On the Revision of Action Laws: An Algorithmic Approach

Ivan José Varzinczak

Meraka Institute
CSIR, Pretoria, South Africa
ivan.varzinczak@meraka.org.za

Abstract

Domain descriptions in reasoning about actions are logical theories and as such they may also evolve. Given that, knowledge engineers also need revision tools to incorporate new incoming laws about the dynamic environment. Here we fill this gap by providing an algorithmic approach for revision of action laws. We give a well defined semantics that ensures minimal change w.r.t. the original models, and show correctness of our algorithms w.r.t. the semantic constructions.

1 Introduction

Like any logical theory, action theories in reasoning about actions may evolve, and thus need revision methods to adequately accommodate new information about the behavior of actions. Recently, update and contraction-based methods for action theory change have been defined [Eiter *et al.*, 2005; Herzig *et al.*, 2006; Varzinczak, 2008]. Here we continue this important though quite new thread of investigation and develop a minimal change approach for *revising* laws of an action domain description.

The motivation is as follows. Consider an agent designed to interact with a coffee machine (Figure 1).

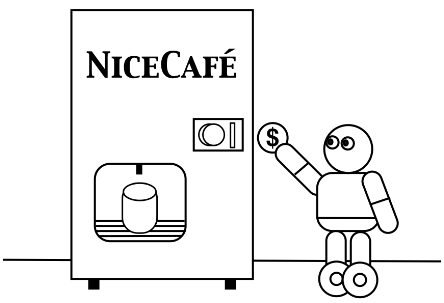

Figure 1: The coffee deliverer agent.

Among her beliefs, the agent may know that a coffee is a hot drink, that after buying she gets a coffee, and that with a token it is possible to buy. We can see the agent's beliefs about the behavior of actions in this scenario as a transition system (Figure 2).

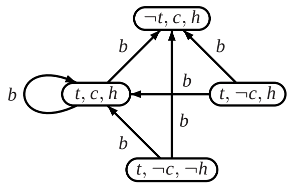

Figure 2: A transition system depicting the agent's knowledge about the dynamics of the coffee machine. b, t, c, and h stand for, respectively, *buy*, *token*, *coffee*, and *hot*.

Now, it may be the case that the agent learns that coffee is the only hot drink available at the machine, or that even without a token she can still buy, or that all possible executions of *buy* should lead to states where $\neg$*token* is the case. These are examples of *revision* with new laws about the dynamics of the environment under consideration. And here we are interested in exactly these kinds of theory modification.

The contributions of the present work are as follows:

- What is the semantics of revising an action theory by a law? How to get minimal change, i.e., how to keep as much knowledge about other laws as possible?
- How to syntactically revise an action theory so that its result corresponds to the intended semantics?

Here we answer these questions.

2 Logical Preliminaries

Our base formalism is multimodal logic K_n [Popkorn, 1994].

2.1 Action Theories in Multimodal K

Let $\mathfrak{A} = \{a_1, a_2, \ldots, a_n\}$ be the set of *atomic actions* of a domain. To each action a there is associated a modal operator $[a]$. $\mathfrak{P} = \{p_1, p_2, \ldots, p_n\}$ denotes the set of *propositions*, or *atoms*. $\mathfrak{L} = \{p, \neg p \ : \ p \in \mathfrak{P}\}$ is the set of literals. ℓ denotes a literal and $|\ell|$ the atom in ℓ.

We use $\varphi, \psi, \ldots$ to denote *Boolean formulas*. $\mathfrak{F}$ is the set of all Boolean formulas. A propositional valuation v is a *maximally consistent* set of literals. We denote by $v \Vdash \varphi$ the fact that v satisfies φ. By $val(\varphi)$ we denote the set of all valuations satisfying φ. $\models_{\mathsf{CPL}}$ is the classical consequence relation. $Cn(\varphi)$ denotes all logical consequences of φ.

With $IP(\varphi)$ we denote the set of *prime implicants* [Quine, 1952] of φ. By π we denote a prime implicant, and $atm(\pi)$ is the set of atoms occurring in π. Given ℓ and π, $\ell \in \pi$ abbreviates 'ℓ is a literal of π'. For a given set A, $\bar{A}$ denotes its complement. Hence $\overline{atm(\pi)}$ denotes $\mathfrak{P} \setminus atm(\pi)$.

We use $\Phi, \Psi, \ldots$ to denote complex formulas (possibly with modal operators). $\langle a \rangle$ is the dual operator of $[a]$ ($\langle a \rangle \Phi =_{\text{def}} \neg[a]\neg\Phi$).

A K_n*-model* is a tuple $\mathscr{M} = \langle W, R \rangle$ where W is a set of valuations, and R maps action constants a to accessibility relations $R_a \subseteq W \times W$. Given $\mathscr{M}$, $\models^{\mathscr{M}}_{w} p$ (p is true at world w of model $\mathscr{M}$) if $w \Vdash p$; $\models^{\mathscr{M}}_{w} [a]\Phi$ if $\models^{\mathscr{M}}_{w'} \Phi$ for every w' s.t. $(w, w') \in R_a$; truth conditions for the other connectives are as usual. By $\mathcal{M}$ we will denote a set of K_n-models.

$\mathscr{M}$ is a model of Φ (noted $\models^{\mathscr{M}} \Phi$) if and only if $\models^{\mathscr{M}}_{w} \Phi$ for all $w \in W$. $\mathscr{M}$ is a model of a set of formulas Σ (noted $\models^{\mathscr{M}} \Sigma$) if and only if $\models^{\mathscr{M}} \Phi$ for every $\Phi \in \Sigma$. Φ is a *consequence of the global axioms* Σ in all K_n-models (noted $\Sigma \models_{\mathsf{K}_n} \Phi$) if and only if for every $\mathscr{M}$, if $\models^{\mathscr{M}} \Sigma$, then $\models^{\mathscr{M}} \Phi$.

In K_n we can state laws describing the behavior of actions. Here we distinguish three types of them.

Static Laws A *static law* is a formula $\varphi \in \mathfrak{F}$ that characterizes the possible states of the world. An example is $coffee \rightarrow hot$: if the agent holds a coffee, then she holds a hot drink. The set of static laws of a domain is denoted by $\mathcal{S}$.

Effect Laws An *effect law for a* has the form $\varphi \rightarrow [a]\psi$, with $\varphi, \psi \in \mathfrak{F}$. Effect laws relate an action to its effects, which can be conditional. The consequent ψ is the effect that always obtains when a is executed in a state where the antecedent φ holds. An example is $token \rightarrow [buy]hot$: whenever the agent has a token, after buying, she has a hot drink. If ψ is inconsistent we have a special kind of effect law that we call an *inexecutability law*. For example, $\neg token \rightarrow [buy]\bot$ says that *buy* cannot be executed if the agent has no token. The set of effect laws of a domain is denoted by $\mathcal{E}$.

Executability Laws An *executability law for a* has the form $\varphi \rightarrow \langle a \rangle \top$, with $\varphi \in \mathfrak{F}$. It stipulates the context in which a is guaranteed to be executable. (In K_n $\langle a \rangle \top$ reads "a's execution is possible".) For instance, $token \rightarrow \langle buy \rangle \top$ says that buying can be executed whenever the agent has a token. The set of executability laws of a domain is denoted by $\mathcal{X}$.

Given a, $\mathcal{E}_a$ (resp. $\mathcal{X}_a$) will denote the set of only those effect (resp. executability) laws about a.

