Simple Default Reasoning in Theories of Action

Hannes Strass and Michael Thielscher

Department of Computer Science
Dresden University of Technology
{hannes.strass,mit}@inf.tu-dresden.de

Abstract. We extend a recent approach to integrate action formalisms and non-monotonic reasoning. The resulting framework allows an agent employing an action theory as internal world model to make useful default assumptions. While the previous approach only allowed for modeling static defaults, that are independent of state properties, our extension allows for the expression of dynamic defaults. Problems that arise due to the interaction of defaults with the solution of the frame problem are dealt with accordingly: we devise a general method of integrating defaults into the formal representation of action effects and show that the method prevents counter-intuitive conclusions.

1 Introduction

Recently, [1] proposed a framework for non-monotonic reasoning in theories of actions and change by embedding them into Raymond Reiter's default logic [2]. The approach presented there used atomic, normal default rules without prerequisites to express static world properties. These properties are assumed once if consistent and then persist over time until supported or refuted by a definite action effect.

In this paper, we extend that mechanism to atomic, normal default rules *with* prerequisites. They allow us to specify dynamic defaults, that is, default properties that arise and elapse with changing world features. This is, as we shall argue, most important to capture the fluctuating nature of dynamic worlds that an intelligent agent might encounter.

As a motivating scenario (and running example of the paper), consider a very simple domain with an action $\mathsf{Fold}(x)$ that turns a sheet of paper x into a paper airplane. From experience, we might be able to say that in general, paper airplanes fly. Yet, we don't want to encode this ability to fly as a definite action effect or general law; we want to retain the possibility of *exceptions*: if the obtained paper airplane is observed to be unable to fly, we do not want to get a contradiction. The extension we present here will allow us to use this kind of defeasible reasoning in theories of actions and change. We show, by means of an example, that a straightforward generalization of the approach presented in [1] to normal default rules allows for unintended default conclusions and then introduce a general, automatic method that is proven to render such conclusions impossible. Finally, we show how the idea behind this method can also be used to specify default effects of non-deterministic actions.

2 Background

This section presents the formal preliminaries of the paper. In the first subsection we familiarize the reader with a unifying action calculus that we use to logically formalize action domains, and in the second subsection we recall Raymond Reiter's default logic [2].

2.1 The Unifying Action Calculus

The action Fold of our motivating example is characterized by two sets denoting its positive and negative effects, respectively. This is the general method of specifying actions we pursue here: the stated action effects are compiled into an effect axiom that incorporates a solution to the frame problem (similar to that of [3, 4]). These effect axioms and action precondition axioms will be formulated in a unifying action calculus (UAC) that was proposed in [5] to provide a universal framework for research in reasoning about actions.

The most notable generalization established by the UAC is its abstraction from the underlying time structure: it can be instantiated with formalisms using the time structure of situations (as the Situation Calculus [6] or the Fluent Calculus [4]), as well as with formalisms using a linear time structure (like the Event Calculus [7]).

The UAC is a sorted logic language which is based on the sorts FLUENT, ACTION, and TIME along with the predicates $<$: TIME $\times$ TIME (denoting an ordering of time points), *Holds* : FLUENT $\times$ TIME (stating whether a fluent evaluates to true at a given time point), and *Poss* : ACTION $\times$ TIME $\times$ TIME (indicating whether an action is applicable for particular starting and ending time points). In this work, we assume a finite number of functions into sorts FLUENT and ACTION and uniqueness-of-names for all of them.

The following definition introduces the most important types of formulas of the unifying action calculus: they allow to express properties of states and applicability conditions and effects of actions.

Definition 1. *Let $\boldsymbol{s}$ be a sequence of variables of sort* TIME.

– *A* state formula $\Phi[\boldsymbol{s}]$ in $\boldsymbol{s}$ *is a first-order formula with free variables $\boldsymbol{s}$ where*
  - *for each occurrence of $Holds(\varphi, s)$ in $\Phi[\boldsymbol{s}]$ we have $s \in \boldsymbol{s}$ and*
  - *predicate Poss does not occur.*

Let s, t be variables of sort TIME *and A be a function into sort* ACTION.

