# Decidable Reasoning in a Fragment of the Epistemic Situation Calculus

**Gerhard Lakemeyer***
Dept. of Computer Science
RWTH Aachen University
52056 Aachen
Germany
gerhard@cs.rwth-aachen.de

**Hector J. Levesque**
Dept. of Computer Science
University of Toronto
Toronto, Ontario
Canada M5S 3A6
hector@cs.toronto.edu

### Abstract

The situation calculus is a popular formalism for reasoning about actions and change. Since the language is first-order, reasoning in the situation calculus is undecidable in general. An important question then is how to weaken reasoning in a principled way to guarantee decidability. Existing approaches either drastically limit the representation of the action theory or neglect important aspects such as sensing. In this paper we propose a model of limited belief for the epistemic situation calculus, which allows very expressive knowledge bases and handles both physical and sensing actions. The model builds on an existing approach to limited belief in the static case. We show that the resulting form of limited reasoning is sound with respect to the original epistemic situation calculus and eventually complete for a large class of formulas. Moreover, reasoning is decidable.

## Introduction

The situation calculus (McCarthy 1963; Reiter 2001) is a popular formalism for reasoning about action and change. Since the language is first-order, one of the advantages of the situation calculus is its high degree of expressiveness. However, this comes at a price: reasoning in the situation calculus is undecidable in general.

There are various ways of addressing this issue in a principled way. One would be to drastically limit the expressiveness, either by moving to a propositional action language such as the language $\mathcal{A}$ (Gelfond and Lifschitz 1993) or its descendants, or by integrating actions into description logics (Baader *et al.* 2005). Another would be to stay within the situation calculus but limit its representational or inferential capabilities (or both) without giving up its first-order nature. In (Gu and Soutchanski 2007), the authors use the two-variable fragment of first-order logic, which is known to be decidable, as the base logic of the situation calculus. In (De Giacomo *et al.* 2012; 2013), decidability is achieved by bounding the extension of fluent predicates by some constant. Going beyond two variables or boundedness requires further restrictions on the inference mechanism to retain decidability. The approach by Liu and Levesque (2005) restricts the representation of what holds in the current situation to literals. Such knowledge bases can then be queried efficiently using a sound but incomplete evaluation-based reasoner proposed in (Levesque 1998). When actions have only local effects, the result of an action can be represented by progressing the current knowledge base to a new set of literals. In (Claßen and Lakemeyer 2009), the restriction to literals is relaxed by allowing disjunctions in the representation of the initial situation. Queries about what holds after actions have been performed are evaluated by first regressing them to a query about the initial situation and then appealing to a logic of limited belief for static knowledge bases with disjunctions developed by Liu et al. (2004). Here a major limitation is that sensing actions cannot be dealt with.

The last two approaches have in common that they separate the reasoning task into two parts: one which addresses the dynamics of actions using progression and regression, respectively, and another which deals with querying a static knowledge base using existing techniques. A downside of this separation is that certain desirable features seem to get lost. For example, it is not at all clear how to handle sensing actions in the case of (Claßen and Lakemeyer 2009).

In order to avoid such limitations, we propose a new model of limited belief, which avoids the above separation by incorporating actions directly into the model. The starting point is the model of limited belief for the static case proposed in (Lakemeyer and Levesque 2013), which itself is based on (Liu et al. 2004). The general idea behind logics of limited belief is to capture, in a semantically perspicuous way, weaker forms of logical entailment. Other examples of such approaches are (Levesque 1984b; Konolige 1986; Vardi 1986; Fagin and Halpern 1988; Fagin *et al.* 1990; Lakemeyer 1996; Cadoli and Schaerf 1992; Delgrande 1995).

In the case of (Liu et al. 2004; Lakemeyer and Levesque 2013), the semantic primitive is a *setup*, a possibly infinite set of ground clauses closed under unit propagation. Roughly, the clauses in a setup can be viewed as those which the agent believes explicitly. Setups are used to give meaning to a sequence of modalities $\boldsymbol{B}_k$, for $k \geq 0$, where $\boldsymbol{B}_k\phi$ should be read as "$\phi$ is believed at level $k$." For example, given a clause $c$, $\boldsymbol{B}_0 c$ is satisfied by a setup $s$ just in case

*Gerhard Lakemeyer is also affiliated with the Dept. of Computer Science at the University of Toronto.

there is a clause in $s$ contained in $c$. In other words, at level 0, belief essentially reduces to retrieval wrt. $s$. At level 1, the agent believes everything that is believed at level 0 and, in addition, the agent is able to reason by cases allowing to split a single clause in $s$. At belief level 2, the number of possible case splits increases to 2, and so on.

In order to extend this framework to a dynamic setting, we borrow from a modal variant of the epistemic situation calculus called $\mathcal{ES}$ (Lakemeyer and Levesque 2011), which features formulas of the form $[n]\alpha$ meaning that $\alpha$ holds after action $n$. Setups are then allowed to not only mention literals $l$ but also extended literals of the form $[n_1]\cdots[n_m]l$. The new setups are thus able to talk about what is true initially and after any number of actions have been performed. As we will see, this allows us to model Reiter's basic action theories (Reiter 2001) as infinite setups. Limited belief will be defined in a way analogous to what we sketched in the previous paragraph.[1]

We then specify a reasoning service in terms of valid sentences of the form $\boldsymbol{B}_0 KB \supset [n_1]\cdots[n_m]\boldsymbol{B}_k\phi$, which may be read as "if *KB* is explicitly believed initially, then after actions $n_1$ to $n_m$ $\phi$ is believed at level $k$." Here $\phi$ itself may mention actions and *KB* represents a basic action theory in the sense of Reiter. We will show that this reasoning service is sound wrt. the situation calculus variant $\mathcal{ES}$. We also obtain an eventual completeness result for a large class of queries in the following sense: If $KB \supset \phi$ is valid in $\mathcal{ES}$, then there is a $k$ such that $\boldsymbol{B}_0 KB \supset \boldsymbol{B}_k\phi$ is valid in the new logic. Moreover, and perhaps most importantly, we obtain a decidability result for so-called bounded queries.

The rest of the paper is organized as follows. In the next section we briefly review the epistemic situation calculus $\mathcal{ES}$. We then introduce our new logic $\mathcal{ESL}$, which is like $\mathcal{ES}$ except that the classical belief operator $\boldsymbol{K}$ is replaced by its limited counterparts $\boldsymbol{B}_k$. We then define our reasoning service and show soundness and eventual completeness. After that we turn to decidability and then conclude.

## The Logic $\mathcal{ES}$

The language of $\mathcal{ES}$ is a modal first-order dialect with equality. It comes equipped with a countably infinite set of *standard names*, which can be thought of as special constants that satisfy the unique name assumption and an infinitary version of domain closure. In this sense, the standard names can be identified with the domain of discourse. As we will see, this greatly simplifies the semantic definitions below like interpreting quantification substitutionally (existing criticism of substitutional quantification notwithstanding (Kripke 1976)).

First-order variables and standard names come in two sorts, *action* (like *repair* and *bestAction*) and *object* (like *block5* and *location*). We let $\mathcal{N}$ denote the set of all standard names and $\mathcal{Z}$ denote the set of all sequences of standard names for actions, including $\langle\,\rangle$, the empty sequence.

A term is either a variable or a standard name. The *well-formed formulas* of the language form the least set such that

1. If $t_1, \ldots, t_k$ are terms, and $P$ is a $k$-ary predicate symbol then $P(t_1, \ldots, t_k)$ is an (atomic) formula;
2. If $t_1$ and $t_2$ are terms, then $(t_1 = t_2)$ is a formula;
3. If $t$ is an action term and $\alpha$ is a formula, then $[t]\alpha$ is a formula;
4. If $\alpha$ and $\beta$ are formulas, and $v$ is a first-order variable, then the following are also formulas: $(\alpha \vee \beta)$, $\neg\alpha$, $\exists v.\,\alpha$, $\Box\alpha$, $\boldsymbol{K}\alpha$.

We read $[n]\alpha$ as "$\alpha$ holds after action $n$", and $\Box\alpha$ as "$\alpha$ holds after any sequence of actions." We will freely use $\wedge, \supset, \equiv$, and $\forall$ as the usual abbreviations. There are two distinguished predicates *SF* and *Poss*: $Poss(n)$ says that action $n$ can be executed, and $SF(n)$ says that action $n$ returns a binary sensing result of 1 or 0.

Given a sequence of actions $z = \langle z_1, z_2, \ldots, z_n\rangle$ and a formula $\alpha$, we usually write $[z]\alpha$ instead of $[z_1][z_2]\cdots[z_n]\alpha$. When $z$ is the empty sequence, then $[z]\alpha = \alpha$. For action sequences $z$ and $z'$ we also write $z \cdot z'$ to mean the action sequence consisting of $z$ followed by $z'$. We write $z \sqsubseteq z'$ when $z$ is a prefix of $z'$. For a set of action sequences $Z$, we write $z \sqsubseteq Z$ to mean that there is some $z' \in Z$ such that $z \sqsubseteq z'$.

