Complexity of Projection with Stochastic Actions in a Probabilistic Description Logic

Benjamin Zarrieß
Institute of Theoretical Computer Science
Technische Universität Dresden
benjamin.zarriess@tu-dresden.de

Abstract

We consider an action language extended with quantitative notions of uncertainty. In our setting, the initial beliefs of an agent are represented as a probabilistic knowledge base with axioms formulated in the Description Logic $\mathcal{ALCO}$. Action descriptions describe the possibly context-sensitive and non-deterministic effects of actions and provide likelihood distributions over the different possible outcomes of actions. In this paper, we prove decidability of the projection problem which is the basic reasoning task needed for predicting the outcome of action sequences. Furthermore, we investigate how the non-determinism in the action model affects the complexity of the projection problem.

Introduction

Integrating probabilistic notions of uncertainty into languages for reasoning about actions is a popular approach to adequately deal for instance with possibly fallible acting and perception. Probabilistic extensions of the Situation Calculus (McCarthy and Hayes 1969; Reiter 2001), where first-order logic is the underlying base logic, have been studied for instance in (Bacchus, Halpern, and Levesque 1999; Gabaldon and Lakemeyer 2007; Belle and Lakemeyer 2017). A basic reasoning problem in such a language is the *projection problem*. For given statements about the initial probabilistic beliefs of an agent and a finite sequence of stochastic actions the problem is to check whether a given query is true after executing the action sequence. Solving such lookahead tasks is for example important in planning and for high-level control of autonomous robots.

As a motivating example, we consider a domestic service robot that has the task of serving food to a person named John. In this scenario, we assume that the knowledge of the robot about the ingredients of food and possible food allergies of John is non-crisp and incomplete. Furthermore, the outcome of actions needed to deliver food is also unknown to a certain degree. For example, in the programming language Readylog (Ferrein, Fritz, and Lakemeyer 2005; Ferrein and Lakemeyer 2008), that has the Situation Calculus as its formal underpinning and is a variant of Golog

(Levesque et al. 1997), one could write the following high-level program

> **while** $\exists x. AvailableFood(x) \wedge WantsFood(\mathsf{john})$ **do**
>   **if** $\boldsymbol{pproj}(Healthy(\mathsf{john}), \mathtt{serve}(x, \mathsf{john})) \geq 0.9$ **then**
> $\mathtt{serve}(x, \mathsf{john})$
>   **end**
> **end**

It describes an agent weighing the outcome of an action sequence named $\mathtt{serve}(x, \mathsf{john})$ in terms of the health status of John before actually serving the food x. The if-condition with $\boldsymbol{pproj}$ is a construct for probabilistic projection. Here, it checks whether the degree of belief in $Healthy(\mathsf{john})$ is above the threshold after projecting $\mathtt{serve}(x, \mathsf{john})$.

The problem is that in representations based on a probabilistic Situation Calculus like the one of (Bacchus, Halpern, and Levesque 1999) the projection problem is undecidable. One well-established principled approach to obtain an expressive action formalism with a decidable projection problem is to use an action language based on Description Logics (DLs) (Baader et al. 2003; 2017) such as the ones in e.g. (Baader et al. 2005; Gu and Soutchanski 2010; Baader, Lippmann, and Liu 2010; Zarrieß and Claßen 2015; Ahmetaj et al. 2017). Most of these formalisms have in common that they can be considered as fragments of the Situation Calculus, that are elaboration tolerant, can handle incomplete information, allow to integrate complex background knowledge in form of DL ontologies and have a decidable projection problem with a complexity that is often not higher than standard reasoning problems in the underlying expressive DLs. For example, in the service robot domain described above it would be useful to also integrate static terminological knowledge about food allergies and other health care related terms as it is provided for example in bio-medical ontologies like SNOMED CT. However, to the best of our knowledge, adding probabilistic degrees of belief and stochastic actions to those DL-based action formalisms has not been considered so far.

In the present work, we propose a DL-based action formalism where one can talk about the probabilistic beliefs of an agent and can represent stochastic actions to deal with uncertainty about action outcomes. Our main contribution is an analysis of the impact of these extensions

on the complexity of the projection problem. In particular, to represent the initial belief state of the agent we consider knowledge bases with subjective probabilities applied to ABox facts (assertions about concrete individuals) and TBox statements (ontologies) formulated in the basic expressive DL $\mathcal{ALCO}$. The resulting logic for formulating KBs can be seen as a member of the Prob-$\mathcal{ALC}$ family of probabilistic DLs studied in detail in (Lutz and Schröder 2010; Gutiérrez-Basulto et al. 2017) and is a decidable fragment of Halpern's Type 2 probabilistic first-order logic (Halpern 1990). For the dynamics, action descriptions similar to the ones in (Baader et al. 2005) represent the possibly conditional effects of primitive deterministic actions. A stochastic action is then decomposed into a finite set of primitive actions representing the set of possible outcomes. A discrete probability distribution on this set describes the likelihood of the respective outcome. Projection queries are again formulated as probabilistic KBs over $\mathcal{ALCO}$-axioms. For example, the question whether action executions preserve the agent's belief in an $\mathcal{ALCO}$-ontology can be formulated as an instance of the projection problem. We show that the projection problem is EXPTIME-complete if only sequences of deterministic actions are considered. However, in presence of stochastic actions we show 2EXPTIME-completeness.

Detailed proofs can be found in a technical report (Zarrieß 2018).

Probabilistic KBs over $\mathcal{ALCO}$-Axioms

First, we provide basic definitions of the standard (non-probabilistic) DL $\mathcal{ALCO}$, and second, we define probabilistic KBs over $\mathcal{ALCO}$-axioms.

Description Logic $\mathcal{ALCO}$

We consider a fixed vocabulary with countably infinite pairwise disjoint sets $\mathsf{N_C}$ of *concept names*, $\mathsf{N_R}$ of *role names*, and $\mathsf{N_I}$ of *individual names*. Concept descriptions are built from the vocabulary and several concept constructors.

Definition 1. An $\mathcal{ALCO}$-*concept description* (*concept* for short) C is built according to the following syntax rule

$$C ::= A \mid \{a\} \mid C \sqcap C \mid \neg C \mid \exists r.C,$$

where $A \in \mathsf{N_C}$ stands for a concept name, $a \in \mathsf{N_I}$ for an individual name and $r \in \mathsf{N_R}$ for a role name. Additional concept constructors are defined as the usual abbreviations: $\top := A \sqcup \neg A$ for some arbitrary but fixed concept name $A \in \mathsf{N_C}$; $\bot := \neg\top$; $C \sqcup D := \neg(\neg C \sqcap \neg D); \forall r.C := \neg\exists r.\neg C$, where C and D are arbitrary concepts and r a role name.

Next, we define the syntax of knowledge bases.

Definition 2. A *concept inclusion* (CI) is an axiom of the form $C \sqsubseteq D$, where C and D are concepts. A *knowledge base* (*KB* for short) φ is built according to the following syntax rule:

$$\varphi ::= C \sqsubseteq D \mid \neg\varphi \mid \varphi \wedge \varphi,$$

where $C \sqsubseteq D$ stands for a CI. Other Boolean connectives like $\vee$ (disjunction) and $\rightarrow$ (implication) are defined as the usual abbreviations.

A CI of the form $\{a\} \sqsubseteq C$ is also written as $C(a)$ (called *concept assertion*) and one of the form $\{a\} \sqsubseteq \exists r.\{b\}$ as $r(a, b)$ (called *role assertion*). The semantics is defined in terms of interpretations.

Definition 3. An *interpretation* is a pair $\mathcal{I} = (\Delta_\mathcal{I}, \cdot^\mathcal{I})$, where $\Delta_\mathcal{I}$ is a non-empty domain and $\cdot^\mathcal{I}$ a mapping that maps each $A \in \mathsf{N_C}$ to a set $A^\mathcal{I} \subseteq \Delta_\mathcal{I}$, each role name $r \in \mathsf{N_R}$ to a binary relation $r^\mathcal{I} \subseteq \Delta_\mathcal{I} \times \Delta_\mathcal{I}$ and each individual name $a \in \mathsf{N_I}$ to an element $a^\mathcal{I} \in \Delta_\mathcal{I}$.

The *extension of a concept C under $\mathcal{I}$*, denoted by $C^\mathcal{I}$, is defined as a subset of $\Delta_\mathcal{I}$ by induction on the structure of C as follows

$$\{a\}^\mathcal{I} := \{a^\mathcal{I}\}; \quad (D \sqcap E)^\mathcal{I} := D^\mathcal{I} \cap E^\mathcal{I};$$
$$(\neg D)^\mathcal{I} := \Delta_\mathcal{I} \setminus D^\mathcal{I};$$
$$(\exists r.D)^\mathcal{I} := \{d \in \Delta_\mathcal{I} \mid \exists e \in \Delta_\mathcal{I}.(d,e) \in r^\mathcal{I} \wedge e \in D^\mathcal{I}\},$$

where $A \in \mathsf{N_C}$, $a \in \mathsf{N_I}$, $r \in \mathsf{N_R}$ and D and E are concepts.

