Default Reasoning about Conditional, Non-Local and Disjunctive Effect Actions

Hannes Strass
Institute of Computer Science
University of Leipzig
strass@informatik.uni-leipzig.de

Abstract

Recently, Baumann et al. [2010] provided a comprehensive framework for default reasoning about actions. Alas, the approach was only defined for a very basic class of domains where all actions have mere unconditional, local effects. In this paper, we show that the framework can be substantially extended to domains with action effects that are conditional (i.e. are context-sensitive to the state in which they are applied), non-local (i.e. the range of effects is not pre-determined by the action arguments) and even disjunctive (thus non-deterministic). Notably, these features can be carefully added without sacrificing important nice properties of the basic framework, such as modularity of domain specifications or existence of extensions.

1 Introduction

Reasoning about actions and non-monotonic reasoning are two important fields of logic-based knowledge representation and reasoning. While reasoning about actions deals with dynamic domains and their evolution over time, default reasoning is usually concerned with closing gaps in incomplete static knowledge bases. Both areas have received considerable attention and have reached remarkable maturity by now. However, a unifying approach that combines the full expressiveness of both fields was still lacking, until a recent paper [Baumann *et al.*, 2010] took an important first step into the direction of uniting these two lines of research. There, a logical framework was proposed that lifted default reasoning about a domain to a temporal setting where defaults, action effects and the frame assumption interact in a well-defined way.

In this paper, we develop a substantial extension of their work: we significantly generalise the theoretical framework to be able to deal with a broad class of action domains where effects may be conditional, non-local and non-deterministic. As we will show in the paper, extending the approach to conditional effects is straightforward. However, retaining their construction of defaults leads to counterintuitive conclusions. Roughly, this is due to eager default application in the presence of incomplete knowledge about action effects. As an example, consider the classical drop action that breaks fragile objects. In the presence of a (simple) state default expressing that objects are to be considered not broken unless there is information to the contrary, this could lead to the following reasoning: After dropping an object x of which nothing further is known, we can apply the default and infer it is not broken. But this means it cannot have been fragile before (since otherwise it *would* be broken). This line of reasoning violates the principle of causality: while a fragile object will be broken after dropping it, this does not mean that objects should be assumed not fragile *before* dropping them. We will formally define when such undesired inferences arise and devise a modification to the basic framework that provably disables them. Interestingly, the counterintuitive consequences occur already with conditional, local-effect actions; our modification however prevents them also for actions with non-deterministic, non-local effects. Since the introduction of effect preconditions represents our most significant change, we will prove that it is a proper generalisation of the original framework: for all action default theories with only unconditional, local effect actions, the "old" and "new" approach yield the same results. For the subsequent extensions it will be straightforward to see that they are proper generalisations.

The paper proceeds as follows. In the next section, we provide the necessary background. The sections thereafter extend the basic approach introduced in [Baumann *et al.*, 2010] by conditional effects (Section 3), non-local effects (Section 4) and disjunctive effects (Section 5). In the penultimate section, we prove several desirable properties of the extended framework; Section 7 discusses related work and concludes.

2 Background

2.1 Unifying Action Calculus

The unifying action calculus (UAC) was proposed in [Thielscher, 2011] to allow for a treatment of problems in reasoning about actions that is independent of a particular calculus. It is based on a finite, sorted logic language with equality which includes the sorts FLUENT, ACTION and TIME along with the predicates $<$: TIME $\times$ TIME, that denotes a (possibly partial) ordering on time points; *Holds* : FLUENT $\times$ TIME, that is used to state that a fluent is true at a given time point; and *Poss* : ACTION $\times$ TIME $\times$ TIME, expressing that an action is possible for given starting and ending time points.

As a most fundamental notion in the UAC, a *state formula*

$\Phi[\vec{s}]$ *in* $\vec{s}$ is a first-order formula with free TIME variables $\vec{s}$ where (1) for each occurrence of $\mathit{Holds}(f, s)$ in $\Phi[\vec{s}]$ we have $s \in \vec{s}$ and (2) predicate *Poss* does not occur. State formulas allow to express properties of action domains at given time points. Although this definition is quite general in that it allows an arbitrary finite sequence of time points, for our purposes two time points will suffice. For a function A into sort ACTION, a *precondition axiom for* $A(\vec{x})$ is of the form

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

where $\pi_A[s]$ is a state formula in s with free variables among $s, t, \vec{x}$. The formula $\pi_A[s]$ thus defines the necessary and sufficient conditions for the action A to be applicable for the arguments $\vec{x}$ at time point s, resulting in t. The UAC also provides a general form for effect axioms; we however omit this definition because we only use a special form of effect axioms here. The last notion we import formalises how action domains are axiomatised in the unifying action calculus.

Definition 1. A *(UAC) domain axiomatisation* consists of a finite set of foundational axioms Ω defining a time structure, a set Π of precondition axioms (1) and a set Υ of effect axioms; the latter two for all functions into sort ACTION; lastly, it contains uniqueness-of-names axioms for all finitely many function symbols into sorts FLUENT and ACTION.

The foundational axioms Ω serve to instantiate the UAC by a concrete time structure, for example the branching situations with their usual ordering from the situation calculus. We restrict our attention to domains that make intuitive sense; one of the basic things we require is that actions actually consume time: A domain axiomatisation is *progressing*, if $\Omega \models (\exists s : \text{TIME})(\forall t : \text{TIME}) s \leq t$ and $\Omega \cup \Pi \models \mathit{Poss}(a, s, t) \supset s < t$. Here, we are only concerned with progressing domain axiomatisations; we use the macro $\mathit{Init}(t) \stackrel{\text{def}}{=} \neg(\exists s) s < t$ to refer to the unique initial time point.

For presentation purposes, we will make use of the concept of *fluent formulas*, where terms of sort FLUENT play the role of atomic formulas, and complex formulas can be built using the usual first-order constructors. For a fluent formula Φ, we will denote by $\Phi[s]$ the state formula that is obtained by replacing all fluent literals $[\neg]f$ in Φ by $[\neg]\mathit{Holds}(f, s)$. The operator $|\cdot|$ will be used to extract the affirmative component of a fluent literal, that is, $|\neg f| = |f| = f$; the polarity of a fluent literal is given by $\mathit{sign}(\neg f) = -$ and $\mathit{sign}(f) = +$.

2.2 Default Logic

Default logic as introduced by [Reiter, 1980] uses *defaults* to extend incomplete world knowledge. They are of the form[1]

$$\frac{\alpha : \beta}{\gamma} \qquad \text{(shorthand: } \alpha : \beta/\gamma\text{)}$$

Here, α, the *prerequisite*, the β, the *justification*, and γ, the *consequent*, are first-order formulas. These expressions are to be read as "whenever we know α and nothing contradicts β, we can safely conclude γ". A default is *normal* if $\beta = \gamma$, that is, justification and consequent coincide. A default is *closed* if its prerequisite, justification and consequent are sentences, that is, have no free variables; otherwise, it is *open*.

