Electronic News Journal on Reasoning about Actions and Change

Editor: Erik Sandewall

Volume 1, Number 4

Contributions during November, 1997

ISSN, print: 1403 - 2031
on-line: 1403 - 204X

Linköping University Electronic Press
http://www.ep.liu.se/ej/enrac/1997/04/

Publication Trail and Update Rules

The Electronic News Journals (ENJ) are a medium for exchange of scientific information and debate. In particular, they serve as the forum where articles received by the ETAI (Electronic Transactions on Artificial Intelligence) are discussed publicly for review, in separate News Journals for each ETAI area.

The Electronic News Journal on Reasoning about Actions and Change (ENRAC), in particular, uses a publication trail where contributed information is first distributed on a daily basis as *Newsletters* using electronic mail and web pages in HTML. Then, contributions during the same month are compiled into an HTML version of the monthly *News Journal.* Finally, the same information is converted via Latex to a postscript version that is suitable for printing on paper and reading off-line. It is formally published and archived by the Linköping University Electronic Press in both its electronic and its paper form, as described on page (iv).

Offprints of individual segments. Each News Journal issue consists of a sequence of *segments* relating to specific topics. "Offprints" of these can be obtained from the URL:s indicated at the head of the first page of each segment. It is intended to maintain these URL:s and their contents for the foreseeable future. Misprints of minor significance are corrected in these offprints (but not in the E-Press version since it is considered archival).

Corresponding HTML edition with dynamic links. The HTML edition contains considerable numbers of links to other pages and structures on the net: articles that are available on-line, home pages of conferences and of individual researchers, links to other part of the ETAI structure, and so on. Due to the natural limitations of the paper medium, only some of the WWW links have been retained here as footnotes. In order to retrieve articles and other information that are referenced in the present issue, it is recommended to look up the corresponding issue in HTML and to use its link. The HTML issues of the News Journals on Reasoning about Actions and Change can be found at the following URL:

`http://www.ida.liu.se/ext/etai/actions/njl/`

It is intended that the HTML issues will by updated continously to the largest extent possible, for example by replacing URL links to the home pages of authors that have changed to another site.

Table of Contents

Dates of Publication

Since the date of publication may be understood either as the date of first public appearance, or as the day of reproduction on paper in many copies, and since both of these definitions may be difficult to apply in the case of electronic publication, we make the following clarifying statement.

The contents of the present issue were put on-line in their original, HTML version during the month of November, 1997. Then the contents were edited and formatted, resulting in the present, formatted version which was published on June 10, 1998, in two concurrent editions: an on-line edition and a paper edition. The on-line edition was timestamped electronically and put on-line by Linköping University Electronic Press at the URL specified on page (i). The paper edition was obtained by printing the on-line edition on a standard computer printer. It was reproduced in 200 copies, legally archived, and made available for distribution.

Electronic News Journal on Reasoning about Actions and Change 117
Vol. 1: *117-117*, 30.11.1997
http://www.ida.liu.se/ext/etai/actions/nj/9711/

The Month of November

Summary of News Journal Contents

The major event during this month was that Paolo Liberatore's article "The Complexity of the Language $\mathcal{A}$" was accepted to the ETAI. This was ETAI's first paper acceptance; it followed after due ETAI procedure that includes both an on-line open discussion (*reviewing*, in ETAI terminology) and a subsequent *refereeing* decision by confidential referees. The first ETAI issue (Vol. 1, Nr. 1-3) contains both Liberatore's article and an editorial note explaining the reviewing procedure in terms of this first case.

The month obtained an even mix of activities in the ENRAC Colloquium. There were contributions to the review debates about two other ETAI submitted articles in addition to Liberatore's. The two "panel" discussions received a number of contributions, many of which centered around the usefulness and relevance of action description languages such as $\mathcal{A}$ and $\mathcal{E}$, as compared to the direct use of first-order logic.

November, 1997 can therefore be identified as the month when both the ETAI publishing scheme and the ENRAC's role in relation to the ETAI became concretely operational: all major aspects of the scheme can now be demonstrated in terms of *how they work*, and not merely in terms of how we intend them to work.

The relationship between our research area and its applications, actual and potential ones, had been mentioned in a letter by Austin Tate in the September, 1997 issue. The same question was again brought up in the context of a CFP for a journal special issue - see page 156 in the present issue.

Some minor changes have been made to the graphical appearance of the latex/ postscript version of the News Journal for November, compared to the previous months. The major difference is that we go to a smaller font size.