Action Theories $\mathcal{T} = \mathcal{S} \cup \mathcal{E} \cup \mathcal{X}$ is an *action theory*.

2.2 The Frame, Ramification and Qualification Problems

To make the presentation more clear to the reader, we here assume that the agent's theory contains all frame axioms. However, all we shall say here can be defined within a formalism with a solution to the frame and ramification problems like done by Herzig *et al.* [2006].

Given the acknowledged difficulty of the qualification problem, we do not assume here any a priori solution to it. Instead, we suppose the knowledge engineer may want to state some (not necessarily fully specified) executability laws for some actions. These may be incorrect at the starting point, but revising wrong executability laws is an approach towards its solution and one of the aims of this work. With further information the knowledge engineer will have the chance to change them so that eventually they will correspond to the intuition (cf. Section 3).

The action theory of our running example will thus be:

$$\mathcal{T} = \left\{ \begin{array}{c} coffee \rightarrow hot, token \rightarrow \langle buy \rangle \top, \\ \neg coffee \rightarrow [buy]coffee, \neg token \rightarrow [buy]\bot, \\ coffee \rightarrow [buy]coffee, hot \rightarrow [buy]hot \end{array} \right\}$$

Figure 2 above shows a K_n-model for the theory $\mathcal{T}$.

2.3 Supra-models

Sometimes it will be useful to consider models whose possible worlds are *all* the possible states allowed by $\mathcal{S}$:

Definition 1 *$\mathscr{M} = \langle W, R \rangle$ is a* big frame *of $\mathcal{T}$ if and only if:*

- *$W = val(\mathcal{S})$; and*
- *$R = \bigcup_{a \in \mathfrak{A}} R_a$, where*

$$R_a = \{(w, w') : \forall.\varphi \rightarrow [a]\psi \in \mathcal{E}_a,\ if\ \models^{\mathscr{M}}_{w} \varphi\ then\ \models^{\mathscr{M}}_{w'} \psi\}$$

Big frames of $\mathcal{T}$ are not unique and not always models of $\mathcal{T}$.

Definition 2 *$\mathscr{M}$ is a* supra-model *of $\mathcal{T}$ iff $\models^{\mathscr{M}} \mathcal{T}$ and $\mathscr{M}$ is a big frame of $\mathcal{T}$.*

Figure 3 depicts a supra-model of our example $\mathcal{T}$.

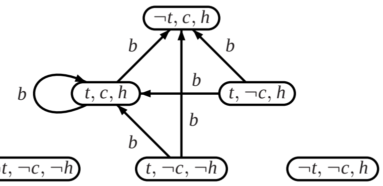

Figure 3: Supra-model for the coffee machine scenario.

2.4 Prime Valuations

An atom p is *essential* to φ if and only if $p \in atm(\varphi')$ for all φ' such that $\models_{\mathsf{CPL}} \varphi \leftrightarrow \varphi'$. For instance, p_1 is essential to $\neg p_1 \wedge (\neg p_1 \vee p_2)$. $atm!(\varphi)$ will denote the essential atoms of φ. (If φ is a tautology or a contradiction, then $atm!(\varphi) = \emptyset$.)

For $\varphi \in \mathfrak{F}$, $\varphi*$ is the set of all $\varphi' \in \mathfrak{F}$ such that $\varphi \models_{\mathsf{CPL}} \varphi'$ and $atm(\varphi') \subseteq atm!(\varphi)$. For instance, $p_1 \vee p_2 \notin p_1*$, as $p_1 \models_{\mathsf{CPL}} p_1 \vee p_2$ but $atm(p_1 \vee p_2) \not\subseteq atm!(p_1)$. Clearly, $atm(\bigwedge \varphi*) = atm!(\bigwedge \varphi*)$. Moreover, whenever $\models_{\mathsf{CPL}} \varphi \leftrightarrow \varphi'$, then $atm!(\varphi) = atm!(\varphi')$ and also $\varphi* = \varphi'*$.

Theorem 1 ([Parikh, 1999]) *$\models_{\mathsf{CPL}} \varphi \leftrightarrow \bigwedge \varphi*$, and $atm(\varphi*) \subseteq atm(\varphi')$ for every φ' s.t. $\models_{\mathsf{CPL}} \varphi \leftrightarrow \varphi'$.*

Thus for every $\varphi \in \mathfrak{F}$ there is a unique least set of elementary atoms such that φ may equivalently be expressed using only atoms from that set. Hence, $Cn(\varphi) = Cn(\varphi*)$.

Given a valuation v, $v' \subseteq v$ is a *subvaluation*. For W a set of valuations, a subvaluation v' *satisfies* $\varphi \in \mathfrak{F}$ modulo W (noted $v' \Vdash_{W} \varphi$) if and only if $v \Vdash \varphi$ for all $v \in W$ such that $v' \subseteq v$. A subvaluation v *essentially satisfies* φ modulo W ($v \Vdash^{!}_{W} \varphi$) if and only if $v \Vdash_{W} \varphi$ and $\{|\ell| : \ell \in v\} \subseteq atm!(\varphi)$.

Definition 3 *Let $\varphi \in \mathfrak{F}$ and W be a set of valuations. A subvaluation v is a* prime subvaluation *of φ (modulo W) if and only if $v \Vdash^!_W \varphi$ and there is no $v' \subseteq v$ s.t. $v' \Vdash^!_W \varphi$.*

A prime subvaluation of a formula φ is one of the weakest states of truth in which φ is true. (Notice the similarity with the syntactical notion of prime implicant [Quine, 1952].) We denote all prime subvaluations of φ modulo W by $base(\varphi, W)$.

Theorem 2 *Let $\varphi \in \mathfrak{F}$ and W be a set of valuations. Then for all $w \in W$, $w \Vdash \varphi$ if and only if $w \Vdash \bigvee_{v \in base(\varphi,W)} \bigwedge_{\ell \in v} \ell$.*

2.5 Closeness Between Models

When revising a model, we perform a change in its structure. Because there can be several ways of modifying a model (not all of them minimal), we need a notion of distance between models to identify those that are closest to the original one.

As we are going to see in more depth in the sequel, changing a model amounts to modifying its possible worlds or its accessibility relation. Hence, the distance between two K_n-models will depend upon the distance between their sets of worlds and accessibility relations. These here will be based on the *symmetric difference* between sets, defined as $X \dot{-} Y = (X \setminus Y) \cup (Y \setminus X)$.

Definition 4 *Let $\mathscr{M} = \langle W, R \rangle$. $\mathscr{M}' = \langle W', R' \rangle$ is* at least as close to *$\mathscr{M}$ as $\mathscr{M}'' = \langle W'', R'' \rangle$, noted $\mathscr{M}' \preceq_{\mathscr{M}} \mathscr{M}''$, iff*

- *either $W \dot{-} W' \subseteq W \dot{-} W''$*
- *or $W \dot{-} W' = W \dot{-} W''$ and $R \dot{-} R' \subseteq R \dot{-} R''$*

This is an extension of Burger and Heidema's [2002] relation to our modal case. Note that other distance notions are also possible, like e.g. the *cardinality* of symmetric differences or Hamming distance. (See Section 7 for an explanation on why we have chosen this particular distance notion here.)

3 Semantics of Revision

Contrary to contraction, where we want the negation of a law to be *satisfiable*, in revision we want a new law to be *valid*. Thus we must eliminate all cases satisfying its negation.