The semantics of defaults is defined via the notion of extensions for default theories. A *default theory* is a pair (W, D), where W is a set of sentences in first-order logic and D is a set of defaults. A default theory is *closed* if all its defaults are closed; otherwise, it is *open*. For a set T of formulas, we say that a default $\alpha : \beta/\gamma$ is *applicable to* T iff $\alpha \in T$ and $\neg\beta \notin T$; we say that the default has been *applied to* T if it is applicable and additionally $\gamma \in T$. Extensions for a default theory (W, D) are deductively closed sets of formulas which contain all elements of W, are closed under application of defaults from D and which are grounded in the sense that each formula in them has a non-cyclic derivation. For closed default theories this is captured by the following definition.

Definition 2 (Theorem 2.1, [Reiter, 1980])**.** Let (W, D) be a closed default theory and E be a set of closed formulas. Define $E_0 \stackrel{\text{def}}{=} W$ and $E_{i+1} \stackrel{\text{def}}{=} \mathit{Th}(E_i) \cup D_i$ for $i \geq 0$, where

$$D_i \stackrel{\text{def}}{=} \left\{ \gamma \;\middle|\; \frac{\alpha : \beta}{\gamma} \in D, \alpha \in E_i, \neg\beta \notin E \right\}$$

Then E is an *extension for* (W, D) iff $E = \bigcup_{i=0}^{\infty} E_i$.

We will interpret open defaults as schemata representing all of their ground instances. Therefore, open default theories can be viewed as shorthand notation for their closed counterparts.[2] When we use an extension E or set of defaults D with an integer subscript, we refer to the E_i and D_i from above. We write $(W, D) \mathrel{|\!\approx} \Psi$ to express that the formula Ψ is contained in each extension of the default theory (W, D).

2.3 Default Reasoning in Action Domains with Unconditional, Local Effect Actions

The approach of [Baumann *et al.*, 2010] combines default logic with the unifying action calculus: domain axiomatisations are viewed as incomplete knowledge bases that are completed by defaults. It takes as input a description of a particular action domain with normality statements. This description comprises the following: (1) a domain signature, that defines the vocabulary of the domain; (2) a description of the direct effects of actions; (3) a set of *state defaults* $\Phi \rightsquigarrow \psi$, constructs that specify conditions Φ under which a fluent literal ψ normally holds in the domain.[3]

The state defaults from the domain description are translated into Reiter defaults, where the special predicates $\mathit{DefT}(f, s, t)$ and $\mathit{DefF}(f, s, t)$ are used to express that a fluent f becomes normally true (false) from s to t.[4] For each state default δ, two Reiter defaults are created: δ_{Init}, that is used for default conclusions about the initial time point; and δ_{Reach}, that is used for default conclusions about time points that can be reached via action application.

[1] Reiter [1980] introduces a more general version of defaults with an arbitrary number of justifications, which we do not need here.

[2] Free variables of formulas not in a default will however be implicitly universally quantified from the outside.

[3] Here, Φ, the *prerequisite*, is a fluent formula; ψ, the *consequent*, being a fluent *literal* also allows to express that a fluent normally does *not* hold in the domain.

[4] It should be noted that $\mathit{DefF}(f, s, t)$ is not the same as $\neg\mathit{DefT}(f, s, t)$ – the latter only means that f becomes not normally true from s to t.

Definition 3. Let $\delta = \Phi \rightsquigarrow \psi$ be a state default.

$$\delta_{Init} \stackrel{\text{def}}{=} \frac{Init(t) \wedge \Phi[t] : \psi[t]}{\psi[t]} \quad (2)$$

$$\delta_{Reach} \stackrel{\text{def}}{=} \frac{Pre_\delta(s,t) : Def(\psi,s,t)}{Def(\psi,s,t)} \quad (3)$$

$$Pre_\delta(s,t) \stackrel{\text{def}}{=} \Phi[t] \wedge \neg(\Phi[s] \wedge \neg\psi[s])$$

$$Def(\psi,s,t) \stackrel{\text{def}}{=} \begin{cases} DefT(\psi,s,t) & \text{if } \psi = |\psi| \\ DefF(|\psi|,s,t) & \text{otherwise} \end{cases}$$

For a set Δ of state defaults, the corresponding defaults are

$$\Delta_{Init} \stackrel{\text{def}}{=} \{\delta_{Init} \mid \delta \in \Delta\} \text{ and } \Delta_{Reach} \stackrel{\text{def}}{=} \{\delta_{Reach} \mid \delta \in \Delta\}.$$

For the *Reach* defaults concerning two time points s,t connected via action application, we ensure that the state default δ was not violated at the starting time point s by requiring $\neg(\Phi[s] \wedge \neg\psi[s])$ in the prerequisite.[5] The consequent is then inferred unless there is information to the contrary.

Being true (or false) by default is then built into the effect axiom by accepting it as a possible "cause" to determine a fluent's truth value. The other causes are the ones already known from monotonic formalisms for reasoning about actions: direct action effects, and a notion of persistence that provides a solution to the frame problem [McCarthy and Hayes, 1969].

Definition 4. Let f : FLUENT and s,t : TIME be variables. The following macros express that f persists from s to t:

$$FrameT(f,s,t) \stackrel{\text{def}}{=} Holds(f,s) \wedge Holds(f,t) \quad (4)$$

$$FrameF(f,s,t) \stackrel{\text{def}}{=} \neg Holds(f,s) \wedge \neg Holds(f,t) \quad (5)$$

Let A be a function into sort ACTION and Γ_A be a set of fluent literals with free variables in $\vec{x}$ that denote the positive and negative direct effects of $A(\vec{x})$, respectively. The following pair of macros expresses that f is a direct effect of $A(\vec{x})$:

$$DirectT(f,A(\vec{x}),s,t) \stackrel{\text{def}}{=} \bigvee_{F(\vec{x}') \in \Gamma_A,\, \vec{x}' \subseteq \vec{x}} f = F(\vec{x}') \quad (6)$$

$$DirectF(f,A(\vec{x}),s,t) \stackrel{\text{def}}{=} \bigvee_{\neg F(\vec{x}') \in \Gamma_A,\, \vec{x}' \subseteq \vec{x}} f = F(\vec{x}') \quad (7)$$

