A correspondence between two different solutions to the projection task with sensing

Mikhail Soutchanski
Dept. of Computer Science
University of Toronto
Toronto, ON M5S 3H5
mes@cs.toronto.edu

Abstract

The paper compares an approach to reasoning about action, knowledge, and sensing that relies on epistemic fluent (and reduces reasoning about knowledge to provability) to another approach that does not rely on epistemic fluents and suggests to account for sense actions directly in successor state axioms. Under the assumptions that the theory of the initial situation consists of only sentences representing beliefs of an agent and all sense actions deliver new information, it is shown that these approaches are essentially equivalent. The second approach is used in a high level program that controls a mobile robot.

1 Introduction

The large scale formalization of common sense reasoning is a long standing but still elusive goal of the AI research [14; 3]. Some of the challenges that the knowledge representation community is facing are characterized in [5]. However, not all areas of common sense reasoning resist in equal degree attempts of logical and computationally tractable formalization. The most advanced results have been achieved in the area of reasoning about actions, knowledge and sensing, where a predicate logic language called the situation calculus provides a well established, clear and easily extendible framework [4; 11; 20] (to mention just a few). In particular, [21] argues that the solution of the frame problem (what logical properties of the world change and what properties persist when an agent executed an action that intuitively effects only a limited number of properties) can be conveniently formulated in the situation calculus. Furthermore, [22] extends this solution to reasoning about agents' knowledge by elaborating an earlier proposal by [16] that an accessibility relation between possible worlds (a model-theoretic construct in modal logics of knowledge and belief) can be represented in a situation calculus logical theory[1] by a binary predicate $K(s, s')$ relating two 'epistemically alternative' situations s and s'; this predicate is called epistemic fluent. The successor state axiom for this fluent proposed in [22] characterizes how knowledge changes from the current situation to a successor situation resulting from execution of an action in the current situation. This axiom solves the frame problem for knowledge: only sense actions may effect the accessibility between situations, more specifically, they shrink the accessibility relation by eliminating those situations which do not 'agree' with the results of sensing, where sense actions are understood in very general terms as actions determining a truth value of an arbitrary situation calculus sentence. The generality and elegance of this approach to reasoning about knowledge and actions inspired many subsequent developments [1; 9; 10; 24], but at the same time this generality may hinder practical computer implementations. In certain application domains alternative approaches to reasoning about knowledge and actions may facilitate implementations and still be sound with respect to the original K-based general specification. For example, in computer animation, [8] incorporates interval arithmetic into situation calculus to provide the representation of an epistemic uncertainty of agents and of the dynamics of this uncertainty when agents obtain a new sensory input. As another example, reduction of knowledge to provability in [20] serves as a sound foundation of knowledge-based programming in an epistemic extension of the logic based high level programming language Golog. Recently, [6; 18] proposed alternative approaches to formalization of reasoning about actions, knowledge and sensing that do not rely on epistemic fluent $K(s, s')$. These proposals led to successful implementation of robotics control systems [17; 12] capable of updating their model of the world by information gathered from sensors. Our own work [25] on implementation of decision-theoretic enrichment of Golog also indicates that in certain application domains it is sufficient to consider a less general set of sense actions and to integrate reasoning about physical actions and sense actions without the epistemic fluent; a mild modification of successor state axioms that provide a solution to the frame problem for physical actions suffices in the case when sense actions can determine truth values of only primitive properties, but not of arbitrary formulas. However, a correspondence between two different approaches: one, that relies on the epistemic fluent and reduces reasoning about knowledge to provability [20], and another, that does not appeal to epistemic fluent, but suggests an appropriate modification of successor state axioms, - remained unexplored. Moreover, were this correspondence established, it would serve as an argument in favor of correctness of the axioms modification approach with respect to K-based specification. Our paper aims to fill this gap and provide a correspondence between two aforementioned approaches. A formal comparison with other alternative approaches [2; 6;

[1] We use the word 'theory' in a technical sense as a conjunction of well-formed logical formulas.

18; 23; 26] remains a topic for further research. Besides giving a formal comparison, this research also sheds a light on a problem of proper interpretation of sensory information that is inconsistent with information entailed from the current model of the world. Consequently, a further research in this direction may result in a rich situation calculus based framework that will integrate reasoning about actions and sensing with the approach to diagnosis advocated in [15]. In addition, when an agent executes a plan (or a Golog program), it may need to check sensors regularly and, if necessary, to compute corrections of the original plan. The approach to dealing with sense actions that relies on modification of successor state axioms can be useful for development of a general realisitc execution monitoring framework that does not assume that the feedback from the environment is provided in terms of actions executed by other agents ([7] assumes that all exogenous actions are observable).

2 The situation calculus

We look to an expressive and well studied knowledge representation framework for modeling processes called the situation calculus. The situation calculus is a predicate logic language for representing and reasoning about actions and properties of dynamically changing worlds. It is employed to model continuous processes and physical systems and to provide the foundations for a logic based language Golog [20; 13]. The version of the situation calculus that we use here has been described in [13; 20], and elsewhere. We will not go over the language here except to note the basic components. All changes to the world are the result of named *actions*. A sequence of actions (history) is represented by a term called a *situation*. The constant S_0 is used to denote the *initial situation*, namely the empty history. Non-empty histories are constructed using a distinguished binary function symbol do; $do(\alpha(x), s)$ denotes the successor situation to s resulting from performing the action denoted by term $\alpha(x)$. It is important to emphasize that each situation represents a sequence of actions (a plan); this feature of the situation calculus allows to construct plans by theorem proving [13]. There is a special predicate $Poss(\alpha(x), s)$ used to state that action $\alpha(x)$ is possible in situation s and a special predicate $s_1 \sqsubset s_2$ that defines a partial order on situations. Relations whose truth values vary from situation to situation are called *relational fluents*. They are denoted by predicate symbols taking a situation term as their last argument. Similarly, functions whose values vary from situation to situation are called *functional fluents* and are denoted by function symbols taking a situation term as their last argument.

To axiomatize the primitive actions and fluents of a domain of application, one must provide a set of axioms that constitute a basic action theory.