Let $\mathcal{I}$ be an interpretation and φ a KB. *Satisfaction of φ in $\mathcal{I}$*, denoted by $\mathcal{I} \models \varphi$, is defined by induction on the structure of φ as follows:

$$\mathcal{I} \models C \sqsubseteq D \text{ iff } C^\mathcal{I} \subseteq D^\mathcal{I}; \quad \mathcal{I} \models \neg\varphi_1 \text{ iff } \mathcal{I} \not\models \varphi_1;$$
$$\mathcal{I} \models \varphi_1 \wedge \varphi_2 \text{ iff } \mathcal{I} \models \varphi_1 \text{ and } \mathcal{I} \models \varphi_2.$$

$\mathcal{I}$ is a *model of a KB φ* iff φ is satisfied in $\mathcal{I}$. We say that a KB is *consistent* iff it has a model.

Checking consistency of a KB is EXPTIME-complete.

Probabilistic Knowledge Bases

Our language for formulating probabilistic KBs is a fragment of Halpern's Type 2 probabilistic first-order logic, i.e. probabilities are subjective and are viewed as degrees of belief which seems to be appropriate for the agent-oriented setting of this paper. The logic we obtain is similar in spirit to the probabilistic DL Prob-$\mathcal{ALC}$ studied in (Lutz and Schröder 2010; Gutiérrez-Basulto et al. 2017). However, we do not allow probabilistic operators for forming concepts and only talk about probabilistic uncertainty at the level of axioms.

Syntax and Semantics Syntactically, our logic is the same as the one in (Fagin, Halpern, and Megiddo 1990) but with KBs in place of atomic propositions.

Definition 4. A *belief term* B is an expression that is built according to the following syntax rule:

$$\mathsf{B} ::= 0 \mid 1 \mid \mathbf{B}\varphi \mid \mathsf{B} + \mathsf{B} \mid \mathsf{B} \times \mathsf{B},$$

where φ stands for a KB. A *belief (in)equality* is of the form

$$\mathsf{B}_1 \sim \mathsf{B}_2 \text{ with } \sim \in \{>, \geq, =, \leq, <\},$$

where B_1 and B_2 are belief terms. A *probabilistic knowledge base* (*PKB* for short) $\mathcal{K}$ is built according the following syntax rule

$$\mathcal{K} ::= \mathsf{B}_1 \sim \mathsf{B}_2 \mid \neg\mathcal{K} \mid \mathcal{K} \wedge \mathcal{K},$$

where $\mathsf{B}_1 \sim \mathsf{B}_2$ stands for a belief inequality.

Intuitively, a belief term of the form $\mathbf{B}\varphi$ stands for the degree of belief in the (objective) KB φ formulated in $\mathcal{ALCO}$. Note that rational constants in inequalities can be expressed by clearing the denominator and appropriate normalization.

The semantics is given in terms of possible worlds, where each world is associated with a standard DL interpretation (Def. 3). There is a discrete probability distribution over the set of all possible worlds.

Definition 5. A *probabilistic interpretation* is of the form
$$\mathfrak{I} = (\Delta, W, (\mathcal{I}_w)_{w \in W}, \mu),$$
where

- Δ is a non-empty *domain*;
- W a non-empty set of *possible worlds*;
- (W, μ) is a discrete probability space with sample space W and probability function μ;
- for each $w \in W$, $\mathcal{I}_w$ is an interpretation with $\Delta_{\mathcal{I}_w} = \Delta$.

Furthermore, we assume *rigid individuals*, i.e. $a^{\mathcal{I}_w} = a^{\mathcal{I}_{w'}}$ for all $a \in \mathsf{N_I}$ and all $w, w' \in W$.

Since (W, μ) is defined as a discrete probability space, W and the probability measure μ have the following general properties: W is countable; for each $w \in W$ we have $\mu(\{w\}) \in [0, 1]$; for each set $S \subseteq W$ it holds that $\mu(S) = \sum_{w \in S} \mu(\{w\})$. Thus, μ is uniquely determined by the values assigned to each singleton set consisting of a possible world.

Definition 6 (semantics of PKBs)**.** Let B be a belief term and let $\mathfrak{I} = (\Delta, W, (\mathcal{I}_w)_{w \in W}, \mu)$ a probabilistic interpretation. By induction on the structure of B we define a *function* $\cdot^{\mathfrak{I}}$ that maps B to its *degree of belief in* $\mathfrak{I}$ denoted by $\mathsf{B}^{\mathfrak{I}}$. It is defined as follows:

$$0^{\mathfrak{I}} := 0 \text{ and } 1^{\mathfrak{I}} := 1;$$
$$(\mathbf{B}\varphi)^{\mathfrak{I}} := \mu(\{w \in W \mid \mathcal{I}_w \models \varphi\});$$
$$(\mathsf{B}_1 + \mathsf{B}_2)^{\mathfrak{I}} := (\mathsf{B}_1)^{\mathfrak{I}} + (\mathsf{B}_2)^{\mathfrak{I}} \text{ and}$$
$$(\mathsf{B}_1 \times \mathsf{B}_2)^{\mathfrak{I}} := (\mathsf{B}_1)^{\mathfrak{I}} \times (\mathsf{B}_2)^{\mathfrak{I}}.$$

Let $\mathcal{K}$ be a PKB and $\mathfrak{I}$ a probabilistic interpretation. *Satisfaction of $\mathcal{K}$ in $\mathfrak{I}$*, written as $\mathfrak{I} \models \mathcal{K}$, is defined by induction on the structure of $\mathcal{K}$ as follows

$$\mathfrak{I} \models \mathsf{B}_1 \sim \mathsf{B}_2 \text{ iff } (\mathsf{B}_1)^{\mathfrak{I}} \sim (\mathsf{B}_2)^{\mathfrak{I}};$$
$$\mathfrak{I} \models \neg\mathcal{K}_1 \quad \text{ iff } \mathfrak{I} \not\models \mathcal{K}_1;$$
$$\mathfrak{I} \models \mathcal{K}_1 \wedge \mathcal{K}_2 \text{ iff } \mathfrak{I} \models \mathcal{K}_1 \text{ and } \mathfrak{I} \models \mathcal{K}_2.$$

$\mathfrak{I}$ is a *model* of $\mathcal{K}$ iff $\mathfrak{I} \models \mathcal{K}$. $\mathcal{K}$ is called *consistent* iff it has a model.

Note that the semantics is based on exactly the same underlying assumptions as in (Lutz and Schröder 2010).

Example 7. Based on the example in the introduction we define a simple PKB about food allergies and some concrete facts about individuals. Concept names start with an uppercase letter and role and individual names with a lowercase letter. First, some known facts are given:

$$\mathbf{B}(Person \sqsubseteq \forall has\text{-}allergy\text{-}to.Allergen) = 1 \wedge$$
$$\mathbf{B}((Person \sqcap Healthy)(\mathsf{john})) = 1 \wedge$$
$$\mathbf{B}(Food(\mathsf{f}_1) \wedge Food(\mathsf{f}_2)) = 1.$$

A person can only have an allergy to an allergen. And we talk about a healthy person john and some concrete food f_1 and f_2. The agent believes that john has an allergy to peanuts with a degree of at least 0.9:

$$\mathbf{B}(has\text{-}allergy\text{-}to(\mathsf{john}, \mathsf{peanut})) \geq 0.9,$$

Furthermore, the agent believes that f_1 is more allergy-friendly than f_2 and that both corresponding degrees are independent of each other:

$$\mathbf{B}\varphi_1 > \mathbf{B}\varphi_2 \wedge$$
$$\mathbf{B}(\varphi_1 \wedge \varphi_2) = \mathbf{B}\varphi_1 \times \mathbf{B}\varphi_2$$

with

$$\varphi_1 := (\exists in.\{\mathsf{f}_1\} \sqsubseteq \neg Allergen),$$
$$\varphi_2 := (\exists in.\{\mathsf{f}_2\} \sqsubseteq \neg Allergen),$$

where φ_1 and φ_2 say that the all the ingredients (related via the role name in) of f_1 and f_2, respectively, are not allergens.

Note that a PKB does not represent a fixed probability distribution. It incompletely describes a probabilistic belief state by providing some constraints for a distribution. Also note that a PKB is a purely subjective theory and does not talk about objective truth.

Deciding Consistency of Probabilistic KBs

For deciding consistency of a PKB the corresponding results for Prob-$\mathcal{ALC}$ in (Gutiérrez-Basulto et al. 2017) do not directly apply to our problem because $\mathcal{ALCO}$ in addition provides nominals and PKBs talk about degrees of belief in concept inclusions which is not possible with Prob-$\mathcal{ALC}$-KBs. However, the chosen extensions for the present work do not pose any major additional difficulties.

The algorithm for deciding consistency is basically of the same kind as the one for Prob-$\mathcal{ALC}$, which in turn uses ideas from (Fagin, Halpern, and Megiddo 1990) (abbreviated by FHM in the following). The main idea is based on KB-types which are maximal consistent subsets of the set of all KBs mentioned in a PKB. Intuitively, a type is a propositional abstraction of a (non-probabilistic) interpretation. To obtain a probabilistic model one has to choose a set of possible worlds and assign types and probabilities to them such that the polynomial inequalities in the PKB are satisfied. As shown in FHM only a polynomial number of types is needed to construct a model. Based on the types and the PKB, a system of polynomial inequalities can be constructed in a way that a solution of this system, if it exists, provides the probabilities we can assign to the types. Thus, the decision procedure computes KB-types by checking consistency of KBs formulated in $\mathcal{ALCO}$ which is in ExpTime, and it checks systems of polynomial inequalities for satisfiability over the real numbers which can be done in PSpace (Canny 1988). In this way, the PSpace procedure from FHM for the propositional case extends to a ExpTime procedure for our PKBs.