An *effect axiom with unconditional effects, the frame assumption and normal state defaults* is of the form

$$\begin{aligned} & Poss(A(\vec{x}),s,t) \supset \\ & \quad (\forall f)(Holds(f,t) \equiv CausedT(f,A(\vec{x}),s,t)) \wedge \\ & \quad (\forall f)(\neg Holds(f,t) \equiv CausedF(f,A(\vec{x}),s,t)) \end{aligned} \quad (8)$$

where

$$\begin{aligned} CausedT(f,A(\vec{x}),s,t) \stackrel{\text{def}}{=} & DirectT(f,A(\vec{x}),s,t) \vee \\ & FrameT(f,s,t) \vee DefT(f,s,t) \end{aligned} \quad (9)$$

$$\begin{aligned} CausedF(f,A(\vec{x}),s,t) \stackrel{\text{def}}{=} & DirectF(f,A(\vec{x}),s,t) \vee \\ & FrameF(f,s,t) \vee DefF(f,s,t) \end{aligned} \quad (10)$$

Note that a default conclusion of a state property in a non-initial state crucially depends on an action execution leading to that state. Hence, whenever it is definitely known that $Holds(f,t)$ after $Poss(a,s,t)$, it follows from the effect axiom that $\neg DefF(f,s,t)$; a symmetrical argument applies if $\neg Holds(f,t)$. This means that definite knowledge about a fluent inhibits the opposite default conclusion. But observe that the addition of *DefT* and *DefF* as "causes" to the effect axiom weakened the solution to the frame problem established earlier. The following definition ensures that the persistence assumption is restored in its full generality.

Definition 5. Let Δ be a set of state defaults, ψ be a fluent literal and s,t be variables of sort TIME. The *default closure axiom for ψ with respect to Δ* is

$$\left(\bigwedge_{\Phi \rightsquigarrow \psi \in \Delta} \neg Pre_{\Phi \rightsquigarrow \psi}(s,t) \right) \supset \neg Def(\psi,s,t) \quad (11)$$

For a fluent literal ψ not mentioned as a consequent in Δ the default closure axiom is just $\top \supset \neg Def(\psi,s,t)$. Given a domain axiomatisation Σ and a set Δ of state defaults, we denote by Σ_Δ the default closure axioms with respect to Δ and the fluent signature of Σ.

The fundamental notion of the solution to the state default problem by [Baumann *et al.*, 2010] is now a default theory where the incompletely specified world consists of a UAC domain axiomatisation augmented by suitable default closure axioms. The default rules are the automatic translations of user-specified, domain-dependent state defaults. For a domain axiomatisation Σ and a set Δ of state defaults, the corresponding *domain axiomatisation with state defaults* is the pair $(\Sigma \cup \Sigma_\Delta, \Delta_{Init} \cup \Delta_{Reach})$. We use a well-known example domain [Reiter, 1991] to illustrate the preceding definitions. To ease the presentation, in this example we instantiate the UAC to the branching time structure of situations.

Example 1 (Breaking Objects)**.** Imagine a robot that can move around and carry objects, among them a vase. When the robot drops an object x, it does not carry x any more and additionally x is broken. Usually, however, objects are not broken unless there is information to the contrary.

The fluents that we use to describe this domain are $\mathsf{Carries}(x)$ (the robot carries x) and $\mathsf{Broken}(x)$ (x is broken); the only function of sort ACTION is $\mathsf{Drop}(x)$. Dropping an object is possible if and only if the robot carries the object:

$$\begin{aligned} & Poss(\mathsf{Drop}(x),s,t) \equiv \\ & \qquad Holds(\mathsf{Carries}(x),s) \wedge t = Do(\mathsf{Drop}(x),s) \end{aligned}$$

The effects of dropping an object x are given by the set

$$\Gamma_{\mathsf{Drop}(x)} = \{\neg\mathsf{Carries}(x), \mathsf{Broken}(x)\}$$

The set of state defaults $\Delta^{break} = \{\top \rightsquigarrow \neg\mathsf{Broken}(x)\}$ says that objects are normally not broken. Applying the definitions from above to this specification results in the domain axiomatisation with defaults $(\Sigma^{break} \cup \Sigma^{break}_\Delta, \Delta^{break}_{Init} \cup \Delta^{break}_{Reach})$, where Σ^{break} contains effect axiom (8) and the above precondition axiom for Drop, the set Δ^{break}_{Init} contains only

$$\frac{Init(t) : \neg Holds(\mathsf{Broken}(x),t)}{\neg Holds(\mathsf{Broken}(x),t)}$$

[5] The reason for this is to prevent application of initially definitely violated state defaults through irrelevant actions. A default violation occurs when the prerequisite $\Phi[s]$ of a state default δ is known to be met, yet the negation of the consequent prevails, $\neg\psi[s]$.

and the defaults Δ^{break}_{Reach} for action application consist of

$$\frac{\neg Holds(\mathsf{Broken}(x), s) : DefF(\mathsf{Broken}(x), s, t)}{DefF(\mathsf{Broken}(x), s, t)}$$

Finally, the default closure axioms for the fluent Broken are $Holds(\mathsf{Broken}(x), s) \supset \neg DefF(\mathsf{Broken}(x), s, t)$ and $\neg DefT(\mathsf{Broken}(x), s, t)$, and $\neg Def(\psi, s, t)$ for all other fluent literals ψ. With $S_1 \stackrel{\text{def}}{=} Do(\mathsf{Drop}(\mathsf{Vase}), S_0)$, the default theory sanctions the sceptical conclusions that the vase is initially not broken, but is so after dropping it:

$$(\Sigma^{break} \cup \Sigma^{break}_{\Delta}, \Delta^{break}_{Init} \cup \Delta^{break}_{Reach}) \approx$$
$$\neg Holds(\mathsf{Broken}(\mathsf{Vase}), S_0) \wedge Holds(\mathsf{Broken}(\mathsf{Vase}), S_1)$$

One of the main theoretical results of [Baumann *et al.*, 2010] was the guaranteed existence of extensions for the class of domain axiomatisations with defaults considered there. As we will see later on, a similar result holds for our generalisation of the theory.

Proposition 1 (Theorem 4, [Baumann *et al.*, 2010])**.** *Let Σ be a domain axiomatisation and Δ be a set of state defaults. Then the corresponding domain axiomatisation with state defaults $(\Sigma \cup \Sigma_{\Delta}, \Delta_{Init} \cup \Delta_{Reach})$ has an extension. If furthermore Σ is consistent, then so are all extensions for $(\Sigma \cup \Sigma_{\Delta}, \Delta_{Init} \cup \Delta_{Reach})$.*

3 Conditional Effects

We first investigate how the default reasoning framework of [Baumann *et al.*, 2010] can be extended to conditional effect actions. As we will show, there is subtle interdependence between conditional effects and default conclusions, which requires a revision of the defaults constructed in Definition 3. We begin by formalising how to represent conditional effects in the domain specification language. Recall that in the unconditional case, action effects were just literals denoting the positive and negative effects. In the case of conditional effects, theses literals are augmented with a fluent formula that specifies the conditions under which the effect materialises.