3 The language and basic action theories

In this paper, we consider a version of the situation calculus with the following two simplifying assumptions: 1) the language has no functional fluents, but non-fluent function symbols are permitted; 2) the language has no non-fluent predicate symbols except for the equality predicate, the predicate symbol $\sqsubset$ (it occurs in the foundational axioms to define the order between situations) and the predicate symbol $Poss$ (it occurs in precondition axioms to characterize actions possible in certain situations). Both assumptions does not lead to any loss of expressivity. First, instead of a functional fluent, the axiomatizer can use a corresponding relational fluent (but should enforce via an appropriate axiomatization that the object argument of the fluent must always exist and be unique). Second, a non-fluent predicate symbol can be replaced by a relational fluent (the axiomatizer should add an obvious successor state axiom saying that the truth value of such a fluent persists from situation to situation no matter what action has been performed). We introduce these restrictions on the language to conform with assumptions made in ([20], Chapter 11). [2]

Following [16; 22] (see also [13; 20]), we represent the fact that situation s is one of the situations that the agent in situation σ considers possible by $K(s, \sigma)$ (where K is a binary predicate symbol that corresponds to an accessibility relation in modal logic of knowledge), the fact that the agent knows ϕ in situation σ by an abbreviation $\mathbf{Knows}(\phi, \sigma) \stackrel{def}{=} (\forall s).\, K(s, \sigma) \supset \phi[s]$ (where $\phi[s]$ denotes that situation calculus formula obtained from ϕ by restoring the situation argument to its rightful place), the fact that an agent knows whether ϕ is true in situation σ by an abbreviation $\mathbf{KWhether}(\phi, \sigma) \stackrel{def}{=} \mathbf{Knows}(\phi, \sigma) \vee \mathbf{Knows}(\neg\phi, \sigma)$, the fact that an agent knows the referent of the term t by an abbreviation $\mathbf{KRef}(t, \sigma) \stackrel{def}{=} (\exists x)\mathbf{Knows}(x = t, \sigma)$. In the sequel, we will also need the definition of an *objective situation-suppressed sentence* and the definition of a *subjective sentence*; both definitions are introduced in [20]. An objective situation-suppressed sentence is an expression without any free variables composed from situation suppressed relational fluent atoms and expressions $t_1 = t_2$ by boolean connectives and quantifiers; in a nutshell, objective situation-suppressed expressions are expressions that do not mention K and have no situational argument. For the sake of brevity, these sentences are called simply objective. A *subjective sentence about* a ground situation term σ is the sentence of the form $\mathbf{Knows}(\beta, \sigma)$, where β is objective, or it has the form $\neg W$, where W is a subjective sentence about σ, or it has the form $W_1 \vee W_2$, where W_1 and W_2 are subjective sentences about σ. In a nutshell, subjective sentences are boolean combinations of statements about what the agent knows or does not know about the objective world.

3.1 Sense Actions

Because we are interested in establishing the correspondence between representations of sensing in [20] and in [25], we introduce here two mutually exclusive sets of sense actions. In the first set (see [20]), all sense actions have the form $sense_\psi(\vec{x})$, where $\psi(\vec{x})$ is an objective situation-suppressed expression. Below we pay a special attention to a strict subset of this set; we consider all sense actions where $\psi(\vec{x})$ is a

[2] The first assumption helped to simplify treatment of knowledge-based programs with sensing in [20]: because $read$ knowledge-producing actions sense the values of functional fluents they can be ignored if there are no functional fluents. The second assumption was introduced there for purely technical reasons.

situation-suppressed relational fluent symbol: $sense_{F_1}(\vec{x}_1)$, ..., $sense_{F_m}(\vec{x}_m)$, where m is the number of fluents. The second set consists of just one binary action term $sense(q, v)$, where q is what to sense, v is term representing a run time value returned by the sensors. In [25], the term representing this sense action is ternary with time as the third argument, but because we consider here actions without temporal argument (to simplify our presentation), in this paper we use a binary action term. In the sequel, we consider only those substitutions of this term which have a situation-suppressed relational fluent atom as their first argument and a truth value constant as their second argument. So, we consider $2m$ distinct substitutions: $sense(F_1(\vec{x}_1), v)$,...,$sense(F_m(\vec{x}_m), v)$, where $v \in \{\text{TRUE}, \text{FALSE}\}$.[3]

3.2 Basic Action Theories

We consider two types of basic action theories:

$$\mathcal{D}^K = \mathcal{D}^K_{ss} \cup \mathcal{D}^K_{ap} \cup \mathcal{D}^K_{S_0} \cup \mathcal{K}_{Init} \cup \Sigma^K \cup \mathcal{D}_{una}$$
$$\mathcal{D} = \mathcal{D}_{ss} \cup \mathcal{D}_{ap} \cup \mathcal{D}_{S_0} \cup \Sigma \cup \mathcal{D}_{una}$$

The differences and correspondences between $\mathcal{D}^K$ and $\mathcal{D}$ are the following. Sense actions $sense(F_i(\vec{x}_i), v)$ may occur only in $\mathcal{D}$ and sense actions $sense_{F_i}(\vec{x}_i)$ may occur only in $\mathcal{D}^K$; a physical action $A(\vec{x})$ occurs in $\mathcal{D}$ if and only if it occurs in $\mathcal{D}^K$.

$\mathcal{D}^K_{ss}$ is a set of successor state axioms for relational fluents, including an axiom (1) for K:

$$\begin{aligned} K(s, do(a, \sigma)) \equiv\ & (\exists s^*).s = do(a, s^*) \wedge K(s^*, \sigma)\ \wedge \\ & (\forall \vec{x}_1)[a = sense_{\psi_1}(\vec{x}_1) \supset \psi_1(\vec{x}_1, s^*) \equiv \psi_1(\vec{x}_1, \sigma)] \\ & \wedge \cdots \wedge \\ & (\forall \vec{x}_m)[a = sense_{\psi_m}(\vec{x}_m) \supset \psi_m(\vec{x}_m, s^*) \equiv \psi_m(\vec{x}_m, \sigma)] \end{aligned} \quad (1)$$

All other successor state axioms in $\mathcal{D}^K_{ss}$ mention physical actions only and have the following form:

$$F(\vec{x}, do(a, s)) \equiv \gamma^+_F(\vec{x}, a, s) \vee F(\vec{x}, s) \wedge \neg\gamma^-_F(\vec{x}, a, s), \quad (2)$$

where $\gamma^+_F(\vec{x}, a, s)$ and $\gamma^-_F(\vec{x}, a, s)$ are formulas with free variables among $\vec{x}$, s, and whose only situation term is s. Successor state axioms embody the solution to the frame problem of Reiter [21]: $\gamma^+_F(\vec{x}, a, s)$ (and $\gamma^-_F(\vec{x}, a, s)$, respectively) characterize all possible conditions under which a physical action a can cause F to become true (false, respectively) in the next situation.

