;

[12] Recall from our discussions in Section 3.1.1 that the objective fragment of $\mathcal{OL}$ (and thus, $\mathcal{ES}$) does not enjoy the Compactness property for the full language.

3. for every $c\theta \in \Gamma' - \Gamma$, we do as follows.

   Note that $c\theta$ mentions at most k names not appearing in H by assumption. Since $c\theta \in \Gamma' - \Gamma$, clearly $N(c\theta)$ mentions names from $H_j^+ - H_k^+$, say l of them. But because of this, $c\theta$ does not mention at least l names from $H_k^+ - H$. So let $*$ be a bijection that swaps every name from $H_j^+ - H_k^+$ appearing in $N(c\theta)$ with l names from $H_k^+ - H$ not appearing in $N(c\theta)$, but leaves the other names unchanged.

   By construction, $c\theta \in \Gamma' - \Gamma$ because there is a $\forall$-clause $\forall(e \supset c) \in \phi$ and $\models e\theta$. Now, $\models e\theta$ iff (by Theorem 3.1.3) $\models (e\theta)^*$ iff (since e does not mention any names from $H_j^+ - H_k^+$ by assumption) $\models e\theta^*$. This implies that $c\theta^*$ is included in $gnd(\phi)$, and in particular, $c\theta^* \in \Gamma$ because after the bijection only names from H_k^+ are mentioned in the sentence. In other words, for every $f(\vec{m}) \circ n$ mentioned in $c\theta$, there is a $f(\vec{m}^*) \circ n$ mentioned in $c\theta^*$. Since $w \models c\theta$ by assumption, let $w'[f(\vec{m}), \langle\rangle] = w[f(\vec{m}^*), \langle\rangle]$ for each primitive term appearing in $c\theta$.

We have completed the construction of w'. It is easy to show (by induction) that w' satisfies Γ because of construction step (1). Similarly, it is easy to show that w' satisfies $\Gamma' - \Gamma$ because of construction step (3). Thus, w' satisfies Γ' and we are done. ∎

Note that the theorem is false if we go beyond the kind of proper$^+$ KBs dealt with in this chapter, which are finite collections of sentences of the form $\forall(\varepsilon \supset c)$ where c is a disjunction of equalities of the form $f_i(\vec{t_i}) \circ_i n_i$. What we mean to emphasize is that the restriction of the values of terms to names in c is an important one. For instance, let $\phi = \forall(x \neq 1 \supset f \neq x)$. It is easy to see that $\phi \models f = 1$ but $gnd(\phi)|H_j^+ \not\models f = 1$ for any finite j.

Example 5.5.11. We now illustrate the theorem. Let ϕ be a proper$^+$ KB and a conjunction of:

- $\forall(in(x) = box \supset status(x) \neq destroyed)$,
- $\forall(x \neq C \wedge x \neq D \supset in(x) \neq box)$,
- $in(C) \neq container$.

Suppose the query α is $in(C) = box$. Since the $\forall$-clauses in ϕ mention only a single variable, let $H_1^+ = \{C, D, box, container, E\}$, where E is the new name. Now compute $gnd(\phi)|H_1^+$, which is equivalent to a conjunction of

$in(C) = box \supset status(C) \neq destroyed, \ldots, in(E) = box \supset status(E) \neq destroyed$,

$in(box) \neq box, in(container) \neq box, in(E) \neq box$,

$in(C) \neq container$.

Pursue $\psi \doteq gnd(\phi)|H_1^+ \wedge \neg\alpha$. We list one set of assignments to the primitive terms wrt which ψ simplifies to TRUE:

- let $\{in(n) = E \mid n \in H_1^+\}$;
- let $\{status(n) = E \mid n \in H_1^+\}$;

Under this assignment, ψ is equivalent to TRUE. Equivalently, the DPLL procedure returns SAT. Therefore the query is not entailed. ∎

The second result needed to extend the scope of the DPLL procedure is an important property we covered earlier in Section 3.1 about inferring the validity of a universal. Mainly, Corollary 3.1.4 states given a theory that mentions only finitely many names, a universally quantified formula can be inferred by a finite number of substitutions. For the substitutions, we consider all the names in the theory and the query plus an arbitrary extra one. This is then combined with Theorem 5.5.10 in the following manner.

Theorem 5.5.12. *Let ϕ be a proper$^+$ KB and α any quantifier-free formula with a single free variable x. Suppose k is the maximum number of variables appearing in a $\forall$-clause in ϕ. Let H be all the names in $\phi \cup \{\alpha\}$ and H_1^+ be the union of H and an arbitrary new name. Then*

$$\phi \models \forall\alpha \quad \textit{iff} \quad gnd(\phi)|J \models \alpha_n^x$$

for all $n \in H_1^+$, where J is the union of the names in H_1^+ and k arbitrary new names.

Proof: By Corollary 3.1.4, $\phi \models \forall\alpha$ iff $\phi \models \alpha_n^x$ for all $n \in H_1^+$. By Theorem 5.5.10, $\phi \models \alpha_n^x$ iff the grounding of ϕ wrt all the names in $\phi \cup \{\alpha_n^x\}$, which may mention all the names in H_1^+, plus k new ones entails α, *i.e.* $gnd(\phi)|J \models \alpha_n^x$. ∎

Example 5.5.13. Let ϕ' be the union of ϕ from Example 5.5.11 and the following sentence

- $\forall(status(x) \neq destroyed \supset near(bomb, x) \neq 1)$.

That is, if an object is not destroyed then it it not near the bomb. From Example query 5.5.11, we also have that if an object is in the box then it is not destroyed. It is clear, then, that if an object is in the box then it is not near the bomb. So let our query be $\forall x.\alpha$ where α is $(in(x) = box \supset near(bomb, x) \neq 1)$.

Let $H_1^+ = \{C, D, box, container, 1, bomb, B\}$ where B is the new name. Let J be the union of H_1^+ and $\{E\}$. Then $gnd(\phi)|J$ is a conjunction of the following sentences:

1. $in(C) = box \supset status(C) \neq destroyed, \ldots,$
   $in(B) = box \supset status(B) \neq destroyed, in(E) = box \supset status(E) \neq destroyed,$
2. $\{in(n) \neq box \mid n \in J - \{D, C\}\},$
3. $in(C) \neq container,$
4. $status(C) \neq destroyed \supset near(bomb, C) \neq 1, \ldots,$
   $status(B) \neq destroyed \supset near(bomb, B) \neq 1, status(E) \neq destroyed \supset near(bomb, E) \neq 1.$

For $\phi' \models \forall x\alpha$ to hold, Theorem 5.5.12 says that $gnd(\phi)|J \wedge [in(x) = box \wedge near(bomb, x) = 1]_n^x$ must be unsatisfiable for every $n \in H_1^+$. It can be shown that this is indeed the case. Below, we argue with a few substitutions explaining, in each case, where conflicts occur because of which SAT is not returned. We leave it to the reader to confirm that such conflicts occur for every substitution, and therefore, every leaf of the DPLL proof tree thereof returns UNSAT.

- For all names $n \in H_1^+ - \{C, D\}$, we verify as follows. Suppose n is B. The other cases are similar. Then $\neg\alpha_B^x$ conflicts with (2) in the sense that $\{\neg\alpha_B^x, (2)\}$ is unsatisfiable. This is because $\neg\alpha_B^x$ entails that B is in the box, while (2) says that B is not in the box. In this manner, it is not hard to show that $gnd(\phi)|J \wedge \neg\alpha_n^x$ is unsatisfiable for names $n \in H_1^+ - \{C, D\}$.

- Suppose n is C. The case where n is D is similar. Pursue $gnd(\phi)|J \wedge \neg\alpha_C^x$. Observe that $\neg\alpha_C^x \models in(C) = box$. Then, from $(1) \cup (4) \cup \{in(C) = box\}$ we infer that that $near(bomb, C) \neq 1$ which, in fact, is not consistent with $\neg\alpha_C^x$, and therefore this branch of assignments does not return SAT. In this manner, it is not hard to show that $gnd(\phi)|J \wedge \neg\alpha_n^x$ is unsatisfiable for names $n \in \{C, D\}$.

Therefore the query is entailed. ∎

5.5.3 Related Work

Since we investigated a query evaluation methodology based on satisfiability for a first-order language, we provide a brief survey of existing methods.

A prototypical NP-complete problem is the satisfiability problem for a propositional clausal theory. Under some restrictions, the satisfiability problem is solvable in linear time; a well known example being propositional Horn theories [Dowling and Gallier, 1984]. Satisfiability solvers are generally based on the DPLL algorithm [Davis and Putnam, 1960; Davis et al., 1962], although in special cases, such as for Horn theories, a kind of resolution is employed.

In the early nineties, an effort was taken in the KR community to investigate the tradeoff between typical-case complexity and worst-case complexity in propositional clausal reasoning [Mitchell et al., 1992]. Owing to this effort, extensive research in satisfiability solvers has been carried out to date [Gomes et al., 2008], and a variety of extensions to the DPLL procedure have been proposed. In fact, current solvers are able to handle as many as a million variables and are used in a numerous applications [Gomes et al., 2008].

The success of satisfiability solvers has also led to many extensions which go beyond propositional logic, by considering variants of the DPLL algorithm. One of the main motivations for this line of work is to reason about mathematical constraints, such as in linear arithmetic. The language here is usually quantifier-free fragments of FOL, often with equality and functions. Some of these sublanguages allow for a natural encoding of *abstract datatypes*, such as *lists* [Shankar and Ruess, 2002], and they are found to be tremendously useful in software verification [Pnueli et al., 1999] as well as in hardware verification [Burch and Dill, 1994].

Arbitrary well-formed expressions in such quantifier-free FOL fragments with equality and functions are known as *ground term algebra*. Early proposals, such as [Ackermann, 1962], were directed towards reducing certain theories into sentences in propositional logic. But since then, many solvers that operate directly on ground term algebra appear in the literature [Shankar and Ruess, 2002; Barrett et al., 2000; Groote and van de Pol, 2000; Badban et al., 2007; Badban and van de Pol, 2005]. Solvers such as MATHSAT [Audemard et al., 2002] are further specialized for efficiently handling the propositional component of the language.

In [Baumgartner, 2000], DPLL is generalized to FOL. But the algorithm proposed there is not terminating, mainly because the satisfiability problem for first-order logic is not decidable. Moreover, equality is not considered.

Let us simply summarize all this work and point out that in contrast to these techniques, our methodology provides a different strategy for finding satisfying assignments to functions and this is made possible because our domain is fixed over standard names, which are all unique. It is also precisely this feature that allows to consider quantifiers both in the KB and the query.[13]

5.6 Concluding Remarks

In this chapter, we presented three cases where progression is first-order definable and efficiently computable, under reasonable assumptions. In particular, we first considered local-effect actions, where progression is always first-order definable. For certain kinds of expressive theories called proper$^+$ KBs it is also very efficient. Second, we considered normal actions, which are not local-effect, where is progression is first-order definable provided that the initial theory is in the semi-Horn form wrt the fluents which the actions affect in a non-local way. Moreover, for proper$^+$ KBs satisfying this requirement, progression is also efficient. Finally, we consider JIT progression for proper$^+$ KBs wrt range-restricted theories. Progressing a theory alone is not sufficient for efficient projection, and so we also introduced a query evaluation mechanism for a large class of queries against proper$^+$ KBs.

The idea behind progression is not new and lies at the heart of most planning systems, including STRIPS [Fikes and Nilsson, 1971]. As we mentioned in Section 5.1.1, Lin and Reiter [1997] were the first to provide a general account of progression in the context of the situation calculus. As part of their work, they also view STRIPS as a mechanism for computing progression.

Lin and Reiter conjectured that progression needs second-order logic in general, in the sense the progression of a theory cannot be represented simply with first-order sentences, even allowing for an infinite number of them. This conjecture was later proved by Vassos and Levesque [2008]. Lin and Reiter were also the first to consider two useful syntactical restrictions on action theories with which progression is first-order definable and efficient, including the *strictly context-free* case mentioned in Section 5.2.2.

In the interest of moving beyond the context-free assumption, Liu and Levesque [2005a] proposed *local-effect theories*, which are a strict generalization of Lin and Reiter's context-free theories. Under the strong assumption that the agent has complete knowledge about the context formula, Liu and Levesque present first-order definability and computability results for certain kinds of initial KBs called *proper KBs*. For the sake of this discussion, proper KBs correspond to a (possibly) infinite set of literals (and thus are strictly less expressive than proper$^+$ KBs).

In later work, Vassos et al. [2008] proved that the progression of an arbitrary first-order sentence wrt local-effect theories is first-order definable. They prove that for a special case, called *strictly local-effect* action theories, it is finite and can be computed. Recently, Liu and Lakemeyer [2009] generalized this result in the sense of showing that the progression of an arbitrary first-order sentence wrt local-effect action theories is indeed computable. But it may not be efficient, and so they prove that for function-free proper$^+$ KBs progression wrt local-effects is efficient. Thus, our first result generalizes the local-effects results from [Liu and Lakemeyer, 2009] to a language with functional fluents. The definability results are inspired by the same ideas as in [Liu and Lakemeyer, 2009], but in the case of functions we have to forget primitive terms

[13] Nonetheless, it remains to be seen if it leads to an implementation that is more efficient than, say, an encoding of the KB (plus, the uniqueness of names) input to suitable existing solvers.

while in the predicate-only case, one forgets primitive atoms. Moreover, in this chapter, we are interested in computing progression in terms the agent's knowledge base, that is, in terms of what is only known, along the work of [Lakemeyer and Levesque, 2009]. Consequently, we are also able to handle non-trivial sensing results which are then a part of the new knowledge base. Our techniques for computing the progression of proper$^+$ KBs efficiently is also different from the one proposed in [Liu and Lakemeyer, 2009], mainly, again due to the fact that we need to forget primitive terms while in the predicate-only case, they forget atoms.

In their paper, Liu and Lakemeyer also introduced normal actions essentially to capture cases such as moving a box or a briefcase, which not only affects the location of the container but also all the objects inside the container. They showed that provided that the initial theory is semi-Horn wrt all fluents on which a normal action has non-local effects, progression is always first-order definable. For a predicate-only proper$^+$ KB satisfying the semi-Horn assumption, they also proved that progression is efficient. Our second result generalizes their work to the case of functional fluents.[14]

Local-effects and normal actions still fall short of being able to capture actions like moving forward and the effect it has on the location of the robot. This motivated Vassos et al. [2009] to introduce the notion of range-restricted theories. More precisely, they prove computability results for DBPCs, which was discussed in Section 5.4. The technique they introduce, however, is quite different from the approach considered in this thesis. The idea there is to progress all possible models of the initial theory and propose conditions under which this can be represented efficiently. They were able to provide such a procedure because DBPCs restrict the possible values of primitive terms to a finite set. On the other hand, we had a considerably simple notion of computing progression, which was based on forgetting. But we could provide this result only because we assumed complete knowledge about the context formula (see Section 5.4.2). So, while our initial theories are in some sense much more expressive than DBPCs, the kind of theories we can deal with are in some sense more restrictive than what Vassos et al. can handle. For instance, while both the approaches can deal with the action of moving forward, only Vassos et al. are able to handle the case where the agent is uncertain about its position. This is arguably useful in many scenarios, such as our example from Figure 4.3. We believe each approach has its benefits, and depends very much on the domain we are interested in. Nevertheless, let us reiterate that, in both accounts, the JIT assumption, which captures the intuition that the bounded effects on non-local actions can be resolved using information in the knowledge base, is problematic when the action theory is used *offline*. Put differently, in settings where the action theory is used by an agent that is able to interact with the world *online* and get new information, say by means of sensing, JIT progression may prove more effective.

Both categories of non-local action theories dealt with have their limitations. Normal actions enable restrictions on the fluents that may appear in the context formula, as a result of which examples such as the exploding bomb action theory cannot be handled. On the other hand, with range-restricted theories the definability of progression depends on the JIT assumption, which we argued above can be problematic in some settings. Part of the reason why this limitation does not arise with normal actions is because we are basically adding the instantiated successor state axioms to the new theory. In the case of range-restricted theories, we *update* the values of primitive terms.

We summarize the results in the literature in the following table. Let us remark that an important second

[14] We remark that Liu and Lakemeyer also mention in their paper that they were able to extend the *first-order definability results* for local-effects and normal actions to a language with functions. But these are not published at the time of writing this thesis.

step to the practical feasibility of the progression formalism is query evaluation. When proposing the progression of predicate-only proper$^+$ KBs, Liu and Lakemeyer rely on a sound but incomplete query evaluation methodology from [Liu and Levesque, 2005b]. In this chapter, we introduced a new methodology for sound and complete reasoning for a large class of queries in the presence of function symbols.

We now review some other results regarding progression in the situation calculus. Under the strong assumption that the initial KB is complete, De Giacomo and Mancini [2004] investigate how to exploit relational database technology to do progression. In particular, they make use of database updating and query service to do progression efficiently. In their work, Shirazi and Amir [2005] investigate the computability of a certain form of first-order progression, but they leave open the cases under which progression is correct. Instead they show that provided progression is first-order definable, their version of *weak progression* is correct for certain kinds of queries. They, then, concentrate on proving results for *unit-case* actions, where the context formulas are unit clauses.

An interesting direction has been pursued in [Liu and Wen, 2011], where they provide an account of computing progression for subjective theories. More precisely, they are concerned with showing that the progression of sentences in the epistemic situation calculus [Scherl and Levesque, 2003] wrt local-effect actions is definable in first-order modal logic (**S5** in particular). The proof techniques are based on similar ideas from [Liu and Lakemeyer, 2009] and the ones used in this chapter, *viz.* by means of forgetting. However, owing to the enriched language they need to consider forgetting in modal logic [Zhang and Zhou, 2009]. For this reason, they are able to prove that progression wrt local-effects is definable in first-order modal logic for only a restricted fragment of the language. They also need to make additional restrictions about the initial KB when sensing is performed. Since the logic in [Liu and Wen, 2011] is **S5** one could argue that by using only knowing, with which we may be able to specify the agent's KB in terms of objective sentences in the scope of the $\boldsymbol{O}$ modal operator, we are able to consider the progression of a non-modal theory and do not need any additional restrictions at least wrt local-effects on the agent's knowledge base. Nevertheless, the results in [Liu and Wen, 2011] are novel, since they clarify how the ideas behind progression is to be extended for basic KBs. Further, extending the results for non-introspective logics such as **K** may prove insightful when considering agents that are not capable of full introspection.

Outside of the situation calculus, we already mentioned in Section 2.3.2 that the fluent calculus also employs a form of progression. This is essentially the result of encoding the dual of basic action theories in terms of state update axioms. Here, after doing an action, progression is based on adding a *description* of the changes between the new and the initial states. But in order to implement this methodology, one needs an inference mechanism that is able to build the new state from the old one and the descriptions. In [Thielscher, 2005], one approach is outlined based on using *logical constraints* to express the description of the change. However, it makes use of a sound but incomplete constraint solver to compute the new state.

As a closing remark, consider that for the chapter we have concerned ourselves with the single agent case, mainly due to the additional technical subtleties that arise in an account of progression in contrast to regression. We leave the multiagent case for future work. We believe that this can be obtained by combining our semantics for multiagent only knowing from the previous chapter, with the notion of progressing world states from the current chapter. Some of the main issues that need clarification are regarding the sensing results that may be distinct between the agents, and how this should be incorporated into the progressed KBs of the agents. Some preliminary work on providing an account of progression in the multiagent case

Figure 5.6: Below, (1) denotes [Liu and Levesque, 2005a], (2) denotes [Vassos et al., 2008], (3) denotes [Liu and Lakemeyer, 2009], (4) denotes [Belle and Lakemeyer, 2011b], and (5) denotes [Vassos et al., 2009].

Class	Results
Local-effect Actions	Introduced in (1) • Definability results in (2) • Computability results (function-free case) in (3) • Efficiency results for function-free proper$^+$ KBs in (3) • Computability results (with functions) in (4) • Efficiency results for proper$^+$ KBs in (4)
Normal Actions	Introduced in (3) • Definability and computability results (function-free case) in (3) • Efficiency results for function-free proper$^+$ KBs in (3) • Definability and computability results (with functions) in (4) • Efficiency results for proper$^+$ KBs in (4)
Range-restricted theories	Introduced in (5) • Definability, computability and efficiency results for DBPCs in (5) • Definability, computability and efficiency results for proper$^+$ KBs in (4), but for different assumptions about the context formula

is presented in [Liu and Wen, 2011], as an extension to their work on the progression of knowledge in the epistemic situation calculus. That is, besides restricting initial knowledge bases due their use of forgetting techniques in a modal context, only knowing is not considered.

Chapter 6

Progression under Uncertainty

In the previous chapters, we considered solutions to the projection problem. An important assumption made there is that the effectors used to execute actions and performing sensing operate deterministically. However, in realistic domains, this is typically not the case. For example, if a robot attempts to move one unit towards a wall (as in Figure 4.2), it is possible that it ends up moving only (say) .9 units, due to the inaccuracies in its effectors. Nevertheless, the robot's degree of belief that it is closer to the goal should increase. Similarly, due to cheap hardware a sensor reading of 1 unit may in reality mean that the robot is anywhere (say) in the range of .9 to 1.1 units. In this chapter, we are concerned with proposing a representation formalism that captures the reasoning required to keep the agent's beliefs contingent with what happens in the world, given such noise in the effectors of the agent. (Noisy sensors is left for future work.) Without the ability to reason with this noise, the agent will not be able to operate in its environment in any purposeful manner.

Clearly, the computational feasibility of such a formalism rests on providing a solution to the projection problem. We consider a framework closely related to Lin and Reiter's notion progression to reason about projection tasks. The idea will be to allow the knowledge base to contain both "ordinary" beliefs, by which we mean first-order sentences taken to be what the agent knows, as well as *probabilistic* ones, which in some sense reflect the agent's degrees of belief. After doing an action, ordinary beliefs can be *progressed* in a standard fashion, much like what was investigated in the previous chapter, while probabilistic beliefs can be *updated* in a computable manner. However, to achieve such results we need to restrict the kind of probabilistic beliefs in the knowledge base. Nevertheless, we believe that the case we make is of practical interest.

The rest of the chapter is organized as follows. We present a new logic whose semantics is based on independent developments on progression and reasoning about uncertainty. We then cover the semantics of progression under uncertainty, and turn to definability theorems that show how the progressed knowledge is obtained from the initial one wrt both faulty and non-faulty effectors.

6.1 The Logic $\mathcal{ES}_\mu$

In Section 2.3.1, we briefly reviewed two extensions to the situation calculus that allow for the representation of noisy effects and degrees of belief. While both formalisms are adequate to represent noisy effects, the extension by Bacchus et al. [1995] resorts to second-order logic when reasoning about probabilistic beliefs.