– *A* precondition axiom *is of the form*

$$Poss(A(\boldsymbol{x}), s, t) \equiv \pi_A[s] \tag{1}$$

where $\pi_A[s]$ is a state formula in s with free variables among $s, t, \boldsymbol{x}$.

– *An* effect axiom *is of the form*

$$Poss(A(\boldsymbol{x}), s, t) \supset (\forall f)(Holds(f, t) \equiv (\gamma_A^+ \vee (Holds(f, s) \wedge \neg\gamma_A^-))) \quad (2)$$

where

$$\gamma_A^+ = \bigvee_{\varphi \in \Gamma_A^+} f = \varphi \quad \text{and} \quad \gamma_A^- = \bigvee_{\psi \in \Gamma_A^-} f = \psi$$

and Γ_A^+ and Γ_A^- are sets of terms of sort FLUENT *with free variables among $\boldsymbol{x}$ that denote the positive and negative effects of action $A(\boldsymbol{x})$.*

This definition of effect axioms is a restricted version of the original definition of [5]—it only allows for deterministic actions with unconditional effects. Extending the binary *Poss* predicate of the Situation Calculus, our ternary version $Poss(a, s, t)$ is to be read as "action a is possible starting at time s and ending at time t".

Definition 2. *A* (UAC) domain axiomatization *consists of a finite set of foundational axioms Ω (that define the underlying time structure and do not mention the predicates Holds and Poss), a set Π of precondition axioms (1), and a set Υ of effect axioms (2); the latter two for all functions into sort* ACTION.

The domain axiomatizations used here will usually also contain a set Σ_0 of state formulas that characterize the state of the world at the initial time point.

We illustrate these definitions with the implementation of the action part of our running example.

Example 3. Consider the domain axiomatization $\Sigma = \Omega_{sit} \cup \Pi \cup \Upsilon \cup \Sigma_0$, where Ω_{sit} contains the foundational axioms for situations from [8], Π contains the precondition axiom $Poss(\mathsf{Fold}(x), s, t) \equiv t = Do(\mathsf{Fold}(x), s)$, Υ contains effect axiom (2) characterized by $\Gamma^+_{\mathsf{Fold}(x)} = \{\mathsf{PaperAirplane}(x)\}$ and $\Gamma^-_{\mathsf{Fold}(x)} = \{\mathsf{SheetOfPaper}(x)\}$, and the initial state is $\Sigma_0 = \{Holds(\mathsf{SheetOfPaper}(\mathsf{P}), S_0)\}$. Using the abbreviation $S_1 = Do(\mathsf{Fold}(\mathsf{P}), S_0)$ we can now employ logical entailment to infer that after folding, the object P is no longer a sheet of paper but a paper airplane:

$$\Sigma \models Holds(\mathsf{PaperAirplane}(\mathsf{P}), S_1) \wedge \neg Holds(\mathsf{SheetOfPaper}(\mathsf{P}), S_1)$$

The next definition introduces reachability of a time point as existence of an action sequence leading to the time point. A second order formula expresses this intuition via defining the predicate *Reach* as the least set containing the minimal elements of sort TIME (the initial time points *Init*) and being closed under possible action application (via *Poss*).

Definition 4. *Let Σ be a domain axiomatization and σ be a time point.*

$$\begin{aligned} Reach(r) &\stackrel{\text{def}}{=} (\forall R)(((\forall s)(Init(s) \supset R(s)) \\ &\qquad \wedge (\forall a, s, t)(R(s) \wedge Poss(a, s, t) \supset R(t))) \supset R(r)) \\ Init(t) &\stackrel{\text{def}}{=} \neg(\exists s) s < t \end{aligned}$$

We say σ is finitely reachable *in Σ if $\Sigma \models Reach(\sigma)$.*

2.2 Default Logic

Introduced in the seminal work by Reiter [2], default logic has become one of the most important formalisms for non-monotonic reasoning. Its fundamental notion is that of *default rules*, that specify how to extend an incomplete knowledge base with vague, uncertain knowledge.