Definition 6. A *conditional effect expression* is of the form Φ/ψ, where Φ is a fluent formula and ψ a fluent literal. Φ/ψ is called *positive* if $sign(\psi) = +$ and *negative* if $sign(\psi) = -$. For an action A and sequence of variables $\vec{x}$ matching A's arity, a conditional effect expression ε is called *local for* $A(\vec{x})$ iff all free variables in ε are among $\vec{x}$.

Throughout the paper, we will assume given a set $\Gamma_{A(\vec{x})}$ of conditional effect expressions for each function A into sort ACTION with matching sequence of variables $\vec{x}$. Such a set $\Gamma_{A(\vec{x})}$ is called *local-effect* if all $\varepsilon \in \Gamma_{A(\vec{x})}$ are local for $A(\vec{x})$. By $\Gamma^{+}_{A(\vec{x})}$ we refer to the positive, by $\Gamma^{-}_{A(\vec{x})}$ to the negative elements of $\Gamma_{A(\vec{x})}$.

With this specification of action effects, it is easy to express the implication "effect precondition implies effect" via suitable formulas. For this purpose, we introduce the new predicates *DirT* and *DirF*. Intuitively, $DirT(f, a, s, t)$ says that f is a direct positive effect of action a from s to t; symmetrically, $DirF(f, a, s, t)$ says that f is a direct *negative* effect.[6]

[6]Notice that these new predicates are in contrast to Definition 4, where *DirectT* and *DirectF* are merely syntactic sugar.

Definition 7. Let $\varepsilon = \Phi/\psi$ be a conditional effect expression and f : FLUENT and s, t : TIME be variables. The following macro expresses that ε *has been activated for f from s to t*:[7]

$$Activated_{\varepsilon}(f, s, t) \stackrel{\text{def}}{=} (f = |\psi| \wedge \Phi[s])$$

Let A be a function into sort ACTION with a set of conditional effect expressions $\Gamma_{A(\vec{x})}$ that is local-effect. The *direct positive and negative effect formulas for* $A(\vec{x})$ are

$$DirT(f, A(\vec{x}), s, t) \equiv \bigvee_{\varepsilon \in \Gamma^{+}_{A(\vec{x})}} Activated_{\varepsilon}(f, s, t) \quad (12)$$

$$DirF(f, A(\vec{x}), s, t) \equiv \bigvee_{\varepsilon \in \Gamma^{-}_{A(\vec{x})}} Activated_{\varepsilon}(f, s, t) \quad (13)$$

An *effect axiom with conditional effects, the frame assumption and normal state defaults* is of the form (8), where

$$CausedT(f, A(\vec{x}), s, t) \stackrel{\text{def}}{=} DirT(f, A(\vec{x}), s, t) \vee FrameT(f, s, t) \vee DefT(f, s, t) \quad (14)$$

$$CausedF(f, A(\vec{x}), s, t) \stackrel{\text{def}}{=} DirF(f, A(\vec{x}), s, t) \vee FrameF(f, s, t) \vee DefF(f, s, t) \quad (15)$$

The only difference between the effect axioms of [Baumann *et al.*, 2010] and the effect axioms defined here is the replacement of their macros *DirectT*, *DirectF* for unconditional direct effects with the predicates *DirT*, *DirF* for conditional effects. In the following, we will understand domain axiomatisations to contain – for each action – effect axioms of the form (8) along with the respective direct positive and negative effect formulas. To ease notation, for predicates with an obvious polarity (like *DirT*, *DirF*), we use a neutral version (like *Dir*) with fluent literals L, where $Dir(L, a, s, t)$ denotes $DirF(F, a, s, t)$ if $L = \neg F$ for some fluent F and $DirT(L, a, s, t)$ otherwise.

While this extended definition of action effects is straightforward, it severely affects the correctness of default reasoning in the action theory: as the following example shows, one cannot naïvely take this updated version of the effect axioms and use the Reiter defaults as before.

Example 1 (Continued)**.** We add a unary fluent Fragile with the obvious meaning and modify the Drop action such that dropping only breaks objects that are fragile: $\Gamma_{\mathsf{Drop}(x)} = \{\top/\neg\mathsf{Carries}(x), \mathsf{Fragile}(x)/\mathsf{Broken}(x)\}$. Assume that all we know is that the robot initially carries the vase, $Holds(\mathsf{Carries}(\mathsf{Vase}), S_0)$. As before, the effect axiom tells us that the robot does not carry the vase any more at S_1. Additionally, since we do not know whether the vase was fragile at S_0, there is no reason to believe that it is broken after dropping it, hence $\neg\mathsf{Broken}(\mathsf{Vase})$ still holds by default at S_1. But now, due to the presence of conditional effects, the effect axiom for $\mathsf{Drop}(\mathsf{Vase})$ clearly entails $\neg Holds(\mathsf{Broken}(\mathsf{Vase}), S_1) \supset \neg Holds(\mathsf{Fragile}(\mathsf{Vase}), S_0)$,[8]

[7]The second time argument t of macro $Activated_{\varepsilon}(f, s, t)$ will only be needed later when we introduce non-deterministic effects.

[8]This is just the contrapositive of the implication expressed by the effect axiom.

and thus we can draw the conclusion

$$(\Sigma^{break} \cup \Sigma_{\Delta}^{break}, \Delta_{Init}^{break} \cup \Delta_{Reach}^{break}) \;\approx\; \neg Holds(\mathsf{Fragile}(\mathsf{Vase}), S_0)$$

This is undesired as it lets us conclude something about the present (S_0) using knowledge about the future (S_1) which we could not conclude using only knowledge and default knowledge about the present (there is no default that could conclude $\neg\mathsf{Fragile}(\mathsf{Vase})$).

The flaw with this inference is that it makes default conclusions about a fluent whose truth value is affected by an action at the same time. This somewhat contradicts our intended usage of defaults about states: we originally wanted to express reasonable assumptions about fluents whose values are unknown.

Generalising the example, the undesired behaviour occurs whenever there exists a default $\Phi_D \rightsquigarrow \psi$ with conclusion ψ whose negation $\neg\psi$ might be brought about by a conditional effect $\Phi_C/\neg\psi$. The faulty inference then goes like this:

$$\Phi_D[t] \supset Def(\psi, s, t) \supset \psi[t] \supset \neg Dir(\neg\psi, s, t) \supset \neg\Phi_C[s]$$

Obviously, this inference is only undesired if there is no information about the effect's precondition at the starting time point of the action. This motivates our formal definition of the conditions under which a so-called *conflict* between an action effect and a default conclusion arises.

