

From Situation Calculus to Dynamic Epistemic Logic

Hans van Ditmarsch

University of Sevilla, Dept. Logic
Calle Camilo Jose Cela s/n, 41018 Sevilla, Spain
hvd@us.es
<http://personal.us.es/hvd>

Andreas Herzig

Université de Toulouse, CNRS, Institut de recherche en informatique de Toulouse
118 route de Narbonne, F-31062 Toulouse Cedex 9, France
herzig@irit.fr
<http://www.irit.fr/~Andreas.Herzig>

Tiago de Lima

CRIL, Université d'Artois
Rue Jean Souvras, SP 18, F-62307 Lens Cedex, France
delima@cril.fr
<http://www.cril.fr/~delima>

February 1, 2010

Abstract

We start from Reiter's solution to the frame problem in terms of successor state axioms and Scherl&Levesque's extension to knowledge, as formulated by Lakemeyer&Levesque in their logic ES. While it was believed up to now that quantification over actions is a characteristic feature of Reiter's solution, we here show that for a reasonably large subset of Reiter's basic action theories one can do without. We do so by recasting restricted basic action theories in a propositional modal logic, viz. dynamic epistemic logic with public announcements and public assignments.

Keywords: Reasoning about actions, situation calculus, frame problem, successor state axioms, knowledge representation and reasoning, dynamic epistemic logic.

1 Introduction

Situation Calculus (SitCalc) and Dynamic Epistemic Logics (DELs) have the same aim: to provide a logical modeling of actions and events in terms of their effects on the world and on the agents' knowledge. But when we take a closer look at their object of study some differences appear.

- Historically, SitCalc research focussed on what might be called ontic actions: actions changing the facts of the world, such as stacking and painting blocks. Dealing with such physical actions in logic puts the representational frame problem: the problem of avoiding to explicitly state non-effects of actions. Currently, most approaches use Reiter's solution to this problem in terms of quantification over actions and so-called successor state axioms (SSAs).

Explicit modeling of the agent's knowledge and ignorance and of non-ontic actions (actions that change an agent's knowledge, alias sensing-actions) was studied recently by Scherl and Levesque [1993, 2003] and Lakemeyer and Levesque [2004, 2005]. However, these extensions only consider a single agent.

- In contrast, DELs considered multiple agents and almost exclusively concentrated on epistemic actions of learning truth of propositions, focussing on epistemic phenomena such as the so-called Moore sentences [Moore, 1942] or knowability [Balbiani et al., 2008]. However, recently assignments of atomic propositions have been integrated with such DELs [van Ditmarsch et al., 2005; van Benthem et al., 2006; Kooi, 2007].

SitCalc and DELs use quite different formal tools:

- SitCalc requires quantification over action variables in order to formulate successor state axioms; worse, quantification over predicate names is necessary to formulate background hypotheses such as the tree-like structure of the set of possible situations. However, Reiter proved decidability of a fragment of SitCalc (viz. for so-called propositional fluents, basic action theories and regressable formulas, see Section 2 of the present paper).
- DELs are propositional modal logics with modal operators of knowledge and of events. They have interesting mathematical properties; in particular they are often decidable, and their complexity is known [Lutz, 2006; Balbiani et al., 2010].

These differences explain that up to now research on SitCalc and DELs was advancing independently, and did not benefit from results, methods and tools that were developed next door. There were only a few papers calling for convergence between SitCalc and modal logics. Blackburn et al. [2001] established a link between SitCalc and hybrid modal logics. Demolombe [2003] provided a translation from SitCalc into dynamic logic, however with explicit frame axioms. The papers by van Ditmarsch et al. [2007a] and van Benthem [2007] were the first to point out close similarities between SitCalc and DELs.

Our aim is to change this situation of mutual ignorance (or should we say splendid isolation?). We are going to show that the propositional fragment of SitCalc (with only so-called propositional fluents) can be embedded into a particular DEL. Precisely, we translate the entailment problem of what is known to be a 'useful fragment of the situation calculus' [Lakemeyer and Levesque, 2005], viz. the logic ES, but without the 'only knowing' operator. The target logic is public announcement logic (PAL) with assignments [van Ditmarsch et al., 2005; van Benthem et al., 2006; Kooi, 2007], that we extend by test actions; we call the resulting logic PALAT.

Our embedding is surprising given that in the language of PALAT Reiter's main device is not available: PALAT does not allow quantification over actions. We show that nevertheless one can do without it, and prove that our embedding is polynomial. To that end we have to slightly restrict basic action theories to what we call basic action theories with explicit successor state axioms. It follows that decision procedures for DELs can be exploited in order to decide SitCalc entailment problems.

Our paper is both more specific and more general than our previous [van Ditmarsch et al., 2007a]. We here elaborate further on the formal link between SitCalc and DELs that we only sketched there: we consider more general action theories, and we establish a precise formal relationship. Complexity issues (that were also investigated in [van Ditmarsch et al., 2007a]) have been left aside.

The remainder of the paper is organized as follows: Section 2 recalls Lakemeyer&Levesque’s logic ES and defines basic action theories with explicit SSAs. Section 3 introduces public announcement logic with assignments and tests PALAT. Section 4 presents a polynomial translation of ES’s entailment problem for basic action theories with explicit SSAs, into the PALAT validity problem. In Section 5 we sketch one of the benefits of our translation, viz. how an agent’s incomplete perception of events can be accounted for in ES, using ideas from [Baltag et al., 1998; Baltag and Moss, 2004; Aucher and Herzig, 2007, 2010]. Section 6 concludes.

2 Background: Lakemeyer&Levesque’s logic ES

The most popular solution to the frame problem is Reiter’s in terms of successor state axioms (SSAs) and regression [Reiter, 1991]. It received a number of extensions such as for probabilistic actions [Bacchus et al., 1999] and knowledge [Scherl and Levesque, 1993, 2003]. We here consider the latter epistemic extension.

Reiter formulated SSAs in SitCalc, which is the mainstream formalism in reasoning about actions. However, because SitCalc is defined axiomatically, properties about action theories that are not direct entailments are very hard to prove. For instance, [Lakemeyer and Levesque, 2004, 2005] mention the long proof in [Reiter, 2001b] for the fact that if $\mathbf{K}\varphi$ entails $\mathbf{K}\psi_1 \vee \mathbf{K}\psi_2$ in a theory Σ , then $\mathbf{K}\varphi$ entails $\mathbf{K}\psi_1$ in Σ , or $\mathbf{K}\varphi$ entails $\mathbf{K}\psi_2$ in Σ . Aiming at a “more workable” semantics for the SitCalc they proposed a variant they called ES. This logic is not as expressive as the entire situation calculus, but it allows for basic action theories à la Reiter and, mutatis mutandis, also Reiter’s solution to the frame problem.

In this section we first give the syntax and semantics of propositional ES, then show how basic action theories with explicit SSAs are defined, and finally recall how regression works.

2.1 Syntax of ES

The full language of ES has modal operators of knowledge and action together with quantification over both actions and objects, and is thus a many-sorted modal language. Dynamic epistemic logics are essentially propositional, and our goal is to embed ES into them; we here therefore disregard quantification over object variables and focus on quantification over action variables, which are Reiter’s technical tool in order to solve the frame problem.

Let U be a nonempty countable set of action variables, let P be a countable set of fluents of arity 0, let A be a countable set of action constants, and let $Poss$ (‘possible’) and SF (‘sensed fluent’) be two predicate symbols of arity 1. The language \mathcal{L}_{ES} is the set of formulas φ defined by the following BNF:

$$\varphi ::= p \mid Poss(t) \mid SF(t) \mid t = t \mid \neg\varphi \mid \varphi \wedge \varphi \mid \mathbf{K}\varphi \mid [t]\varphi \mid \Box\varphi \mid \forall x\varphi$$

where p ranges over P , t ranges over $A \cup U$, and x over U .

The predicate $Poss$ models executability preconditions of actions. The intuition is that $Poss(a)$ is used to abbreviate a formula which holds if and only if the action a is executable. The predicate SF is used to model the result of sensing actions. The formula $SF(a)$ abbreviates a formula whose truth value is known by the agent after the execution of a . As in epistemic logics, the operator \mathbf{K} models the agent’s knowledge. The operator $[\cdot]$ is used to model actions. A formula of the form $[a]\varphi$ is read ‘ φ holds after the execution of a ’. The formula $\Box\varphi$ is read ‘ φ holds after the execution of any sequence of actions’.¹

¹The original language of ES also contains an ‘only knowing’ operator \mathbf{OK} that we do not consider here.

We define $P(\varphi)$ to be the set of those elements of P occurring in φ . We similarly define the sets of actions $A(\varphi)$ and the set of variables $U(\varphi)$. For example when φ is $(x=a_1 \wedge p) \vee \forall y(y=a_2 \wedge q)$ then we have $P(\varphi) = \{p, q\}$, $A(\varphi) = \{a_1, a_2\}$ and $U(\varphi) = \{x, y\}$.

We use the common abbreviations for the operators \vee , \rightarrow and \leftrightarrow . The symbols \perp and \top respectively abbreviate $p \wedge \neg p$ and $\neg(p \wedge \neg p)$, for some $p \in P$.

DEFINITION 1 The following are subsets of \mathcal{L}_{ES} .

- *Atomic formulas* are elements of P and formulas of the form $\text{Poss}(t)$ and $SF(t)$, for $t \in A \cup U$.
- *Ground formulas* are formulas wherein no action variable occurs.
- *Ground box-free formulas* are ground formulas without the operator \Box . The set of ground box-free formulas is noted $\mathcal{L}_{\text{ES}}^0$.
- *Fluent formulas* are formulas without Poss , SF , \mathbf{K} , $[a]$, $[x]$, \Box .
- *Boolean formulas* are formulas without Poss , SF , $=$, \mathbf{K} , $[a]$, $[x]$, \Box , \forall .

For example, $\forall x(x = a)$ is a fluent formula, but not a Boolean formula; $\text{Poss}(x)$ is atomic but not ground; $\mathbf{K}SF(a)$ is ground but not Boolean.

Every Boolean formula is a fluent formula. The set of ground atomic formulas of ES consists of the fluents from P , the set of $\text{Poss}(a)$ such that $a \in A$, and the set of $SF(a)$ such that $a \in A$.

2.2 Semantics of ES

Let A^* be the set of all sequences of actions from A , where ϵ denotes the empty sequence and $\alpha \cdot \alpha'$ is the concatenation of α and α' . Let Ω_{ES} be the set of all mappings from ground atomic formulas and action sequences to $\{0, 1\}$, i.e. Ω_{ES} is the set of functions ω such that

$$\omega : ((P \cup \{\text{Poss}(a) \mid a \in A\} \cup \{SF(a) \mid a \in A\}) \times A^*) \longrightarrow \{0, 1\}$$

Formulas in \mathcal{L}_{ES} are evaluated in triples of the form $\langle e, \omega, \alpha \rangle$ such that: $e \subseteq \Omega_{\text{ES}}$ is the agent's epistemic state, $\omega \in e$ is the actual state, and $\alpha \in A^*$ is the sequence of actions that has been performed. (Note that the possible worlds ω that are used here are functions, and are therefore different from the Kripkean possible worlds of modal logic that we are going to introduce in Section 3.)

To interpret what is known by the agent after a sequence of actions, Lakemeyer&Levesque inductively define a relation of *indistinguishability of two worlds with respect to a sequence of actions*:

- $\omega \sim_{\epsilon} \omega'$ for all $\omega, \omega' \in \Omega_{\text{ES}}$, and
- $\omega \sim_{\alpha \cdot a} \omega'$ iff $\omega \sim_{\alpha} \omega'$ and $\omega(SF(a), \alpha) = \omega'(SF(a), \alpha)$.

That is, ω and ω' are indistinguishable after action a if they were so before, and if a 's sensed fluent has the same value at ω and ω' before a .

The *satisfaction relation* \models is between triples and sentences, alias formulas without free variables. It

is defined inductively by:

$\langle e, \omega, \alpha \rangle \models \varphi$	iff	$\omega(\varphi, \alpha) = 1$ if φ is a ground atomic formula
$\langle e, \omega, \alpha \rangle \models a_1 = a_2$	iff	a_1 and a_2 are (syntactically) identical
$\langle e, \omega, \alpha \rangle \models \neg \varphi$	iff	$\langle e, \omega, \alpha \rangle \not\models \varphi$
$\langle e, \omega, \alpha \rangle \models \varphi \wedge \psi$	iff	$\langle e, \omega, \alpha \rangle \models \varphi$ and $\langle e, \omega, \alpha \rangle \models \psi$
$\langle e, \omega, \alpha \rangle \models \mathbf{K}\varphi$	iff	for all $\omega' \in e$, if $\omega \sim_\alpha \omega'$ then $\langle e, \omega', \alpha \rangle \models \varphi$
$\langle e, \omega, \alpha \rangle \models [a]\varphi$	iff	$\langle e, \omega, \alpha \cdot a \rangle \models \varphi$
$\langle e, \omega, \alpha \rangle \models \Box \varphi$	iff	for all $\alpha' \in A^*$, $\langle e, \omega, \alpha \cdot \alpha' \rangle \models \varphi$
$\langle e, \omega, \alpha \rangle \models \forall x \varphi$	iff	for all $a \in A$, $\langle e, \omega, \alpha \rangle \models \varphi[x \backslash a]$

where $\varphi[x \backslash a]$ is the formula resulting from replacing all free occurrences of variable x in φ by a . Remember that the set of ground atomic formulas is $P \cup \{Poss(a) \mid a \in A\} \cup \{SF(a) \mid a \in A\}$. We write $\langle e, \omega \rangle \models \varphi$ instead of $\langle e, \omega, \epsilon \rangle \models \varphi$.