Moreover, despite having epistemic features, it is not even clear what the knowledge base should look like after performing actions [Gabaldon and Lakemeyer, 2007]. While some of these issues are addressed in Gabaldon and Lakemeyer [2007], the latter approach is also not without its problems. For instance, after doing a noisy actions, an agent is only allowed to reason about probabilistic beliefs, whose semantics is quite involved. But perhaps the main issue is that both the approaches do not propose a solution to the projection problem. Be that as it may, it does not seem entirely obvious how the regression operator should be defined in the presence of noisy actions and probabilistic beliefs. However, a progression-based solution can be given, as we shall investigate in this chapter, and for this purpose we introduce a new logic $\mathcal{ES}_\mu$.

Before turning to the formal aspects, let us informally see how uncertainty is represented. In the situation calculus, we say actions are deterministic in the sense that when executing an action it is typically assumed that this results in a *unique* successor state. Bacchus et al. then propose to model nondeterminism in actions by essentially mapping a noisy action to a set of primitive (deterministic) actions. The nondeterminism here is that the agent does not know exactly which primitive action was executed. But importantly, by modeling nondeterminism this way, a solution to the frame problem as proposed by Reiter [2001] can be applied wrt the execution of the underlying primitive actions individually. For these reasons, we will also capture noisy actions in $\mathcal{ES}_\mu$ using the same trick. Roughly speaking, the language will syntactically distinguish noisy actions, and models for the logic will include a mapping from noisy actions to ordinary ones. In order to represent probabilistic beliefs, the language will also include a new modal operator $\boldsymbol{B}$. For readability purposes, in what follows, we use the term "knowledge" with the modality $\boldsymbol{K}$, and use the term "beliefs" with $\boldsymbol{B}$. We reiterate that neither of the two modalities require that the agent has true beliefs.

The Language

We let symbols be taken from a vocabulary consisting of first-order variables, second-order rigid function variables, fluent and rigid functions, distinguished functions *Poss*, *prob*, *choice* and the following logical connectives: $\neg, \forall, \wedge, [v], [\![v]\!], \Box, \boldsymbol{K}, \boldsymbol{B}$ and $\boldsymbol{O}$. Note that we are essentially dropping *SF*, since we intend to leave sensing as future work. The purpose of the distinguished functions *prob* and *choice* is considered shortly.

We assume that functions and variables now come in three sorts: *object*, (ordinary) *action* and *noisy actions* with the understanding that actions are used with $[\cdot]$ and noisy actions are used with $[\![\cdot]\!]$. As an extension to our assumptions in $\mathcal{ES}$, we suppose that all actions and noisy actions are of the rigid type.

We will have three types of standard names:

- $\mathcal{N}$ is a countably infinite set of object names, such as ${}^{\#}0, {}^{\#}1, \ldots, \mathit{obj5}, \ldots$ $\mathcal{N}$ includes the set of rational numbers $\mathbb{Q}$ closed under standard arithmetical operators $+, -, \times, \div$. Let $\mathbb{Q}_{[0,1]}$ denote the subset of $\mathbb{Q}$ between 0 and 1 inclusive.

- Let $\mathcal{A} = \{A(m_1, \ldots, m_k) \mid m_i \in \mathcal{N}$ and A is a function of the action sort$\}$ be the set of action names, *e.g.* $\mathit{drop}(\mathit{obj5}), \mathit{forward}$.

- Let $\mathcal{S} = \{A(m_1, \ldots, m_k) \mid m_i \in \mathcal{N}$ and A is a function of the noisy action sort$\}$ be the set of noisy action names, *e.g.* *noisyReverse*.

Now, let $\mathcal{Q} = \mathcal{N} \cup \mathcal{A} \cup \mathcal{S}$. Thus, the language includes an additional class of names compared to $\mathcal{ES}$. We

now define terms and formulas in the same way as in the previous chapters, extended to the new sort in an obvious way.

Terms

Terms are of the sort action, noisy action or object, and they are the least set of expressions such that:

- Every first-order variable and name is a term.
- If $\vec{t}$ is a vector of terms of the object sort and A is a function of the action or the noisy action sort, then $A(\vec{t})$ is a term.
- If $\vec{t}$ is a vector of terms of any sort and f is a function of the object sort, then $f(\vec{t})$ is a term.
- If $\vec{t}$ is a vector of terms and P is a second-order variable then $P(\vec{t})$ is a term.

By *primitive term,* we mean one of the form $f(\vec{m})$ where $m_i \in \mathcal{Q}$. By *primitive second-order term,* we mean one of the form $P(\vec{m})$ where $m_i \in \mathcal{Q}$.

Formulas

The well-formed formulas of the language are:

- if t and t' are terms then $t = t'$ is a formula;
- if x is a first-order variable, P is a second-order variable, α and β are formulas, then so are $\alpha \wedge \beta, \neg\alpha, \forall x\alpha, \forall P\alpha, [t]\alpha, [\![t]\!]\alpha, \Box\alpha, \boldsymbol{K}\alpha, \boldsymbol{O}\alpha$;
- if α is a formula then $\boldsymbol{B}\alpha \geq b$, where $b \in \mathbb{Q}_{[0,1]}$, is a formula.

As in the previous chapter, we will make the restriction (and assume henceforth) that second-order quantifiers are *only* applied to formulas that do not mention $\boldsymbol{K}$ and $\boldsymbol{O}$.

For the new language, by a *fluent formula* we will mean those that do not mention *Poss*, *prob*, *choice*, v, $[\![v]\!]$, $\Box$, $\boldsymbol{K}$, $\boldsymbol{O}$ and $\boldsymbol{B}$. We will syntactically restrict formulas appearing in the scope of $\boldsymbol{B}$ to be fluent formulas. We refer to formulas of the form $\boldsymbol{B}\alpha \geq b$ as *probability* or *belief* atoms.

We read $[t]\alpha$, $\boldsymbol{K}\alpha$ and $\boldsymbol{O}\alpha$ as before. We read $[\![t]\!]\alpha$ as "α holds after the noisy action t". We read $\boldsymbol{B}\alpha \geq b$ as "α is believed with a probability $\geq b$".

We remark that $\boldsymbol{B}\alpha \geq b$ is the only kind of belief inequality expression in the language. But this is without any loss of generality, since we can express other inequalities in terms of $\geq$ as follows:

- $\boldsymbol{B}\alpha = b$ is an abbreviation for $\boldsymbol{B}\alpha \geq b \wedge \boldsymbol{B}\neg\alpha \geq 1 - b$;
- $\boldsymbol{B}\alpha > b$ is an abbreviation for $\boldsymbol{B}\alpha \geq b \wedge \neg(\boldsymbol{B}\alpha = b)$;
- $\boldsymbol{B}\alpha < b$ is an abbreviation for $\boldsymbol{B}\neg\alpha > 1 - b$.

The Semantics

We begin by defining $\mathcal{Z}$ as the set of all finite sequences of names from $\mathcal{A}$, including $\langle\rangle$. The set of all possible worlds $\mathcal{W}$ are defined as before, that is, where a world is a function:

- from primitive object terms and $\mathcal{Z}$ to $\mathcal{N}$, and
- from primitive second-order terms to $\mathcal{N}$.

The initial beliefs of the agent is given by an epistemic state $e \subseteq \mathcal{W}$ which is any set of worlds.

Terms are interpreted as follows. As usual, names are rigid designators. We extend the idea of co-referring names for arbitrary terms as follows. Given a term t, a world w, and an action sequence $z \in \mathcal{Z}$, we define $|t|^z_w$ by:

- if t is a name then $|t|^z_w = t$;
- if f is a function of the object sort and $\vec{t}$ is a vector of names, then $|f(\vec{t})|^z_w = w[f(\vec{n}), z]$ where $n_i = |t_i|^z_w$ for terms t_i in $\vec{t}$;
- if A is a function of the action or the noisy action sort, then $|A(\vec{t})|^z_w = A(\vec{n})$ where $n_i = |t_i|^z_w$ for terms t_i in $\vec{t}$;
- if P is a second-order variable then $|P(\vec{t})|^z_w = P(\vec{n})$ where $n_i = |t_i|^z_w$.

When $z = \langle\rangle$, we write $|t|_w$ instead of $|t|^{\langle\rangle}_w$.

Readers will notice that we have simply extended our previous notions regarding co-referring names for the new sort in an obvious way. Second-order variables are understood as before, and we use the notation $w \sim_P w'$ to mean that w and w' agree on everything except maybe assignments involving P.

To reason about uncertainty, we now begin with some definitions. First, to reason about noisy actions, we introduce functions $\Pr$ and Π. For every $t \in \mathcal{N}$, $\Pi(t) = t$ and for every $s \in \mathcal{S}$, $\Pi(s)$ is a finite set of names from $\mathcal{N}$. That is, Π essentially maps actions to actions, and maps noisy actions to a set of actions. Intuitively, it models the nondeterminism in noisy actions. With this in hand, we let $\Pr(v) : \Pi(v) \to \mathbb{Q}_{[0,1]} - \{0\}$ be a probability distribution *i.e.* $\sum \Pr(v) = 1$, which essentially maps the choices of noisy actions to strictly positive probabilities. We will shortly see that they are interpreted in the language by means of *choice* and *prob* respectively.

Next, to interpret belief atoms over $\mathcal{W}$, we introduce the notion of a probability space [Halmos, 1950; Fagin and Halpern, 1994].

Definition 6.1.1. (Probability space.) A probability space is a tuple $(\mathcal{D}, \mathcal{X}, \mu)$ where $\mathcal{D}$ is a set called the *sample space*, $\mathcal{X}$ is a σ-algebra of subsets of $\mathcal{D}$ (*i.e.* a set of subsets containing $\mathcal{D}$ and closed under complementation and countable union), and a measure $\mu : \mathcal{D} \to [0, 1]$ satisfying the following two properties:

1. $\mu(\emptyset) = 0$ and $\mu(\mathcal{D}) = 1$;
2. if A and B are disjoint elements of $\mathcal{X}$, then $\mu(A \cup B) = \mu(A) + \mu(B)$. ∎

For our purposes it is sufficient to restrict ourselves to probability spaces that satisfy:

M1. $\mathcal{X} = 2^{\mathcal{D}}$, *i.e.* $\mathcal{X}$ consists of all possible sets of the sample space;

M2. $\mu : \mathcal{X} \to \mathbb{Q}_{[0,1]}$, *i.e.* the measure is restricted to the space of rationals which is reasonable for practical applications;

M3. $\mu(X) > 0$ for all $X \in \mathcal{X}$ such that $X \neq \emptyset$.

The idea is, much like Fagin and Halpern [1994], to associate each world $w \in \mathcal{W}$ with a probability space. So suppose we have a function that maps w to the probability space $(\mathcal{D}^w, \mathcal{X}^w, \mu^w)$ where $\mathcal{D}^w \subseteq \mathcal{W}$. That is, at each world w, the agent imagines a sample space consisting of possible worlds. The argument then is whether a natural definition can be specified for precisely which set of worlds are in $\mathcal{D}^w$. Fagin and Halpern argue that this set must necessarily be a subset of the worlds considered epistemically possible from w, since it is unintuitive for the agent to assign positive probabilities to worlds that he does not consider epistemically possible. We go further, and say that $\mathcal{D}^w$ is precisely the set of worlds considered epistemically possible at w.[1] Uncertainty is then interpreted wrt the agent's initial beliefs, and this is what we will need. Moreover, since the set of epistemically possible worlds is fixed by e, it follows that we need to only consider a single probability space whose sample space is e.[2]

However, owing to the language, e may be uncountable. Instead of working with an infinite sample space [Halpern, 2003], we use a notion from [Gabaldon and Lakemeyer, 2007] and reduce e to a finite set of equivalence classes of worlds. So let $\mathcal{F}$ be a finite set of fluents and H be a finite set of names. Then let

$$\Delta = \{f(\vec{m}) \mid f \in \mathcal{F}, m_i \in H\}.$$

Intuitively, the idea is to assume that $\mathcal{F}$ includes all the fluents over which we define a basic action theory, and this sublanguage represents every fluent and name that the agent encounters during its operation.

We write $w \approx w'$ to mean that for all fluent primitive terms d from Δ, $w[d, \langle\rangle] = w'[d, \langle\rangle]$. Now, given an epistemic state e, define

$$\|w\| = \{w' \mid w' \in e, \text{ and } w' \approx w\}$$

as the set of epistemically possible worlds that agree on Δ initially.[3] Analogously, for any $e' \subseteq e$, let $\|e'\| = \{\|w\| \mid w \in e'\}$ which is always finite. Putting this together, let us now define a single probability space: $(\|e\|, 2^{\|e\|}, \mu)$.

We are now ready to define the notion of a model for the logic. It has the following components:

- an epistemic state e;
- the real world w;
- a measure μ that assigns a probability to all possible subsets of $\|e\|$;

[1] Fagin and Halpern [1994] point out that the definition of $\mathcal{D}^w$ depends on the application. Thus, there are situations where letting $\mathcal{D}^w$ be the set of all epistemically possible worlds is perhaps not appropriate. But for our purposes, this assumption seems reasonable. Moreover, it greatly simplifies the technical treatment.

[2] As argued by Gabaldon and Lakemeyer [2007], this often leads to agents holding precise beliefs about every formula. But consider a basket of oranges and apples, where their proportion is not clear. Then, the agent may not be able to assign an exact probability to the event "a selected fruit is orange". One remedy is to allow a set of measures to capture the entire range of possibilities. We ignore such issues for simplicity.

[3] Note that equivalence classes are understood wrt a particular epistemic state. But since it will always be clear from the context which epistemic we mean, we avoid the notational clutter.

- and the pair $\langle \Pi, \Pr \rangle$, which we denote by δ.

Readers will notice that besides the new notions to reason about uncertainty, different from $\mathcal{ES}$, the model does not include an action sequence $z \in \mathcal{Z}$. This is because we are proposing a semantics based on progression by means of which all the above components are updated when doing an action.

The main purpose of the semantics is to clarify how fluents and belief atoms are to be understood. The account given is closely related to earlier work by Lakemeyer and Levesque [2009], who define a notion of progressing worlds and epistemic states, and Bacchus et al. [1995] who gave an account of how probabilities should be assigned to successor states.

- Recap the first idea from Definition 5.1.1. Different from that definition, however, we will not consider the compatibility relation $\simeq_z$. To be precise:

  Definition 6.1.2. (Progression of a set of worlds.) Suppose w is a world. Let w_r be a world such that $w_r[p, z'] = w[p, r \cdot z']$ for all primitive terms p and action sequences z'. Given any set of worlds e, let $e_r = \{w'_r \mid w' \in e\}$.

  We say that w_r is the *progression* of w wrt r and that e_r is the *progression* of e wrt r. ∎

- According to the second idea, probabilities on a situation are transferred to successor situations when ordinary actions are performed, and are weakened by the probability of the particular choice of action on doing noisy actions. More precisely, if situation s has a probability of b and a is a ordinary action, then $do(a, s)$ has the same probability as s. In the case of noisy actions, which are nondeterministic by definition, it may be that executing a noisy action is equivalent to executing a_1 or a_2, both of which are ordinary actions, with probabilities b_1 and b_2. Then the probability on the situation $do(a_1, s)$ is $b \times b_1$. This is illustrated in Figure 6.1.

In a sense, this intuition roughly tells us that the progression of worlds in $\|w\|$ must obtain the same probability as $\|w\|$ when ordinary actions are executed. (Analogously, for noisy actions.) Unfortunately, this would make the notion ill-defined in our case, mainly because even if w and w' are two different worlds, w_r and w'_r, where r is an action, may be identical. As a consequence, if w and w' belong in two different equivalence classes, this no longer holds for the progressed worlds. Be that as it may, this is not a bug. With progression, we are essentially forgetting the past but in the case of Bacchus et al., the initial theory and hence the initial situations are kept around.

It turns out that the only technical device we need is the idea of *normal worlds*, which is not only a simple notion but also fits very well with purpose of this chapter, as we shall shortly see.

Definition 6.1.3. (Normal worlds.) Let w and w' be any two worlds, and suppose $w \approx w'$. The worlds are said to be *normal* if $w_z \approx w'_z$ for all $z \in \mathcal{Z}|H$ which is the restriction of $\mathcal{Z}$ to all sequences that only mention names of the action sort from H.

A *normal epistemic state* is any set of normal worlds. ∎

That is, this definition says that if normal worlds belong in an equivalence class, then the same holds for their progressed versions. With this in hand, we define the progression of models of the logic, which is central to our semantics.

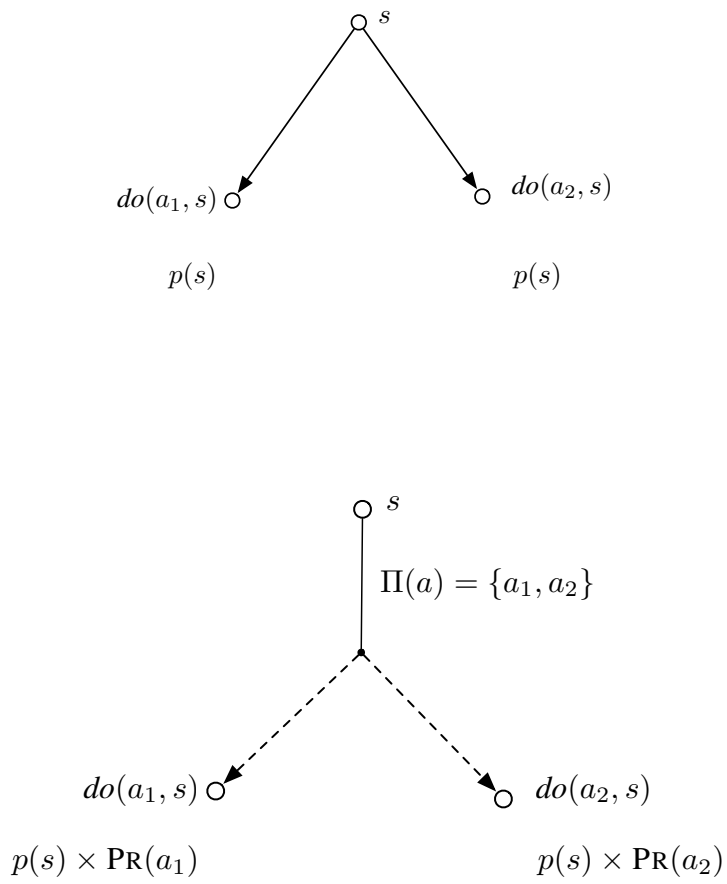

Figure 6.1: This illustrates the probabilities on successor situations after the execution of ordinary and noisy actions. In the former, the probabilities remain the same, that is, the probability on *do*(*a*, *s*) is the probability on *s*, given by *p*(*s*). In the case of the latter, the probabilities on the successor situations are weakened depending on the choice for the noisy action and the probability of that choice being executed.

Definition 6.1.4. (Progression wrt actions.) Given a model $M = (e, w, \mu, \delta)$, where e is normal, its progression wrt a primitive action r is $(e_r, w_r, \mu_r, \delta)$:

- w_r and e_r are as in Definition 6.1.2;
- for $w^* \in e_r$,
  - let $\|w^*\| = \{w' \in e_r \mid w' \approx w^*\}$;
  - let $\mu_r(\|w^*\|) = \mu(\bigcup_{\{w' \in e | w'_r \approx w^*\}} \|w'\|)$. ∎

Observe that the notion of equivalence classes is adapted for the new epistemic state in a natural way. Essentially, what Definition 6.1.4 says is that if we progress worlds in e different classes may merge since they end up agreeing on Δ, in which case a sum of the weights on the earlier classes must apply to the merged one in order to maintain normalization. Thus, we maintain the intuitions of Bacchus et al. in our definition but while taking the progression of worlds and their equivalence classes into account. Here is how we extend this idea and define the progression of models wrt noisy actions:

Definition 6.1.5. (Progression wrt noisy actions.) Given M as above, its progression wrt $s \in \mathcal{S}$, where $\Pi(s) = \{r_1, \ldots, r_k\}$ is $(e_s, w_{r_i}, \mu_s, \delta)$:

- let w_{r_i} and e_{r_i} are as in Definition 6.1.2;
- let $e_s = \bigcup_i e_{r_i}$;
- for $w^* \in e_s$,
  - let $\|w^*\| = \{w' \in e_s \mid w' \approx w^*\}$;
  - let $\mu_s(\|w^*\|) = \sum_i \mu(\bigcup_{\{w' \in e | w'_{r_i} \approx w^*\}} \|w'\|) \times \text{PR}(r_i)$.[4] ∎

This definition follows the same principles as Definition 6.1.4 except to incorporate the nondeterminism in noisy actions. That is, if we let $\Pi(s)$ be a single action then this is simply Definition 6.1.4 once again. More generally, it says that the probability assigned to $\|w_{r_i}\|$ is essentially the probability assigned to $\|w\|$ weakened by a factor of $\text{PR}(r_i)$, while taking the merging of equivalence classes into account. It is easy to see that here too we maintain the intuitions of Bacchus et al. when progressing models. It is worth noting that δ which stands for the pair $\langle \Pi, \text{PR} \rangle$ does not change during the progression of models.

One desirable property that we get from Definition 6.1.4 and Definition 6.1.5 is that both $\mu_r(\|e_r\|)$ and $\mu_s(\|e_s\|)$ are always 1. More precisely,

Proposition 6.1.6. *Let $M = (e, w, \mu, \delta)$ be a model.*

1. *Suppose $r \in \mathcal{A}$. Then, $(\|e_r\|, 2^{\|e_r\|}, \mu_r)$ is a probability space and satisfies* **M1** − **M3**.
2. *Suppose $s \in \mathcal{S}$. Then $(\|e_s\|, 2^{\|e_s\|}, \mu_s)$ is a probability space and satisfies* **M1** − **M3**.

[4] Here we mean $\text{PR}(s)[r_i]$, *i.e.* the probability assigned to r_i by the distribution $\text{PR}(s)$. We abbreviate this as $\text{PR}(r_i)$ for readability.

Proof: The argument turns out to be quite simple. To show item (1), for ease of exposition, we suppose that e is made up of two equivalence classes, $\|w^*\|$ and $\|w^{**}\|$. The case of k classes is straightforward but tedious.

We begin by noting that since e is normal, it follows that for every pair of worlds $w', w'' \in e$ such that $w' \approx w'$, it also holds that $w'_r \approx w''_r$. Thus, e_r is also normal. Further, on considering $\|w_r\|$, it follows the progression of every world from $\|w\|$ is in $\|w_r\|$. Then there are two possibilities:

1. Suppose $w^*_r \not\approx w^{**}_r$. Then, the following hold:
   - For any $w' \in \|w^*_r\|$ and any $w'' \in \|w^{**}_r\|$, $w' \not\approx w''$ because we are dealing with normal worlds.
   - $\|e_r\|$ consists of two elements, *i.e.* $\|e_r\| = \{\|w^*_r\|, \|w^{**}_r\|\}$.
   - $\mu_r(\|w'_r\|) = \mu(\|w'\|)$ for $w' \in \{w^*, w^{**}\}$. This means that $\mu_r(\|e_r\|) = \mu_r(\|w^*_r\| \cup \|w^{**}_r\|) = \mu(\|w^*\| \cup \|w^{**}\|) = \mu(\|e\|) = 1$.
2. Suppose $w^*_r \approx w^{**}_r$. Then the following hold:
   - For every $w', w'' \in e_r$, $w' \approx w''$.
   - $\|e_r\|$ consists only of a single element, *i.e.* $\|e_r\| = \{\|w^*_r\|\}$.
   - $\mu_r(\|w^*_r\|) = \mu(\|w^*\| \cup \|w^{**}\|) = \mu(\|e\|) = 1$.

