Underlying Semantics for the Assessment of Reiter's Solution to the Frame Problem

Tania Bedrax-Weiss[1] and Leopoldo E. Bertossi[2]

[1] CIRL and University of Oregon, 1269 University of Oregon, Eugene OR 97403-1269
[2] Departamento de Ciencia de la Computacion, Escuela de Ingenieria, Pontificia Unviersidad Catolica de Chile, Casilla 306, Correo 22, Santiago, Chile

Abstract. This paper presents an application of Sandewall's methodology [16] for assessing nonmonotonic entailment criteria for reasoning about actions and change. We establish the correctness of Reiter's general solution to the frame problem [14] for a broad and well characterized class of problems. This is done by: (1) identifying a nonmonotonic entailment criterion corresponding to Reiter's solution; (2) discovering ontological and epistemological assumptions that we believe underly the domains of possible applicability of Reiter's approach; (3) defining an underlying semantics corresponding to these assumptions; and (4) providing a semantic correspondence between the models selected by the entailment criterion and the models obtained from the underlying semantics.
We also compare our methodology to Kartha's evaluation [6] of Reiter's general solution with respect to action logic semantics [4].

1 Introduction

This paper presents an application of Sandewall's methodology [16] for assessing nonmonotonic entailment criteria for reasoning about actions and change. We establish the correctness of Reiter's general solution to the frame problem based on the situation calculus [14] for a broad and well characterized class of problems. Given a preliminary axiomatization describing the effects of actions, Reiter's solution is a syntactic completion of this axiomatization. A summary of this process is presented in Section 2. Sandewall's methodology is briefly presented in Section 3. In Section 4 we assess Reiter's solution according to Sandewall's methodology. In Section 5, we compare our methodology to Kartha's evaluation [6] of Reiter's general solution with respect to action logic semantics [4]. Finally, we present the conclusions and future work.

2 Reiter's General Solution

Reiter [14] proposes a formalism to reason about action in first-order logic. Building on work by Haas [5], Pednault [9], and Schubert [17], he syntactically transforms a preliminary axiomatization into a specification that completely describes each fluent at states that result from action executions. The preliminary specification is given in the situation calculus [8] with predicates, fluents, actions, and a function symbol, *do*. The function symbol *do* takes as argument an action and a situation and denotes the situation that results from executing the action in that situation. A distinguished predicate, *Poss*, is introduced to this language to specify the conditions under which actions may be possible to execute. If these conditions are satisfied in the state the action is to be executed, it will be referred to as a legal action. It will be referred to as an illegal action, otherwise.

2.1 Axiomatization

The preliminary axiomatization consists of knowledge about S_0, the initial state of the world, which need not be complete, state independent knowledge which defines the initial database, plus the following axioms:
Action Precondition Axioms: For each action name A we will have a precondition axiom of the form

$$Poss(A(\bar{x}), s) \equiv \pi_A(\bar{x}, s).$$

Positive Effects Axioms: For all pairs formed by a fluent R and an action name A, where the action causes fluent R to become true, an axiom of the form:

$$Poss(A(\bar{x}), s) \wedge \varphi(\bar{x}, \bar{y}, s) \supset R(\bar{y}, do(A(\bar{x}), s)). \tag{1}$$

That is, if action A is possible, and the preconditions for fluent R represented by the metaformula $\varphi(\bar{x}, \bar{y}, s)$ are true at state s, then fluent R becomes true in the successor state $do(A(\bar{x}), s)$ obtained after executing action A in state s. Here, $\bar{x}$ and $\bar{y}$ are parameters of the action and the fluent, respectively.
Negative Effects Axioms: For all pairs formed by a fluent R and an action name A, where the action causes fluent R to become false, an axiom of the form:

$$Poss(A(\bar{x}), s) \wedge \psi(\bar{x}, \bar{y}, s) \supset \neg R(\bar{y}, do(A(\bar{x}), s)). \tag{2}$$

The syntactic completion process is the following. For each fluent we combine the positive effect axioms that contain that fluent into a general single positive effect axiom.