We call a formula without free variables a *sentence*. By a *primitive sentence* we mean a formula of the form $P(n_1, \ldots, n_m)$ where $P$ is a predicate symbol and all of the $n_i$ are standard names. For simplicity all predicates are considered to be fluent, that is, they may vary as a result of actions and their value may be unknown. It is convenient to distinguish between certain types of formulas using the following terminology:

- a formula with no $\Box$ operators is called *bounded*;
- a formula with no $\Box$ or $[t]$ operators is called *static;*
- a formula with no $\boldsymbol{K}$ operators is called *objective;*
- a formula with no fluent, $\Box$, or $[t]$ operators outside the scope of a $\boldsymbol{K}$ is called *subjective;*
- a formula with no $\boldsymbol{K}$, $\Box$, $[t]$, *Poss*, or *SF* is called a *fluent* formula;

Intuitively, to determine whether or not a sentence $\alpha$ is true after a sequence of actions $z$ has been performed, we need to specify two things: a world $w$ and an epistemic state $e$. We write $e, w, z \models \alpha$. A world determines truth values for the primitive sentences after any sequence of actions. An epistemic state is defined by a set of worlds, as in possible-world semantics. More precisely:

- a world $w \in W$ is any function from the primitive sentences and $\mathcal{Z}$ to $\{0, 1\}$.
- an epistemic state $e \subseteq W$ is any set of worlds.

To interpret what is known after a sequence of actions possibly including sensing has taken place, we define $w' \simeq_z w$ (read: $w'$ and $w$ agree on the sensing throughout action sequence $z$) inductively by the following:

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

[1] A preliminary version of our model was developed by Capes (2010).

Putting all these together, here is the semantic definition of truth. Given $e \subseteq W$ and $w \in W$, we define $e, w \models \alpha$ (read: $\alpha$ is true) as $e, w, \langle\,\rangle \models \alpha$, where for any $z \in \mathcal{Z}$:

1. $e, w, z \models P(n_1, \ldots, n_m)$ iff $w[P(n_1, \ldots, n_m), z] = 1$, where $P(n_1, \ldots, n_m)$ is a primitive sentence;
2. $e, w, z \models (n_1 = n_2)$ iff $n_1$ and $n_2$ are identical standard names;
3. $e, w, z \models (\alpha \vee \beta)$ iff $e, w, z \models \alpha$ or $e, w, z \models \beta$;
4. $e, w, z \models \neg\alpha$ iff $e, w, z \not\models \alpha$;
5. $e, w, z \models \exists v.\, \alpha$ iff $e, w, z \models \alpha^x_n$, for some standard name $n$ (of the same sort as $v$);
6. $e, w, z \models [n]\alpha$ iff $e, w, z \cdot n \models \alpha$;
7. $e, w, z \models \Box\alpha$ iff $e, w, z \cdot z' \models \alpha$, for every $z' \in \mathcal{Z}$;
8. $e, w, z \models \boldsymbol{K}\alpha$ iff $e, w', z \models \alpha$, for every $w' \in e$ such that $w' \simeq_z w$.

When $\alpha$ is *objective* (has no $\boldsymbol{K}$ operators), we can leave out the $e$ and write $w \models \alpha$. When $\Sigma$ is a set of sentences and $\alpha$ is a sentence, we write $\Sigma \models \alpha$ (read: $\Sigma$ logically entails $\alpha$) 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 $\models \alpha$ (read: $\alpha$ is valid) to mean $\{\} \models \alpha$.

We will not go into details of the logical properties of $\mathcal{ES}$ except to note that $\boldsymbol{K}$ behaves like an ordinary *K45* modal operator (Chellas 1980), after any number of actions.

## Basic Action Theories

While $\mathcal{ES}$ in general is less expressive than the classical situation calculus, we showed in (Lakemeyer and Levesque 2011) that it suffices to fully capture Reiter's *basic action theories (BATs)*.

**Definition 1** *Given a set of fluents $\mathcal{F}$, a set $\Sigma \subseteq \mathcal{ES}$ of sentences is called a basic action theory over $\mathcal{F}$ iff $\Sigma = \Sigma_0 \cup \Sigma_{\text{pre}} \cup \Sigma_{\text{post}} \cup \Sigma_{\text{sense}}$ where $\Sigma$ mentions only fluents in $\mathcal{F}$ and*

1. *$\Sigma_0$ is any set of fluent sentences;*
2. *$\Sigma_{\text{pre}}$ is a singleton sentence of the form $\Box Poss(a) \equiv \pi$, where $\pi$ is a fluent formula;*[2]
3. *$\Sigma_{\text{post}}$ is a set of sentences of the form $\Box[a]F(\vec{v}) \equiv \gamma_F$, one for each relational fluent $F$, where $\gamma_F$ is a fluent formula.*[3]
4. *$\Sigma_{\text{sense}}$ is a sentence (parallel to the one for Poss) of the form $\Box SF(a) \equiv \varphi$, where $\varphi$ is a fluent formula.*

$\Sigma_0$ is intended to express what is true initially; $\Sigma_{\text{pre}}$ characterizes the preconditions of actions in a single axiom; $\Sigma_{\text{post}}$ contains the *successor state axioms*, one per fluent, which incorporate Reiter's solution to the frame problem (Reiter 2001); following (Scherl and Levesque 2003), $\Sigma_{\text{sense}}$ expresses the sensing outcomes of every action. (In case an action does not return a meaningful sensing results, the value of *SF* can be defined to be vacuously true.)

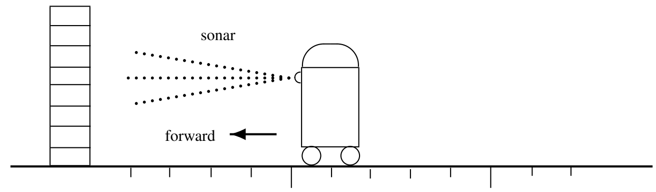


Figure 1: A simple robot

To illustrate BATs, let us consider the example depicted in Figure 1. Here a robot has two actions: *forward* moves it one unit forward towards the wall, and *sonar* is a sensing action, which returns 1 if the robot is close to the wall. Locations are denoted by propositional fluents $d_i$, which may be read as "the robot is located $i$ units from the wall."[4]

$$\begin{aligned}&\Box\, Poss(a) \;\equiv\\ &\quad a = \mathit{forward} \;\wedge\; \neg d_0 \quad\vee\\ &\quad a = \mathit{sonar} \;\wedge\; \text{TRUE}.\end{aligned}$$

In other words, *sonar* is always possible, and *forward* is executable only when the robot is not already at the wall. Next, we define the sensing results for the actions:

$$\begin{aligned}&\Box\, SF(a) \;\equiv\\ &\quad a = \mathit{forward} \;\wedge\; \text{TRUE} \quad\vee\\ &\quad a = \mathit{sonar} \;\wedge\; (d_0 \vee d_1).\end{aligned}$$

While *forward* always returns *true* as a sensing result, $SF(sonar)$ is 1 in case the robot is either at location $d_0$ or $d_1$. Finally, here are the successor state axioms for the $d_i$:

$$\begin{aligned}&\Box\, [a]d_i \;\equiv\\ &\quad a = \mathit{forward} \;\wedge\; d_{i+1} \quad\vee\\ &\quad a \neq \mathit{forward} \;\wedge\; d_i \qquad \text{for } i > 0.\\ &\Box\, [a]d_0 \;\equiv\\ &\quad a = \mathit{forward} \;\wedge\; d_1 \;\vee\; d_0\end{aligned}$$

In other words, after *forward*, $d_i$ becomes true if $d_{i+1}$ was true before, and false otherwise, except in the case of $d_0$, which remains true after *forward* was executed in $d_1$. After *sonar* the value of $d_i$ does not change.

For a given basic action theory $\Sigma$, projection queries, which are bounded sentences, are of particular importance, as they are a basic ingredient of action languages such as Golog (Levesque et al. 1997), among other things.

Let $\Sigma$ consist of the above axioms together with $(d_2 \vee d_3), \neg d_1$, and $\neg d_0$. Then it follows from $\boldsymbol{K}\Sigma$ that the robot believes that it is initially either 2 or 3 units away from the wall. Suppose the robot is actually 2 units away from the wall. Abbreviating *forward* as $f$ and *sonar* as $s$, respectively, let $S = SF(f) \wedge [f]SF(s)$ represent the actual sensing values obtained when executing $f$ and $s$ after $f$. We write *Close* instead of $(d_0 \vee d_1)$. Then we have:

[2]We assume that $\Box$ has lower syntactic precedence than the logical connectives, so that $\Box Poss(a) \equiv \pi$ stands for $\forall a.\Box(Poss(a) \equiv \pi)$.

[3]The $[t]$ construct has higher precedence than the logical connectives. So $\Box[a]F(\vec{x}) \equiv \gamma_F$ abbreviates $\forall a.\Box([a]F(\vec{x})) \equiv \gamma_F$.

[4]This way of handling distances is needed because we are not assuming any sort of built-in arithmetic in $\mathcal{ESL}$.

1. $\models \boldsymbol{K}\Sigma \supset \boldsymbol{K}\neg \mathit{Close}$;
2. $\not\models S \wedge \boldsymbol{K}\Sigma \supset [f]\boldsymbol{K}\mathit{Close}$;
3. $\models S \wedge \boldsymbol{K}\Sigma \supset [f][s]\boldsymbol{K}\mathit{Close}$;
4. $\models S \wedge \boldsymbol{K}\Sigma \supset \boldsymbol{K}[f][f]\mathit{Close}$;

In other words: initially the robot knows that it is not close to the wall (1); it does not know that it is close after executing a forward action (2); it knows that it is close after a forward and sensing the distance to the wall (3); it knows that it will be close if it were to perform two forward actions (4).

