From: AAAI Technical Report SS-95-07. Compilation copyright © 1995, AAAI (www.aaai.org). All rights reserved.

A Causal Theory of Ramifications and Qualifications (Extended Abstract)

Norman McCain* and Hudson Turner†
Department of Computer Sciences
University of Texas at Austin
Austin, TX 78712-1188, USA
{mccain,hudson}@cs.utexas.edu

1 Introduction

This paper is concerned with the problem of determining the indirect effects or ramifications of actions. The problem is usually investigated, as in [Kartha and Lifschitz, 1994], in a framework in which action domains are described in part by state constraints. (Informally, a state constraint is a formula that says of a proposition that it is true in every possible state of the world.) Our main objective is to argue that an adequate theory of ramifications requires the representation of information of a kind that is not conveyed by state constraints. In particular, what is required is the representation of the causal relations (or, more generally, the determination relations) that hold between states of affairs. It turns out that this is also the information that is needed for an adequate theory of derived action preconditions or qualifications.

Previous approaches to the problem of ramifications have assumed a definition of the following kind: A ramification, roughly speaking, is a change (not explicitly described) that is implied by the performance of an action. In our approach, we substitute the word "caused" for the word "implied". In determining the ramifications of actions, it is not enough, we say, to infer that a change must occur when an action is performed; it is necessary to infer that the action causes the change to occur. As we will see, this stronger requirement makes it possible to avoid unintended ramifications and to infer derived qualifications. (The need for the latter is argued in [Ginsberg and Smith, 1988] and [Lin and Reiter, 1994].) Again roughly speaking, our theory of qualifications is this: An action cannot be performed if the performance of the action implies a change that it does not cause.

The paper is divided into three parts. We begin in Section 2 by focusing on the core problem of defining the possible states of the world that may result when an action is performed. In order to motivate our final definition, we present a series of definitions for various frameworks in which different kinds of background knowledge are assumed. Specifically, we give definitions for the case without background knowledge, as in [Gelfond and Lifschitz, 1993], for the case in which there are state constraints, as in [Winslett, 1988] and [Kartha and Lifschitz, 1994], and for the case in which there are directed "causal laws", represented as inference rules. We also illustrate the differences between our framework with causal laws and the framework of Brewka and Hertzberg [1993], who also use inference rules to represent causal laws. In Section 3 we show how to express ramification and qualification constraints as causal laws. This completes the first part of the paper.

Representing causal laws by inference rules is a useful approximation, but it is not entirely adequate. In the second part of the paper (Sections 4 and 5), we give a semantic account of "causal laws", replacing inference rules by a rule-like conditional, for which a modal, partial state semantics is defined. Doing so allows us to clarify the sense in which our theory of ramifications and qualifications is a "causal" theory.

The third part of the paper is not included in this extended abstract. In the full version, we go on to define a complete high-level action description language $\mathcal{AC}$, based on the semantic framework with causal laws. Roughly speaking, $\mathcal{AC}$ is obtained from the language $\mathcal{AR}_0$ of Kartha and Lifschitz [1994] by replacing the state constraints of $\mathcal{AR}_0$ by causal laws.

2 Defining the Possible Next States

In this section, we focus on the core problem of defining the possible states of the world that may result when an action is performed.

We assume that we are given a standard language of propositional logic. We represent an interpretation for the language by a maximal consistent set of literals. Informally, we think of an interpretation as a logically possible state of the world. By an *explicit*

*Partially supported by National Science Foundation grant #IRI-9306751.

†Partially supported by an IBM Graduate Fellowship and by NSF grant #IRI-9306751.

effect we mean a set of formulas. Intuitively, these are the formulas that an action is explicitly said to cause. By a *formula constraint* we mean a formula. Informally, we think of a formula constraint as a state constraint.

The central problem in determining the ramifications of an action is to properly define $Res(E, S)$, the set of possible next states after performing an action with the explicit effect E in the state S. We will present a series of definitions for four different frameworks characterized by the presence or absence of the following elements: (i) the assumption of inertia, (ii) a set B of formula constraints, and (iii) a set C of inference rules. The framework that includes inertia and inference rules is the most important, since it is the one needed for an adequate account of ramifications and qualifications.[1]

In each of the frameworks, our definition will take the following form: For all interpretations S and S', and any explicit effect E, $S' \in Res(E, S)$ if S' is precisely the set of literals that are derivable from E and the background information given in the framework (possibly including information provided by the assumption of inertia). When a set B of formula constraints or a set C of inference rules is present in a framework we will write them as subscripts to *Res*. Throughout this paper, we will use the symbol L to stand exclusively for literals. The frameworks and their corresponding definitions will be numbered 1–4. We begin with the simplest framework.