The idea in our semantics is as follows: we initially have a set of models $\mathcal{M}$ in which a given formula Φ is (potentially) not valid, i.e., Φ is (possibly) not true in every model in $\mathcal{M}$. In the result we want to have only models of Φ. Adding Φ-models to $\mathcal{M}$ is of no help. Moreover, adding models makes us lose laws: the resulting theory would be more liberal.

One solution amounts to deleting from $\mathcal{M}$ those models that are not Φ-models. Of course removing only some of them does not solve the problem, we must delete every such a model. By doing that, all resulting models will be models of Φ. (This corresponds to *theory expansion*, when the resulting theory is satisfiable.) However, if $\mathcal{M}$ contains no model of Φ, we will end up with $\emptyset$. Consequence: the resulting theory is inconsistent. (This is the main revision problem.) In this case the solution is to *substitute* each model $\mathscr{M}$ in $\mathcal{M}$ by its *nearest modifications* $\mathscr{M}^{\star}_{\Phi}$ that makes Φ true. This lets us to keep as close as possible to the original models we had.

Before defining revision of sets of models, we present what modifications of (individual) models are.

3.1 Revising a Model by a Static Law

Suppose that our agent discovers that the only hot drink that is served on the machine is coffee. In this case, we might want to revise her beliefs with the new static law *coffee* $\leftrightarrow$ *hot*.

Considering the model in Figure 3, we see that $\neg$*coffee* $\wedge$ *hot* is satisfiable. As we do not want this, the first step is to *remove* all worlds in which $\neg$*coffee* $\wedge$ *hot* is true. The second step is to guarantee all the remaining worlds satisfy the new law. This issue has been largely addressed in the literature on belief revision and update [Gärdenfors, 1988; Winslett, 1988; Katsuno and Mendelzon, 1992; Herzig and Rifi, 1999]. Here we can achieve that with a semantics similar to that of classical revision operators: basically one can change the set of possible valuations, by removing or adding worlds.

In our example, removing the possible worlds $\{t, \neg c, h\}$ and $\{\neg t, \neg c, h\}$ would do the job (there is no need to add new valuations since the new static law is satisfied in at least one world of the original model).

The delicate point in removing worlds is that it may result in the loss of some executability laws: in the example, if there were only one arrow leaving some world w and pointing to $\{\neg t, \neg c, h\}$, then removing the latter from the model would make the action under concern no longer executable in w. Here we claim that this is intuitive: if the state of the world to which we could move is no longer possible, then we do not have a transition to that state anymore. Hence, if that transition was the only one we had, it is natural to lose it.

One could also ask what to do with the accessibility relation if new worlds must be added (revision case). We claim that it is reckless to blindly add new elements to R. Instead, we shall postpone correction of executability laws, if needed. This approach is debatable, but with the information we have at hand, it is the safest way of changing static laws.

The semantics for revision of one model by a static law is as follows:

Definition 5 *Let $\mathscr{M} = \langle W, R \rangle$. $\mathscr{M}' = \langle W', R' \rangle \in \mathscr{M}^{\star}_{\varphi}$ iff $W' = (W \setminus val(\neg\varphi)) \cup val(\varphi)$ and $R' \subseteq R$.*

Clearly $\models^{\mathscr{M}'} \varphi$ for all $\mathscr{M}' \in \mathscr{M}^{\star}_{\varphi}$. The minimal models of the revision of $\mathscr{M}$ by φ are those closest to $\mathscr{M}$ w.r.t. $\preceq_{\mathscr{M}}$:

Definition 6 $rev(\mathscr{M}, \varphi) = \bigcup \min\{\mathscr{M}^{\star}_{\varphi}, \preceq_{\mathscr{M}}\}$.

In the example of Figure 3, $rev(\mathscr{M}, coffee \leftrightarrow hot)$ is the singleton $\{\mathscr{M}'\}$, with $\mathscr{M}'$ as shown in Figure 4.

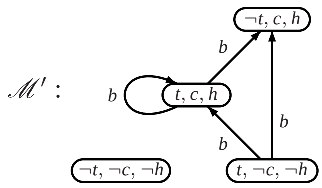

Figure 4: Revising model $\mathscr{M}$ in Figure 3 with *coffee* $\leftrightarrow$ *hot*.

3.2 Revising a Model by an Effect Law

Let's suppose now that our agent eventually discovers that after buying coffee she does not keep her token. This means that her theory should now be revised by the new effect law *token* $\rightarrow$ [*buy*]$\neg$*token*. Looking at model $\mathscr{M}$ in Figure 3, this amounts to guaranteeing that *token* $\wedge \langle$*buy*$\rangle$*token* is satisfiable in none of its worlds. To do that, we have to look at all the worlds satisfying this formula (if any) and

- either make *token* false in each of these worlds,
- or make $\langle$*buy*$\rangle$*token* false in all of them.

If we chose the first option, we will essentially flip the truth value of literal *token* in the respective worlds, which changes the set of valuations of the model. If we chose the latter, we will basically remove *buy*-arrows leading to *token*-worlds, which amounts to changing the accessibility relation.

In our example, the worlds $w_1 = \{token, coffee, hot\}$, $w_2 = \{token, \neg coffee, hot\}$ and $w_3 = \{token, \neg coffee, \neg hot\}$ satisfy *token* $\wedge \langle$*buy*$\rangle$*token*. Flipping *token* in all of them to $\neg$*token* would do the job, but would also have as consequence the introduction of a new static law: $\neg$*token* would now be valid, i.e., the agent never has a token! Do we want this?

We claim that changing action laws should not have as side effect a change in the static laws. These have a special status, and should change only if required (see Section 3.1). Hence each world satisfying *token* $\wedge \langle$*buy*$\rangle$*token* has to be changed so that $\langle$*buy*$\rangle$*token* becomes untrue in it. In the example, we thus should remove (w_1, w_1), (w_2, w_1) and (w_3, w_1) from *R*.

The semantics of one model revision for the case of a new effect law is:

Definition 7 *Let* $\mathscr{M} = \langle W, R\rangle$. $\mathscr{M}' = \langle W', R'\rangle \in \mathscr{M}^{\star}_{\varphi \rightarrow [a]\psi}$ *iff:*

- $W' = W$, $R' \subseteq R$, $\models^{\mathscr{M}'} \varphi \rightarrow [a]\psi$, *and*
- *If* $(w, w') \in R \setminus R'$, *then* $\models^{\mathscr{M}}_{w} \varphi$

The minimal models resulting from the revision of a model $\mathscr{M}$ by a new effect law are those closest to $\mathscr{M}$ w.r.t. $\preceq_{\mathscr{M}}$:

Definition 8 $rev(\mathscr{M}, \varphi \rightarrow [a]\psi) = \bigcup \min\{\mathscr{M}^{\star}_{\varphi \rightarrow [a]\psi}, \preceq_{\mathscr{M}}\}$.

Taking $\mathscr{M}$ as in Figure 3, $rev(\mathscr{M}, token \rightarrow [buy]\neg token)$ will be the singleton $\{\mathscr{M}'\}$ depicted in Figure 5.

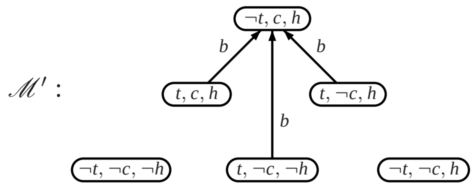

Figure 5: Revising $\mathscr{M}$ in Figure 3 with *token* $\rightarrow$ [*buy*]$\neg$*token*.