A set $\mathcal{D}_{ss}$ does not include a successor state axiom for K. Recall that we have sense actions that can determine truth values of fluents only. All successor state axioms in $\mathcal{D}_{ss}$ mention both physical and sense actions (recall that only sense actions $sense(F_i(\vec{x}_i), v)$ may occur in $\mathcal{D}_{ss}$). Every axiom is obtained by the following transformation from the corresponding successor state axiom (2) in $\mathcal{D}^K_{ss}$:

$$\begin{aligned} & F(\vec{x}, do(a, s)) \equiv \\ & \qquad \gamma^+_F(\vec{x}, a, s)\ \vee\ a = sense(F(\vec{x}), \text{TRUE})\ \vee \\ & F(\vec{x}, s) \wedge (\ \neg\gamma^-_F(\vec{x}, a, s) \wedge a \neq sense(F(\vec{x}), \text{FALSE})\ ). \end{aligned} \quad (3)$$

[3]Thus, we assume that all fluents are directly observable. We would have less sense actions if some fluents were not directly observable, i.e., data about their truth values were not available to sensors.

In other words, the fluent $F(\vec{x}, do(a, s))$ is true iff a is a physical action and the condition $\gamma^+_F(\vec{x}, a, s)$ holds or a is a sense action and sensors tell that the fluent is true or the fluent was true in s and neither $\gamma^-_F(\vec{x}, a, s)$ is true nor a is a sense action determining that the fluent is false.

$\mathcal{D}_{ap}$ is a set of action precondition axioms, one for each primitive action $A(\vec{x})$, having the syntactic form

$$Poss(A(\vec{x}), s) \equiv \Pi_A(\vec{x}, s),$$

where $\Pi_A(\vec{x}, s)$ is a formula with free variables among $\vec{x}$, s, and whose only situation term is s. Action precondition axioms characterize (via the formula $\Pi_A(\vec{x}, s)$) the conditions under which it is possible to execute action $A(\vec{x})$ in situation s. If $A(\vec{x})$ is a sense action, then the right hand side of this axiom is the constant *true*: every sense action $sense(F_i(\vec{x}_i), v)$ is possible in any situation s.

$\mathcal{D}^K_{ap}$ is also a set of action precondition axioms. These coincide with axioms in $\mathcal{D}_{ap}$ for physical actions. For every sense action $sense_{F_i}(\vec{x}_i)$, the right hand side of its action precondition axiom is *true*. Recall that sense actions from another set never occur in axioms $\mathcal{D}^K_{ap}$.

$\mathcal{D}^K_{S_0}$ is a set of first order sentences whose only situation terms are S_0, and they are all of the form **Knows**(κ_i, S_0), $i = 1, \ldots, n$, where each κ_i is objective. In other words, the initial database consists exclusively of sentences declaring what the agent knows about the world she inhabits, but there are no objective sentences, and there are no sentences declaring what the agent knows about what she knows. So we can simply suppose that $\mathcal{D}^K_{S_0}$ consists of a single sentence of the form **Knows**(κ, S_0), where κ is objective (possibly, it is a conjunction of other objective sentences). Then, $\mathcal{D}_{S_0} = \kappa_1[S_0] \wedge \cdots \wedge \kappa_n[S_0]$, i.e., we can simply suppose that a special class of initial databases consists of a single objective sentence of the form $\kappa[S_0]$. Note how $\mathcal{D}_{S_0}$ and $\mathcal{D}^K_{S_0}$ correspond to each other and note that we consider in this paper a special class of initial databases (because this class is considered in [20], Chapter 11).

The accessibility relation is reflexive in initial situations if $(\forall s).Init(s) \supset K(s, s)$, symmetric in initial situations if $(\forall s_1, s_2).Init(s_1) \wedge Init(s_2) \supset [K(s_1, s_2) \supset K(s_2, s_1)]$. The solution (1) to the frame problem for K guarantees that provided the accessibility relation is reflexive, symmetric (transitive, Euclidean) in all initial situations, then it will have the same property in all situations [20]. $\mathcal{K}_{Init}$ consists of the axiom about reflexivity of $K(s, s')$ in all initial situations and any subset of other initial accessibility axioms. One can prove the following lemma:

Lemma 3.1: *When $\mathcal{K}_{Init}$ includes the reflexivity axiom,*

$$\mathcal{D}^{\mathcal{K}} \models (\forall \vec{y}, s).\textbf{Knows}(\psi(\vec{y}), do(sense_\psi(\vec{y}), s)) \equiv \psi(\vec{y}, s),$$
$$\mathcal{D}^{\mathcal{K}} \models (\forall \vec{y}, s).\textbf{Knows}(\neg\psi(\vec{y}), do(sense_\psi(\vec{y}), s)) \equiv \neg\psi(\vec{y}, s).$$

Σ are the standard foundational axioms:

$$do(a_1, s_1) = do(a_2, s_2) \supset a_1 = a_2 \wedge s_1 = s_2, \quad (4)$$

$$(\forall P).P(S_0) \wedge (\forall a, s)[P(s) \supset P(do(a, s))] \supset (\forall s)P(s), \quad (5)$$

$$\neg s \sqsubset S_0, \quad (6)$$

$$s \sqsubset do(a, s') \equiv s \sqsubseteq s'. \tag{7}$$

Let $Init(s) \stackrel{def}{=} \neg(\exists a, s')s = do(a, s')$, then Σ^K is the union of (4), (6), (7), (8) and (9):

$$(\forall P).(\forall s)[Init(s) \supset P(s)] \wedge \\ (\forall a, s)[P(s) \supset P(do(a, s))] \supset (\forall s)P(s). \tag{8}$$

$$K(s, s') \supset [Init(s) \equiv Init(s')]. \tag{9}$$

$\mathcal{D}_{una}$ is the set of unique names axioms for actions. These will include unique names axioms for sense actions.

4 A blocks world example

In this section, we consider an example of reasoning about effects of sequences of actions using theories $\mathcal{D}$ and $\mathcal{D}^{\mathcal{K}}$. In particular, we rely on a standard axiomatization of the blocks world with successor state and action precondition axioms, but we add an agent argument to all physical action terms. Assume that there is a robot who wants to put all blocks on the table (initially, there may be some towers of blocks) and possibly there are several other agents. The robot is not able to observe occurrences of their actions directly, but the robot can determine effects of these actions by doing sense actions. We leave sense actions without an agent argument because we attribute them to the robot only. We use the following function and predicate constants.

Fluents

- $On(x, y, s)$: Block x is on block y, in situation s.
- $Clear(x, s)$: Block x has no other blocks on top of it, in situation s.
- $Ontable(x, s)$: Block x is on the table in s.

Actions

- $move(g, x, y)$: Agent g moves block x onto block y, provided both are clear.
- $moveToTable(g, x)$: Agent g moves block x onto the table, provided x is clear and is not on the table.
- $sense_{Ontable}(x)$: Sense whether block x is on the table.
- $sense_{Clear}(x)$: Sense whether block x is clear.
- $sense_{On}(x, y)$: Sense whether block x is on top of y. The outcome of all aforementioned sense actions can be either "yes" or "no".
- $sense(Ontable(x), v)$: Connect to a sensor, if block x is on the table, then v has the value TRUE, if not - v is FALSE.
- $sense(Clear(x), v)$: Connect to a sensor, if block x is clear, then the value v is TRUE, otherwise it is FALSE.
- $sense(On(x, y), v)$: Connect to a sensor, if block x is on top of y, then v is TRUE, otherwise it is FALSE.