Definition 8. Let (Σ, Δ) be a domain axiomatisation with defaults, E be an extension for (Σ, Δ), α be a ground action and $\delta = \Phi \rightsquigarrow \psi$ be a ground state default. We say that there is a *conflict between α and δ in E* iff there exist ground time points σ and τ such that for some $i \geq 0$ we have

1. (a) $E_i \not\models Poss(\alpha, \sigma, \tau) \supset \neg Dir(\neg\psi, \alpha, \sigma, \tau)$
   (b) $E_i \not\models Def(\psi, \alpha, \sigma, \tau)$
2. (a) $E_{i+1} \models Poss(\alpha, \sigma, \tau) \supset \neg Dir(\neg\psi, \alpha, \sigma, \tau)$
   (b) $E_{i+1} \models Def(\psi, \sigma, \tau)$

In words, a conflict arises in an extension if up to some stage i, before we make the default conclusion ψ, we cannot conclude the effect $\neg\psi$ will not occur (1); after concluding ψ by default, we infer that $\neg\psi$ cannot occur as direct effect (2). We can now go back to the example seen earlier and verify that the counter-intuitive conclusion drawn there was indeed due to a conflict in the sense of the above definition.

Example 1 (Continued)**.** Consider the only extension E^{break} for $(\Sigma^{break} \cup \Sigma_{\Delta}^{break}, \Delta_{Init}^{break} \cup \Delta_{Reach}^{break})$. Before applying any defaults whatsoever, we know that dropping the vase is possible: $E_0^{break} \models Poss(\mathsf{Drop}(\mathsf{Vase}), S_0, S_1)$; but we do not know if the vase is fragile and hence $E_0^{break} \not\models \neg DirT(\mathsf{Broken}(\mathsf{Vase}), \mathsf{Drop}(\mathsf{Vase}), S_0, S_1)$ (item 1). After applying all the defaults, we know that the vase is not broken at S_1: $E_1^{break} \models DefF(\mathsf{Broken}(\mathsf{Vase}), S_0, S_1)$. Hence, it cannot have been broken by dropping it in S_0, that is, $E_1^{break} \models \neg DirT(\mathsf{Broken}(\mathsf{Vase}), \mathsf{Drop}(\mathsf{Vase}), S_0, S_1)$ (item 2), thus cannot have been fragile in the initial situation.

In the following, we will modify the definition of Reiter defaults from [Baumann *et al.*, 2010] to eliminate the possibility of such conflicts. The underlying idea is to apply a default only if it is known that a conflict cannot arise, that is, if it is known that the contradictory direct effect cannot materialise. To this end, we extend the original default prerequisite $Pre_\delta(s,t) = \Phi[t] \wedge \neg(\Phi[s] \wedge \neg\psi[s])$ that only requires the precondition to hold and the default not to be violated previously: we will additionally stipulate that any action a happening at the same time cannot create a conflict.

Definition 9. Let $\delta = \Phi \rightsquigarrow \psi$ be a state default and s, t : TIME be variables.

$$Safe_\delta(s,t) \stackrel{\text{def}}{=} (\forall a)(Poss(a,s,t) \supset \neg Dir(\neg\psi, a, s, t))$$
$$\delta_{Poss} \stackrel{\text{def}}{=} \frac{Pre_\delta(s,t) \wedge Safe_\delta(s,t) : Def(\psi, s, t)}{Def(\psi, s, t)} \quad (16)$$

For a set Δ of state defaults, $\Delta_{Poss} \stackrel{\text{def}}{=} \{\delta_{Poss} \mid \delta \in \Delta\}$.

In the example domain, applying the above definition yields the following.

Example 1 (Continued)**.** For the state default δ^{break} saying that objects are usually not broken, we have $Safe_{\delta^{break}}(s,t) = (\forall a)(Poss(a,s,t) \supset \neg DirT(\mathsf{Broken}(x), a, s, t))$. This expresses that the state default can be safely applied from s to t whenever for any action a happening at the same time, it is known that a does not cause a violation of this default at the ending time point t. The resulting default δ_{Poss}^{break} is

$$\frac{\neg Holds(\mathsf{Broken}(x), s) \wedge Safe_{\delta^{break}}(s,t) : DefF(\mathsf{Broken}(x), s, t)}{DefF(\mathsf{Broken}(x), s, t)}$$

As we will see later (Theorem 3), the default closure axioms $\neg Pre_{\Phi \rightsquigarrow \psi}(s,t) \supset \neg Def(\psi, s, t)$ for preserving the commonsense principle of inertia in the presence of inapplicable defaults need not be modified. With our new defaults, we can now redefine the concept of a domain axiomatisation with defaults for conditional effect actions.

Definition 10. Let Σ be a domain axiomatisation where the effect axioms are given by Definition 7 and let Δ be a set of state defaults. The corresponding *domain axiomatisation with defaults* is the pair $(\Sigma \cup \Sigma_\Delta, \Delta_{Init} \cup \Delta_{Poss})$.

The direct effect formulas that determine *DirT* and *DirF* will be redefined twice in this paper. We will understand the above definition to be retrofitted with their latest version. The extension to conditional effects is a proper generalisation of the original approach of Section 2.3 for the special case of unconditional effect actions, as is shown below.

Theorem 2. *Consider a domain axiomatisation with only unconditional action effects and a set Δ of state defaults. Let $\Xi_1 = (\Sigma \cup \Sigma_\Delta, \Delta_{Init} \cup \Delta_{Reach})$ be the corresponding domain axiomatisation with defaults of [Baumann* et al.*, 2010], and let $\Xi_2 = (\Sigma' \cup \Sigma_\Delta, \Delta_{Init} \cup \Delta_{Poss})$ be the domain axiomatisation with defaults according to Definition 10. For a state formula Ψ and time point τ, we have $\Xi_1 \approx \Psi[\tau]$ iff $\Xi_2 \approx \Psi[\tau]$.*

Proof sketch. For unconditional effects, a ground *Dir* atom is by Definition 7 equivalent to the corresponding *Direct* macro, hence the effect axioms of the two approaches are equivalent. Furthermore, the truth values of ground *DirT* and *DirF* atoms are always fixed, and consequently each Reiter default (16) defined above is applicable whenever the original *Reach* default (3) of [Baumann *et al.*, 2010] is applicable. □

4 Non-Local Effects

Up to here, conditional effect expressions for an action $A(\vec{x})$ were restricted to contain only variables among $\vec{x}$. Considering a ground instance $A(\vec{\varsigma})$ of an action, this means that the set of objects that can possibly be affected by this action is already fixed to $\vec{\varsigma}$. This is a restriction because it can make the specification of certain actions at least cumbersome or utterly impossible, for example actions that affect a vast number of (or all of the) domain elements at once.

The gain in expressiveness when allowing non-local action effects comes at a relatively low cost: it suffices to allow additional free variables $\vec{y}$ in the conditional effect expressions. They represent the objects that may be affected by the action without being among the action arguments $\vec{x}$.

Definition 11. Let A be a function into sort ACTION and $\vec{x}$ a sequence of variables matching A's arity. Let ε be a conditional effect expression of the form $\Phi/F(\vec{x}', \vec{y})$ or $\Phi/\neg F(\vec{x}', \vec{y})$ with free variables $\vec{x}', \vec{y}$, where $\vec{x}' \subseteq \vec{x}$ and $\vec{y}$ is disjoint from $\vec{x}$.