Note that adding effect laws will never require new arrows. This is the job of executability-revision.

3.3 Revising a Model by an Executability Law

Let us now suppose that at some stage it has been decided to grant free coffee to everybody. Faced with this information, we have to revise the agent's laws to reflect the fact that *buy* can also be executed in $\neg$*token*-contexts: $\neg$*token* $\rightarrow \langle$*buy*$\rangle\top$ is a new executability law (and hence we will have $\langle$*buy*$\rangle\top$ in all new models of the agent's beliefs).

Considering model $\mathscr{M}$ in Figure 3, we observe that $\neg$*token* $\wedge$ [*buy*]$\bot$ is satisfiable. Hence we must throw $\neg$*token* $\wedge$ [*buy*]$\bot$ away to ensure the new law becomes true.

To remove $\neg$*token* $\wedge$ [*buy*]$\bot$ we have to look at all worlds satisfying it and modify $\mathscr{M}$ so that they no longer satisfy that formula. Given worlds $w_4 = \{\neg token, \neg coffee, \neg hot\}$ and $w_5 = \{\neg token, \neg coffee, hot\}$, we have two options: change the interpretation of *token* in both or add new arrows leaving these worlds. A question that arises is 'what choice is more drastic: change a world or an arrow'? Again, here we claim that changing the world's content (the valuation) is more drastic, as the existence of such a world is foreseen by some static law and is hence assumed to be as it is, unless we have enough information supporting the contrary, in which case we explicitly change the static laws (see Section 3.1). Thus we shall add a new *buy*-arrow from each of w_4 and w_5.

Having agreed on that, the issue now is: which worlds should the new arrows point to? In order to comply with minimal change, the new arrows shall point to worlds that are relevant targets of each of the $\neg$*token*-worlds in question.

Definition 9 *Let* $\mathscr{M} = \langle W, R\rangle$, $w, w' \in W$, *and* $\mathcal{M}$ *be a set of models s.t.* $\mathscr{M} \in \mathcal{M}$. *Then* w' *is a* relevant target world *of* w *w.r.t.* $\varphi \rightarrow \langle a\rangle\top$ *for* $\mathscr{M}$ *in* $\mathcal{M}$ *iff* $\models^{\mathscr{M}}_{w} \varphi$ *and*

- *If there is* $\mathscr{M}' = \langle W', R'\rangle \in \mathcal{M}$ *such that* $R'_a(w) \neq \emptyset$*:*
  - *for all* $\ell \in w' \setminus w$, *there is* $\psi' \in \mathfrak{F}$ *s.t. there is* $v' \in base(\psi', W)$ *s.t.* $v' \subseteq w'$, $\ell \in v'$, *and* $\models^{\mathscr{M}_i}_{w} [a]\psi'$ *for every* $\mathscr{M}_i \in \mathcal{M}$
  - *for all* $\ell \in w \cap w'$, *either there is* $\psi' \in \mathfrak{F}$ *s.t. there is* $v' \in base(\psi', W)$ *s.t.* $v' \subseteq w'$, $\ell \in v'$, *and* $\models^{\mathscr{M}_i}_{w} [a]\psi'$ *for all* $\mathscr{M}_i \in \mathcal{M}$*; or there is* $\mathscr{M}_i \in \mathcal{M}$ *s.t.* $\not\models^{\mathscr{M}_i}_{w} [a]\neg\ell$
- *If* $R'_a(w) = \emptyset$ *for every* $\mathscr{M}' = \langle W', R'\rangle \in \mathcal{M}$*:*
  - *for all* $\ell \in w' \setminus w$, *there is* $\mathscr{M}_i = \langle W_i, R_i\rangle \in \mathcal{M}$ *s.t. there is* $u, v \in W_i$ *s.t.* $(u, v) \in R_{i_a}$ *and* $\ell \in v \setminus u$
  - *for all* $\ell \in w \cap w'$, *there is* $\mathscr{M}_i = \langle W_i, R_i\rangle \in \mathcal{M}$ *s.t. there is* $u, v \in W_i$ *s.t.* $(u, v) \in R_{i_a}$ *and* $\ell \in u \cap v$, *or for all* $\mathscr{M}_i = \langle W_i, R_i\rangle \in \mathcal{M}$, *if* $(u, v) \in R_{i_a}$, *then* $\neg\ell \notin v \setminus u$

By $rt(w, \varphi \rightarrow \langle a\rangle\top, \mathscr{M}, \mathcal{M})$ *we denote the set of all relevant target worlds of* w *w.r.t.* $\varphi \rightarrow \langle a\rangle\top$ *for* $\mathscr{M}$ *in* $\mathcal{M}$.

In our example, $w_6 = \{\neg token, coffee, hot\}$ is the only relevant target world here: the two other $\neg$*token*-worlds violate the direct effect *coffee* of action *buy*, while the three *token*-worlds would make us violate the frame axiom $\neg$*token* $\rightarrow$ [*buy*]$\neg$*token*.

The semantics for one model revision by a new executability law is as follows:

Definition 10 *Let* $\mathscr{M} = \langle W, R\rangle$. $\mathscr{M}' = \langle W', R'\rangle \in \mathscr{M}^{\star}_{\varphi \rightarrow \langle a\rangle\top}$ *iff:*

- $W' = W$, $R \subseteq R'$, $\models^{\mathscr{M}'} \varphi \rightarrow \langle a\rangle\top$, *and*
- *If* $(w, w') \in R' \setminus R$, *then* $w' \in rt(w, \varphi \rightarrow [a]\bot, \mathscr{M}, \mathcal{M})$

The minimal models resulting from revising a model $\mathscr{M}$ by a new executability law are those closest to $\mathscr{M}$ w.r.t. $\preceq_{\mathscr{M}}$:

Definition 11 $rev(\mathscr{M}, \varphi \rightarrow \langle a\rangle\top) = \bigcup \min\{\mathscr{M}^{\star}_{\varphi \rightarrow \langle a\rangle\top}, \preceq_{\mathscr{M}}\}$.

In our running example, $rev(\mathscr{M}, \neg token \rightarrow \langle buy\rangle\top)$ is the singleton $\{\mathscr{M}'\}$, where $\mathscr{M}'$ is as shown in Figure 6.

$\mathscr{M}'$:

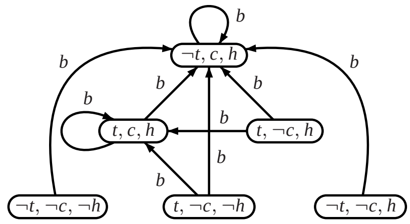

Figure 6: The result of revising model $\mathscr{M}$ in Figure 3 by the new executability law $\neg token \rightarrow \langle buy \rangle \top$.

3.4 Revising Sets of Models

Up until now we have seen what the revision of single models means. Now we are ready for a unified definition of revision of a set of models $\mathcal{M}$ by a new law Φ (cf. Section 5):

Definition 12 *Let $\mathcal{M}$ be a set of models and Φ a law. Then*

$$\mathcal{M}^{\star}_{\Phi} = (\mathcal{M} \setminus \{\mathscr{M} : \not\models^{\mathscr{M}} \Phi\}) \cup \bigcup_{\mathscr{M} \in \mathcal{M}} rev(\mathscr{M}, \Phi)$$

Definition 12 comprises both *expansion* and *revision*: in the former, addition of the new law gives a satisfiable theory; in the latter a deeper change is needed to get rid of inconsistency.