2.1 Framework 1

In Framework 1, we do not assume the principle of inertia, nor do we include formula constraints or inference rules. Consequently, every literal in any possible next state must be derivable from E alone. This motivates the following definition.

Definition 1. For all interpretations S and S', and any explicit effect E, $S' \in Res^1(E, S)$ if

$$S' = \{L : E \vdash L\}.$$

Clearly, if the set $\{L : E \vdash L\}$ is an interpretation then this interpretation is the only element of $Res^1(E, S)$. Otherwise, either E does not imply a value for every atom or E is inconsistent. Since E is required to specify not only the values of the atoms that change but also the values of those that do not, the frame problem is unsolved in this framework.

Consider, for example, the state $S = \{p, q\}$ and an action that makes p false. Choosing $E = \{\neg p \wedge q\}$, we have $Res^1(E, S) = \{\{\neg p, q\}\}$.

[1] In [Przymusinski and Turner, 1994], it is shown that essentially this framework simplifies and generalizes the related notion of "revision programming" due to Marek and Truszczyński [1994]. It is also shown there that this framework has a simple embedding in default logic.

2.2 Framework 2

In Framework 2, we assume the principle of inertia, but do not include formula constraints or inference rules. This is the framework of the language $\mathcal{A}$ of Gelfond and Lifschitz [1993]. Here the frame problem is solved by the principle of inertia, which makes it possible to specify only the values of the atoms that change. The values of the other atoms are assumed to remain the same in S' as in S. The literals in $S \cap S'$ are those whose values are preserved by inertia. We obtain the definition for Framework 2 by adding the literals in $S \cap S'$ as additional premises to the derivability condition in Definition 1.

Definition 2. For all interpretations S and S', and any explicit effect E, $S' \in Res^2(E, S)$ if

$$S' = \{L : (S \cap S') \cup E \vdash L\}.$$

Consider again the state $S = \{p, q\}$ and an action that makes p false. Choosing $E = \{\neg p\}$, we have $Res^2(E, S) = \{\{\neg p, q\}\}$. Because of the assumption of inertia, $Res^2(E, S)$ may be nonempty even when the explicit effect does not imply a value for every atom.

The solutions to the equation in Definition 2 are the interpretations S' that are fixpoints of the function $\lambda X.\{L : (S \cap X) \cup E \vdash L\}$. The following example shows that there may be more than one fixpoint. Let $S = \{\neg p, \neg q\}$ and $E = \{p \vee q\}$. Then $Res^2(E, S) = \{\{\neg p, q\}, \{p, \neg q\}\}$. In the language $\mathcal{A}$ [Gelfond and Lifschitz, 1993], the explicit effect E is required to be a set of literals. Under this restriction, it is easy to see that whenever E is consistent $Res^2(E, S)$ will be a singleton.

2.3 Framework 3

In Framework 3, we assume the principle of inertia and include a set B of formula constraints. We do not include inference rules. A standard example of a formula constraint is the formula $(Walking \supset Alive)$ [Baker, 1991]. The framework with inertia and formula constraints is the one investigated, for instance, by Winslett [1988] and by Kartha and Lifschitz [1994]. We obtain the definition for Framework 3 by adding B to the premises of the derivability condition in Definition 2.

Definition 3. For all interpretations S and S', any explicit effect E, and any set B of formula constraints, $S' \in Res^3_B(E, S)$ if

$$S' = \{L : (S \cap S') \cup E \cup B \vdash L\}.$$

For example, let $S = \{Alive, Walking\}$, let $E = \{\neg Alive\}$, and let $B = \{Walking \supset Alive\}$. Then $Res^3_B(E, S) = \{\{\neg Alive, \neg Walking\}\}$. Here, $\neg Walking$ is a ramification.

Let us compare Definition 3 to Winslett's definition. In our terminology, the latter may be stated as follows.

Definition W [Winslett, 1988]. For all interpretations S and S', any explicit effect E, and any set B of formula constraints, $S' \in Res^W_B(E, S)$ if

(1) S' satisfies $E \cup B$, and

(2) no other interpretation that satisfies (1) differs from S on fewer atoms, where "fewer" is defined by set inclusion.