We start our example with considering the theory $\mathcal{D}^{\mathcal{K}}$ and later in this section we consider the theory $\mathcal{D}$. The theory $\mathcal{D}^{\mathcal{K}}$ includes the following axioms.

Successor state axioms (D_{ss}^K)

$$On(x, y, do(a, s)) \equiv (\exists g).\, a = move(g, x, y) \vee \\ On(x, y, s) \wedge \neg(\exists g).\, a = moveToTable(g, x) \wedge \\ \neg(\exists g, z)\, a = move(g, x, z).$$

$$Ontable(x, do(a, s)) \equiv (\exists g).a = moveToTable(g, x) \vee \\ Ontable(x, s) \wedge \neg(\exists g, y).\, a = move(g, x, y).$$

$$Clear(x, do(a, s)) \equiv (\exists g, y, z).[\, a = move(g, y, z) \wedge z \neq x \vee \\ a = moveToTable(g, y))\,] \wedge On(y, x, s) \vee \\ Clear(x, s) \wedge \neg(\exists g, w)\, a = move(g, w, x).$$

Note that these axioms have no occurrences of sense actions. The version of the axiom (1) is also included in D_{ss}^K: it mentions only actions $sense_{Ontable}(x)$, $sense_{Clear}(x)$, $sense_{On}(x, y)$.

Action precondition axioms (D_{ap}^K)

$$Poss(move(g, x, y), s) \equiv Clear(x, s) \wedge \\ Clear(y, s) \wedge x \neq y.$$

$$Poss(moveToTable(g, x), s) \equiv Clear(x, s) \wedge \\ \neg Ontable(x, s).$$

In this example, all sense actions are always possible, i.e., the right hand sides of precondition axioms for sense actions are *true*.

Axioms $\mathcal{D}^{\mathcal{K}}$ also include the standard set of unique name axioms for actions.

Let the initial situation be such that $Knows(Clear(B), S_0)$, nothing else is known about which blocks are where and whether they are clear or not. In addition, there are state constraints associated with the blocks world, and we must suppose the robot knows that these constraints hold initially:

$$\mathbf{Knows}((\forall x, y).On(x, y) \supset \neg On(y, x), S_0),$$
$$\mathbf{Knows}((\forall x, y, z).On(y, x) \wedge On(z, x) \supset y = z, S_0),$$
$$\mathbf{Knows}((\forall x, y, z).On(x, y) \wedge On(x, z) \supset y = z, S_0).$$

The conjunction of axioms about the initial situation is $\mathbf{Knows}(\kappa, S_0)$. Because this set of axioms is weak (it does not even entail whether block B is on the table or not), we are intended to augment $\mathcal{D}^{\mathcal{K}}$ with axioms representing sensory information whenever the robot does one of sense actions.

Let the robot do the following sequence of actions. First, the robot senses whether B is on the table or not; suppose, that B is not on the table. In this case, because it is not known initially where B is located we want to add the axiom $Knows(\neg Ontable(B), do(sense_{Ontable}(B), S_0))$ to $\mathcal{D}^{\mathcal{K}}$: this axiom will supplement axioms about the initial situation. In the sequel, $S_N = do(sense_{Ontable}(B), S_0))$. Second, the robot moves B to the table. We are interested in solving the following entailment task:

$$\mathcal{D}^{\mathcal{K}} \wedge \mathbf{Knows}(\neg Ontable(B), S_N) \\ \models \mathbf{Knows}(Ontable(B), do(moveToTable(R, B), S_N)).$$

One can prove that the formula at the right hand side is entailed by axioms on the left hand side. Now, suppose that another agent moves block B from the table onto another block and suppose that the robot cannot observe the occurrence of this action; in other words, from the robot's perspective the current situation is $do(moveToTable(R, B), S_N)$ (we denote this term by S_M). At this moment, if the robot does the sense action $sense_{Ontable}(B)$, then the outcome from sensors is 'no'. One can prove that the conjunction $\mathcal{D}^{\mathcal{K}} \wedge$ **Knows**$(\neg Ontable(B), S_N)$ is inconsistent with **Knows**$(\neg Ontable(B), do(sense_{Ontable}(B), S_M))$ and for this reason we cannot simply add the last formula to represent the most recent outcome from sensors. Thus, if exogenous actions are not observable, we cannot add axioms corresponding to sensory information without checking for consistency of these axioms with the previously obtained set of axioms.

Let us consider now the theory $\mathcal{D}$. It includes the following set $\mathcal{D}_{ss}$ of successor state axioms.

$$\begin{array}{l} On(x, y, do(a, s)) \equiv (\exists g).\, a = move(g, x, y) \vee \\ \quad a = sense(On(x, y), \text{TRUE}) \vee \\ \quad On(x, y, s) \wedge \neg(\exists g).\, a = moveToTable(g, x) \wedge \\ \quad \neg(\exists g, z)\, a = move(g, x, z) \wedge a \neq sense(On(x, y), \text{FALSE}). \end{array}$$

$$\begin{array}{l} Ontable(x, do(a, s)) \equiv (\exists g).a = moveToTable(g, x) \vee \\ \quad a = sense(Ontable(x), \text{TRUE}) \vee \\ \quad Ontable(x, s) \wedge \neg(\exists g, y).\, a = move(g, x, y) \wedge \\ \quad a \neq sense(Ontable(x), \text{FALSE}). \end{array}$$

$$\begin{array}{l} Clear(x, do(a, s)) \equiv (\exists g, y, z).[\, a = move(g, y, z) \wedge z \neq x \vee \\ \quad a = moveToTable(g, y))\,] \wedge On(y, x, s) \vee \\ \quad a = sense(Clear(x), \text{TRUE}) \vee \\ \quad Clear(x, s) \wedge \neg(\exists g, w)\, a = move(g, w, x) \wedge \\ \quad a \neq sense(Clear(x), \text{FALSE}). \end{array}$$

The set $\mathcal{D}_{ss}$ does not include the axiom (1). The set of precondition axioms $\mathcal{D}_{ap}$ coincides with $\mathcal{D}^{K}_{ap}$; D also includes the set of unique names axioms for actions. Let D_{S_0} be $Clear(B, S_0)$. Consider again the same sequence of actions and again suppose that initially B is not on the table. First, the robot does $sense(Ontable(B), \text{FALSE})$; we denote $do(sense(Ontable(B), \text{FALSE}), S_0)$ by S'_n. Second, the robot moves B to the table. Clearly,

$$\mathcal{D} \models Ontable(B, do(moveToTable(R, B), S'_n)).$$

In the sequel, we denote $do(moveToTable(R, B), S'_n)$ by S'_m. Let someone move B from the table to a different place, this action be not observable, and after that the robot do $sense(Ontable(B), \text{FALSE})$ in the situation S'_m. Then,

$$\mathcal{D} \models \neg Ontable(B, do(sense(Ontable(B), \text{FALSE}), S'_m)).$$

Thus, we can obtain correct entailments from $\mathcal{D}$ even if exogenous actions are not observable.[4] In the subsequent sections, we consider only the case when axioms corresponding to new sensory information are consistent with previously obtained set of axioms: we make this simplifying assumption because we are primarily interested in correspondence between reasoning about effects of actions in $\mathcal{D}^{\mathcal{K}}$ and reasoning about effects of actions in $\mathcal{D}$.