4 Algorithms for Revision of Laws

We now turn our attention to the syntactical counterpart of revision. Our endeavor here is to perform minimal change also at the syntactical level. By $\mathcal{T}^{\star}_{\Phi}$ we denote the result of revising an action theory $\mathcal{T}$ with a new law Φ.

4.1 Revising a Theory by a Static Law

Looking at the semantics of revision by Boolean formulas, we see that revising an action theory by a new static law may conflict with the executability laws: some of them may be lost and thus have to be changed as well.

The approach here is to preserve as many executability laws as we can in the old possible states. To do that, we look at each possible valuation that is common to the new $\mathcal{S}$ and the old one. Every time an executability used to hold in that state and no inexecutability holds there now, we make the action executable in such a context. For those contexts not allowed by the old $\mathcal{S}$, we make a inexecutable (cf. Section 3.1). Algorithm 1 deals with that (here $\mathcal{S} \star \varphi$ denotes the classical revision of $\mathcal{S}$ by φ built upon some well established method from the literature [Winslett, 1988; Katsuno and Mendelzon, 1992; Herzig and Rifi, 1999]. The choice of a particular operator for classical revision/update is not the main matter here, but rather whether it gives us a modified set of static laws entailing the new one).

In our example, revising the action theory $\mathcal{T}$ with a new static law $coffee \leftrightarrow hot$ will give us

$$\mathcal{T}^{\star}_{coffee \leftrightarrow hot} = \left\{ \begin{array}{c} coffee \leftrightarrow hot, \\ (token \wedge coffee \wedge hot) \rightarrow \langle buy \rangle \top, \\ (token \wedge \neg coffee \wedge \neg hot) \rightarrow \langle buy \rangle \top, \\ \neg coffee \rightarrow [buy]coffee, \neg token \rightarrow [buy]\bot, \\ coffee \rightarrow [buy]coffee, hot \rightarrow [buy]hot \end{array} \right\}$$

Algorithm 1 Revision by a Static Law

input: $\mathcal{T}, \varphi$
output: $\mathcal{T}^{\star}_{\varphi}$
  $\mathcal{S}' := \mathcal{S} \star \varphi$ /* classically revise $\mathcal{S}$ */
  $\mathcal{E}' := \mathcal{E}$ /* effect laws remain unchanged */
  $\mathcal{X}' := \emptyset$ /* executability laws will be 'recovered' from old $\mathcal{T}$ */
  for all $\pi \in IP(\mathcal{S}')$ **do**
    for all $A \subseteq \overline{atm(\pi)}$ **do**
      $\varphi_A := \bigwedge_{\substack{p_i \in \overline{atm(\pi)} \\ p_i \in A}} p_i \wedge \bigwedge_{\substack{p_i \in \overline{atm(\pi)} \\ p_i \notin A}} \neg p_i$
      /* by extending π with φ_A we get a valuation */
      if $\mathcal{S}' \not\models_{\mathsf{CPL}} (\pi \wedge \varphi_A) \rightarrow \bot$ /* context not removed */ **then**
        if $\mathcal{S} \not\models_{\mathsf{CPL}} (\pi \wedge \varphi_A) \rightarrow \bot$ **then**
          if $\mathcal{T} \models_{\mathsf{K}_n} (\pi \wedge \varphi_A) \rightarrow \langle a \rangle \top$ **and** $\mathcal{S}', \mathcal{E}', \mathcal{X} \not\models_{\mathsf{K}_n} \neg(\pi \wedge \varphi_A)$ **then**
            $\mathcal{X}_a' := \{(\varphi_i \wedge \pi \wedge \varphi_A) \rightarrow \langle a \rangle \top : \varphi_i \rightarrow \langle a \rangle \top \in \mathcal{X}_a\}$
            /* preserve executability law in state not removed */
        else
          $\mathcal{E}' := \mathcal{E}' \cup \{(\pi \wedge \varphi_A) \rightarrow [a]\bot\}$
  $\mathcal{T}^{\star}_{\varphi} := \mathcal{S}' \cup \mathcal{E}' \cup \mathcal{X}'$

4.2 Revising a Theory by an Effect Law

When revising a theory by a new effect law $\varphi \rightarrow [a]\psi$, we want to eliminate all possible executions of a leading to $\neg\psi$-states. To achieve that, we look at all φ-contexts and every time a transition to some $\neg\psi$-context is not always the case, i.e., $\mathcal{T} \not\models_{\mathsf{K}_n} \varphi \rightarrow \langle a \rangle \neg\psi$, we can safely force $[a]\psi$ for that context. On the other hand, if in such a context there is always an execution of a to $\neg\psi$, then we should strengthen the executability laws to make room for the new effect in that context we want to add. Algorithm 2 below does the job.

Algorithm 2 Revision by an Effect Law

input: $\mathcal{T}, \varphi \rightarrow [a]\psi$
output: $\mathcal{T}^{\star}_{\varphi \rightarrow [a]\psi}$
  $\mathcal{T}' := \mathcal{T}$
  for all $\pi \in IP(\mathcal{S} \wedge \varphi)$ **do**
    for all $A \subseteq \overline{atm(\pi)}$ **do**
      $\varphi_A := \bigwedge_{\substack{p_i \in \overline{atm(\pi)} \\ p_i \in A}} p_i \wedge \bigwedge_{\substack{p_i \in \overline{atm(\pi)} \\ p_i \notin A}} \neg p_i$
      /* by extending π with φ_A we get a valuation */
      if $\mathcal{S} \not\models_{\mathsf{CPL}} (\pi \wedge \varphi_A) \rightarrow \bot$ /* is an allowed context */ **then**
        for all $\pi' \in IP(\mathcal{S} \wedge \neg\psi)$ **do**
          if $\mathcal{T}' \models_{\mathsf{K}_n} (\pi \wedge \varphi_A) \rightarrow \langle a \rangle \pi'$ /* $\neg\psi$ is achievable */ **then**

$$\mathcal{T}' := \begin{array}{r} (\mathcal{T}' \setminus \mathcal{X}_a') \cup \{(\varphi_i \wedge \neg(\pi \wedge \varphi_A)) \rightarrow \langle a \rangle \top : \\ \varphi_i \rightarrow \langle a \rangle \top \in \mathcal{X}_a'\} \end{array}$$

            /* weaken executability laws */
        $\mathcal{T}' := \mathcal{T}' \cup \{(\pi \wedge \varphi_A) \rightarrow [a]\psi\}$ /* safely add the law */
        if $\mathcal{T}' \not\models_{\mathsf{K}_n} (\pi \wedge \varphi_A) \rightarrow [a]\bot$ **then**
          $\mathcal{T}' := \mathcal{T}' \cup \{(\varphi_i \wedge \pi \wedge \varphi_A) \rightarrow \langle a \rangle \top : \varphi_i \rightarrow \langle a \rangle \top \in \mathcal{T}\}$
          /* preserve other previous transitions */
  $\mathcal{T}^{\star}_{\varphi \rightarrow [a]\psi} := \mathcal{T}'$

In our running example, revision of the action theory $\mathcal{T}$ with the new effect law $token \rightarrow [buy]\neg token$ would give us