For variables f : FLUENT and s, t : TIME, the following macro expresses that ε *has been activated for* f *from* s *to* t:

$$Activated_\varepsilon(f, s, t) \stackrel{\text{def}}{=} (\exists \vec{y})(f = F(\vec{x}', \vec{y}) \wedge \Phi[s])$$

The *direct positive and negative effect formulas* are of the form (12) and (13).

Note that according to this definition, free variables $\vec{y}$ are quantified existentially when they occur in the context Φ and universally when they occur in the consequence ψ. They thus not only express non-local effects but also non-local contexts.

Example 2 (Exploding Bomb [Reiter, 1991])**.** In this domain, objects might get broken not by getting dropped, but because a bomb in their proximity explodes: $\Gamma_{\mathsf{Detonate}(b)} = \{\mathsf{Bomb}(b) \wedge \mathsf{Near}(b, x)/\mathsf{Broken}(x)\}$. Def. 11 yields the direct effect formulas $DirT(f, \mathsf{Detonate}(b), s, t) \equiv (\exists x)(f = \mathsf{Broken}(x) \wedge Holds(\mathsf{Near}(x, b), s))$ and $DirF(f, \mathsf{Detonate}(b), s, t) \equiv \bot$.

In this example, the defaults from Definition 9 also prevented conflicts possibly arising from non-local effects. We will later see that this is the case for all domains with local and non-local effect actions.

Like the original framework, our extension implements a particular preference ordering between causes that determine a fluent's truth value. This means that whenever two causes are in conflict – for example, a state default says an object is not broken, and an action effect says it *is* – the preferred cause takes precedence. The preferences are

$$\text{direct effects} < \text{default conclusions} < \text{persistence},$$

where $a < b$ means "a is preferred to b". The theorem below proves that this preference ordering is indeed established.

Theorem 3. *Let Σ be a domain axiomatisation, Δ be a set of state defaults, $\delta = \Phi \rightsquigarrow \psi \in \Delta$ be a state default, E be an extension for the domain axiomatisation with state defaults $(\Sigma \cup \Sigma_\Delta, \Delta_{Init} \cup \Delta_{Poss})$, φ be a ground fluent, and $E \models Poss(\alpha, \sigma, \tau)$ for ground action α and time points σ, τ.*

1. *Effects override everything:*

$$\Phi/(\neg)\varphi \in \Gamma_\alpha \text{ and } E \models \Phi[\sigma] \text{ imply } E \models (\neg)\varphi[\tau].$$

2. *Defaults override persistence:*
   (A) Let $\Phi''/\psi, \Phi''/\neg\psi \notin \Gamma_\alpha$ for all Φ'';
   (B) for each $\delta' = \Phi' \rightsquigarrow \neg\psi \in \Delta$, let δ' be not applicable to E; and
   (C) $E \models Pre_\delta(\sigma, \tau) \wedge Safe_\delta(\sigma, \tau)$.
   Then $E \models \psi[\tau]$.
3. *The frame assumption is correctly implemented: For all fluent formulas Φ'', let $\Phi''/\psi, \Phi''/\neg\psi \notin \Gamma_\alpha$ and for all state defaults δ' with consequent ψ or $\neg\psi$, let $E \models \neg Pre_{\delta'}(\sigma, \tau)$. Then $E \models \psi[\sigma] \equiv \psi[\tau]$.*

Proof sketch. Similar to the proof of Theorem 3 in [Baumann *et al.*, 2010], adapted to our definition of Reiter defaults. □

5 Disjunctive Effects

The next and final addition to effect axiom (8) is the step of generalising purely deterministic action effects. Disjunctive action effects have been studied in the past [Kartha, 1994; Shanahan, 1997; Giunchiglia *et al.*, 1997; Thielscher, 2000]. Our contribution in this paper is two-fold. First, we express disjunctive effects by building them into the effect axiom inspired by work on nonmonotonic causal theories [Giunchiglia *et al.*, 2004]. This works without introducing additional function symbols – called *determining fluents* [Shanahan, 1997] – for which persistence is not assumed and that are used to derive indeterminate effects via conditional effects. The second and more important contribution is the combination of non-deterministic effects with state defaults. We claim that it brings a significant representational advantage: Disjunctive effects can explicitly represent potentially different outcomes of an action of which none is necessarily predictable. At the same time, state defaults can be used to model the action effect that *normally* obtains. For example, dropping an object might not always completely break it, but most of the time only damage it. This can be modelled in our framework by specifying "broken or damaged" as disjunctive effect of the drop action, and then including the default "normally, dropped objects are damaged" to express the usual outcome.

Next, we define how disjunctive effects are declared by the user and accommodated into the theory. The basic idea is to allow disjunctions of fluent literals $\psi_1 \vee \ldots \vee \psi_n$ in the effect part of a direct effect expression. The intended meaning of these disjunctions is that after action execution, at least one of the effects ψ_i holds.

Definition 12. Let Φ be a fluent formula and $\Psi = \psi_1 \vee \ldots \vee \psi_n$ be a disjunction of fluent literals. The pair Φ/Ψ is called a *conditional disjunctive effect expression (or cdee)*.

Firstly, we want to guarantee that at least one effect out of $\psi_1 \vee \ldots \vee \psi_n$ occurs. To this end, we say for each ψ_i that non-occurrence of all the other effects ψ_j with $j \neq i$ is a sufficient cause for ψ_i to occur. We build into the effect axiom (in the same way as before) the n implications

$$\Phi[s] \wedge \neg\psi_2[t] \wedge \ldots \wedge \neg\psi_n[t] \supset Caused(\psi_1, a, s, t)$$

$$\vdots$$

$$\Phi[s] \wedge \neg\psi_1[t] \wedge \ldots \wedge \neg\psi_{n-1}[t] \supset Caused(\psi_n, a, s, t)$$

This, together with the persistence assumption, is in effect an exclusive or where only exactly one effect occurs (given that no other effects occur simultaneously). Thus we add, for each literal, its truth as sufficient cause for itself being true:

$$\Phi[s] \wedge \psi_1[t] \supset \mathit{Caused}(\psi_1, a, s, t)$$
$$\vdots$$
$$\Phi[s] \wedge \psi_n[t] \supset \mathit{Caused}(\psi_n, a, s, t)$$

This makes every interpretation where at least one of the mentioned literals became true a model of the effect axiom. For the next definition, we identify a disjunction of literals $\Psi = \psi_1 \vee \ldots \vee \psi_n$ with the set of literals $\{\psi_1, \ldots, \psi_n\}$.