Definition 5. *A* normal default rule *(or* normal default*) is of the form* $\alpha[s]/\beta[s]$ *where* $\alpha[s]$ *and* $\beta[s]$ *are state formulas in* s : TIME*.*

A default rule is called prerequisite-free *or* supernormal *iff* $\alpha = \top$*.*

Default rules with free (non-TIME) variables are semantically taken to represent their ground instances. By $\mathcal{D}[\sigma]$ we denote the set of defaults in $\mathcal{D}[s]$ where s has been instantiated by the term σ.

Example 3 (continued). The statement "in general, paper airplanes fly" from Section 1 can easily be modeled by the default rule

$$Holds(\mathsf{PaperAirplane}(y), s)/Holds(\mathsf{Flies}(y), s) \tag{3}$$

Definition 6. *A* default theory *is a pair* $(W, \mathcal{D})$ *where* W *is a set of closed formulas and* $\mathcal{D}$ *a set of default rules.*

The set W of a default theory is the set of indefeasible knowledge that we are unwilling to give up under any circumstances.

The semantics of default logic is defined through extensions: they can be seen as a way of applying to W as many default rules from $\mathcal{D}$ as consistently possible.

Definition 7. *Let* $(W, \mathcal{D})$ *be a default theory. For any set of closed formulas* S*, define* $\Gamma(S)$ *as the smallest set such that:*

- $W \subseteq \Gamma(S)$,
- $Th(\Gamma(S)) = \Gamma(S)$[1], *and*
- *for all* $\alpha/\beta \in \mathcal{D}$*, if* $\alpha \in \Gamma(S)$ *and* $\neg\beta \notin S$*, then* $\beta \in \Gamma(S)$*.*

A set of closed formulas E *is called an* extension for $(W, \mathcal{D})$ *iff* $\Gamma(E) = E$*, that is,* E *is a fixpoint of* Γ*.*

The set of generating defaults *of an extension* E *for* $(W, \mathcal{D})$ *is*

$$gd(E) \stackrel{\text{def}}{=} \{\alpha/\beta \in \mathcal{D} \mid \alpha \in E, \neg\beta \notin E\}$$

We denote the set of all extensions for a default theory by $Ex(W, \mathcal{D})$*.*

By a result from [2], extensions are completely characterized by the consequents of their generating defaults:

[1] $Th(F)$ for a set of formulas F denotes the set of its logical consequences, i.e. $Th(F) \stackrel{\text{def}}{=} \{\varphi \mid F \models \varphi\}$.

Lemma 8 (Reiter). *Let E be an extension for $(W, \mathcal{D})$.*

$$E = Th(W \cup \{\beta \mid \alpha/\beta \in gd(E)\})$$

Based on extensions, one can define skeptical and credulous conclusions for default theories: skeptical conclusions are formulas that are contained in every extension, credulous conclusions are those that are contained in at least one extension.

Definition 9. *Let $(W, \mathcal{D})$ be a normal default theory and Ψ be a formula.*

$$W \approx_{\mathcal{D}}^{skept} \Psi \overset{\text{def}}{\equiv} \Psi \in \bigcap_{E \in Ex(W,\mathcal{D})} E, \quad W \approx_{\mathcal{D}}^{cred} \Psi \overset{\text{def}}{\equiv} \Psi \in \bigcup_{E \in Ex(W,\mathcal{D})} E$$

Example 3 (continued). Taking the indefeasible knowledge

$$W = \{Holds(\mathsf{PaperAirplane}(\mathsf{P}), S)\}$$

for a TIME constant S and $\mathcal{D}[s]$ to contain the default rule (3), we can instantiate the default with time point S and skeptically conclude that P flies:

$$W \approx_{\mathcal{D}[S]}^{skept} Holds(\mathsf{Flies}(\mathsf{P}), S)$$

2.3 Domain Axiomatizations with Supernormal Defaults

We recall the notion of a domain axiomatization with supernormal defaults[2] from [1]. It is essentially a supernormal default theory where the set containing the indefeasible knowledge is an action domain axiomatization.