5 The forward projection task

Our main goal is to find the connection between solutions to the forward projection task in two different frameworks outlined above. First, we consider this task when $\mathcal{D}$ is the domain axiomatization and sense actions have the form $sense(F_i(\vec{x}), v_i)$. Second, we consider this task when the domain axiomatization is $\mathcal{D}^K$ and sense actions have the form $sense_{F_i}(\vec{x})$.

Informally, the forward projection task is this: given a sequence of ground action terms, and a formula ϕ, determine whether ϕ is true in the situation resulting from performing these actions. This task can be easily formulated in the situation calculus. Let $\mathcal{T}$ be a basic action theory of the domain and let $\phi(s)$ be a situation calculus formula that has the free variable s as the only situational argument. Suppose we are given a ground situational term S that mentions physical actions only. The forward projection task is to determine whether

$$\mathcal{T} \models \phi(S).$$

In the case when $\mathcal{T}$ is a domain theory without knowledge and sensing, regression [19; 20] is a sound and computationally efficient way of solving the forward projection task for a large interesting class of formulas called regressable [5]: these formulas occur naturally in planning and database query evaluation. We do not go here into details on how to use the regression operator $\mathcal{R}$ to solve the task, but recall only the intuition expressed in the definition of the regression operator. Suppose that $F(\vec{x}, do(a, s)) \equiv \Psi(\vec{x}, a, s)$ is a successor state axiom characterizing the relational fluent atom F, where $\Psi(\vec{x}, a, s)$ is a situation calculus formula. If $\phi(do(A, S))$ mentions F, we can determine a logically equivalent formula ϕ' by substituting $\Psi(\vec{x}, A, S)$ for $F(\vec{x}, do(A, S))$ in ϕ. Because the formula ϕ' has no occurrences of $F(\vec{x}, do(A, S))$ involving the complex situation term $do(A, S)$, this formula is simpler than the original formula $\phi(do(A, S))$. Moreover, the same operation can be repeated with other fluents in ϕ'. Similarly, the regression operator eliminates $Poss$ atoms in favor of their definitions as given by action precondition axioms. After repeatedly applying the regression operator $\mathcal{R}$ a necessary number of times, the original projection task reduces to first order theorem proving in the initial database $\mathcal{T}_{S_0}$ together with unique names axioms for actions T_{una}:

Theorem 5.1: (The Regression Theorem, [20; 19]) *Suppose that $\mathcal{T}$ is a basic action theory (without knowledge), and that $\phi(S)$ is a regressable sentence. Then*

$$\mathcal{T} \models \phi(S) \quad \textit{iff} \quad \mathcal{T}_{una} \cup \mathcal{T}_{S_0} \models \mathcal{R}_{\mathcal{T}}[\phi(S)]$$

Note that $\mathcal{R}_{\mathcal{T}}[\phi(S)]$ can mention only situation independent atoms and fluent atoms whose only situation argument is S_0.

[4] The last entailment can be considered as indication that the actual situation just before doing the second sense action is different from S'_m, but the task of determining what happened between S'_m and the moment when the robot did the second sense action remains outside the scopes of this paper (solution of this task is necessary to compute the situation term that includes occurrences of exogenous actions). The approach to solving diagnostic problems suggested in [15] can be used to compute actual situation terms.

[5] Let $\phi(s)$ be a situation calculus formula that has the free variable s as the only situational argument, does not mention the predicate symbol $\sqsubset$, nor does it mention equalities between situation terms and whenever it mentions the predicate $Poss(a, s)$, the first argument is a ground action term. Then, $\phi(s)$ is a regressable formula.

Observe also that the regression operator is defined relative to the set of successor state and precondition axioms in $\mathcal{T}$: if one varies these axioms, the result of regression varies as well.

5.1 The theory $\mathcal{D}$

Let a basic action theory $\mathcal{T}$ has the form $\mathcal{D}$. It is surprisingly straightforward to use regression in this setting to solve the forward projection task if $\phi(S_1)$ is an objective regressable sentence and we are given a ground situation term S_1 that may mention both physical and sensing actions (recall that in this paper the second argument of all sense actions mentioned in $\mathcal{D}$ is either the constant TRUE or the constant FALSE). No modifications of the regression operator are required and the regression theorem 5.1 applies also in this setting:

Theorem 5.2: (The Regression Theorem in $\mathcal{D}$)
Suppose that $\mathcal{D}$ is a basic action theory, and that $\phi(S_1)$ is a regressable sentence that does not mention K. Then
$\mathcal{D} \models \phi(S_1)$ *iff* $\mathcal{D}_{una} \cup \mathcal{D}_{S_0} \models \mathcal{R}_{\mathcal{D}}[\phi(S_1)]$

The subscript in $\mathcal{R}_{\mathcal{D}}$ indicates that the regression operator depends on $\mathcal{D}$; in particular, the regression operator eliminates all occurrences (if any) of fluents with complex situation terms $do(sense(F_i, V), S')$ using axioms (3).

Note the price that we pay to enjoy this solution: no occurrences of **Knows** are permitted in the formula $\phi(S)$. Hence, if an agent needs to evaluate a formula that mixes objective and subjective sentences, the theorem 5.2 cannot be applied. However, as long as the agent needs to reason only about the world, but does not need to reflect about knowledge of the world that she possesses, this theorem remains useful.