A formula $\varphi \in \mathcal{L}_{ES}$ is a *valid ES consequence* of a set of formulas $\Psi \subseteq \mathcal{L}_{ES}$, noted $\Psi \models_{ES} \varphi$, if and only if for all e and ω , if $\langle e, \omega \rangle \models \psi$ for all $\psi \in \Psi$ then $\langle e, \omega \rangle \models \varphi$. A formula φ is *ES valid*, noted $\models_{ES} \varphi$, if and only if $\emptyset \models_{ES} \varphi$. For example $\models_{ES} [a]\varphi \leftrightarrow \neg[a]\neg\varphi$. Note that this equivalence is not valid in dynamic logic.

When $U(\varphi) = \emptyset$, i.e. when no variables occur in φ , then all equalities $a_1 = a_2$ can be replaced by either \top or \perp , depending on whether a_1 and a_2 are syntactically identical or not. Moreover \top and \perp can also be eliminated, except when φ is ES equivalent to \top or \perp . We note $\varphi \downarrow$ the result of eliminating these equalities between constants and occurrences of \top or \perp from a variable-free φ , and say that $\varphi \downarrow$ is the *normal form* of φ . For example, $([a](a_1 = a_1 \wedge p) \vee a_1 = a_2) \downarrow = ([a](\top \wedge p) \vee \perp) \downarrow = ([a]p \vee \perp) \downarrow = [a]p$.

PROPOSITION 2 If $U(\varphi) = \emptyset$ then $\varphi \downarrow$ is a Boolean formula, and $\models_{ES} \varphi \leftrightarrow \varphi \downarrow$.

In [Lakemeyer and Levesque, 2004, Theorem 4] it is shown that that the following *successor state axiom for knowledge* (SSAK) is valid:

$$\text{SSAK.} \quad \models_{ES} \forall x \Box ([x]\mathbf{K}\varphi \leftrightarrow ((SF(x) \wedge \mathbf{K}(SF(x) \rightarrow [x]\varphi)) \vee (\neg SF(x) \wedge \mathbf{K}(\neg SF(x) \rightarrow [x]\varphi))))$$

It also shown (Theorem 2) that the positive introspection principle $\Box(\mathbf{K}\varphi \rightarrow \mathbf{K}\mathbf{K}\varphi)$ and the negative introspection principle $\Box(\neg \mathbf{K}\varphi \rightarrow \mathbf{K}\neg \mathbf{K}\varphi)$ are both ES valid.

REMARK 3 The original papers by Lakemeyer&Levesque do not require that ω be in e . We added this constraint in order to be in accordance with the standard logic of knowledge S5, which stipulates that the formula schema $\mathbf{K}\varphi \rightarrow \varphi$ should be valid. This would not be the case if we allowed for $\omega \notin e$. However, our approach also applies to the original version of ES, see Remark 10 and the discussion in Section 6.

2.3 Explicit successor state axioms

DEFINITION 4 A *successor state axiom* (SSA) for a fluent p is a formula of the form

$$\forall x \mathbf{K}\Box ([x]p \leftrightarrow \gamma_p(x))$$

where $\gamma_p(x)$ is a fluent formula.

We are going to focus on SSAs where the only action variable of $\gamma_p(x)$ is x and where for every a not occurring in $\gamma_p(x)$, the instance $\gamma_p(a)$ is propositionally equivalent to p modulo normal forming.

DEFINITION 5 A successor state axiom $\forall x \mathbf{K}\Box ([x]p \leftrightarrow \gamma_p(x))$ is *explicit* iff $U(\gamma_p(x)) = \{x\}$, and for every $a \in A$, if $a \notin A(\gamma_p(x))$ then $\gamma_p(a) \downarrow \leftrightarrow p$ is propositionally valid.

Examples of explicit SSAs are Reiter’s original SSAs [Reiter, 1991], where $\gamma_p(x)$ takes the form

$$(x=a_1 \wedge \gamma_p^+(a_1)) \vee \cdots \vee (x=a_n \wedge \gamma_p^+(a_n)) \vee (p \wedge \neg(x=a'_1 \wedge \gamma_p^-(a'_1)) \wedge \cdots \wedge \neg(x=a'_m \wedge \gamma_p^-(a'_m)))$$

such that $\gamma_p^+(a_i)$ and $\gamma_p^-(a'_j)$ are Boolean formulas. The formula $\gamma_p^+(a_i)$ characterizes the condition under which a_i makes p true, and the formula $\gamma_p^-(a'_j)$ characterizes the condition under which a'_j makes p false. It is easy to see that for such right hand sides $\gamma_p(x)$, if a is different from all a_i , $1 \leq i \leq n$ and all a'_j , $1 \leq j \leq m$, then $\gamma_p(a) \downarrow = p$.

Reiter later extended SSAs to a more general form (see the remark below), and our explicit SSAs are ‘in between’ Reiter’s original proposal and his later generalization. Our reason to opt for them is that their form guarantees that actions a that do not occur in the SSA for p preserve the truth value of p :

PROPOSITION 6 Let $\forall x \mathbf{K}\Box ([x]p \leftrightarrow \gamma_p(x))$ be an explicit SSA. If $a \notin A(\gamma_p(x))$ then

$$\forall x \mathbf{K}\Box ([x]p \leftrightarrow \gamma_p(x)) \models_{\text{ES}} \mathbf{K}\Box ([a]p \leftrightarrow p).$$

REMARK 7 Lakemeyer&Levesque adopt the generalization of [Reiter, 2001a] and allow SSAs to take the form

$$\forall x \mathbf{K}\Box ([x]p \leftrightarrow \gamma_p(x)),$$

where $\gamma_p(x)$ is any fluent formula. Radical examples of such SSAs are $\forall x \mathbf{K}\Box ([x]p \leftrightarrow \top)$: “ p is true after every action” and $\forall x \mathbf{K}\Box ([x]p \leftrightarrow \neg p)$: “every action toggles p ”. We note in passing that such more general SSAs conflict with Reiter’s intuition that only few actions affect a given fluent: “The intuition leading to the frame problem is that most actions do not affect the fluent. So few actions affect it. So its successor state axiom is short.” [Reiter, 2001a, p. 34] They also appear to be less elaboration tolerant in McCarthy’s sense [McCarthy, 1998]. But most importantly, Proposition 6 does not hold for these more general SSAs. To see this it suffices to look at the radical $\forall x \mathbf{K}\Box ([x]p \leftrightarrow \top)$ that they allow: p does not occur in \top , but $\forall x \mathbf{K}\Box ([x]p \leftrightarrow \top) \not\models_{\text{ES}} \mathbf{K}\Box ([a]p \leftrightarrow p)$. Our method cannot handle their SSAs exactly for that reason, as we will explain in Subsection 4.1.

Technically, if the set of action constants A is finite then explicit SSAs are just as expressive as Lakemeyer&Levesque’s SSAs: given one of their more general SSAs $\forall x \mathbf{K}\Box ([x]p \leftrightarrow \gamma_p(x))$, we replace the right-hand side $\gamma_p(x)$ by

$$\gamma_p(x) \vee \bigvee_{\{a | a \notin A(\gamma_p(x)), \not\models \gamma_p(a) \leftrightarrow p\}} (x = a \wedge \neg x = a)$$

In words, we add inconsistent ‘dummy’ disjuncts for all those action constants which do not occur in $\gamma_p(x)$ but may change p . The result is well-defined when the set of action constants is finite, and is an explicit SSA that is equivalent to the original SSA.

2.4 Simple basic action theories

Reiter’s solution to the frame problem requires that action preconditions and effects be described by what he calls *basic action theories*. Actions can be described in this way if we accept some hypotheses. Reiter’s non-epistemic solution relies on the following three. (1) All actions are deterministic. (2) For each $a \in A$ there is a Boolean formula $\delta_{\text{Poss}}(a)$ that characterizes the conditions under which a is executable (action precondition completeness). (3) For each p that is affected by a there is a Boolean formula $\gamma_p(a)$ characterizing the conditions under which p is true after the execution of a (causal completeness). Scherl&Levesque’s epistemic extension relies on the following supplementary hypotheses: (4) The agent

knows the basic action theory Σ under concern. (5) The agent learns about all action occurrences (perfect perception). (6) For each action a , there is a formula $\delta_{SF}(a)$ that characterizes what is perceived by the agent via the execution of a .

DEFINITION 8 A *basic action theory with explicit SSAs* is a set of formulas $\Sigma = \Sigma_0 \cup \Sigma_{\text{Poss}} \cup \Sigma_{\text{SF}} \cup \Sigma_{\text{SSA}}$ such that:

- Σ_0 is a collection of formulas of the form either φ or $\mathbf{K}\varphi$ or $\neg\mathbf{K}\varphi$, where φ is a fluent formula;
- Σ_{Poss} is a collection of formulas $\Sigma_{\text{Poss}}(a) = \mathbf{K}\Box (Poss(a) \leftrightarrow \delta_{Poss}(a))$ with $\delta_{Poss}(a)$ Boolean, one for each $a \in A$;
- Σ_{SF} is a collection of formulas $\Sigma_{\text{SF}}(a) = \mathbf{K}\Box (SF(a) \leftrightarrow \delta_{SF}(a))$ with $\delta_{SF}(a)$ Boolean, one for each $a \in A$;
- Σ_{SSA} is a collection of explicit successor state axioms, one for each $p \in P$.

Remember that the set of fluents P is not necessarily finite: it is just required to be countable. Therefore our definition admits infinite theories with actions a affecting an infinite number of fluents.

As Reiter explains, basic action theories count as a solution to the representational frame problem. In particular, for finite numbers of fluents m and actions n , the size of the set of all SSAs can be expected to be of the order of m . This is much smaller than in the case of explicit frame axioms, where the action theory is of the order of twice the product of the number of actions and the number of fluents, i.e. $2 \times n \times m$.

Given a basic action theory with explicit successor state axioms Σ and a formula φ , the *entailment problem* in ES is to decide whether $\Sigma \models_{\text{ES}} \varphi$, i.e. whether φ is a valid ES consequence of Σ .

EXAMPLE 9 ([LAKEMEYER AND LEVESQUE, 2005]) There is a robot in a 1-dimensional world, and that can move towards or away from a fixed wall. It has a sonar sensor that tells him when it gets close to the wall, where ‘close’ is understood as being zero or one unit away from the wall. The fluents $dist_0, dist_1, dist_2, \dots$ express that the robot is k units away from the wall. The available ontic actions are fwd and bwd moving the robot one unit forward or backward, and the only epistemic action is $sonar$ which tests the robot’s distance from the wall and by which the robot learns whether he is close to the wall ($dist_0 \vee dist_1$) or not ($\neg dist_0 \wedge \neg dist_1$). Initially the robot is one unit away from the wall, but does not know this.

A basic action theory with explicit SSAs for this example is made up of:

$$\begin{aligned} \Sigma_0 &= \{ dist_1 \wedge \neg\mathbf{K}dist_1 \} \cup \{ \neg dist_k \wedge \neg\mathbf{K}\neg dist_k \mid k \geq 2 \} \\ \Sigma_{\text{Poss}} &= \{ \mathbf{K}\Box (Poss(fwd) \leftrightarrow \neg dist_0), \mathbf{K}\Box (Poss(bwd) \leftrightarrow \top), \mathbf{K}\Box (Poss(sonar) \leftrightarrow \top) \} \\ \Sigma_{\text{SF}} &= \{ \mathbf{K}\Box (SF(fwd) \leftrightarrow \top), \mathbf{K}\Box (SF(bwd) \leftrightarrow \top), \mathbf{K}\Box (SF(sonar) \leftrightarrow (dist_0 \vee dist_1)) \} \\ \Sigma_{\text{SSA}} &= \{ \forall x \mathbf{K}\Box ([x]dist_k \leftrightarrow ((x=fwd \wedge dist_{k+1}) \vee (x=bwd \wedge dist_{k-1}) \vee \\ &\quad (dist_k \wedge \neg(x=fwd) \wedge \neg(x=bwd)))) \mid k \in \mathbb{Z} \} \end{aligned}$$

We have e.g. that $\Sigma \models_{\text{ES}} [fwd](dist_0 \wedge \neg\mathbf{K}dist_0)$. Note that the sets of fluents affected by fwd and bwd are both infinite.²

²It is our restriction to propositional fluents which requires an infinite Σ_{SSA} . This differs from Lakemeyer and Levesque’s papers, whose language has object variables, functional fluents and (at least a bit of) integer arithmetics at its disposal. There, the SSA for the distance fluent is $\forall x \forall d \mathbf{K}\Box [x](dist = d) \leftrightarrow \gamma(x, d)$ with $\gamma(x, d)$ being:

$$(x = fwd \wedge dist = d+1) \vee (x = bwd \wedge dist = d-1) \vee (dist = d \wedge \neg(x = fwd \vee x = bwd))$$

As $\gamma(a, d)$ is logically equivalent to $dist = d$ for every action constant a different from fwd and bwd , such an SSA would also be explicit.

