# Description Logic-based Framework for Planning with Sensing Actions

**Giuseppe De Giacomo, Luca Iocchi, Daniele Nardi, Riccardo Rosati**
Dipartimento di Informatica e Sistemistica
Università di Roma "La Sapienza"
Via Salaria 113, 00198 Roma, Italy
{degiacomo,iocchi,nardi,rosati}@dis.uniroma1.it

## 1 Introduction

In recent years there has been an attempt to reconcile the theoretical work done on reasoning about action with the realization of agents, in particular mobile robots. Such a field of research has been characterized with the term *Cognitive Robotics* [9].

A mobile robot can indeed be regarded as an intelligent agent, that is designed both to achieve high-level goals and to be able to promptly react and adjust its behavior based on the information acquired through the sensors. Reactive capabilities are necessary to cope with the uncertainties of the real-world; action planning is important as well, if the robot is faced with situations where the knowledge of the environment is incomplete, subject to varying constraints. The integration of the two kinds of functionalities mentioned above is a critical issue in the design of intelligent agents.

The work reported in the present paper builds on a previous proposal [3], which provides a formal framework for reasoning about action derived from *Dynamic Logics* [14] by exploiting the correspondence between such logics and Description Logics (DLs). In such a framework, a number of features that had been analyzed for DLs have proved useful for reasoning about action. In particular, we have extended the language with an epistemic operator interpreted in terms of minimal knowledge, that allows us to express the knowledge about actions in such a way that we can effectively address the planning problem.

We extend the previous proposal with the ability of expressing sensing actions [10], i.e. knowledge producing actions that affect the agent's knowledge, but not the environment. This requires a new kind of axioms for sensing actions and the ability of propagating knowledge to successor states, although in a controlled way. We also extend the implementation with the ability of devising and executing conditional plans, and the underlying control system, by providing new behaviors realizing the sensing actions.

We have implemented our proposal for reasoning about actions on the mobile robot "Tino", which belongs to the *Erratic* family [8]. The implementation relies on the reasoning facilities offered by the DL-based system Classic [2].

## 2 DL-based framework for reasoning about actions

Our general framework for representing dynamic systems follows the lines of Rosenschein's work [14], originally based on Propositional Dynamic Logics (PDLs), which has been initially proposed in [3]. It makes use of the tight correspondence between PDLs and DLs, which allows for considering PDLs and DLs as notational variants of each other. Here, as in [3], we use the notation of DLs, focusing on the well-known DL $\mathcal{ALC}$, corresponding to the standard PDL with atomic programs only.

In addition, we use two nonmonotonic modal operators: a *minimal knowledge operator* **K** and a *default assumption operator* **A**. These are interpreted according to the nonmonotonic modal logic *MKNF* [11], and give rise to the so-called Autoepistemic Description Logic $\mathcal{ALCK}_{\mathcal{NF}}$ [5]. We do not have the space here to formally introduce such a logical framework, we refer the reader to [3] and [5]. Rather, we give an intuition of the underlying semantics.

The interpretation structures of DLs (PDLs) are essentially *graphs* labeled both on nodes and arcs. *Nodes*, called *individuals* in DLs, (states in PDLs) are labeled by *concepts* (formulae in PDLs) that denote properties of individuals. *Arcs*, called *links* in DL (state transitions in PDLs) are labeled by *roles* (actions in PDLs). Such interpretation structures can be concretely bound to the robot's behavior (possible courses of actions): individuals represent states of the robot and are labeled by concepts representing what is true in that state; links between individuals represent transitions between states of the robot, and are labeled by roles representing the actions that cause the state transition.

However, in general there is not enough information about the robot's environment to model its behavior

| Axioms | Set Name | Syntax |
|---|---|---|
| Static axioms | $\Gamma_S$ | $C_1 \sqsubseteq C_2$ |
| Action precondition axioms | $\Gamma_P$ | moving actions: $\mathbf{K}C \sqsubseteq \exists \mathbf{K}R_M.\top$<br>sensing actions: $\mathbf{K}C \sqcap \neg\mathbf{A}S \sqcap \neg\mathbf{A}\neg S \sqsubseteq \exists \mathbf{K}R_S.\top$ |
| Effect axioms | $\Gamma_E$ | moving actions: $\mathbf{K}C \sqsubseteq \forall \mathbf{K}R_M.\mathbf{K}D$<br>sensing actions: $\mathbf{K}\top \sqsubseteq \forall \mathbf{K}R_S.\mathbf{K}S \sqcup \mathbf{K}\neg S$ |
| Frame axioms schema | $\Gamma_{FR}$ | sensing actions only: $\mathbf{K}\varphi \sqsubseteq \forall \mathbf{K}R_S.\mathbf{K}\varphi$ |
| Initial state description axioms | $\Gamma_I$ | $C(init)$ |