Definition 10. *A* domain axiomatization with supernormal defaults *is a pair $(\Sigma, \mathcal{D}[s])$, where Σ is a UAC domain axiomatization and $\mathcal{D}[s]$ is a set of default rules of the form*

$$\top/(\neg)Holds(\psi, s)$$

where ψ is a term of sort FLUENT.

3 Domain Axiomatizations with Normal Defaults

As mentioned before, we loosen the restriction to supernormal defaults and allow default rules with prerequisites. The rest of the definition stays the same.

[2] The endorsement "supernormal" is only used in this work to distinguish the approaches.

Definition 11. *A* domain axiomatization with (normal) defaults *is a pair* $(\Sigma, \mathcal{D}[s])$*, where* Σ *is a UAC domain axiomatization and* $\mathcal{D}[s]$ *is a set of default rules of the form*

$$(\neg)Holds(\varphi, s)/(\neg)Holds(\psi, s) \quad \textit{or} \quad \top/(\neg)Holds(\psi, s)$$

where φ, ψ *are terms of sort* FLUENT.

For notational convenience, we identify *Holds* statements with the mentioned fluent and indicate negation by overlining: the default $Holds(\varphi, s)/\neg Holds(\psi, s)$, for example, will be written as $\varphi/\overline{\psi}$. Generally, $\overline{\alpha} = \neg\alpha$ and $\overline{\neg\alpha} = \alpha$. We furthermore use $|\cdot|$ to extract the affirmative component of a fluent literal, that is, $|\neg\alpha| = |\alpha| = \alpha$. Both notions generalize to sets of fluents in the obvious way.

We now show the straightforward implementation of our motivating example.

Example 3 (continued). Recall the domain axiomatization Σ from Section 2.1 and let the set of defaults $\mathcal{D}[s]$ contain the single default rule (3). We see that, after applying the action $\mathsf{Fold}(\mathsf{P})$, we can indeed infer that P flies:

$$\Sigma \mathrel{\approx\!\!\!\!\!\mid}^{skept}_{\mathcal{D}[S_1]} Holds(\mathsf{Flies}(\mathsf{P}), S_1)$$

Note that we need to instantiate the defaults with the resulting situation S_1 (instantiating the defaults with S_0 would not yield the desired result). Now taking a closer look at effect axiom (2) and its incorporated solution to the frame problem, we observe that also

$$\Sigma \mathrel{\approx\!\!\!\!\!\mid}^{skept}_{\mathcal{D}[S_1]} Holds(\mathsf{Flies}(\mathsf{P}), S_0)$$

This is because $\mathsf{Flies}(\mathsf{P})$ was not a positive effect of the action—according to the effect axiom it must have held beforehand. This second inference is unintended: first of all, the conclusion “the sheet of paper already flew before it was folded” does not correspond to our natural understanding of the example domain. The second, more subtle, reason is that we used defaults about $S_1 = Do(\mathsf{Fold}(\mathsf{P}), S_0)$ to conclude something about S_0 that could not be concluded with defaults about S_0. In practice, it would mean that to make all possible default conclusions about a time point, we had to instantiate the defaults with all future time points (of which there might be infinitely many), which is clearly infeasible.

4 Relaxing the Frame Assumption

We next extend our specification of actions—up to now only via positive and negative effects—with another set of fluents, called *occlusions* (the term first occurred in [9]; our usage of occlusions is inspired by this work). They do not fix a truth value for the respective fluents in the resulting time point of the action and thus allow them to fluctuate freely. In particular, it is then impossible to determine an occluded fluent’s truth value at the starting time point employing only information about the ending time point.