Winslett's definition expresses the idea of minimizing change. Definition 3 has a very different form; it is given in terms of a fixpoint condition. Despite this difference, the two definitions are equivalent, as the following proposition shows.

Proposition 1 *For any interpretation S, any explicit effect E, and any set B of formula constraints, $Res^W_B(E, S) = Res^3_B(E, S)$.*

2.4 Framework 4

In Framework 4, we assume the principle of inertia and include a set C of inference rules. We do not include formula constraints, because, as we will see, for the purpose of defining *Res* they can be easily represented by inference rules. We will write an inference rule as an expression of the form

$$\phi \Rightarrow \psi$$

where ϕ and ψ are formulas. An inference rule can be thought of as a kind of "directed constraint". Informally, we think of an inference rule $\phi \Rightarrow \psi$ as expressing a relation of determination between the states of affairs that make ϕ and ψ true. One kind of determination relation is causal. (For a discussion of noncausal determination relations see [Kim, 1974].) As an example, we think of the rule $\neg Alive \Rightarrow \neg Walking$ as expressing the fact that not being alive causes not walking. Of course, walking does not similarly cause being alive, which shows that the causal relation, like the inference rule, is "noncontrapositive". Later we will see that the representation of causal laws by inference rules, although a useful approximation, is not entirely adequate.

The standard derivability relation $\vdash$ of propositional logic is easily extended to take account of inference rules. Given a set Γ of formulas, a set C of inference rules, and a formula ϕ, we write

$$\Gamma \vdash_C \phi$$

to mean that ϕ is an element of the smallest set of formulas that contains Γ and is closed under C and propositional logic. We obtain the definition for *Res* in the present framework by replacing $\vdash$ by $\vdash_C$ in Definition 2.

Definition 4. For all interpretations S and S', any explicit effect E, and any set C of inference rules, $S' \in Res^4_C(E, S)$ if

$$S' = \{L : (S \cap S') \cup E \vdash_C L\}.$$

For example, let $S = \{Alive, Walking\}$, let $E = \{\neg Alive\}$, and let $C = \{\neg Alive \Rightarrow \neg Walking\}$. Then $Res^4_C(E, S) = \{\{\neg Alive, \neg Walking\}\}$. Again, $\neg Walking$ is a ramification.

One advantage of Framework 4 over Framework 3 is illustrated by the following variation on the previous example. Let $S = \{\neg Alive, \neg Walking\}$, $E = \{Walking\}$, with C as before and with $B = \{Walking \supset Alive\}$. Then $Res^4_C(E, S)$ is empty, whereas $Res^3_B(E, S) = \{\{Alive, Walking\}\}$. Intuitively, $Res^4_C(E, S)$ is correct. In state S one cannot perform an action whose explicit effect is $\{Walking\}$, because this effect implies a change (namely, making *Alive* true) that the action does not cause.[2] This is an example of a derived qualification.

Another advantage of Framework 4 can be illustrated using the domain introduced in [Lifschitz, 1990] in which there are two switches and a light. Let $S = \{\neg Up1, Up2, \neg On\}$ and $E = \{Up1\}$. In Framework 3, let $B = \{On \equiv (Up1 \equiv Up2)\}$. Then $Res^3_B(E, S)$ contains two states: $\{Up1, Up2, On\}$ and $\{Up1, \neg Up2, \neg On\}$. The second state in $Res^3_B(E, S)$ is anomalous, and results from the unintended ramification $\neg Up2$. In [Lifschitz, 1990] and [Kartha and Lifschitz, 1994], this ramification is blocked by declaring *Up1* and *Up2* to be "in the frame" and *On* to be "not in the frame". In Framework 4, the use of inference rules in place of formula constraints makes the frame/nonframe distinction unnecessary for the purpose of limiting possible ramifications. For instance, let C contain the inference rules $(Up1 \equiv Up2) \Rightarrow On$ and $\neg(Up1 \equiv Up2) \Rightarrow \neg On$. Then $Res^4_C(E, S) = \{\{Up1, Up2, On\}\}$.

Notice that in the previous examples, $Res^4_C(E, S)$ is a subset of $Res^3_B(E, S)$. The following proposition shows that this relationship holds whenever B and C are related as above.