Table 1: Axioms forming the knowledge base

by means of a single interpretation structure, since the robot's behavior will depend on external circumstances that will be known only at execution time. Rather, we model the robot's behavior with suitable axioms which reflect our (partial) knowledge and which are satisfied by *multiple* interpretation structures. As a consequence, in order to decide which action to perform next the robot can use only those facts that are "valid" in its current state, i.e. that are true in the representative of its current state in all possible interpretation structures. To do so the logical formalism must provide:

- A mechanism to isolate an individual representative of a given robot's state, in each possible interpretation, establishing a one-to-one mapping between the individuals in the different interpretation structures that represent the same robot's state.
- A mechanism to represent that a certain property (concept) is "valid" in a robot's state, i.e. true in the representatives of that state in all possible interpretations.

The minimal knowledge operator $\mathbf{K}$ gives us both the above mechanisms. On the one hand, it allows for isolating the representatives of robot's states in the different structures establishing a one-to-one mapping among them through the so-called *known individuals*. In general, known individuals will be only those that are explicitly *named* in some axiom (in our case, we will have a single such named individual, *init*, denoting the initial state of the robot) and those generated by a special use of $\mathbf{K}$ on roles denoting actions. On the other hand, it allows for denoting the "validity" of a property in a robot's state. In particular, an epistemic implication of the form $\mathbf{K}C \sqsubseteq D$ differs from the non-modal implication $C \sqsubseteq D$ since $D$ is concluded for a given known individual only if $C$ is necessarily true ("valid") for that known individual. This prevents forms of reasoning by cases such as the following: let $\Sigma = \{[C_1 \sqcup C_2](init), \mathbf{K}C_1 \sqsubseteq D, \mathbf{K}C_2 \sqsubseteq D\}$, then $\Sigma \not\models D(init)$, while let $\Sigma' = \{[C_1 \sqcup C_2](init), C_1 \sqsubseteq D, C_2 \sqsubseteq D\}$, then $\Sigma' \models D(init)$. Moreover, for $\mathbf{K}C \sqsubseteq D$ the contrapositive does not hold, i.e. $\neg D$ does not imply $\neg C$. Epistemic sentences $\mathbf{K}C \sqsubseteq D$ can be naturally interpreted in terms of *rules*, i.e. a forward reasoning mechanism.

The default assumption operator $\mathbf{A}$ allows for expressing justifications of default rules [13], and the combined usage of $\mathbf{K}$ and $\mathbf{A}$ allows for formalizing defaults in terms of modal formulas. We use it here in relation with sensing actions in a very specific way (see below).

## 3 Robot's behavior representation

We distinguish two kinds of robot's actions: *moving actions* (actions that result in a change in the environment) and *sensing actions* (actions changing only the knowledge of the robot). Both kinds of actions are considered *deterministic*, in the sense that a unique successor state will be generated by each action.

Like most approaches to reasoning about actions we express our knowledge in terms of a finite set of axioms forming a knowledge base $\Sigma$. Such axioms are partitioned in the classes shown in Table 1[1], each formalized in a specific way.

**Static axioms** are used for representing background knowledge, which is invariant with respect to the execution of actions: they hold in every state, and they do not depend on actions.

**Action precondition axioms** describe under which circumstances it is possible to execute an action. There are two different forms depending on the kind of action. For moving actions, these axioms can be read as: if $C$ holds in the (known individual denoting the) current state $s$, then there exists a (known individual denoting a) state $s'$ which is the $R_M$-successor of $s$. While for sensing actions they can be interpreted as: if $C$ holds in the current state $s$ and the truth value of $S$ is not known (i.e. it is consistent to assume both that $S$ holds in $x$ in every interpretation and that $\neg S$ holds in $s$ in every interpretation), then it is possible to perform $R_S$, in the sense that there exists a unique $R_S$-successor $s'$ of $s$ which is the same in every interpretation.