REMARK 10 Lakemeyer&Levesque’s original definition of basic action theories differs from ours in several respects.

In Lakemeyer&Levesque’s original definition Σ_{Poss} and Σ_{SF} are singletons (with universal quantification over actions). Contrarily to our definition this does not allow for an infinite number of actions whose preconditions are all different.

More importantly, our elements of Σ_{Poss} , Σ_{SF} and Σ_{SSA} are prefixed by epistemic operators, while theirs are not. This allows them to consider two different basic action theories Σ and Σ' : Σ is about the real world and its dynamics, while Σ' is about the agent’s knowledge and its dynamics. They then prefix the conjunction of formulas in Σ' by an epistemic operator when considering entailments of basic action theories. Formally, they consider ES entailment problems of the form $\Sigma, \mathbf{K}\Sigma' \models_{\text{ES}} \varphi$, where $\mathbf{K}\Sigma'$ has to be understood as the conjunction of the (hence finite) set Σ' . While this looks like an appealing way to model an agent’s incomplete information about both the state of the world and its dynamics, such an approach runs into problems if one considers that Σ' represents an agent’s knowledge, that has to imply truth. Consider a simple example where there is a light and an action *push* of toggling a switch. Suppose the agent believes that toggling switches the light on, while in reality this is the case only if the light is not broken, and suppose initially the light is both off and broken. This is reflected by

$$\begin{aligned}\Sigma_0 &= \{\neg \text{light} \wedge \text{broken}\} \\ \Sigma'_0 &= \{\neg \text{light}\} \\ \Sigma_{\text{SSA}} &= \{\forall x \mathbf{K} \Box ([x]\text{light} \leftrightarrow (x = \text{push} \wedge \neg \text{broken}))\} \\ \Sigma'_{\text{SSA}} &= \{\forall x \mathbf{K} \Box ([x]\text{light} \leftrightarrow (x = \text{push}))\}\end{aligned}$$

Then $\Sigma, \mathbf{K}\Sigma' \models_{\text{ES}} [\text{push}](\neg \text{light} \wedge \mathbf{K}\text{light})$: we are no longer talking about the concept of knowledge. Worse, we have $\Sigma, \mathbf{K}\Sigma' \models_{\text{ES}} [\text{push}][\text{checkLight}](\neg \text{light} \wedge \mathbf{K}\perp)$, for an appropriately defined action *checkLight* of sensing whether the light is on: the agent gets inconsistent after toggling and sensing the state of the light. As said in Remark 3, we have preferred to stick to the standard logic of knowledge, i.e. S5. Our example illustrates that this cannot be guaranteed when the agent’s view of the action laws differs from reality.

What we just have said concerns only the dynamic part of Σ . Notwithstanding we might represent an agent’s incomplete information about the initial situation in its static part Σ_0 . Concerning that component we are slightly more general than [Lakemeyer and Levesque, 2004], whose descriptions of the initial situation are put into two distinct sets: their set Σ_0 contains factual information, and their set Σ'_0 contains information about the agent’s knowledge about the facts (fluent formulas). Our set Σ_0 contains information about both, and may moreover contain information about its ignorance, as illustrated in Example 9 where we put $\Sigma_0 = \{\text{dist}_1, \neg \mathbf{K}\text{dist}_1\}$.

3 Public announcement logic with assignments and tests PALAT

A tradition in modeling dynamics of knowledge that is quite different from the SitCalc approach focusses on particular epistemic actions that make the agents expand their knowledge without changing the world itself, as pursued by Plaza [1989], Baltag et al. [1998], Gerbrandy [1999] and van Benthem [2006]. The basic logic is Plaza’s public announcement logic (PAL) [Plaza, 1989]. It is only recently that ontic actions were introduced into these dynamic epistemic logics [van Ditmarsch et al., 2005; van Benthem et al., 2006; Kooi, 2007]. In this section we recall this extension, that we baptize *public announcement logic with assignments* and that we augment by test actions. We call the result PALAT.

All these dynamic epistemic logics are based on standard epistemic logic, that we recall first.

3.1 Background: epistemic logic EL

Epistemic logics are a family of modal logics that use possible worlds semantics to represent agents' knowledge. This idea, originally due to Hintikka [1962], has known great development in more recent works such as [Meyer and van der Hoek, 1995], [Fagin et al., 1995] and [van Ditmarsch et al., 2007b]. We only present single-agent epistemic logic because epistemic extensions of the SitCalc are restricted to a single agent (with the exception of [Kelly and Pearce, 2008] and [Ghaderi et al., 2007]).

The *language* \mathcal{L}_{EL} of monoagent epistemic logic is the set of formulas φ defined by the following BNF:

$$\varphi ::= p \mid \neg\varphi \mid \varphi \wedge \varphi \mid \mathbf{K}\varphi$$

where p ranges over the nonempty countable set of propositional letters P .

An *epistemic model* (EL model) is a tuple $M = \langle W, R, V \rangle$ such that:

- W is a nonempty set of possible worlds
- $R \subseteq (W \times W)$ is an equivalence relation
- $V : P \rightarrow \wp(W)$ associates an interpretation $V(p) \subseteq W$ to each $p \in P$

For every $w \in W$, the pair (M, w) is a *pointed epistemic model*.

For convenience, we define $R(w) = \{w' \mid (w, w') \in R\}$. The elements of $R(w)$ are the worlds the agent considers possible at w .

Let $(M, w) = (\langle W, R, V \rangle, w)$ be a pointed epistemic model. The truth conditions for EL are:

$M, w \models p$	iff	$w \in V(p)$
$M, w \models \neg\varphi$	iff	$M, w \not\models \varphi$
$M, w \models \varphi \wedge \psi$	iff	$M, w \models \varphi$ and $M, w \models \psi$
$M, w \models \mathbf{K}\varphi$	iff	$R(w) \subseteq \llbracket \varphi \rrbracket_M$

where $\llbracket \varphi \rrbracket_M \stackrel{\text{def}}{=} \{w \mid M, w \models \varphi\}$ is the extension of φ in M .

A formula $\varphi \in \mathcal{L}_{\text{EL}}$ is *EL valid*, noted $\models_{\text{EL}} \varphi$, if and only if for all pointed EL models (M, w) , $(M, w) \models \varphi$.

3.2 Syntax of PALAT

The *language of public announcement logic with assignment and test* $\mathcal{L}_{\text{PALAT}}$ is the set of formulas φ defined by the following BNF:

$$\begin{aligned} \varphi &::= p \mid \neg\varphi \mid \varphi \wedge \varphi \mid \mathbf{K}\varphi \mid [!\varphi]\varphi \mid [??\varphi]\varphi \mid [\sigma]\varphi \\ \sigma &::= \emptyset \mid p := \varphi, \sigma \end{aligned}$$

where p ranges over the nonempty countable set of propositional letters P and where \emptyset is the *empty assignment*. The *language of public announcement logic* \mathcal{L}_{PAL} is $\mathcal{L}_{\text{PALAT}}$ without tests and without assignments.

Let α be one of $!\varphi$, $??\varphi$ or σ ; the formula $[\alpha]\varphi$ reads ' φ holds after all possible executions of α '. The action $!\varphi$ is the public announcement of φ ; the action $??\varphi$ is the public test whether or not φ ; and the action $p := \varphi$ is the public assignment of φ to the atom p . For example, the formula $[??p](\mathbf{K}p \vee \mathbf{K}\neg p)$ —that is going to be valid—expresses that after the public test of p the agent knows whether p ; the expression $p := \perp$ is an assignment making p false; and $\mathbf{K}[p := \perp]\neg p$ is a formula expressing that the agent knows this. In a complex assignment, the same propositional letter can appear only once on the left hand side of

the operator ‘:=’. The elements of such a complex assignment are supposed to be executed in parallel. It is convenient to use a set $\{p_1 := \varphi_1, \dots, p_n := \varphi_n\}$ to denote a complex assignment; the empty assignment \emptyset is then identified with the empty set.

We show that announcements and tests are enough to model epistemic actions, and that assignments are enough to model ontic actions. For example, the epistemic action *sonar* of Example 9 will be modeled as the public test of the sensing formula $\delta_{SF}(\text{sonar})$ associated to *sonar*, i.e. as $??(\text{dist}_0 \vee \text{dist}_1)$. The ontic action *fwd* requires an infinite number of assignments, namely $\{\text{dist}_k := \text{dist}_{k+1} \mid k \in \mathbb{Z}\}$; a complication that we resolve in Section 4.1.

3.3 Semantics of PALAT

Just as formulas of epistemic logic, formulas of $\mathcal{L}_{\text{PALAT}}$ are interpreted in pointed epistemic models. Let $(M, w) = (\langle W, R, V \rangle, w)$ be a pointed epistemic model. The truth conditions are those of Section 3.1 plus the following three:

$$\begin{array}{lll} M, w \models [! \varphi] \psi & \text{iff} & M, w \models \varphi \text{ implies } M^{! \varphi}, w \models \psi \\ M, w \models [?? \varphi] \psi & \text{iff} & M^{?? \varphi}, w \models \psi \\ M, w \models [\sigma] \varphi & \text{iff} & M^\sigma, w \models \varphi \end{array}$$

The models $M^{! \varphi}$, $M^{?? \varphi}$ and M^σ are so-called *updates* of the epistemic model M and are respectively defined as:

$$\begin{array}{lll} M^{! \varphi} = \langle W^{! \varphi}, R^{! \varphi}, V^{! \varphi} \rangle & M^{?? \varphi} = \langle W^{?? \varphi}, R^{?? \varphi}, V^{?? \varphi} \rangle & M^\sigma = \langle W^\sigma, R^\sigma, V^\sigma \rangle \\ W^{! \varphi} = W \cap \llbracket \varphi \rrbracket_M & W^{?? \varphi} = W & W^\sigma = W \\ R^{! \varphi} = R \cap (\llbracket \varphi \rrbracket_M \times \llbracket \varphi \rrbracket_M) & R^{?? \varphi} = R^{! \varphi} \cup R^{! \neg \varphi} & R^\sigma = R \\ V^{! \varphi}(p) = V(p) \cap \llbracket \varphi \rrbracket_M & V^{?? \varphi} = V & V^\sigma(p) = \llbracket \sigma(p) \rrbracket_M \end{array}$$

where in $V^\sigma(p)$, $\sigma(p)$ is the formula assigned to p by σ . If there is no such formula, i.e., if there is no $p := \varphi$ in σ , then $\sigma(p) = p$. For example $\emptyset(p) = p$ for all p , and $\{p := \neg p\}(p) = \neg p$, and $\{p := q, q := p\}(p) = q$. Observe that when $M, w \not\models \varphi$ then $W^{! \varphi}$ is empty, and $M^{! \varphi}$ is therefore not an epistemic model. Updates are therefore partial functions only; however, our truth condition is well-defined because we test that $M, w \models \varphi$ before updating.

To illustrate this definition let (M, w) be any pointed epistemic model. We have $M, w \models [p := \perp] \neg p$ because $V^{p := \perp}(p) = \llbracket (p := \perp)(p) \rrbracket_M = \llbracket \perp \rrbracket_M = \emptyset$; and we have $M, w \models [! p] \mathbf{K} p$ because $V^{! p}(p) = W^{! p}$.

A formula $\varphi \in \mathcal{L}_{\text{PALAT}}$ is *PALAT valid*, noted $\models_{\text{PALAT}} \varphi$, if and only if $(M, w) \models \varphi$ for every pointed epistemic model (M, w) .

For example, $[p := \perp] \neg p$, $[! p] \mathbf{K} p$, $p \rightarrow [?? p] \mathbf{K} p$, and $[?? p](\mathbf{K} p \vee \mathbf{K} \neg p)$ are all PALAT valid, for atomic p . Note that the schema $[! \varphi] \varphi$ is not PALAT valid, nor is $[! \varphi] \mathbf{K} \varphi$ (to see that consider the so-called Moore sentence $\varphi = p \wedge \neg \mathbf{K} p$).

Observe that $[?? \psi]$ is a normal modal operator. A number of valid schemata will be used in proofs in subsequent sections.

PROPOSITION 11 The following equivalences are PALAT valid.