Thus we obtain a probability space satisfying the desired properties for item (1).

To now show item (2), for ease of exposition, we will assume e is made up of a single equivalence class, say $\|w^*\|$, and that $\Pi(s) = \{r_1, r_2\}$. (The case where e may correspond to k equivalence classes and where $\Pi(s) = \{r_1, \ldots, r_{k'}\}$ is straightforward but tedious. Mainly, we will have to consider the progression of each of k equivalence classes wrt each of k' primitive actions.) As we have done above, it is easy to argue that e_s is normal if e is. Further, for every pair of worlds $w, w' \in \|w^*\|$, it follows that $w_{r_i} \approx w'_{r_i}$. Thus, we have two cases

1. Suppose $w^*_{r_1} \not\approx w^*_{r_2}$. Then the following hold:
   - for any $w' \in \|w^*_{r_1}\|$ and $w'' \in \|w^*_{r_2}\|$, $w' \not\approx w''$.
   - $\|e_s\|$ consists of two elements, *i.e.* $\|e_s\| = \{\|w^*_{r_1}\|, \|w^*_{r_2}\|\}$.
   - $\mu_s(\|w^*_{r_i}\|) = \mu(\|w^*\|) \times \Pr(r_i)$. This means that $\mu_s(\|e_s\|) = \mu(\|w^*\|) \times \Pr(r_1) + \mu(\|w^*\|) \times \Pr(r_2) = \mu(\|w^*\|) = \mu(\|e\|) = 1$.
2. Suppose $w^*_{r_1} \approx w^*_{r_2}$. Then the following hold:
   - for any $w', w'' \in e_s$, $w' \approx w''$.
   - $\|e_s\|$ is a singleton, *i.e.* $\|e_s\| = \{\|w^*_{r_1}\|\}$.
   - $\mu_s(\|w^*_{r_i}\|) = \mu(\|w^*\|)$. Clearly then $\mu_s(\|e_s\|) = \mu(\|e\|) = 1$. ∎

Given a model $M = (e, w, \mu, \delta)$ and any $z = \langle r_1 \cdot \ldots \cdot r_k \rangle$, define $(e_z, w_z, \mu_z, \delta)$ as the result of progression M wrt z in an iterative manner. The complete semantic definition is:

1. $e, w, \mu, \delta \models t_1 = t_2$ iff n_1 and n_2 are the same where $|t_i|_w = n_i$;

2. $e, w, \mu, \delta \models \neg\alpha$ iff $e, w, \mu, \delta \not\models \alpha$;
3. $e, w, \mu, \delta \models \alpha \vee \beta$ iff $e, w, \mu, \delta \models \alpha$ or $e, w, \mu, \delta \models \beta$;
4. $e, w, \mu, \delta \models \forall x\alpha$ iff $e, w, \mu, \delta \models \alpha^x_n$ for all names of the appropriate sort;
5. $e, w, \mu, \delta \models \forall P\alpha$ iff $e, w', \mu, \delta \models \alpha$ for every $w' \sim_P w$;[5]
6. $e, w, \mu, \delta \models [t]\alpha$ iff $e_r, w_r, \mu_r, \delta \models \alpha$ where $|t|_w = r$;
7. $e, w, \mu, \delta \models [\![t]\!]\alpha$ iff $e_s, w_{r_i}, \mu_s, \delta \models \alpha$ for all $r_i \in \Pi(s)$, where $|t|_w = s$;
8. $e, w, \mu, \delta \models \mathit{choice}(t, t') = 1$ iff $r \in \Pi(s)$, where $|t|_w = s$ and $|t'|_w = r$;
9. $e, w, \mu, \delta \models \mathit{prob}(t, t') = b$ iff $r \in \Pi(s)$ and $\textsc{Pr}(r) = b$, where $|t|_w = s$ and $|t'|_w = r$;
10. $e, w, \mu, \delta \models \Box\beta$ iff $e_z, w_z, \mu_z, \delta \models \alpha$ for all $z \in \mathcal{Z}$;
11. $e, w, \mu, \delta \models \boldsymbol{K}\alpha$ iff for all $w' \in e$, $e, w', \mu, \delta \models \alpha$;
12. $e, w, \mu, \delta \models \boldsymbol{O}\alpha$ iff for all w', $w' \in e$ iff $e, w', \mu, \delta \models \alpha$;
13. $e, w, \mu, \delta \models \boldsymbol{B}\alpha \geq b$ iff $\mu(\|[\alpha]_e\|) \geq b$;

where, for the fluent formula α,

$$[\alpha]_e = \{w \mid w \models \alpha, w \in e\}.$$

That is, the semantics for the belief atoms is specified by obtaining the sum of the probabilities on all worlds that satisfy α.

We say a sentence is true for (e, w, μ, δ) if $e, w, \mu, \delta \models \alpha$. Given a set of sentences Σ, we write $\Sigma \models \alpha$ if for every normal e, w, μ, δ such that $e, w, \mu, \delta \models \alpha'$ for every $\alpha' \in \Sigma$, then $e, w, \mu, \delta \models \alpha$. Finally, we write $\models \alpha$ to mean $\{\} \models \alpha$.

Properties

It is easy to verify that the semantics inherits all of the properties of $\mathcal{ES}$. More precisely, we note that only knowing a formula also implies knowing that formula:

$$\models \Box(\boldsymbol{O}\alpha \supset \boldsymbol{K}\alpha).$$

Meanwhile, $\boldsymbol{K}$ also has the usual properties regarding positive and negative introspection:

$$\models \Box(\boldsymbol{K}\alpha \supset \boldsymbol{K}\boldsymbol{K}\alpha);$$

$$\models \Box(\neg\boldsymbol{K}\alpha \supset \boldsymbol{K}\neg\boldsymbol{K}\alpha).$$

So what we will focus on are the additional properties of the logic, mainly concerning the relationship between knowledge and belief.

[5] Recall our discussion from Section 5.1 that our semantics for second-order quantifiers works as intended only when α does not mention $\{\boldsymbol{K}, \boldsymbol{O}\}$, which is the case by way of our syntactic restriction.

Proposition 6.1.7. *Let α and β be fluent sentences. Then the following sentences are valid:*

1. $\Box(\boldsymbol{K}\alpha \supset \boldsymbol{B}\alpha \geq b)$ *for every* $0 \leq b \leq 1$;
2. $\Box(\boldsymbol{B}\alpha \geq b \supset \neg\boldsymbol{K}\neg\alpha)$ *for every* $0 < b \leq 1$;
3. $\Box(\boldsymbol{B}(\alpha \wedge \beta) \geq b_1 \wedge \boldsymbol{B}(\alpha \wedge \neg\beta) \geq b_2 \supset \boldsymbol{B}\alpha \geq b_1 + b_2)$.

Proof: Let $M = (e, w, \mu, \delta)$ be a model.

1. Suppose $M \models \boldsymbol{K}\alpha$. Then $[\alpha]_e = e$. Clearly then $\mu(\|[\alpha]_e\|) \geq b$ for every $0 \leq b \leq 1$ since by definition $\mu(\|e\|) = 1$. Therefore $M \models \boldsymbol{B}\alpha \geq b$.
2. Suppose $M \models \boldsymbol{B}\alpha \geq b$ for some b such that $0 < b \leq 1$. This implies that $\mu(\|[\alpha]_e\|) > 0$, that is, there is some $w' \in e$, such that $e, w', \mu, \delta \models \alpha$. Therefore $M \not\models \boldsymbol{K}\neg\alpha$.
3. Suppose $M \models \boldsymbol{B}(\alpha \wedge \beta) \geq b_1 \wedge \boldsymbol{B}(\alpha \wedge \neg\beta) \geq b_2$. Let $e_1 = [\alpha \wedge \beta]_e$ and let $e_2 = [\alpha \wedge \neg\beta]_e$. Note that $[\alpha]_e = e_1 \cup e_2$, and e_1 and e_2 are disjoint. By assumption, $\mu(\|e_1\|) \geq b_1$ and $\mu(\|e_2\|) \geq b_2$. Recall the property of probability spaces where $\mu(A \cup B) = \mu(A) + \mu(B)$ if A and B are disjoint. Therefore $\mu(\|[\alpha]_e\|) \geq b_1 + b_2$, that is, $M \models \boldsymbol{B}\alpha \geq b_1 + b_2$. ∎

These properties essentially tell us that knowing a formula implies believing a formula with a probability $\geq b$ for all $b \geq 0$. Conversely, believing a formula with a strictly positive probability implies that the negation of the formula is not known. The last property discusses the additivity of probabilities over beliefs, similar to [Fagin and Halpern, 1994], which holds after any sequence of actions.

We end our discussion of $\mathcal{ES}_\mu$ by proving a few useful lemmas about the progression of models. The first says that the probability assigned to the equivalence classes consisting of a set of worlds is less than or equal to the probability assigned to the equivalence class consisting of the progressed worlds.

In what follows, when convenient, we often treat an equivalence class simply as a set of worlds. Further, given a set of worlds $W \subseteq e$, we write $(W)_r$ to mean $\{w_r \mid w \in W\}$.

Lemma 6.1.8. *Suppose $M = (e, w, \mu, \delta)$ is a model and r is a primitive action. Let $M_r = (e_r, w_r, \mu_r, \delta)$ be the progression of M wrt r. For any $w^* \in e$, $\mu_r(\|w_r^*\|) \geq \mu(\|w^*\|)$.*

Proof: The proof is quite straightforward. The idea is that since e is normal, all worlds in $\|w^*\|$ also belong in $\|w_r^*\|$ by definition. That is, for any $w', w'' \in \|w^*\|$, it follows that $w' \approx w''$ and that $w'_r \approx w''_r$. Now by the definition of μ_r, the probability on $\|w_r^*\|$ is $\mu(\|w^*\|)$ plus other equivalence classes in e, say $\|w'\|$, such that $w'_r \approx w_r^*$. Therefore $\mu_r(\|w_r^*\|)$ is at least $\mu(\|w^*\|)$, if not greater. ∎

We obtain a simple corollary thereof:

Corollary 6.1.9. *Let M and M_r be as above. For any set of worlds $W \subseteq e$, $\mu_r(\|(W)_r\|) \geq \mu(\|W\|)$.*

Proof: Suppose $\|W\| = W_1 \cup \ldots \cup W_k$, where W_i are equivalence classes $\|w^*\|$ for $w^* \in W$. Since W_i and W_j are disjoint for every i, j provided $i \neq j$, it follows that $\mu(\|W\|) = \sum_i b_i$ where $b_i = \mu(W_i)$. Now consider $\|w_r^*\|$ for some $w^* \in W$. By the arguments from Lemma 6.1.8, it follows that $\mu_r(\|w_r^*\|)$ is $\mu(\|w^*\|)$ plus the

probability of other classes $\|w'\|$ such that $w' \in e$ and $w'_r \approx w^*_r$. Now, if $\|w'\| \subseteq \|W\|$, *i.e.* if $\|w'\|$ is some W_j, then clearly $\|w^*_r\|$ is at least $b_i + b_j$. (That is, even if $\|w'_r\|$ and $\|w^*_r\|$ are the same, $\mu_r(\|w^*_r\|)$ will be at least $\mu(\|w'\| \cup \|w^*\|)$.) Otherwise, if $\mu(\|w'\|) = b'$, then $\|w^*_r\|$ is at least $b_i + b'$. Thus, $\|(W)_r\|$ is at least $\sum_i b_i$, if not greater. ∎

The second lemma of interest is to show that there is either a one to one or many to one correspondence between the equivalence class of e and e_r.

Lemma 6.1.10. *Suppose $M = (e, w, \mu, \delta)$ and M_r are as above. Consider the equivalence classes in e and e_r. That is, suppose W_i denotes an equivalence class in e, say $\|w^*\|$ for some $w^* \in e$, and suppose $e = W_1 \cup \ldots \cup W_k$. Similarly, suppose $e_r = W'_1 \cup \ldots \cup W'_{k'}$. Then for any W_i and W'_j*

1. *either $(W_i)_r = W'_j$,*
2. *or $(W_i)_r \subset W'_j$, and in this case, $(W_i \cup W_{h1} \ldots \cup W_{hk})_r = W'_j$ for some $h1, \ldots, hk$.*

Proof: Let $W_i = \|w^*\|$ for some $w^* \in e$. Note that for any $w', w'' \in \|w^*\|$, it follows that $w' \approx w''$, and since the worlds are normal, $w'_r \approx w''_r$. It then follows that the progression of every world in W_i is also in $\|w^*_r\|$. Of course, there may be other classes $\|w'\|$ for $w' \in e$ such that $w' \not\approx w^*$ but $w'_r \approx w^*_r$, which means that the progression of the worlds from $\|w'\|$ are in $\|w^*_r\|$ as well.

Now, since $\|w^*_r\|$ is some W'_j, we obtain $W'_j \supseteq (W_i)_r$. That is, either $W'_j = (W_i)_r$, thereby showing item (1), or $W'_j \supset (W_i)_r$ and this case W'_j contains the progressed versions of some other equivalence classes. Since there are only finitely many equivalence classes, this then shows item (2). ∎

The final lemma of interest is a simple one regarding noisy actions, and can be seen as analogue to Lemma 6.1.8. Here we prove that the probability on an equivalence class is weakened after progression.

Lemma 6.1.11. *Suppose M is as above and let M_s be the progression of M wrt s. Suppose $\Pi(s) = \{r_1, \ldots, r_k\}$. For any $w^* \in e$, $\mu_s(\|w^*_{r_i}\|) \geq \mu(\|w^*\|) \times \Pr(r_i)$.*

Proof: Consider some $w^* \in e$. By the definition of μ_s, it follows that the probability assigned to $\|w^*_{r_i}\|$ is obtained by considering the progression of every $w' \in e$ such that $w'_{r_j} \approx w^*_{r_i}$ weakened by a factor of $\Pr(r_j)$, for every j. Clearly this will at least consider the progression of all the worlds from $\|w^*\|$ wrt r_i. Thus, $\mu_s(\|w^*_{r_i}\|)$ is at least $\mu(\|w^*\|) \times \Pr(r_i)$, if not greater. ∎

We obtain the following corollary as an analogue to Corollary 6.1.9:

Corollary 6.1.12. *Let $M = (e, w, \mu, \delta)$ and M_s be as above. For any set of worlds $W \subseteq e$, $\mu_s(\|(W)_{r_i}\|) \geq \mu(\|W\|) \times \Pr(r_i)$.*

Proof: The formal arguments follow Corollary 6.1.9 while considering the progression of the worlds wrt the different choices for s as in Lemma 6.1.11. ∎

6.2 The Semantics of Progression

We begin by considering the equivalent of situation calculus basic action theories. These are essentially the same as the ones considered in Definition 4.1.2, with the exception of two additional components that axiomatize the uncertainty in the domain.

6.2.1 Basic Action Theories

Definition 6.2.1. (Basic action theory.) Given a set of fluents $\mathcal{F}$, a set $\Sigma \subseteq \mathcal{ES}_\mu$ is called the *basic action theory* over $\mathcal{F}$ if it is the union of:

- $\Sigma_0, \Sigma_{pre}, \Sigma_{post}$ as in Definition 4.1.2;
- Σ_Π is a sentence of the form $\Box choice(x, y) = 1 \equiv \lambda$ where λ is a fluent formula only mentioning variables and names;
- Σ_{PR} is a sentence of the form $\Box prob(x, y) = u \equiv \eta$ where η is a fluent formula only mentioning variables and names. ∎

That is, Σ_Π and Σ_{PR} capture the nondeterminism in noisy actions and in a sense, axiomatically model Π and PR respectively.[6]

As in the previous chapters, we assume that a basic action theory is all that the agent knows.[7] We often denote the initial theory as ϕ and denote the rest as $\Box\beta$. But in addition to what the agent knows in terms of an action theory, it may have a number of probabilistic beliefs, which we represent as a conjunction of belief atoms. Putting all of this together, in what follows we will concern ourselves with a background theory T of the form:

$$\boldsymbol{O}(\phi \wedge \Box\beta) \wedge \bigwedge \boldsymbol{B}\alpha \geq b.$$

Example 6.2.2. (The simple robot domain reconsidered.) Let us illustrate the idea of a basic action theory and the corresponding theory with an example. We reconsider the robot domain from Example 4.1.3 with the following changes. We assume that in addition to moving forwards, the robot is also capable of moving backwards. However, in contrast to moving forward, we suppose that the reverse mechanism is noisy. More precisely, the robot can execute *noisyReverse* which may result in an actual reverse, with a success rate of .9, or the robot may end up just staying in place. The domain is formalized in Figure 6.2. Let $\Sigma = \Sigma_0 \cup \Sigma_{pre} \cup \Sigma_{post} \cup \Sigma_\Pi \cup \Sigma_{\text{PR}}$.

[6] In other accounts involving noisy sensors, such as [Gabaldon and Lakemeyer, 2007], the *rhs* of the equivalent of these axioms are allowed to be arbitrary fluent formulas. To see an example that can be axiomatized with this feature, think of having a robot whose effectors are noisy in the sense that moving forward by a unit results in moving forward by either 0 unit or 1 unit. With Definition 6.2.1, one may only say that moving 0 units is possible with a probability of b while moving 1 unit is possible with a probability of $1 - b$. However, by allowing the *rhs* of Σ_{PR} to be arbitrary fluent formulas, we may additionally express that the probability of moving 1 unit is $1 - b$ provided the robot is not already at its destination; otherwise, it is 0. That is, if the robot is already at the wall, the nondeterministic choice of moving by a unit should not be applicable. We avoid this generality for simplicity. See [Bacchus et al., 1999] for more discussions on independence assumptions such as these.

[7] Given any set of primitive formulas, an action theory determines precisely which of these are true after actions. It then follows that worlds that satisfy a basic action theory are *normal*. In this sense, if a basic action theory is all that the agent knows, then its epistemic state is also normal.

$\Sigma_0 = \{distance = 4 \vee distance = 5\};$

$\Sigma_{pre} = \{\Box Poss(v) = 1 \equiv \text{TRUE}\};$

$\Sigma_{post} = \{\Box[v]distance = x \equiv$

$v = forward \wedge distance = x + 1 \vee$

$v = reverse \wedge distance = x - 1 \vee$

$distance = x \wedge v \neq reverse \wedge v \neq forward\};$

$\Sigma_{\Pi} = \{\Box choice(x, y) = 1 \equiv x = noisyReverse \wedge$

$y = reverse \vee y = noop\};$

$\Sigma_{\text{Pr}} = \{\Box prob(x, y) = u \equiv x = noisyReverse \wedge$

$y = reverse \wedge u = .9 \vee$

$y = noop \wedge u = .1\}.$

Figure 6.2: The simple robot domain reconsidered.

Let us now suppose that the agent quantifies the uncertainty in the initial theory by $\boldsymbol{B}(distance = 4) \geq .4$ and $\boldsymbol{B}(distance = 5) \geq .6$. Then, the background theory T is given as

$$\boldsymbol{O}\Sigma \wedge \boldsymbol{B}distance = 4 \geq .4 \wedge \boldsymbol{B}distance = 5 \geq .6. \quad \blacksquare$$

6.2.2 Formal Foundations

In the sequel we are concerned with the progression of a theory $T = \boldsymbol{O}\Sigma \wedge \bigwedge \boldsymbol{B}\alpha \geq b$. The question we must now answer is this: what is progression in the presence of belief atoms and noisy actions? We now address this question and establish the foundations of progression in the context of uncertainty.

Consider *classical progression*. In Section 5.1.1, we briefly reviewed that Lin and Reiter [1997] provide a model-theoretic definition for the progression of situation calculus basic action theories and discuss several properties that the new definition must satisfy. The main message is that the new and initial theory agree on arbitrary queries about the future. It turns out a very similar account also works for us.

Definition 6.2.3. (Progression.) Suppose T is a theory as above. Let r and s be a primitive action and primitive noisy action, respectively. We call T' the progression of T wrt r (or s) iff for every model M, M is a model of T' iff there is a model M' of T such that M is the progression of M' wrt r (or s). $\blacksquare$

Suppose T' exists. Then it follows that it has the right properties in the sense that T' is fully compatible with T on unrestricted queries about the future.

Theorem 6.2.4. *Let T, r and s be as above. Then,*

1. *Suppose T' is the progression of T wrt r. Then given any formula α, $T \models [r]\alpha$ iff $T' \models \alpha$.*
2. *Suppose T' is the progression of T wrt s. Then given any formula α, $T \models [\![s]\!]\alpha$ iff $T' \models \alpha$.*

Proof: The proofs are similar and so we only show item (1). For item (2), the only change to the proof is that instead of arguing with the progression of models wrt r, we argue with the progression of models wrt s.

Suppose $T \models [r]\alpha$. Let M be a model of T'. By Definition 6.2.3, there is a model M' of T such that $M'_r = M$. But if $M' \models T$ then $M' \models [r]\alpha$ and by the semantics, $M'_r \models \alpha$, *i.e.* $M \models \alpha$. Therefore $T' \models \alpha$.

Conversely, suppose $T' \models \alpha$. Let M be an arbitrary model of T. Now consider the progression of M, *i.e.* M_r. Since T' is the progression of T, $M_r \models T'$, and this means that $M_r \models \alpha$. That is, $M \models [r]\alpha$ by definition, and therefore $T \models [r]\alpha$. ∎

Thus, given that T' has the desired properties, the obvious question is whether it always exists. For the first step, observe that when belief atoms do not appear in T and we are only concerned with progressing wrt ordinary actions, then this is precisely the case studied in the previous chapter, *viz.* Theorem 5.1.4. We now show that regarding this case, the previous result can also be proved for the new logic:

Theorem 6.2.5. *Let $T = \boldsymbol{O}(\phi \wedge \Box\beta)$. Then the progression of T wrt a primitive action r is*

$$\boldsymbol{O}(Prog(\phi) \wedge \Box\beta)$$

where $Prog(\phi) = \exists \vec{P}[\phi^{\vec{F}}_{\vec{P}} \wedge \bigwedge \forall \vec{x}, y.\ f(\vec{x}) = y \equiv \gamma_{f_r}{}^{\nu^{\vec{F}}}_{\vec{P}}]$.