ExpTime-hardness follows from the hardness of deciding consistency of an ordinary non-probabilistic KB. Let φ be a KB. It holds that φ is consistent iff the PKB

$$(\mathbf{B}\varphi) > 0$$

is consistent. We obtain the following result.

Theorem 8. *The problem of checking consistency for PKBs is* EXPTIME-*complete.*

Stochastic Actions and Projection Queries

In this section, we define syntax and semantics of an *action theory* and define a query language for talking about degrees of belief *after* actions.

Action Description

Similar to a STRIPS-like language the domain designer provides a complete add- and delete-list of literals for each action. In our case, KBs in $\mathcal{ALCO}$ are used as effect conditions. The semantics of an action is then defined in terms of interpretation updates.

First, the notion of *effect descriptions* is defined.

Definition 9. Let $A \in N_C$ be a concept name, $r \in N_R$ a role name, $o, o' \in N_I$ individual names and φ a KB. An *effect description* (*effect* for short) has one of the following forms

$$\varphi \triangleright \langle A(o) \rangle^+, \varphi \triangleright \langle r(o, o') \rangle^+ \text{ (called } add\text{-}effect),$$

$$\varphi \triangleright \langle A(o) \rangle^-, \varphi \triangleright \langle r(o, o') \rangle^- \text{ (called } delete\text{-}effect),$$

where φ is called *effect condition*. In case the effect condition φ is a tautology like for example $\top \sqsubseteq \top$, then the effect description is called *unconditional* and is written without the effect condition. We use the symbol | to denote an unconditional effect.

In an *action theory*, we distinguish between a *primitive action* that is deterministic and is associated with a set of effects, and a *stochastic action* that is associated with a finite set of primitive actions describing all possible outcomes of this stochastic action. In addition, each stochastic action is equipped with a probability distribution over the set of outcomes.

Definition 10. An *action theory* is a tuple of the form

$$\Sigma = (A_P, A_S, \text{Eff}, \text{Out}, (\ell_{\boldsymbol{\alpha}})_{\boldsymbol{\alpha} \in A_S}),$$

where

- A_P is a finite set of *primitive action names*;
- A_S is a finite set of *stochastic action names*;
- Eff maps each primitive action name $\alpha \in A_P$ to a finite set of effects $\text{Eff}(\alpha)$;
- Out maps each stochastic action $\boldsymbol{\alpha} \in A_S$ to a set of primitive actions $\text{Out}(\boldsymbol{\alpha}) \subseteq A_P$, and
- for each $\boldsymbol{\alpha} \in A_S$ there is a probability distribution $\ell_{\boldsymbol{\alpha}}$ over $\text{Out}(\boldsymbol{\alpha})$.

In the following, the symbols (possibly indexed or primed) α, β stand for primitive action names, the bold symbols $\boldsymbol{\alpha}, \boldsymbol{\beta}$ for stochastic action names, and the symbols σ and $\boldsymbol{\sigma}$ for a sequence of primitive actions and stochastic actions, respectively. In an action theory Σ, we assume that for each $\alpha \in A_P$ and for each $\boldsymbol{\alpha} \in A_S$ the sets $\text{Eff}(\alpha)$ and $\text{Out}(\boldsymbol{\alpha})$, respectively, are explicitly provided. For each element $\alpha \in \text{Out}(\boldsymbol{\alpha})$ the likelihood value as a rational number, denoted by $\ell_{\boldsymbol{\alpha}}(\alpha)$, is also explicitly given. Note that a stochastic action $\boldsymbol{\alpha}$ can also be deterministic in case $\text{Out}(\boldsymbol{\alpha})$ is a singleton set.

Example 11. We extend the representation of the domain from Example 7 with describing some dynamic aspects using an action theory Σ.

For primitive action names $\text{eat}(\text{john}, f_1)$ and $\text{eat}(\text{john}, f_2)$ we describe how the health status of john might be affected by eating the food with the following effect set:

$$\text{Eff}(\text{eat}(\text{john}, f_1)) := \{\varphi \triangleright \langle \mathit{Healthy}(\text{john}) \rangle^-\}$$

with $\varphi := (\exists \mathit{has\text{-}allergy\text{-}to}.\exists in.\{f_1\})(\text{john})$. If john has an allergy to something that is an ingredient of f_1, then john is no longer healthy and nothing else is changed. Otherwise, if φ is not satisfied, then nothing is changed. A stochastic version of the action where it is uncertain whether an allergic reaction actually occurs or nothing happens is defined for the stochastic action name $\text{s-eat}(\text{john}, f_1)$ with

$$\text{Out}(\text{s-eat}(\text{john}, f_1)) := \{\text{eat}(\text{john}, f_1), \text{no-eff}\},$$

where $\text{Eff}(\text{no-eff}) := \emptyset$ and the likelihood of a possible allergic reaction is $\ell_{\text{s-eat}}(\text{eat}(\text{john}, f_1)) := 0.7$. The action $\text{eat}(\text{john}, f_2)$ is defined analogously. There is also a deterministic action for adding an ingredient to some food with

$$\text{Eff}(\text{add-ingr}(f_1, \text{peanut})) := \{\langle in(\text{peanut}, f_1) \rangle^+\}.$$

Thus, the action adds the pair of individuals to the interpretation of in and changes nothing else.

Projection

To formulate queries about the agent's probabilistic belief *after* the execution of an action sequence an *action modality* $[\![\cdot]\!]_\Sigma$ is written in front of belief terms. We write

$$[\![\boldsymbol{\sigma}]\!]_\Sigma \mathbf{B}\varphi$$

to denote the probabilistic degree of belief in φ *after* doing the action sequence $\boldsymbol{\sigma}$ described in the action theory Σ. Those expressions can then be used to formulate *dynamic belief inequalities* as in the static case.

Definition 12. Let $\Sigma = (A_P, A_S, \text{Eff}, \text{Out}, (\ell_{\boldsymbol{\alpha}})_{\boldsymbol{\alpha} \in A_S})$ be an action theory. A *dynamic* belief term w.r.t. Σ, denoted by B_Σ, is built according to the following syntax rule

$$B_\Sigma ::= 0 \mid 1 \mid [\![\boldsymbol{\sigma}]\!]_\Sigma \mathbf{B}\varphi \mid B_\Sigma + B_\Sigma \mid B_\Sigma \times B_\Sigma,$$

where φ stands for an objective KB and $\boldsymbol{\sigma} \in A_S^*$ for a sequence of stochastic action names from Σ. With *dynamic belief terms* as primitives *dynamic belief inequalities* over Σ and *dynamic probabilistic KBs* (*dPKB* for short) over Σ are defined accordingly as for the static case in Definition 4.

The semantics of the actions enclosed in the modality $[\![\cdot]\!]_\Sigma$ is defined in terms of *updates of probabilistic interpretations*. We start with defining how a non-probabilistic interpretation is updated given a set of effect descriptions.

Definition 13. Let $\mathcal{I} = (\Delta_{\mathcal{I}}, \cdot^{\mathcal{I}})$ be an interpretation and L a set of unconditional effects. The *update of $\mathcal{I}$ with* L is an interpretation denoted by $\mathcal{I}^L$ and is defined as follows

- $\Delta_{\mathcal{I}^L} := \Delta_{\mathcal{I}}$;
- $A^{\mathcal{I}^L} := A^{\mathcal{I}} \setminus \{a^{\mathcal{I}} \mid \langle A(a) \rangle^- \in L\} \cup \{b^{\mathcal{I}} \mid \langle A(b) \rangle^+ \in L\}$ for all $A \in N_C$;

- $r^{\mathcal{I}^{\mathsf{L}}} := r^{\mathcal{I}} \setminus \{(a^{\mathcal{I}}, b^{\mathcal{I}}) \mid \langle r(a,b) \rangle^- \in \mathsf{L}\} \cup$
  $\{(c^{\mathcal{I}}, d^{\mathcal{I}}) \mid \langle r(c,d) \rangle^+ \in \mathsf{L}\}$ for all $r \in \mathsf{N_R}$

- $a^{\mathcal{I}^{\mathsf{L}}} := a^{\mathcal{I}}$ for all $a \in \mathsf{N_I}$.

Let E be a set of (possibly conditional) effects. The *update of $\mathcal{I}$ with E*, denoted by $\mathcal{I}^{\mathsf{E}}$, is given by the update $\mathcal{I}^{\mathsf{E}(\mathcal{I})}$ with

$$\mathsf{E}(\mathcal{I}) := \{\mathsf{l} \mid (\varphi \triangleright \mathsf{l}) \in \mathsf{E}, \mathcal{I} \models \varphi\}.$$

The semantics respects the frame assumption: the domain designer is only required to describe all the changes using effect descriptions whereas the non-changes are kept implicit.