1. $\models_{\text{PALAT}} [?? \psi] \neg \varphi \leftrightarrow \neg [?? \psi] \varphi$
2. $\models_{\text{PALAT}} [p := \psi] \neg \varphi \leftrightarrow \neg [p := \psi] \varphi$
3. $\models_{\text{PALAT}} [?? \psi] \varphi \leftrightarrow \varphi$ if φ is Boolean

$$4. \models_{\text{PALAT}} [??\psi]\varphi \leftrightarrow ([!\psi]\varphi \wedge [!\neg\psi]\varphi)$$

Items 1 and 2 say that tests and assignments are deterministic and executable. Item 3 says that tests do not modify Boolean formulas. The last item tells us that the test operator could be defined in terms of the announcement operator. It therefore does not increase the expressivity of PALAT. Nevertheless, its definition as a primitive operator allows us to provide a polynomial translation of ES sensing actions into PALAT. The action of testing whether φ is nothing but a nondeterministic composition of the announcement of φ and the announcement of $\neg\varphi$. In contrast, if tests were defined as abbreviations then the translation of an ES formula with sensing actions of Section 4 might be exponentially larger than the original formula.

The logic PALAT, but without the ‘test whether’ as a primitive in the language, has been axiomatized and proven complete, see e.g. [Kooi, 2007]. Item 4 of Proposition 11 for ‘test whether’ can be used as a notational abbreviation in that logic. We do not list the axioms and rules of this axiomatization, as they serve no role in the translation that we define in the next section, and the subsequent results.

REMARK 12 The semantics of both the announcement and the test operator is different from that of the dynamic logic test operator ‘?’. First, in dynamic logic $[\varphi?]\psi$ is equivalent to $\varphi \rightarrow \psi$, while $[!\varphi]\psi$ is not equivalent to $\varphi \rightarrow \psi$ in PALAT. Second, the PALAT formula $\neg[??\varphi]\perp$ is valid, while the dynamic logic formula $[\varphi?]\perp$ is not. Also note the difference between our reading of $??\varphi$ as ‘test *whether* φ ’, and the dynamic logic reading of $\varphi?$ as ‘test *that* φ ’.

Now we are ready to establish the link between ES and PALAT.

4 Translation from ES to PALAT

We now show that ES entailment from a basic action theory with explicit SSAs can be translated into a PALAT validity problem. Basically, the change brought about by an action a according to a basic action theory Σ , can be expressed as a PALAT test of the sensed fluents $\delta_{SF}(a)$ followed by a complex assignment simulating the relevant explicit SSAs: the SSAs for those p in φ that are affected by a . Precisely, given a basic action theory Σ we define a polynomial transformation τ_Σ mapping ground box-free formulas φ of the language of ES to PALAT formulas $\tau_\Sigma(\varphi)$, for which we then show that Σ entails φ in ES if and only if $\tau_\Sigma(\varphi)$ is PALAT valid. Before we can do that, we need to bridge one final gap: the case of an infinite number of assignments.

4.1 From actions to assignments

Let Σ be a basic action theory with explicit SSAs as defined in Definition 8, and let $\forall x \mathbf{K}\Box ([x]p \leftrightarrow \gamma_p(x)) \in \Sigma$ be the SSA for p . We associate to every action $a \in A$ a set $\sigma_{\text{SSA}}(a)$ as follows:

$$\sigma_{\text{SSA}}(a) = \{p := \gamma_p(a)\downarrow \mid a \in A(\gamma_p(x))\}$$

The formula $\gamma_p(a) = \gamma_p(x)[x\backslash a]$ is obtained from $\gamma_p(x)$ by replacing every occurrence of x in $\gamma_p(x)$ by a . Remember that $\gamma_p(a)\downarrow$ is the normal form of $\gamma_p(a)$ resulting from the elimination of equalities, and that $\gamma_p(a)\downarrow$ is a Boolean formula. Every $\sigma_{\text{SSA}}(a)$ associates propositional letters with Boolean formulas. When a is purely epistemic (such as *sonar* in our running example) then $\sigma_{\text{SSA}}(a) = \emptyset$.

When $\sigma_{\text{SSA}}(a)$ is finite then it can be identified with a PALAT assignment. However, the set $\sigma_{\text{SSA}}(a)$ is not always finite—and in that case does not correspond to an assignment in $\mathcal{L}_{\text{PALAT}}$. For our running

example we have

$$\begin{aligned}\sigma_{\text{SSA}}(\text{fwd}) &= \{ \text{dist}_k := ((\text{fwd}=\text{fwd} \wedge \text{dist}_{k+1}) \vee (\text{fwd}=\text{bwd} \wedge \text{dist}_{k-1}) \vee \\ &\quad (\text{dist}_k \wedge \neg(\text{fwd}=\text{fwd} \wedge \text{dist}_k) \wedge \neg(\text{fwd}=\text{bwd} \wedge \text{dist}_k))) \downarrow \mid k \in \mathbb{Z} \} \\ &= \{ \text{dist}_k := \text{dist}_{k+1} \mid k \in \mathbb{Z} \}\end{aligned}$$

Similarly, for *bwd* we get

$$\sigma_{\text{SSA}}(\text{bwd}) = \{ \text{dist}_k := \text{dist}_{k-1} \mid k \in \mathbb{Z} \}.$$

The translation that we put forward in the next subsection is defined on formulas $\varphi \in \mathcal{L}_{\text{ES}}^0$. Therefore, we only have to translate actions a in subformulas $[a]\psi$ of such φ . As in $[a]\psi$ the formula ψ contains a finite number of propositional letters only, we can restrict any assignment appearing in the translation to that set of atoms. This solves our predicament:

First, viewing $\sigma_{\text{SSA}}(a)$ as a function, we define the *restriction* of $\sigma_{\text{SSA}}(a)$ to the set of propositional letters $S \subseteq P$ in the following way:

$$\sigma_{\text{SSA}}(a) \downarrow_S = \{ p := \psi \in \sigma_{\text{SSA}}(a) \mid p \in S \}$$

PROPOSITION 13 Let Σ be a basic action theory with explicit SSAs. Let $S \subseteq P$ be a finite set of propositional letters.

1. $\sigma_{\text{SSA}}(a) \downarrow_S$ is a well-formed PALAT assignment for every action a .
2. If $p \in S$ then $\models_{\text{PALAT}} [??\delta_{SF}(a)][\sigma_{\text{SSA}}(a) \downarrow_S]p \leftrightarrow \gamma_p(a) \downarrow$.
3. If φ is a PALAT formula such that $P(\varphi) \subseteq S$ then $\models_{\text{PALAT}} [\sigma_{\text{SSA}}(a) \downarrow_S]\varphi \leftrightarrow [\sigma_{\text{SSA}}(a) \downarrow_{P(\varphi)}]\varphi$.

PROOF. The first and the last result are quite obvious, and we only prove the second item.

Suppose $p \in S$. Observe that

$$\models_{\text{PALAT}} [\sigma_{\text{SSA}}(a) \downarrow_S]p \leftrightarrow \gamma_p(a) \downarrow,$$

which means that the assignment $\sigma_{\text{SSA}}(a) \downarrow_S$ captures in PALAT what Σ stipulates about the behavior of action a in ES. From that we get

$$\models_{\text{PALAT}} [??\delta_{SF}(a)][\sigma_{\text{SSA}}(a) \downarrow_S]p \leftrightarrow [??\delta_{SF}(a)]\gamma_p(a) \downarrow$$

because $[??\delta_{SF}(a)]$ is a normal modal operator. By Proposition 11 the right hand side $[??\delta_{SF}(a)]\gamma_p(a) \downarrow$ is PALAT equivalent to $\gamma_p(a) \downarrow$; we therefore have

$$\models_{\text{PALAT}} [??\delta_{SF}(a)][\sigma_{\text{SSA}}(a) \downarrow_S]p \leftrightarrow \gamma_p(a) \downarrow.$$

■

REMARK 14 We can now explain why we had to require Lakemeyer&Levesque's more general SSAs to be explicit. Consider the set

$$\Sigma_{\text{SSA}} = \{ \forall x \mathbf{K} \Box ([x]p \leftrightarrow \top) \mid p \in P \}$$

where every action makes every p true. None of these SSAs is explicit. According to our definition, $\sigma_{\text{SSA}}(a) = \emptyset$ for every action a . Therefore $[\sigma_{\text{SSA}}(a) \downarrow_{P(p)}]p \leftrightarrow p$; and $\gamma_p(a) \downarrow$ being \top , this illustrates that Item 2 of Proposition 13 does not hold for basic action theories with non-explicit SSAs. In words: $\sigma_{\text{SSA}}(a)$ does not account correctly for the behavior of a ; intuitively, the set $\sigma_{\text{SSA}}(a)$ should not be empty, but rather $\{p := \top \mid p \in P\}$.

4.2 The translation

We are now going to map ground box-free ES formulas to PALAT formulas by replacing every occurrence of an action constant a by a test of the sensed fluent of a that is followed by the complex assignment $\sigma_{\text{SSA}}(a)$. In order to guarantee finiteness the latter is restricted to the propositional letters occurring in the scope of the modal operator $[a]$.

DEFINITION 15 Let Σ be a basic action theory with explicit SSAs as defined in Definition 8. We inductively define the mapping τ_Σ from the set of ground box-free formulas $\mathcal{L}_{\text{ES}}^0$ to $\mathcal{L}_{\text{PALAT}}$:

$$\begin{aligned}\tau_\Sigma(p) &= p \text{ for } p \in P \\ \tau_\Sigma(a_1=a_2) &= (a_1=a_2)\downarrow \\ \tau_\Sigma(Poss(a)) &= \delta_{Poss}(a) \\ \tau_\Sigma(SF(a)) &= \delta_{SF}(a) \\ \tau_\Sigma(\neg\varphi) &= \neg\tau_\Sigma(\varphi) \\ \tau_\Sigma(\varphi_1 \wedge \varphi_2) &= \tau_\Sigma(\varphi_1) \wedge \tau_\Sigma(\varphi_2) \\ \tau_\Sigma(\mathbf{K}\varphi) &= \mathbf{K}\tau_\Sigma(\varphi) \\ \tau_\Sigma([a]\varphi) &= [??\delta_{SF}(a)][\sigma_{\text{SSA}}(a)\downarrow_{P(\tau_\Sigma(\varphi))}]\tau_\Sigma(\varphi)\end{aligned}$$

Remember that $(a_1=a_2)\downarrow$ in the second item is the normal form of $a_1=a_2$ (cf. Proposition 2); hence $\tau_\Sigma(a_1=a_2)$ is \top if a_1 and a_2 are (syntactically) identical, and is \perp otherwise. The formulas $\delta_{Poss}(a)$, $\delta_{SF}(a)$, $\gamma_p^+(a)$ and $\gamma_p^-(a)$ are those from Σ_{Poss} , Σ_{SF} and Σ_{SSA} of Definition 8. In the last item, the assignment $\sigma_{\text{SSA}}(a)$ that is restricted to the language of $\tau_\Sigma(\varphi)$ is as defined in Section 4.1.

PROPOSITION 16 For every $\text{ES} \in \mathcal{L}_{\text{ES}}^0$ formula φ , its translation $\tau_\Sigma(\varphi)$ is a well-formed PALAT formula.

PROOF. This proposition can be proved by induction on the structure of φ , using the following arguments. (1) The definition of τ_Σ neither requires clauses for \forall nor for \Box because φ is ground and box-free. (2) The equality symbol is eliminated during $\tau_\Sigma(\varphi)$, and neither $\delta_{Poss}(a)$ nor $\delta_{SF}(a)$ contain equalities. (3) The predicates *Poss* and *SF* are eliminated. (4) For the case $[a]\varphi$, $\sigma_{\text{SSA}}(a)\downarrow_{P(\tau_\Sigma(\varphi))}$ is a well-formed PALAT assignment (cf. Subsection 4.1). ■

EXAMPLE 17 Consider again our running example. For the epistemic action *sonar* we have $\sigma_{\text{SSA}}(\text{sonar}, \varphi) = \emptyset$, and we get:

$$\begin{aligned}\tau_\Sigma([\text{sonar}]\mathbf{K}(\text{dist}_0 \vee \text{dist}_1)) &= [??(\text{dist}_0 \vee \text{dist}_1)][\emptyset]\tau_\Sigma(\mathbf{K}(\text{dist}_0 \vee \text{dist}_1)) \\ &= [??(\text{dist}_0 \vee \text{dist}_1)][\emptyset]\mathbf{K}(\text{dist}_0 \vee \text{dist}_1)\end{aligned}$$

The last line is PALAT equivalent to $[??(\text{dist}_0 \vee \text{dist}_1)]\mathbf{K}(\text{dist}_0 \vee \text{dist}_1)$ which is PALAT valid.

For the ontic action *fwd* we have $\sigma_{\text{SSA}}(\text{fwd})\downarrow_{P(\tau_\Sigma(\text{dist}_0))} = \{\text{dist}_0 := \text{dist}_1\}$, hence we get:

$$\tau_\Sigma([\text{fwd}]\text{dist}_0) = [??\top][\text{dist}_0 := \text{dist}_1]\text{dist}_0$$

Since $[??\top]\varphi$ simplifies to φ by Proposition 11, this is PALAT equivalent to

$$[\text{dist}_0 := \text{dist}_1]\text{dist}_0$$

We therefore have $P(\tau_\Sigma([\text{fwd}]\text{dist}_0)) = \{\text{dist}_0, \text{dist}_1\}$. This can be used in the following computation:

$$\tau_\Sigma([\text{fwd}][\text{fwd}]\text{dist}_0) = [??\top][\{\text{dist}_0 := \text{dist}_1, \text{dist}_1 := \text{dist}_2\}][??\top][\text{dist}_0 := \text{dist}_1]\text{dist}_0$$

The right hand side successively reduces to $[\{\text{dist}_0 := \text{dist}_1, \text{dist}_1 := \text{dist}_2\}]\text{dist}_1$ and to dist_2 .