Proof: Let T' denote $\boldsymbol{O}(Prog(\phi) \wedge \Box\beta)$. Let $M = (e, w, \mu, \delta)$ be an arbitrary model of T'. We now construct a model for T, say $M' = (e', w', \mu', \delta)$, such that $M'_r = M$. We proceed as follows:

- Let e' be any set of worlds satisfying $\boldsymbol{O}(\phi \wedge \Box\beta)$. By means of Lemma 5.1.5 and Lemma 5.1.6, it follows that $w^* \in e'_r$ iff $w^* \models Prog(\phi) \wedge \Box\beta$, *i.e.* e'_r satisfies $\boldsymbol{O}(Prog(\phi) \wedge \Box\beta)$. Moreover, $e'_r = e$.[8]
- Construct a world w' with the following properties:
  - for all primitive terms d, $w'[d, \langle\rangle]$ is an arbitrary name;
  - for all primitive terms d, $w'[d, r \cdot z] = w[d, z]$ for all $z \in \mathcal{Z}$.
- Note that since T does not mention any belief atoms, μ' can be any arbitrary measure provided it is a well defined in the sense of satisfying **M1** − **M3**. We construct such a measure now.

  Let us consider the equivalence classes of e and e'. That is, suppose $e = W_1 \cup \ldots \cup W_k$, where W_i is $\|w^*\|$ for some $w^* \in e$. Analogously, suppose $e' = W'_1 \cup \ldots \cup W'_{k'}$.

  By Lemma 6.1.10, it follows that for any W'_i and W_j

[8] Recall from our discussions in Section 3.1.2, which although was in the context of $\mathcal{OL}$, that any epistemic state satisfying $\boldsymbol{O}\phi$, if ϕ is an objective sentence, is unique and maximal. (See Theorem 8.3.1 in [Levesque and Lakemeyer, 2001].) The argument is the same for $\mathcal{ES}$ and its variants since we are simply constructing $\{w \mid w \models \phi\}$ which results in a unique set of worlds provided ϕ is a fluent sentence.

1. $(W'_i)_r = W_j$, or
2. $(W'_i)_r \subset W_j$, and so suppose $(W'_i \cup W'_{h1} \cup \ldots \cup W'_{hl})_r = W_j$, for some $h1, \ldots, hl$.

That is, either the progression of an equivalence class in e' results in an equivalence class in e, or the equivalence classes may merge. If (1) and $\mu(W_j) = b$ then let μ' be a measure such that $\mu'(W'_h) = b$. If the latter and $\mu(W_j) = b$ then let $\mu'(W'_h) = b/(l+1)$ for $h \in \{i, h1, \ldots, hl\}$. It then follows that if (2) then $\mu'(\bigcup W'_h) = b$.

In sum, we have assigned measures to all the equivalence classes in e'.

It is now straightforward to verify that $M' = (e', w', \mu', \delta)$ is the desired model in the sense that $M'_r = M$. ∎

Thus, we are able to precisely define the progression of $\boldsymbol{O}(\phi \wedge \Box\beta)$ wrt ordinary actions.[9] Unfortunately, we do not have a proof whether T' exists in general. By this we mean that, when T mentions belief atoms, or when we are interested in the progression wrt noisy actions, it is not clear how T' is to be formulated for an arbitrary T.

For the rest of the chapter, we are interested in a practical case, which is, in fact, motivated by Example 6.2.2. For this case, under certain assumptions, we show that one is able to obtain the progression of a theory T wrt both ordinary and noisy actions.

6.2.3 Progression for a Practical Case wrt Ordinary Actions

The practical case for which the progression of a background theory T is definable is motivated by Example 6.2.2, where we encounter reasoning problems of the following sort:

> *Suppose the robot believes that it is 5 units away with a .6 probability. Then after moving forward, it now believes that it is 4 units away with a .6 probability.*

We observe that for realistic applications such as these, it often suffices to maintain beliefs about literals. Moreover, the agent must be able to update the values of such literals after doing the action. But clearly if the context formula of the corresponding successor state axiom mentions fluents about which the agent does not have complete information, this is no longer possible. To give a simple example, suppose we have the following successor state axiom in the basic action theory:

$$\Box[v]f = y \equiv v = A \wedge g = 0 \wedge y = 1 \vee v \neq A.$$

Then after executing the action A, the fluent f obtains a value of 1 provided the value of g is 0. But if the agent does not know the value of g, then it will not be able to update f to 1 after executing A.

To this end, the only assumption we will need to make is for the initial KB to have complete knowledge about the fluents in the successor state axioms. However, in many cases where we need to deal with beliefs this is too strong an assumption. For instance, in Example 6.2.2, we note that the new value of the fluent *distance* depends on the previous one, about which the agent has incomplete knowledge. Therefore, in order to capture the kind of applications we have in mind, we introduce *essentially local-effect action theories*:[10]

[9] Note that, as investigated in the previous chapter, while the progressed basic action theory requires second-order logic in general, there are cases where it is first-order definable.

[10] In an earlier version of our results [Belle and Lakemeyer, 2011c], we refer to essentially local-effect action theories as *normal successor state axioms*. We avoid confusion with the notions from Section 5.2.3 by renaming our concept.

$\Box[a]status(x) = y \equiv$

$a = drop(x) \wedge fragile(x) = 1 \wedge y = destroyed \vee$

$a \neq drop \wedge status(x) = y.$

Figure 6.3: Dropping of an object.

Definition 6.2.6. (Essentially local-effect action theory.) Let the successor state axiom for fluent f be of the form:

$$\Box[v]f(\vec{x}) = y \equiv \gamma_f(\vec{x}, y, v) \vee f(\vec{x}) = y \wedge \neg\exists h\gamma_f(\vec{x}, h, v).$$

The successor state axiom is *essentially local-effect* if $\gamma_f(\vec{x}, y, v)$ is a disjunction of formulas of the form:

$$\exists\vec{u}, h.\ [v = A(\vec{z}) \wedge \zeta(\vec{z}) \wedge f(\vec{x}) = h \wedge y = \Theta(\vec{z}, h)].$$

where $\vec{z}$ mentions $\vec{x}$ and $\vec{u}$ are the remaining variables in $\vec{z}$, $\zeta(\vec{z})$ is any fluent formula not mentioning actions, and $\Theta(\vec{z}, h)$ is any arithmetical expression involving the previous value of $f(\vec{x})$ and $\vec{z}$. (On substituting $\vec{z}$ and h with names, say $\vec{m}$ and n, $\Theta(\vec{m}, n)$ resolves to another standard name.)

We call $\zeta(\vec{z})$ the *context formula.* ∎

Example 6.2.7. Consider $\gamma_{distance}(x, v)$ from Example 6.2.2, which is essentially local-effect. This can be rewritten as:

$v = forward \wedge \exists h[distance = h \wedge x = h - 1] \vee$

$v = reverse \wedge \exists h[distance = h \wedge x = h + 1].$

There is no context formula in $\gamma_{distance}$. ∎

Example 6.2.8. Consider γ_{status} given in Figure 6.3, which is essentially local-effect. The successor state axiom says that dropping a fragile object destroys it. In γ_{status}, the context formula is
$fragile(x) = 1$. ∎

Essentially local-effect action theories are similar to local-effect theories except that the argument of the action, *i.e.* the vector of variables $\vec{z}$, need not mention the value of the fluent, *i.e.* the variable y. In this sense, essentially local-effect successor state axioms are more general than local-effect successor state axioms encountered in Definition 5.2.10. The idea now is that it is sufficient for the initial KB to only have complete knowledge about the fluents appearing in the context formula. All this is made precise below.

Definition 6.2.9. (Completeness property.) A fluent sentence ϕ is *complete* wrt a set of primitive formulas Γ if for all $p \in \Gamma$, either $\phi \models p$ or $\phi \models \neg p$.

We say ϕ is complete wrt a fluent f if it is complete wrt all instances of f which are primitive formulas. ∎

That is, given an arbitrary fluent formula α, which only mentions fluents wrt which ϕ is complete, either $\phi \supset \alpha$ or $\phi \supset \neg\alpha$ is valid.

Example 6.2.10. Let ϕ be a conjunction of the following sentences:

- $\forall(\mathit{fragile}(x) = 1 \equiv x = D)$,
- $\mathit{status}(D) \neq \mathit{destroyed} \vee \mathit{status}(D) \neq \mathit{cracked}$
- $\mathit{status}(C) = \mathit{open}$

It is complete wrt *fragile*, but not wrt *status*. However, ϕ is complete wrt the primitive formula $\mathit{status}(C) = \mathit{open}$. ∎

For the next step, we isolate the fluents wrt which such a property is necessary. Given a theory $T = \boldsymbol{O}(\Sigma) \wedge \bigwedge \boldsymbol{B}p \geq b$, where p is either a positive or negative primitive literal, we let $\mathcal{F}_{\boldsymbol{B}} \subseteq \mathcal{F}$ denote all the fluents mentioned in the belief atoms.

Definition 6.2.11. (Context-completeness.) Suppose $T = \boldsymbol{O}(\phi \wedge \Box\beta) \wedge \bigwedge \boldsymbol{B}p \geq b$ is a theory, where $\phi \wedge \Box\beta$ is a basic action theory. We say that T is *context-complete* (for its beliefs) iff:

- the successor state axiom for every $f \in \mathcal{F}_{\boldsymbol{B}}$ is essentially local-effect;
- ϕ is complete wrt all fluents appearing in the context formulas of these successor state axioms. ∎

Example 6.2.12. Consider the basic action theory from Example 6.2.2. The background theory is context-complete, because it only mentions belief atoms about the fluent *distance* which is essentially local-effect without context formulas. ∎

Example 6.2.13. Consider the successor state axiom from Figure 6.3, which we denote as $\mathit{SSA}_{\mathit{status}}$. Let $\Sigma_0 = \{\phi\}$ from Example 6.2.10. Let T be

$$\boldsymbol{O}(\Sigma_0 \wedge \mathit{SSA}_{\mathit{status}}) \wedge \boldsymbol{B}\mathit{status}(D) \neq \mathit{destroyed} \geq .9$$

Then T is context-complete, because it maintains a belief about an instance of *status*, whose successor state axiom is essentially local-effect, and T is complete wrt all the fluents appearing in the context formula, *viz.* $\mathit{fragile}(x) = 1$, of this successor state axiom. ∎

With this in hand, we prove a preliminary result before presenting results on updating belief atoms:

Proposition 6.2.14. *Suppose Σ is a basic action theory, and the successor state axiom for the fluent f is essentially local-effect. Let r denote the primitive action $A(\vec{o})$. Then there is a formula $\psi(\vec{x}, y)$ of the following form:*

$$\bigvee_i (\vec{x} = \vec{m}_i \wedge \zeta_i(\vec{o}) \wedge \exists h.\, [f(\vec{m}_i) = h \wedge y = \Theta_i(h, \vec{o})])$$

where $\vec{m}_i$ are name vectors appearing in $\vec{o}$, and $\zeta_i(\vec{o})$ do not mention any free variables such that the following holds:

$$\models \forall(\gamma_f(\vec{x}, y, r) \equiv \psi(\vec{x}, y)).$$

Proof: The formal arguments are similar to the simplifications we pursued with local-effect action theories, *i.e.* Proposition 5.2.12. More precisely, since the successor state axiom for f is essentially local-effect, by Definition 6.2.6, γ_f is a disjunction of formulas of the form $\exists \vec{u}, h.[v = A(\vec{z}) \wedge \zeta(\vec{z}) \wedge f(\vec{x}) = h \wedge y = \Theta(h, \vec{z})]$. By the uniqueness of actions, $\gamma_f(\vec{x}, y, A(\vec{o}))$ simplifies to $\exists h.\ [\vec{x} = \vec{m} \wedge \zeta(\vec{o}) \wedge f(\vec{x}) = h \wedge y = \Theta(h, \vec{o})]$, where $\vec{x}$ are variables in $\vec{z}$. ∎

In what follows, we assume without any loss of generality that $\gamma_f(\vec{x}, y, r)$ is simplified to the form indicated by Proposition 6.2.14.

Proposition 6.2.15. *Suppose $T = \boldsymbol{O}(\phi \wedge \Box\beta) \wedge \boldsymbol{B} f(\vec{m}) \circ n \geq b$ is context-complete, where $f(\vec{m}) \circ n$ is a literal and $\circ \in \{=, \neq\}$. Suppose M is a model of T and r is as above. Then $M_r \models \boldsymbol{B}(f(\vec{m}) \circ n^*) \geq b$, where n^* is as follows:*

- *if $\vec{x} = \vec{m}$ does not appear in $\psi(\vec{x}, y)$ (as obtained from Proposition 6.2.14), then n^* is n;*
- *if $\vec{x} = \vec{m} \wedge \zeta(\vec{o}) \wedge \exists h.\ [f(\vec{m}) = h \wedge y = \Theta(h, \vec{o})]$ appears in $\psi(\vec{x}, y)$ and*
  - *if $\phi \models \zeta(\vec{o})$ then n^* is $\Theta(n, \vec{o})$;*[11]
  - *otherwise, n^* is n.*

Proof: For the first step, observe that for any world w that satisfies $\phi \wedge \Box\beta \wedge f(\vec{m}) \circ n$, it follows that $w \models [r] f(\vec{m}) \circ n^*$ as stated by the conditions above. This can be argued for as follows. Since f is essentially local-effect by assumption, $w \models [v] f(\vec{x}) = y \equiv \gamma_f(\vec{x}, y, v) \vee f(\vec{x}) = y \wedge \neg\exists h \gamma_f(\vec{x}, h, v)$ where γ_f is essentially local-effect. Now, on substituting v with r and simplifying $\gamma_f(\vec{x}, y, r)$ as in Proposition 6.2.14, we obtain the formula $\psi(\vec{x}, y)$. Clearly if $\vec{x} = \vec{m}$ does not appear in $\psi(\vec{x}, y)$ then $w \models ([r] f(\vec{m}) = y) \equiv f(\vec{m}) = y \wedge \neg\exists h \bigvee_i (\vec{m} = \vec{m}_i \wedge \zeta_i(\vec{o}) \wedge \exists h'[f(\vec{m}_i) = h' \wedge h = \Theta_i(h', \vec{o})])$, *i.e.* $w \models ([r] f(\vec{m}) = y) \equiv f(\vec{m}) = y$ because $\vec{m} = \vec{m}_i$ is equivalent to FALSE for every i. Since $w \models f(\vec{m}) \circ n$, it follows then that $w \models [r] f(\vec{m}) \circ n$. On the other hand if $\vec{x} = \vec{m}$ does appear, and

- if $\phi \models \zeta(\vec{o})$ then $w \models [r] f(\vec{m}) \circ \Theta(n, \vec{o})$;
- else, if $\phi \models \neg\zeta(\vec{o})$ then $w \models [r] f(\vec{m}) = y \equiv f(\vec{m}) = y$.

Note that due to the context-completeness assumption, either $\phi \models \zeta(\vec{o})$ or $\phi \models \neg\zeta(\vec{o})$. Thus, it follows that $w, r \models f(\vec{m}) \circ n^*$. From Lemma 5.2.19, we have $w_r \models f(\vec{m}) \circ n^*$.

Now, Let M denote the tuple (e, w, μ, δ). By assumption, $\mu(\|[f(\vec{m}) = n]_e\|) \geq b$. Let W denote the set of worlds $\|[f(\vec{m}) = n]_e\|$.

Next, pick an arbitrary $w^* \in W$. By construction, $w^* \models f(\vec{m}) \circ n$. By the argument above, $w^*_r \models f(\vec{m}) \circ n^*$. Since w^* is an arbitrary world from W, it follows that the progression of all the worlds in W satisfies $f(\vec{m}) \circ n^*$.

Now $(W)_r \subseteq [f(m) = n^*]_{e_r}$. So, by way of Corollary 6.1.9, it follows that if $\mu(\|W\|) \geq b$ then $\mu_r(\|(W)_r\|) \geq b$ as well. Therefore $\mu(\|[f(m) = n^*]_{e_r}\|) \geq b$. ∎

[11] Recall that Θ is an arithmetical expression and if its arguments are names then it resolves to a name.

This proposition says that if a literal p, say $f(\vec{m}) = n$, is believed with a probability $\geq b$ in T, then after doing an action r, the updated literal $f(\vec{m}) = n^*$ is believed with a probability $\geq b$. Note that the new value for $f(\vec{m})$ depends crucially on the action r, since this determines which fluent terms of f are affected. This may vary for each primitive action, as in the local-effects case.

Henceforth, for the purpose of readability if we denote a literal $f(\vec{m}) \circ n$ by p, then we denote the updated literal wrt r, *i.e.* $f(\vec{m}) \circ n^*$, by p^*. The above result can be extended to an arbitrary set of beliefs atoms in the following manner:

Proposition 6.2.16. *Suppose T, r and M are as above. Then:*

1. *If $M \models \boldsymbol{B}p \geq b_1 \wedge \boldsymbol{B}p \geq b_2$ then*

$$M_r \models \boldsymbol{B}p^* \geq max(b_1, b_2).$$

2. *If $M \models \bigwedge \boldsymbol{B}p_i \geq b_i$, where p_i's are different, then*

$$M_r \models \bigwedge \boldsymbol{B}p_i^* \geq \sum_{\{j | p_j^* \text{ is the same as } p_i^*\}} b_j.$$

Proof: Item (1) follows as a straightforward corollary of Proposition 6.2.15. This is because by the definition of the semantics, $M \models \boldsymbol{B}p \geq max(b_1, b_2)$ and then we can apply Proposition 6.2.15.

For item (2), we have two cases. The easy one is if p_i^* and p_j^* are different, for every i, j and $i \neq j$. Then the proposition is asking us to show that $M_r \models \boldsymbol{B}p_i^* \geq b_i$ for each i, which is precisely what Proposition 6.2.15 establishes.

Now, suppose that p_i^* and p_j^* for $i \neq j$ are the same, then both p_i and p_j must be literals mentioning the same primitive term. So suppose that $\{p_1, \ldots, p_k\}$ are literals appearing in belief atoms in T mentioning the same primitive term. For ease of exposition, let $k = 2$. (The argument for $k > 2$ is straightforward but tedious.) Now, let W_1 be the set of worlds $[p_1]_e$ and let W_2 be the set of worlds $[p_2]_e$. It is easy to see that W_1 and W_2 are disjoint.

Now, suppose that $\mu(\|W_1\|) \geq b_1$ and $\mu(\|W_2\|) \geq b_2$. Since W_1 and W_2 are disjoint, it follows that $\mu(\|W_1\| \cup \|W_2\|) \geq b_1 + b_2$, or $\mu(\|W_1 \cup W_2\|) \geq b_1 + b_2$. Since p_1^* and p_2^* are the same, clearly $[p_1^*]_{e_r} \supseteq (W_1 \cup W_2)_r$. Therefore, $\mu_r(\|[p_1^*]_{e_r}\|)$ is greater than or equal to $\mu_r(\|(W_1 \cup W_2)_r\|)$, which by Corollary 6.1.9, is greater than or equal to $\mu(\|W_1 \cup W_2\|) \geq b_1 + b_2$. ∎

Note that in item (2) of the above proposition, we assumed without loss of generality that the p_i's are different because if we have multiple beliefs about p_i then we can simplify it as indicated by item (1) of the proposition, *i.e.* by choosing the maximum of the probabilities.

Example 6.2.17. We consider some variants of Example 6.2.2. Let Σ be the basic action theory from that example. Let r denote the action *forward*. Then:

- Suppose $T = \boldsymbol{O}\Sigma \wedge \boldsymbol{B}(distance = 4) \geq .4$ and let M be a model of T. Then $M_r \models \boldsymbol{B}(distance = 3) \geq .4$.
- Suppose $T = \boldsymbol{O}\Sigma \wedge \boldsymbol{B}(distance = 4) \geq .3 \wedge \boldsymbol{B}(distance = 4) \geq .4$. If M is a model of T, then $M_r \models \boldsymbol{B}(distance = 3) \geq .4$.

- Suppose $T = \boldsymbol{O}\Sigma \wedge \boldsymbol{B}(distance = 4) \geq .4 \wedge \boldsymbol{B}(distance = 5) \geq .6$. If M is a model of T, then $M_r \models \boldsymbol{B}(distance = 3) \geq .4 \wedge \boldsymbol{B}(distance = 4) \geq .6$.

In each of these cases, we have obtained the new value for the fluent *distance* by means of Proposition 6.2.15. ∎

Example 6.2.18. To inspect a case where two updated literals may end up being the same, as entertained by Proposition 6.2.16, consider the following successor state axiom, which simply sets a 0-ary fluent to 1 after doing an action A.

$$\Box[v]f = y \equiv$$

$$v = A \wedge y = 1 \vee$$

$$v \neq A \wedge f = y.$$

Now, suppose M is a model of

$$\boldsymbol{O}(f = 1 \vee f = 0 \wedge \text{SSA}_f) \wedge \boldsymbol{B}(f = 1) \geq .4 \wedge \boldsymbol{B}(f = 0) \geq .6$$

then $M_A \models \boldsymbol{B}(f = 1) = 1$. That is, the next value of f is 1 irrespective of what it is previously after doing A. Thus, the previous beliefs are summed. ∎

What comes out of Proposition 6.2.15 and Proposition 6.2.16 is that the belief atoms in the progressed model are definable via simple steps. That is, given the set of belief atoms appearing in T, we are able to write down the belief atoms appearing in T' which is the progression of T wrt a primitive action.

We now turn to the main result. Below, we prove a theorem that determines what the progression of a theory looks like after executing an action. For ease of exposition, we consider a theory T that only mentions a single belief atom. When T mentions a conjunction of belief atoms, the theorem is generalized by means of Proposition 6.2.16 which indicates how beliefs atoms are updated after progression.

Theorem 6.2.19. *Suppose $T = \boldsymbol{O}(\phi \wedge \Box\beta) \wedge \boldsymbol{B}p \geq b$ is context-complete and r is as above. Then the progression of T wrt r is*

$$\boldsymbol{O}(Prog(\phi) \wedge \Box\beta) \wedge \boldsymbol{B}p^* \geq b$$

where $Prog(\phi)$ is as above.

Proof: Let us denote the progressed theory by T'. Suppose $M = (e, w, \mu, \delta)$ is any model of T', we now shown that we can construct one for T, say $M' = (e', w', \mu', \delta)$, such that $M'_r = M$.

Let e' be an epistemic state that satisfies $\boldsymbol{O}(\phi \wedge \Box\beta)$. See Theorem 6.2.5 to verify that $e'_r = e$. Further, that theorem also instructs us how a world w' is to be constructed such that $w'_r = w$.

To construct a measure μ' note that the only constraint imposed by the belief atoms from T is that $\boldsymbol{B}p \geq b$. Given that μ is a measure that satisfies $\boldsymbol{B}p^* \geq b$, suppose that $\mu(\|[p^*]_e\|) = b'$, where $b' \geq b$. If $e - [p^*]_e \neq \emptyset$, then $b' \neq 1$ (by **M3**) and $\mu(\|e - [p^*]_e\|) = 1 - b'$.