Since $\mathcal{ES}$ is a full first-order language, answering projection queries is clearly undecidable, even in the case without any actions at all. So let us now consider a variant of $\mathcal{ES}$ with a limited model of belief as the basis for deciding projection queries.

## The Logic $\mathcal{ESL}$

The language of $\mathcal{ESL}$ is that of $\mathcal{ES}$ except that the operator $\boldsymbol{K}$ is replaced by operators $\boldsymbol{B}_k$ for $k = 0, 1, 2, \ldots$ For the purposes of this paper, we also assume that the $\boldsymbol{B}_k$ do not occur nested. The definitions of bounded, static, objective, subjective, and fluent formulas carry over in the obvious way, with $\boldsymbol{B}_k$ replacing $\boldsymbol{K}$.

As extended forms of literals and clauses play an important role in the semantics, we introduce them here together with some additional terminology. An *extended literal* is a sequence of actions (possibly empty) followed by a literal, where a literal is either an atomic formula not mentioning $=$ or its negation. $\neg P(n)$ and $[m_1][m_2]P(n)$ are examples of extended literals. The complement of an extended literal $l$, denoted as $\bar{l}$, is obtained by complementing the literal it contains. For example, the complement of $[m_1][m_2]P(n)$ is $[m_1][m_2]\neg P(n)$. An *extended clause* is a disjunction of extended literals. Often an extended clause is identified with the set of extended literals it contains. In the following we will often refer to extended clauses simply as clauses.

The empty clause is denoted as $[]$. A unit clause is a clause with a single extended literal. A primitive clause is a clause whose atomic formulas are primitive sentences, and similar for primitive literals. When $z$ is a sequence of actions and $c$ a clause we write $[z]c$ to mean $c$ with every extended literal $l$ replaced by $[z]l$.

A *setup* is a (possibly infinite) set of primitive clauses. For any setup $s$, the closure of $s$ under unit propagation, which we denote as $\mathsf{UP}(s)$, is defined as the least set $s'$ which contains $s$, and if unit clause $l \in s'$ and $\{\bar{l}\} \cup c \in s'$, then $c \in s'$. $\mathsf{VP}(s)$ is defined as the set $\{c \mid c \text{ is a primitive clause and there exists a } c' \in \mathsf{UP}(s) \text{ such that } c' \subseteq c\}$.

Similar to $\mathcal{ES}$ the semantics of $\mathcal{ESL}$ is defined wrt. a world, an agent's epistemic state, and a sequence of actions. While worlds and action sequences are the same as in $\mathcal{ES}$, the epistemic state is now characterized by a setup. Intuitively, the clauses in a setup determine what the agent believes explicitly, or at level 0. Beliefs at higher levels are then obtained by reasoning by cases.

Given a world $w$, a setup $s$, and an action sequence $z$, the truth of a sentence $\alpha$, written as $s, w, z \models \alpha$, is inductively defined as follows:

1. $s, w, z \models P(n_1, \ldots, n_m)$ iff $w[P(n_1, \ldots, n_m), z] = 1$, where $P(n_1, \ldots, n_m)$ is a primitive formula;
2. $s, w, z \models (n_1 = n_2)$ iff $n_1$ and $n_2$ are identical;
3. $s, w, z \models (\alpha \vee \beta)$ iff $s, w, z \models \alpha$ or $s, w, z \models \beta$;
4. $s, w, z \models \neg\alpha$ iff $s, w, z \not\models \alpha$;
5. $s, w, z \models \exists x.\, \alpha$ iff $s, w, z \models \alpha^x_n$, for some standard name $n$ (of the same sort as $x$);
6. $s, w, z \models [n]\alpha$ iff $s \cup \{[z] \pm \mathit{SF}(n)\}, w, z \cdot n \models \alpha$;[5]
7. $s, w, z \models \Box\alpha$ iff $s, w, z \models [z']\alpha$, for all $z' \in \mathcal{Z}$;
8. $s, w, z \models \boldsymbol{B}_k\alpha$ iff $s, z, k \models \alpha$.

Note that Rules 1–5 are essentially identical to those of $\mathcal{ES}$. Rule 6 differs in that we explicitly remember the sensing result of the action in $s$. Roughly, this has the same effect as the compatibility relation $\simeq_z$ of $\mathcal{ES}$ when it comes to interpreting belief. Rule 7 is again essentially the same as in $\mathcal{ES}$. Let us now turn to the semantics of limited belief (Rule 8). Let $\alpha$ be objective. $s, z, k \models \alpha$ is the least relation such that

9. $s, z, k \models \alpha$ if $[] \in \mathsf{UP}(s)$;
10. $s, z, k \models c$ if $k = 0$, $c$ is a clause and $[z]c \in \mathsf{VP}(s)$;
11. $s, z, k \models \alpha$ if $k > 0$ and there is an extended primitive literal $l$ such that
$s \cup \{l\}, z, k-1 \models \alpha$ and
$s \cup \{\bar{l}\}, z, k-1 \models \alpha$;
12. $s, z, k \models (n = m)$ if $n, m$ are identical std. names;
$s, z, k \models \neg(n = m)$ if $n, m$ are distinct std. names;
13. $s, z, k \models \neg\neg\alpha$ if $s, z, k \models \alpha$;
14. $s, z, k \models (\alpha \vee \beta)$ if $s, z, k \models \alpha$ or $s, z, k \models \beta$;
$s, z, k \models \neg(\alpha \vee \beta)$ if $s, z, k \models \neg\alpha$ and $s, z, k \models \neg\beta$;
15. $s, z, k \models \exists x.\alpha$ if $s, z, k \models \alpha^x_n$ for some $n$;
$s, z, k \models \neg\exists x.\alpha$ if $s, z, k \models \neg\alpha^x_n$ for all $n$;
16. $s, z, k \models [n]\alpha$ if $s \cup \{[z]\mathit{SF}(n)\}, z \cdot n, k \models \alpha$ and $s \cup \{[z]\neg\mathit{SF}(n)\}, z \cdot n, k \models \alpha$;
$s, z, k \models \neg[n]\alpha$ if $s \cup \{[z]\mathit{SF}(n)\}, z \cdot n, k \models \neg\alpha$ and $s \cup \{[z]\neg\mathit{SF}(n)\}, z \cdot n, k \models \neg\alpha$;
17. $s, z, k \models \Box\alpha$ if $s, z, k \models [z']\alpha$ for all $z' \in \mathcal{Z}$;
$s, z, k \models \neg\Box\alpha$ if $s, z, k \models [z']\neg\alpha$ for some $z' \in \mathcal{Z}$.

Rules 9–15 are similar to those proposed for limited belief in the static case (Lakemeyer and Levesque 2013). Rule 9 says that whenever the empty clause can be derived by unit propagation everything is believed. Rule 10, which we will also refer to as the subsumption rule, says that a clause $c$ is believed at level 0 after $z$ just in case $[z]c$ is subsumed by a clause in $\mathsf{UP}(s)$. Rule 11, which we will also refer to as the split rule, says that $\alpha$ is believed at level $k > 0$ if the belief holds at level $k - 1$ after splitting on an extended literal.[6] Rule 12 deals with equality in a classical way, independent of any setup. In other words, we assume perfect reasoning

[5] Here $\pm\mathit{SF}(n)$ stands for $\mathit{SF}(n)$ if $w[\mathit{SF}(n), z] = 1$ and $\neg\mathit{SF}(n)$ if $w[\mathit{SF}(n), z] = 0$, respectively.

[6] In (Lakemeyer and Levesque 2013) we considered splitting a clause in $s$ instead. As we will see below, splitting an arbitrary literal has certain advantages.

when it comes to equality. Rules 13–15 make sure that certain "obvious" beliefs are obtained. For example, to believe $(\alpha \vee \beta)$ at level $k$ it suffices to believe either $\alpha$ or $\beta$. Rule 16 considers the case of believing that $\alpha$ (or its negation) holds after an action $n$. Note that in thinking about what will hold after action $n$, the agent has as yet no access to the sensing information that will result from performing this action. Therefore it needs to entertain both possibilities ($SF(n)$ and $\neg SF(n)$) when evaluating whether $\alpha$ is believed after $n$. Finally, $\Box$ is handled in the obvious way by considering all action sequences. Note that Rules 7 and 17 are well-founded when including the number of $\Box$ operators occurring in $\alpha$ in the measure for $\alpha$.[7]

When a formula $\alpha$ is objective, we sometimes write $w, z \models \alpha$ instead of $s, w, z \models \alpha$. For subjective $\alpha$, we write $s, z \models \alpha$. If, in addition, $z$ is empty, we simply write $w \models \alpha$ and $s \models \alpha$, respectively.