$$Poss(a, s) \wedge \chi(a, s) \supset R(do(a, s)). \tag{3}$$

In this axiom, a is a variable for actions that is considered (tacitly) universally quantified. Similarly, we obtain a general negative effect axiom for each fluent of the form:

$$Poss(a, s) \wedge \phi(a, s) \supset \neg R(do(a, s)). \tag{4}$$

The general effect axioms do not solve the frame problem. Reiter relies on a "Generalized Completeness Assumption" to do so. He makes the common sense assumption that relatively few actions change the value of any particular fluent and that the actions that do so are completely known. This assumption, though, is made at a metalevel so the transformation remains first-order. After this assumption is made, we can construct the:
Successor State Axiom:

$$Poss(a, s) \supset [R(do(a, s)) \equiv [\chi(a, s) \vee (R(s) \wedge \neg\phi(a, s))]]. \tag{5}$$

This axiom is tacitly universally quantified over actions. Intuitively, if action a is possible, then fluent R is true at the successor state that results from the execution of action a if and only if a is one of the actions causing R to be true (for which the fluent preconditions are true), or R was already true when a was executed and a is not one of the actions causing R to be false. There is one successor state axiom for each fluent. In order for things to work properly, unique names axioms [11] are required for actions and states.

It is important to point out that Reiter's solution to the frame problem [14], works only for deterministic actions, that is, actions whose effects are completely known and where the preconditions of the actions are valid in the state they are executed in. Nonetheless, there is nothing in the logic to prevent the execution of illegal actions (actions whose preconditions are not satisfied in the state they are executed), so we must consider them but we choose to ignore its effects.[3]

2.2 Reiter's Nonmonotonic Entailment Criterion

Reiter's approach is monotonic only after the described syntactic transformation has been performed. This method is reminiscent of Clark's completion of logic programs [3]. From this point of view, Reiter's solution represents a syntactic method of selection of the intended models.

Assume that we have a set of formulas Σ corresponding to a preliminary axiomatization of a domain as described at the beginning of Section 2.1. Let $\mathcal{R}(\Sigma)$ denote Reiter's syntactic completion of Σ, as described above. A particular entailment criterion, $\models\!\!\approx_R$, is associated with this completion process with a natural definition: for a formula φ in the language of Σ,

$$\Sigma \models\!\!\approx_R \varphi \quad \Leftrightarrow \quad \mathcal{R}(\Sigma) \models \varphi. \tag{6}$$

On the right-hand side of the equivalence we have the classical entailment criterion, i.e. defined in terms of a "surface semantics" for a first-order predicate calculus. This surface semantics is essentially given in terms of models for first-order languages that represent worlds that evolve in time. Thus, they consider all the possible histories of the world. The criterion $\models\!\!\approx_R$ on the left-hand side is nonmonotonic and can be interpreted as a model preference criterion that selects some of the models of Σ in the class determined by the surface semantics in order to check the validity of φ.

The preferred models of Σ can be inductively defined on actions as follows. In the base case (no actions are executed), select all first-order models which satisfy the initial domain description. In the inductive step we make use of the successor state axiom. There can be two cases: an action may be legal or illegal in that state. If a legal action is executed, select all first-order models such that for each pair (fluent, action), if the successor state axiom for the fluent mentions the action, the value of the fluent in the selected model is true if and only if in equation (5) a is in χ or if it was true and a does not belong to ϕ, and it

[3] In a real application one actually restricts to the case of accessible states, that is, states that can be reached after a finite sequence of legal actions.

is negative otherwise. It can be checked that a fluent will be assigned only one value [14], so at least one model is always selected.

If the action is illegal (its preconditions are not satisfied in the state the action is executed), nothing can be derived. We interpret this as a selection of all possible models of the "successor situation". As we have said before, there is no way to exclude this case from consideration since there is nothing in the logic that restricts the execution to legal actions only.[4] If we introduce a predicate of executability that restricts actions to this case, the definition of the selected models restricted to this predicate remains the same, but without illegal actions.

Remark. The models of Σ we have constructed are exactly those of $\mathcal{R}(\Sigma)$.

The class of models selected by Reiter's entailment criterion will be compared to those defined according to Sandewall's methodology in the following sections.