Definition 12. *An* effect axiom with unconditional effects and occlusions *is of the form*

$$Poss(A(\boldsymbol{x}), s, t) \supset (\forall f)(\gamma_A^{?} \vee (Holds(f, t) \equiv (\gamma_A^{+} \vee (Holds(f, s) \wedge \neg\gamma_A^{-})))) \quad (4)$$

where

$$\gamma_A^{+} = \bigvee_{\varphi \in \Gamma_A^{+}} f = \varphi, \quad \gamma_A^{-} = \bigvee_{\psi \in \Gamma_A^{-}} f = \psi, \quad \gamma_A^{?} = \bigvee_{\chi \in \Gamma_A^{?}} f = \chi,$$

and Γ_A^{+}, Γ_A^{-}, and $\Gamma_A^{?}$ are sets of terms of sort FLUENT *with free variables among $\boldsymbol{x}$ that denote the positive and negative effects and occlusions of action $A(\boldsymbol{x})$.*

It is easily seen that effect axiom (2) is a special case of the above effect axiom with $\gamma_A^{?} = \perp$ (i.e. $\Gamma_A^{?} = \emptyset$).

4.1 ... to Prevent Default Reasoning Backwards in Time

Example 3 (continued). Set $\Gamma_{\mathsf{Fold}(x)}^{?} := \{\mathsf{Flies}(x)\}$ and let $\Sigma' = \Omega_{sit} \cup \Pi \cup \Upsilon' \cup \Sigma_0$, where Υ' contains effect axiom (4) for the action $\mathsf{Fold}(x)$. We see that the desired conclusion is preserved, and the undesired one is now disabled:

$$\Sigma \mathrel{|\!\!\approx}_{\mathcal{D}[S_1]}^{skept} Holds(\mathsf{Flies}(\mathsf{P}), S_1) \text{ and } \Sigma \mathrel{|\!\!\not\approx}_{\mathcal{D}[S_1]}^{skept} Holds(\mathsf{Flies}(\mathsf{P}), S_0)$$

Specifying the occlusions for the action in the example was easy—there was only one default rule, and we had a precise understanding of the desired and undesired inferences. In general, however, defaults might interact and it might become less obvious which of them to exclude from the frame assumption.

Algorithm 1 below implements a general method of identifying the fluents that are to be occluded, taking into account given default rules. It takes as input positive and negative effects Γ_A^{+} and Γ_A^{-} of an action A and a set $\mathcal{D}$ of defaults and computes the set $\Gamma_A^{?\mathcal{D}}$ of default occlusions for A with respect to $\mathcal{D}$. The intuition behind it is simple: it iterates over a set S of fluents potentially influenced by A. This set is initialized with the definite action effects and then extended according to default rules until a fixpoint obtains.

Algorithm 1 Computing the default occlusions

Input: Γ_A^{+}, Γ_A^{-}, $\mathcal{D}$
Output: $\Gamma_A^{?\mathcal{D}}$

$S := \Gamma_A^{+} \cup \{\overline{\gamma} \mid \gamma \in \Gamma_A^{-}\}$ *// initialization: literals stating the definite effects*
while there is $\gamma \in S$, $\alpha/\beta \in \mathcal{D}$, a substitution θ with $\alpha\theta = \gamma$; and $\beta\theta \notin S$ **do**
  $S := S \cup \{\beta\theta\}$ *// $\beta\theta$ might become default effect of A*
end while
return $|S| \setminus (\Gamma_A^{+} \cup \Gamma_A^{-})$ *// exclude definite effects from occlusions*

Note that prerequisite-free defaults do not contribute to the computation of occlusions: the symbol $\top$ does not unify with any explicitly mentioned action effect. This behavior is semantically perfectly all right: the intended reading of prerequisite-free defaults is that of static world properties that are once assumed (if consistent) and then persist over time until an action effect either refutes or confirms them.

It is easily seen that Algorithm 1 applied to our running example creates the exact set of occlusions that we figured out earlier "by hand".

For the following theoretical results of this paper, let $(\Sigma, \mathcal{D}[s])$ be a domain axiomatization with defaults where all effect axioms are of the form (2), and let Σ' denote the domain axiomatization with effect axioms (4) where the $\Gamma^?$ are constructed by applying Algorithm 1 to each action of Σ. It should be noted that Σ' is consistent whenever Σ is consistent: default occlusions only weaken the restrictions on successor states, thus any model for Σ is a model for Σ'.