To conclude the semantic definition, we say that a set of sentences $\Sigma$ logically entails a sentence $\alpha$ (written $\Sigma \models \alpha$) if for every setup $s$ and world $w$, if $s, w, \langle\rangle \models \alpha'$ for every $\alpha' \in \Sigma$, then $s, w, \langle\rangle \models \alpha$. Finally, $\alpha$ is valid (written $\models \alpha$) if $\{\} \models \alpha$. (So far we have used $\models$ for entailment in both $\mathcal{ES}$ and $\mathcal{ESL}$. In the following, we will write $\models_{\mathcal{ES}}$ when referring to entailment in $\mathcal{ES}$.)

## Some Properties

Here we will not go into a detailed discussion of the properties of $\mathcal{ESL}$ except to note the following. The logic of limited belief presented in (Lakemeyer and Levesque 2013) is very similar to $\mathcal{ESL}$ when restricted to formulas without nested beliefs. This is because, in the case of static formulas, the semantics of both logics is the same except for the split rule. The same applies to a slight variant of the logic presented in (Liu et al. 2004), and the properties discussed there for static formulas generally carry over in a straightforward way to the dynamic case. For example, a simple induction proof establishes that for any $k$

$\models \Box(\boldsymbol{B}_k \neg\neg\alpha \equiv \boldsymbol{B}_k \alpha)$
$\models \Box(\boldsymbol{B}_k(\alpha \wedge \beta) \equiv \boldsymbol{B}_k \alpha \wedge \boldsymbol{B}_k \beta)$
$\models \Box(\boldsymbol{B}_k \forall x.\alpha \equiv \forall x.\boldsymbol{B}_k \alpha)$

While most of the equivalence preserving transformations of formulas such as removing double negations are preserved at all belief levels, there are exceptions. For example,

$$\boldsymbol{B}_k(p \wedge (q \vee r)) \supset \boldsymbol{B}_k((p \wedge q) \vee (p \wedge r))$$

is not valid. See (Liu et al. 2004) for a detailed discussion of why this is so.

Intuitively, one would expect belief to be monotonic in the sense that the beliefs held at a certain setup and level of belief continue to hold when moving to larger setups or increasing the level of belief. We now show that this is indeed the case.

**Theorem 1** *Let $s$ and $s'$ be setups such that $s \subseteq s'$. Then for any $z$, $k$, and objective $\phi$, if $s, z, k \models \phi$ then $s', z, k \models \phi$.*

[7]We remark that there are equivalent versions of Rule 7 and 17, where the sequence $z'$ is only mentioned on the left-hand side of the rules, but at the expense of a more complicated definition.

**Proof:** The proof is by induction on $k$ and the structure of $\phi$. Let $k = 0$. The only interesting case is when $\phi$ is a clause $c$. It is easy to show that $\mathsf{VP}(s) \subseteq \mathsf{VP}(s')$ given that $s \subseteq s'$. Hence, if $[z]c \in \mathsf{VP}(s)$ then $[z]c \in \mathsf{VP}(s')$. Note also that, in the case of actions (Rule 16+17), the same *SF*-formulas are added to $s$ and $s'$ so that the subset relation between the augmented setups continues to hold.

Suppose the lemma holds for some $k$ and let $s, z, k+1 \models \phi$. Again, the proof is by induction on the structure of $\phi$. Here the only interesting case is the split rule. So suppose, there is some extended literal $l$ such that $s \cup \{l\}, z, k \models \phi$ and $s \cup \{\bar{l}\}, z, k \models \phi$. In both cases we can apply the induction hypothesis for $k$ and obtain $s', z, k+1 \models \phi$. ∎

**Corollary 1** *If $\mathsf{VP}(s) \subseteq \mathsf{VP}(s')$ then $s, z, k \models \phi$ implies $s', z, k \models \phi$.*

**Theorem 2** *If $s, z, k \models \phi$ then $s, z, k+1 \models \phi$.*

**Proof:** Let $s, z, k \models \phi$ and let $l$ be any extended literal. Then, by Theorem 1, $s \cup \{l\}, z, k \models \phi$ and $s \cup \{\bar{l}\}, z, k \models \phi$. Hence, by the split rule, $s, z, k+1 \models \phi$. ∎

As an immediate corollary we obtain

**Corollary 2** $\models \Box(\boldsymbol{B}_k \phi \supset \boldsymbol{B}_{k+1} \phi)$.

## Eventual completeness

When considering sentences $e$ that only mention $=$, limited belief behaves exactly like $\boldsymbol{K}$, even at level 0. A simple induction shows that $\models_{\mathcal{ES}} \Box(\boldsymbol{K}e \equiv e)$ and $\models \Box(\boldsymbol{B}_0 e \equiv e)$, that is, $e$ is believed just in case $e$ is true, after any number of actions. In general, of course, $\boldsymbol{B}_k$ is much weaker than $\boldsymbol{K}$, especially when $k$ is small. For example, while $\boldsymbol{K}(p \vee \neg p)$ is valid, $\boldsymbol{B}_0(p \vee \neg p)$ is not. For example, if $s$ is the empty setup, then $s, \langle\rangle \not\models \boldsymbol{B}_0(p \vee \neg p)$ because neither Rule 10 nor Rule 14 applies. On the other hand, $\boldsymbol{B}_1(p \vee \neg p)$ is valid, since we can apply the split rule using $p$. This is not accidental. We will now show that for any objective valid bounded propositional formula $\phi$ of $\mathcal{ESL}$ we can always find a $k$ such $\boldsymbol{B}_k \phi$ is valid. By a propositional formula we mean a formula which does not contain quantifiers and which does not mention $=$.

We begin with a useful lemma that shows the connection between worlds and their corresponding setups for arbitrary objective formulas.

**Definition 2** *Given a world $w$ let*

$$s(w) = \{l \mid l \textit{ is an extended literal and } w \models l\}.$$

In other words, $s(w)$ is the setup consisting of all extended literals true at $w$. Note that $[z]l \in s(w)$ iff $w, z \models l$ for any literal $l$.

**Lemma 1** *For any objective $\phi$ and sequence of actions $z$, $s(w), z, k \models \phi$ iff $w, z \models \phi$ for all $k$.*

**Proof:** We first prove the lemma for $k = 0$ by induction on the structure of $\phi$. For a clause $c$ we have $s(w), z, 0 \models c$ iff $[z]l \in s(w)$ for some $l \in c$ (by the subsumption rule) iff $w, z \models l$ for some $l \in c$ (by the construction of $s(w)$) iff $w, z \models c$. The lemma clearly holds for $(n = m)$ and $\neg(n =$

$m$) and also (by induction) for $\neg\neg\phi$, $(\phi \wedge \psi)$, $\neg(\phi \wedge \psi)$, $\forall x.\phi$, and $\neg\forall x.\phi$.

$s(w), z, 0 \models [n]\phi$ iff $s(w) \cup \{[z]\mathit{SF}(n)\}, z \cdot n, 0 \models \phi$ and $s(w) \cup \{[z]\neg\mathit{SF}(n)\}, z \cdot n, 0 \models \phi$ iff $s(w), z \cdot n, 0 \models \phi$ (because $s(w)$ already contains either $[z]\mathit{SF}(n)$ or $[z]\neg\mathit{SF}(n)$ and hence either does not change by adding the extended literal or it becomes inconsistent so that Rule 9 applies) iff $w, z \cdot n \models \phi$ (by induction) iff $w, z \models [n]\phi$. (The case for $\neg[n]\phi$ is similar.) $s(w), z, 0 \models \Box\phi$ iff $s(w) \cup \sigma, z, 0 \models [z']\phi$ for all $z'$ iff $s(w), z \cdot z', 0 \models \phi$ for all $z'$ (since, similar to the case of a single action, adding all combinations of sensing results for $z'$ to $s(w)$ leads either to inconsistency or leaves $s(w)$ as is) iff $w, z \cdot z' \models \phi$ for all $z'$ iff $w, z \models \Box\phi$.

We now show that $s(w), z, k \models \phi$ iff $s(w), z, k+1 \models \phi$, from which the lemma follows. The only-if direction follows from Lemma 2. For the if direction, suppose $s(w), z, k{+}1 \models \phi$. The proof is by induction on the structure of $\phi$. For a clause $c$ and since $k + 1 > 0$, the only applicable rule is the split rule. Hence there is an extended literal $l$ such that $s(w) \cup \{l\}, z, k \models \phi$ and $s(w) \cup \{\bar{l}\}, z, k \models \phi$. Since $s(w)$ is a complete set of literals, either $l$ or $\bar{l}$ is in $s(w)$. Suppose $l \in s(w)$. Then $s(w) \cup \{l\} = s(w)$ and, hence, $s(w), z, k \models \phi$. The other case is symmetric. The case of $=$-expression is obvious. For the induction step, if any of the rules (12)–(17) apply, then the induction hypothesis applies immediately. In case of the split rule the argument is the same as above. ∎

For the rest of this subsection, we only consider propositional bounded formulas. The following definition characterizes the positive extended literals contained in a formula, or PEL($\phi$) for short.