The rest of the section is devoted to prove that our translation is correct and that the length of $\tau_\Sigma(\varphi)$ is polynomial in that of φ and Σ .

4.3 Correctness

The next theorem is the main result of our paper, and the rest of this subsection is devoted to its proof.

THEOREM 18 Let Σ be a basic action theory with explicit SSAs, and let φ_0 be a ground box-free formula of $\mathcal{L}_{\text{ES}}^0$. Then

$$\Sigma \models_{\text{ES}} \varphi_0 \text{ if and only if } \models_{\text{PALAT}} \tau_{\Sigma}(\varphi_0)$$

First of all, given a basic action theory with explicit SSAs Σ , an ES formula φ_0 and an epistemic model M , it will be notationally convenient to define the update of M by a sequence of actions from A^* :

$$\begin{aligned} M^\epsilon &= M \\ M^{\alpha \cdot a} &= ((M^\alpha)^{??\delta_{SF}(a)})^{\sigma_{\text{SSA}}(a)} \end{aligned}$$

Strictly speaking, M^α is not well defined because the set $\sigma_{\text{SSA}}(a)$ might be infinite. In this case $\sigma_{\text{SSA}}(a)$ is not a legal assignment, and our definition of M^σ of Subsection 3.3 does not apply. However, that definition can be generalized to infinite sets in the obvious way. The next lemma can be proved using bisimulations restricted to sublanguages.

LEMMA 19 Let $\sigma \subseteq \{p := \chi \mid p \in P, \chi \in \mathcal{L}_{\text{PALAT}}\}$ be a (possibly infinite) set. Let (M, w) be any pointed model. Let φ be any PALAT formula. Then $M^\sigma, w \models \varphi$ iff $M^{\sigma \downarrow P(\varphi)}, w \models \varphi$.

Now we are going to prove the two directions of Theorem 18 by way of two propositions.

PROPOSITION 20 If $\Sigma \models_{\text{ES}} \varphi_0$ then $\models_{\text{PALAT}} \tau_{\Sigma}(\varphi_0)$.

PROOF. Suppose $\not\models_{\text{PALAT}} \tau_{\Sigma}(\varphi_0)$. Then there is a pointed epistemic model $(M, w_0) = (\langle W, R, V \rangle, w_0)$ such that $M, w_0 \not\models \tau_{\Sigma}(\varphi_0)$. From M we define an ES model as follows. Let e_M be the set of all those functions ω_w mapping ground atomic ES formulas to $\{0, 1\}$ such that $w \in R(w_0)$ and

$$\begin{aligned} \omega_w(p, \alpha) &= 1 \text{ iff } M^\alpha, w \models p, \text{ for } p \in P \\ \omega_w(\text{Poss}(a), \alpha) &= 1 \text{ iff } M^\alpha, w \models \delta_{\text{Poss}}(a) \\ \omega_w(SF(a), \alpha) &= 1 \text{ iff } M^\alpha, w \models \delta_{SF}(a) \end{aligned}$$

As R is reflexive we have $\omega_{w_0} \in e_M$. We are going to prove that $\langle e_M, \omega_{w_0} \rangle \not\models \varphi_0$ and that $\langle e_M, \omega_{w_0} \rangle \models \psi$ for every $\psi \in \Sigma$.

LEMMA 21 If φ is a Boolean formula then $M^\alpha, w \models \varphi$ iff $\langle e_M, \omega_w, \alpha \rangle \models \varphi$.

PROOF. The proof is straightforward by induction on the structure of φ . For the base case we use the definition of e_M . ■

The next lemma will be useful in our inductive proof in order to account for the case of the epistemic operator.

LEMMA 22 For every ω_w and $\omega_{w'}$ in e_M we have $\omega_w \sim_\alpha \omega_{w'}$ iff $wR^\alpha w'$.

PROOF. This is proved by induction on the length of α . The base case is clear from the definition of e_M . For the induction step we have:

$$\begin{aligned}
\omega_w \sim_{\alpha \cdot a} \omega_{w'} & \text{ iff } \omega_w \sim_{\alpha} \omega_{w'} \text{ and } \omega_w(SF(a), \alpha) = \omega_{w'}(SF(a), \alpha) \\
& \text{ iff } wR^{\alpha}w' \text{ and } (M^{\alpha}, w \models \delta_{SF}(a) \text{ iff } M^{\alpha}, w' \models \delta_{SF}(a)) \quad (\text{by I.H. and def. of } e_M) \\
& \text{ iff } w(R^{\alpha})^{??\delta_{SF}(a)}w' \\
& \text{ iff } w((R^{\alpha})^{??\delta_{SF}(a)})^{\sigma_{SSA}(a)}w' \\
& \text{ iff } wR^{\alpha \cdot a}w'
\end{aligned}$$

■

The next lemma establishes that $\langle e, \omega_{w_0} \rangle \models \varphi_0$. It uses the preceding lemmas.

LEMMA 23 For every $w \in W$, $\alpha \in A^*$ and every formula φ , $\langle e_M, \omega_w, \alpha \rangle \models \varphi$ iff $M^{\alpha}, w \models \tau_{\Sigma}(\varphi)$.

PROOF. We proceed by induction on φ . The base cases are:

$$\begin{aligned}
\langle e_M, \omega_w, \alpha \rangle \models p & \text{ iff } M^{\alpha}, w \models p & (\text{by Lemma 21}) \\
\langle e_M, \omega_w, \alpha \rangle \models Poss(a) & \text{ iff } \omega_w(Poss(a), \alpha) = 1 \\
& \text{ iff } M^{\alpha}, w \models \delta_{Poss}(a) & (\text{by def. of } e_M) \\
\langle e_M, \omega_w, \alpha \rangle \models SF(a) & \text{ iff } \omega_w(SF(a), \alpha) = 1 \\
& \text{ iff } M^{\alpha}, w \models \delta_{SF}(a) & (\text{by def. of } e_M)
\end{aligned}$$

For the induction step the cases of the Boolean operators are straightforward. For the epistemic operator we have:

$$\begin{aligned}
\langle e_M, \omega_w, \alpha \rangle \models \mathbf{K}\varphi & \text{ iff } \langle e_M, \omega_{w'}, \alpha \rangle \models \varphi \text{ for every } \omega_{w'} \text{ such that } \omega_w \sim_{\alpha} \omega_{w'} \\
& \text{ iff } M^{\alpha}, w' \models \tau_{\Sigma}(\varphi) \text{ for every } \omega_{w'} \text{ such that } \omega_w \sim_{\alpha} \omega_{w'} & (\text{by I.H.}) \\
& \text{ iff } M^{\alpha}, w' \models \tau_{\Sigma}(\varphi) \text{ for every } w' \text{ such that } wR^{\alpha}w' & (\text{by Lemma 22}) \\
& \text{ iff } M^{\alpha}, w \models \mathbf{K}\tau_{\Sigma}(\varphi) \\
& \text{ iff } M^{\alpha}, w \models \tau_{\Sigma}(\mathbf{K}\varphi)
\end{aligned}$$

Finally for the action operator we have:

$$\begin{aligned}
\langle e_M, \omega_w, \alpha \rangle \models [a]\varphi & \text{ iff } \langle e_M, \omega_w, \alpha \cdot a \rangle \models \varphi \\
& \text{ iff } M^{\alpha \cdot a}, w \models \tau_{\Sigma}(\varphi) & (\text{by I.H.}) \\
& \text{ iff } ((M^{\alpha})^{??\delta_{SF}(a)})^{\sigma_{SSA}(a)}, w \models \tau_{\Sigma}(\varphi) \\
& \text{ iff } ((M^{\alpha})^{??\delta_{SF}(a)})^{\sigma_{SSA}(a) \downarrow_{P(\tau_{\Sigma}(\varphi))}}, w \models \tau_{\Sigma}(\varphi) & (\text{by Lemma 19}) \\
& \text{ iff } (M^{\alpha})^{??\delta_{SF}(a)}, w \models [\sigma_{SSA}(a) \downarrow_{P(\tau_{\Sigma}(\varphi))}] \tau_{\Sigma}(\varphi) \\
& \text{ iff } M^{\alpha}, w \models [??\delta_{SF}(a)][\sigma_{SSA}(a) \downarrow_{P(\tau_{\Sigma}(\varphi))}] \tau_{\Sigma}(\varphi) \\
& \text{ iff } M^{\alpha}, w \models \tau_{\Sigma}([a]\varphi)
\end{aligned}$$

■

It follows from this lemma that $\langle e_M, \omega_{w_0} \rangle \models \varphi_0$ iff $M, w_0 \models \tau_{\Sigma}(\varphi_0)$. We still have to prove that for every $\psi \in \Sigma$ we have $\langle e_M, \omega_{w_0} \rangle \models \psi$. Due to the definition of e_M things are clear for the elements of Σ_{Poss}

and Σ_{SF} . So let us consider explicit SSAs of the form $\forall x \mathbf{K}\Box ([x]p \leftrightarrow \gamma_p(x))$ from Σ_{SSA} . We are going to show that $\langle e_M, \omega_w, \alpha \rangle \models [a]p \leftrightarrow \gamma_p(a)$ for every $\omega_w \in e_M$, $\alpha \in A^*$ and $a \in A$:

$$\begin{aligned}
\langle e_M, \omega_w, \alpha \rangle \models [a]p & \text{ iff } \langle e_M, \omega_w, \alpha \cdot a \rangle \models p \\
& \text{ iff } M^{\alpha \cdot a}, w \models p & (\text{by Lemma 21}) \\
& \text{ iff } ((M^\alpha)^{??\delta_{SF}(a)})^{\sigma_{SSA}(a)}, w \models p \\
& \text{ iff } ((M^\alpha)^{??\delta_{SF}(a)})^{\sigma_{SSA}(a) \downarrow_{[p]}}, w \models p \\
& \text{ iff } M^\alpha, w \models [??\delta_{SF}(a)][\sigma_{SSA}(a) \downarrow_{[p]}]p \\
& \text{ iff } M^\alpha, w \models \gamma_p(a) \downarrow & (\text{by Prop. 13.2}) \\
& \text{ iff } \langle e_M, \omega_w, \alpha \rangle \models \gamma_p(a) \downarrow & (\text{by Lemma 21}) \\
& \text{ iff } \langle e_M, \omega_w, \alpha \rangle \models \gamma_p(a) & (\text{by Prop. 2})
\end{aligned}$$

We have just shown that for every $\psi \in \Sigma$ we have $\langle e_M, \omega_{w_0} \rangle \models \psi$, and that $\langle e_M, \omega_{w_0} \rangle \not\models \varphi_0$. Therefore $\Sigma \not\models_{ES} \varphi_0$. This ends the proof of Proposition 20, which is half of the correctness proof. ■

PROPOSITION 24 If $\models_{PALAT} \tau_\Sigma(\varphi_0)$ then $\Sigma \models_{ES} \varphi_0$.

PROOF. Suppose $\Sigma \not\models_{ES} \varphi_0$. Then there is e and $\omega_0 \in e$ such that $\langle e, \omega_0 \rangle \models \psi$ for every $\psi \in \Sigma$ and $\langle e, \omega_0 \rangle \not\models \varphi_0$. From e we define an epistemic model $M_e = \langle W, R, V \rangle$ such that $W = e$, $R = e \times e$ and $V(p) = \{\omega \in e \mid \omega(p, \epsilon) = 1\}$. Clearly R is an equivalence relation; M_e is therefore an EL model.

LEMMA 25 If φ is a Boolean formula then $\langle e, \omega, \alpha \rangle \models \varphi$ iff $M_e^\alpha, \omega \models \varphi$.

PROOF. We use induction on the length of α . For the base case we prove by induction on φ that $\langle e, \omega, \epsilon \rangle \models \varphi$ iff $M_e^\epsilon, \omega \models \varphi$. The base of that sub-induction is clear by definition of M_e , and the cases of the Boolean operators are straightforward.