Proposition 2 *Let C be a set of inference rules, and let $B = \{\phi \supset \psi : \phi \Rightarrow \psi \in C\}$. For any interpretation S and explicit effect E, $Res^4_C(E, S)$ is a subset of $Res^3_B(E, S)$.*

The relationship between Frameworks 3 and 4 that is captured in this proposition can be attributed to

[2] Intuitively, since not being alive causes not walking, the conditional $(Walking \supset Alive)$ holds in every possible state of the world, and in this sense *Walking* implies *Alive*. However, the inference rule $(\neg Alive \Rightarrow \neg Walking)$ does not capture this intuition. (It is weaker than the conditional, not stronger.) This is a deficiency in the representation of causal laws by inference rules which is remedied in Sections 4 and 5.

the fact that an inference rule $\phi \Rightarrow \psi$ is weaker than its corresponding formula constraint $(\phi \supset \psi)$, so that some of the changes implied by the formula constraint are not derivable via the inference rule.

Framework 4 can be compared with the framework of Brewka and Hertzberg [1993], who also use inference rules to represent causal laws. They present their framework as a modification of Winslett's [1988] in which causal laws play a role in the definition of minimal change between states. Because of the role of minimal change in their framework, they cannot obtain derived qualifications of the kind illustrated by the *Walking/Alive* example above. Nor can they express qualification constraints of the kind discussed in Section 3.

Even in cases where derived qualifications are not involved, Brewka and Hertzberg may obtain results different from those of Framework 4. In the following example we extend a domain discussed in [Brewka and Hertzberg, 1993]. Consider an action whose explicit effect is that either water or wine is on the table. Suppose that water is always available, but wine is not. A necessary condition for wine being on the table is that we have some. Unfortunately, we do not, and neither wine nor water have yet been served. Let S be the state

$$\{\neg Have(Wine), \neg Ontable(Water), \neg Ontable(Wine)\}$$

and let

$$E = \{Ontable(Water) \vee Ontable(Wine)\},$$
$$C = \{\neg Have(Wine) \Rightarrow \neg Ontable(Wine)\}.$$

Then $Res^4_C(E, S)$ contains the single state

$$\{\neg Have(Wine), Ontable(Water), \neg Ontable(Wine)\}.$$

In the framework of Brewka and Hertzberg, there is another possible resulting state, namely,

$$\{Have(Wine), \neg Ontable(Water), Ontable(Wine)\}$$

in which $Have(Wine)$ is a ramification. Intuitively, however, performing an action whose explicit effect is $\{Ontable(Water) \vee Ontable(Wine)\}$ does not cause one to have wine. So it appears that they obtain a ramification that is not caused. This never occurs in our theory of ramifications, as is explained in Section 5.

3 Ramification and Qualification Constraints

Lin and Reiter [1994] draw a pragmatic distinction between two kinds of state constraints: *ramification constraints*, which yield indirect effects, and *qualification constraints*, which yield action preconditions. As they observe, the same distinction was drawn earlier by Ginsberg and Smith [1988]. In the language of inference rules, we can give a syntactic form to this distinction. Suppose that ϕ is a formula constraint. If we wish ϕ to function as a ramification constraint, we write the rule $True \Rightarrow \phi$; if we wish ϕ to function as a qualification constraint we instead write the rule $\neg\phi \Rightarrow False$.

In Framework 3 all formula constraints function as ramification constraints. The correctness of our encoding of ramification constraints is demonstrated by the following proposition.

Proposition 3 *Let B be a set of formula constraints. Let C be the set of inference rules $\{True \Rightarrow \phi : \phi \in B\}$. For every interpretation S and explicit effect E, $Res^3_B(E, S) = Res^4_C(E, S)$.*

This proposition also shows that Framework 3 is subsumed by Framework 4.

As an example of a domain in which a state constraint is intended to function as a qualification constraint, we consider a simplified version of a domain from [Lin and Reiter, 1994]. Imagine an ancient kingdom in which there are two blocks. Either block may be painted yellow, but by order of the emperor at most one of the blocks is permitted to be yellow at a time. Consider a state in which the second block is yellow. Intuitively, in this state it is not possible to paint the first block yellow. Representing the emperor's decree by a ramification constraint does not conform to this intuition. Let $S = \{\neg Y_1, Y_2\}$, $E = \{Y_1\}$, and $C = \{True \Rightarrow \neg(Y_1 \wedge Y_2)\}$. Then $Res^4_C(E, S) = \{\{Y_1, \neg Y_2\}\}$. So painting the first block yellow changes the color of the second block! On the other hand, if we represent the emperor's decree as a qualification constraint by redefining C as $\{(Y_1 \wedge Y_2) \Rightarrow False\}$, then $Res^4_C(E, S)$ is empty. This conforms to our intuition that it is impossible to paint the first block yellow in state S.