---

[1] In the table $C$ and $D$ are $\mathcal{ALC}$ concepts describing state properties, $S$ are special atomic concepts denoting sensed properties, $\varphi$ stands for any $\mathcal{ALC}$ concept, $R_M$,$R_S$ are roles representing respectively moving and sensing actions, *init* is an individual denoting the initial state.

**Effect axioms** specify the effects of executing an action $R_M$ or $R_S$ in a state satisfying certain *premises.* Effect axioms for moving actions can be read as: if $C$ holds in the (known individual denoting the) current state $s$ and there exists a (known individual denoting the) $R_M$-successor $s'$ of $s$, then $D$ holds in $s'$ in all interpretations. While each sensing action has a *unique* effect axiom, which expresses that after having performed $R_S$ the robot knows the truth-value of sensed proposition $S$, i.e. it knows *whether* $S$ holds or not.

**Initial state description axioms** specify the facts that hold in the initial state of the robot, by asserting that $C$ holds in the state *init* in every possible interpretation.

Finally we enforce a **frame axiom schema** for each sensing action $R_S$, that propagates all concepts that hold in the current state $s$ to the next state $s'$. Observe that the $\neg\mathbf{A}S \sqcap \neg\mathbf{A}\neg S$ in the premises of the precondition axioms for $R_S$ prevents the execution of $R_S$ in case either $\mathbf{K}S$ or $\mathbf{K}\neg S$ holds in the previous state. Hence, no contradiction may be generated from instances of the frame axiom schema and the effect axiom for $R_S$.

Note that we do not try to address the frame problem for moving actions by enforcing some general form of *common sense inertia law.* Hence if a property $C$ persists after a certain action $R_M$ is performed, a specific effect axiom of the form $\mathbf{K}C \sqsubseteq \forall\mathbf{K}R_M.\mathbf{K}C$ must be included.

**The planning problem** The robot's ability of sensing can be used to extend the notion of plan considered in [3] to the notion of *conditional plan.* Indeed the robot may use its sensing capability to choose different courses of actions leading to a given goal, depending on the value of the sensed propositions.

In deductive planning one is typically interested in answering the following question: "Is there a sequence of actions that, starting from an initial state, leads to a state where a given property (the goal) holds?". This is captured in our framework by the following logical implication:

$$\Sigma \models \Pi_G(init) \qquad (1)$$

where: (i) $\Sigma$ is the knowledge base including the static axioms $\Gamma_S$, the action precondition axioms $\Gamma_P$ and the effect axioms $\Gamma_E$ for both moving and sensing actions plus the frame axiom schema $\Gamma_{FR}$ for the sensing actions, and the initial state description axioms $\Gamma_I$; (ii) $\Pi_G(init)$ denotes that $\Pi_G$ holds in the initial state *init*, where $\Pi_G$ is *any* concept belonging to the set $\mathcal{P}$ defined inductively as follows: (i) $\mathbf{K}G \in \mathcal{P}$; (ii) if $C \in \mathcal{P}$, then $\exists\mathbf{K}R_{M_i}.C \in \mathcal{P}$, for every moving action $R_{M_i}$; (iii) if $C_1, C_2 \in \mathcal{P}$, then $\exists\mathbf{K}R_{S_i}.(\mathbf{K}S_i \sqcap C_1) \sqcup (\mathbf{K}\neg S_i \sqcap C_2) \in \mathcal{P}$, for every sensing action $R_{S_i}$.

Notice that, if only moving actions are considered, then $\Pi_G$ stands for any concept expression of the form $\exists\mathbf{K}R_{M_1}.\exists\mathbf{K}R_{M_2}.\ldots.\exists\mathbf{K}R_{M_n}.\mathbf{K}G$ and it expresses the fact that from the initial state *init* there exists a sequence of successors (the same in every interpretation) that terminates in a state (the same in every interpretation) where $G$ holds (in every interpretation). When sensing actions are added, $\Pi_G$ denotes a conditional plan, in which each branch leads to a state satisfying the goal.

# 4 Plan generation

To the aim of generating plans in the framework proposed, we introduce the notion of *first-order extension* of a (epistemic) knowledge base $\Sigma = \Gamma_S \cup \Gamma_P \cup \Gamma_E \cup \Gamma_{FR} \cup \Gamma_I$ containing the specification of the robot's behavior in the terms described above.