$\mathcal{T}^{\star}_{token \rightarrow [buy]\neg token} =$

$$\left\{ \begin{array}{c} coffee \rightarrow hot, \\ (token \wedge \neg(token \wedge coffee \wedge hot)) \rightarrow \langle buy \rangle \top, \\ (token \wedge coffee \wedge hot) \rightarrow \langle buy \rangle \top, \\ (token \wedge \neg(token \wedge \neg coffee \wedge hot)) \rightarrow \langle buy \rangle \top, \\ (token \wedge \neg coffee \wedge hot) \rightarrow \langle buy \rangle \top, \\ (token \wedge \neg(token \wedge \neg coffee \wedge \neg hot)) \rightarrow \langle buy \rangle \top, \\ (token \wedge \neg coffee \wedge \neg hot) \rightarrow \langle buy \rangle \top, \\ \neg coffee \rightarrow [buy]coffee, \neg token \rightarrow [buy]\bot, \\ coffee \rightarrow [buy]coffee, hot \rightarrow [buy]hot, \\ token \rightarrow [buy]\neg token \end{array} \right\}$$

Regarding the bunch of new executability laws introduced in the resulting theory, observe that they can be easily simplified to the single one $token \rightarrow \langle buy \rangle \top$.

4.3 Revising a Theory by an Executability Law

Revision of a theory by a new executability law has as consequence a change in the effect laws: all those laws preventing the execution of *a* shall be weakened. Moreover, to comply with minimal change, we must ensure that in all models of the resulting theory there will be at most *one* transition by *a* from those worlds in which $\mathcal{T}$ precluded *a*'s execution.

Let $(\mathcal{E}_a^{\varphi,\bot})_1, \ldots, (\mathcal{E}_a^{\varphi,\bot})_n$ denote minimum subsets (w.r.t. set inclusion) of $\mathcal{E}_a$ such that $\mathcal{S}, (\mathcal{E}_a^{\varphi,\bot})_i \models_{\mathsf{K}_n} \varphi \rightarrow [a]\bot$. (According to Herzig and Varzinczak [2007], one can ensure at least one such a set always exists.) Let $\mathcal{E}_a^- = \bigcup_{1 \leq i \leq n} (\mathcal{E}_a^{\varphi,\bot})_i$. The effect laws in $\mathcal{E}_a^-$ will serve as guidelines to get rid of $[a]\bot$ in each φ-world allowed by $\mathcal{T}$: they are the laws to be weakened to allow for $\langle a \rangle \top$ in φ-contexts.

Our algorithm works as follows. To force $\varphi \rightarrow \langle a \rangle \top$ to be true in all models of the resulting theory, we visit every possible φ-context allowed by it and make the following operations to ensure $\langle a \rangle \top$ is the case for that context: Given a φ-context, if $\mathcal{T}$ does not always preclude *a* from being executed in it, we can safely force $\langle a \rangle \top$ without modifying other laws. On the other hand, if *a* is always inexecutable in that context, then we should weaken the laws in $\mathcal{E}_a^-$. The first thing we must do is to preserve all old effects in all other φ-worlds. To achieve that we specialize the above laws to each possible valuation (maximal conjunction of literals) satisfying φ but the actual one. Then, in the current φ-valuation, we must ensure that action *a* may have any effect, i.e., from this φ-world we can reach any other possible world. We achieve that by weakening the *consequent* of the laws in $\mathcal{E}_a^-$ to the exclusive disjunction of all possible contexts in $\mathcal{T}$. Finally, to get minimal change, we must ensure that all literals in this φ-valuation that are not forced to change are preserved. We do this by stating a conditional frame axiom of the form $(\varphi_k \wedge \ell) \rightarrow [a]\ell$, where φ_k is the above-mentioned φ-valuation.

Algorithm 3 gives the pseudo-code for that.

Algorithm 3 Revision by an Executability Law

input: $\mathcal{T}, \varphi \rightarrow \langle a \rangle \top$
output: $\mathcal{T}^{\star}_{\varphi \rightarrow \langle a \rangle \top}$
$\mathcal{T}' := \mathcal{T}$
for all $\pi \in IP(\mathcal{S} \wedge \varphi)$ **do**
for all $A \subseteq \overline{atm(\pi)}$ **do**
$\varphi_A := \bigwedge_{\substack{p_i \in \overline{atm(\pi)} \\ p_i \in A}} p_i \wedge \bigwedge_{\substack{p_i \in \overline{atm(\pi)} \\ p_i \notin A}} \neg p_i$
/* by extending π with φ_A we get a valuation */
if $\mathcal{S} \not\models_{\mathsf{CPL}} (\pi \wedge \varphi_A) \rightarrow \bot$ /* is an allowed context */ **then**
if $\mathcal{T}' \models_{\mathsf{K}_n} (\pi \wedge \varphi_A) \rightarrow [a]\bot$ **then**

$$\mathcal{T}' := \begin{array}{r} (\mathcal{T}' \setminus \mathcal{E}'^{-}_{a}) \cup \{(\varphi_i \wedge \neg(\pi \wedge \varphi_A)) \rightarrow [a]\psi_i : \\ \varphi_i \rightarrow [a]\psi_i \in \mathcal{E}'^{-}_{a}\} \cup \\ \{(\varphi_i \wedge \pi \wedge \varphi_A) \rightarrow [a] \bigoplus_{\substack{\pi' \in IP(\mathcal{S}) \\ A' \subseteq \overline{atm(\pi')}}} (\pi' \wedge \varphi_{A'}) : \\ \varphi_i \rightarrow [a]\psi_i \in \mathcal{E}'^{-}_{a}\} \end{array}$$

/* weaken the effect laws */
for all $L \subseteq \mathfrak{L}$ **do**
if $\mathcal{S} \models_{\mathsf{CPL}} (\pi \wedge \varphi_A) \rightarrow \bigwedge_{\ell \in L} \ell$ **then**
for all $\ell \in L$ **do**
if $\mathcal{T} \models_{\mathsf{K}_n} \ell \rightarrow [a]\bot$ **or** $(\mathcal{T} \not\models_{\mathsf{K}_n} \ell \rightarrow [a]\neg\ell$ **and** $\mathcal{T} \models_{\mathsf{K}_n} \ell \rightarrow [a]\ell)$ **then**
$\mathcal{T}' := \mathcal{T}' \cup \{(\pi \wedge \varphi_A \wedge \ell) \rightarrow [a]\ell\}$
/* preserve non-affected literals */
$\mathcal{T}' := \mathcal{T}' \cup \{(\pi \wedge \varphi_A) \rightarrow \langle a \rangle \top\}$ /* safely add the law */
$\mathcal{T}^{\star}_{\varphi \rightarrow \langle a \rangle \top} := \mathcal{T}'$

In our example, revising the action theory $\mathcal{T}$ with the executability law $\neg token \rightarrow \langle buy \rangle \top$ gives us $\mathcal{T}^{\star}_{\neg token \rightarrow \langle buy \rangle \top} =$

$$\left\{ \begin{array}{c} coffee \rightarrow hot, token \rightarrow \langle buy \rangle \top, \\ \neg coffee \rightarrow [buy]coffee, \\ (\neg token \wedge \neg(\neg token \wedge coffee \wedge hot) \wedge \\ \neg(\neg token \wedge \neg coffee \wedge hot) \wedge \\ \neg(\neg token \wedge \neg coffee \wedge \neg hot)) \rightarrow [buy]\bot, \\ coffee \rightarrow [buy]coffee, hot \rightarrow [buy]hot, \\ (\neg token \wedge coffee \wedge hot) \rightarrow [buy]\neg token, \\ (\neg token \wedge \neg coffee \wedge hot) \rightarrow [buy]\neg token, \\ (\neg token \wedge \neg coffee \wedge \neg hot) \rightarrow [buy]\neg token, \\ \neg token \rightarrow \langle buy \rangle \top \end{array} \right\}$$

Again, the resulting theory can be post-processed to give us a much more compact representation of the new laws that have been added.