**Definition 3** *Let $\phi$ be bounded and objective. Then PEL($\phi$) is defined inductively as follows:*

*PEL($p$) $= \{p\}$ when $p$ is a primitive sentence;*
*PEL($\neg\phi$) $=$ PEL($\phi$);*
*PEL($\phi \vee \psi$) $=$ PEL($\phi$) $\cup$ PEL($\psi$);*
*PEL($[n]\phi$) $= \{[n \cdot z]p \mid [z]p \in$ PEL($\phi$)$\}$.*

For example, if $\phi = [n](p \vee [m](\neg q \vee [l]r))$ then PEL($\phi$) $= \{[n]p, [n{\cdot}m]q, [n{\cdot}m{\cdot}l]r\}$. Note that PEL ignores negations. Let PEL$(\phi)_z = \{[z \cdot z']p \mid [z']p \in \text{PEL}(\phi)\}$ for any action sequence $z$.

**Definition 4 (Worlds compatible with a setup)** *Let $s$ be a setup. Then*

$$C(s) = \{w \mid w \text{ is a world and } w \models c \text{ for all } c \in s\}$$

**Lemma 2** *Let $A = \{p_1, \ldots, p_n\}$ be a set of positive extended literals and $L = \{l_1, \ldots, l_n\}$ with $l_i = p_i$ or $l_i = \overline{p}_i$. Let $s$ be a setup such that $L \subseteq s$. Then for all sequences of actions $z$, all $\phi$ with PEL$_z(\phi) \subseteq A$, and all $w \in C(s)$, $w, z \models \phi$ iff $s, z, 0 \models \phi$.*

**Proof:** First note that in case $s$ is inconsistent the lemma holds vacuously, since there are no compatible worlds in this case. Now suppose $s$ is consistent. Then $s$ does not contain $\bar{l}$ for any $l \in L$ as $s$ already contains $l$.

The proof is by induction on the structure of $\phi$. For a clause $c$ suppose $w, z \models c$. Then $w, z \models l$ for some extended literal $l \in c$. By assumption, either $[z]l \in A$ or $[z]\bar{l} \in A$. Since $w$ is compatible with $s$ and, in particular, with $L$, $[z]l \in L$ and hence $[z]l \in s$ by assumption. Therefore, by the subsumption rule, $s, z, 0 \models c$.

Conversely, let $s, z, 0 \models c$. Then $[z]c \in \mathsf{VP}(s)$ by subsumption. Since $L \subseteq s$ and since $s$ is consistent, there must be some extended literal $l \in c$ such that $[z]l \in s$ and $[z]l \in L$. Thus $w, z \models l$ follows and, hence, $w, z \models c$.

The cases for $\neg\neg\phi$, $(\phi \vee \psi)$, and $\neg(\phi \vee \psi)$ follow trivially by induction. Let $w, z \models [n]\phi$. Then $w, z \cdot n \models \phi$ and, by induction, $s, z \cdot n, 0 \models \phi$. (Note that PEL$_z([n]\phi) =$ PEL$_{z \cdot n}(\phi)$. By monotonicity, $s \cup \{[z]\mathit{SF}(n)\}, z \cdot n, 0 \models \phi$ and $s \cup \{[z]\neg\mathit{SF}(n)\}, z \cdot n, 0 \models \phi$. Hence, $s, z, 0 \models [n]\phi$. Conversely, let $s, z, 0 \models [n]\phi$. Then $s \cup \{[z]\mathit{SF}(n)\}, z \cdot n, 0 \models \phi$ and $s \cup \{[z]\neg\mathit{SF}(n)\}, z \cdot n, 0 \models \phi$. Suppose $w, z \models \mathit{SF}(n)$ (the other case is symmetric). Then $w \in C(s \cup \{[z]\mathit{SF}(n)\})$. Hence, by induction, $w, z \cdot n \models \phi$ and, therefore $w, z \models [n]\phi$. ∎

We are now ready to prove our eventual-completeness result. The interesting part is that any valid propositional, bounded $\phi$ is believed at some level $k$. Roughly, this is because we can always split on every extended literal contained in $\phi$ so that the resulting setups are essentially complete valuations of all the extended literals mentioned.

**Theorem 3** *For any propositional, bounded, objective $\phi$, $\models \phi$ iff $\models \boldsymbol{B}_k\phi$ for some $k$.*

**Proof:** For the if direction, suppose $\models \boldsymbol{B}_k\phi$ for some $k$ and consider any world $w$. Then $s(w), \langle\rangle, k \models \phi$ and, by Lemma 1, $w, \langle\rangle \models \phi$.

Conversely, let $\models \phi$. Let $n$ be the size of PEL($\phi$), and for any setup $s$, let $\nu(s)$ be the number of elements $p$ of PEL($\phi$) such that either $p$ or $\overline{p}$ appears in $s$. We will first show that for any $s$ and $k = n - \nu(s)$, $s \models \boldsymbol{B}_k\phi$. The proof is by induction on $k$.

For $k = 0$ we need to show that $s, \langle\rangle, 0 \models \phi$. If $[] \in \mathsf{UP}(s)$ then clearly $s, \langle\rangle, 0 \models \phi$ because of Rule 9. Otherwise, since $\nu(s) = n$, we have that for every atom $p$ in $\phi$, exactly one of $p$ or $\overline{p}$ is in $s$. Let $L = \{l_1, \ldots, l_n\}$ be the set of all such literals. Let $w \in C(L)$. Since $w \models \phi$ by assumption, $L \models \boldsymbol{B}_0\phi$ holds by Lemma 2. Since $L \subseteq s$, by Theorem 1, $s \models \boldsymbol{B}_0\phi$.

Suppose for all $s$ with $k = n - \nu(s)$ and $0 \leq k < n$, $s \models \boldsymbol{B}_k\phi$. Now consider a setup $s'$ with $n - \nu(s') = k + 1$. Let $p$ be an atom occurring in $\phi$ such that neither $p$ nor $\overline{p}$ is in $s'$. Such $p$ must exist since $n \neq \nu(s')$ by assumption. Since $n - \nu(s' \cup \{p\}) = n - \nu(s' \cup \{\overline{p}\}) = k$, by induction, $s' \cup \{p\} \models \boldsymbol{B}_k\phi$ and $s' \cup \{\neg p\} \models \boldsymbol{B}_k\phi$. Since $\phi$ is objective, it follows that $s' \cup \{p\}, \langle\rangle, k \models \phi$ and $s' \cup \{\neg p\}, \langle\rangle, k \models \phi$. Hence, using the split rule, we obtain, $s', \langle\rangle, k+1 \models \phi$, that is, $s' \models \boldsymbol{B}_{k+1}\phi$.

To complete the proof of the only-if direction, let $s$ be any setup and let $k = n$. By the above argument, $s \models \boldsymbol{B}_{k'}\phi$ for $k' = n - \nu(s)$. Since $0 \leq k' \leq n$, $s \models \boldsymbol{B}_k\phi$ (using $\models \boldsymbol{B}_{k'}\phi \supset \boldsymbol{B}_{k'+1}\phi$ if needed). ∎

We remark that this significantly strengthens an earlier result by Liu et al. (2004), who showed that eventual completeness in the static case obtains only when querying knowledge bases and only when the query belongs to a restricted class of formulas in a normal form defined in (Levesque 1998). The reason for our stronger result is our modified split rule. While Liu et al. restricted splitting to clauses in the setup, we allow splitting arbitrary literals, which is strictly more general. In the next section we will revisit the issue of eventual completeness also in the context of querying knowledge bases.

## A Reasoning Service Based on $\mathcal{ESL}$

We now turn to the use of the logic as a specification of a reasoning service for an agent with limited beliefs. In particular, we phrase it in terms of the valid sentences of the following form:

$$\boldsymbol{B}_0\phi \supset [z]\boldsymbol{B}_k\psi, \tag{1}$$

where $\phi$ typically encodes a basic action theory of the kind introduced earlier, and $\psi$ is a bounded formula, that is, it may contain actions but not $\Box$. We will refer to sentences (1) as *belief implications*, and they can be read as "if $\phi$ is explicitly believed initially then after actions $z$ have occurred $\psi$ is believed at level $k$." Belief implications specify a reasoning service in the sense that determining beliefs at level $k$ after some actions requires computing whether the corresponding belief implication is valid.

We begin our investigations by showing that belief implications are sound with respect to the original epistemic situation calculus $\mathcal{ES}$.

**Theorem 4** *If* $\models \boldsymbol{B}_0\alpha \supset [z]\boldsymbol{B}_k\beta$ *then* $\models_{\mathcal{ES}} \boldsymbol{K}\alpha \supset [z]\boldsymbol{K}\beta$.

**Proof:** Here we prove it for the case of a single action $n$. Generalizing it to an arbitrary $z$ is straightforward. Suppose $\models \boldsymbol{B}_0\alpha \supset [n]\boldsymbol{B}_k\beta$ and let $e, w \models \boldsymbol{K}\alpha$. Then $w' \models \alpha$ for all $w' \in e$. Then $s(w'), \langle\,\rangle, 0 \models \alpha$ by Lemma 1. Let $w' \in e$ and suppose $w' \simeq_n w$. Thus $s(w'), w \models \boldsymbol{B}_0\alpha$. By assumption, $s(w') \cup \{\pm SF(n)\}, w, \langle n\rangle \models \boldsymbol{B}_k\beta$ and, therefore, $s(w') \cup \{\pm SF(n)\}, \langle n\rangle, k \models \beta$. Since $w' \simeq_z w$ we have $s(w') \cup \{\pm SF(n)\} = s(w')$ and hence, by Lemma 1, $w', \langle n\rangle \models \beta$. As this holds for all $w' \in e$ such that $w' \simeq_n w$, $e, w \models [n]\boldsymbol{K}\beta$ follows. ∎

Liu et al. (2004) considered so-called proper$^+$ knowledge bases, for which they showed that reasoning under limited belief is decidable in the static case. We will now extend this idea to knowledge bases representing basic action theories. In the following we let $e$ range over *ewffs*, which are quantifier-free fluent formulas that mention only $=$ and no other predicate symbols. Let $\theta$ be a mapping from variables to standard names (of the appropriate sort). Then we write $e\theta$ to mean $e$ with each variable substituted by the standard name according to $\theta$. For a formula $\alpha$ we write $\forall\alpha$ to denote its universal closure.