Definition 13. Let $\varepsilon = \Phi/\Psi$ be a conditional disjunctive effect expression, $\psi \in \Psi$ and f : FLUENT and s, t : TIME be variables. The following macro expresses that *effect* ψ *of cdee* ε *has been activated for* f *from* s *to* t:

$$\mathit{Activated}_{\varepsilon,\psi}(f, s, t) \stackrel{\text{def}}{=}$$
$$f = |\psi| \wedge \Phi[s] \wedge \left(\left(\bigwedge_{\psi' \in \Psi \setminus \{\psi\}} \neg\psi'[t] \right) \vee \psi[t] \right)$$

Let A be a function into sort ACTION and Γ_A be a set of conditional disjunctive effect expressions with free variables in $\vec{x}$ that denote the direct conditional disjunctive effects of $A(\vec{x})$. The *direct positive and negative effect formulas* are

$$\mathit{DirT}(f, A(\vec{x}), s, t) \equiv \bigvee_{\substack{\Phi/\Psi \in \Gamma_{A(\vec{x})},\\ \psi \in \Psi,\, \mathit{sign}(\psi) = +}} \mathit{Activated}_{\varepsilon,\psi}(f, s, t) \quad (17)$$

$$\mathit{DirF}(f, A(\vec{x}), s, t) \equiv \bigvee_{\substack{\Phi/\Psi \in \Gamma_{A(\vec{x})},\\ \psi \in \Psi,\, \mathit{sign}(\psi) = -}} \mathit{Activated}_{\varepsilon,\psi}(f, s, t) \quad (18)$$

The implementation of the example sketched above illustrates the definition.

Example 1 (Continued)**.** We once again modify the action $\mathsf{Drop}(x)$. Now a fragile object that is dropped becomes not necessarily completely broken, but might only get damaged. To this end, we record in the new fluent $\mathsf{Dropped}(x)$ that the object has been dropped and write the state default $\delta = \mathsf{Dropped}(x) \rightsquigarrow \mathsf{Damaged}(x)$ saying that dropped objects are usually damaged. Together, these two express the *normal* outcome of the action drop. Formally, the action effects are $\Gamma_{\mathsf{Drop}(x)} = \{\, \top/\neg\mathsf{Carries}(x), \top/\mathsf{Dropped}(x), \mathsf{Fragile}(x)/\mathsf{Broken}(x) \vee \mathsf{Damaged}(x)\}$. Constructing the direct effect formulas as per Definition 13 yields

$$\begin{aligned} &\mathit{DirT}(f, \mathsf{Drop}(x), s, t) \equiv \\ &\quad f = \mathsf{Dropped}(x) \\ &\quad \vee (f = \mathsf{Broken}(x) \wedge \mathit{Holds}(\mathsf{Fragile}(x), s) \wedge \\ &\qquad (\neg\mathit{Holds}(\mathsf{Damaged}(x), t) \vee \mathit{Holds}(\mathsf{Broken}(x), t))) \\ &\quad \vee (f = \mathsf{Damaged}(x) \wedge \mathit{Holds}(\mathsf{Fragile}(x), s) \wedge \\ &\qquad (\neg\mathit{Holds}(\mathsf{Broken}(x), t) \vee \mathit{Holds}(\mathsf{Damaged}(x), t))) \end{aligned}$$

Since the effect axiom of $\mathsf{Drop}(x)$ is itself not determined about the status of $\mathsf{Broken}(x)$ and $\mathsf{Damaged}(x)$ (but is determined about $\mathsf{Damaged}(x)$ not being among its negative effects), the default δ_{Poss} is applicable and we conclude

$$(\Sigma^{\mathit{break}} \cup \Sigma_\Delta^{\mathit{break}}, \Delta_{\mathit{Init}}^{\mathit{break}} \cup \Delta_{\mathit{Poss}}^{\mathit{break}}) \approx$$
$$\mathit{Holds}(\mathsf{Carries}(\mathsf{Vase}), S_0) \wedge \mathit{Holds}(\mathsf{Damaged}(\mathsf{Vase}), S_1)$$

If we now observe that the vase is broken after all – $\mathit{Holds}(\mathsf{Broken}(\mathsf{Vase}), S_1)$ – and add this information to the knowledge base, we will learn that this was an action effect:

$$(\Sigma^{\mathit{break}} \cup \Sigma_\Delta^{\mathit{break}}, \Delta_{\mathit{Init}}^{\mathit{break}} \cup \Delta_{\mathit{Poss}}^{\mathit{break}}) \approx$$
$$\mathit{Holds}(\mathsf{Broken}(\mathsf{Vase}), S_1) \supset$$
$$\mathit{DirT}(\mathsf{Broken}(\mathsf{Vase}), \mathsf{Drop}(\mathsf{Vase}), S_0, S_1)$$

Furthermore, the observation allows us to rightly infer that the vase was fragile at S_0.

It is worth noting that for a cdee Φ/Ψ with deterministic effect $\Psi = \{\psi\}$, the macro $\mathit{Activated}_{\Phi/\Psi,\psi}(f, s, t)$ expressing activation of this effect is equivalent to $\mathit{Activated}_{\Phi/\psi}(f, s, t)$ from Definition 7 for activation of the conditional effect; hence the direct effect formulas (17) for disjunctive effects are a generalisation of (12), the ones for deterministic effects. We have considered here only *local* non-deterministic effects to keep the presentation simple. Of course, the notion can be extended to non-local effects without harm.

6 Properties of the Extended Framework

We have already seen in previous sections that the approach to default reasoning about actions presented here has certain nice properties: it is a generalisation of the basic approach [Baumann *et al.*, 2010] and it implements a particular preference ordering among causes. While those results were mostly straightforward adaptations, the theorem below is novel. It states that conflicts between conditional effects and default conclusions in the sense of Definition 8 cannot occur.

Theorem 4. *Let* (Σ, Δ) *be a domain axiomatisation with defaults,* E *be an extension for* (Σ, Δ) *and* $\delta = \Phi \rightsquigarrow \psi$ *be a state default. Furthermore, let* $i \geq 0$ *be such that* $\mathit{Def}(\psi, \sigma, \tau) \notin E_i$ *and* $\mathit{Def}(\psi, \sigma, \tau) \in E_{i+1}$*. Then for all ground actions* α*,* $\mathit{Poss}(\alpha, \sigma, \tau) \supset \neg\mathit{Dir}(\neg\psi, \alpha, \sigma, \tau) \in E_i$*.*

Proof. According to Def. 2, we have $E_{i+1} = \mathit{Th}(E_i) \cup \Delta_i$; hence, $\mathit{Def}(\psi, \sigma, \tau) \in E_{i+1}$ can have two possible reasons:

1. $\mathit{Def}(\psi, \sigma, \tau) \in \mathit{Th}(E_i) \setminus E_i$. By construction, this can only be due to effect axiom (8), more specifically, we have (1) $E_i \models \mathit{Caused}(\psi, \alpha, \sigma, \tau) \wedge \neg\mathit{Frame}(\psi, \sigma, \tau) \wedge \neg\mathit{Dir}(\psi, \sigma, \tau)$ and (2) $E_i \models \neg\mathit{Caused}(\neg\psi, \alpha, \sigma, \tau)$, whence $E_i \models \neg\mathit{Dir}(\neg\psi, \alpha, \sigma, \tau)$ proving the claim.
2. $\mathit{Def}(\psi, \sigma, \tau) \in \Delta_i$. By definition of δ_{Poss} in Def. 9, $\mathit{Pre}_\delta(\sigma, \tau) \wedge \mathit{Safe}_\delta(\sigma, \tau) \in E_i$, whereby we can conclude $\mathit{Poss}(\alpha, \sigma, \tau) \supset \neg\mathit{Dir}(\neg\psi, \alpha, \sigma, \tau) \in E_i$. ☐

Note that conflicts already arise with conditional, local effects; the framework however makes sure there are no conflicts even for conditional, non-local, disjunctive effects.