3 Sandewall's Methodology

Sandewall's methodology [16] consists of first, identifying a class of problems (or domains) one wishes to work with characterized by different epistemological and ontological assumptions about the domains. Epistemological assumptions are assumptions about what kind of information about the description of the domain is available such as, whether there is complete or incomplete information about the initial state, whether there is information about states other than the initial one, etc. There are also ontological assumptions, assumptions about the kinds of worlds that are allowed to exist in the domain, such as, whether there are concurrent actions or not, whether there are delayed effects, etc.

Secondly, Sandewall defines an underlying semantics for the identified class of problems. The underlying semantics takes into account the assumptions underlying the class. The semantics is basically, a simulation semantics where the starting point is a scenario description or a *chronicle* (a description of a representative of the class of problems to be simulated consistent with the identified epistemological assumptions) and where the simulation is generated by an interaction between an ego and a world. This interaction intuitively represents an interaction between a single agent (the ego) who perceives the reality of the problem simulation and invokes actions, and the world, who changes (or not) in response to the ego's actions. Thus, the world "knows" of the action descriptions and the ontological assumptions. We describe the relevant assumptions and the simulation in the following sections. This simulation yields the intended models for the given scenario description.

4 Analysis of Reiter's Solution

In order to compare the intended models to those selected by Reiter's formalism, we must first obtain a translation of Reiter's preliminary axiomatization

[4] Notice that *do* is a total function.

into a scenario description in Sandewall's logic. This translation will define a structure called a chronicle which contains several kinds of axioms: observations (corresponding to the description of the initial domain description), action laws (corresponding to a translation of the effect axioms), and a schedule (corresponding to a sequence of actions that will define the state about which we want to query). We give a brief description of this translation later on.

Secondly, the intended models are obtained from this chronicle according to the underlying semantics. Finally, we state the semantic correspondence between the models selected by Reiter's formalism and Sandewall's underlying semantics.

4.1 Ontological and Epistemological Assumptions

By taking a philosophical approach to the situation calculus, to Reiter's solution, and to the problems with inertia (worlds resisting change), we discover the ontological and epistemological assumptions which characterize the class of problems for which Reiter's solution is intended to be applicable. Epistemological assumptions are those that refer to the description of the domain and ontological assumptions are those that refer to the properties of the domain, as in [16].

Ontological Assumptions From the situation calculus we can recognize two assumptions: *single step actions* executed from one situation to the next and *sequential actions* since no concurrent actions are allowed. More assumptions are identified in Reiter's framework: *inertia* restricted to legal actions only since the successor state is undetermined for illegal actions, *deterministic actions*, *alternative results of actions*, i.e., state-dependent actions, *structural dependency*, i.e., definitions of fluents in terms of properties of a different object, and *local dependency*, i.e., restrictions on the values a fluent may take that are imposed by features of the same object.

Epistemological Assumptions We identified the following assumptions: *incomplete knowledge about the initial state*, *each change is completely characterized* in the sense that all actions that produce a change are known, *the axiomatization is consistent*, and *actions are defined in terms of preconditions*, so the effects of an action are completely known only if it is a legal action.

4.2 Translation into Sandewall's Logic

Sandewall defines an Elementary Feature Logic which is basically a first-order temporal logic [16]. This logic is a language to describe a chronicle or a particular scenario of a domain. The entailment criteria are defined separately by giving its intended set of models. In order to assess Reiter's solution must translate the preliminary axiomatization into a chronicle in Sandewall's logic and define the set of intended models for his entailment criterion.

The translation is straightforward so we will only briefly sketch how this is done. We will interpret each situation as corresponding to a timepoint[5], and such that the initial situation corresponds to timepoint 0, and situation s_i corresponds

[5] We follow Sandewall's logic, but the analysis could have been done with situations.

to timepoint i. For each statement of the form $R(x, S_0)$ we introduce an observation statement of the form $[0]R(x) \hat{=} T$. The symbol $\hat{=}$ assigns an observed value (the symbol at the right) to the symbol at the left. For each statement that does not mention any state φ, we introduce an observation statement of the form $[]\varphi$. To translate the effect and precondition axioms into action laws we need to compile for each action, the effect axioms and precondition axioms that mention that action to obtain a first-order equivalent "action law":