5.2 The theory $\mathcal{D}^K$

Let a basic action theory $\mathcal{T}$ has the form $\mathcal{D}^K$. Let ground situation term S mention both physical actions $A(\vec{Y})$ and sense actions $sense_{F_i}(\vec{X})$. Assume that all sense actions in S provide some useful new information, i.e., for every sense action $sense_{F_i}(\vec{X})$ in S we have: $\mathcal{D}^K \not\models$ **Knows**$(F_i(\vec{X}), do(sense_{F_i}(\vec{X}), S))$, $\mathcal{D}^K \not\models$ **Knows**$(\neg F_i(\vec{X}), do(sense_{F_i}(\vec{X}), S))$ (every sense action may have only two outcomes: either it determines that a fluent is true, or it determines that a fluent is false). Given that the knowledge about fluents can be axiomatized only in $\mathcal{D}^K_{S_0}$, this assumption amounts to assuming that $\mathcal{D}^K_{S_0}$ includes neither **Knows**$(F_i(\vec{X}), S_0)$, nor **Knows**$(\neg F_i(\vec{X}), S_0)$. Under this reasonable assumption, we no longer can solve the projection problem for a formula $\phi(do(sense_{F_i}(\vec{X}), S))$ if ϕ includes an atom **Knows** about the fluent F_i: our axioms are not strong enough to entail whether the agent knows about F_i or not. To solve the projection problem we need extension of our basic action theory by axioms about outcomes of sense actions. Consequently, we have to define the result of augmenting $\mathcal{D}^K$ with knowledge about the outcomes of all the sense actions occurring in S. To do this, we need a concept of the *sense outcomes* of a given action sequence S.

Definition 5.3: Sense outcome function [6]
A *sense outcome function* is any mapping Ω from ground situation terms to knowledge sentences, such that:

1. $\Omega(S_0) = \{\}$,
2. If α is not a sense action,
   $\Omega(do(\alpha, S)) = \Omega(S)$.
3. If α is a sense action $sense_\psi(\vec{X})$,
   $\Omega(do(\alpha, S)) = \Omega(S) \cup \{\textbf{Knows}(\psi(\vec{X}), do(sense_\psi(\vec{X}), S))\}$
   or
   $\Omega(do(\alpha, S)) = \Omega(S) \cup \{\textbf{Knows}(\neg\psi(\vec{X}), do(sense_\psi(\vec{X}), S))\}$.

In the sequel, we shall be interested in $\mathcal{D} \cup \Omega(S)$, namely, the original basic action theory $\mathcal{D}$, augmented by knowledge about the outcomes of all sense actions in the action history S, according to the sense outcome function Ω. Because we are intended to use regression as a solution to the forward projection problem and obtain a version of the regression theorem for augmented basic action theories we need also a definition of $\Omega^\rho(S)$ - a regressed version of $\Omega(S)$.

Recall that when $\psi(\vec{x})$ is an objective expression, and σ a situation term, then $\psi(\vec{x})[\sigma]$ is that situation calculus formula resulting from adding an extra situation argument σ to every situation-suppressed fluent mentioned by ψ. Recall also that $\mathcal{R}[W]$ is a situation calculus formula (whose only situation term is S_0) obtained by applying the regression operator to a regressable sentence W. Thus, $\mathcal{R}[\psi(\vec{x})[\sigma]]$ is a situation calculus formula obtained by applying the regression operator to the formula $\psi(\vec{x})[\sigma]$.

Suppose Ω is a sense outcome function, σ is a ground situation term and σ' is a sub-term of σ.

$$\Omega^\rho(\sigma) = \bigwedge\{\mathcal{R}[\psi(\vec{x})[\sigma']] \mid \textbf{Knows}(\psi(\vec{x}), do(sense_\psi(\vec{x}), \sigma')) \in \Omega(\sigma)\} \wedge \bigwedge\{\mathcal{R}[\neg\psi(\vec{x})[\sigma']] \mid \textbf{Knows}(\neg\psi(\vec{x}), do(sense_\psi(\vec{x}), \sigma')) \in \Omega(\sigma)\},$$

where both conjunctions are over all sub-terms of σ such that $\Omega(\sigma)$ contains a subjective sentence. By convention, $\Omega^\rho(\sigma) =$ TRUE when $\Omega(\sigma) = \{\}$.

One can prove the following lemma formulated in [20]: Because $\mathcal{K}_{Init}$ includes the reflexivity axiom, $\Omega^\rho(\sigma)$ and the conjunction of the sentences in $\Omega(\sigma)$ are logically equivalent, relative to $\mathcal{D}^K$.

The following theorem is a version of the regression theorem for a special case when the regressable formula has the form **Knows**(ϕ, σ) [20]. Recall that $\mathcal{D}^K_{S_0} =$ **Knows**(κ, S_0), $\mathcal{D}_{S_0} = \kappa[S_0]$, where κ is objective and $\mathcal{K}_{Init}$ includes the reflexivity axiom.

Theorem 5.4:
Suppose that ϕ is an objective regressable expression, S_2 is a ground situation term. Then
$\mathcal{D}^K \cup \Omega(S_2) \models$ **Knows**(ϕ, S_2) *iff*
$\mathcal{D}_{una} \cup \mathcal{D}_{S_0} \cup \Omega^\rho(S_2) \models \mathcal{R}_{\mathcal{K}}[\phi[S_2]]$.

Note that in this theorem the regression operator $\mathcal{R}_{\mathcal{K}}$ is defined with respect to $\mathcal{D}^{\mathcal{K}}$.

5.3 A formal comparison

Before we formulate the result about the connection between two approaches to the solution of the forward projection task

[6] Both the notation and results formulated in the rest of this section are introduced in [20] where the reader can find additional details.

that involves both physical and sense actions, we make two important observations.

First, in Theorem 5.2, the regression operator $\mathcal{R}_{\mathcal{D}}$ is defined relative to a different set of successor state axioms, than the regression operator $\mathcal{R}_{\mathcal{K}}$ in Theorem 5.4. Second, in Theorem 5.2, a ground situation term S_1 may include occurrences of sense actions $sense(F_i, V)$, but in Theorem 5.4, a ground situation term S_2 may include occurrences of sense actions $sense_{F_i}(\vec{X})$. We call situation terms S_1 and S_2 similar if in the result of substitution of sense actions of one style for sense actions of another style the terms become equal.