Let $W = \{w' \in e' \mid w' \models p, w'_r \models p^*\}$. Clearly $(W)_r \subseteq [p^*]_e$. There are two cases,

1. If $(W)_r = [p^*]_e$, then precisely the progression of those worlds in $W \subseteq e'$ satisfy p^* in e. Now let $\mu'(\|W\|) = b'$. Since $b' \geq b$, it follows that μ' satisfies the constraint imposed by the belief atom in T. If $\|W\|$ does not correspond to a single equivalence class, but say k of them, then let μ' assign a probability of b'/k to each of them.

   Now, consider $W' = e' - W$. It is easy to see that if $e - [p^*]_e \neq \emptyset$ then $W' \neq \emptyset$. If $\|W'\|$ contains (say) k' equivalence classes, then let μ' assign $(1 - b')/k'$ to each of these equivalence classes.

2. If $(W)_r \subset [p^*]_e$ then there are worlds outside of $W \subseteq e'$ which also satisfy p^* on progression. That is, let $W' = \{w' \in e' \mid w' \not\models p, w'_r \models p^*\}$ and by assumption $W' \neq \emptyset$.

   Now, we construct μ' based on the following conditions:

   - Suppose $b' - b \neq 0$. Then let $\epsilon = b' - b$. Let $\mu'(\|W\|) = b + \epsilon/2$, let $\mu'(\|e' - W\|) = 1 - b - \epsilon/2$. If $\|W\|$ and $\|e' - W\|$ correspond to many equivalence classes, then we do as above.
   - Suppose $b' = b$. Suppose p denotes $f(\vec{m}) = n$. Now, note that since $W' \neq \emptyset$, there are worlds in e' that satisfy a different value to $f(\vec{m})$, say n'. Then we claim that for T to be satisfiable, $b \neq 1$. Suppose otherwise. Then for any model of T the probability on the set of worlds in the epistemic state which satisfy $f(\vec{m}) = n$ is 1 which means that the probability assigned to the set of worlds that satisfy $f(\vec{m}) = n'$, which is non-empty by assumption, is 0. Thus a model cannot be obtained, since every probability space will fail to satisfy **M3**.

     So let $\mu'(\|W\|) = b$ and let $\mu'(\|e' - W\|) = 1 - b$. If $\|W\|$ and $\|e' - W\|$ correspond to many equivalence classes, then we do as above.

   The constraint imposed by the belief atom in T is satisfied.

It is now easy to verify that M' is a model of T and $M'_r = M$. ∎

Example 6.2.20. Consider the progression of the background theory T from Example 6.2.2 wrt *forward*. We proceed as follows:

- $Prog(\phi)$. This is obtained as $\exists P[(P = 4 \vee P = 5) \wedge \forall x.distance = x \equiv forward = forward \wedge x = P - 1]$. On simplification $Prog(\phi) = \{distance = 3 \vee distance = 4\}$.

- Since there are two belief atoms in T, where the literal mentioned in the belief atoms are different, we apply (2) from Proposition 6.2.16. Recall from Example 6.2.17 that the belief atom $\boldsymbol{B}(distance = 4) \geq .4$ is updated to $\boldsymbol{B}(distance = 3) \geq .4$. Similarly, the belief atom $\boldsymbol{B}(distance = 5) \geq .6$ is updated to $\boldsymbol{B}(distance = 4) \geq .6$.

  Putting this together, the progressed theory T' is:

  $$\boldsymbol{O}(distance = 3 \vee distance = 4 \wedge \Box\beta) \wedge$$

  $$\boldsymbol{B}(distance = 3) \geq .4 \wedge \boldsymbol{B}(distance = 4) \geq .6.$$ ∎

6.2.4 Progression for a Practical Case wrt Noisy Actions

Computing the progression of a theory wrt a noisy action is considerably more complex. To see a simple example as to why this might be the case, reconsider Example 6.2.2. But now suppose that the agent has complete knowledge regarding its distance to the wall. If we let Σ denote the basic action theory from that example, then the following sentence is shown to be valid:

$$\boldsymbol{O}(\{distance = 5\} \cup \Sigma) \supset [\![noisyReverse]\!]\boldsymbol{B}(distance = 6) = .9$$

That is, even if the agent has complete knowledge initially, the noisy actuator on execution generates probabilistic beliefs. Naturally, our task for obtaining a definition of the progression of a theory has to somehow take this into account. Perhaps the simplest way is to require that the theory already maintains beliefs about primitive terms that a stochastic action can affect and then monitor these beliefs after a noisy action is executed. It turns out this idea can be formally accounted for, and this is what we will pursue for the rest of the section.

Since noisy actions generalize ordinary actions, not surprisingly we inherit the restriction that all the fluents appearing in the belief atoms must be essentially local-effect. The question then is whether we can capture every literal that noisy actions affect. We proceed as follows to confirm this. Let s be a noisy action, and let $\Pi(s) = \{A_1(\vec{o}_1), A_2(\vec{o}_2), \ldots, A_k(\vec{o}_k)\}$. Then,

- Let H be the set of all names appearing in $\cup_i \vec{o}_i$.
- Let $\mathcal{F}' \subseteq \mathcal{F}$ denote the set of fluents f from $\mathcal{F}$ such that $v = A_i(\vec{z})$ appears in $\gamma_f(\vec{x}, y, v)$. Intuitively, we are gathering every fluent that an instance of A_i affects.
- Now, let $\Delta = \{f(\vec{m}) \mid f \in \mathcal{F}', m_i \in H\}$. Intuitively, this is the set of all fluent terms that may be affected after doing s.

With this in hand, we now make precise the assumptions about a theory T.

Definition 6.2.21. (Determinate property.) Suppose $T = \boldsymbol{O}(\phi \wedge \Box\beta) \wedge \bigwedge \boldsymbol{B}p_i \geq b_i$. Let s be a primitive noisy action and let Δ be a set of primitive terms obtained as above. We say that T is *determinate* wrt s if

- for every $d \in \Delta$:
  1. there are a finite set of names n_j such that $\phi \models \bigvee_j d = n_j$;
  2. T has belief atoms for each $d = n_j$.
- T is context-complete (for its beliefs). ∎

Intuitively, this says that ϕ entails a finite number of *possible values* for each $d \in \Delta$, and these possibilities have probabilities assigned to them. As mentioned earlier, we inherit the assumptions needed in the ordinary actions case since noisy actions generalize the former. That is, for every fluent f whose instance is a belief atom in T, f is essentially local-effect and ϕ is complete wrt the context formulas mentioned in γ_f.

Example 6.2.22. When resorting to Example 6.2.2, we see that the only noisy action is *noisyReverse*, whose choices are either *reverse* or *noop*. These actions affect the fluent *distance*, and in fact, the initial theory contains a possible value clause for this fluent, *viz.* $distance = 4 \vee distance = 5$, together with beliefs about each disjunct. Moreover, since the fluent is essentially local-effect without context formulas, the theory is determinate by our definition. ∎

In general, we do not believe the new assumptions lead to serious problems because in most realistic domains, we imagine possible values to range over a small number of names. Moreover, if there is no reason to consider one value any more likely than the other, one typically assumes that every possibility is equally likely. For example, when tossing a coin, unless it is known that it is biased, a heads or a tails can be expected with the same probability.

We are now ready to define the progression of a theory wrt a noisy action s, provided that the theory is determinate wrt s. However, to get an impression of what this definition should look like, we prove some lemmas.

In what follows, we will need to distinguish between the effects of the individual action choices for a given noisy action s. To that end, given a theory $T = \boldsymbol{O}(\phi \wedge \Box\beta) \wedge \bigwedge \boldsymbol{B}p_i \geq b_i$ that is determinate wrt s, and $r_i \in \Pi(s)$:

- Let us denote by $Prog(\phi, r_i)$ the progression of the fluent sentence ϕ wrt the ordinary action r_i. More precisely, let $Prog(\phi, r_i)$ denote

$$\exists \vec{P}[\phi^{\vec{F}}_{\vec{P}} \wedge \bigwedge \forall \vec{x}, y.\ f(\vec{x}) = y \equiv \gamma_{f^{v}_{r_i}}{}^{\vec{F}}_{\vec{P}}].$$

- Suppose $\boldsymbol{B}p \geq b$ is a belief atom appearing in T, where p denotes $f(\vec{m}) \circ n$ with $\circ \in \{=, \neq\}$. Let $\psi(\vec{x}, y)$ be the formula obtained on simplifying $\gamma_f(\vec{x}, y, r_i)$ in a manner as indicated by Proposition 6.2.14. Then let $p^*_{r_i}$ denote the following primitive formula:

$$f(\vec{m}) \circ \begin{cases} n & \text{if } \vec{x} = \vec{m} \text{ does not appear in } \psi(\vec{x}, y) \\ n & \text{if } \vec{x} = \vec{m} \wedge \zeta(\vec{o}) \wedge \exists h[f(\vec{m}) = h \wedge y = \Theta(h, \vec{o})] \text{ appears in } \psi(\vec{x}, y) \text{ and } \phi \not\models \zeta(\vec{o}) \\ \Theta(n, \vec{o}) & \text{if } \vec{x} = \vec{m} \wedge \zeta(\vec{o}) \wedge \exists h[f(\vec{m}) = h \wedge y = \Theta(h, \vec{o})] \text{ appears in } \psi(\vec{x}, y) \text{ and } \phi \models \zeta(\vec{o}). \end{cases}$$

We begin by showing how belief atoms are updated after noisy actions occur. We first consider a theory mentioning a single belief atom, and later, in Proposition 6.2.25, we generalize the result to the case where a theory mentions a conjunction of belief atoms.

Lemma 6.2.23. *Suppose s is a primitive noisy action and $\Pi(s) = \{r_1, \ldots, r_k\}$. Suppose $T = \boldsymbol{O}(\phi \wedge \Box\beta) \wedge \boldsymbol{B}p \geq b$ is determinate wrt s. If M is a model of T, then M_s satisfies:*

$$\bigwedge_i \boldsymbol{B}p^*_{r_i} \geq b \times \sum_{\{j | p^*_{r_j} \text{ is the same as } p^*_{r_i}\}} \Pr(r_j).$$

Proof: For ease of exposition, let $\Pi(s) = \{r_1, r_2\}$. For ease of readability, let p_i denote $p^*_{r_i}$. There are two cases, either p_1 and p_2 are the same, or they are different.

- Suppose they are different. The lemma is asking us to show that $M_s \models \boldsymbol{B}p_1 \geq b \times \Pr(r_1) \wedge \boldsymbol{B}p_2 \geq b \times \Pr(r_2)$.

  Let $W = [p]_e$. For ease of exposition, suppose that W is a single equivalence class, say $\|w'\|$. By using the formal arguments from Lemma 6.2.15, it is not hard to see that $w'_{r_1} \models p_1$. Analogously, $w'_{r_2} \models p_2$. Thus, $w'_{r_1} \not\approx w'_{r_2}$. In other words, $\|w'_{r_1}\|$ and $\|w'_{r_2}\|$ are disjoint.

  Of course, $(W)_{r_1} \subseteq [p_1]_{e_s}$. By assumption, $\mu(\|W\|) \geq b$. By definition $\mu_s(\|[p_1]_{e_s}\|)$ is at least $\mu_s(\|(W)_{r_1}\|)$, which by Lemma 6.1.11, is greater than or equal to $\Pr(r_1) \times b$. Analogously, $\mu_s(\|[p_2]_{e_s}\|) \geq \Pr(r_2) \times b$. Thus, the lemma is proved for this case.

- Instead, suppose p_1 and p_2 are the same. The lemma is asking us to show that $M_s \models \boldsymbol{B}p_1 \geq b$. As in the previous case, for ease of exposition, suppose that $W = [p]_e$ is a single equivalence class, say $\|w'\|$. Since $\mu(\|w'\|) \geq b$ by assumption, Lemma 6.1.11 establishes that $\mu_s(\|w'_{r_1}\|) \geq b \times \Pr(r_1)$ and $\mu_s(\|w'_{r_2}\|) \geq b \times \Pr(r_2)$.

  Now, there are two cases, either $\|w'_{r_1}\|$ and $\|w'_{r_2}\|$ are disjoint, or they are the same.

  1. Suppose they are disjoint. Since $[p_1]_{e_s} \supseteq \|w'_{r_1}\| \cup \|w'_{r_2}\|$, $\mu_s(\|[p_1]_{e_s}\|)$ is at least $\mu_s(\|w'_{r_2}\| \cup \|w'_{r_1}\|)$ $\geq b \times \Pr(r_1) + b \times \Pr(r_2) \geq b$. This is because $\Pr(r_1) + \Pr(r_2) = 1$.
  2. Suppose they are the same. We have that $[p_1]_{e_s} \supseteq \|w'_{r_1}\|$. Now, by the construction of μ_s, we have $\mu_s(\|w'_{r_1}\|) = \mu(\|w'\|) \times \sum_i \Pr(r_i) = \mu(\|w'\|)$ which is $\geq b$.

  Thus, the lemma is proved in this case as well. ∎

The proposition says that after doing a noisy action, the agent has to consider that its beliefs are updated wrt each choice for s. If any of the updated primitive formulas are the same, then their beliefs can be summed. We obtain the following simple corollary thereof:

Corollary 6.2.24. *Suppose s is as above. Suppose $T = \boldsymbol{O}(\phi \wedge \Box\beta) \wedge \boldsymbol{B}p \geq b \wedge \boldsymbol{B}p \geq b'$ is determinate wrt s. If M is a model of T, then M_s satisfies:*

$$\bigwedge_i \boldsymbol{B}p^*_{r_i} \geq max(b, b') \times \sum_{\{j | p^*_{r_j} \text{ is the same as } p^*_{r_i}\}} \Pr(r_j).$$

Proof: By the definition of the semantics, if $M \models \boldsymbol{B}p \geq b \wedge \boldsymbol{B}p \geq b'$ then $M \models \boldsymbol{B}p \geq max(b, b')$. Then we apply Lemma 6.2.23. ∎

Here is how these results are extended to an arbitrary number of belief atoms mentioning different primitive terms.

Proposition 6.2.25. *Suppose s is as above and suppose $T = \boldsymbol{O}(\phi \wedge \Box\beta) \wedge \bigwedge_i \boldsymbol{B}p_i \geq b_i$, where the p_i's are different, is determinate wrt s. If M is a of model of T, then M_s satisfies the following sentence:*

$$\bigwedge_i \bigwedge_j (\boldsymbol{B}(p_i)^*_{r_j} \geq \sum_u b_u \times (\sum_{\{h | (p_u)^*_{r_h} \text{ is the same as } (p_i)^*_{r_j}\}} \Pr(h))).$$

Proof: For ease of exposition, suppose that $\Pi(s) = \{r_1, r_2\}$ and that T only mentions the belief atoms $\boldsymbol{B}p_1 \geq b_1$ and $\boldsymbol{B}p_2 \geq b_2$. (The case where $\Pi(s) = \{r_1, \ldots, r_k\}$ and T mentions k' atoms is straightforward but tedious.) There are two main cases:

- Suppose p_1 and p_2 do not mention the same fluent term. Then for arbitrary h and u, $(p_1)^*_{r_h}$ will not be the same as $(p_2)^*_{r_u}$. Thus, we are asked to show that

$$M_s \models \bigwedge_j \boldsymbol{B}(p_i)^*_{r_j} \geq b_i \times \sum_{\{h | (p_i)^*_{r_h} \text{ is the same as } (p_i)^*_{r_j}\}} \text{Pr}(r_h)$$

  for each i. This is precisely what Lemma 6.2.23 demonstrates, and so we are done.

- Suppose p_1 and p_2 mention the same fluent term. Then depending on which of the literals from $\{(p_1)^*_{r_1}, (p_2)^*_{r_1}, (p_1)^*_{r_2}, (p_2)^*_{r_2}\}$ are the same, we need slightly different arguments. The arguments are straightforward to adapt, so we show two cases.

  1. Suppose they are all different. Then we need to show that $M_s \models \boldsymbol{B}(p_i)^*_{r_j} \geq b_i \times \text{Pr}(r_j)$ for every i, j.

     Let $M = (e, w, \mu, \delta)$ and let $W = [p_i]_e$ for any i. From Lemma 6.2.15, it is easy to see that for every $w' \in W, w'_{r_j} \models (p_i)^*_{r_j}$ for any j. Since $[(p_i)^*_{r_j}]_{e_s} \supseteq (W)_{r_j}$, it follows that $\mu_s(\|[(p_i)^*_{r_j}]_{e_s}\|)$ is at least $\mu_s(\|(W)_{r_j}\|)$, which by Corollary 6.1.12, is greater than or equal to $\mu(\|W\|) \times \text{Pr}(r_j) \geq b_i \times \text{Pr}(r_j)$.

  2. Suppose only $(p_1)^*_{r_1}$ and $(p_1)^*_{r_2}$ are the same. Then we need to show that $M_s \models \boldsymbol{B}(p_1)^*_{r_1} \geq b_1 \wedge \boldsymbol{B}(p_2)^*_{r_1} \geq b_2 \times \text{Pr}(r_1) \wedge \boldsymbol{B}(p_2)^*_{r_2} \geq b_2 \times \text{Pr}(r_2)$. Proving the beliefs about $(p_2)^*_{r_i}$ is what Lemma 6.2.23 demonstrates, and so we focus on $(p_1)^*_{r_1}$.

     For ease of exposition, suppose that $\mu(\|[p_1]_e\|)$ is a single equivalence class, say $\|w'\|$. Since $\mu(\|w'\|) \geq b$ by assumption, Lemma 6.1.11 establishes that $\mu_s(\|w'_{r_i}\|) \geq b \times \text{Pr}(r_i)$ for each i.

     So suppose that $w'_{r_1} \approx w'_{r_2}$, then $\|w'_{r_1}\|$ and $\|w'_{r_2}\|$ are the same. Since $[(p_1)^*_{r_1}]_{e_s} \supseteq \|w'_{r_1}\|$, it follows that $\mu_s(\|[(p_1)^*_{r_1}]_{e_s}\|)$ is at least $\mu_s(\|w'_{r_1}\|)$. By construction of μ_s we have $\mu_s(\|w'_{r_1}\|) = \mu(\|w'\|) \times \sum_i \text{Pr}(r_i)$ which is $\geq b_1$. On the other hand, if $w'_{r_1} \not\approx w'_{r_2}$ then $\|w'_{r_1}\|$ and $\|w'_{r_2}\|$ are disjoint. Since $[(p_1)^*_{r_1}]_{e_s} \supseteq \|w'_{r_1}\| \cup \|w'_{r_2}\|$ it follows that $\mu_s(\|[(p_1)^*_{r_1}]_{e_s}\|)$ is at least $\mu_s(\|w'_{r_1}\| \cup \|w'_{r_2}\|) \geq b_1 \times \text{Pr}(r_1) + b_1 \times \text{Pr}(r_2) \geq b_1$.

  Thus, this case is proved as well, which completes the proof. ∎

Note that in the above proposition we assumed without any loss of generality that the p_i's are different because of Corollary 6.2.24, which shows how belief atoms about the same literal in T are to be handled. Let us illustrate this proposition with an example:

Example 6.2.26. Let us revisit Example 6.2.2. If Σ is the basic action theory from that example, recall that we are dealing with a theory T of the form:

$$\boldsymbol{O}(\Sigma) \wedge \boldsymbol{B}(\mathit{distance} = 4) \geq .4 \wedge \boldsymbol{B}(\mathit{distance} = 5) \geq .6.$$

Now let M be any model of T and if we consider the progression of M wrt *noisyReverse*, then that model will satisfy the following belief atoms:

- $\boldsymbol{B}(distance = 6) \geq .6 \times .9$

  Updating the literal $distance = 5$ wrt *reverse*, which is executed with a probability of .9, results in $distance = 6$. By Proposition 6.2.25, it obtains a probability of the degree of belief for $distance = 5$ initially, *viz.* $\geq .6$, weakened by .9 which is $\geq .54$.

- $\boldsymbol{B}(distance = 4) \geq .4 \times .1$

  Updating the literal $distance = 4$ wrt *noop*, which is executed with a probability of .1, results in $distance = 4$ itself. By Proposition 6.2.25, it obtains a probability of the degree of belief for $distance = 4$ initially, *viz.* $\geq .4$, weakened by .1 which is $\geq .04$.

- $\boldsymbol{B}(distance = 5) \geq (.4 \times .9 + .6 \times .1)$

  Updating the literal $distance = 5$ wrt *noop*, which is executed with a probability of .1, results in $distance = 5$ itself. Additionally, updating the literal $distance = 4$ wrt *reverse*, which is executed with a probability of .9, also results in $distance = 5$. Therefore, by Proposition 6.2.25, $distance = 5$ obtains the sum of these two possibilities, *viz.* $\geq (.6 \times .1 + .4 \times .9)$, *i.e.* $\geq .42$. ∎

Now, we turn to what the agent should only know after doing a noisy action:

Proposition 6.2.27. *Suppose* $T = \boldsymbol{O}(\phi \wedge \Box\beta) \wedge \bigwedge_j \boldsymbol{B} p_j \geq b_j$ *is determinate wrt s. Let M be a model of T. Let s be a noisy action such that* $\Pi(s) = \{r_1, \ldots, r_k\}$. *Then* $M_s \models \boldsymbol{O}(\bigvee_i Prog(\phi, r_i) \wedge \Box\beta)$.

Proof: Let M be the tuple (e, w, μ, δ). Given that $w' \in e$ iff $e, w', \mu, \delta \models \phi \wedge \Box\beta$. We need to show that $w' \in e_s$ iff $e_s, w', \mu_s, \delta \models \bigvee_i Prog(\phi, r_i) \wedge \Box\beta$.

For the if direction, suppose $w' \in e_s$. By construction, there is some $w^* \in e$ and some $r \in \Pi(s)$ such that $w^*_r = w'$. By adapting Lemma 5.2.20, we can show that for each i, if r is r_i then $w' \models Prog(\phi, r_i) \wedge \Box\beta$. Thus $w' \models \bigvee_i Prog(\phi, r_i) \wedge \Box\beta$.

Conversely, suppose $w' \models \bigvee_i Prog(\phi, r_i) \wedge \Box\beta$. Then $w' \models Prog(\phi, r_i) \wedge \Box\beta$ for some r_i. By adapting Lemma 5.2.21, it can be shown that there is a world w^* such that $w^*_{r_i} = w'$ and such that $w^* \models \phi \wedge \Box\beta$. By assumption, $w^* \in e$ and thus, $w^*_{r_i} \in e_s$, or $w' \in e_s$. ∎

Example 6.2.28. Consider what the agent from Example 6.2.2 should only know after *noisyReverse*. By applying Proposition 6.2.27, we see that the progression of any model of

$$\boldsymbol{O}((distance = 4 \vee distance = 5) \wedge \Box\beta) \wedge \bigwedge \boldsymbol{B} p_i \geq b_i$$

must satisfy

$$\boldsymbol{O}((\varphi_1 \vee \varphi_2) \wedge \Box\beta)$$

where

- φ_1 is $Prog(distance = 4 \vee distance = 5, reverse)$, which is $\exists P[(P = 4 \vee P = 5) \wedge distance = x \equiv x = P + 1]$ that simplifies to $distance = 5 \vee distance = 6$; and

- φ_2 is $Prog(distance = 4 \vee distance = 5, noop)$, which is $\exists P[(P = 4 \vee P = 5) \wedge distance = x \equiv x = P]$ that simplifies to $distance = 4 \vee distance = 5$.