The first proposition shows that the default occlusions computed by Algorithm 1 are sound with respect to default conclusions about starting time points of actions: whenever defaults about a resulting time point can be utilized to infer a state property of the starting time point, this state property can also be inferred locally, that is, with defaults about the starting time point itself.

Lemma 13. *Let α be a ground action and σ, τ be terms of sort* TIME *such that $\Sigma' \models Poss(\alpha, \sigma, \tau)$, and let $\Psi[s]$ be a state formula.*

$$\Sigma' \approx^{skept}_{\mathcal{D}[\tau]} \Psi[\sigma] \text{ implies } \Sigma' \approx^{skept}_{\mathcal{D}[\sigma]} \Psi[\sigma]$$

Proof. (Sketch.) We prove the contrapositive. Let $\Sigma' \not\approx^{skept}_{\mathcal{D}[\sigma]} \Psi[\sigma]$. Then there exists an extension E for $(\Sigma, \mathcal{D}[\sigma])$ with $\Psi[\sigma] \notin E$. We construct an extension F for $(\Sigma, \mathcal{D}[\tau])$ as follows. By Lemma 8, E is characterized by the consequents of its generating defaults (all of which are Holds literals in σ). We determine F's characterizing set of default consequences by removing the ones that are contradicted via action effects and adding consequents of newly applicable normal defaults. All those new default conclusions are, due to the construction of $\Gamma^{?\mathcal{D}}_{\alpha}$ via Algorithm 1, backed by occlusions and do not influence σ. Thus $\Psi[\sigma] \notin F$. $\square$

The absence of unintended inferences about time points connected via a single action then immediately generalizes to time points connected via a sequence of actions and trivially generalizes to disconnected time points. This is the main result of the paper stating the impossibility of undesired default conclusions about the past.

Theorem 14. *Let σ, τ be time points such that σ is reachable and $\sigma \leq \tau$.*

$$\Sigma' \approx^{skept}_{\mathcal{D}[\tau]} \Psi[\sigma] \text{ implies } \Sigma' \approx^{skept}_{\mathcal{D}[\sigma]} \Psi[\sigma]$$

Another noteworthy property of the presented default reasoning mechanism is the preservation of default conclusions: even if the prerequisite of a default rule is invalidated due to a contradicting action effect, the associated consequent (if not also contradicted) stays intact. This means the algorithm does not occlude unnecessarily many fluents. It would be fairly easy to modify Algorithm 1 such that the resulting effect axioms also "forget" default conclusions whose generating rules have become inapplicable—we would just have to replace all occurrences of literals by their respective affirmative component.

4.2 ... to Model Default Effects of Actions

The usage of occlusions as advocated up to this point is of course not the only way to make use of this concept. When they are specified by the user along with action effects as opposed to computed automatically, occlusions are an excellent means of modeling default effects of non-deterministic actions:

Example 15 (Direct Default Effect). We model the action of tossing a coin via excluding the fluent Heads (whose intention is to denote whether heads is showing upwards after tossing the coin) from the action Toss's frame axiom, that is, $\Gamma^{?}_{\mathsf{Toss}} := \{\mathsf{Heads}\}$. However, the coin of this example is unbalanced and has a strong tendency towards landing with heads facing upwards. This is modeled by having a default that states the result Heads as "usual outcome":

$$\mathit{Holds}(\mathsf{Heads}, s) \tag{5}$$

There is another action, Wait, that is always possible and does not change the truth value of any fluent. All $\gamma^{+/-/?}$ not explicitly mentioned are thus to be taken as the empty disjunction, i.e. false. Using the domain axiomatization Σ, that contains the precondition axioms and effect axioms (4) stated above, situations as time structure, and the observation $\Sigma_O = \{\neg \mathit{Holds}(\mathsf{Heads}, \mathit{Do}(\mathsf{Toss}, S_0))\}$ we can draw the conclusion

$$\Sigma \cup \Sigma_O \models \neg \mathit{Holds}(\mathsf{Heads}, \mathit{Do}(\mathsf{Wait}, \mathit{Do}(\mathsf{Toss}, S_0))) \tag{6}$$

which shows that the observation "the outcome of tossing was tail" persists during Wait, that is, the fluent Heads does not change its truth value during an "irrelevant" action. Tossing the coin again (which results in situation $S_3 = \mathit{Do}(\mathsf{Toss}, \mathit{Do}(\mathsf{Wait}, \mathit{Do}(\mathsf{Toss}, S_0))))$, this time without an observation about the outcome, rule (5) can be applied and yields the default result regardless of previous observations:

$$\Sigma \cup \Sigma_O \mathrel{|\!\!\approx}^{\mathit{skept}}_{\mathcal{D}[S_3]} \mathit{Holds}(\mathsf{Heads}, S_3)$$