Take action $A(\bar{x})$ and all the axioms of the form 1 and 2 that mention the action and write an axiom of the following form, where we have already substituted $Poss(A(\bar{x}), s)$ by its definition.

$$\begin{aligned}[t, t+1]A(\bar{x}) \Rightarrow \quad & [t]\pi_A(\bar{x}) \rightarrow [t]\chi \rightarrow [t+1]R := T \wedge \\ & [t]\pi_A(\bar{x}) \rightarrow [t]\phi \rightarrow [t+1]R := F \wedge \\ & [t]\pi_A(\bar{x}) \rightarrow [t]\epsilon \rightarrow [t+1]Q := T \wedge \ldots\end{aligned}$$

The symbol $:=$ has an implicit meaning of possible change, that is, it means that the fluent will have the value at the right of the symbol regardless of what its value was at time t.

A sequence of actions $do(a_{n-1}, \ldots, do(a_1, do(a_0, S_0)) \ldots)$ will be translated:

$$[0]a_0, [1]a_1, \ldots, [n-1]a_{n-1}.$$

The models of this chronicle are recovered from the class of structures determined by the underlying semantics. We will compare these models to those selected by Reiter's nonmonotonic entailment criterion as defined in Section 2.2. This will be done in the following section.

4.3 Assessment

According to Sandewall's methodology described in Section 3, we identified the class of problems we intend to solve with Reiter's solution. This class was defined according to the ontological and epistemological assumptions. We need to characterize the intended models that correspond to this class by means of the underlying semantics. This underlying semantics is defined as a simulation (interaction) between an ego and a world.

Underlying Semantics The interaction between the ego and the world determines all the possible histories of the particular instance of the domain for a sequence of actions. More precisely, given: (a) an arbitrary axiomatization, (b) a sequence of transactions, (c) an ego, and (d) a world (all of these satisfying the ontological and epistemological assumptions), the interaction will determine the possible histories of that chronicle. The intended models will be obtained from these histories, called *developments*.

Formally, a *development* is a tuple $\langle B, M, O_s, R, \mathcal{F}, \mathcal{C}\rangle$, where:

B: is the set of interpretations of the form $\{0, \ldots, k\}$ for breakpoints (which in the case of single-step actions coincide with the timepoints), where k corresponds to the current state,

M: is the valuation for timepoints (assigns members of B to timepoints) and objects in the universal object domain,
O: is the object domain
R: is the history of the fluent changes from timepoint 0 to timepoint k, actually, we consider this to be an indexed set of assignments of values to fluents,
$\mathcal{F}$: is the set of actions that have been executed,
$\mathcal{C}$: is the set of actions that currently are being executed; since only one action can be executed at a time, it will always contain a single action.

In $\mathcal{F}$ actions are time stamped indicating the starting and termination times, while in $\mathcal{C}$ actions are only time stamped with the starting times.

The moves of the ego and the world are recorded in the developments as follows: If the ego chooses an action, a specified in the chronicle, the world updates $\mathcal{C}$ with the tuple $\langle t, a\rangle$ and reacts to the action in two possible ways: (1) If the preconditions of a are satisfied in R, it updates that history by overriding any previous fluent values with new ones as determined by the action laws in the chronicle. It also updates B by adding a new timepoint $t+1$ and removes $\langle t, \tau\rangle$ from $\mathcal{C}$ and adds $\langle t, \tau, t+1\rangle$ to $\mathcal{F}$. (2) If the preconditions are not satisfied in R, it branches out (one change each time it branches) generating all possible updates for R, and it updates B, $\mathcal{C}$, and $\mathcal{F}$ as before.

The interaction goes on repeatedly for all actions defined in the chronicle and for all histories that arise up to a time t. This interaction is defined for every initial state of the particular domain and every valuation of the timepoints that satisfies the time restrictions imposed by the chronicle (sequential single-step transactions). This generates all the possible histories of the world, given all the possible executions of actions in all possible initial states.