That is, we obtain $\boldsymbol{O}((distance = 4 \vee distance = 5 \vee distance = 6) \wedge \Box\beta)$.

It is worth noting that, not surprisingly, what the agent only knows after *noisyReverse* is compatible with the generated beliefs from Example 6.2.26. ∎

We now state the main result for this section.

Theorem 6.2.29. *Suppose $T = \boldsymbol{O}(\phi \wedge \Box\beta) \wedge \bigwedge \boldsymbol{B}p_i \geq b_i$ is determinate wrt s. Then the progression of T wrt s, where $\Pi(s) = \{r_1, \ldots, r_k\}$ is*

$$\boldsymbol{O}(\textstyle\bigvee_j Prog(\phi, r_j) \wedge \Box\beta)) \wedge$$

$$\textstyle\bigwedge_i \bigwedge_j (\boldsymbol{B}(p_i)^*_{r_j} \geq \sum_u b_u \times (\sum_{\{h \mid (p_u)^*_{r_h} \text{ is the same as } (p_i)^*_{r_j}\}} \Pr(h))).$$

Proof: Let T' denote the progression of T wrt s. Given any model of T', say $M = (e, w, \mu, \delta)$, we prove that there is a model M' of T such that $M'_s = M$.

Let e' be an epistemic state satisfying $\boldsymbol{O}(\phi \wedge \Box\beta)$. Clearly such an epistemic can be constructed and is unique. By means of Proposition 6.2.27, it is easy to see that e'_s satisfies $\boldsymbol{O}(\bigvee Prog(\phi, r_i) \wedge \Box\beta)$. As we argued in Theorem 6.2.5, an epistemic state satisfying $\boldsymbol{O}(\alpha)$ when α is a fluent sentence always exists and is unique. Therefore $e'_s = e$.

Next, for all primitive terms d, let w' be a world such that $w'[d, \langle\rangle]$ is an arbitrary name, and $w'[d, r_i \cdot z] = w[d, z]$ for all z.

We now construct a measure μ' satisfying the belief atoms in T using μ. For ease of exposition, let $\Pi(s) = \{r_1, r_2\}$ and let T mention only the belief atoms $\boldsymbol{B}p_1 \geq b_1$ and $\boldsymbol{B}p_2 \geq b_2$. (The case of $\Pi(s) = \{r_1, \ldots, r_k\}$ and T mentioning k' atoms is straightforward but tedious.) Since by assumption, T should contain beliefs about all possible values of a primitive term, the interesting case is when p_1 and p_2 are about the same primitive term. That is, worlds in e' either satisfy p_1 or p_2. Let us denote $(p_i)^*_{r_j}$ by p_{ij}. Similar to Proposition 6.2.25, the arguments differ slightly depending on which of the literals from $\{p_{11}, p_{12}, p_{21}, p_{22}\}$ are the same. Since the proofs are easy to adapt, we show the case when they are all different.

In this case, T' mentions $\boldsymbol{B}p_{ij} \geq b_i \times \Pr(r_j)$ for every i, j. Note then that $[p_{ij}]_e$ are disjoint for every i, j by assumption. Now, if $e - \bigcup_{i,j}[p_{ij}]_e \neq \emptyset$, then $\sum_i b_i \neq 1$. For suppose otherwise. Then $\mu(\| \bigcup_{i,j}[p_{ij}]_e\|) \geq b_1 + b_2$ by the disjointness property, and this would mean that $\|e - \bigcup_{i,j}[p_{ij}]_e\|$ obtains a probability of 0 if $b_1 + b_2 = 1$. That is, μ does not satisfy **M3** which contradicts our definition of a model.

Now suppose $\mu(\|[p_{11}]_e\|) = b_{11}$ and $\mu(\|[p_{12}]_e\|) = b_{12}$, where clearly $b_{1i} \geq b_1 \times \Pr(r_i)$ for every i. Construct $W_{1i} = \{w' \in e' \mid w' \models p_1, w'_{r_i} \models p_{1i}\}$. Since worlds in e' either satisfy p_1 or p_2, it follows that $W_{11} \cup W_{12} = [p_1]_{e'}$. Let μ' assign $\sum_i b_{1i}$ to $\|W_{11} \cup W_{12}\|$, where clearly $\sum_i b_{1i} \geq b_1$. Pursuing a similar construction for $[p_2]_{e'}$ will result in an assignment of a probability greater than or equal to b_2 to $\|[p_2]_{e'}\|$. Thus, all the constraints imposed by the belief atoms in T are satisfied by μ'.

Finally, if $e - \bigcup_{i,j}[p_{ij}]_e \neq \emptyset$ then $e' - \bigcup_{ij} W_{ij} \neq \emptyset$. We argued above that in this case $b_1 + b_2 \neq 1$ and so let μ' assign a probability of $1 - b_1 - b_2$ to $\|e' - \bigcup_{ij} W_{ij}\|$. Throughout when assigning a probability, say b, to a set of equivalence class, say of size k, let the probability on each equivalence class be b/k.

It is now easy to see that (e', w', μ', δ) is the desired model. ∎

Example 6.2.30. We now progress the theory T from Example 6.2.2 by using Theorem 6.2.29. The agent beliefs after *noisyReverse* was investigated in Example 6.2.26. Next, we investigated what the agent only knows after *noisyReverse* in Example 6.2.28.

Putting this together, the progression of T wrt *noisyReverse* is:

$$\boldsymbol{O}((distance = 4 \vee distance = 5 \vee distance = 6) \wedge \Box\beta) \wedge$$

$$\boldsymbol{B}(distance = 4) \geq .04 \wedge \boldsymbol{B}(distance = 5) \geq .42 \wedge \boldsymbol{B}(distance = 6) \geq .54. \blacksquare$$

6.3 Concluding Remarks

In this chapter, we proposed a new model for reasoning about uncertainty and action. Among the main features is a semantics that clarifies a notion of progression, closely related to Lin and Reiter's, in the presence of noisy actions and probabilistic beliefs. Our work is inspired by previous results on progression and noisy effectors. While we did not obtain a general result about the existence of progression in this setting, we obtained preliminary results for an important practical case. For this case, we are able to define the progression of a theory, containing both first-order beliefs and probabilistic ones, wrt ordinary and noisy actions. The results obtained seem to coincide with our intuitions regarding the synchronization of knowledge and beliefs under uncertainty.

The idea of assigning probabilities to possible worlds is based on previous proposals such as [Fagin and Halpern, 1994; Bacchus et al., 1995; Halpern, 2003], among others. Fagin and Halpern were among the first to consider a logical formalism to reason about knowledge and uncertainty. Fagin and Halpern even consider the many agent case, essentially by considering accessibility relations over the possible world for each agent, but for a propositional language. They also do not consider actions. Actions are also not considered in earlier first-order treatments about probability such as [Halpern, 1990]. While actions are dealt in [Halpern and Tuttle, 1993], and in [Van Benthem et al., 2009] for more recent work, they are not first-order formalisms. Similarly, the framework of Darwiche and Goldszmidt [1994], which integrates a model of actions and Bayesian nets [Pearl, 1988], is also not a first-order formalism. As Fagin and Halpern [1994] also point out, probabilistic knowledge has been of great concern to economists [Osborne and Rubinstein, 1994], although they do not consider formal languages. On a related note, probabilistic variants of dynamic logic have appeared in early program verification literature [Kozen, 1985], but typically with the intention of monitoring properties that hold after the probabilistic execution of programs. See [Fagin and Halpern, 1994] for discussions.

The closest approaches to our work is [Bacchus et al., 1995] and [Gabaldon and Lakemeyer, 2007], both of which do not propose a solution to the projection problem. Let us remark while [Gabaldon and Lakemeyer, 2007] is also in the framework of $\mathcal{ES}$, the amalgamation of a model of uncertainty with $\mathcal{ES}$ is considerably different from the one considered in this thesis. For example, after executing noisy actions, they only allow reasoning about probabilistic statements whose semantics is rather involved.

A number of directions present themselves for future work:

- We did not consider any noisy sensing, and this is essential for a complete specification of the agent. We believe an investigation in the line of [Gabaldon and Lakemeyer, 2007], where noisy sensors are

modeled exactly as noisy actions are, will prove fruitful while remaining coherent with the other technical details of our formalism.

- Regarding projection, we address reasoning about action by means of progression. After progressing, for the practical case considered in the chapter, the agent only knows a basic action theory and believes a conjunction of belief atoms. Because of that, reasoning about *knowledge*, that is, evaluating sentences of the form $\boldsymbol{K}\alpha$ where α is a basic formula wrt the progressed theory is a first-order theorem proving task due to the representation theorem (see Theorem 3.1.11 and Theorem 4.1.10.) However, we may also want to reason about *beliefs*, that is, evaluating sentences of the form $\boldsymbol{B}\alpha$ wrt the progressed theory. This is not addressed in the chapter and will be a topic for future research. Perhaps, for restricted cases, the decision procedure of Fagin and Halpern [1994] for reasoning about probabilities may provide hints as to how this problem can be approached.

A more general question is whether the results presented in this chapter regarding practical cases can be extended to a broader class of theories. But arguably, the most pressing issue in this direction is to resolve the question about whether progression always exists, and if it does, whether it can be given a finite representation.

Chapter 7

Conclusions

In this thesis, we proposed a general methodology for reasoning about incomplete knowledge bases with many agents, in dynamic domains. In particular, it is argued that having a knowledge base differs from simply believing a set of propositions in that it is meant to represent all that is known. This, in turn, implies believing those propositions while, simultaneously, not believing the propositions that do not follow from the knowledge base. Armed with this simple concept, we investigated various semantical and computational considerations that a knowledge-based agent has to address when solving projection tasks. The technical contributions of this thesis are as follows:

1. We extended Levesque's logic of only knowing $\mathcal{OL}$ to many agents. Among the prominent approaches to capture multiagent only knowing, ours is the first that is proposed for a quantified language with equality, while still maintaining all of the desirable properties of Levesque's framework. Levesque also proposed a sound and complete axiomatization for the propositional fragment of $\mathcal{OL}$. We then obtained an axiomatization that faithfully lifts Levesque's axiomatization to the many agent case. Finally, we also discussed the relationship to some of the earlier approaches.

2. Based on these results, we proposed an amalgamation of the situation calculus and multiagent only knowing. Our ideas directly extend the action formalism $\mathcal{ES}$, proposed originally by Lakemeyer and Levesque, that integrates Reiter's refinement of the situation calculus with the modal framework of $\mathcal{OL}$. By means of the regression operator, projection queries are reduced to static ones, and by means of the representation theorem, static queries about knowledge are reduced to pure first-order reasoning tasks.

3. We investigated the computational feasibility of Lin and Reiter's concept of progression in the context of our knowledge bases, which contained functional fluents. This addresses an important concern raised in the reasoning about action community that regression alone is not sufficient for projection tasks, especially during the operation of long-lived agents. Building on earlier first-order definability and computability results from [Vassos and Levesque, 2008; Liu and Lakemeyer, 2009; Vassos et al., 2009], we were able to prove the following:

   (a) For local-effect actions [Liu and Levesque, 2005a], we were able to show that progression is

first-order definable for arbitrary theories. When the initial knowledge base is a proper$^+$ KB, we proved that not only is progression first-order definable, it is also computable in linear time under reasonable assumptions.

(b) For normal actions [Liu and Lakemeyer, 2009], we were able to show that progression is first-order definable for theories that are semi-Horn wrt some functional fluents. When the initial knowledge base is a proper$^+$ KB and semi-Horn wrt the same set of fluents, we proved that progression is first-order definable and computable in linear time under reasonable assumptions.

(c) For range-restricted theories [Vassos et al., 2009], we were able to prove that when the initial knowledge base is a proper$^+$ KB, progression is efficient provided that the conditions under which an action affects objects is specified using information from the initial knowledge base.

(d) We were able to provide a novel sound and complete algorithm for evaluating a large class of queries against proper$^+$ KBs. This involved identifying conditions under which it suffices to consider a finite version of a proper$^+$ KB that, in general, is equivalent to a (possibly) infinite set of primitive clauses.

In terms of previous work, local-effects and normal actions in particular, {(a),(b)} generalized results from [Liu and Lakemeyer, 2009] by extending their computability result for function-free finite theories to finite theories mentioning functional fluents. Moreover, {(a),(b)} considers a strict generalization of the predicate-only proper$^+$ KB from [Liu and Lakemeyer, 2009] and proves that progression for these knowledge bases is also efficiently computable. Finally, (c) proves a variant of the definability and computability results from [Vassos et al., 2009] for proper$^+$ KBs.

4. We examined resolving projection tasks when there is nondeterminism in the execution of actions, and in the process the agent maintains degrees of belief. Our solution consisted of proposing a notion of progression which, in fact, is closely related to and inspired by Lin and Reiter's concept. In particular, we formalized a model-theoretic property regarding what the new knowledge base should look like, and identified a useful case where such a new knowledge base is definable given an initial knowledge base consisting of both ordinary (first-order) sentences as well as probabilistic beliefs.

We conclude with a brief list of topics for future research.

1. Extensions to the results obtained in this thesis remain to be explored:

   (a) In Chapter 4, we identified regression and representation theorems for one particular stipulation about the initial knowledge of multiple agents. In particular, we did not consider any (AEL) defaults when reasoning about action. It would be worthwhile to investigate how these theorems can be extended to other cases. It would also be interesting to investigate limitations to the initial knowledge bases and action theories, such that after the application of regression and representation theorems, reasoning about the initial knowledge base is efficient (or at least decidable, say, by means of the result in Chapter 5).

   (b) In Chapter 5, we investigated progression, but we restricted ourselves to the single agent case. We would like to extend our ideas from Chapter 3 further, and propose a semantical account of

progression with multiple agents. As we pointed out earlier, some preliminary work has been carried out in [Liu and Wen, 2011], although not in the context of only knowing.

(c) In Chapter 5, we proposed progression techniques for three classes of basic action theories. We would like to implement these procedures, and study under what conditions these procedures work well in practice. Since STRIPS and its extensions have been quite successful in the planning community, we believe progression-based solutions such as the one considered in this thesis offer techniques that are, at least in principle, able to handle a broader set of applications.

(d) In this thesis, we proposed a reasoning mechanism for proper$^+$ KBs wrt a class of queries. We would like to extend this class to also consider existentially quantified queries. Moreover, we would like to implement this procedure and compare it to existing state-of-the-art solvers over an encoding of the ground proper$^+$ KB (together with axioms about the uniqueness of names).

(e) In this thesis, we proposed a notion of progression under uncertainty. However, we were not able to obtain a general result regarding the definability of the progressed knowledge base in Chapter 6. An interesting question is whether progression always exists in this setting, and whether it is finitely representable. Moreover, we would like to extend our results to noisy sensing, and perhaps consider more practical cases where progression can be computed easily.

2. The main thrust of the thesis is to address certain knowledge representation and reasoning problems that arise in high-level control programs for autonomous agents, operating in incompletely known and dynamic worlds. The underlying assumption was that a tight coupling of such cognitive tasks and low-level behaviors can be achieved. However, this assumption, which allowed us to formalize and treat the agent's cognitive module in a clean and natural way, has to be examined closely. Think, for example, of the simple action of a robot moving forward. This action involves (say) starting its motors, issuing a low-level request of going forward by some units to a (calibrated) effector, stopping the motor after the action, *etc.* In other words, while we treated moving forward as a primitive action throughout this thesis, we see that in practical settings, it may not be one. This raises the question as to which set of actions are primitive and when should the inner workings of an action be made available to the agent? This brings to the forefront concerns about the *granularity* of *behavioral primitives*. On a more general level, there may be parts of the robot's software which perhaps operate by means of different mathematical representations. Examples include robotic mapping and localization [Thrun, 2002], and vision. It then becomes necessary to provide agent architectures that not only allow different modules to be parts of the same system, but these modules may need to interact with each other. Think of a robot fast approaching a wall, governed by a high-level control program that says that provided the fluent *NoObstable* is true, it should move forward. In this case, the robot must, by means of its vision system and object recognition software, recognize the wall as an obstacle and set *NoObstacle* to false. Moreover, if failures occur in the operation of these software components, the robot must be able to do a reasoned failure recovery. Pertinent questions such as these connect high-level control formalisms with traditional (low-level) robotics, thereby suggesting ways to realize and build autonomous and intelligent agents.

Appendix A

Long Proofs

A.1 Proof of the Regression Property

In this section, we prove Theorem 4.2.12. We begin with a few useful lemmas before turning to the theorem. In what follows, we will make use of the following special construction. Given a world w, we define another world w_Σ which is like w except that it satisfies $\Sigma_{pre}, \Sigma_{post}$ and Σ_{sense} sentences of Σ. Similarly, given w, we define $w_{\Sigma'}$ which is like w except that it satisfies the corresponding components of Σ'. We define w_Υ as another world which is like w except that it satisfies the corresponding components of Υ.

Definition A.1.1. Let w be a world, $z \in \mathcal{Z}$ and Σ a basic action theory over fluents $\mathcal{F}$. Then w_Σ is a world satisfying the following conditions:

1. for $f \notin \mathcal{F}$, $w_\Sigma[f(\vec{n}), z] = w[f(\vec{n}), z]$;
2. for $f \in \mathcal{F}$, w_Σ is defined inductively by:
   (a) $w_\Sigma[f(\vec{n}), \langle\rangle] = w[f(\vec{n}), \langle\rangle]$;
   (b) $w_\Sigma[f(\vec{n}), z \cdot r] = m$ iff $w_\Sigma, z \models (\gamma_f)^{v\ y\ \vec{x}}_{r\ m\ \vec{n}}$;
3. $w_\Sigma[Poss(r), z] = 1$ iff $w_\Sigma, z \models \pi^v_r$;
4. $w_\Sigma[SF_i(r), z] = m$ iff $w_\Sigma, z \models \varphi_{i\,r\ m}^{v\ x}$;

Note that this definition uses the *rhs* of Σ.

The following properties can be shown regarding w_Σ in relation to w:

Lemma A.1.2. *[Lakemeyer and Levesque, 2004]*

1. *For any w, w_Σ exists and is unique.*
2. *If $w \models \Sigma_0$ then $w_\Sigma \models \Sigma$.*
3. *If $w \models \Sigma$ then $w = w_\Sigma$.*

4. *Let α be any bounded objective sentence, and suppose that it is rectified and in* FNF. *Let $z \in \mathcal{Z}$. Then $w \models \mathcal{R}[z, \alpha]$ iff $w_\Sigma, z \models \alpha$.*

Proof: We show item 4. The proof is by induction on the length of α. We treat the length of $Poss(r)$ and $SF_i(r)$ as the length of π_r^v and $\varphi_{i_r}^v$ plus 1. We only consider the non-trivial cases below:

case $Poss(r)$.

We have $w_\Sigma, z \models Poss(r) = 1$

iff $w_\Sigma, z \models \pi_r^v$ by definition of w_Σ

iff $w \models \mathcal{R}[z, \pi_r^v]$ by induction

iff $w \models \mathcal{R}[z, Poss(r) = 1]$ by definition of $\mathcal{R}$.

case $SF_i(r) = m$.

We have $w_\Sigma, z \models SF_i(r) = m$

iff $w_\Sigma, z \models \varphi_{i_r}^v$ by definition of w_Σ

iff $w \models \mathcal{R}[z, \varphi_{i_r}^v]$ by induction

iff $w \models \mathcal{R}[z, SF_i(r) = m]$ by definition of $\mathcal{R}$.

case fluents $f \in \mathcal{F}$. Note that, by definition of FNF, ground atoms are of the form $f(\vec{n}) = m$. The proof is by sub-induction on z.

1. $w_\Sigma \models f(\vec{n}) = m$

   iff $w \models f(\vec{n}) = m$ by definition of w_Σ

   iff $w \models \mathcal{R}[\langle\rangle, f(\vec{n}) = m]$ by definition of $\mathcal{R}$.

2. $w_\Sigma, z \cdot r \models f(\vec{n}) = m$

   iff $w_\Sigma, z \models \gamma_{f_r}^{v\ y\ \vec{x}}{}_{m\ \vec{n}}$ by definition of w_Σ

   iff $w \models \mathcal{R}[z, \gamma_{f_r}^{v\ y\ \vec{x}}{}_{m\ \vec{n}}]$ by sub-induction

   iff $w \models \mathcal{R}[z \cdot r, f(\vec{n}) = m]$ by definition of $\mathcal{R}$. ∎

We now proceed to prove similar properties about epistemic states. Given e^k and a basic action theory Σ, let us define ${e_\Sigma}^k$ inductively by:

1. ${e_\Sigma}^1 = \{(w_\Sigma, \{\}) \mid (w, \{\}) \in e^1\}$;
2. ${e_\Sigma}^k = \{(w_\Sigma, {e_\Sigma}^{k-1}) \mid (w, e^{k-1}) \in e^k\}$.

In addition, for brevity, let

- $\psi = OKnow_\Sigma[A, k] \wedge OKnow_{\Sigma'}[B, j]$, and
- $\psi_0 = OKnow_{\Sigma_0}[A, k] \wedge OKnow_{\Sigma_0'}[B, j]$.

Then, item 2 of Lemma A.1.2 is extended for knowledge in the following manner.

Lemma A.1.3. *Suppose* $e_A^k, e_B^j, w \models \psi_0$. *Then* $e_{\Sigma_A}^k, e_{\Sigma'}{}_B^j, w \models \psi$.

Proof: Since $OKnow_\Sigma[i, *]$ is interpreted wrt i's epistemic state, the proof is a simple induction on the *modal depth* of the background theory. (Refer to Lemma 4.2.8 for the formal definition.) That is, when the modal depth of the background theory is l, then we have a sentence of the form $OKnow_{\Sigma_0}[A, k] \wedge OKnow_{\Sigma_0'}[B, j]$ such that $k \leq l$, $j \leq l$ and k or j is l.

The base case is a background theory of modal depth 1. That is, we may have a background theory of $\boldsymbol{O}_A\Sigma_0 \wedge \boldsymbol{O}_B\Sigma_0'$ (or $\boldsymbol{O}_A\Sigma_0$ or $\boldsymbol{O}_B\Sigma_0'$). So suppose $e_A^1, e_B^1, w \models \boldsymbol{O}_A(\Sigma_0) \wedge \boldsymbol{O}_B(\Sigma_0')$. We need to show that for all worlds w', $(w', \{\}) \in e_{\Sigma_A}^1$ iff $w' \models \Sigma$. The case of $e_{\Sigma'}{}_B^1$ is analogous.

Suppose $w \models \Sigma$. Then $w \models \Sigma_0$ and therefore, by assumption, $w \in e_A^1$. By Lemma A.1.2, $w = w_\Sigma$ and therefore, $(w, \{\}) \in e_{\Sigma_A}^1$.