The following proposition shows that rules of the form we write for qualification constraints cannot lead to ramifications; they can only rule them out.

Proposition 4 *Let C be a set of inference rules and ϕ be a formula. Let $C' = C \cup \{\neg\phi \Rightarrow False\}$. For all interpretations S and S', and any explicit effect E, $S' \in Res^4_{C'}(E, S)$ iff $S' \in Res^4_C(E, S)$ and $S' \models \phi$.*

4 The Logic of S-Conditionals

Representing "causal laws" by inference rules is convenient for the purpose of exposition, but it is not ultimately satisfactory, for two reasons. First, an inference rule is not an object language sentence, and therefore, unlike, for example, a formula constraint, is not interpreted declaratively. As a result, whereas Definitions 1–3 of Section 2 may be recast in semantic terms by simply replacing the derivability relation $\vdash$ by the consequence relation $\models$, this is not true of Definition 4. Secondly, although a causal law intuitively implies its corresponding material conditional (indeed, it implies the corresponding strict conditional, which says that the material conditional holds in every possible state of the world), an inference rule does not. In this section, we prepare to remedy these deficiencies in the representation of "causal laws" by defining a new conditional logic $\mathcal{C}_{\text{flat}}$, which is an extension of S5 modal logic.

The language $\mathcal{C}_{\text{flat}}$ is obtained by adding the following clause to the usual inductive definition of formulas of modal logic: if ϕ and ψ are formulas of propositional logic then

$$(\phi \Rightarrow \psi) \tag{1}$$

is a formula (of $\mathcal{C}_{\text{flat}}$).[3] The formula (1) is called an *s-conditional*. Intuitively, (1) may be read as: the truth of ϕ determines the truth of ψ.[4] Note that s-conditionals have the same syntactic form that we have used for inference rules. Informally, $\phi \Rightarrow \psi$ is true just in case in every part of every possible state of the world in which ϕ is true, ψ is true as well.

An S5 structure can be defined as a pair (Σ, S), where Σ is a nonempty set of interpretations, and S is a distinguished interpretation in Σ. A $\mathcal{C}_{\text{flat}}$ structure is obtained by replacing the set Σ by a set Ω of sets of interpretations.

A structure for a specific $\mathcal{C}_{\text{flat}}$ language $\mathcal{L}$ is a pair (Ω, S), where Ω is a nonempty set of nonempty sets of interpretations of $\mathcal{L}$, and S is an interpretation of $\mathcal{L}$ such that $\{S\} \in \Omega$. The elements of Ω are called *partial states*. By $\mathcal{S}(\Omega)$ we designate the set of *states*, defined as: $\mathcal{S}(\Omega) = \{S : \{S\} \in \Omega\}$. (The set $\mathcal{S}(\Omega)$ corresponds to the set Σ in an S5 structure.) We impose the following structure condition: for every partial state $U \in \Omega$, there is a state $S \in \mathcal{S}(\Omega)$ such that $S \in U$. This reflects the natural requirement that every partial state be a part of some state.

For any set U of interpretations and any propositional formula ϕ, we write $U \models \phi$ as an abbreviation for the expression: for all $S \in U$, $S \models \phi$.

We define when a structure (Ω, S) *satisfies* a formula ϕ (in symbols, $(\Omega, S) \models \phi$) as follows. For all formulas ϕ and ψ (except in the last clause below),

$$
\begin{aligned}
&(\Omega, S) \models \phi \text{ iff } \phi \in S, \quad \text{if } \phi \text{ is an atom},\\
&(\Omega, S) \models \neg\phi \text{ iff } (\Omega, S) \not\models \phi,\\
&(\Omega, S) \models \phi \wedge \psi \text{ iff } (\Omega, S) \models \phi \text{ and } (\Omega, S) \models \psi,\\
&(\Omega, S) \models \Box\phi \text{ iff for all } S \in \mathcal{S}(\Omega),\ (\Omega, S) \models \phi,\\
&(\Omega, S) \models \phi \Rightarrow \psi \text{ iff}\\
&\qquad\qquad \text{for all } U \in \Omega, \text{ if } U \models \phi \text{ then } U \models \psi .
\end{aligned}
$$

In the last clause ϕ and ψ are formulas of propositional logic.