**Definition 5 (proper$^+$ Knowledge Bases)** *Let $e$ be an ewff and $c$ an extended clause. Then we call a formula of the form $\forall(e \supset c)$ a $\forall$-clause and a formula of the form $\Box\forall(e \supset c)$ a $\Box$-clause. A KB is called proper$^+$ if it is a finite set of $\forall$- and $\Box$-clauses.*

*Given a proper$^+$ KB, we let gnd(KB) be the infinite setup which is the union of* $\{c\theta \mid \forall(e \supset c) \in \mathit{KB} \text{ and } \models e\theta\}$ *and* $\{[z]c\theta \mid \Box\forall(e \supset c) \in \mathit{KB},\ \models e\theta,\ \text{and } z \in \mathcal{Z}\}$.

In the following we will not consider proper$^+$ *KB*s in their full generality but only those which are the translation of a basic action theory as defined earlier. We call such *KB*s *proper$^+$ BATs*. It will also be necessary to consider proper$^+$ BATs augmented with a finite set of extended literals. We call such *KB*s augmented proper$^+$ BATs.

Note that a proper$^+$ BAT has two kinds of $\Box$-formulas: Those $\Box\forall(e \supset c)$ which are the result of the translation of a successor state axiom are such that $c$ mentions a single extended literal of the form $[a]l$, where $l$ is a literal and $a$ is an action variable. All other $\Box\forall(e \supset c)$ like the ones resulting from the translation of the *SF*-axiom are such that $c$ mentions only literals and no actions, that is, $c$ is static.

As an example, let us consider the BAT for our robot example. It is easy to translate the original axioms into a proper$^+$ KB. Here we consider only the translations of the sensing axiom and the successor state axiom for $d_i$ with $i > 0$. The *SF*-axiom can be represented by the following five $\Box$-clauses:

$\Box\forall(a \neq \mathit{forward} \supset (\neg SF(a) \vee d_0 \vee d_1))$;
$\Box\forall(a \neq \mathit{forward} \wedge a \neq \mathit{sonar} \supset \neg SF(a))$;
$\Box\forall(a = \mathit{forward} \supset SF(a)$;
$\Box\forall(a = \mathit{sonar} \supset (\neg d_0 \vee SF(a))$;
$\Box\forall(a = \mathit{sonar} \supset (\neg d_1 \vee SF(a))$.

The successor state axiom for $d_i$ with $i > 0$ is captured by these $\Box$-clauses:

$\Box\forall(a = \mathit{forward} \supset (\neg d_{i+1} \vee [a]d_i))$;
$\Box\forall(a \neq \mathit{forward} \supset (\neg d_i \vee [a]d_i))$;
$\Box\forall(a \neq \mathit{forward} \supset ([a]\neg d_i \vee d_i))$;
$\Box\forall(a = \mathit{forward} \supset ([a]\neg d_i \vee d_{i+1}))$;

When considering belief implications, the following result is very useful as it reduces belief implications for proper$^+$ *KB*s to truth in the setup gnd(*KB*).

**Theorem 5** $\models \boldsymbol{B}_0\mathit{KB} \supset \boldsymbol{B}_k\phi$ *iff gnd(KB)* $\models \boldsymbol{B}_k\phi$.

**Proof:** For the only-if direction, let $\models \boldsymbol{B}_0\mathit{KB} \supset \boldsymbol{B}_k\phi$. It suffices to show that gnd(*KB*) $\models \boldsymbol{B}_0\mathit{KB}$. It is easy to see that this is the case. For consider $\Box\forall(e \supset c) \in \mathit{KB}$. We need to show that gnd(*KB*)$, z, 0 \models (\neg e\theta \vee c\theta)$ for any $z$ and substitution $\theta$. Note that gnd(*KB*)$, \langle\,\rangle, 0 \models [z]e\theta$ iff $\models e\theta$. If $\not\models e\theta$, then the claim holds by Rule 14. Otherwise, the subsumption rule applies. (The case of a $\forall$-clause is proved the same way except that $z = \langle\,\rangle$.)

For the if direction, suppose that gnd(*KB*) $\models \boldsymbol{B}_k\phi$. Let $s \models \boldsymbol{B}_0\mathit{KB}$. Then $\mathsf{VP}(\text{gnd}(\mathit{KB})) \subseteq \mathsf{VP}(s)$. By Corollary 1, $s \models \boldsymbol{B}_k\phi$. ∎

**Corollary 3**
$\models \boldsymbol{B}_0\mathit{KB} \supset [n]\boldsymbol{B}_k\phi$ *iff gnd(KB)* $\cup \{SF(n)\}, n, k \models \phi$ *and gnd(KB)* $\cup \{\neg SF(n)\}, n, k \models \phi$.

Let *KB* be the proper$^+$ BAT of our robot example, including clauses $(d_2 \vee d_3), \neg d_1$, and $\neg d_0$ for the initial situation. As before, suppose the robot is 2 units away from the wall and let $S = SF(f) \wedge [f]SF(s)$. Then

1. $\models \boldsymbol{B}_0 KB \supset \boldsymbol{B}_0(\neg Close)$;
2. $\not\models S \wedge \boldsymbol{B}_0 KB \supset [f]\boldsymbol{B}_0(d_1 \vee d_2)$;
3. $\models S \wedge \boldsymbol{B}_0 KB \supset [f]\boldsymbol{B}_1(d_1 \vee d_2)$;
4. $\models S \wedge \boldsymbol{B}_0 KB \supset [f][s]\boldsymbol{B}_1 Close$;

The robot believes that it is not close initially at level 0 by subsumption, using the clauses $\neg d_0$ and $\neg d_1$ from gnd($KB$); after *forward*, it believes that it is at $d_1$ or $d_2$ only at level 1, because it needs to split a literal, for example, $d_2$, and then use unit propagation on the $(\neg d_i \vee [a]d_i)$ for $i = 2$ and $i = 3$; After a *forward* and then sensing the distance to the wall, we can show that the robot believes at level 1 that it is close using the following argument: since the *KB* is augmented by $[f]SF(s)$, $[f](d_0 \vee d_1)$ obtains by unit propagation from $[f](\neg SF(s) \vee d_0 \vee d_1)) \in \text{gnd}(KB)$. Then splitting $[f]d_0$ and using $[f](\neg d_i \vee [s]d_i)$ the result obtains.

## Decidability

We will now turn to the problem of deciding belief implications for proper$^+$ BATs and strictly bounded queries, which are bounded objective formulas not mentioning any action variables. In other words, all actions mentioned in such queries are standard names, as in the examples above.

We will mostly consider belief implications of the form $\boldsymbol{B}_0 KB \supset \boldsymbol{B}_k\phi$. Once we have established decidability for those, the extension to $\boldsymbol{B}_0 KB \supset [z]\boldsymbol{B}_k\phi$-formulas will be straightforward.

The following definition shows how to extract the action sequences mentioned in such queries in a way very similar to PEL (Definition 3).

**Definition 6** *Let $\phi$ be strictly bounded and objective. Then $AS(\phi)$ is defined inductively as follows:*

*$AS(p) = \{\langle\rangle\}$ when $p$ is an atomic formula;*
*$AS((t = t')) = \{\langle\rangle\}$;*
*$AS(\neg\phi) = AS(\phi)$;*
*$AS(\phi \vee \psi) = AS(\phi) \cup AS(\psi)$;*
*$AS(\exists x.\phi) = AS(\phi)$;*
*$AS([n]\phi) = \{n \cdot z \mid z \in AS(\phi)\}$.*

*Let $AS_z(\phi) = \{z \cdot z' \mid z' \in AS(\phi)\}$, where $z$ is any action sequence.*

The next result shows that, when considering a strictly bounded query $\phi$, we only need to consider those instances of $\Box$-formulas in the *KB* which mention action sequences that are subsequences of those in AS($\phi$). We begin by defining such instantiated *KB*s:

**Definition 7** *Let KB be proper$^+$ and $Z$ a set of action sequences. The $Z$-instantiation $KB{\downarrow_Z}$ is then defined as the union of the following sets:*

- $\{[z]\forall(e \supset c) \mid z \sqsubseteq Z, \Box\forall(e \supset c) \in KB, \text{ and } c \text{ static}\}$
- $\{[z]\forall(e \supset c^a_n) \mid z{\cdot}n \sqsubseteq Z, \Box\forall(e \supset c) \in KB, \ c \text{ not static}\}$
- $\{\forall(e \supset c) \mid \forall(e \supset c) \in KB\}$