Electronic News Journal on Reasoning about Actions and Change 118
Vol. 1: *118-121*, 30.11.1997
http://www.ida.liu.se/ext/etai/received/actions/001/

Protocol of on-line discussion during November, 1997
about the following research article:

Paolo Liberatore:

The Complexity of the Language $\mathcal{A}$

Q5. **Anonymous reviewer** (21.11)

Section 6 in the article addresses domain descriptions in which some states are not reachable from the initial state. I think the modification of the semantics of $\mathcal{A}$ as proposed in that section does not allow to solve the problem of unreachable states. Consider the situation where you replace the observation that H and L are initially both false by what you observe after two steps, i.e. replace the two 'initially' propositions by

H after $I; I$
$\neg L$ after $I; I$

Intuitively, the model should be the same. But the reachable states semantics now considers $\{H, L\}$ to be a possible initial state. This is because $\Psi_D(I, \{H, L\})$ is undefined there, hence both value propositions are weakly true. I.o.w., you get back to the orginal $\mathcal{A}$ semantics you wanted to avoid.

Generally, my feeling is that in order to solve the problem that your example highlights (and I think it is a problem) you cannot do in the language of $\mathcal{A}$. What you need is a notion such as "action A is executable if ...".

Also, on page 5 you write:

> *"to prove that D entails* $\ulcorner F$ after $A_1; ...; A_m\urcorner$ *it suffices to prove that* $\ulcorner D \cup \neg F$ after $A_1; ...; A_m\urcorner$ *is inconsistent."*

This reduction seems not to go through for your modified semantics of section 6.

The rest (i.e. the main part of the paper) is just perfect as far as I can see. And it was a pleasure to read the paper.

A5. **Paolo Liberatore** (24.11)

You wrote:

> *Section 6 in the article addresses domain descriptions in which some states are not reachable from the initial state. I think the modification of the semantics of $\mathcal{A}$ as proposed in that section does not allow to solve the problem of unreachable states. Consider ...*

You are right. In this case, the semantics does not capture the intuitive meaning of the domain description.

You also wrote:

> *Generally, my feeling is that in order to solve the problem that your example highlights (and I think it is a problem) you cannot do in the language of $\mathcal{A}$. What you need is a notion such as "action A is executable if ...".*

The intended semantics for the actions in $\mathcal{A}$ is: all actions are executable in any state. When $\Psi_D(I,\sigma)$ is undefined, the choice of the original semantics of $\mathcal{A}$ is that the whole domain description is inconsistent. What I meant is instead that A is not executable in σ. In this sense, in $\mathcal{A}$ *it is possible* to express propositions like "I is executable if", but only in a non-intuitive way: in the example of the counter I is not executable in $\{H, L\}$. We need an appropriate semantics for such propositions.

(If there is a proposition "I is executable if ..." there is no need to make Ψ defined on I in states in which I is not executable. As a result, there is nothing that prevent us from using the indefiniteness of Ψ to infer the non-executability of actions. The drawback is that this definition is less intuitive.)

Analyzing the problem of unreachable states, my first idea for solving it was: an interpreted structure (σ, Φ) is a model of D if and only if

1. each action is executable in any state reachable from σ
2. each value proposition is satisfied (as in the old semantics of $\mathcal{A}$)

However, I discarded this idea, and instead I defined the one that is presented in the paper. Note that this semantics, although correct wrt your example, suffers from some drawbacks. For example, if for any state there is an action not executable there, then the domain description is inconsistent. Consider for example the case

I is executable if F
J is executable if $\neg F$

This domain is inconsistent, because in the initial state in which F is true the action J is not executable, while in the initial state in which F is false the action I is not executable. In such cases, it seems possible to infer that, if the initial state is $\{F\}$ then the first action executed is I. For instance, if there is also a proposition

initially F

the first executed action is I. The semantics of $\mathcal{A}$ does not infer any statement about the execution of actions (in $\mathcal{E}$ and similar languages there are propositions like "I at 0" or "I happens at 0"). A statement like "F after I" may be interpreted in two ways:

1. if F *were* executed the result would be a state that implies F
2. F is executable in the initial state, and the resulting state implies F

(The choice of the semantics of the paper is the first one, while the discarded one uses the second one.) Statements like "I at 0" or "I happens at 0" means that I is executed in the time point 0 (and thus the action must be executable in that state).

I discarded the first semantics (preferring the one of the paper) because the impossibility of executing an action in the initial state (or any other state) may influence the set of the possible initial states. The semantics of the paper first determines the set of possible initial states, and only then determines which states are reachable from them. However, in the example of the counter, the initial state $\{H, L\}$ must be rejected *because* the action I is not executable from there. This is why the semantics of the paper fails.