4.4 Complexity Issues

Algorithms 1–3 terminate. However, they come with a considerable computational cost: the K_n-entailment test with global axioms is known to be EXPTIME-complete. Computing all possible contexts allowed by the theory is clearly exponential. Moreover, the computation of $IP(.)$ might result in exponential growth [Marquis, 2000].

Given that theory change can be done offline, from the knowledge engineer's perspective what is more important is the size of the computed contracted theories. In that matter, our results are positive:

Theorem 3 *Let $\mathcal{T}$ be an action theory, and Φ be a law. Then the size (number of formulas) of $\mathcal{T}^{\star}_{\Phi}$ is linear in that of $\mathcal{T}$.*

5 Correctness of the Algorithms

Suppose we have two atoms p_1 and p_2, and one action a. Let $\mathcal{T}_1 = \{\neg p_2, p_1 \rightarrow [a]p_2, \langle a\rangle\top\}$. The only model of $\mathcal{T}_1$ is $\mathscr{M}$ in Figure 7. Revising such a model by $p_1 \vee p_2$ gives us the models $\mathscr{M}'_i$, $1 \leq i \leq 3$, in Figure 7. Now, revising $\mathcal{T}_1$ by $p_1 \vee p_2$ will give us $\mathcal{T}_1{}^{\star}_{p_1 \vee p_2} = \{p_1 \wedge \neg p_2, p_1 \rightarrow [a]p_2\}$. The only model of $\mathcal{T}_1{}^{\star}_{p_1 \vee p_2}$ is $\mathscr{M}'_1$ in Figure 7. This means that the semantic revision may produce models (viz. $\mathscr{M}'_2$ and $\mathscr{M}'_3$ in Figure 7) that are not models of the revised theories.

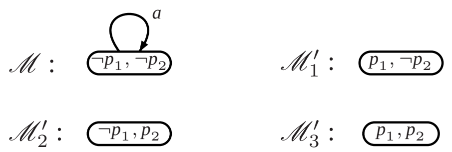

Figure 7: Model $\mathscr{M}$ of $\mathcal{T}_1$ and revision of $\mathscr{M}$ by $p_1 \vee p_2$.

The other way round the algorithms may give theories whose models do not result from revision of models of the initial theory: let $\mathcal{T}_2 = \{(p_1 \vee p_2) \rightarrow [a]\bot, \langle a\rangle\top\}$. Its only model is $\mathscr{M}$ (Figure 7). Revising $\mathscr{M}$ by $p_1 \vee p_2$ is as above. But $\mathcal{T}_2{}^{\star}_{p_1 \vee p_2} = \{p_1 \vee p_2, (p_1 \vee p_2) \rightarrow [a]\bot\}$ has a model $\mathscr{M}'' = \langle\{\{p_1, p_2\}, \{p_1, \neg p_2\}, \{\neg p_1, p_2\}\}, \emptyset\rangle$ not in $\mathscr{M}^{\star}_{p_1 \vee p_2}$.

All this happens because the possible states are not completely characterized by the static laws. Fortunately, concentrating on supra-models of $\mathcal{T}$, we get the right result.

Theorem 4 *If $\mathcal{M} = \{\mathscr{M} : \mathscr{M}$ is a supra-model of $\mathcal{T}\}$ and there is $\mathscr{M}' \in \mathcal{M}$ s.t. $\models^{\mathscr{M}'} \Phi$, then $\bigcup_{\mathscr{M} \in \mathcal{M}} rev(\mathscr{M}, \Phi) \subseteq \mathcal{M}$.*

Then, revision of models of $\mathcal{T}$ by a law Φ in the semantics produces models of the output of the algorithms $\mathcal{T}^{\star}_{\Phi}$:

Theorem 5 *If $\mathcal{M} = \{\mathscr{M} : \mathscr{M}$ is a supra-model of $\mathcal{T}\} \neq \emptyset$, then for every $\mathscr{M}' \in \mathcal{M}^{\star}_{\Phi}$, $\models^{\mathscr{M}'} \mathcal{T}^{\star}_{\Phi}$.*

Also, models of $\mathcal{T}^{\star}_{\Phi}$ result from revision of models of $\mathcal{T}$ by Φ:

Theorem 6 *If $\mathcal{M} = \{\mathscr{M} : \mathscr{M}$ is a supra-model of $\mathcal{T}\} \neq \emptyset$, then for every $\mathscr{M}'$, if $\models^{\mathscr{M}'} \mathcal{T}^{\star}_{\Phi}$, then $\mathscr{M}' \in \mathcal{M}^{\star}_{\Phi}$.*

Sticking to supra-models of $\mathcal{T}$ is not a big deal. We can use existent algorithms in the literature [Herzig and Varzinczak, 2007] to ensure that $\mathcal{T}$ is characterized by its supra-models and that $\mathcal{M} \neq \emptyset$.

6 Related Work

The problem of action theory change has only recently received attention in the literature, both in action languages [Baral and Lobo, 1997; Eiter *et al.*, 2005] and in modal logic [Herzig *et al.*, 2006; Varzinczak, 2008].

Baral and Lobo [1997] introduce extensions of action languages that allow for some causal laws to be stated as defeasible. Their work is similar to ours in that they also allow for weakening of laws: in their setting, effect propositions can be replaced by what they call defeasible (weakened versions of) effect propositions. Our approach is different from theirs in the way executability laws are dealt with. Here executability laws are explicit and we are also able to change them. This feature is important when the qualification problem is considered (cf. the Introduction).

The work by Eiter *et al.* [2005; 2006] is similar to ours in that they also propose a framework that is oriented to updating action laws. They mainly investigate the case where e.g. a new effect law is added to the description (and then has to be true in all models of the modified theory).

In Eiter *et al.*'s framework, action theories are described in a variant of a narrative-based action description language. Like in the present work, the semantics is also in terms of transition systems: directed graphs having arrows (action occurrences) linking nodes (configurations of the world). Contrary to us, however, the minimality condition on the outcome of the update is in terms of inclusion of sets of laws, which means the approach is more syntax oriented.

In their setting, during an update an action theory $\mathcal{T}$ is seen as composed of two pieces, $\mathcal{T}_u$ and $\mathcal{T}_m$, where $\mathcal{T}_u$ stands for the part of $\mathcal{T}$ that is not supposed to change and $\mathcal{T}_m$ contains the laws that may be modified. In our terms, when revising by a static law we would have $\mathcal{T}_m = \mathcal{S} \cup \mathcal{X}_a$, when revising by an effect law $\mathcal{T}_m = \mathcal{E}_a \cup \mathcal{X}_a$, and when revising with executability laws $\mathcal{T}_m = \mathcal{E}^-_a \cup \mathcal{X}_a$. The difference here is that in our approach it is always clear what laws should not change in a given type of revision, and $\mathcal{T}_u$ and $\mathcal{T}_m$ do not need to be explicitly specified prior to the update.