Theorem 5.5:
Suppose that ϕ is an objective regressable expression, S_1 and S_2 are similar ground situation terms, S_2 never mentions the same sense action twice and $\mathcal{K}_{Init}$ includes the reflexivity axiom. Then

$$\mathcal{D}^K \cup \Omega(S_2) \models \mathbf{Knows}(\phi, S_2) \quad \textit{iff} \quad \mathcal{D} \models \phi(S_1).$$

Proof (sketch).
According to the theorems 5.2 and 5.4, we have to prove

$$\mathcal{D}_{una} \cup \mathcal{D}_{S_0} \cup \Omega^{\rho}(S_2) \models \mathcal{R}_{\mathcal{K}}[\phi[S_2]] \quad \text{iff}$$
$$\mathcal{D}_{una} \cup \mathcal{D}_{S_0} \models \mathcal{R}_{\mathcal{D}}[\phi(S_1)]$$

As the preliminary step in our proof, look at the formulas at the right hand side of entailments: they are composed from regressions of relational fluents. For this reason, the most important case is when we consider a relational fluent F_i that occurs either in ϕ or in a formula obtained from ϕ by regression. Let's consider $F_i(\vec{X}, do(A, S_1))$ and $F_i(\vec{X}, do(A, S_2))$, where S_1 and S_2 are similar terms. If A is a physical action or it is a sense action that determines a truth value of fluent different from F_i, then the right hand sides of successor state axioms 3 and 2 for F_i are equivalent; consequently, the regression operators $\mathcal{R}_{\mathcal{D}}$ and $\mathcal{R}_{\mathcal{K}}$ will replace all occurrences of F_i by equivalent expressions. If A is a sense action $sense(F_i, V)$, then $\mathcal{R}_{\mathcal{D}}$ replaces all occurrences of $F_i(\vec{X}, do(sense(F_i, V), S_1))$ by *true* if V=TRUE, or by *false* if V=FALSE. Note that in both cases $\mathcal{R}_{\mathcal{K}}$ replaces all occurrences of $F_i(\vec{X}, do(sense_{F_i}(\vec{X}), S_2))$ by $F_i(\vec{X}, S_2)$, but in addition the set of regressed sense outcomes is augmented by an appropriate formula. More specifically, suppose that V=TRUE, then $\Omega^{\rho}(S_2)$ contains regression of $F_i(\vec{X}, S_2)$; otherwise, if V=FALSE, then $\Omega^{\rho}(S_2)$ contains regression of $\neg F_i(\vec{X}, S_2)$.

Because the statement that we need to prove is equivalent to the following statement

$$\mathcal{D}_{una} \cup \mathcal{D}_{S_0} \models \Omega^{\rho}(S_2) \supset \mathcal{R}_{\mathcal{K}}[\phi[S_2]] \quad \text{iff}$$
$$\mathcal{D}_{una} \cup \mathcal{D}_{S_0} \models \mathcal{R}_{\mathcal{D}}[\phi(S_1)]$$

we can complete our proof by considering an arbitrary model of $\mathcal{D}_{una} \cup \mathcal{D}_{S_0}$.

First, suppose that the formula $\Omega^{\rho}(S_2) \supset \mathcal{R}_{\mathcal{K}}[\phi[S_2]]$ is true in that model. This is possible only if when every conjunct in $\Omega^{\rho}(S_2)$ is true in that model, then $\mathcal{R}_{\mathcal{K}}[\phi[S_2]]$ is also true in that model. Consequently, if we replace every occurrence of $F_i(\vec{X}, \sigma_2)$ in $\Omega^{\rho}(S_2)$ and in $\mathcal{R}_{\mathcal{K}}[\phi[S_2]]$ by *true* and every occurrence of $\neg F_i(\vec{X}, \sigma_2)$ in $\Omega^{\rho}(S_2)$ by *true*, but the corresponding occurrence of $F_i(\vec{X}, \sigma_2)$ in $\mathcal{R}_{\mathcal{K}}[\phi[S_2]]$ by *false*, then the resulting formula will be equivalent to (*true*$\supset \mathcal{R}_{\mathcal{D}}[\phi[S_1]]$) if we replace sense actions by similar sense actions. Because we started with an arbitrary model, this proves that $\mathcal{D}_{una} \cup \mathcal{D}_{S_0} \models \mathcal{R}_{\mathcal{D}}[\phi(S_1)]$

Second, suppose that the formula $\mathcal{R}_{\mathcal{D}}[\phi(S_1)]$ is true in an arbitrary model of $\mathcal{D}_{una} \cup \mathcal{D}_{S_0}$. Consider an arbitrary step in iterative application of the regression operator $\mathcal{R}_{\mathcal{D}}$ and suppose that just before this step $\phi(S_1)$ was regressed to a logically equivalent formula $\psi(\sigma_1)$. The most interesting case is when $\sigma_1 = do(sense(F_i, V), \sigma_1')$ and there are occurrences of F_i in $\psi(\sigma_1)$. In this case, we substitute $F_i(\vec{X}, \sigma_1')$ for all occurrences of $F_i(\vec{X}, do(sense(F_i, V), \sigma_1')$ in $\psi(\sigma_1)$ (denote the resulting formula by $\psi'(\sigma_1')$) and add $F_i(\vec{X}, \sigma_1')$ as a premise to $\psi'(\sigma_1')$ if V=TRUE, or add $\neg F_i(\vec{X}, \sigma_1')$ as a premise to $\psi'(\sigma_1')$ if V=FALSE. Note that the resulting formula $\xi(\sigma_1') \supset \psi'(\sigma_1')$, where $\xi(\sigma_1')$ is the conjunction of the most recently added premise with all premises added on previous iterations (if we add a premise for a first time, then $\xi(\sigma_1')$ is just the premise itself) is equivalent to one-step regression of $\psi(\sigma_1)$ using $\mathcal{R}_{\mathcal{D}}$. In the case when $\sigma_1 = do(A, \sigma_1')$, where A is a physical action, we do the ordinary one step transformation using $\mathcal{R}_{\mathcal{D}}$ from $\psi(\sigma_1)$ to $\psi'(\sigma_1')$ without adding any new premises. Consequently, $\mathcal{R}_{\mathcal{D}}[\phi(S_1)]$ is equivalent to $\Omega^{\rho}(S_2) \supset \mathcal{R}_{\mathcal{K}}[\phi[S_2]]$, where S_2 is a situation term similar to S_1.

6 Discussion

When an axiomatizer expresses in her axioms dynamic properties of a domain with respect to actions that her program will execute, she describes how internal logical model of the world (truth values assigned to properties according to axioms) should evolve as the result of actions (commands that her program sends to effectors). If her program runs in the external (with respect to the program) real world, and not all dynamic aspects of the world are captured in the logical model, then real effects of actions may not coincide with effects predicted by axioms of the model. In addition, the axiomatization of the initial situation (the situation that exists before the program does any actions) will be incomplete in many realistic applications. For this reason, if the program is capable to gaining a sensory feedback from the real world, it can use it to supplement knowledge expressed in the internal model, e.g. to compare sensory readings with predicted values of properties in the model. Consequently, sensory feedback is intended to complete the gaps of unavoidable ignorance of the axiomatizer about numerous properties and causal relationships of the world (the axiomatizer is either not aware of them or not able to formalize them). Even if successor state axioms capture all relevant causal dependencies between actions and domain properties, in many realistic scenarios occurrences of exogenous actions are not observable, but can be determined indirectly from sensing; this is another reason why an agent executing a program in the real world needs sense actions.