The next step is to define how a stochastic action *updates a probabilistic interpretation*.

Definition 14. Let $\Sigma = (\mathsf{A_P}, \mathsf{A_S}, \mathsf{Eff}, \mathsf{Out}, (\ell_{\boldsymbol{\alpha}})_{\boldsymbol{\alpha} \in \mathsf{A_S}})$ be an action theory, $\boldsymbol{\alpha} \in \mathsf{A_S}$ a stochastic action name and

$$\mathfrak{I} = (\Delta, W, (\mathcal{I}_w)_{w \in W}, \mu)$$

a probabilistic interpretation. The *update of $\mathfrak{I}$ with $\boldsymbol{\alpha}$ w.r.t. Σ* is a probabilistic interpretation of the form

$$\mathfrak{I}^{\Sigma(\boldsymbol{\alpha})} = (\Delta, W^{\boldsymbol{\alpha}}, (\mathcal{J}_w)_{w \in W^{\boldsymbol{\alpha}}}, \mu^{\boldsymbol{\alpha}})$$

such that the following conditions are satisfied

- $W^{\boldsymbol{\alpha}} = \bigcup_{w \in W} \{w[\alpha] \mid \alpha \in \mathsf{Out}(\boldsymbol{\alpha})\}$;

- for each $w[\beta] \in W^{\boldsymbol{\alpha}}$ (with $w \in W$ and $\beta \in \mathsf{Out}(\boldsymbol{\alpha})$) we have
  $$\mathcal{J}_{w[\beta]} = (\mathcal{I}_w)^{\mathsf{Eff}(\beta)};$$

- for each $w[\beta] \in W^{\boldsymbol{\alpha}}$ (with $w \in W$ and $\beta \in \mathsf{Out}(\boldsymbol{\alpha})$) we have
  $$\mu^{\boldsymbol{\alpha}}(\{w[\beta]\}) = \ell_{\boldsymbol{\alpha}}(\beta) \times \mu(\{w\}).$$

Let $\boldsymbol{\sigma} \in \mathsf{A_S}^*$ be a sequence of stochastic action names. The *update of $\mathfrak{I}$ with $\boldsymbol{\sigma}$ w.r.t. Σ*, denoted by $\mathfrak{I}^{\Sigma(\boldsymbol{\sigma})}$, is defined by induction on the length of $\boldsymbol{\sigma}$ in the obvious way.

For each possible world $w \in W$, *before* executing $\boldsymbol{\alpha}$ and each possible outcome $\alpha \in \mathsf{Out}(\boldsymbol{\alpha})$ of $\boldsymbol{\alpha}$ there is a new possible world denoted by $w[\alpha]$ *after* doing $\boldsymbol{\alpha}$. The (non-probabilistic) interpretation associated to $w[\alpha]$ is obtained by updating the interpretation associated to w with the effects of α. The probability of w given by μ is split up among the successor worlds according to the likelihood of the respective outcome.

Let $\alpha_1 \alpha_2 \cdots \alpha_n \in (\mathsf{A_P})^*$ be a sequence of primitive action names and $w \in W$ a possible word in some probabilistic interpretation. We write $w[\alpha_1 \alpha_2 \cdots \alpha_n]$ to denote the world after doing the sequence $\alpha_1 \alpha_2 \cdots \alpha_n$ instead of $w[\alpha_1][\alpha_2] \cdots [\alpha_n]$. Note that the update does not change the domain.

Now, we are ready to define the semantics of dynamic probabilistic KBs over action theories. To evaluate a dynamic belief term of the form $[\![\boldsymbol{\sigma}]\!]_\Sigma \mathbf{B}\varphi$ in a probabilistic interpretation $\mathfrak{I}$, we evaluate $\mathbf{B}\varphi$ in its corresponding updated version $\mathfrak{I}^{\Sigma(\boldsymbol{\sigma})}$.

Definition 15. Let $\Sigma = (\mathsf{A_P}, \mathsf{A_S}, \mathsf{Eff}, \mathsf{Out}, (\ell_{\boldsymbol{\alpha}})_{\boldsymbol{\alpha} \in \mathsf{A_S}})$ be an action theory, B_Σ a dynamic belief term over Σ and $\mathfrak{I} = (\Delta, W, (\mathcal{I}_w)_{w \in W}, \mu)$ a probabilistic interpretation. By induction on the structure of B_Σ we define a *function* $\cdot^{\mathfrak{I}}$ that

maps B_Σ to its *degree of belief in $\mathfrak{I}$* denoted by $(\mathsf{B}_\Sigma)^{\mathfrak{I}}$. If B_Σ is of the form $[\![\boldsymbol{\sigma}]\!]_\Sigma \mathbf{B}\varphi$ for some $\boldsymbol{\sigma} \in \mathsf{A_S}^*$ and some KB φ, then we define

$$([\![\boldsymbol{\sigma}]\!]_\Sigma \mathbf{B}\varphi)^{\mathfrak{I}} := (\mathbf{B}\varphi)^{\mathfrak{I}^{\Sigma(\boldsymbol{\sigma})}}. \tag{1}$$

The remaining cases are as in Definition 6. Likewise, for a dPKB ψ over Σ, the definition of *satisfaction of ψ in $\mathfrak{I}$*, denoted by $\mathfrak{I} \models_\Sigma \psi$, extends to the dynamic case in the obvious way.

Next, we are ready to define the *projection problem*.

Definition 16. Let $\Sigma = (\mathsf{A_P}, \mathsf{A_S}, \mathsf{Eff}, \mathsf{Out}, (\ell_{\boldsymbol{\alpha}})_{\boldsymbol{\alpha} \in \mathsf{A_S}})$ be an action theory, $\mathcal{K}$ a PKB representing the *initial beliefs* and ψ a dPKB over Σ representing the *projection query*.

We say that ψ is *satisfiable* w.r.t. $\mathcal{K}$ iff there exists a probabilistic interpretation $\mathfrak{I}$ such that $\mathfrak{I} \models \mathcal{K}$ and $\mathfrak{I} \models_\Sigma \psi$. We say that ψ is *entailed* by $\mathcal{K}$, denoted by $\mathcal{K} \models_\Sigma \psi$ iff every probabilistic interpretation $\mathfrak{I}$ with $\mathfrak{I} \models \mathcal{K}$ also satisfies $\mathfrak{I} \models_\Sigma \psi$.

It holds that ψ is entailed by $\mathcal{K}$ iff $\neg\psi$ is not satisfiable w.r.t. $\mathcal{K}$. Thus, we focus on the satisfiability problem in the following.

Example 17. For an ontology $\mathcal{T}$ (given as a conjunction of concept inclusions) that is a known part of the initial PKB and an action sequence $\boldsymbol{\sigma}$ one might want to check whether the agent's belief in the ontology is preserved after executing $\boldsymbol{\sigma}$ using the query

$$([\![\boldsymbol{\sigma}]\!]_\Sigma(\mathbf{B}\mathcal{T})) = 1.$$

Next, let $\mathcal{K}$ be the PKB from Example 7 and Σ the action theory from Example 11. For john it is more healthy to eat $\mathsf{f_1}$ than $\mathsf{f_2}$:

$$\mathcal{K} \models_\Sigma ([\![\mathsf{eat}(\mathsf{john}, \mathsf{f_1})]\!]_\Sigma \mathbf{B}) > ([\![\mathsf{eat}(\mathsf{john}, \mathsf{f_2})]\!]_\Sigma \mathbf{B})$$

with $\mathbf{B} := \mathbf{B}(Healthy(\mathsf{john}))$, because in $\mathcal{K}$ it is more likely that $\mathsf{f_1}$ does not contain any allergens that can cause the effect condition of the eat action to be true. This might change if peanut is added as an ingredient to $\mathsf{f_1}$. We can for example formulate that

$$[\![\boldsymbol{\sigma}]\!]_\Sigma \mathbf{B}(\neg Healthy(\mathsf{john})) \geq 0.6$$

with $\boldsymbol{\sigma} = \mathsf{add\text{-}ingr}(\mathsf{f_1}, \mathsf{peanut})\, \mathsf{s\text{-}eat}(\mathsf{john}, \mathsf{f_1})$.

Using projection queries one can directly compare the degrees of belief after different action sequences. This is useful to represent the behavior of an agent that has to make a decision among different action alternatives. It is possible to formulate both qualitative and quantitative statements.

Deciding the Projection Problem

In this section, we show a 2EXPTIME upper bound for the projection problem.

In the reasoning about actions literature there are two major approaches for solving projection. One is called *progression* where one updates the initial KB such that it only talks about the situation after doing the given action sequence and then checks whether the projection query is entailed by

the progressed KB. The reversed method is called *regression* that rewrites the projection query through the action sequence into a query about the initial situation and then answers it w.r.t. the initial KB. The problem with pure regression (as it can be done with axioms in a basic action theory of the Situation Calculus (Reiter 2001)) is that due to conditional effects it can cause an avoidable exponential blow-up of the query even in the deterministic non-stochastic case with a single action sequence (Gu and Soutchanski 2010). Pure progression is also problematic in our setting because the query possibly mentions different dynamic belief terms involving different action sequences that need to be evaluated in the same probabilistic interpretation.