Perhaps the discarded solution is better then the chosen one. The problem is left open. A minimal requirement for a semantics that takes into account the reachability of states is that if D does not have "executable" propositions (or, if Ψ_D is total) then the new semantics must coincide with G&L's semantics. The semantics of the paper (and the one of the previous paragraph) has this property (clearly, this is not enough.)

You also wrote:

> *Also, on page 5 you write:*
>
> > "to prove that D entails $\ulcorner F$ after $A_1; ...; A_m\urcorner$ it suffices to prove that $\ulcorner D \cup \neg F$ after $A_1; ...; A_m\urcorner$ is inconsistent."
>
> *This reduction seems not to go through for your modified semantics of section 6.*

Indeed, this property holds only for the classical semantics of $\mathcal{A}$ (this property is used for proving the complexity of the entailment in the classical semantics of $\mathcal{A}$).

Q6. **Thomas Drakengren** (24.11)

Paolo,

Here are some questions and suggestions about details in your article:

page 16, line -14: What about the case A causes G, A causes $\neg G$? That would be inconsistent. What is the intended meaning in this case?

page 17, line 8: This case is similar to the IJCAI '97 paper by myself and Marcus Bjäreland [c-ijcai-97-1447], where we require one negative precondition and one negative postcondition (which is of course equivalent, replacing true and false). You can probably add some nondeterminism here, retaining tractability, if you do not allow a precondition, the same way as we're doing it (you can then have a Horn postcondition).

page 19, line 3: An inconsistent domain description could never entail the same propositions as a consistent one. What is the intended meaning in this case?

References:

[c-ijcai-97-1447] Thomas Drakengren and Marcus Bjäreland. *Reasoning about Action in Polynomial Time.* Proc. International Joint Conference on Artificial Intelligence, 1997, pp. 1447-1453.

A6. **Paolo Liberatore** (1.12)

Dear Thomas,

Thank you for your suggestions. About your observations:

> *page 16, line -14: What about the case A* causes *G, A* causes *$\neg G$? That would be inconsistent. What do you mean in this case?*

In this case Ψ_D is not total. This can be verified in polynomial time (Lemma 1).

> *page 19, line 3: An inconsistent domain description could never entail the same propositions as a consistent one! What do you mean in this case?*

The original domain description implies (under the semantics of reachable states) the same propositions of the modified one (using Gelfond & Lifshitz's semantics). The point is that I am using two entailment relations for the original domain description and modified one.

Paolo

Vol. 1: *122-122*, 30.11.1997
http://www.ida.liu.se/ext/etai/received/actions/002/

Protocol of on-line discussion during November, 1997
about the following research article:

Michael Thielscher

A Theory of Dynamic Diagnosis

Q1. **Rob Miller** (3.11)

Michael, a question regarding the relative likelihoods of different components failing:

It seems to me that, in the absence of specific domain knowledge to the contrary, it's right to prefer explanations of system failure which are minimal in terms of the number of failures of individual components, other things being equal (this is one aspect of how your main example works, if I've understood correctly). But you also mention that good diagnosis is able to "take into account a priori knowledge of differences in the likelihood of components to break" (an example might be the knowledge that relays are more likely to fail than resistors). Could you say a little more about this in relation to your approach? Would you perhaps use more than one "ab" predicate, with some sort of prioritised minimisation between the predicates?

A1. **Michael Thielscher** (3.11)

Rob,

My theory does indeed respect domain knowledge of the kind you mention. This knowledge is formally represented by a partial ordering among the instances of the (unique) *ab* predicate–which is equivalent to your suggestion of allowing a priority hierarchy of different *ab* predicates. The notion of preferred models (Definition 10) reflects a given partial ordering in that more unlikely instances of *ab* are preferably minimized. As a consequence, the axiomatization of the theory by means of the Fluent Calculus (Section 5) employs Brewka's prioritized extension to classical Default Logic.

Electronic News Journal on Reasoning about Actions and Change 123
Vol. 1: *123-130*, 30.11.1997
http://www.ida.liu.se/ext/etai/received/actions/003/

Protocol of on-line discussion during November, 1997
about the following research article:

Antonis Kakas and Rob Miller:

Reasoning about Actions, Narratives and Ramification

C2-1. **Alessandro Provetti** (10.11)

Dear Antonis and Rob,

I'd like to comment on Tom's example about the role of h-statements in $\mathcal{A}$-languages. In the language $\mathcal{L}$ of Baral et al. the theories:

F at $S0$

and

F at $S0$
A occurs-at $S0$

have different models.