Conversely, let $(w, \{\}) \in e_{\Sigma_A}^1$. By definition, there is a $(w', \{\}) \in e_A^1$ such that $w'_\Sigma = w$. But since $w' \models \Sigma_0$, it follows from Lemma A.1.2 that $w \models \Sigma$. Thus, $e_{\Sigma_A}^1, \{\}, w \models \boldsymbol{O}_A(\Sigma)$.

Assume that the hypothesis holds for theories of modal depth $k-1$, that is, if e_A^{k-1} satisfies $OKnow_{\Sigma_0}[A, k-1]$ then $e_{\Sigma_A}^k$ satisfies $OKnow_\Sigma[A, k-1]$. (This is stated for B analogously.) Now, suppose that $e_A^k, e_B^j, w \models \psi_0$. Then, $(w', e_B^{k-1}) \in e_A^k$ iff $e_A^k, e_B^{k-1}, w' \models \Sigma_0 \wedge OKnow_{\Sigma_0}[B, k-1]$. We have to prove that $(w', e_B^{k-1}) \in e_{\Sigma_A}^k$ iff $e_{\Sigma_A}^k, e_B^{k-1}, w' \models \Sigma \wedge OKnow_\Sigma[B, k-1]$. The argument is then symmetric for e_B^j.

Consider any e_B^{k-1} and w such that $e_{\Sigma_A}^k, e_B^{k-1}, w \models \Sigma \wedge OKnow_\Sigma[B, k-1]$. Now, consider $e_B'{}^{k-1}$ such that $\{\}, e_B'{}^{k-1}, w \models OKnow_{\Sigma_0}[B, k-1]$. Since $w \models \Sigma$, by Lemma A.1.2 $w = w_\Sigma$ and also, $w \models \Sigma_0$. It follows that $(w, e_B'{}^{k-1}) \in e_A^k$ by assumption. By induction hypothesis, $\{\}, e_{\Sigma'}{}_B^{k-1}, w \models OKnow_\Sigma[B, k-1]$. By definition, $(w, e_{\Sigma'}{}_B^{k-1}) \in e_{\Sigma_A}^k$. An easy argument shows that $e_{\Sigma'}{}_B^{k-1} = e_B^{k-1}$.

Conversely, consider any $(w, e_B^{k-1}) \in e_A^k$. By assumption, $\{\}, e_B^{k-1}, w \models \Sigma_0 \wedge OKnow_{\Sigma_0}[B, k-1]$. By Lemma 4.2.8, $w_\Sigma \models \Sigma$. By induction hypothesis, $\{\}, e_{\Sigma_B}^{k-1}, w \models OKnow_\Sigma[B, k-1]$. By definition, $(w_\Sigma, e_{\Sigma_B}^{k-1}) \in e_{\Sigma_A}^k$. ∎

We now generalize item 4 of Lemma A.1.2 for knowledge.

Lemma A.1.4. $e_A^k, e_B^j, w \models \mathcal{R}[\Upsilon, \Sigma, \Sigma', z, \alpha]$ *iff* $e_{\Sigma_A}^k, e_{\Sigma'}{}_B^j, w_\Upsilon, z \models \alpha$.

Proof: The proof is by induction on z, a sub-induction on α.

Let $z = \langle\rangle$. The case of objective formulas proceeds as in Lemma A.1.2. We now consider A-subjective formulas. The case of B-subjective formulas is symmetric.

We have $e_{\Sigma_A}^k, e_{\Sigma'}{}_B^j, w_\Upsilon, z \models \boldsymbol{K}_A\alpha$

iff for all $(w, e_B^{k-1}) \in e_{\Sigma_A}^k$, $e_{\Sigma_A}^k, e_B^{k-1}, w \models \alpha$

iff for all $(w, e_B^{k-1}) \in e_A^k$, $e_{\Sigma_A}^k, e_{\Sigma_B}^{k-1}, w_\Sigma \models \alpha$ by definition of $e_{\Sigma_A}^k$

iff for all $(w, e_B^{k-1}) \in e_A^k$, $e_A^k, e_B^{k-1}, w \models \mathcal{R}[\langle\rangle, \alpha]$ by sub-induction

iff $e_A^k, e_B^j, w \models \boldsymbol{K}_A\mathcal{R}[\langle\rangle, \alpha]$

iff $e_A^k, e_B^j, w \models \mathcal{R}[\langle\rangle, \boldsymbol{K}_A\alpha]$ by definition of $\mathcal{R}$.

Now, we consider the case of $z \cdot r$. The proof is precisely as in the base case, except for subjective formulas, which we prove as follows:

$e_{\Sigma_A}^k, e_{\Sigma'}{}_B^j, w_\Upsilon, z \cdot r \models \boldsymbol{K}_A\alpha$

iff $e_{\Sigma_A}^k, e_{\Sigma'}{}_B^j, w_\Upsilon, z \models \beta_r^v$ by Theorem 4.2.10 where β is the *rhs*

iff $e_A^k, e_B^j, w \models \mathcal{R}[z, \beta_r^v]$ by the main induction

iff $e_A^k, e_B^j, w \models \mathcal{R}[z \cdot r, \boldsymbol{K}_A\alpha]$ by definition of $\mathcal{R}$. ∎

We are now prepared to prove Theorem 4.2.12.

Proof: Let us denote $\Upsilon \wedge \psi$ as Γ and $\Upsilon_0 \wedge \psi_0$ as Γ_0.

For the only-if direction, suppose that $\Gamma \models \alpha$ and suppose that $e_A^k, e_B^j, w \models \Gamma_0$. That is, $w \models \Upsilon_0$ and by Lemma A.1.2, $w_\Upsilon \models \Upsilon$. Further, by Lemma A.1.3, $e_{\Sigma_A}^k, e_{\Sigma'}{}_B^j, w_\Upsilon \models \Gamma$. By assumption, $e_{\Sigma_A}^k, e_{\Sigma'}{}_B^j, w_\Upsilon \models \alpha$. Then, by Lemma A.1.4, $e_A^k, e_B^j, w \models \mathcal{R}[\langle\rangle, \alpha]$.

Conversely, suppose that $\Gamma_0 \models \mathcal{R}[\langle\rangle, \alpha]$ and let $e_A^k, e_B^j, w \models \Gamma$. Then $w \models \Upsilon_0$. Suppose that $e'{}_A^k, e'{}_B^j, w \models \psi_0$. By assumption $e'{}_A^k, e'{}_B^j, w \models \mathcal{R}[\langle\rangle, \alpha]$. By Lemma A.1.4, $e_{\Sigma'}{}_A^k, e_{\Sigma'}{}'_B^j, w_\Upsilon \models \alpha$. By Lemma A.1.2, $w_\Upsilon = w$. By Lemma A.1.3, $e_{\Sigma'}{}_A^k, e_{\Sigma'}{}'_B^j, w_\Upsilon \models \Gamma$. An easy argument shows that $e_{\Sigma'}{}_A^k = e_A^k$ and $e_{\Sigma'}{}'_B^j = e_B^j$. Therefore $e_A^k, e_B^j, w \models \alpha$. ∎

A.2 Proof of the Representation Theorem

To prove Theorem 4.3.2, we first obtain two useful results in $\mathcal{OL}_n$. Now, given the set of possible $\mathcal{OL}_n$-worlds $\mathcal{W}$ and an objective sentence ϕ, define the following:

- let $\mathcal{W}_\phi = \{w \mid w \models \phi\}$;
- let $e_\phi{}^1 = \mathcal{W}_\phi \times \{\{\}\}$;
- let $e_\phi{}^k = \{(w, e_\phi{}^{k-1}) \mid w \in \mathcal{W}_\phi\}$.

Lemma A.2.1. *Let ϕ and ϕ' be objective $\mathcal{OL}_n$ sentences and let $e_{\phi_A}^k$ and $e_{\phi'}{}_B^j$ be as above. Let α be any objective formula with free variables $\vec{x}$. For any vector of standard names $\vec{n}$ and world w:*

$$e_{\phi_A}^k, e_{\phi'}{}_B^j, w \models \boldsymbol{K}_A\alpha_{\vec{n}}^{\vec{x}} \text{ iff } \models \textsc{Res}[\alpha, \phi]_{\vec{n}}^{\vec{x}}.$$

Analogously for $\boldsymbol{K}_B\alpha_{\vec{n}}^{\vec{x}}$.

Proof: From Lemma 3.2.9, it follows that $e_{\phi_A}^k, \{\}, w \models \boldsymbol{K}_A\alpha_{\vec{n}}^{\vec{x}}$ iff $e_{\phi_A}{\downarrow_1^k}, \{\}, w \models \boldsymbol{K}_A\alpha_{\vec{n}}^{\vec{x}}$ because $\boldsymbol{K}_A\alpha$ has A-depth 1. So it is sufficient to show that:

$$e_{\phi_A}{\downarrow_1^k}, \{\}, w \models \boldsymbol{K}_A\alpha_{\vec{n}}^{\vec{x}} \text{ iff } \models \textsc{Res}[\alpha, \phi]_{\vec{n}}^{\vec{x}}. \tag{A.1}$$

Note that $e_{\phi_A}{\downarrow_1^k} = \{(w, \{\}) \mid w \models \phi\}$, and so (A.1) can be simply proved in $\mathcal{OL}$. The proof is given in [Levesque and Lakemeyer, 2001, see Lemma 7.2.2]. ∎

Theorem A.2.2. *Let α be any basic $\mathcal{OL}_n$ formula of maximal A,B-depth k, j and with free variables $\vec{x}$. Let ${e_{\phi}}_A^k, {e_{\phi'}}_B^j$ be as before, w any world, and $\vec{n}$ be a vector of names. Then*

$$ {e_{\phi}}_A^k, {e_{\phi'}}_B^j, w \models \alpha_{\vec{n}}^{\vec{x}} \textit{ iff } w \models \|\alpha\|_{\phi,\phi'\ \vec{n}}^{\ \vec{x}}. $$

Proof: The proof is by induction on the structure of α. If α is an atom or an equality, the lemma clearly holds since α is objective. By induction, the lemma also holds for negations, disjunctions and quantifiers.

Now, consider $\boldsymbol{K}_A\alpha$. (The case of $\boldsymbol{K}_B\alpha$ is analogous.) We have

${e_{\phi}}_A^k, \{\}, w \models \boldsymbol{K}_A\alpha_{\vec{n}}^{\vec{x}}$

iff ${e_{\phi}}_A^k, {e_B}^{k-1}, w' \models \alpha_{\vec{n}}^{\vec{x}}$ for all $(w', {e_B}^{k-1}) \in {e_{\phi}}_A^k$

iff $w' \models \|\alpha\|_{\phi,\phi'\ \vec{n}}^{\ \vec{x}}$ by the induction hypothesis

iff ${e_{\phi}}_A^k, \{\}, w \models \boldsymbol{K}_A\|\alpha\|_{\phi,\phi'\ \vec{n}}^{\ \vec{x}}$ since $\|\alpha\|_{\phi,\phi'\ \vec{n}}^{\ \vec{x}}$ is objective

iff $\models \textsc{Res}[\|\alpha\|_{\phi,\phi'\ \vec{n}}^{\ \vec{x}}, \phi]_{\vec{n}}^{\vec{x}}$ by Lemma A.2.1

iff $\models \|\boldsymbol{K}_A\alpha\|_{\phi,\phi'\ \vec{n}}^{\ \vec{x}}$ by definition of RES

iff $w \models \|\boldsymbol{K}_A\alpha\|_{\phi,\phi'\ \vec{n}}^{\ \vec{x}}$ because the result of RES is an objective formula that does not use predicates and function symbols. Therefore, $\|\boldsymbol{K}_A\alpha\|_{\phi,\phi'\ \vec{n}}^{\ \vec{x}}$ is either valid or its negation is valid. ∎

We now consider the first main result about the representation theorem.

Theorem A.2.3. *Suppose α is of maximal A,B-depth k, j. Let ϕ, ϕ' and θ be objective $\mathcal{OL}_n$ sentences. Then*

$$ \theta \wedge \psi \models \alpha \textit{ iff } \models \theta \supset \|\alpha\|_{\phi,\phi'}. $$

where $\psi = OKnow_{\phi}[A,k] \wedge OKnow_{\phi'}[B,j]$.

Proof: For the if direction, suppose (e_A^k, e_B^j, w) is a model of $\psi \wedge \theta$. It is easy to verify that $e_A^k = {e_{\phi}}_A^k$ and $e_B^j = {e_{\phi'}}_B^j$, and so, w is any world satisfying θ. Since $\psi \wedge \theta \models \alpha$, $e_A^k, e_B^j, w \models \alpha$ iff $w \models \|\alpha\|_{\phi,\phi'}$ by Theorem A.2.2. So any model of θ satisfies $\|\alpha\|_{\phi,\phi'}$. Therefore, $\theta \models \|\alpha\|_{\phi,\phi'}$ or $\models \theta \supset \|\alpha\|_{\phi,\phi'}$.

Conversely, suppose $\theta \models \|\alpha\|_{\phi,\phi'}$. Now, let (e_A^k, e_B^j, w) be any model of $\psi \wedge \theta$. It is easy to verify that $e_A^k = {e_{\phi}}_A^k$ and $e_B^j = {e_{\phi'}}_B^j$. Further, since $w \models \theta$ we have $w \models \|\alpha\|_{\phi,\phi'}$. By Theorem A.2.2, $e_A^k, e_B^j, w \models \alpha$. ∎

We now turn to the proof for Theorem 4.3.2, which follows rather directly from the above result.

Proof: Consider that $\Upsilon \wedge OKnow_{\Sigma}[A,k] \wedge OKnow_{\Sigma'}[B,j] \models \alpha$

iff $\Upsilon_0 \wedge OKnow_{\Sigma_0}[A,k] \wedge OKnow_{\Sigma_0'}[B,j] \models \mathcal{R}[\langle\rangle, \alpha]$ by the regression property (Theorem 4.2.12)

iff $\Upsilon_0 \models \|\mathcal{R}[\langle\rangle, \alpha]\|_{\Sigma_0,\Sigma_0'}$ by Theorem A.2.3 because

1. Υ_0, Σ_0 and ${\Sigma_0}'$ are fluent sentences and therefore objective $\mathcal{OL}$-sentences, and
2. $\mathcal{R}[\langle\rangle, \alpha]$ is a basic $\mathcal{OL}$-sentence by a straightforward adaptation of Lemma 4.1.12. ∎

A.3 Proof of Compactness

In this section, we prove Theorem 5.5.9. The basic idea will be to show given a proper$^+$ KB ϕ and a closed quantifier-free formula α, $gnd(\phi) \cup \{\neg\alpha\}$ is satisfiable iff a propositional encoding of that theory is satisfiable in propositional logic (**PL**). This will then allow us to invoke the Compactness property for **PL** so as to say that if all finite subsets of $gnd(\phi) \cup \{\neg\alpha\}$ is satisfiable then so is $gnd(\phi) \cup \{\neg\alpha\}$.

We begin by showing that for any proper$^+$ KB ϕ, if $f(\vec{m})$ is a primitive term mentioned in $gnd(\phi)$, then $gnd(\phi)$ only mentions a finite number of equalities mentioning $f(\vec{m})$. We identify this property as the *limit property*. More precisely,

Definition A.3.1. (Limit property.) Given a (possibly infinite) set of primitive clauses S, we say that S has the limit property if for every $f(\vec{m})$ mentioned in S, the set

$$\{f(\vec{m}) \circ n \mid \circ \in \{=, \neq\}, f(\vec{m}) \circ n \text{ appears in a clause in } S\}$$

is finite. ∎

Proposition A.3.2. *Suppose ϕ is a proper$^+$ KB. For every primitive term $f(\vec{m})$ mentioned in $gnd(\phi)$, the set*

$$\{f(\vec{m}) \circ n \mid \circ \in \{=, \neq\}, f(\vec{m}) \circ n \textit{ appears in a clause in } gnd(\phi)\}$$

is finite.

Proof: The proof is a rather simple one, since by definition, a proper$^+$ KB ϕ is a satisfiable and finite set of sentences of the form

$$\bigwedge_i (\varepsilon_i \supset f_i^1(\vec{t}_i^{\,1}) \circ_i^1 n_i^1 \vee \ldots \vee f_i^k(\vec{t}_i^{\,k}) \circ_i^k n_i^k)$$

where $\vec{t}_i^{\,j}$ are either variables or names. So then, an equality $f(\vec{m}) \circ n$ appears in a clause, say $c\theta$, in $gnd(\phi)$ only if $\forall(\varepsilon \supset c) \in \phi, \models e\theta$ for some substitution of variables with names θ. That is, while $gnd(\phi)$ may mention an infinite set of primitive terms, each primitive term $f(\vec{m})$ appears in only finitely many equalities, say $f(\vec{m}) \circ n_1, \ldots, f(\vec{m}) \circ n_k$, such that $f(\vec{t}) \circ n_i$ appears in some c and $\forall(\varepsilon \supset c) \in \phi$. ∎

Corollary A.3.3. *Suppose ϕ is a proper$^+$ KB and α is a quantifier-free closed formula. Then $gnd(\phi) \cup \{\neg\alpha\}$ has the limit property, assuming without loss of generality that $\neg\alpha$ is represented as a set of primitive clauses.*

Proof: By Proposition A.3.2 and the fact that α is a quantifier-free sentence. ∎

For the next step, we demonstrate how to construct a propositional theory from a (possibly infinite) set of primitive clauses S satisfying the limit property. We will assume a propositional language $\mathcal{L}'$ such that for each fluent primitive term $f(\vec{m}) \in \mathcal{ES}_o$, there are an infinite number of propositional variables $p_{\#0}^{f(\vec{m})}, p_{\#1}^{f(\vec{m})}, \ldots$ in $\mathcal{L}'$. Intuitively, every primitive equality in $\mathcal{ES}_o$ has a corresponding proposition in $\mathcal{L}'$. So then given S let:

- Γ_S be a set of sentences constructed in the following manner:

  For every clause $f_1(\vec{m}_1) \circ_1 n_1 \vee \ldots \vee f_k(\vec{m}_k) \circ_k n_k$ in S, where $\circ_i \in \{=, \neq\}$, let Γ_S include the sentence $\diamond_1 p_{n_1}^{f_1(\vec{m}_1)} \vee \ldots \vee \diamond_k p_{n_k}^{f_k(\vec{m}_k)}$, where $\diamond_i$ is the $\neg$ symbol if $\circ_i$ is $\neq$ and dropped otherwise. Intuitively, we are constructing a propositional formula that replaces equalities and inequalities by their corresponding propositional variables and their negations respectively.

- Δ_S be a set of sentences constructed in the following manner:

  For every $f(\vec{m})$ appearing in S, let $\{f(\vec{m}) \circ_1 n_1, \ldots, f(\vec{m}) \circ_k n_k\}$ be all the appearances of $f(\vec{m})$ in equalities in S. Recall that, due to the limit property, we can enumerate a finite set of equalities as required. Now, add the following sentences to Δ_S

$$p_{n_i}^{f(\vec{m})} \supset \neg(\bigwedge p_{n_j}^{f(\vec{m})}) \quad \text{for } i \neq j \text{ and } i, j \in \{1, \ldots, k\}.$$

We now prove the notion of satisfiability that we are after.

Theorem A.3.4. *Suppose S is a (possibly infinite) set of primitive clauses satisfying the limit property. Then for some w, $w \models S$ iff $\Delta_S \cup \Gamma_S$ is satisfiable in* **PL**.

Proof: We begin by constructing a *Boolean valuation*, that is, a mapping from propositional variables to $\{0, 1\}$, using the world w. More precisely, let us define a Boolean valuation $\mathcal{V}^w$ from w as follows:

For every primitive term $f(\vec{m})$, $w \models f(\vec{m}) = n$ iff $\mathcal{V}^w(p_n^{f(\vec{m})}) = 1$.

That is, $\mathcal{V}^w$ is well-defined in the sense that it assigns a truth value to every variable from $\mathcal{L}'$. We extend the definition of $\mathcal{V}^w$ for arbitrary propositional formulas over Boolean connectives in an obvious way [see, for instance, Smullyan, 1995]. Now, it is not hard to see that Δ_S is true under $\mathcal{V}^w$. So what we will prove is that $w \models S$ iff Γ_S is true under $\mathcal{V}^w$.

Now, let $c = f_1(\vec{m}_1) \circ_1 n_1 \vee \ldots \vee f_k(\vec{m}_k) \circ_k n_k$ be any primitive clause in S. We have $w \models \bigvee f_i(\vec{m}_i) \circ_i n_i$

iff $w \models f_i(\vec{m}_i) \circ_i n_i$ for some i, by definition

iff $\mathcal{V}^w(p_{n_i}^{f_i(\vec{m}_i)}) = Bool_i$ where $Bool_i$ is 0 if $\circ_i$ is $\neq$ and 1 otherwise, by construction

iff $\diamond_i p_{n_i}^{f_i(\vec{m}_i)}$ is true under $\mathcal{V}^w$, where $\diamond_i$ is $\neg$ if $\circ_i$ is $\neq$ and dropped otherwise

iff $\bigvee \diamond_i p_{n_i}^{f_i(\vec{m}_i)}$ is true under $\mathcal{V}^w$

iff the propositional encoding of c in Γ_S is true under $\mathcal{V}^w$.

That is, every primitive clause $c \in S$ is satisfied at w iff its propositional encoding in Γ_S is true under $\mathcal{V}^w$. Therefore $w \models S$ iff Γ_S is true under $\mathcal{V}^w$. ∎

Corollary A.3.5. *Suppose S is as above. Then S is satisfiable iff $\Gamma_S \cup \Delta_S$ is satisfiable in* **PL**.

With this in hand, we prove Theorem 5.5.9.

Proof: The only-if direction is immediate. Now, consider that S has the limit property by Corollary A.3.3. Clearly, then, every finite subset $S' \subseteq S$ also has the limit property. By Corollary A.3.5, S' is satisfiable iff $\Gamma_{S'} \cup \Delta_{S'}$ is satisfiable in **PL**. By assumption, then, every finite subset of $\Gamma_S \cup \Delta_S$ is satisfiable in **PL**. By Compactness property, $\Gamma_S \cup \Delta_S$ is satisfiable. Therefore, by Corollary A.3.5, S is satisfiable. ∎

BIBLIOGRAPHY

Abiteboul, S., R. Hull, and V. Vianu [1995]. *Foundations of Databases*. Addison-Wesley.

Ackermann, W. [1935]. Untersuchungen über das eliminationsproblem der mathematischen logik. *Mathematische Annalen 110*(1), 390–413.

Ackermann, W. [1962]. *Solvable cases of the decision problem*. North-Holland.

Alchourròn, C. E., P. Gärdenfors, and D. Makinson [1985]. On the logic of theory change: Partial meet contraction and revision functions. *Journal of Symbolic Logic 50*, 510–530.

Amir, E. and S. Russell [2003]. Logical filtering. In *Proc. IJCAI*, pp. 75–82.