**Theorem 6** *Let KB be a proper$^+$ BAT, $\phi$ a strictly bounded objective formula, and $Z = AS(\phi)$. Then $\models \boldsymbol{B}_0 KB \supset \boldsymbol{B}_k\phi$ iff $\models \boldsymbol{B}_0 KB{\downarrow_Z} \supset \boldsymbol{B}_k\phi$.*

The proof is not hard. It involves showing that instances of $\Box$-clauses that are not subsequences of action sequences in AS($\phi$) are irrelevant in establishing beliefs about formulas whose action sequences are included in AS($\phi$). We remark that the analogous theorem holds in $\mathcal{ES}$ as well, that is, $\models \boldsymbol{K}KB \supset \boldsymbol{K}\phi$ iff $\models \boldsymbol{K}KB{\downarrow_Z} \supset \boldsymbol{K}\phi$. It is crucial, though, that *KB* is a proper$^+$ BAT, that is, the $\Box$-clauses in *KB* correspond to the dynamic part of a basic action theories. Otherwise, it is easy to construct a counterexample: for instance, there is a basic action theory *KB* such that $\models \boldsymbol{B}_0 KB \supset \boldsymbol{B}_0[n_1]F$ and $\models \boldsymbol{B}_0 KB \supset \boldsymbol{B}_0[n_1][n_2]F$, yet $\models \boldsymbol{B}_0 KB \supset \boldsymbol{B}_0[n_1][n_2][n_3]\neg F$. Then adding $\Box F$ to *KB* would lead to an inconsistency by unit propagation at level 0, that is, $[] \in \mathsf{UP}(\text{gnd}(KB \cup \{\Box F\}))$, and we obtain that $\boldsymbol{B}_0(KB \cup \{\Box F\}) \supset \boldsymbol{B}_0 p$ is valid for any atomic proposition $p$. The problem is that we would need to instantiate $KB \cup \{\Box F\}$ using all subsequences of $n_1 \cdot n_2 \cdot n_3$ to discover this, yet $\text{AS}(p) = \{\}$.

Given Theorem 6, we can restrict our attention to AS($\phi$)-instantiated proper$^+$*KB*s for any given query $\phi$. In order to arrive at a decision procedure for belief implications we need two more things: restrict the set of literals to choose from when splitting and, most importantly, restrict the possible substitutions of standard names to consider to a finite set. To achieve the former we let

$$\text{PEL}(KB, \phi, Z) = \{[z]p \mid p \text{ is an atom in } KB \text{ or } \phi, z \sqsubseteq Z\}.$$

The idea is that only literals that mention atoms from the *KB* or the query together with their respective action sequences need to be considered. To limit substitutions of standard names, we follow (Liu et al. 2004) and define a sufficiently large but finite set of standard names for a given *KB* and query $\phi$:

For any $m \geq 0$ let $H^+_m$ be the set of standard names mentioned in *KB* and $\phi$ plus $m$ new standard names of type object. Let $n$ be the maximum number of (object) variables appearing in a $\forall$-clause appearing in *KB*. Then $\text{gnd}(KB)|_{H^+_n}$ denotes gnd($KB$) with substitutions $\theta$ restricted to names in $H^+_n$. Similarly, $\text{PEL}(KB, \phi, Z)|_{H^+_n}$ denotes the set of extended literals in $\text{PEL}(KB, \phi, Z)$ with all variables replaced by names in $H^+_n$.

With these abbreviations in hand, we now define a computable function $V$, which, as will be shown below, returns 1 just in case $\text{gnd}(KB), z, k \models \phi$, that is, $V$ decides belief implications.

**Definition 8** *Let $\phi$ be a strictly bounded objective sentence, $z$ an action sequence, KB an $AS(\phi)_z$-instantiated augmented BAT, and $k \geq 0$.*

$$V[KB, \phi, z, k] = \begin{cases} 1 & \textit{if any of the following} \\ & \textit{conditions (1)–(12) holds} \\ 0 & \textit{otherwise} \end{cases}$$

1. *$[] \in \mathsf{UP}(gnd(KB)|H^+_n)$;*
2. *$k = 0$, $\phi$ is a clause $c$, and there exists a $c' \in \mathsf{UP}(gnd(KB)|H^+_n)$ such that $c' \subseteq [z]c$;*
3. *$k > 0$ and for some extended literal $l \in PEL(KB, \phi, AS(\phi)_z)|_{H^+_n}$, $V[KB \cup \{l\}, \phi, z, k-1] = 1$ and $V[KB \cup \{\bar{l}\}, \phi, z, k-1] = 1$*

4. $\phi = (n = m)$, *and* $n, m$ *are the same standard names;*
5. $\phi = \neg(n = m)$, *and* $n, m$ *are distinct standard names;*
6. $\phi = \neg\neg\psi$, *and* $V[KB, \psi, z, k] = 1$*;*
7. $\phi = (\psi \vee \chi)$, $V[KB, \psi, z, k] = 1$ *or* $V[KB, \chi, z, k] = 1$*;*
8. $\phi = \neg(\psi \vee \chi)$, $V[KB, \neg\psi, z, k] = 1$ *and* $V[KB, \neg\chi, z, k] = 1$*;*
9. $\phi = \exists x.\psi$, *and* $V[KB, \psi^x_n, z, k] = 1$ *for some* $n \in H^+_1$*;*
10. $\phi = \neg\exists x.\psi$, *and* $V[KB, \neg\psi^x_n, z, k] = 1$ *for all* $n \in H^+_1$*;*
11. $\phi = [n]\psi$, $V[KB \cup \{[z]SF(n)\}, \psi, z \cdot n, k] = 1$ *and* $V[KB \cup \{[z]\neg SF(n)\}, \psi, z \cdot n, k] = 1$*;*
12. $\phi = \neg[n]\psi$, $V[KB \cup \{[z]SF(n)\}, \neg\psi, z \cdot n, k] = 1$ *and* $V[KB \cup \{[z]\neg SF(n)\}, \neg\psi, z \cdot n, k] = 1$.

We will now show that the above restrictions, that is, considering only a finite number of substitution instances of the *KB* and the query, and only finitely many split literals, are indeed sufficient to correctly compute belief implications. As in (Liu et al. 2004), the main idea is that names not mentioned in the *KB* or the query are, in a sense, indistinguishable and hence interchangeable. The following definition of a bijection over names makes this precise:

Let $^*$ be a bijection over the set of names (preserving sorts). For any formula $\alpha$, let $\alpha^*$ be $\alpha$ with every name $n$ replaced by $n^*$. We extend this notion to sets of formulas, including setups, and substitutions $\theta$ in the obvious way. The following properties of $^*$ are easy to prove:

**Lemma 3**

- $\models [z]e$ *iff* $\models [z^*]e^*$ *iff* $\models e^*$*;*
- $c \in \mathsf{UP}(s)$ *iff* $c^* \in \mathsf{UP}(s^*)$*;*
- $s, z, k \models \phi$ *iff* $s^*, z^*, k \models \phi^*$*;*
- *gnd*$(KB)^* =$ *gnd*$(KB^*)$.

**Lemma 4** *Let* $m_1, \ldots, m_n$ *be the names in* $H^+_n$ *that do not occur in KB and* $\phi$. *Let* $c \in \mathsf{UP}(gnd(KB))$ *contain names* $m'_1, \ldots, m'_l$ $(l \leq n)$ *not mentioned in* $H^+_n$. *Let* $^*$ *be the bijection which swaps* $m_i$ *and* $m'_i$ *for all* $1 \leq i \leq l$ *and is the identity otherwise. Then* $c^* \in \mathsf{UP}(gnd(KB)|_{H^+_n})$.

The proof essentially follows the same argument as Lemma 4 in (Liu et al. 2004). Note that any $c \in \mathsf{UP}(\text{gnd}(KB))$ mentions at most $n$ names not in $H^+_n$. Also note that the lemma covers the special case that, if $[] \in \mathsf{UP}(\text{gnd}(KB))$ then $[] \in \mathsf{UP}(\text{gnd}(KB)|_{H^+_n})$.

**Lemma 5** *Let* $\phi$ *be a strictly bounded objective formula with a single object variable* $x$. *Let* $n, m$ *be object names not occurring in KB or* $\phi$.
*Then gnd*$(KB), z, k \models \phi^x_n$ *iff gnd*$(KB), z, k \models \phi^x_m$.

The lemma is easily proved using Lemma 3. Note that this justifies that restricting substitutions of names for quantified variables (Rule 9+10) to elements of $H^+_1$ is correct.

**Lemma 6**
*Suppose that gnd*$(KB), z, k \models \phi$ *by splitting an extended primitive literal* $l$. *Then gnd*$(KB), z, k \models \phi$ *by splitting an extended primitive literal* $l$ *in PEL*$(KB, \phi, AS(\phi)_z)|_{H^+_n}$.