The developments are tree structures. Since the initial state of the domain description is not completely specified, we will have several tree structures corresponding to the evolution of the world, namely, the trees whose history component of the initial development is consistent with the given initial state. The models that define the underlying semantics are obtained from the developments. They are first-order logic models of the history slices of the developments at all timepoints.

The underlying semantics defined in this way is independent of Reiter's entailment criterion, since it depends only on the assumptions made about the domain in question. It is also true that the underlying semantics completely captures these assumptions [2].

Semantic Correspondence In this section we state the correspondence between the first-order models of the values assigned to fluents in both frameworks: Reiter's and Sandewall's. This result applies in the legal as well as in the illegal case.

Let us consider a chronicle Γ, arising from a preliminary axiomatization Σ, of a particular domain description in Reiter's formalism. This chronicle contains a sequence of actions A. The underlying semantics provides the set of models for Γ according to the execution of A.

Theorem 1. *The class of first-order models of Γ arising from the underlying semantics at each time slice coincides with the set of first-order models of $\mathcal{R}(\Sigma)$.*[6]

The proof of the theorem is done by induction on the executed actions. The complete proof can be found on [2].

Note that the models of the chronicle according to both frameworks are formally different since Reiter's models contain information about the fluents at each situation, while Sandewall's models contain the history of the fluents at each point in time. This is why we are forced to compare only the first-order models of the values of the fluents at corresponding timepoints.[7]

This theorem can be thought of as stating that Reiter's solution is applicable to all domains that belong to the class defined according to the ontological and epistemological assumptions of Section 4.1.

¿From the theorem and remark 2.2 we can obtain a soundness and completeness result:

Corollary 2. *For any sentence φ, $\Sigma \approx_R \varphi$ iff φ is true in all models of Σ arising from the underlying semantics.*

5 Comparison to Kartha's Evaluation

Kartha [6] has already presented an analysis of Reiter's solution with respect to Gelfond and Lifschitz's action logic $\mathcal{A}$ [4]. He proves that Reiter's solution captures the semantics of $\mathcal{A}$ starting from a description (chronicle) in $\mathcal{A}$ and then translating it into Reiter's formalism. This translation is sound and complete.

Although our work is similar to Kartha's in the sense that we both characterize and compare the models for a given scenario description, there are several differences. In $\mathcal{A}$ the semantics of a description is fixed, i.e., the entailment criterion is the same for all descriptions, forcing the translation to be from $\mathcal{A}$ into Reiter's formalism. Sandewall's logic, on the other hand, is only a base logic, that is, the entailment criterion is independent of the language used to describe the chronicle. Because of this, we can assess different nonmonotonic entailment criteria by defining equivalent selection criteria in Sandewall's framework. Sandewall's methodology provides a certain degree of flexibility that is not present in Kartha's approach.

There is another difference with respect to both methodologies. Sandewall gives a general language and a general list of assumptions which he restricts when he identifies exactly the class of problems to be addressed. Any assessment based on $\mathcal{A}$, on the other hand, must extend the language and semantics (if necessary). So the difference is one of narrowing or extending the language.

[6] See Remark 2.2.

[7] We are indebted to Matt Ginsberg for pointing this out to us.

Another difference is that Sandewall analyses a criterion with respect to the ontological and epistemological assumptions that underlie a particular domain while Kartha's evaluation considers the models given by two different entailment criteria. In this sense, Sandewall's methodology may be said to check the models selected by the entailment criterion against "reality", whereas Kartha is forced to compare models, that is, abstractions of reality.

Our work is a little bit less restrictive than Kartha's work since we restrict the domains that can be considered to those that can be expressed in Reiter's framework, while Kartha restricts to those expressible in the language $\mathcal{A}$. The difference appears to be that the language $\mathcal{A}$ considers named actions and fluents with no quantification over any of these, so it can be seen as a propositional case. Reiter's framework is full first-order logic.

6 Conclusions

We have made several contributions: (1) we identified a nonmonotonic entailment criterion corresponding to Reiter's syntactic compilation and we inductively defined the preferred models; (2) we identified the ontological and epistemological assumptions that underlie the situation calculus and Reiter's solution in general; (3) we defined an underlying semantics that corresponds to the class of assumptions, thereby extending Sandewall's work; (4) we stated a semantic correspondence between Reiter's models and the models of the underlying semantics; and (5) as a consequence, we obtained that Reiter's solution is correct and complete for a broad class of problems.