Audemard, G., P. Bertoli, A. Cimatti, A. Kornilowicz, and R. Sebastiani [2002]. A SAT based approach for solving formulas over boolean and linear mathematical propositions. In *Proc. International Conference on Automated Deduction*, pp. 195–210.

Bacchus, F., J. Y. Halpern, and H. J. Levesque [1995]. Reasoning about noisy sensors and effectors in the situation calculus. In *Proc. IJCAI*, pp. 1933–1940.

Bacchus, F., J. Y. Halpern, and H. J. Levesque [1999]. Reasoning about noisy sensors and effectors in the situation calculus. *Artificial Intelligence 111*(1–2), 171 – 208.

Badban, B. and J. van de Pol [2005]. Zero, successor and equality in BDDs. *Annals of Pure and Applied Logic 133*(1-3), 101–123.

Badban, B., J. van de Pol, O. Tveretina, and H. Zantema [2007]. Generalizing DPLL and satisfiability for equalities. *Information and Computation 205*, 1188–1211.

Baral, C. and M. Gelfond [2005]. Logic programming and reasoning about actions. In *Handbook of Temporal Reasoning in Artificial Intelligence*, pp. 389–426. Elsevier.

Barrett, C., D. Dill, and A. Stump [2000]. A framework for cooperating decision procedures. In *Proc. International Conference on Automated Deduction*, pp. 79–98.

Baumgartner, P. [2000]. FDPLL−a first order Davis-Putnam-Longeman-Loveland procedure. In *Proc. International Conference on Automated Deduction*, pp. 200–219.

Belle, V. and G. Lakemeyer [2010a]. Multi-agent only-knowing revisited. In *Proc. KR*, pp. 49–60.

Belle, V. and G. Lakemeyer [2010b]. Reasoning about imperfect information games in the epistemic situation calculus. In *Proc. AAAI*, pp. 255–261.

Belle, V. and G. Lakemeyer [2011a]. Multi-agent only-knowing. In G. Lakemeyer and S. A. McIlraith (Eds.), *Knowing, Reasoning, and Acting: Essays in Honour of Hector J. Levesque*, pp. 67–86. College Publications.

Belle, V. and G. Lakemeyer [2011b]. On progression and query evaluation in first-order knowledge bases with function symbols. In *Proc. IJCAI*, pp. 255–260.

Belle, V. and G. Lakemeyer [2011c]. A semantical account of progression in the presence of uncertainty. In *Proc. AAAI*, pp. 165–170.

Blackburn, P., J. Kamps, and M. Marx [2001]. Situation calculus as hybrid logic: First steps. In *Proc. Portuguese Conference on Artificial Intelligence on Progress in Artificial Intelligence, Knowledge Extraction, Multi-agent Systems, Logic Programming and Constraint Solving*, pp. 253–260.

Boerger, E., E. Grädel, and Y. Gurevich [1997]. *The classical decision problem*. Springer Verlag.

Boutilier, C., R. Reiter, M. Soutchanski, and S. Thrun [2000]. Decision-theoretic, high-level agent programming in the situation calculus. In *Proc. AAAI*, pp. 355–362.

Burch, J. and D. Dill [1994]. Automatic verification of pipelined microprocessor control. In *Proc. Computer Aided Verification*, pp. 68–80.

Castilho, M., O. Gasquet, and A. Herzig [1999]. Formalizing action and change in modal logic I: the frame problem. *Journal of Logic and Computation 9*(5), 701–735.

Chellas, B. [1980]. *Modal logic*. Cambridge University Press.

Cimatti, A. and M. Roveri [2000]. Conformant planning via symbolic model checking. *Journal of Artificial Intelligence Research 13*, 305–338.

Darwiche, A. and M. Goldszmidt [1994]. Action networks: A framework for reasoning about actions and change under uncertainty. In *Proc. UAI*, pp. 136–144.

Davis, M., G. Logemann, and D. Loveland [1962]. A machine program for theorem-proving. *Commun. ACM 5*(7), 394–397.

Davis, M. and H. Putnam [1960]. A computing procedure for quantification theory. *J. ACM 7*(3), 201–215.

De Giacomo, G., H. Levesque, and S. Sardina [2001]. Incremental execution of guarded theories. *ACM Transactions on Computational Logic 2*(4), 495–525.

De Giacomo, G. and H. J. Levesque [1999]. Projection using regression and sensors. In *Proc. IJCAI*, pp. 160–165.

De Giacomo, G. and T. Mancini [2004]. Scaling up reasoning about actions using relational database technology. In *Proc. AAAI*, pp. 245–256.

de Lima, T. [2007]. *Optimal Methods for Reasoning about Actions and Plans in Multi-Agent Systems*. Ph.D. thesis, IRIT, University of Toulouse.

Demolombe, R. [2003]. Belief change: from situation calculus to modal logic. In *Proc. Nonmonotonic Reasoning, Action, and Change (NRAC)*.

Demolombe, R., A. Herzig, and I. Varzinczak [2003]. Regression in modal logic. *Journal of Applied Non-Classical Logics 13*(2), 165–185.

Dowling, W. and J. Gallier [1984]. Linear-time algorithms for testing the satisfiability of propositional horn formulae. *The Journal of Logic Programming 1*(3), 267–284.

Enderton, H. [1972]. *A mathematical introduction to logic*. Academic press New York.

Fagin, R. and J. Y. Halpern [1994]. Reasoning about knowledge and probability. *J. ACM 41*(2), 340–367.

Fagin, R., J. Y. Halpern, Y. Moses, and M. Y. Vardi [1995]. *Reasoning About Knowledge*. The MIT Press.

Fikes, R., P. Hart, and N. Nilsson [1972]. Learning and executing generalized robot plans. *Artificial intelligence 3*, 251–288.

Fikes, R. and N. J. Nilsson [1971]. STRIPS: A new approach to the application of theorem proving to problem solving. In *Proc. IJCAI*, pp. 608–620.

Fritz, C. [2009]. *Monitoring the Generation and Execution of Optimal Plans*. Ph.D. thesis, University of Toronto.

Gabaldon, A. and G. Lakemeyer [2007]. ESP: A logic of only-knowing, noisy sensing and acting. In *Proc. AAAI*, pp. 974–979.

Gabbay, D. and H. Ohlbach [1992]. Quantifier elimination in second-order predicate logic. In *Proc. KR*, pp. 425–435.

Gelfond, M. and V. Lifschitz [1993]. Representing action and change by logic programs. *The Journal of Logic Programming 17*(2-4), 301–321.

Gelfond, M. and V. Lifschitz [1998]. Action languages. *Electronic Transactions on Artificial Intelligence 2*, 193–210.

Ghallab, M., D. Nau, and P. Traverso [2004]. *Automated Planning: theory and practice*. Morgan Kaufmann Publishers.

Gomes, C. P., H. Kautz, A. Sabharwal, and B. Selman [2008]. Satisfiability solvers. In *Handbook of Knowledge Representation*, pp. 89–134. Elsevier.

Gottlob, G. [1993]. The power of beliefs or translating default logic into standard autoepistemic logic. In *Proc. IJCAI*, pp. 570–575.

Groote, J. and J. van de Pol [2000]. Equational binary decision diagrams. In *Proc. International Conference on Logic for Programming and Automated Reasoning*, pp. 161–178.

Halmos, P. [1950]. Measure theory. *Van Nostrad Reinhold Company*.

Halpern, J. [1990]. An analysis of first-order logics of probability. *Artificial Intelligence 46*(3), 311–350.

Halpern, J. [1997]. A Critical Reexamination of Default Logic, Autoepistemic Logic, and Only Knowing. *Computational Intelligence 13*(1), 144–163.

Halpern, J. and G. Lakemeyer [2001]. Multi-agent only knowing. *Journal of Logic and Computation 11*(1), 251–265.

Halpern, J. Y. [1993]. Reasoning about only knowing with many agents. In *Proc. AAAI*, pp. 655–661.

Halpern, J. Y. [2003]. *Reasoning about Uncertainty*. The MIT Press.

Halpern, J. Y. and G. Lakemeyer [1995]. Levesque's axiomatization of only knowing is incomplete. *Artificial Intelligence 74*(2), 381–387.

Halpern, J. Y. and Y. Moses [1984]. Towards a theory of knowledge and ignorance: Preliminary report. In *Proc. NMR*, pp. 125–143.

Halpern, J. Y. and M. R. Tuttle [1993]. Knowledge, probability, and adversaries. *J. ACM 40*, 917–960.

Harel, D., D. Kozen, and J. Tiuryn [2000]. *Dynamic logic*. The MIT Press.

Herzig, A., J. Lang, D. Longin, and T. Polacsek [2000]. A logic for planning under partial observability. In *Proc. AAAI / IAAI*, pp. 768–773.

Hindriks, K., F. De Boer, W. Van der Hoek, and J. Meyer [1999]. Agent programming in 3APL. *Proc. AAMAS 2*(4), 357–401.

Hintikka, J. [1962]. *Knowledge and belief: an introduction to the logic of the two notions*. Cornell University Press.

Hölldobler, S. and J. Schneeberger [1990]. A new deductive approach to planning. *New Generation Computing 8*(3), 225–244.

Iocchi, L., T. Lukasiewicz, D. Nardi, and R. Rosati [2009]. Reasoning about actions with sensing under qualitative and probabilistic uncertainty. *ACM Transactions on Computational Logic 10*, 5:1–5:41.

Johnson, D. [1990]. A catalog of complexity classes. In *Handbook of Theoretical Computer Science, Volume A: Algorithms and Complexity (A)*, pp. 67–161. Elsevier.

Kaplan, D. [1968]. Quantifying in. *Synthese 19*(1), 178–214.

Kelly, R. F. and A. R. Pearce [2008]. Complex epistemic modalities in the situation calculus. In *Proc. KR*, pp. 611–620.

Konolige, K. [1989]. On the relation between autoepistemic logic and circumscription. In *Proc. IJCAI*, pp. 1213–1218.

Kowalski, R. and M. Sergot [1986]. A logic-based calculus of events. *New Generation Computing 4*, 67–95.

Kozen, D. [1985]. A probabilistic PDLL. *Journal of Computer and System Sciences 30*(2), 162–178.

Kripke, S. [1959]. A completeness theorem in modal logic. *Journal of Symbolic Logic 24*(1), 1–14.

Kripke, S. [1963]. Semantical considerations on modal logic. *Acta Philosophica Fennica 16*, 83–94.

Kripke, S. A. [1976]. Is there a problem about substitutional quantification? In *Truth and Meaning*, pp. 324–419. Oxford University Press.

Lakemeyer, G. [1993]. All they know: A study in multi-agent autoepistemic reasoning. In *Proc. IJCAI*, pp. 376–381.

Lakemeyer, G. [1996]. Only knowing in the situation calculus. In *Proc. KR*, pp. 14–25.

Lakemeyer, G. and H. Levesque [1998]. AOL: a logic of acting, sensing, knowing, and only knowing. In *Proc. KR*, pp. 316–329.

Lakemeyer, G. and H. Levesque [2009]. A semantical account of progression in the presence of defaults. In *Conceptual Modeling: Foundations and Applications*, pp. 82–98. Springer.

Lakemeyer, G. and H. J. Levesque [2002]. Evaluation-based reasoning with disjunctive information in first-order knowledge bases. In *Proc. KR*, pp. 73–81.

Lakemeyer, G. and H. J. Levesque [2004]. Situations, si! situation terms, no! In *Proc. KR*, pp. 516–526.

Lakemeyer, G. and H. J. Levesque [2007]. Cognitive robotics. In *Handbook of Knowledge Representation*, pp. 869–886. Elsevier.

Lakemeyer, G. and H. J. Levesque [2011]. A semantic characterization of a useful fragment of the situation calculus with knowledge. *Artificial Intelligence 175*, 142–164.

Lenzen, W. [1978]. *Recent work in epistemic logic*. North-Holland.

Levesque, H. [1984]. Foundations of a functional approach to knowledge representation. *Artificial Intelligence 23*(2), 155–212.

Levesque, H. [2005]. Planning with loops. In *Proc. IJCAI*, pp. 509–515.

Levesque, H. and G. Lakemeyer [2001]. *The logic of knowledge bases*. The MIT Press.

Levesque, H. and R. Reiter [1998]. High-level robotic control: Beyond planning. Position paper at AAAI Spring Symposium on Integrating Robotics Research.

Levesque, H., R. Reiter, Y. Lespérance, F. Lin, and R. Scherl [1997]. Golog: A logic programming language for dynamic domains. *Journal of Logic Programming 31*, 59–84.

Levesque, H. J. [1990]. All I know: a study in autoepistemic logic. *Artificial Intelligence 42*(2-3), 263–309.

Levesque, H. J. [1996]. What is planning in the presence of sensing? In *Proc. AAAI / IAAI*, pp. 1139–1146.

Levesque, H. J. [1998]. A completeness result for reasoning with incomplete first-order knowledge bases. In *Proc. KR*, pp. 14–23.

Liberatore, P. [1997]. The complexity of the language A. *Electronic Transactions on Artificial Intelligence*, 13–38.

Lin, F. and R. Reiter [1994]. Forget it. In *Working Notes of AAAI Fall Symposium on Relevance*, pp. 154–159.

Lin, F. and R. Reiter [1997]. How to progress a database. *Artificial Intelligence 92*(1-2), 131–167.

Liu, Y. and G. Lakemeyer [2009]. On first-order definability and computability of progression for local-effect actions and beyond. In *Proc. IJCAI*, pp. 860–866.

Liu, Y. and H. Levesque [2005a]. Tractable reasoning with incomplete first-order knowledge in dynamic systems with context-dependent actions. In *Proc. IJCAI*, pp. 522–527.

Liu, Y. and H. J. Levesque [2005b]. Tractable reasoning in first-order knowledge bases with disjunctive information. In *Proc. AAAI*, pp. 639–644.

Liu, Y. and X. Wen [2011]. On the progression of knowledge in the situation calculus. In *Proc. IJCAI*, pp. 976–982.

Lloyd, J. [1987]. *Foundations of Logic Programming*. Springer Verlag.

McCarthy, J. [1968]. Situations, Actions and Causal Laws. Technical report, Stanford University, 1963. Also in M. Minsky (ed.), Semantic Information Processing, MIT Press, Cambridge, MA.

McCarthy, J. and P. J. Hayes [1969]. Some philosophical problems from the standpoint of artificial intelligence. In *Machine Intelligence*, Volume 4, pp. 463–502.

McDermott, D., M. Ghallab, A. Howe, C. Knoblock, A. Ram, M. Veloso, D. Weld, and D. Wilkins [1998]. PDDL−the planning domain definition language. Technical report, Yale Center for Computational Vision and Control.

Mitchell, D. G., B. Selman, and H. J. Levesque [1992]. Hard and easy distributions of sat problems. In *Proc. AAAI*, pp. 459–465.

Moore, R. C. [1985a]. A Formal Theory of Knowledge and Action. In J. R. Hobbs and R. C. Moore (Eds.), *Formal Theories of the Commonsense World*, pp. 319–358. Norwood, NJ: Ablex.

Moore, R. C. [1985b]. Semantical considerations on nonmonotonic logic. *Artificial Intelligence 25*(1), 75–94.

Morgenstern, L. and S. A. McIlraith [2011]. John McCarthy's legacy. *Artificial Intelligence 175*(1), 1 – 24.

Mueller, E. [2008]. Event calculus. In *Handbook of Knowledge Representation*. Elsevier.

Newell, A. [1993]. Reflections on the knowledge level. *Artificial Intelligence 59*(1-2), 31–38.

Nonnengart, A., H. J. Ohlbach, and A. Szalas [1999]. Quantifier elimination for second-order predicate logic. *Logic, Language and Reasoning. Essays in honour of Dov Gabbay, Part I, Kluwer Academic Press*.

Osborne, M. J. and A. Rubinstein [1994]. *A Course in Game Theory*. The MIT Press.

Pearl, J. [1988]. *Probabilistic reasoning in intelligent systems: networks of plausible inference*. Morgan Kaufmann.

Pednault, E. [1989]. ADL: Exploring the middle ground between STRIPS and the situation calculus. In *Proc. KR*, pp. 324–332.

Pnueli, A., Y. Rodeh, O. Shtrichman, and M. Siegel [1999]. Deciding equality formulas by small domains instantiations. In *Proc. International Conference on Automated Deduction*, pp. 687–688.

Pratt-Hartmann, I. [2000]. Total knowledge. In *Proc. AAAI*, pp. 423–428.

Prior, A. [1967]. *Past, present and future*. Oxford University Press.

Pylyshyn, Z. [1986]. *Computation and cognition: Toward a foundation for cognitive science.* The MIT Press.

Reiter, R. [1977]. On closed world data bases. In *Logic and Databases*, pp. 55–76.

Reiter, R. [1984]. Towards a logical reconstruction of relational database theory. In *On Conceptual Modelling: Perspectives from Artificial Intelligence Databases and Programming Languages*, pp. 191–238. Springer Verlag.

Reiter, R. [1987]. Nonmonotonic reasoning. *Annual Review of Computer Science 2*(1), 147–186.

Reiter, R. [1991]. The frame problem in situation the calculus: a simple solution (sometimes) and a completeness result for goal regression. In *Artificial intelligence and mathematical theory of computation: Papers in honor of John McCarthy*, pp. 359–380. Academic Press.

Reiter, R. [1992]. Formalizing database evolution in the situation calculus. In *Proc. International Conference on Fifth Generation Computer Systems*, pp. 600–609.

Reiter, R. [2001]. *Knowledge in action: logical foundations for specifying and implementing dynamical systems*. The MIT Press.

Rogers Jr., H. [1987]. *Theory of recursive functions and effective computability*. The MIT Press.

Rosati, R. [2000]. On the decidability and complexity of reasoning about only knowing. *Artificial Intelligence 116*(1-2), 193–215.

Scherl, R. B. and H. J. Levesque [2003]. Knowledge, action, and the frame problem. *Artificial Intelligence 144*(1-2), 1–39.

Schoppers, M. [1987]. Universal plans for reactive robots in unpredictable environments. In *Proc. IJCAI*, pp. 1039–1046.

Shanahan, M. [1999]. The event calculus explained. In *Artificial Intelligence Today*, pp. 409–430.

Shankar, N. and H. Ruess [2002]. Combining shostak theories. In *Proc. International Conference of Rewriting Techniques and Applications*, pp. 1–18.

Shapiro, S., Y. Lespérance, and H. Levesque [2002]. The cognitive agents specification language and verification environment for multiagent systems. In *Proc. AAMAS*, pp. 19–26.

Shirazi, A. and E. Amir [2005]. First-order logical filtering. In *Proc. IJCAI*, pp. 589–595.

Shoham, Y. [1993]. Agent-oriented programming. *Artificial intelligence 60*(1), 51–92.

Smullyan, R. [1995]. *First-order logic*. Dover Publications.

Son, T. and C. Baral [2001]. Formalizing sensing actions–a transition function based approach. *Artificial Intelligence 125*(1-2), 19–91.

Srivastava, S., N. Immerman, and S. Zilberstein [2010]. Computing applicability conditions for plans with loops. In *Proc. ICAPS*, pp. 161–168.

Steele, G. [1984]. *COMMON LISP: The language*. Digital Press (Burlington, MA).

Thielscher, M. [1999]. From situation calculus to fluent calculus: state update axioms as a solution to the inferential frame problem. *Artificial Intelligence 111*(1-2), 277–299.

Thielscher, M. [2001]. Planning with noisy actions (preliminary report). In *Proc. Australian Joint Conference on Artificial Intelligence*, pp. 27–45.

Thielscher, M. [2005]. Flux: A logic programming method for reasoning agents. *Theory and Practice of Logic Programming 5*(4-5), 533–565.

Thrun, S. [2002]. Robotic mapping: A survey. In G. Lakemeyer and B. Nebel (Eds.), *Exploring artificial intelligence in the new millennium*, pp. 1–35. Morgan Kaufmann.

Van Benthem, J., J. Gerbrandy, and B. Kooi [2009]. Dynamic update with probabilities. *Studia Logica 93*(1), 67–96.

Van Ditmarsch, H., A. Herzig, and T. De Lima [2007]. Optimal regression for reasoning about knowledge and actions. In *Proc. AAAI*, pp. 1070–1075.

Vardi, M. [1982]. The complexity of relational query languages (extended abstract). In *Proc. Annual ACM Symposium on Theory of Computing*, pp. 137–146.

Vardi, M. [1995]. On the complexity of bounded-variable queries. In *Proc. Principles of Database Systems*, pp. 266–276.

Vardi, M. Y. [1986]. Querying logical databases. *Journal of Computer and System Sciences 33*(2), 142–160.

Vassos, S., G. Lakemeyer, and H. Levesque [2008]. First-Order Strong Progression for Local-Effect Basic Action Theories. In *Proc. KR*, pp. 662–672.

Vassos, S. and H. Levesque [2007]. Progression of Situation Calculus Action Theories with Incomplete Information. In *Proc. IJCAI*, pp. 2024–2029.

Vassos, S. and H. Levesque [2008]. On the Progression of Situation Calculus Basic Action Theories: Resolving a 10-year-old Conjecture. In *Proc. AAAI*, pp. 1004–1009.

Vassos, S., S. Sardina, and H. Levesque [2009]. Progressing basic action theories with non-local effect actions. In *Proc. Commonsense*, pp. 135–140.

Von Wright, G. [1951]. *An essay in modal logic*. North-Holland.

Waaler, A. [2004]. Consistency proofs for systems of multi-agent only knowing. In R. A. Schmidt, I. Pratt-Hartmann, M. Reynolds, and H. Wansing (Eds.), *Advances in Modal Logic*, pp. 347–366. King's College Publications.

Waaler, A. and B. Solhaug [2005]. Semantics for multi-agent only knowing: extended abstract. In *Proc. TARK*, pp. 109–125.

Waldinger, R. [1977]. Achieving several goals simultaneously. In *Machine Intelligence*, Volume 8, pp. 94–136.

Zhang, Y. and Y. Zhou [2009]. Knowledge forgetting: Properties and applications. *Artificial Intelligence 173*(16-17), 1525–1537.

Curriculum Vitae

Last name: Belle
First name: Vaishak
Date of birth: 05.12.1983
Place of birth: Mangalore, India
Nationality: Indian
Marital status: Married

Qualifications

2008–2012: Doctoral studies in Computer Science, RWTH Aachen University
Thesis supervised by Prof. Gerhard Lakemeyer
Part of graduate school GK 643 and B-IT research school

2005–2008: M. Sc. in Computer Science, RWTH Aachen University
Thesis supervised by Prof. Gerhard Lakemeyer and Prof. Enrico Blanzieri
Part of European Master in Informatics double degree program

2005–2008: M. Sc. in Computer Science, University of Trento, Italy

2001–2005: B. E. in Computer Science, B. M. S. College of Engineering, India

1999–2001: Pre-university education, Christ College, Bangalore, India

1989–1999: Baldwin Boys High School, Bangalore, India