To deal with these problems we use a combined approach with two steps that are briefly outlined in the following before we discuss them in more detail. The overall goal is to reduce the projection problem to the consistency problem for static PKBs.

1. We first put aside the probabilities in the input of the problem and view stochastic actions as non-deterministic choices between primitive deterministic actions that in turn describe updates on non-probabilistic interpretations. The execution of a sequence of non-deterministic updates in an interpretation then yields a tree of interpretations with one node for each execution step. Those interpretation trees are encoded into a single model of a tree-shaped reduction $\mathcal{ALCO}$-KB that is constructed by introducing time-stamped copies for each relevant subformula in the input and for each execution step. The construction is very similar to the reduction approach that was first introduced in (Baader et al. 2005) with the difference that we now deal with several trees of updates rather than just one sequence. This step corresponds to the progression part of our approach.

2. To deal with the probabilities in the initial belief state we simply take the initial PKB and replace each objective subformula in it by the copy that refers to the root of the execution tree. With the projection query we proceed as follows: each dynamic belief term of the form $[\![\boldsymbol{\sigma}]\!]_\Sigma \mathbf{B}\varphi$ in the query is replaced by a sum of static belief terms. There is one static belief term for each leaf of the execution tree that uses the corresponding copies at the leaf. Each of the static belief terms in the sum has a factor that is given by the likelihood of the primitive action sequence leading to that leaf. This way we split up the initial probability among the different possible outcomes. The overall construction yields a static PKB that is consistent iff the projection query is satisfiable w.r.t. the initial PKB $\mathcal{K}$. This problem is then decidable.

Construction of the Reduction KB

For a given action theory $\Sigma = (\mathsf{A_P}, \mathsf{A_S}, \mathsf{Eff}, \mathsf{Out}, (\ell_{\boldsymbol{\alpha}})_{\boldsymbol{\alpha} \in \mathsf{A_S}})$, initial PKB $\mathcal{K}$ and projection query ψ (dPKB over Σ) we construct a PKB that is consistent iff ψ is satisfiable w.r.t. $\mathcal{K}$. In this subsection, we describe the construction of an objective $\mathcal{ALCO}$-KB $\mathsf{Trees}(\Sigma, \mathcal{K}, \psi)$ encoding trees of interpretations.

First, some auxiliary notions are introduced. With $\mathsf{Seq}(\psi)$

we denote the set of all sequences of stochastic action names mentioned in ψ. Let $\boldsymbol{\sigma} \in (\mathsf{A_S})^*$ be a sequence of length at least one. We define the set of outcomes of $\boldsymbol{\sigma}$, denoted by $\mathsf{Out}(\boldsymbol{\sigma})$, as a subset of $(\mathsf{A_P})^*$ by induction on the length of $\boldsymbol{\sigma}$. In case $\boldsymbol{\sigma}$ is a single action, $\mathsf{Out}(\boldsymbol{\sigma})$ is already defined in the action theory. For a sequence longer than one we have

$$\mathsf{Out}(\boldsymbol{\sigma}' \cdot \boldsymbol{\alpha}) := \{\sigma \cdot \alpha \mid \sigma \in \mathsf{Out}(\boldsymbol{\sigma}'), \alpha \in \mathsf{Out}(\boldsymbol{\alpha})\}.$$

Let $\sigma \in (\mathsf{A_P})^*$ be a sequence of primitive action names. With $\mathsf{pref}(\sigma)$ we denote the *set of all prefixes* of σ including the empty sequence denoted by ϵ. Let $\boldsymbol{\sigma}$ be a sequence of stochastic action names. We define

$$\mathsf{Pref}(\boldsymbol{\sigma}) := \bigcup_{\sigma \in \mathsf{Out}(\boldsymbol{\sigma})} \mathsf{pref}(\sigma).$$

Thus, for each $\boldsymbol{\sigma} \in \mathsf{Seq}(\psi)$ we obtain a tree $\mathsf{Pref}(\boldsymbol{\sigma})$ where the nodes are sequences of primitive deterministic actions.

With Ind we denote the finite set of individual names mentioned in Σ and in ψ and with $\mathsf{sub}(\Sigma, \mathcal{K}, \psi)$ we denote the set of all subconcepts occurring in the input, i.e. in $\mathcal{K}$, in some effect condition of some action from $\mathsf{A_P}$ and in ψ.

We now adopt the *reduction approach* introduced in (Baader et al. 2005). For each possible execution step, each relevant concept name, role name and subconcept a new name is introduced as follows.

- For each concept name A occurring in Σ, $\mathcal{K}$, or ψ and each $\rho \in \mathsf{Pref}(\boldsymbol{\sigma})$ for some $\boldsymbol{\sigma} \in \mathsf{Seq}(\psi)$ there is a new name $A^{(\rho)}$. The name $A^{(\epsilon)}$ represents the initial extension of A and $A^{(\rho)}$ with $\rho \neq \epsilon$ represents the set of individual names from Ind that belong to A after executing ρ. Similarly, for each role name r mentioned in the input a new role name $r^{(\rho)}$ is introduced.

- For each $C \in \mathsf{sub}(\Sigma, \mathcal{K}, \psi)$ and each $\rho \in \mathsf{Pref}(\boldsymbol{\sigma})$ for some $\mathsf{Seq}(\psi)$ a new concept name $T_C^{(\rho)}$ is introduced representing the extension of C after an update with the effects of ρ.

- A new concept name N is introduced representing the set of all individual names from Ind.

Let φ be a KB that mentions only concepts from $\mathsf{sub}(\Sigma, \mathcal{K}, \psi)$ and $\rho \in \mathsf{Pref}(\boldsymbol{\sigma})$ for some $\mathsf{Seq}(\psi)$ a sequence. With

$$\varphi^{(\rho)}$$

we denote the KB that is obtained from φ by replacing each concept C occurring in φ by $T_C^{(\rho)}$.

The new names of the form $T_C^{(\rho)}$ are defined in a KB named φ_{copy} which is a conjunction of CIs defined in the same fashion as in (Baader et al. 2005). For example, for a concept name A the definition (as a conjunction of two CIs) is as follows:

$$T_A^{(\rho)} \equiv \left(N \sqcap A^{(\rho)} \right) \sqcup \left(\neg N \sqcap A^{(\epsilon)} \right).$$

Thus, we make use of the fact that only named individuals are changed by primitive actions. The other definitions can be found in the technical report.

To represent the updates caused by an action execution we define for each $\rho \cdot \alpha \in \mathsf{Pref}(\boldsymbol{\sigma})$ for some $\boldsymbol{\sigma} \in \mathsf{Seq}(\psi)$ a

KB as follows. For add-effects on a role name we have the axiom

$$\bigwedge_{(\varphi \triangleright \langle P(a,b)\rangle^+) \in \mathsf{Eff}(\alpha)} \varphi^{(\rho)} \to \left(r^{(\rho \cdot \alpha)}(a,b) \right).$$

The KB denoted by $\varphi_{\mathsf{eff}}^{\rho \cdot \alpha}$ consists of such a conjunct for each case of add-effects and delete-effects on role names and concept names. In a KB named $\varphi_{\mathsf{min}}^{\rho \cdot \alpha}$ for each $\rho \cdot \alpha \in \mathsf{Pref}(\boldsymbol{\sigma})$ the frame assumption is explicitly encoded for all the named individuals in the input. To represent the tree of all possible outcomes of the sequences in $\mathsf{Seq}(\psi)$ we now obtain the following KB:

$$\mathsf{Trees}(\Sigma, \mathcal{K}, \psi) := \varphi_{\mathsf{copy}} \wedge$$
$$\bigwedge_{\boldsymbol{\sigma} \in \mathsf{Seq}(\psi)} \bigwedge_{\rho \cdot \alpha \in \mathsf{Pref}(\boldsymbol{\sigma})} \left(\varphi_{\mathsf{eff}}^{\rho \cdot \alpha} \wedge \varphi_{\mathsf{min}}^{\rho \cdot \alpha} \right). \tag{2}$$

To show some properties of $\mathsf{Trees}(\Sigma, \mathcal{K}, \psi)$ we need another auxiliary notion regarding the execution of a sequence of primitive actions. Let $\mathcal{I}$ be an interpretation and $\sigma \in (\mathsf{A_P})^*$ a sequence of actions. With $\mathsf{Eff}(\mathcal{I}, \sigma)$ we denote the set of unconditional effects we obtain if σ is executed in $\mathcal{I}$. The set is defined by accumulating the individual sets of effects.

Lemma 18. *Let $\boldsymbol{\sigma} \in (\mathsf{A_S})^*$ be a sequence of stochastic action names, $\mathfrak{I} = (\Delta, W, (\mathcal{I}_w)_{w \in W}, \mu)$ a probabilistic interpretation and $\mathfrak{I}^{\Sigma(\boldsymbol{\sigma})} = (\Delta, W^{\boldsymbol{\sigma}}, (\mathcal{J}_w)_{w \in W^{\boldsymbol{\sigma}}}, \mu^{\boldsymbol{\sigma}})$ the update of $\mathfrak{I}$ with $\boldsymbol{\sigma}$ w.r.t. Σ. The following is true. For every $w[\sigma] \in W^{\boldsymbol{\sigma}}$ with $\sigma \in \mathsf{Out}(\boldsymbol{\sigma})$ it holds that $\mathcal{J}_{w[\sigma]} = (\mathcal{I}_w)^{\mathsf{Eff}(\mathcal{I}_w, \sigma)}$.*

We now characterize the models of $\mathsf{Trees}(\Sigma, \mathcal{K}, \psi)$. Note that the models of $\mathsf{Trees}(\Sigma, \mathcal{K}, \psi)$ are ordinary non-probabilistic interpretations.