Let T be a set of formulas. A *model* of T is a structure that satisfies every formula in T. We say that T *entails* a formula ϕ (in symbols, $T \models \phi$) if every model of T is a model of ϕ.

It is easy to see that an s-conditional entails its corresponding strict conditional, that is,

$$\phi \Rightarrow \psi \models \Box(\phi \supset \psi). \tag{2}$$

It is also easy to see that an s-conditional does not necessarily entail its contrapositive. For example, let $S_1 = \{a, b\}$, $S_2 = \{a, \neg b\}$, $S_3 = \{\neg a, \neg b\}$, and $M = (\{\{S_1\}, \{S_3\}, \{S_2, S_3\}\}, S_1)$. Observe that $M \models a \Rightarrow b$, but $M \not\models \neg b \Rightarrow \neg a$. These are two important properties of "causal laws".

5 "Causes" versus "Implies"

In this section, we recast Framework 4 semantically, representing causal laws as s-conditionals, instead of as inference rules. Doing so will allow us to clarify the sense in which our theory of ramifications and qualifications is a "causal" theory.

Given a set C of s-conditionals, a model (Ω, S) of C is called *maximal* if there is no model (Ω', S') of C such that Ω is a proper subset of Ω'. It is clear that the maximal models of C can differ only in their second component.

We recast Definition 4 in semantic terms, using the language $\mathcal{C}_{\text{flat}}$, as follows.

Definition 4′. Let C be a set of s-conditionals, and let $M = (\Omega, S'')$ be a maximal model of C. For all states $S, S' \in \mathcal{S}(\Omega)$ and any explicit effect E, $S' \in Res^{4'}_M(E, S)$ if

$$S' = \left\{ L : \begin{array}{l} \textit{for all } U \in \Omega, \\ \quad \textit{if } U \models (S \cap S') \cup E \textit{ then } U \models L \end{array} \right\}.$$

The following proposition shows that Definitions 4 and 4′ agree where both are defined.

Proposition 5 *Let C be a set of expressions of the form $\phi \Rightarrow \psi$, where ϕ and ψ are formulas of propositional logic. Let $M = (\Omega, S')$ be a maximal model of C. For all states $S \in \mathcal{S}(\Omega)$ and any explicit effect E, $Res^4_C(E, S) = Res^{4'}_M(E, S)$.*

Given Propositions 3 and 4, it follows from Proposition 5 that s-conditionals of the forms $\textit{True} \Rightarrow \phi$ and $\neg\phi \Rightarrow \textit{False}$ function as ramification and qualification constraints, respectively. It is interesting to note that the first of these says that ϕ holds in every partial state, whereas the second says that $\neg\phi$ holds in no partial state. Because partial states have truth value gaps, these meanings are distinct.

In order to explain what is distinctive about our theory of ramifications and qualifications, it will be convenient to assume, for the time being, that $(S \cap$

[3] Requiring ϕ and ψ to be formulas of propositional logic is not essential. In fact, the language $\mathcal{C}_{\text{flat}}$ is a simplified version of a more general language $\mathcal{C}$, in which ϕ and ψ are permitted to be arbitrary formulas, possibly containing the modal operators $\Box$ and $\Diamond$, or $\Rightarrow$. The advantage of $\mathcal{C}_{\text{flat}}$ is that it is possible to give its semantics using a simpler kind of model structure than is required for $\mathcal{C}$.

[4] An alternative reading is: ϕ's being true is a sufficient condition for ψ's being true. This reading is our motivation for the name "s-conditional".

S') $\cup E$ is finite. Furthermore, in order to simplify notation, whenever $(S \cap S') \cup E$ appears in a formula we will take it to stand for the (finite) conjunction of its elements. Using this convention, we can say that $S' \in Res^{4'}_M(E, S)$ if and only if

$$S' = \{L : M \models (S \cap S') \cup E \Rightarrow L\}. \quad (3)$$

Given an $S' \in Res^{4'}_M(E, S)$, we will say that a literal $L \in S'$ is a ramification if $(S \cap S') \cup E \not\models L$. Assume that L is such a ramfication. It is interesting to contrast the condition

$$M \models (S \cap S') \cup E \Rightarrow L$$

from Equation (3), which roughly speaking says that L is "caused", with a similar condition

$$M \models \Box ((S \cap S') \cup E \supset L)$$

which says that L is "implied", in the sense that in every possible state in which $(S \cap S') \cup E$ is true, L is true. Since, by (2), $(S \cap S') \cup E \Rightarrow L$ entails $\Box ((S \cap S') \cup E \supset L)$, it is clear that the former condition is stronger than the latter.