For the induction step we have to prove that $\langle e, \omega, \alpha \cdot a \rangle \models \varphi$ iff $M_e^{\alpha \cdot a}, \omega \models \varphi$. Again we use induction on φ to prove this. The base case of the sub-induction is:

$$\begin{aligned}
\langle e, \omega, \alpha \cdot a \rangle \models p & \text{ iff } \langle e, \omega, \alpha \rangle \models [a]p \\
& \text{ iff } \langle e, \omega, \alpha \rangle \models \gamma_p(a) & (\text{because } \langle e, \omega \rangle \models \Sigma) \\
& \text{ iff } \langle e, \omega, \alpha \rangle \models \gamma_p(a) \downarrow & (\text{by Prop. 2}) \\
& \text{ iff } M_e^\alpha, \omega \models \gamma_p(a) \downarrow & (\text{by I.H., because } \gamma_p(a) \downarrow \text{ Boolean}) \\
& \text{ iff } M_e^\alpha, \omega \models [??\delta_{SF}(a)]\gamma_p(a) \downarrow & (\text{by Prop. 11}) \\
& \text{ iff } M_e^\alpha, \omega \models [??\delta_{SF}(a)][\sigma_{SSA}(a) \downarrow_{P(p)}]p & (\text{by Prop. 13.2}) \\
& \text{ iff } ((M_e^\alpha)^{??\delta_{SF}(a)})^{\sigma_{SSA}(a) \downarrow_{P(p)}}, \omega \models p \\
& \text{ iff } ((M_e^\alpha)^{??\delta_{SF}(a)})^{\sigma_{SSA}(a)}, \omega \models p & (\text{by Lemma 19}) \\
& \text{ iff } M_e^{\alpha \cdot a}, \omega \models p
\end{aligned}$$

The cases of the Boolean operators are straightforward. ■

The following property of M_e will be useful in the subsequent induction in order to account for the case of the epistemic operator.

LEMMA 26 For every ω and $\omega' \in e$ we have $\omega \sim_\alpha \omega'$ iff $\omega R^\alpha \omega'$.

PROOF. We use induction on the length of the sequence α . Clearly, $\omega \sim_\epsilon \omega'$ iff $\omega R^\epsilon \omega'$ by definition of M_e . For the induction step we have:

$$\begin{aligned}
\omega \sim_{\alpha \cdot a} \omega' &\text{ iff } \omega \sim_\alpha \omega' \text{ and } \omega(SF(a), \alpha) = \omega'(SF(a), \alpha) \\
&\text{ iff } \omega R^\alpha \omega' \text{ and } (\langle e, \omega, \alpha \rangle \models SF(a) \text{ iff } \langle e, \omega', \alpha \rangle \models SF(a)) && \text{(by I.H.)} \\
&\text{ iff } \omega R^\alpha \omega' \text{ and } (\langle e, \omega, \alpha \rangle \models \delta_{SF}(a) \text{ iff } \langle e, \omega', \alpha \rangle \models \delta_{SF}(a)) && \text{(because } \langle e, \omega \rangle \models \Sigma) \\
&\text{ iff } \omega R^\alpha \omega' \text{ and } (M_e^\alpha, \omega \models \delta_{SF}(a) \text{ iff } M_e^\alpha, \omega' \models \delta_{SF}(a)) && \text{(by Lemma 25)} \\
&\text{ iff } \omega(R^\alpha)^{??\delta_{SF}(a)} \omega' \\
&\text{ iff } \omega((R^\alpha)^{??\delta_{SF}(a)})^{\sigma_{SSA}(a)} \omega' \\
&\text{ iff } \omega R^{\alpha \cdot a} \omega'
\end{aligned}$$

■

To complete the proof we finally show that for every $\omega \in e$ and every α we have $\langle e, \omega, \alpha \rangle \models \varphi$ iff $M_e^\alpha, \omega \models \tau_\Sigma(\varphi)$. We do so by induction on the structure of φ . For the base cases $Poss(a)$ and $SF(a)$ we use that $\langle e, \omega, \alpha \rangle \models \psi$ for every $\psi \in \Sigma$, and for the base case p we use Lemma 25. For the epistemic operator we have:

$$\begin{aligned}
\langle e, \omega, \alpha \rangle \models \mathbf{K}\varphi &\text{ iff } \langle e, \omega', \alpha \rangle \models \varphi \text{ for every } \omega' \text{ such that } \omega \sim_\alpha \omega' \\
&\text{ iff } M_e^\alpha, \omega' \models \tau_\Sigma(\varphi) \text{ for every } \omega' \text{ such that } \omega \sim_\alpha \omega' && \text{(by I.H.)} \\
&\text{ iff } M_e^\alpha, \omega \models \tau_\Sigma(\varphi) \text{ for every } \omega' \text{ such that } \omega R^\alpha \omega' && \text{(by Lemma 26)} \\
&\text{ iff } M_e^\alpha, \omega \models \mathbf{K}\tau_\Sigma(\varphi) \\
&\text{ iff } M_e^\alpha, \omega \models \tau_\Sigma(\mathbf{K}\varphi)
\end{aligned}$$

And for the action operator we have:

$$\begin{aligned}
\langle e, \omega, \alpha \rangle \models [a]\varphi &\text{ iff } \langle e, \omega, \alpha \cdot a \rangle \models \varphi \\
&\text{ iff } M_e^{\alpha \cdot a}, \omega \models \tau_\Sigma(\varphi) && \text{(by I.H.)} \\
&\text{ iff } ((M_e^\alpha)^{??\delta_{SF}(a)})^{\sigma_{SSA}(a)}, \omega \models \tau_\Sigma(\varphi) \\
&\text{ iff } ((M_e^\alpha)^{??\delta_{SF}(a)})^{\sigma_{SSA}(a) \downarrow_{P(\tau_\Sigma(\varphi))}}, \omega \models \tau_\Sigma(\varphi) && \text{(by Lemma 19)} \\
&\text{ iff } M_e^\alpha, \omega \models [??\delta_{SF}(a)][\sigma_{SSA}(a) \downarrow_{P(\tau_\Sigma(\varphi))}] \tau_\Sigma(\varphi) \\
&\text{ iff } M_e^\alpha, \omega \models \tau_\Sigma([a]\varphi)
\end{aligned}$$

It follows that $\langle e, \omega_0 \rangle \models \varphi_0$ iff $M_e, \omega_0 \models \tau_\Sigma(\varphi_0)$. This ends the proof of Proposition 24, which is the second half of Theorem 18. ■

Together, Propositions 20 and 24 establish Theorem 18.

4.4 Polynomial length

To prove that for finite action theories Σ our transformation is polynomial we recursively define the function len returning the *length* of assignments, formulas and finite sets of formulas. Basically, we count

the number of occurrences of fluents, propositional literals, action names and logical connectives. For formulas in \mathcal{L}_{EL} , we define:

$$\begin{aligned}\text{len}(p) &= 1 \\ \text{len}(\neg\varphi) &= 1 + \text{len}(\varphi) \\ \text{len}(\varphi \wedge \psi) &= 1 + \text{len}(\varphi) + \text{len}(\psi) \\ \text{len}(\mathbf{K}\varphi) &= 1 + \text{len}(\varphi)\end{aligned}$$

For ground box-free formulas in \mathcal{L}_{ES}^0 we moreover define:

$$\begin{aligned}\text{len}(a_1=a_2) &= 5 \\ \text{len}(\text{Poss}(a)) &= \text{len}(SF(a)) = 1 \\ \text{len}([a]\varphi) &= 3 + \text{len}(\varphi)\end{aligned}$$

and for formulas of \mathcal{L}_{PALAT} we moreover define:

$$\begin{aligned}\text{len}([\psi]\varphi) &= 1 + \text{len}(\psi) + \text{len}(\varphi) \\ \text{len}([\psi]\varphi) &= 1 + \text{len}(\psi) + \text{len}(\varphi) \\ \text{len}([\sigma]\varphi) &= 1 + \text{len}(\sigma) + \text{len}(\varphi) \\ \text{len}(\emptyset) &= 0 \\ \text{len}(p := \varphi, \sigma) &= 1 + \text{len}(\varphi) + \text{len}(\sigma)\end{aligned}$$

where we consider σ as a set of assignments. For sets we sum up the lengths of their elements.

For example, $\text{len}(\perp) = \text{len}(p \wedge \neg p) = 4$, $\text{len}(\top) = 5$, and

$$\begin{aligned}\text{len}(\{p := q, q := p \wedge q\}\mathbf{K}p) &= 1 + \text{len}(\{p := q, q := p \wedge q\}) + \text{len}(\mathbf{K}p) \\ &= 1 + ((1 + 1) + (1 + 2)) + (1 + 1) \\ &= 8\end{aligned}$$

THEOREM 27 Let Σ be a finite basic action theory with explicit SSAs, and let $\varphi \in \mathcal{L}_{ES}^0$. Then $\text{len}(\tau_\Sigma(\varphi)) \leq \text{len}(\varphi) \times \text{len}(\Sigma)$.

PROOF. The proof is by induction on the structure of formulas. First, observe that $\text{len}(\Sigma) \geq 1$ because the set of fluents is non-empty, and that the lengths of $\delta_{\text{Poss}}(a)$, $\delta_{SF}(a)$ and $\sigma_{\text{SSA}}(a)$ are bound by $\text{len}(\Sigma)$, as well as the sum of the length of $\delta_{SF}(a)$ and $\sigma_{\text{SSA}}(a) \downarrow_{P(\tau_\Sigma(\varphi))}$, for all φ .

The basic cases where φ is of the form $p \in P$, $a_1 = a_2$, $\text{Poss}(a)$ and $SF(a)$ are straightforward. For equality remember that $\tau_\Sigma(a_1 = a_2)$ is either \top or \perp , and hence of length 4 or 5, which is why we chose $\text{len}(a_1=a_2) = 5$.

For the induction step, the cases of negation, conjunction and of the epistemic operator are proved by straightforward application of the induction hypothesis. For the remaining case we have:

$$\begin{aligned}\text{len}(\tau_\Sigma([a]\varphi_1)) &= \text{len}([\delta_{SF}(a)]\tau_\Sigma(\varphi_1)) \\ &\leq \text{len}([\delta_{SF}(a)]\tau_\Sigma(\varphi_1)) && \text{(because } \Sigma \text{ is finite)} \\ &= 1 + \text{len}(\delta_{SF}(a)) + 1 + \text{len}(\tau_\Sigma(\varphi_1)) && \text{(by def.)} \\ &\leq 2 + \text{len}(\Sigma) + \text{len}(\tau_\Sigma(\varphi_1)) && \text{(bec. } \text{len}(\delta_{SF}(a)) + \text{len}(\sigma_{\text{SSA}}(a)) \leq \text{len}(\Sigma)) \\ &\leq 2 + \text{len}(\Sigma) + (\text{len}(\varphi_1) \times \text{len}(\Sigma)) && \text{(by IH)} \\ &\leq (3 + \text{len}(\varphi_1)) \times \text{len}(\Sigma) \\ &= \text{len}([a]\varphi_1) \times \text{len}(\Sigma)\end{aligned}$$

This ends the proof. ■

Hence for finite basic action theories with explicit SSAs the problem of deciding whether $\Sigma \models_{\text{ES}} \varphi$ can be polynomially reduced to a validity problem in PALAT. For infinite action theories Σ , $\tau_\Sigma(\varphi)$ might be exponentially longer than φ . We have:

THEOREM 28 Let Σ be a basic action theory with explicit SSAs such that every element of the set Σ has length at most N , for $N \geq 2$, and let $\varphi \in \mathcal{L}_{\text{ES}}^0$. Then $\text{len}(\tau_\Sigma(\varphi)) \leq \text{len}(\varphi) \times N^{\text{len}(\varphi)}$.

PROOF. Observe first that $\text{len}(\sigma_{\text{SSA}}(a) \downarrow_{P(\tau_\Sigma(\psi))}) \leq N \times \text{len}(\tau_\Sigma(\psi))$.

The basic cases where φ is of the form $p \in P$, $a_1 = a_2$, $\text{Poss}(a)$ and $SF(a)$ are straightforward.

For the induction step, the cases of negation, conjunction and of the epistemic operator are proved by straightforward application of the induction hypothesis. For the remaining case we have:

$$\begin{aligned}
 \text{len}(\tau_\Sigma([a]\varphi_1)) &= \text{len}([?\delta_{SF}(a)][\sigma_{\text{SSA}}(a) \downarrow_{P(\tau_\Sigma(\varphi_1))}]\tau_\Sigma(\varphi_1)) \\
 &\leq 1 + N + 1 + (N \times \text{len}(\tau_\Sigma(\varphi_1))) + \text{len}(\tau_\Sigma(\varphi_1)) && \text{(cf. obs. above)} \\
 &= 2 + N + (N + 1) \times \text{len}(\tau_\Sigma(\varphi_1)) \\
 &\leq 2 + N + (N + 1) \times \text{len}(\varphi_1) \times N^{\text{len}(\varphi_1)} && \text{(by I.H.)} \\
 &\leq (\text{len}(\varphi_1) + 3) \times N^{\text{len}(\varphi_1) + 3} && \text{(because } N \geq 2) \\
 &= \text{len}([a]\varphi_1) \times N^{\text{len}([a]\varphi_1)}
 \end{aligned}$$

■

5 Perspectives: an extension of ES by event models

In ES it is supposed that the agent perceives what is happening both soundly and completely. In this section we show how the bridge we built in the preceding section can be taken to extend ES in order to account for the agent's incomplete perception. To that end we import ideas from [Baltag et al., 1998; Baltag and Moss, 2004; Aucher and Herzig, 2007, 2010] into ES. We only sketch how this can be done, leaving the details for another occasion.