Lemma 19. *For every interpretation $\mathcal{I} = (\Delta_{\mathcal{I}}, \cdot^{\mathcal{I}})$ there exists an interpretation $\mathcal{J} = (\Delta_{\mathcal{J}}, \cdot^{\mathcal{J}})$ with $\mathcal{J} \models \mathsf{Trees}(\Sigma, \mathcal{K}, \psi)$ and $\Delta_{\mathcal{J}} = \Delta_{\mathcal{I}}$ such that*

$$\mathcal{I}^{\mathsf{Eff}(\mathcal{I}, \rho)} \models \varphi \text{ iff } \mathcal{J} \models \varphi^{(\rho)}$$

is true for any $\rho \in \mathsf{Pref}(\boldsymbol{\sigma})$ for some $\boldsymbol{\sigma} \in \mathsf{Seq}(\psi)$ and any KB φ that mentions only concepts from $\mathsf{sub}(\Sigma, \mathcal{K}, \psi)$. The other direction is true as well: for every interpretation $\mathcal{J} = (\Delta_{\mathcal{J}}, \cdot^{\mathcal{J}})$ with $\mathcal{J} \models \mathsf{Trees}(\Sigma, \mathcal{K}, \psi)$ there exists an interpretation $\mathcal{I} = (\Delta_{\mathcal{I}}, \cdot^{\mathcal{I}})$ with $\Delta_{\mathcal{I}} = \Delta_{\mathcal{J}}$ such that

$$\mathcal{J} \models \varphi^{(\rho)} \text{ iff } \mathcal{I}^{\mathsf{Eff}(\mathcal{I}, \rho)} \models \varphi$$

is true for any $\rho \in \mathsf{Pref}(\boldsymbol{\sigma})$ for some $\boldsymbol{\sigma} \in \mathsf{Seq}(\psi)$ and any KB φ that mentions only concepts from $\mathsf{sub}(\Sigma, \mathcal{K}, \psi)$.

Incorporating the Probabilities

In the next step of the reduction, we integrate $\mathsf{Trees}(\Sigma, \mathcal{K}, \psi)$ with the belief inequalities in the initial PKB $\mathcal{K}$ of Σ.

Let $\mathcal{K}$ be the initial PKB. With $\mathcal{K}^{(\epsilon)}$ we denote the PKB that is obtained from $\mathcal{K}$ by replacing each belief term $\mathbf{B}\varphi$ occurring in $\mathcal{K}$ with the belief term $\mathbf{B}(\varphi^{(\epsilon)})$. Furthermore, we define

$$\mathsf{Init}(\mathcal{K}) := \mathcal{K}^{(\epsilon)} \wedge \mathbf{B}(\mathsf{Trees}(\Sigma, \mathcal{K}, \psi)) = 1.$$

Next, we rewrite the projection query.

The likelihood distribution $\ell_{\boldsymbol{\alpha}}$ is defined as a probability distribution over $\mathsf{Out}(\boldsymbol{\alpha})$ for each single stochastic action $\boldsymbol{\alpha} \in \mathsf{A_S}$. First, we define inductively a corresponding distribution $\ell_{\boldsymbol{\sigma}}$ over $\mathsf{Out}(\boldsymbol{\sigma})$ for a sequence $\boldsymbol{\sigma}$. In case $\boldsymbol{\sigma}$ is a single action $\ell_{\boldsymbol{\sigma}}$ is already defined. For each $\sigma' \cdot \alpha \in \mathsf{Out}(\boldsymbol{\sigma}' \cdot \boldsymbol{\alpha})$ we have

$$\ell_{\boldsymbol{\sigma}' \cdot \boldsymbol{\alpha}}(\sigma' \cdot \alpha) := \ell_{\boldsymbol{\sigma}'}(\sigma') \times \ell_{\boldsymbol{\alpha}}(\alpha).$$

Let ψ be the projection query (dPKB over Σ). With

$$\mathsf{Red}(\psi)$$

we denote the PKB that is obtained from ψ by replacing each occurring dynamic belief term of the form $[\![\boldsymbol{\sigma}]\!]_\Sigma \mathbf{B}\varphi$ by the following static belief term

$$\sum_{\sigma \in \mathsf{Out}(\boldsymbol{\sigma})} \ell_{\boldsymbol{\sigma}}(\sigma) \times \mathbf{B}\left(\varphi^{(\sigma)} \right).$$

The next lemma formulates the correctness.

Lemma 20. *ψ is satisfiable w.r.t. $\mathcal{K}$ iff $\mathsf{Init}(\mathcal{K}) \wedge \mathsf{Red}(\psi)$ is consistent.*

Proof. We only sketch the proof of one direction. Assume that ψ is satisfiable w.r.t. $\mathcal{K}$. By definition there exists a probabilistic interpretation $\mathfrak{I} = (\Delta, W, (\mathcal{I}_w)_{w \in W}, \mu)$ such that

$$\mathfrak{I} \models \mathcal{K} \text{ and } \mathfrak{I} \models_\Sigma \psi.$$

Using Lemma 19 it can be shown that there exists a probabilistic interpretation

$$\mathfrak{I}' = (\Delta, W, (\mathcal{Y}_w)_{w \in W}, \mu)$$

that differs from $\mathfrak{I}$ only in the non-probabilistic interpretations associated to the possible worlds such that $\mathfrak{I}' \models \mathsf{Init}(\mathcal{K})$ is true and for all $w \in W$, all KBs φ occurring in ψ and all $\sigma \in \mathsf{Out}(\boldsymbol{\sigma})$ it holds that

$$\mathcal{Y}_w \models \varphi^{(\sigma)} \text{ iff } (\mathcal{I}_w)^{\mathsf{Eff}(\mathcal{I}_w, \sigma)} \models \varphi. \tag{3}$$

Let $[\![\boldsymbol{\sigma}]\!]_\Sigma \mathbf{B}\varphi$ be a dynamic belief term occurring in ψ. In the following we show that

$$([\![\boldsymbol{\sigma}]\!]_\Sigma \mathbf{B}\varphi)^{\mathfrak{I}} = \left(\sum_{\sigma \in \mathsf{Out}(\boldsymbol{\sigma})} \ell_{\boldsymbol{\sigma}}(\sigma) \times \mathbf{B}\left(\varphi^{(\sigma)} \right) \right)^{\mathfrak{I}'}$$

is true which then implies $\mathfrak{I}' \models \mathsf{Init}(\mathcal{K}) \wedge \mathsf{Red}(\psi)$. Let

$$\mathfrak{I}^{\Sigma(\boldsymbol{\sigma})} = (\Delta, W^{\boldsymbol{\sigma}}, (\mathcal{J}_w)_{w \in W^{\boldsymbol{\sigma}}}, \mu^{\boldsymbol{\sigma}})$$

be the update of $\mathfrak{I}$ with $\boldsymbol{\sigma}$ w.r.t. Σ. From Lemma 18 it follows that

$$W^{\boldsymbol{\sigma}} = \{w[\sigma] \mid w \in W, \sigma \in \mathsf{Out}(\boldsymbol{\sigma})\}. \tag{4}$$

For every $w[\sigma] \in W^{\boldsymbol{\sigma}}$ with $\sigma \in \mathsf{Out}(\boldsymbol{\sigma})$ and $w \in W$ it holds that

$$\mathcal{J}_{w[\sigma]} = (\mathcal{I}_w)^{\mathsf{Eff}(\mathcal{I}_w, \sigma)} \wedge \mu^{\boldsymbol{\sigma}}(\{w[\sigma]\}) = \ell_{\boldsymbol{\sigma}}(\sigma) \times \mu(\{w\}). \tag{5}$$