By choosing the situation calculus as our representation language, we decide to represent properties of our application by fluents with situation argument. Consequently, fluents are internal subjective representations ('beliefs') of the objective real world, actions in the logical model are symbolic representations of commands that the program sends to its effectors (these actions are called physical actions) or symbolic representations of sensory inputs that the program receives from outside (these actions are called sense actions). Hence, it is convenient to treat both physical and sense actions equally in axioms.

In contrast to this perspective, a traditional approach to axiomatization of physical and sense actions separates them in different sets of axioms. A set of axioms is intended to capture the dynamics of 'objective' fluents with respect to causal effects of physical actions. Another set of axioms is intended to capture the dynamics of the epistemic fluent – something in the mind of an agent. It was assumed that the agent exists outside of the axiomatizer and may not be informed about certain properties of the real world that are accessible to a more knowledgeable axiomatizer. However, if the axiomatizer is supposed to express what she really knows about the world, she does her best to achieve this and she has no reason to hide a valuable information from her program (that will drive actions of the agent in the real world), the first perspective seems also reasonable. Moreover, by adding an agent argument to all fluents, the axiomatizer may wish to distinguish between beliefs of agents vs. 'facts of nature', but we did not consider this option in our paper.

7 Acknowledgments

Thanks to Ray Reiter and anonymous reviewers for comments on preliminary versions of this paper.

References

[1] Fahiem Bacchus, Joseph Y. Halpern, and Hector J. Levesque. Reasoning about noisy sensors in the situation calculus. In *Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence*, pages 1933–1940, Montreal, 1995.

[2] C. Baral and T. Son. Formalizing sensing actions – a transition function based approach. *Artificial Intelligence*, To appear; available at: http://www.public.asu.edu/c̃baral/, 2001.

[3] Ernest Davis. *Representations of Commonsense Knowledge*. Morgan Kaufmann, 1990.

[4] Ernest Davis. Knowledge preconditions for plans. *Journal of Logic and Computation*, 4, no. 5:721–766, 1994. Available at: http://www.cs.nyu.edu/faculty/davise/.

[5] Ernest Davis. The naive physics perplex. *AI Magazine*, 19(4):51–79, 1998.

[6] G. De Giacomo and H. Levesque. Projection using regression and sensors. In *Proceedings of the Sixteenth International Joint Conference on Artificial Intelligence*, Stockholm, Sweden, 1999.

[7] G. De Giacomo, R. Reiter, and M.E. Soutchanski. Execution monitoring of high-level robot programs. In *Principles of Knowledge Representation and Reasoning: Proc. of the 6th International Conference (KR'98)*, pp. 453–464, Italy, 1998.

[8] J. Funge. *Making Them Behave: Cognitive Models for Computer Animation, Ph.D. Thesis*. Dept. of Computer Science, Univ. of Toronto, 1998.

[9] G. Lakemeyer. On sensing and off-line interpreting in golog. In Levesque and Pirri, editors, *Logical Foundations for Cognitive Agents: Contributions in Honor of Ray Reiter*, pages 173–189. Springer, 1999.

[10] G. Lakemeyer and H.J. Levesque. AOL: a logic of acting, sensing, knowing, and only knowing. In *Proceedings of the Sixth International Conference on Principles of Knowledge Representation and Reasoning*, pages 316–327, 1998.

[11] Y. Lespérance and H.J. Levesque. Indexical knowledge and robot action - a logical account. *Artificial Intelligence*, 73:69–115, 1995.

[12] H.J. Levesque and M. Pagnucco. Legolog: Inexpensive experiments in cognitive robotics. In *The 2nd International Cognitive Robotics Workshop (held in conjunction with ECAI-2000)*, pages 104–109, Berlin, Germany, 2000.

[13] H.J. Levesque, F. Pirri, and R. Reiter. Foundations for the situation calculus. *Linköping Electronic Articles in Computer and Information Science. Available at: http://www.ep.liu.se/ea/cis/1998/018/*, vol. 3, N 18, 1998.

[14] John McCarthy. *Formalization of common sense, papers by John McCarthy edited by V. Lifschitz*. Ablex, 1990.

[15] S. McIlraith. Explanatory diagnosis: Conjecturing actions to explain obsevations. In *Principles of Knowledge Representation and Reasoning: Proc. of the 6th International Conference (KR'98)*, pages 167–177, Italy, 1998.

[16] R.C. Moore. A formal theory of knowledge and action. In *Formal Theories of the Commonsense World*, pages 319–358. Ablex, 1985.

[17] G. Piazzolla, M. Vaccaro, A. Finzi, and F. Pirri. Armhand: a mobile manipulator for the blocks world. In *The 2nd International Cognitive Robotics Workshop (held in conjunction with ECAI-2000)*, pages 110–116, Berlin, Germany, 2000.

[18] F. Pirri and A. Finzi. An approach to perception in theory of actions: part 1. *Linköping Electronic Articles in Computer and Information Science*, 4(41), 1999.

[19] F. Pirri and R. Reiter. Some contributions to the metatheory of the situation calculus. *Journal of the ACM*, 46(3):261–325, 1999.

[20] R. Reiter. *Knowledge in Action: Logical Foundations for Describing and Implementing Dynamical Systems*. `http://www.cs.toronto.edu/~cogrobo/`, 2000.

[21] Raymond Reiter. The frame problem in the situation calculus: A simple solution (sometimes) and a completeness result for goal regression. In V. Lifschitz, editor, *Artificial Intelligence and Mathematical Theory of Computation (Papers in Honor of John McCarthy)*, pages 359–380. Academic Press, 1991.

[22] R. Scherl and H.J. Levesque. The frame problem and knowledge producing actions. In *Proceedings of the Eleventh National Conference on Artificial Intelligence*, pages 689–695, Washington, DC, 1993.

[23] M.P. Shanahan. A logical account of the common sense informatic situation for a mobile robot. *Electronic Transactions on Artificial Intelligence*, 2:69–104; available at: http://www–ics.ee.ic.ac.uk/m̃psha/pubs.html, 1998.

[24] S. Shapiro, M. Pagnucco, Y. Lespérance, and H.J. Levesque. Iterated belief change in the situation calculus. In *Proceedings of the Seventh International Conference on Principles of Knowledge Representation and Reasoning*, Breckenridge, Colorado, 2000. Morgan Kaufmann Publishers, San Francisco, CA.

[25] M Soutchanski. An on-line decision-theoretic golog interpreter. In *Technical Report*, Dep. of Computer Science, University of Toronto, 2000. `http://www.cs.toronto.edu/~mes/papers`.

[26] Michael Thielscher. Representing the knowledge of a robot. In A. Cohn, F. Giunchiglia, and B. Selman, editors, *Proceedings of the International Conference on Principles of Knowledge Representation and Reasoning (KR)*, pages 109–120, Breckenridge, CO, April 2000.