Consider a variant of our running example where the robot does not move by himself, but is pushed forward or backward by some other agent, and suppose that the robot only perceives that it is moved, but is not able to distinguish whether it is the *fwd* or the *bwd* event which is taking place. (It is more appropriate to talk of events instead of actions here.) Such a situation cannot be modeled directly in ES.³ In order to do that we extend ES to a new logic ES'. Following [Aucher and Herzig, 2007, 2010] we augment the language of ES by converse events $[a]$, where $[a]\varphi$ reads “if a occurred then φ was true before”. They allow to talk about the agent's perception of events: $\neg[a]\perp$ expresses that a has just occurred; $\mathbf{K}\neg[fwd]\perp$ expresses that the agent knows this, and $\neg\mathbf{K}[fwd]\perp$ expresses that it is possible for the agent that *fwd* has just occurred.

In the semantics we consider pairs (ω, α) where $\omega \in \Omega_{\text{ES}}$ as in Subsection 2.2 and $\alpha \in A^*$, and replace the set e by an equivalence relation

$$E \subseteq (\Omega_{\text{ES}} \times A^*) \times (\Omega_{\text{ES}} \times A^*)$$

³This holds for our version of ES. It cannot be implemented directly either in the more general original version of [Lakemeyer and Levesque, 2004] which allows for two different sets of SSAs Σ_{SSA} and Σ'_{SSA} , where Σ_{SSA} describes the evolution of the world, and Σ'_{SSA} describes the evolution of the agent's knowledge. The reason is that the agent's perception of the *fwd* action is that of a nondeterministic event. This could be implemented by resorting to nondeterministic composition.

If $(\omega, \epsilon)E(\omega', \epsilon)$ then the agent cannot distinguish ω from ω' ; If $(\omega, \alpha \cdot a)E(\omega', \alpha' \cdot a')$ then the agent cannot distinguish the occurrence of event a in (ω, α) from the occurrence of event a' in (ω', α') . The relation E must satisfy the following constraints:

- if $(\omega, \alpha \cdot a)E(\omega', \alpha' \cdot a')$ then $(\omega, \alpha)E(\omega', \alpha')$ and $\omega(SF(a), \alpha) = \omega'(SF(a'), \alpha')$;
- if $(\omega, \alpha)E(\omega, \epsilon)$ then $\alpha = \epsilon$;
- if $(\omega, \alpha)E(\omega', \alpha')$ and $(\omega, \alpha \cdot a)E(\omega'', \alpha'' \cdot a'')$ then $(\omega, \alpha \cdot a)E(\omega', \alpha' \cdot a'')$.

The last constraint says that when a'' is possible for the agent after performing a at (ω, α) (i.e., when $(\omega, \alpha \cdot a)E(\omega'', \alpha'' \cdot a'')$) then after a the agent systematically takes into account the occurrence of a'' at every (ω', α') that was possible for the agent before a . The original ES models are a particular case: they can be retrieved by requiring moreover that $(\omega, \alpha)E(\omega', \alpha')$ implies that α and α' are identical. Then for a given ω we may set e to the E equivalence class of (ω, ϵ) , and we have $\langle e, \omega, \alpha \rangle \models \varphi$ iff $\langle E, \omega, \alpha \rangle \models \varphi$ for every ES formula φ .

Formulas of ES are interpreted in triples $\langle E, \omega, \alpha \rangle$. For the modal operators we stipulate:

$$\begin{array}{lll}
\langle E, \omega, \alpha \rangle \models \mathbf{K}\varphi & \text{iff} & \langle E, \omega', \alpha \rangle \models \varphi \text{ for all } (\omega', \alpha') \text{ such that } (\omega', \alpha')E(\omega, \alpha) \\
\langle E, \omega, \alpha \rangle \models [a]\varphi & \text{iff} & \langle E, \omega, \alpha \cdot a \rangle \models \varphi \\
\langle E, \omega, \alpha \rangle \models [a]\neg\varphi & \text{iff} & \text{if } \alpha = \alpha_0 \cdot a \text{ then } \langle E, \omega, \alpha_0 \rangle \models \neg\varphi \\
\langle E, \omega, \alpha \rangle \models \Box\varphi & \text{iff} & \text{for all } \alpha' \in A^*, \langle E, \omega, \alpha \cdot \alpha' \rangle \models \varphi
\end{array}$$

The truth conditions for atoms and for Boolean operators are as expected. Observe that $\langle E, \omega, \alpha \rangle \models \neg[a]\perp$ iff $\alpha = \alpha_0 \cdot a$. We write $\Psi \models_{\text{ES}} \varphi$ when φ is a valid ES consequence of Ψ , which is defined in the obvious way. Observe that while $\models_{\text{ES}} \neg[a]\neg\varphi \rightarrow [a]\varphi$, we do not have $\models_{\text{ES}} [a]\varphi \rightarrow \neg[a]\neg\varphi$.

In ES, the formula $[fwd](\neg\mathbf{K}[fwd]\perp \wedge \neg\mathbf{K}[bwd]\perp)$ expresses that the agent cannot distinguish the occurrence of fwd from that of bwd . Suppose given a function $SE : A \rightarrow 2^A$ such that $a \in SE(a)$ for every $a \in A$. The set of sensed events $SE(a)$ is the set of all those events which possibly occur for the agent when a occurs. Consider the set of formulas

$$\Sigma_{\text{SE}}(a) = \forall x \Box [a](\neg\mathbf{K}[x]\perp \leftrightarrow (\bigvee_{a' \in SE(a)} x = a'))$$

For our running example we have

$$\Sigma_{\text{SE}}(fwd) = \forall x \Box [fwd](\neg\mathbf{K}[x]\perp \leftrightarrow (x = fwd \vee x = bwd))$$

Observe that when $\langle E, \omega, \alpha \rangle \models \Sigma_{\text{SE}}(a)$ for every $\alpha \in A^*$ then $(\omega, \alpha \cdot a)E(\omega', \alpha' \cdot a')$ is the case iff $(\omega, \alpha)E(\omega', \alpha')$ and $a' \in SE(a)$. The collection of $\Sigma_{\text{SE}}(a)$, for all $a \in A$, makes up a set Σ_{SE} of what may be called event sensing axioms.

We augment basic action theories by Σ_{SE} , and check whether $\Sigma \models_{\text{ES}} \varphi$. If we do so for our example and change $\Sigma_0 = \{dist_1, \neg\mathbf{K}dist_1\}$ to $\Sigma'_0 = \{\mathbf{K}dist_1\}$, then we get $\Sigma' \models_{\text{ES}} [fwd](\neg\mathbf{K}\neg dist_0 \wedge \neg\mathbf{K}\neg dist_2)$. In words, after being moved forward it is possible for the robot that it is at distance 0 or at distance 2 from the wall.

The following successor state axiom for knowledge (SSAK)

$$\begin{array}{l}
\text{SSAK:} \quad \forall x \Box ([x]\mathbf{K}\varphi \leftrightarrow ((SF(x) \wedge \mathbf{K}(SF(x) \rightarrow \bigwedge_{x' \in SE(x)} [x']\varphi)) \vee \\
\quad (\neg SF(x) \wedge \mathbf{K}(\neg SF(x) \rightarrow \bigwedge_{x' \in SE(x)} [x']\varphi))))
\end{array}$$

is ES entailed by Σ_{SE} . (In contrast, the original SSAK is ES plainly valid.) SSAK allows to regress all ground box-free formulas where no converse operator $[\cdot]$ occurs.

As we have mentioned in the beginning of the section, our example could be implemented in ES if we extend the version of [Lakemeyer and Levesque, 2004] by nondeterministic composition. However, we think that the approach by Baltag et al. is more general; in particular it can be extended straightforwardly to multiple agents, in order to reason about an agent’s perception of what another agent perceives, etc.

6 Conclusion

We have built a bridge between the propositional fragment of Lakemeyer&Levesque’s logic ES and dynamic epistemic logic DEL, by providing a polynomial translation of ES entailment problems into validity problems in a particular DEL: ‘public announcement logic with assignments and tests’, PALAT. In consequence, existing results and algorithms for DELs apply now to ES and the fragment of SitCalc corresponding to ES. In particular, the reduction axioms for public announcement logic with assignments of [van Ditmarsch et al., 2005; van Benthem et al., 2006; Kooi, 2007; van Ditmarsch et al., 2007a] allow for a decision procedure.

Technically, our bridge is built on the close similarities between SitCalc’s successor state axioms and PALAT’s reduction axioms. Most importantly, PALAT has a ‘built-in’ solution to the frame problem: for different propositional letters p and q , the frame axioms $q \rightarrow [p := \varphi]q$ and $\neg q \rightarrow [p := \varphi]\neg q$ are PALAT valid, and therefore need not be included in action theories. We exploit this in the key step of our translation where we encode successor state axioms into assignments. That this is possible is surprising at least at first glance, given that Reiter claims [Reiter, 2001a, p. 34]: “the conciseness and perspicuity of this axiomatization relies on two things: (1) Quantification over actions. (2) The assumption that relatively few actions affect a given fluent.”

As we have stated in Remark 3, we considered a version of the semantics of ES where the actual state ω belongs to the agent’s epistemic state e , although this was not required in Lakemeyer&Levesque’s original papers. We did so in order to ensure validity of the standard ‘knowledge implies truth’ principle $\mathbf{K}\varphi \rightarrow \varphi$. From a technical point of view our approach equally applies to the original version of ES: it suffices to base the dynamic epistemic logic of Section 3 on transitive and Euclidean relations⁴ instead of on equivalence relations, i.e. to move from modal logic S5 to K45. Then the original version of ES can be mapped to that K45 based dynamic epistemic logic via the same translation, and Theorem 18 transfers.⁵ However, both the original version of ES and K45 allow for situations where the agent wrongly believes that p , i.e. situations where $p \wedge \mathbf{K}\neg p$ holds. Subsequent sensing of p then reduces the agent’s set of possibilities to the empty set: his beliefs get inconsistent. To get around this one has to integrate belief revision mechanisms; we wanted to avoid such a complication.

Our bridge should be further enlarged: the full language of ES is much more expressive than the propositional fragment of the present paper, having not only first-order quantification and sorts, but also a special modality of only-knowing. For reasons of decidability and elegance, the DEL community usually avoids adding quantifiers, predicates and functions. However, the large body of existing work on first-order modal logics [Garson, 1984; Fitting, 1998; Gabbay et al., 2009] would provide a formal background for first-order DELs. Reiter’s, Scherl&Levesque’s and Lakemeyer&Levesque’s approaches demonstrate that under some reasonable restrictions quantifiers can be used in practice, as done in the GOLOG programming language [Levesque et al., 1997].

Our bridge highlights that the roads SitCalc and DEL take are converging: as we have seen, SitCalc has started to integrate epistemic logic and epistemic actions, while DEL’s focus was recently extended

⁴A relation R is transitive iff $v \in R(w)$ implies $R(v) \subseteq R(w)$, and R is Euclidean iff $v \in R(w)$ implies $R(w) \subseteq R(v)$. Put together, R is both transitive and Euclidean iff $v \in R(w)$ implies $R(v) = R(w)$.

⁵The only differences are that (1) in the proof of Proposition 20, when building an ES model from a K45 model we do not have to prove any more that $\omega_{w_0} \in e_M$, and that (2) in the proof of Proposition 24, when building a K45 model from an ES model we do not have to prove any more that R is an equivalence relation, but only that it is transitive and Euclidean.

from epistemic to ontic actions. Here is a list of novelties that could be easily brought to the SitCalc bank of the river via our bridge:

- SitCalc does not allow for epistemic operators in the formulas $\delta_{Poss}(a)$, $\delta_{SF}(a)$ and $\gamma_p(x)$ of basic action theories, while both the announcement $!\varphi$ and the assignment $p := \varphi$ may contain such operators. Clearly, ES can be straightforwardly extended here.
- Up to now, epistemic extensions of SitCalc only considered the single agent case (with the exception of [Kelly and Pearce, 2008]), while DEL is multiagent right from the start, providing for group attitudes such as common knowledge and distributed knowledge.
- Up to now, almost all epistemic extensions of SitCalc only considered public events. In order to integrate incomplete perception of events one could proceed as in [Baltag et al., 1998; Baltag and Moss, 2004] and add so-called event models that represent the agents' perception of events. We have sketched a possible extension of ES in Section 5.
- Up to now there was not too much work on the integration of belief revision into SitCalc [Shapiro et al., 2000; Shapiro and Pagnucco, 2004; Demolombe and Pozos Parra, 2006]. Work in this area could cross-fertilize with recent work on the DEL side by Baltag and Smets [2007, 2008, 2009] and others [Ditmarsch, 2005; Benthem, 2007; Aucher, 2008]. The latter provide a more general account of revision, and also provide reduction-based decision procedures.
- SitCalc could benefit from existing model checkers for PAL such as DEMO [van Eijck, 2004] or MCK [Gammie and van der Meyden, 2004].