It holds that

$$([\![\boldsymbol{\sigma}]\!]_\Sigma \mathbf{B}\varphi)^{\mathfrak{I}} = (\mathbf{B}\varphi)^{\mathfrak{I}^{\Sigma(\boldsymbol{\sigma})}}$$

$$= \mu^{\boldsymbol{\sigma}} \left(\{ w \in W^{\boldsymbol{\sigma}} \mid \mathcal{J}_w \models \varphi \} \right)$$

$$= \mu^{\boldsymbol{\sigma}} \left(\bigcup_{\sigma \in \mathsf{Out}(\boldsymbol{\sigma})} \{ w[\sigma] \in W^{\boldsymbol{\sigma}} \mid \mathcal{J}_{w[\sigma]} \models \varphi \} \right) \quad \text{with (4)}$$

$$= \sum_{\sigma \in \mathsf{Out}(\boldsymbol{\sigma})} \mu^{\boldsymbol{\sigma}} \left(\{ w[\sigma] \in W^{\boldsymbol{\sigma}} \mid \mathcal{J}_{w[\sigma]} \models \varphi \} \right)$$

$$= \sum_{\sigma \in \mathsf{Out}(\boldsymbol{\sigma})} \ell_{\boldsymbol{\sigma}}(\sigma) \times \mu \left(\{ w \in W \mid (\mathcal{I}_w)^{\mathsf{Eff}(\mathcal{I}_w, \sigma)} \models \varphi \} \right) \text{ with }$$
(5)

$$= \sum_{\sigma \in \mathsf{Out}(\boldsymbol{\sigma})} \ell_{\boldsymbol{\sigma}}(\sigma) \times \mu \left(\{ w \in W \mid \mathcal{Y}_w \models \varphi^{(\sigma)} \} \right) \quad \text{with (3)}$$

$$= \sum_{\sigma \in \mathsf{Out}(\boldsymbol{\sigma})} \ell_{\boldsymbol{\sigma}}(\sigma) \times \left(\mathbf{B} \left(\varphi^{(\sigma)} \right) \right)^{\mathfrak{I}'}$$

$$= \left(\sum_{\sigma \in \mathsf{Out}(\boldsymbol{\sigma})} \ell_{\boldsymbol{\sigma}}(\sigma) \times \mathbf{B} \left(\varphi^{(\sigma)} \right) \right)^{\mathfrak{I}'}$$

$\square$

To decide the projection problem we need to compute the PKB $\mathsf{Init}(\mathcal{K}) \wedge \mathsf{Red}(\psi)$ and check it for consistency.

The *size of the input of the projection problem* is the sum of the size of the action theory, the length of the action sequence and the length of the projection query. The reduction KB $\mathsf{Init}(\mathcal{K}) \wedge \mathsf{Red}(\psi)$ is at most exponentially large in the size of the input. It consists of exponentially many symbols in the length of the action sequence. Therefore, the consistency check requires double-exponential time.

Theorem 21. *The projection problem is decidable in* 2EXPTIME.

In case of deterministic actions, i.e. if $\mathsf{Out}(\boldsymbol{\alpha})$ is a singleton set for each stochastic action $\boldsymbol{\alpha}$, then the reduction knowledge base $\mathsf{Init}(\mathcal{K}) \wedge \mathsf{Red}(\psi)$ is of polynomial size and the projection problem can be decided in EXPTIME.

Corollary 22. *The projection problem with only deterministic actions is* EXPTIME-*complete.*

Note that for the reduction linear belief inequalities are sufficient.

Hardness

In this section, we show that the projection problem is 2EXPTIME-hard. We reduce the problem of checking consistency of an $\mathcal{ALCOV}$-TBox that extends $\mathcal{ALCO}$ with *nominal schemas* (Krötzsch et al. 2011; Krötzsch and Rudolph 2014) to the projection problem. A nominal schema looks like an ordinary nominal concept but with a variable name in place of an individual that is called schema variable. The semantics is defined in terms of substitutions for a given finite set of individual names. The problem of checking whether a $\mathcal{ALCOV}$-TBox is consistent is 2EXPTIME-complete (Krötzsch and Rudolph 2014). In our reduction to the projection problem, we use primitive actions that mimick the possible variable substitutions and the stochastic actions (one for each schema variable) choose non-deterministically a possible grounding. With the projection query we then ask whether the TBox is known after doing the sequence of stochastic actions.

Reasoning with Nominal Schemas Let $\mathsf{N_V}$ be a countably infinite set of *variable names*. An $\mathcal{ALCOV}$-concept C is of the form

$$C ::= A \mid \{t\} \mid \neg C \mid C \sqcap C \mid \exists r.C$$

with $t \in \mathsf{N_V} \cup \mathsf{N_I}$. Let C be an $\mathcal{ALCOV}$-concept. A *variable mapping* ν is a function of the form $\nu : \mathsf{N_V} \to \mathsf{N_I}$. With C^ν we denote the $\mathcal{ALCO}$-concept obtained from C by replacing each variable name x in C by the individual $\nu(x)$.

An $\mathcal{ALCOV}$-*TBox* is a finite set of CIs with $\mathcal{ALCOV}$-concepts. Let $\mathcal{T}$ be an $\mathcal{ALCOV}$-TBox and Var the finite set of all variable names mentioned in $\mathcal{T}$ and Ind a finite set of individual names. Furthermore, let $\mathsf{Map}(\mathsf{Var}, \mathsf{Ind})$ be the set of all variable mappings of the form $\nu : \mathsf{N_V} \cap \mathsf{Var} \to \mathsf{N_I}$ satisfying $\nu(x) \in \mathsf{Ind}$ for all $x \in \mathsf{Var}$. The *grounding of* $\mathcal{T}$ *w.r.t.* Ind, denoted by $\mathsf{ground}(\mathcal{T}, \mathsf{Ind})$ is given by

$$\mathsf{ground}(\mathcal{T}, \mathsf{Ind}) := \{ C^\nu \sqsubseteq D^\nu \mid C \sqsubseteq D \in \mathcal{T},$$
$$\nu \in \mathsf{Map}(\mathsf{Var}, \mathsf{Ind}) \}.$$

We say that $\mathcal{T}$ is consistent w.r.t. Ind iff $\mathsf{ground}(\mathcal{T}, \mathsf{Ind})$ is consistent. Note that the set $\mathsf{ground}(\mathcal{T}, \mathsf{Ind})$ is exponentially large in the size of $\mathcal{T}$ and Ind. In (Krötzsch and Rudolph 2014), it is shown that nominal schemas raise the complexity of consistency by one exponential.

Theorem 23 ((Krötzsch and Rudolph 2014)). *Deciding consistency of an $\mathcal{ALCOV}$-TBox w.r.t. a finite set of individual names is* 2EXPTIME-*complete.*

Reduction to the Projection Problem We reduce consistency of an $\mathcal{ALCOV}$-TBox w.r.t. a finite set of individual names to the projection problem.

Let $\mathcal{T}$ be a $\mathcal{ALCOV}$-TBox, $\mathsf{Var} = \{x_1, \ldots, x_n\}$ for some $n > 0$ the finite set of variable names mentioned in $\mathcal{T}$ and $\mathsf{Ind} = \{a_1, \ldots, a_m\}$ for some $m > 0$ a finite set of individual names.

We define an action theory Σ and an initial PKB as follows. First, we introduce a set of new concept names $\{A_{x_1}, \ldots, A_{x_n}\}$ not mentioned in $\mathcal{T}$, one for each variable name in Var. Let $\mathcal{T}_A$ be the TBox that is obtained from $\mathcal{T}$ by replacing each occurrence of a nominal schema $\{x_i\}$ with $i \in \{1, \ldots, n\}$ by the concept name A_{x_i}. The initial PKB $\mathcal{K}$ is given by

$$(\mathbf{B}\mathcal{T}_A) = 1 \wedge \bigwedge_{x \in \mathsf{Var}} \mathbf{B} \left(A_x \sqsubseteq \{a_1\} \sqcup \cdots \sqcup \{a_m\} \right) = 1$$

We use the following set of primitive action names

$$\mathsf{A_P} := \{\alpha_0\} \cup \{\alpha_{x_i}^{a_j} \mid j \in \{1, \ldots, m\}, i \in \{1, \ldots, n\}\}.$$

There is one action α_0 for initialization and one action α_x^a for each pair $(a, x) \in \mathsf{Ind} \times \mathsf{Var}$. The effects are given as follows

$$\mathsf{Eff}(\alpha_0) := \{ \langle A_{x_i}(a_j) \rangle^- \mid x_i \in \mathsf{Var}, a_j \in \mathsf{Ind} \}$$
$$\mathsf{Eff}(\alpha_{x_i}^{a_j}) := \{ \langle A_{x_i}(a_j) \rangle^+ \} \text{ for all } (a_j, x_i) \in \mathsf{Ind} \times \mathsf{Var}.$$

After executing α_0 in a model of $\mathcal{K}$ all concept names A_x are interpreted as empty sets. With the execution of α_x^a we assign individual a to the variable x by making $A_x(a)$ true

and change nothing else in the interpretation. We use the following set of stochastic action names

$$\mathsf{A_S} := \{\boldsymbol{\alpha}_0\} \cup \{\boldsymbol{\alpha}_{x_1}, \ldots, \boldsymbol{\alpha}_{x_n}\}.$$

There is one stochastic action for each variable name. The outcomes are defined as follows

$$\mathsf{Out}(\boldsymbol{\alpha}_0) := \{\alpha_0\}$$
$$\mathsf{Out}(\boldsymbol{\alpha}_x) := \{\alpha_x^{a_1}, \ldots \alpha_x^{a_m}\} \text{ for each } x \in \mathsf{Var}.$$

The likelihood distribution $\ell_{\boldsymbol{\alpha}_x}$ is a uniform distribution for each $x \in \mathsf{Var}$. Executing a stochastic action $\boldsymbol{\alpha}_x$ with $x \in \mathsf{Var}$ means non-deterministically choosing an assignment for the variable x. Since the outcome of the actions $\boldsymbol{\alpha}_x$ is unknown, all possible groundings are present in the belief state after executing the action sequence. It is now straightforward to show that

Lemma 24. $\mathcal{T}$ *is consistent w.r.t.* Ind *iff*

$$[\![\boldsymbol{\alpha}_0 \boldsymbol{\alpha}_{x_1} \cdots \boldsymbol{\alpha}_{x_n}]\!]_\Sigma (\mathbf{B} \mathcal{T}_A) = 1$$

is satisfiable w.r.t. $\mathcal{K}$.