Assume that there are other fluents than F, the former has models which differ one to another by the interpretation of the initial state (except -of course- for F) while agreeing on the fact that nothing happened at all.

The latter yields the same models as far as the initial state is concerned, *but* all of them sanction that A has happened. As a result, the latter theory implies the formula A occurs-at $S0$.

It appears to me that the equivalence of the two theories above under $\mathcal{E}$-semantics does not mean in general that $\mathcal{A}$-style semantics cannot account for h-propositions.

You may want to comment on this in the paper or -possibly- proceed to work on the entailment associated to $\mathcal{E}$.

Hope this helps. Ciao!

Alessandro Provetti

C2-2. **Tom Costello** (11.11)

Dear Antonis and Rob, (and Alessandro)

While the languages $\mathcal{L}_0$ and $\mathcal{L}_1$ of Baral et al. give truth conditions for propositions stating happens , precedes and holds , they do not give truth conditions for causes propositions. Like Baral and Gelfond, and Kartha and Lifschitz their models are functions from sequences of actions to (sets of) states. Because of this they conflate domain descriptions that are not conflated by models that are functions from states to sets of states (Res etc.).

Consider the following domain description, stated in $\mathcal{A}$.

A causes F

initially F

or in $\mathcal{L}_0$

A causes F

F at $S0$

These have the same functions from sequences of actions to sets of states as,

A causes F

A causes G if not F

initially F

or in $\mathcal{L}_0$

A causes F

A causes G if not F

F at $S0$

However, if we consider functions from states to sets of states, then these have different models. Thus domain descriptions that were distinguished by $\mathcal{A}$, are conflated by later languages.

These later approaches conflate models that intuitively differ.

I agree with Alessandro that $\mathcal{E}$ type languages can give semantics to h-propositions. My complaint is that current approaches fail to give semantics to all their propositions. As $\mathcal{A}$ and $\mathcal{E}$ type languages do not have a proof theory, save by being translated into other approaches, it seems strange that they do not even have a model theory for all their propositions. In Antonis and Rob's case they lack truth conditions for some of their propositions, and worse, it seems that it is not even possible to define truth conditions. The same problem arises for causal statements in Baral et al., Baral and Gelfond, and Kartha and Lifschitz. Other models of $\mathcal{A}$ type languages do not have this problem of collapsing domain descriptions $\mathcal{A}$ considered distinct, for causal statements, for instance, E. Giunchiglia, N. Kartha and V. Lifschitz, "Representing action: indeterminacy and ramifications". Therefore, I argue that action language models should define truth conditions for all their propositions, and further, should ensure that intuitively different models are distinct.

Tom

C2-3. **The Authors** (12.11)

Hi Tom,

You wrote:

> *In Antonis and Rob's case they lack truth conditions for some of their propositions, and worse, it seems that it is not even possible to define truth conditions.*

As we said in our original answer to your question, it's trivial to extend the semantics of the Language $\mathcal{E}$ to include truth conditions for h- and c-propositions, but superfluous (to the main themes of the present and previous papers). However, for the record, you can do this by defining an interpretation as a tuple $\langle H, J, K\rangle$. H is as before, J is a function

$$\textsf{Actions} \times \textsf{Time-points} \longrightarrow \{\textsf{true}, \textsf{false}\}$$

and K is a function

$$\textsf{Actions} \times \textsf{Fluent-literals} \times 2^{\textsf{Fluent-literals}} \longrightarrow \{\textsf{true}, \textsf{false}\}$$

The definition of a model is exactly as before (Definition 9), with the additional conditions:

- $J(A,T) =$ true iff ⌜A happens-at T⌝ is in D
- $K(A,L,C) =$ true iff either ⌜A initiates L when C⌝ is in D, or $L = -F$ and ⌜A terminates F when C⌝ is in D.

But this doesn't really add much insight; you just get that D entails a given h- or c-proposition iff the proposition is in D. Of course, for other extensions of $\mathcal{E}$ it might become worthwhile complicating the structure of an interpretation in this way. (Similarly for r-propositions.) Again, you might find Van Belleghem, Deneker and Dupre interesting in this respect.

Rob and Tony

C2-4. **Tom Costello** (13.11)

Dear Rob and Tony,

Your definition of truth for c-propositions seems very unintuitive to me. I would think that if A terminates F if G, then A terminates F if G, H.