Finally, the existence of extensions for domain axiomatisations with state defaults can still be guaranteed for the extended framework.

Theorem 5. *Let Σ be a domain axiomatisation and Δ be a set of state defaults. Then the corresponding domain axiomatisation with defaults $(\Sigma \cup \Sigma_\Delta, \Delta_{Init} \cup \Delta_{Poss})$ has an extension. If furthermore Σ is consistent, then so are all extensions for $(\Sigma \cup \Sigma_\Delta, \Delta_{Init} \cup \Delta_{Poss})$.*

Proof. Existence of an extension is a corollary of Theorem 3.1 in [Reiter, 1980] since the defaults in $\Delta_{Init} \cup \Delta_{Poss}$ are still normal. If Σ is consistent, then so is $\Sigma \cup \Sigma_\Delta$ by the argument in the proof of Theorem 4 in [Baumann *et al.*, 2010]. Consistency of all extensions then follows from Corollary 2.2 in [Reiter, 1980]. □

Additionally, it is easy to see that the domain specifications provided by the user are still modular: different parts of the specifications, such as conditional effect expressions and state defaults, are completely independent of each other from a user's point of view. Yet, the intricate semantic interactions between them are correctly dealt with.

7 Discussion

We have presented an extension to a recently introduced framework for default reasoning in theories of actions and change. The extension increases the range of applicability of the framework while fully retaining its desirable properties: we can now express context-dependent effects of actions, actions with a potentially global effect range and indeterminate effects of actions – all the while domain descriptions have not become significantly more complex, and default extensions of the framework still provably exist.

There is not much related work concerning the kind of default reasoning about actions we consider here. [Denecker and Ternovska, 2007] enriched the situation calculus [Reiter, 2001] with inductive definitions. While they provide a nonmonotonic extension of an action calculus, the intended usage is to solve the ramification problem rather than to do the kind of defeasible reasoning we are interested in this work. [Lakemeyer and Levesque, 2009] provide a progression-based semantics for state defaults in a variant of the situation calculus, but without looking at nondeterministic actions. In an earlier paper [Strass and Thielscher, 2009], we explored default effects of nondeterministic actions, albeit in a much more restricted setting: there, actions had only unconditional effects – either deterministic or disjunctive of the form $f \vee \neg f$ –, and defaults had only atomic components, that is, they were of the form $(\neg)Holds(f, t) : (\neg)Holds(g, t)/(\neg)Holds(g, t)$. Most recently, [Michael and Kakas, 2011] gave an argumentation-based semantics for propositional action theories with state defaults. While being more flexible in terms of preferences between causes, their approach is constricted to a linear time structure built into the language and does not make a clear ontological distinction between fluents and actions.

References

[Baumann *et al.*, 2010] Ringo Baumann, Gerhard Brewka, Hannes Strass, Michael Thielscher, and Vadim Zaslawski. State Defaults and Ramifications in the Unifying Action Calculus. In *Proceedings of KR*, pages 435–444, Toronto, Canada, May 2010.

[Denecker and Ternovska, 2007] Marc Denecker and Eugenia Ternovska. Inductive Situation Calculus. *AIJ*, 171(5–6):332–360, 2007.

[Giunchiglia *et al.*, 1997] Enrico Giunchiglia, G. Neelakantan Kartha, and Vladimir Lifschitz. Representing Action: Indeterminacy and Ramifications. *AIJ*, 95(2):409–438, 1997.

[Giunchiglia *et al.*, 2004] Enrico Giunchiglia, Joohyung Lee, Vladimir Lifschitz, Norman McCain, and Hudson Turner. Nonmonotonic Causal Theories. *AIJ*, 153(1-2):49–104, 2004.

[Kartha, 1994] G. Neelakantan Kartha. Two Counterexamples Related to Baker's Approach to the Frame Problem. *AIJ*, 69(1–2):379–391, 1994.

[Lakemeyer and Levesque, 2009] Gerhard Lakemeyer and Hector Levesque. A Semantical Account of Progression in the Presence of Defaults. In *Proceedings of IJCAI*, pages 842–847, 2009.

[McCarthy and Hayes, 1969] John McCarthy and Patrick J. Hayes. Some Philosophical Problems from the Standpoint of Artificial Intelligence. In *Machine Intelligence*, pages 463–502. Edinburgh University Press, 1969.

[Michael and Kakas, 2011] Loizos Michael and Antonis Kakas. A Unified Argumentation-Based Framework for Knowledge Qualification. In E. Davis, P. Doherty, and E. Erdem, editors, *Proceedings of the Tenth International Symposium on Logical Formalizations of Commonsense Reasoning*, Stanford, CA, March 2011.

[Reiter, 1980] Raymond Reiter. A Logic for Default Reasoning. *AIJ*, 13:81–132, 1980.

[Reiter, 1991] Raymond Reiter. 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*, pages 359–380. Academic Press, 1991.

[Reiter, 2001] Raymond Reiter. *Knowledge in Action: Logical Foundations for Specifying and Implementing Dynamical Systems*. The MIT Press, September 2001.

[Shanahan, 1997] Murray Shanahan. *Solving the Frame Problem: A Mathematical Investigation of the Common Sense Law of Inertia*. The MIT Press, February 1997.

[Strass and Thielscher, 2009] Hannes Strass and Michael Thielscher. Simple Default Reasoning in Theories of Action. In *Proceedings of AI*, pages 31–40, Melbourne, Australia, December 2009. Springer-Verlag Berlin Heidelberg.

[Thielscher, 2000] Michael Thielscher. Nondeterministic Actions in the Fluent Calculus: Disjunctive State Update Axioms. In *Intellectics and Computational Logic (to Wolfgang Bibel on the occasion of his 60th birthday)*, pages 327–345, Deventer, The Netherlands, The Netherlands, 2000. Kluwer, B.V.

[Thielscher, 2011] Michael Thielscher. A Unifying Action Calculus. *AIJ*, 175(1):120–141, 2011.