Their approach and ours can both be described as *constraint-based* update, in that the theory change is carried out relative to some restrictions (a set of laws that we want to hold in the result). In our framework, for example, all changes in the action laws are relative to the set of static laws $\mathcal{S}$ (and that is why we concentrate on supra-models: models of $\mathcal{T}$ having $val(\mathcal{S})$ as worlds). When changing a law, we want to keep the same set of states. The difference w.r.t. Eiter *et al.*'s approach is that there it is also possible to update a theory relatively to e.g. executability laws: when expanding $\mathcal{T}$ with a new effect law, one may want to constrain the change so that the action under concern is guaranteed to be executable in the result.[1] As shown in the referred work, this may require the withdrawal of some static law. Hence, in Eiter *et al.*'s framework, static laws do not have the same status as in ours.

7 Discussion and Perspectives

Here we have studied what revising action theories by a law means, both in the semantics and at the syntactical (algorithmic) level. We have defined a semantics based on distances between models that also captures minimal change w.r.t. the preservation of effects of actions. With our algorithms and the correctness results we have established the link between the semantics and the syntax for theories with supra-models. (Due to page limits, proofs have been omitted here.)

For the sake of presentation, here we have abstracted from the frame and ramification problems. However our definitions could have been stated in a formalism with a suitable solution to them, like e.g. Castilho *et al.*'s approach [1999]. With regards to the qualification problem, this is not ignored here:

[1]We can emulate that in our approach with two modifications of $\mathcal{T}$: first adding the effect law and then an executability law.

revising wrong executability laws is an approach towards its solution. Indeed, given the difficulty of stating all sufficient conditions for executability of an action, the knowledge engineer writes down some of them and lets the theory 'evolve' via subsequent revisions.

The reason why we have chosen such a simple notion of distance between models is that with other distances one may not always get the intended result. This is better illustrated with the *contraction counterpart* of our operators [Varzinczak, 2008]. Suppose one wants to remove an executability law $\varphi \rightarrow \langle a \rangle \top$. Then we do that by removing *a*-arrows from φ-worlds. Suppose we have a model with two φ-worlds, w_1 with one leaving *a*-arrow and w_2 with two *a*-arrows. Then with e.g. Dalal's distance [1988], the associated contraction operator would always exclude the resulting model in which w_2 loses its two arrows, simply because deleting 1 arrow is Dalal-better than deleting 2. This problem doesn't happen with our distance, which gives us a version of maxichoice [Hansson, 1999].

One criticism to the approach here developed concerns the precedence of static laws in the revision process, which could make the revision operators to be interpreted as incoherent. As agreed in the literature, however, given that static laws are much easier to state, they are more likely to be correct, and then it makes sense to give them precedence. Supporting this is the fact that most of the attention in the reasoning about actions area has been paid to effect laws and executability laws, which are much more difficult to completely specify. Our approach is in line with that.

Our next step is to analyze the behavior of our operators w.r.t. AGM-like postulates [Alchourrón *et al.*, 1985] for modal theories and the relationship between our revision method and contraction. What is known is that Levi identity [Levi, 1977], $\mathcal{T}^{\star}_{\Phi} = \mathcal{T}^{-}_{\neg\Phi} \cup \{\Phi\}$, in general does not hold for action laws. The reason is that up to now there is no contraction operator for $\neg\Phi$ where Φ is an action law. Indeed this is the general contraction problem for action theories: contraction of a theory $\mathcal{T}$ by a general formula (like $\neg\Phi$ above) is still an open problem in the area. The definition of a general method will mostly benefit from the semantic modifications we studied here (addition/removal of arrows and worlds).

Given the relationship between modal logics and description logics, a revision method for DL TBoxes [Baader *et al.*, 2003] would also benefit from the constructions that we have defined here.

References

[Alchourrón *et al.*, 1985] C. Alchourrón, P. Gärdenfors, and D. Makinson. On the logic of theory change: Partial meet contraction and revision functions. *J. of Symbolic Logic*, 50:510–530, 1985.

[Baader *et al.*, 2003] F. Baader, D. Calvanese, D. McGuinness, D. Nardi, and P. Patel-Schneider, editors. *Description Logic Handbook*. Cambridge University Press, 2003.

[Baral and Lobo, 1997] C. Baral and J. Lobo. Defeasible specifications in action theories. In *Proc. IJCAI*, pages 1441–1446, 1997.

[Burger and Heidema, 2002] I.C. Burger and J. Heidema. Merging inference and conjecture by information. *Synthese*, 131(2):223–258, 2002.

[Castilho *et al.*, 1999] M. Castilho, O. Gasquet, and A. Herzig. Formalizing action and change in modal logic I: the frame problem. *J. of Logic and Computation*, 9(5):701–735, 1999.

[Dalal, 1988] M. Dalal. Investigations into a theory of knowledge base revision: preliminary report. In *Proc. AAAI*, pages 475–479, 1988.

[Eiter *et al.*, 2005] T. Eiter, E. Erdem, M. Fink, and J. Senko. Updating action domain descriptions. In *Proc. IJCAI*, pages 418–423, 2005.

[Eiter *et al.*, 2006] T. Eiter, E. Erdem, M. Fink, and J. Senko. Resolving conflicts in action descriptions. In *Proc. ECAI*, pages 367–371, 2006.

[Gärdenfors, 1988] P. Gärdenfors. *Knowledge in Flux: Modeling the Dynamics of Epistemic States*. MIT Press, 1988.

[Hansson, 1999] S. Hansson. *A Textbook of Belief Dynamics: Theory Change and Database Updating*. Kluwer, 1999.

[Herzig and Rifi, 1999] A. Herzig and O. Rifi. Propositional belief base update and minimal change. *Artificial Intelligence*, 115(1):107–138, 1999.

[Herzig and Varzinczak, 2007] A. Herzig and I. Varzinczak. Metatheory of actions: beyond consistency. *Artificial Intelligence*, 171:951–984, 2007.

[Herzig *et al.*, 2006] A. Herzig, L. Perrussel, and I. Varzinczak. Elaborating domain descriptions. In *Proc. ECAI*, pages 397–401, 2006.

[Katsuno and Mendelzon, 1992] H. Katsuno and A. Mendelzon. On the difference between updating a knowledge base and revising it. In *Belief revision*, pages 183–203. Cambridge, 1992.

[Levi, 1977] I. Levi. Subjunctives, dispositions and chances. *Synthese*, 34:423–455, 1977.

[Marquis, 2000] P. Marquis. Consequence finding algorithms. In *Alg. for Defensible and Uncertain Reasoning*, pages 41–145. 2000.

[Parikh, 1999] R. Parikh. Beliefs, belief revision, and splitting languages. In *Logic, Language and Computation*, pages 266–278, 1999.

[Popkorn, 1994] S. Popkorn. *First Steps in Modal Logic*. Cambridge University Press, 1994.

[Quine, 1952] W. V. O. Quine. The problem of simplifying truth functions. *American Mathematical Monthly*, 59:521–531, 1952.

[Varzinczak, 2008] I. Varzinczak. Action theory contraction and minimal change. In *Proc. KR*, pages 651–661, 2008.

[Winslett, 1988] M.-A. Winslett. Reasoning about action using a possible models approach. In *Proc. AAAI*, pages 89–93, 1988.