If we wished to require merely that ramifications be implied, rather than caused, we would instead impose the fixpoint condition

$$S' = \{L : M \models \Box ((S \cap S') \cup E \supset L)\}. \quad (4)$$

This condition corresponds to Framework 3. To see this, let B be a set of formula constraints, $\Box B = \{\Box \phi : \phi \in B\}$, and M be a maximal $\mathcal{C}_{\text{flat}}$ model of $\Box B$.[5] It can be shown that

$$(S \cap S') \cup E \cup B \vdash L \textit{ iff } M \models \Box ((S \cap S') \cup E \supset L).$$

It follows that S' is a fixpoint of Equation (4) if and only if S' belongs to $Res^3_B(E, S)$.

Let ϕ be a formula of propositional logic. It is interesting to observe that in $\mathcal{C}_{\text{flat}}$ the state constraint $\Box \phi$ is logically equivalent to the s-conditional $\neg \phi \Rightarrow \textit{False}$, which has the form of a qualification constraint. This may seem puzzling, since qualification constraints do not lead to ramifications, while state constraints traditionally have been used to do precisely that. The puzzle is resolved, however, by recalling that in the causal theory of Definition 4′ a requirement stronger than the usual one is placed on ramifications: namely, the requirement that ramifications be "caused" (as in Equation 3) and not merely "implied" (as in Equation 4).

Acknowledgements

The authors would like to thank Vladimir Lifschitz for many useful discussions on the subject of this paper. We are also grateful to Nicholas Asher, Chitta Baral, Enrico Giunchiglia, G. N. Kartha, and Rob Koons for their comments.

[5]The elements of $\Box B$ are state constraints; each of them says of the corresponding formula in B that it is true in every possible state of the world.

References

[Baker, 1991] Andrew Baker. Nonmonotonic reasoning in the framework of situation calculus. *Artificial Intelligence*, 49:5–23, 1991.

[Brewka and Hertzberg, 1993] Gerhard Brewka and Joachim Hertzberg. How to do things with worlds: On formalizing actions and plans. *Journal of Logic and Computation*, 3(5), 1993.

[Gelfond and Lifschitz, 1993] Michael Gelfond and Vladimir Lifschitz. Representing action and change by logic programs. *Journal of Logic Programming*, 17:301–322, 1993.

[Ginsberg and Smith, 1988] M.L. Ginsberg and D.E. Smith. Reasoning about actions II: The qualification problem. *Artificial Intelligence*, 35:311–342, 1988.

[Kartha and Lifschitz, 1994] G. Neelakantan Kartha and Vladimir Lifschitz. Actions with indirect effects. In *Proc. of the Fourth Int'l Conf. on Principles of Knowledge Representation and Reasoning*, pages 341–350, 1994.

[Kim, 1974] Jaegwon Kim. Noncausal connections. *Noûs*, 8:41–52, 1974.

[Lifschitz, 1990] Vladimir Lifschitz. Frames in the space of situations. *Artificial Intelligence*, 46:365–376, 1990.

[Lin and Reiter, 1994] Fangzhen Lin and Raymond Reiter. State constraints revisited. *Journal of Logic and Computation, Special Issue on Actions and Processes*, 4(5):655–678, 1994.

[Marek and Truszczyński, 1994] W. Marek and M. Truszczyński. Revision specifications by means of revision programs. In *Logics in AI. Proceedings of JELIA '94*. Lecture Notes in Artificial Intelligence. Springer-Verlag, 1994.

[Przymusinski and Turner, 1994] Teodor Przymusinski and Hudson Turner. Update by means of inference rules. Submitted for publication, 1994.

[Reiter, 1991] Raymond Reiter. The frame problem in the situation calculus: a simple solution (sometimes) and a completeness result for goal regression. In Vladimir Lifschitz, editor, *Artificial Intelligence and Mathematical Theory of Computation: Papers in Honor of John McCarthy*, pages 359–380. Academic Press, 1991.

[Winslett, 1988] Marianne Winslett. Reasoning about action using a possible models approach. In *Proc. AAAI-88*, pages 89–93, 1988.