Thus, we get the following result.

Theorem 25. *The projection problem with stochastic actions is* 2EXPTIME-*complete.*

From the reduction it follows that the hardness result already holds for a purely qualitative setting. The only belief equality used is of the form $\mathbf{B}(\cdot) = 1$ that can be viewed as an S5 knowledge modality. Obviously, the proof would also work for non-deterministic but non-probabilistic actions where the outcome is uncertain for the agent. Furthermore, for the reduction unconditional effects on concept names that are used to simulate variable substitutions are sufficient.

Conclusion

We have studied the computational properties of the probabilistic projection problem in an action language based on the DL $\mathcal{ALCO}$ with stochastic actions and for projection queries where one can express both qualitative and quantitative statements about probabilistic beliefs. Our results show that in presence of beliefs and probabilistic uncertainty about the outcome of actions the complexity raises from EXPTIME in the deterministic case to 2EXPTIME. The hardness proof is done by reducing consistency of TBoxes with nominal schemas to projection. In (Krötzsch et al. 2011; Krötzsch and Rudolph 2014) it has been shown that one can encode exponentially large TBoxes using nominal schemas. Thus, the exponential blow-up of the belief state representation in the length of the action sequence seems to be unavoidable in general. However, if, for instance, the lookahead depth of the agent is limited to one, then the complexity does not increase.

Decision procedures for probabilistic projection have already been devised for propositional languages. Kooi shows decidability of *probabilistic dynamic epistemic logic* (Kooi 2003) in a multi-agent setting and provides a polynomial space lower bound and an exponential space upper bound for

the validity problem. Recently, Lang and Zanuttini (2015) have shown that the problem of verifying propositional probabilistic knowledge-based programs w.r.t. a finite horizon (for the single-agent case) is decidable in polynomial space. In (Rens, Meyer, and Lakemeyer 2014) a dynamic logic for specifying and reasoning about MDPs is proposed. Decidability is shown but the complexity is not discussed.

A Situation Calculus in a first-order modal logic with modalities for actions, probabilistic degrees of belief and introspection has been introduced in (Belle and Lakemeyer 2017). In order to formalize introspection they propose a general semantics without the restriction to only discrete probability distributions, which is an assumption we have made here and which is present in Prob-$\mathcal{ALC}$ and Halpern's logic (Halpern 1990) as well. However, how to perform effective reasoning in the language of Belle and Lakemeyer has not been studied in their paper.

For the study of the impact of probabilistic uncertainty about action outcomes on the complexity of the projection problem in this work, a relatively simple action language was already sufficient. To increase the expressiveness for use in an high-level programming language it would be however interesting to add for example context-sensitive likelihood distributions on the outcome of a stochastic actions or a representation of uncertain observations to model perception of an agent. Furthermore, verification of action programs with loops in a setting with probabilistic beliefs is an interesting topic for future work.

References

Ahmetaj, S.; Calvanese, D.; Ortiz, M.; and Simkus, M. 2017. Managing change in graph-structured data using description logics. *ACM Trans. Comput. Log.* 18(4):27:1–27:35.

Baader, F.; Calvanese, D.; McGuinness, D. L.; Nardi, D.; and Patel-Schneider, P. F., eds. 2003. *The Description Logic Handbook: Theory, Implementation, and Applications.* Cambridge University Press.

Baader, F.; Lutz, C.; Milicic, M.; Sattler, U.; and Wolter, F. 2005. Integrating description logics and action formalisms: First results. In Veloso, M. M., and Kambhampati, S., eds., *Proceedings, The Twentieth National Conference on Artificial Intelligence and the Seventeenth Innovative Applications of Artificial Intelligence Conference, July 9-13, 2005, Pittsburgh, Pennsylvania, USA*, 572–577. AAAI Press / The MIT Press.

Baader, F.; Horrocks, I.; Lutz, C.; and Sattler, U. 2017. *An Introduction to Description Logic*. Cambridge University Press.

Baader, F.; Lippmann, M.; and Liu, H. 2010. Using causal relationships to deal with the ramification problem in action formalisms based on description logics. In Fermüller, C. G., and Voronkov, A., eds., *Logic for Programming, Artificial Intelligence, and Reasoning - 17th International Conference, LPAR-17, Yogyakarta, Indonesia, October 10-15, 2010. Proceedings*, volume 6397 of *Lecture Notes in Computer Science*, 82–96. Springer.

Bacchus, F.; Halpern, J. Y.; and Levesque, H. J. 1999. Rea-

soning about noisy sensors and effectors in the situation calculus. *Artif. Intell.* 111(1-2):171–208.

Belle, V., and Lakemeyer, G. 2017. Reasoning about probabilities in unbounded first-order dynamical domains. In Sierra, C., ed., *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI 2017, Melbourne, Australia, August 19-25, 2017*, 828–836. ijcai.org.

Canny, J. F. 1988. Some algebraic and geometric computations in PSPACE. In *STOC*, 460–467. ACM.

Fagin, R.; Halpern, J. Y.; and Megiddo, N. 1990. A logic for reasoning about probabilities. *Inf. Comput.* 87(1/2):78–128.

Ferrein, A., and Lakemeyer, G. 2008. Logic-based robot control in highly dynamic domains. *Robotics and Autonomous Systems* 56(11):980–991.

Ferrein, A.; Fritz, C.; and Lakemeyer, G. 2005. Using golog for deliberation and team coordination in robotic soccer. *KI* 19(1):24.

Gabaldon, A., and Lakemeyer, G. 2007. ESP: A logic of only-knowing, noisy sensing and acting. In *Proceedings of the Twenty-Second AAAI Conference on Artificial Intelligence, July 22-26, 2007, Vancouver, British Columbia, Canada*, 974–979. AAAI Press.

Gu, Y., and Soutchanski, M. 2010. A description logic based situation calculus. *Ann. Math. Artif. Intell.* 58(1-2):3–83.

Gutiérrez-Basulto, V.; Jung, J. C.; Lutz, C.; and Schröder, L. 2017. Probabilistic description logics for subjective uncertainty. *J. Artif. Intell. Res.* 58:1–66.

Halpern, J. Y. 1990. An analysis of first-order logics of probability. *Artif. Intell.* 46(3):311–350.

Kooi, B. P. 2003. Probabilistic dynamic epistemic logic. *Journal of Logic, Language and Information* 12(4):381–408.

Krötzsch, M., and Rudolph, S. 2014. Nominal schemas in description logics: Complexities clarified. In Baral, C.; Giacomo, G. D.; and Eiter, T., eds., *Principles of Knowledge Representation and Reasoning: Proceedings of the Fourteenth International Conference, KR 2014, Vienna, Austria, July 20-24, 2014*. AAAI Press.

Krötzsch, M.; Maier, F.; Krisnadhi, A.; and Hitzler, P. 2011. A better uncle for OWL: nominal schemas for integrating rules and ontologies. In Srinivasan, S.; Ramamritham, K.; Kumar, A.; Ravindra, M. P.; Bertino, E.; and Kumar, R., eds., *Proceedings of the 20th International Conference on World Wide Web, WWW 2011, Hyderabad, India, March 28 - April 1, 2011*, 645–654. ACM.

Lang, J., and Zanuttini, B. 2015. Probabilistic knowledge-based programs. In Yang, Q., and Wooldridge, M., eds., *Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence, IJCAI 2015, Buenos Aires, Argentina, July 25-31, 2015*, 1594–1600. AAAI Press.

Levesque, H. J.; Reiter, R.; Lespérance, Y.; Lin, F.; and Scherl, R. B. 1997. GOLOG: A logic programming language for dynamic domains. *J. Log. Program.* 31(1-3):59–83.

Lutz, C., and Schröder, L. 2010. Probabilistic description logics for subjective uncertainty. In Lin, F.; Sattler, U.; and Truszczynski, M., eds., *Principles of Knowledge Representation and Reasoning: Proceedings of the Twelfth International Conference, KR 2010, Toronto, Ontario, Canada, May 9-13, 2010*. AAAI Press.

McCarthy, J., and Hayes, P. 1969. Some philosophical problems from the standpoint of artificial intelligence. In Meltzer, B., and Michie, D., eds., *Machine Intelligence 4*. New York: American Elsevier. 463–502.

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

Rens, G.; Meyer, T.; and Lakemeyer, G. 2014. SLAP: specification logic of actions with probability. *J. Applied Logic* 12(2):128–150.

Zarrieß, B., and Claßen, J. 2015. Verification of knowledge-based programs over description logic actions. In Yang, Q., and Wooldridge, M., eds., *Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence, IJCAI 2015, Buenos Aires, Argentina, July 25-31, 2015*, 3278–3284. AAAI Press.

Zarrieß, B. 2018. Complexity of projection with stochastic actions in a probabilistic description logic. Technical Report 18-05, Chair for Automata Theory, Institute of Theoretical Computer Science, Technische Universität Dresden, Dresden, Germany. see https://lat.inf.tu-dresden.de/research/reports.html.