**Proof:** Clearly, only elements of $\text{PEL}(KB, \phi, \text{AS}(\phi)_z)$ are relevant when choosing a literal for splitting. The fact that only substitution instances from $H^+_n$ are needed can again be shown using Lemma 3. ∎

Finally, we are able to show that the beliefs of a proper knowledge base at any level can be decided by $V$:

**Theorem 7** *gnd*$(KB), z, k \models \phi$ *iff* $V[KB, \phi, z, k] = 1$.

**Proof:** Given Lemma 4–6, the proof is a simple induction on $k$ and the structure of $\phi$. ∎

Given Theorem 5, we immediately obtain:

**Corollary 4** *The validity problem for sentences of the form* $\boldsymbol{B}_0 KB \supset \boldsymbol{B}_k\phi$, *where KB is a proper*$^+$ *BAT,* $\phi$ *a strictly bounded objective sentence, and* $k \geq 0$, *is decidable.*

The result can easily be extended to the case of beliefs after a sequence of actions. For simplicity, consider a single action $n$. Then $\boldsymbol{B}_0 KB \supset [n]\boldsymbol{B}_k\phi$ iff $\text{gnd}(KB) \cup \{SF(n)\}, n, k \models \phi$ and $\text{gnd}(KB) \cup \{\neg SF(n)\}, n, k \models \phi$ (Corollary 3). In both cases, we obtain decidability because Theorem 7 applies.

**Corollary 5** *The validity problem for sentences of the form* $\boldsymbol{B}_0 KB \supset [z]\boldsymbol{B}_k\phi$ *is decidable.*

Given the result on eventual completeness in the propositional case (Theorem 3), it is not hard to obtain a similar result for belief implications when the query is propositional.

**Theorem 8** *Let KB be a proper*$^+$ *BAT and* $\phi$ *a propositional strictly bounded objective sentence. Then* $\models KB \supset \phi$ *iff* $\models \boldsymbol{B}_0 KB \supset \boldsymbol{B}_k\phi$ *for some* $k$.

**Proof:** (Sketch) Note that $\models \boldsymbol{B}_0 KB \supset \boldsymbol{B}_k\phi$ iff $\text{gnd}(KB{\downarrow_Z})|_{H^+_n} \models \boldsymbol{B}_k\phi$ for $Z = \text{AS}(\phi)$ (by Theorem 6,7). Moreover, $\text{gnd}(KB{\downarrow_Z})|_{H^+_n}$ is finite. By an argument very similar to the proof of Theorem 3 one can show that, provided $\models KB \supset \phi$, $\text{gnd}(KB{\downarrow_Z})|_{H^+_n} \models \boldsymbol{B}_k\phi$ holds where $k$ is the size of $\text{PEL}(\text{gnd}(KB{\downarrow_Z})|_{H^+_n}, \phi, Z)$. ∎

We can even generalize the theorem to universal queries of the form $\forall\vec{x}.\phi$ where $\phi$ mentions no quantifiers. Here we only consider the case of a single universal quantifier:

**Corollary 6**
$\models KB \supset \forall x.\phi$ *iff* $\models \boldsymbol{B}_0 KB \supset \boldsymbol{B}_k\forall x.\phi$ *for some* $k$.

**Proof:** $\models \boldsymbol{B}_0 KB \supset \boldsymbol{B}_k\forall x\phi$ iff $\models \boldsymbol{B}_0 KB \supset \forall x.\boldsymbol{B}_k\phi$ iff $\models \boldsymbol{B}_0 KB \supset \boldsymbol{B}_k\phi^x_n$ for all $n$ mentioned in *KB* and $\phi$ plus one new name (using Lemma 5). Now apply the theorem for each such $n$ and choose $k$ to be the largest among all of them. ∎

## Conclusions

In this paper we extended ideas of limited belief, which were first explored by Liu et al. (2004) for static knowledge bases, to the dynamic case. By modifying the split rule considered by Liu et al. we were able to substantially extend results on eventual completeness. Moreover, for proper$^+$ knowledge bases which encode Reiter-style basic action theories and strictly bounded queries we showed decidability.

In (Lakemeyer and Levesque 2013), we considered decidable reasoning with nested beliefs and unknown individuals, but only in the static case. We believe that these results carry over to the dynamic setting without much difficulty.

Furthermore, it is not hard to generalize our results to bounded queries that may contain action variables. For example, we could then ask whether $\exists a_1 \exists a_2 [a_1][a_2]\phi$ is believed at some level $k$. Hence, with bounded queries we are able to express a form of bounded planning problems. Allowing $\Box$ within a query is much more problematic, as $\Box\phi$ expresses a state constraint about $\phi$ holding after any sequence of actions. Handling $\Box$ as part of a query is problematic for reasons similar to those that make it essential that proper$^+$ *KB*s are restricted to those representing basic action theories. In fact, we conjecture that answering queries involving $\Box$ is undecidable even for limited belief.

Finally, a possible extension we would like to consider in the future is to allow for action functions like *pickup*$(x)$. These are clearly desirable when constructing a basic action theory for a given domain. The main question will be how to achieve this without losing decidability.

## References


F. Baader, C. Lutz, M. Milicic, U. Sattler, and F. Wolter, Integrating Description Logics and Action Formalisms: First Results. *The Twentieth National Conference on Artificial Intelligence (AAAI)*, 572–577, 2005.

M. Cadoli and M. Schaerf, Approximate Reasoning and Non-Omniscient Agents. In *Proc. of the 4th Conference on Theoretical Aspects of Reasoning about Knowledge (TARK-92)*, pages 159–183, 1992.

T. Capes, *A Logic for the Progression of Knowledge Bases as a Consequence of Unit Propagation* M.Sc. Thesis, University of Toronto, 2010.

B. Chellas, *Modal Logic: An Introduction*. Cambridge University Press, Cambridge, 1980.

J. Claßen and and G. Lakemeyer, Tractable First-Order Golog with Disjunctive Knowledge Bases, Proceedings of the Ninth International Symposium on Logical Formalizations of Commonsense Reasoning (Commonsense 2009), Toronto, Canada, UTSePress, pp. 27–33, 2009.

G. De Giacomo, Y. Lesprance, and F. Patrizi. Bounded Situation Calculus Action Theories and Decidable Verification, KR 2012.

G. De Giacomo, Y. Lesprance, and F. Patrizi. Bounded Epistemic Situation Calculus Theories. IJCAI 2013.

J.P. Delgrande, A framework for logics of explicit belief. *Computational Intelligence*, 11(1):47–88, 1995.

R. Fagin and J.Y. Halpern, Belief, awareness, and limited reasoning. *Artificial Intelligence*, 34(1):39–76, 1988.

R. Fagin, J.Y. Halpern, and M.Y. Vardi, A nonstandard approach to the logical omniscience problem. In *Proc. of the 3rd Conference on Theoretical Aspects of Reasoning about Knowledge (TARK-90)*, pages 41–55, 1990.

M. Gelfond and V. Lifschitz, Representing Actions and Change by Logic Programs. *Journal of Logic Programming,* 17(2–4), 301–323, 1993.

Y. Gu and M. Soutchanski, Decidable Reasoning in a Modified Situation Calculus. *In Proc. of the 20th International Joint Conference on Artificial Intelligence (IJCAI),* 1891–1897, 2007.

K. Konolige, *A Deduction Model of Belief*. Brown University Press, 1986.

S. Kripke, Is there a problem with substitutional quantification? In G. Evans and J. McDowell, editors, *Truth and Meaning*, pages 325–419. Clarendon Press, Oxford, 1976.

G. Lakemeyer, *Limited Reasoning in First-Order Knowledge Bases with Full Introspection Artif. Intell.* 84(1-2): 209–255, 1996.

G. Lakemeyer, H.J. Levesque, A Semantic Characterization of a Useful Fragment of the Situation Calculus with Knowledge. *Artificial Intelligence,* 175, pages 142–164, 2011.

G. Lakemeyer, H.J. Levesque, Decidable Reasoning in a Logic of Limited Belief with Introspection and Unknown Individuals. *23rd International Joint Conference on Artificial Intelligence (IJCAI),* 2013.

H.J. Levesque, A logic of implicit and explicit belief. In *Proc. of AAAI-84*, 198–202, 1984.

H.J. Levesque, A completeness result for reasoning with incomplete first-order knowledge bases. In *Proc. KR-98*, pages 14–23, 1998.

H. J. Levesque, R. Reiter, Y. Lespérance, F. Lin, and R. B. Scherl, Golog: A logic programming language for dynamic domains. *Journal of Logic Programming,* **31**, 59–84, 1997.

Y. Liu, G. Lakemeyer, H.J. Levesque, A Logic of Limited Belief for Reasoning with Disjunctive Information. *Int. Conf. on the Principles of Knowledge Representation and Reasoning (KR),* 587–597, 2004.

Y. Liu and H.J. Levesque, Tractable reasoning with incomplete first-order knowledge in dynamic systems with context-dependent actions. In Proc. of the Nineteenth International Joint Conference on Artificial Intelligence (IJCAI-05),522–527, 2005.

J. McCarthy, *Situations, Actions and Causal Laws.* Technical Report, Stanford University, 1963. Also in M. Minsky (ed.), *Semantic Information Processing*, MIT Press, Cambridge, MA, 410–417, 1968.

R. Reiter, *Knowledge in Action: Logical Foundations for Describing and Implementing Dynamical Systems*. MIT Press, 2001.

R. B. Scherl and H. J. Levesque, Knowledge, action, and the frame problem. *Artificial Intelligence* 144(1-2), 1-39, 2003.

M.Y. Vardi, On epistemic logic and logical omniscience. In *Proc. of the 1st Conference on Theoretical Aspects of Reasoning about Knowledge (TARK-86)*, pages 293–305, 1986.