7 Future Work

There are still several issues that need to be addressed. We have only considered Reiter's original solution, not its extension to some kinds of ramifications [7].

In [15, 10], Reiter formalizes the evolution of a database according to his solution to the frame problem [14]. In [1], this formalization was positively assessed with Sandewall's methodology. In that paper, there was a restriction to ground databases, i.e., without null values. It seems interesting to extend their results to null values by making a logical reconstruction of the database as described by Reiter [12, 13]. Other extensions to databases may also be considered, such as, concurrent actions and active databases.

8 Acknowledgments

We are grateful to Ray Reiter, Erik Sandewall, and Javier Pinto for stimulating conversations and advice. This research has been supported partly by DIUC, SAREC/CONICYT, Fundacion DICTUC, and FONDECYT (#1930554) grants, partly by the Air Force Office of Scientific Research under contract 92-0693 and by ARPA/Rome Labs under contracts F30602-91-C-0036 and F30602-93-C-00031. Tania Bedrax-Weiss is specially thankful to CIRL and to David Etherington for patient reviews of this work. Leopoldo Bertossi visited Linköping University in Sweden for a couple of weeks and is grateful to Erik Sandewall for his hospitality.

References

1. Bedrax-Weiss, T., Bertossi, L.: An assessment of Reiter's general monotonic solution to the frame problem and its applicability to database updates. ECAI-LAC (1994)
2. Bedrax-Weiss, T., Bertossi, L.: An assessment of Reiter's general monotonic solution to the frame problem and its applicability to database updates. Tech. Rept. P. Universidad Catolica de Chile, Dept. de Ciencia de la Computacion, Santiago, Chile, RP-PUC-DCC-94-1 (1994) 1–25
3. Clark, K.L.: Negation as failure. Logic and Databases, Plenum Press, New York (1978) 292–392
4. Gelfond, M. and Lifschitz, V.: Describing action and change by logic programs. Joint International Conference and Symposium on Logic Programming (1992)
5. Haas, A.: The case for domain-specific frame axioms. Workshop: The frame problem in artificial intelligence. (1987) 343–348
6. Kartha, G.N.: Soundness and completeness theorems for three formalizations of action. IJCAI-93, Chambery, France (1993) 724–729
7. Lin, F., Reiter, R.: State constraints revisited. Journal of Logic and Computation, Special issue on actions and processes **4** (1994) 655-678
8. McCarthy, J., Hayes, P.: Some philosophical problems from the standpoint of artificial intelligence. Machine Intelligence **4** (1969) 463–502
9. Pednault, E.: ADL: Exploring the middle ground between STRIPS and the situation calculus. KR-89, (1989) 324–332
10. Reiter, R.: Formalizing database evolution in the situation calculus. Fifth Generation Computer Systems, Tokyo, Japan (1992)
11. Reiter, R.: Equality and domain closure in first-order databases. JACM **27** (1980) 235–249
12. Reiter, R.: Towards a logical reconstruction of relational databases theory. On Conceptual Modelling: Perspectives from Artificial Intelligence, Databases and Programming Languages, Springer-Verlag (1984) 191–233
13. Reiter, R.: A sound and sometimes complete query evaluation algorithm for relational databases with null values. JACM **2** (1986) 349–370
14. Reiter, R.: The frame problem in the situation calculus: a simple solution (sometimes) and a completeness result for goal regression. AI and Math. Theory of Computation: Papers in Honor of John McCarthy, Academic Press, (1991) 359–380
15. Reiter, R.: On specifying database updates. Tech. Rept. University of Toronto, Dept. of Computer Science, Toronto, Canada, KRR-TR-92-3 (1992)
16. Sandewall, E.: Features and Fluents. The Representation of Knowledge about Dynamical Systems **1**. Oxford University Press (1994)
17. Schubert, L.: Monotonic solution of the frame problem in the situation calculus: an efficient method for worlds with fully specified actions. Knowledge Representation and Defeasible Reasoning, Kluwer Academic Press, (1990) 23–67