At this stage, the reader might wonder whether one could also cross the bridge the other way round: find a translation τ^{-1} from PALAT into ES such that

$$\models_{\text{PALAT}} \varphi \text{ iff } \Sigma(\varphi) \models_{\text{ES}} \tau^{-1}(\varphi)$$

This can be done straightforwardly by ‘freezing’ all the assignments $p := \varphi$, announcements $!\varphi$ and tests $??\varphi$, i.e., by respectively replacing them by new action constants $a_{p:=\varphi}$, $a_{!\varphi}$ and $a_{??\varphi}$. The behavior of these actions can then be encoded into an action theory $\Sigma(\varphi)$ as follows:

- $Poss(a_{p:=\psi}) = SF(a_{p:=\psi}) = \top$, and $\gamma_p(x) = \bigvee_{\{\psi | p:=\psi \text{ occurs in } \varphi\}} (x = a_{p:=\psi} \wedge \psi) \vee (p \wedge \neg \bigvee_{\{\psi | p:=\psi \text{ occurs in } \varphi\}} (x = a_{p:=\psi} \wedge \neg \psi))$;
- $Poss(a_{!\varphi}) = SF(a_{!\varphi}) = \varphi$, and $\gamma_p^+(a_{!\varphi}) = \gamma_p^-(a_{!\varphi}) = \perp$;
- $Poss(a_{??\varphi}) = \top$ and $SF(a_{??\varphi}) = \varphi$, and $\gamma_p^+(a_{??\varphi}) = \gamma_p^-(a_{??\varphi}) = \perp$.

Given a PALAT formula φ , we may in this way extract a simple basic action theory $\Sigma(\varphi)$ from φ . It can then be shown that φ is PALAT valid if and only if $\Sigma(\varphi)$ entails $\tau^{-1}(\varphi)$ in ES.

Acknowledgements

We would like to thank the JLC reviewers whose comments helped us to improve the paper.

Tiago de Lima's contribution was partially realized at the Eindhoven University of Technology, and founded by the research program Moral Responsibility in R&D Networks, supported by the Netherlands Organisation for Scientific Research (NWO), under grant number 360-20-160.

Hans van Ditmarsch is also affiliated to the University of Otago.

References

- Guillaume Aucher. *Perspectives on belief and change*. PhD thesis, Université de Toulouse, July 2008.
- Guillaume Aucher and Andreas Herzig. From DEL to EDL: exploring the power of converse events. In Khaled Mellouli, editor, *Proc. 9th Eur. Conf. on Symbolic and Quantitative Approaches to Reasoning with Uncertainty (ECSQARU 2007)*, volume 4724 of *LNCS*, pages 199–209. Springer Verlag, 2007.
- Guillaume Aucher and Andreas Herzig. Exploring the power of converse events. In Olivier Roy, Patrick Girard, and Mathieu Marion, editors, *Dynamic Epistemology: Contemporary Perspectives*, volume to appear of *Synthese Library*. Springer Verlag, 2010.
- Fahiem Bacchus, Joseph Halpern, and Hector Levesque. Reasoning about noisy sensors and effectors in the situation calculus. *Artificial Intelligence*, 111(1–2):171–208, 1999.
- Philippe Balbiani, Alexandru Baltag, Hans van Ditmarsch, Andreas Herzig, Tomohiro Hoshi, and Tiago de Lima. ‘Knowable’ as ‘known after an announcement’. *Review of Symbolic Logic*, 1(3):305–334, 2008.
- Philippe Balbiani, Hans van Ditmarsch, Andreas Herzig, and Tiago de Lima. Tableaux for public announcement logics. *Journal of Logic and Computation*, 20(2), 2010.
- Alexandru Baltag and Lawrence Moss. Logics for epistemic programs. *Synthese*, 139(2):165–224, 2004.
- Alexandru Baltag and Sonja Smets. From conditional probability to the logic of doxastic actions. In Dov Samet, editor, *Proc. 11th Conference on Theoretical Aspects of Rationality and Knowledge (TARK-2007)*, pages 52–61, 2007.
- Alexandru Baltag and Sonja Smets. Probabilistic dynamic belief revision. *Synthese*, 165(2):179–202, 2008.
- Alexandru Baltag and Sonja Smets. Group belief dynamics under iterated revision: fixed points and cycles of joint upgrades. In Hiroakira Ono, Makoto Kanazawa, and Ruy J. G. B. de Queiroz, editors, *Proc. 16th International Workshop on Logic, Language, Information and Computation, WoLLIC 2009, Tokyo, Japan, June 21-24, 2009*, volume 5514 of *Lecture Notes in Computer Science*, pages 41–50. Springer, 2009. ISBN 978-3-642-02260-9.
- Alexandru Baltag, Lawrence Moss, and Slawomir Solecki. The logic of common knowledge, public announcements, and private suspicions. In *Proceedings of the seventh Theoretical Aspects of Rationality and Knowledge conference (TARK)*, pages 43–46. Morgan Kaufmann Publishers Inc., 1998.
- Johan van Benthem. Dynamic logic for belief revision. *Journal of Applied Non-Classical Logics (special issue on belief revision and dynamic logic)*, 17(2):129–156, 2007.
- Patrick Blackburn, Jaap Kamps, and Maarten Marx. Situation calculus as hybrid logic: First steps. In P. Brazdil and A. Jorge, editors, *Progress in Artificial Intelligence*, number 2258 in *Lecture Notes in Computer Science*, pages 253–260. Springer Verlag, 2001.
- Robert Demolombe. Belief change: From situation calculus to modal logic. *Journal of Applied Non-Classical Logics*, 13(2):187–198, 2003.
- Robert Demolombe and Pilar Pozos Parra. Belief revision in the situation calculus without plausibility levels. In F. Esposito, Z. W. Ras, D. Malerba, and Semeraro G., editors, *Foundations of Intelligent Systems*, number 4203 in *LNAI*, pages 504–5013. Springer Verlag, 2006.

- Hans P. van Ditmarsch. Prolegomena to dynamic logic for belief revision. *Knowledge, Rationality & Action (Synthese)*, 147:229–275, 2005.
- Ronald Fagin, Joseph Halpern, Yoram Moses, and Moshe Vardi. *Reasoning about Knowledge*. The MIT Press, 1995.
- Melvin Fitting. *First-order modal logic*. Synthese Library. Kluwer Academic Publishers, 1998.
- Dov M. Gabbay, Valentin Shehtman, and Dmitriy Skvortsov. *Quantification in Nonclassical Logic, Volume 1*. Number 153 in Studies in Logic and the Foundations of Mathematics. Elsevier, 2009.
- Peter Gammie and Ronald van der Meyden. MCK: Model checking the logic of knowledge. In R. Alur and D. Peled, editors, *Proceedings of the 16th International Conference on Computer Aided Verification (CAV 2004)*, pages 479–483. Springer, 2004.
- James W. Garson. Quantification in modal logic. In Dov M. Gabbay and Franz Günthner, editors, *Handbook of Philosophical Logic*, volume II, pages 249–307. Reidel, 1984.
- Jelle Gerbrandy. *Bisimulations on Planet Kripke*. PhD thesis, ILLC, University of Amsterdam, 1999.
- Hojjat Ghaderi, Hector Levesque, and Yves Lespérance. A logical theory of coordination and joint ability. In *Proceedings of the Twenty-Second AAAI Conference on Artificial Intelligence*, pages 421–426. AAAI Press, 2007.
- Jaakko Hintikka. *Knowledge and Belief*. Cornell University Press, 1962.
- Ryan F. Kelly and Adrian R. Pearce. Complex epistemic modalities in the situation calculus. In *Proc. 11th Int. Conf. on Principles of Knowledge Representation and Reasoning (KR’2008)*, pages 611–620. AAAI Press, 2008.
- Barteld Kooi. Expressivity and completeness for public update logic via reduction axioms. *Journal of Applied Non-Classical Logics (special issue on belief revision and dynamic logic)*, 17(2):231–253, 2007.
- Gerhard Lakemeyer and Hector Levesque. Situations, si! Situation terms, no! In *Proceedings of the Ninth International Conference Principles of Knowledge Representation and Reasoning (KR)*, pages 516–526. AAAI Press, 2004.
- Gerhard Lakemeyer and Hector Levesque. Semantics for a useful fragment of the situation calculus. In *Proceedings of the Nineteenth International Joint Conference on Artificial Intelligence (IJCAI)*, pages 490–496. Professional Book Center, 2005.
- Hector Levesque, Raymond Reiter, Yves Lespérance, Fangzhen Lin, and Richard Scherl. GOLOG: A logic programming language for dynamic domains. *Journal of Logic Programming*, 31(1–2):59–83, 1997.
- Carsten Lutz. Complexity and succinctness of public announcement logic. In P. Stone and G. Weiss, editors, *Proceedings of the Fifth International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, pages 137–144, 2006.
- John McCarthy. Elaboration tolerance. In *Proceedings of Common Sense’98*, London, 1998. Available at <http://www-formal.stanford.edu/jmc/elaboration.html>.
- John-Jules Meyer and Wiebe van der Hoek. *Epistemic Logic for AI and Computer Science*. Number 41 in Cambridge Tracts in Theoretical Computer Science. Cambridge University Press, 1995.

- G.E. Moore. A reply to my critics. In P.A. Schilpp, editor, *The Philosophy of G.E. Moore*, pages 535–677. Northwestern University, Evanston IL, 1942. The Library of Living Philosophers (volume 4).
- Jan Plaza. Logics of public communications. In M. L. Emrich, M. Hadzikadic, M. S. Pfeifer, and Z. W. Ras, editors, *Proceedings of the Fourth International Symposium on Methodologies for Intelligent Systems (ISMIS)*, pages 201–216, 1989.
- Raymond Reiter. The frame problem in the situation calculus: A simple solution (sometimes) and a completeness result for goal regression. In V. Lifschitz, editor, *Papers in Honor of John McCarthy*, pages 359–380. Academic Press Professional Inc., 1991.
- Raymond Reiter. *Knowledge in Action: Logical Foundations for Specifying and Implementing Dynamical Systems*. The MIT Press, 2001a.
- Raymond Reiter. On knowledge-based programming with sensing in the situation calculus. *ACM Transactions on Computational Logic*, pages 433–437, 2001b.
- Richard Scherl and Hector Levesque. The frame problem and knowledge-producing actions. In *Proceedings of the Eleventh National Conference on Artificial Intelligence (AAAI)*, pages 689–695. The AAAI Press, 1993.
- Richard Scherl and Hector Levesque. Knowledge, action and the frame problem. *Artificial Intelligence*, 144(1–2):1–39, 2003.
- Steven Shapiro and Maurice Pagnucco. Iterated belief change and exogenous actions in the situation calculus. In R. Lopez de Mntaras and L. Saitta, editors, *Proc. 16th European Conference on Artificial Intelligence (ECAI-04)*, pages 878–882. IOS Press, 2004.
- Steven Shapiro, Maurice Pagnucco, Yves Lespérance, and Hector J. Levesque. Iterated belief change in the situation calculus. In *Proceedings of the Seventh International Conference Principles of Knowledge Representation and Reasoning (KR)*, pages 527–538, 2000.
- Johan van Benthem. “one is a lonely number”: Logic and communication. In Z. Chatzidakis, P. Koepke, and W. Pohlers, editors, *Logic Colloquium’02*, volume 27 of *Lecture Notes in Logic*, pages 96–129. ASL & A.K. Peters, 2006.
- Johan van Benthem. Modal logic meets situation calculus. Technical Report PP-2007-04, ILLC, University of Amsterdam, 2007.
- Johan van Benthem, Jan van Eijck, and Barteld Kooi. Logics of communication and change. *Information and Computation*, 204(11):1620–1662, 2006.
- Hans van Ditmarsch, Wiebe van der Hoek, and Barteld Kooi. Dynamic epistemic logic with assignment. In F. Dignum, V. Dignum, S. Koenig, S. Kraus, M. Singh, and M. Wooldridge, editors, *Proceedings of the Fourth International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, pages 141–148. ACM, 2005.
- Hans van Ditmarsch, Andreas Herzig, and Tiago de Lima. Optimal regression for reasoning about knowledge and actions. In *Proceedings of the Twenty-Second AAAI Conference on Artificial Intelligence*, pages 1070–1075. AAAI Press, 2007a.
- Hans van Ditmarsch, Wiebe van der Hoek, and Barteld Kooi. *Dynamic Epistemic Logic*, volume 337 of *Synthese Library*. Springer, 2007b.
- Jan van Eijck. Dynamic epistemic modelling. Technical report, Centrum voor Wiskunde en Informatica, Amsterdam, 2004. CWI Report SEN-E0424.

Contents

1	Introduction	2
2	Background: Lakemeyer&Levesque's logic ES	3
2.1	Syntax of ES	3
2.2	Semantics of ES	4
2.3	Explicit successor state axioms	5
2.4	Simple basic action theories	6
3	Public announcement logic with assignments and tests PALAT	8
3.1	Background: epistemic logic EL	9
3.2	Syntax of PALAT	9
3.3	Semantics of PALAT	10
4	Translation from ES to PALAT	11
4.1	From actions to assignments	11
4.2	The translation	12
4.3	Correctness	14
4.4	Polynomial length	17
5	Perspectives: an extension of ES by event models	19
6	Conclusion	21