Informally, the first-order extension of $\Sigma$ (denoted as $FOE(\Sigma)$) is an $\mathcal{ALC}$ knowledge base which consists of (1) the static axioms in $\Gamma_S$; (2) the specification of the initial state (the assertions on *init* in $\Gamma_I$) augmented by the assertions which are consequences (up to renaming of individuals) of the epistemic sentences in $\Sigma$. The FOE of $\Sigma$ provides a unique characterization of the knowledge that is shared by all the models of $\Sigma$, which is relevant wrt the planning problem.

In the first-order extension, we replace each sensing action $R_S$ by two special actions $R_S^+$ and $R_S^-$. We denote by $\Gamma_E^\pm$ the set of effect axioms $\Gamma_E$ in which those for the sensing actions $R_S$ are replaced by

$$\mathbf{K}\top \sqsubseteq \forall\mathbf{K}R_S^+.\mathbf{K}S \qquad\qquad \mathbf{K}\top \sqsubseteq \forall\mathbf{K}R_S^-.\mathbf{K}\neg S.$$

We also use only a finite number of instances of the frame axiom schemas. We denote by $\Gamma_{IFR}^\pm$ the set of axioms:

$$\mathbf{K}C \sqsubseteq \forall\mathbf{K}R_S^+.\mathbf{K}C \qquad\qquad \mathbf{K}C \sqsubseteq \forall\mathbf{K}R_S^-.\mathbf{K}C$$

obtained by: (1) instantiating the frame axiom schemas in $\Gamma_{FR}$ for each concept $C$ such that either $C(init) \in \Gamma_I$, or $\mathbf{K}C$ is in the postcondition of some effect axiom in $\Gamma_E$ (i.e., $C$ such that $\mathbf{K}D \sqsubseteq \forall\mathbf{K}R_M.\mathbf{K}C$, or $C, \neg C$ such that $\mathbf{K}\top \sqsubseteq \forall\mathbf{K}R_S.\mathbf{K}C \sqcup \mathbf{K}\neg C$ in $\Gamma_E$); (2) replacing each sensing action $R_S$ by the two special actions $R_S^+$ and $R_S^-$.

The algorithm computing the FOE, starting from the initial state *init*, applies to each state the rules in the sets $\Gamma_E^\pm \cup \Gamma_{IFR}^\pm$ which are triggered by such a state. A new state is thus generated, unless a state with the same properties had already been created. In this way the effect of the rules is computed, obtaining a sort of "completion" of the knowledge base. See [4] for a detailed description of the algorithm.

The notion of first-order extension constitutes the basis of a sound and complete planning method. More specifically, the planning problem in $\Sigma$ expressed by (1) can be reduced to an entailment problem in $FOE(\Sigma)$, by making use of the following translation function $\tau(\cdot)$.

**Definition 4.1** *Let $C$ be a concept belonging to the set $\mathcal{P}$. Then, $\tau(C)$ is the concept expression obtained as follows:*
*(i) if* $C = \mathbf{K}G$ *then* $\tau(C) = \mathbf{K}G$;
*(ii) if* $C = \exists\mathbf{K}R_{M_i}.C_1$ *then* $\tau(C) = \exists\mathbf{K}R_{M_i}.\tau(C_1)$;
*(iii) if* $C = \exists\mathbf{K}R_{S_i}.(\mathbf{K}S_i \sqcap C_1) \sqcup (\mathbf{K}\neg S_i \sqcap C_2)$ *then* $\tau(C) = \exists\mathbf{K}R^{+}_{S_i}.\tau(C_1) \sqcap \exists\mathbf{K}R^{-}_{S_i}.\tau(C_2)$.

**Theorem 4.2** *Let* $C \in \mathcal{P}$. *Then,* $\Sigma \models C(init)$ *iff* $FOE(\Sigma) \models \tau(C)(init)$.

## 5 Implementation

The framework previously presented has been actually used to describe the knowledge of the mobile robot Tino of the Erratic family [8]. In such implementation we use a restricted DL language to represent the robot's knowledge, which allows us to rely on the reasoning services provided by the well-known DL system Classic [2]. In particular, we make use of the built-in instance checking mechanism to check the validity of a concept in a state, and of triggering of rules to propagate effects. However, Classic does not provide an implementation for **K** and **A**, which are therefore handled by ad hoc attached procedures.

The planning procedure, given an initial state and a goal, generates a conditional plan that, when executed starting from the initial state, leads to a state in which the goal is satisfied. Furthermore, dynamic execution of plans is supervised by the monitor, which is responsible for integrating planning and control.