Hence, Algorithm 1 can also be used to complete a user-specified set of occlusions regarding potential default effects of actions. When trying to achieve the above behavior without specifying the occlusions manually, that is, using a procedure in the spirit of Algorithm 1 that takes as input only definite effects and default rules, one is unlikely to succeed: automatically creating occlusions for all prerequisite-free defaults will cause all these defaults to apply after every action. In the example above, the coin would then magically flip its side (into the default state Heads) after Wait in $\mathit{Do}(\mathsf{Toss}, S_0)$ and we could not infer (6), which contradicts our intuition that Wait has no effects.

5 Conclusions and Future Work

The paper presented a generalization of a recently proposed mechanism for default reasoning in theories of actions and change. Unlike the approach from [1], our work used a logic that allows to express dynamic defaults in addition to static ones. We observed undesired inferences that arose from the interplay of defaults and the solution of the frame problem, and presented an automatic method of adjusting the action effect axioms to preclude the unintended conclusions. Unfortunately, there seems to be a price to pay for being able to express dynamic defaults. The main result of [1] stated the sufficiency of default instantiation in the least time point when restricted to atomic supernormal defaults. This does not apply to our generalization: occlusions may make a previously inapplicable default rule applicable after action execution, therefore defaults need to be locally instantiated to yield a complete picture of the current state of the world.

It is somewhat clear that the syntax-based approach of Algorithm 1, when generalized to formulas rather than single literals, is prone to occlude both too many fluents (for example if the prerequisite is tautological but not $\top$) and too few fluents (for example if the prerequisite is not fulfilled by an action effect alone, but requires some additional state property). In the future, we will therefore be concerned with suitably generalizing the approach for a more expressive class of defaults. The second direction of generalization will be in terms of considered actions: up to now, we allowed only deterministic actions with unconditional effects. Further research will be undertaken to incorporate nondeterminism and conditional effects.

References

1. Strass, H., Thielscher, M.: On Defaults in Action Theories. In Mertsching, B., Hund, M., Aziz, Z., eds.: Proceedings of the 32nd German Annual Conference on Artificial Intelligence (KI'09), Paderborn, Germany, Springer-Verlag Berlin Heidelberg (2009) 298–305
2. Reiter, R.: A Logic for Default Reasoning. Artificial Intelligence **13** (1980) 81–132
3. Reiter, R.: The Frame Problem in the Situation Calculus: A Simple Solution (Sometimes) and a Completeness Result for Goal Regression. In: Artificial Intelligence and Mathematical Theory of Computation – Papers in Honor of John McCarthy, Academic Press (1991) 359–380
4. Thielscher, M.: From Situation Calculus to Fluent Calculus: State Update Axioms as a Solution to the Inferential Frame Problem. Artificial Intelligence **111**(1–2) (1999) 277–299
5. Thielscher, M.: A Unifying Action Calculus. Artificial Intelligence (2009, to appear)
6. McCarthy, J.: Situations and Actions and Causal Laws (1963) Stanford Artificial Intelligence Project: Memo 2.
7. Kowalski, R.A., Sergot, M.J.: A Logic-based Calculus of Events. New Generation Computing **4**(1) (1986) 67–95
8. Pirri, F., Reiter, R.: Some Contributions to the Metatheory of the Situation Calculus. Journal of the ACM **46**(3) (1999) 325–361
9. Sandewall, E.: Features and Fluents: The Representation of Knowledge about Dynamical Systems. Oxford University Press, Oxford (1994)