Conceptually, the generation of conditional plans is achieved in two steps. First, the FOE of the knowledge base, which can be seen as an action graph representing all possible plans starting from the initial state, is generated. Then, such a graph is visited building a term (the conditional plan) representing a tree in which: (i) sensing actions generate branches; (ii) each branch leads to a state in which the goal is satisfied. We refer to [4] for further details.

## 6 Conclusions

We have proposed a logical framework for reasoning about actions which provides for the formalization of sensing actions. In particular, we have shown that the use of the epistemic state of the agent (represented through the modal operators of $\mathcal{ALCK}_{\mathcal{NF}}$) allows for the formalization of sensing actions. Our approach has strong connections with previous research on logical formalization of knowledge-producing actions [15, 12]. Our work on sensing is also related to [6], which presents a formalization of sensing actions based on an extension of STRIPS constructs. Sensing actions are distinguished by means of an annotation mechanism on the postconditions. A particular use of annotated propositions allows for expressing constraints on the plan, for example sensing the color of a door in order to enter into a room with a blue door is allowed, while painting a door blue in order to enter into such a room is forbidden. We are currently studying the possibility of adding plan constraints to our planner. In particular, we want to exploit the ideas reported in [1], which have been shown effective for speeding up the planning process.

## References

[1] Fahiem Bacchus and Froduald Kabanza. Using temporal logic to control search in a forward chaining planner. In *Proceedings of the Third European Workshop on Planning*, 1995.

[2] Alexander Borgida, Ronald J. Brachman, Deborah L. McGuinness, and Lori Alperin Resnick. CLASSIC: A structural data model for objects. In *Proceedings of the ACM SIGMOD International Conference on Management of Data*, pages 59–67, 1989.

[3] Giuseppe De Giacomo, Luca Iocchi, Daniele Nardi, and Riccardo Rosati. Moving a robot: the KR&R approach at work. In *Proceedings of the Fifth International Conference on the Principles of Knowledge Representation and Reasoning (KR-96)*, 1996.

[4] Giuseppe De Giacomo, Luca Iocchi, Daniele Nardi, and Riccardo Rosati. Planning with sensing for a mobile robot. In *Proceedings of the Fourth European Conference on Planning (ECP-97)*, Toulouse (France), 1997.

[5] Francesco M. Donini, Daniele Nardi, and Riccardo Rosati. Autoepistemic description logics. In *Proceedings of the Fifteenth International Joint Conference on Artificial Intelligence (IJCAI-97)*, pages 136–141, 1997.

[6] Oren Etzioni, Steve Hanks, Daniel Weld, Denise Draper, Neal Lesh, and Mike Williamson. An approach to planning with incomplete information. *Proceedings of the Third International Conference on the Principles of Knowledge Representation and Reasoning (KR-92)*, 1992.

[7] Matthew L. Ginsberg, editor. *Readings in Nonmonotonic Reasoning.* Morgan Kaufmann, Los Altos, 1987.

[8] Kurt Konolige. Erratic competes with the big boys. *AAAI Magazine*, 2:61–67, 1995.

[9] Yves Lesperance, Hector J. Levesque, Fangzhen Lin, Daniel Marcu, Raymond Reiter, and Richard B. Scherl. A logical approach to high-level robot programming – a progress report. In *Control of the Physical World by Intelligent Systems, Papers*

*from the 1994 AAAI Fall Symposium*, B. Kuipers ed., 1994.

[10] Hector J. Levesque. What is planning in presence of sensing? In *Proceedings of the Thirteenth National Conference on Artificial Intelligence (AAAI-96)*, pages 1139–1149, 1996.

[11] Vladimir Lifschitz. Minimal belief and negation as failure. *Artificial Intelligence Journal*, 70:53–72, 1994.

[12] Fangzhen Lin and Yoav Shoham. On non-forgetting and minimal learning. In *Proceedings of the 1993 International Colloquium on Cognitive Science*, 1993.

[13] Raymond Reiter. A logic for default reasoning. *Artificial Intelligence Journal*, 13:81–132, 1980. Republished in [7].

[14] Stanley J. Rosenschein. Plan synthesis: a logical approach. In *Proceedings of the Eighth International Joint Conference on Artificial Intelligence*, 1981.

[15] Richard Scherl and Hector J. Levesque. The frame problem and knowledge producing action. In *Proceedings of the Eleventh National Conference on Artificial Intelligence (AAAI-93)*, pages 689–695, 1993.