Your definition does not give this result. The reason I ask for truth conditions for your propositions is that I cannot understand what the intuitive consequences of a set of propositions should be, unless I understand what the propositions say. If the propositions are expressed in a standard logic, then I understand them using the definition of truth in a model. However, your propositions are not in a standard logic, and therefore, to understand what

A terminates F if G

means, I have to know when it is true.

Your paper introduces a new type of proposition,

F whenever $G_1, ..., G_n$

There are some obvious choices for truth conditions for this type of proposition. In particular, it can be understood that every this is obeyed at every time-point, or that this is a property of every "possible" state, not every "actual" state. Without knowing which notion this proposition is trying to express, I cannot understand what the proposition says.

I do not think truth conditions are a side point to the main theme of your paper. As you say, action languages are supposed to be "understandable and intuitive". Languages cannot be understood without semantics.

Yours,

Tom

C2-5. **The Authors** (17.11)

Tom,

We think that perhaps we're in danger of going round in circles in this discussion. As we've said in other answers, we've much sympathy for your stance on the benefits of general purpose logics (and in particular classical logic), and that's why we've stated on numerous occcasions that languages such as $\mathcal{E}$ are perhaps best regarded as intermediate stages in the development of formalisms written in such logics. However, we do feel that they have a use in initially discussing and illustrating approaches to particular issues - in our case, to ramifications - in a relatively intuitive and uncluttered way. But we do recognise that what is intuitive for one person might not be so for another. (In particular, of course, as regards formalising common sense it is possible to supply classical logic axiomatisations which are intuitive to some people but not others).

Again, it would be interesting to get some views from more people who have developed $\mathcal{A}$ style languages on some of the general issues that you've raised (if not here, then perhaps in a more general ENRAC panel discussion on the advantages and disadvantages of specialised action languages).

Rob and Tony

> **Editor's note:** *continued discussion on the merits and demerits of Action Description Languages will be referred to the panel discussion on ontologies.*

C2-6. **The Authors** (28.11)

Tom,

In ENRAC 21.11, in the context of the general discussion on action description languages, you asked:

> *Similarly, does*
>
> $\mathsf{Always}\ F, G$
>
> *or Kakas and Miller's*
>
> $F\ \mathsf{whenever}\ \neg G$
>
> *mean that every actual state satisfies F,G, or every possible state.*

In the light of this remark, it now occurs to us that a possible partial explanation of your difficulty in gaining an intuition about the meaning of $\mathcal{E}$'s c- and r-propositions is that you're thinking in terms of states and state-transitions (natural enough if one is used to working with the Situation Calculus and related formalisms). But $\mathcal{E}$'s vocabulary and underlying ontology doesn't include (global) states - just fluents, actions and time-points. So it's difficult for us to see what you might be refering to by a "possible state" in the context of $\mathcal{E}$.

To understand our intentions, it's better to think just in terms of local cause and effect, i.e. to think of the r-proposition $\ulcorner L\ \mathsf{whenever}\ C\urcorner$ as meaning "C is a minimally sufficient cause for F", and the c-proposition $\ulcorner A\ \mathsf{initiates}\ F\ \mathsf{when}\ C\urcorner$ as meaning "C is a minimally sufficient set of conditions for an occurrence of A to have an initiating effect on F".

We include "minimally" here to express our feeling that it's not intuitive to include completely irrelevant fluents in the set C. Hence, as we indicated before, if we were to extend the semantics and entailment relation to include h-, c- and r-propositions, we really would want such propositions to be entailed if and only if they were in the domain description, at least for the simple classes of domain descriptions we've defined so far. (Hence, strictly speaking, we might want to forbid pairs of statements within a single domain description such as $\ulcorner L$ whenever $C1\urcorner$ and $\ulcorner L$ whenever $C2\urcorner$ where $C1$ was a proper subset of $C2$, because the second proposition is redundant).

However, we retain sympathy for your general arguments about the need, ultimately, for theories in classical logic or similar, and for defining entailment in terms of truth functions (as we've effectively done for t-propositions). It is of course debatable whether such theories need to be centered around the notions of global states and state transitions. One's intuitions and preferences about this are probably coloured by one's experience.

Rob and Tony

Q4. **Michael Gelfond** (3.11)

Dear Tony and Rob. I am trying to understand the relationship between your language $\mathcal{E}$ and language $\mathcal{L}$ by Baral, Provetti and myself.

To do that I need some good intuitive understanding of the meaning of statements of $\mathcal{E}$ and I am having some difficulties here. My feeling is that the meaning really depends on what you call "the structure of time". If time is linear then your happens-at corresponds exactly to our occurs-at and your $\ulcorner F$ holds-at $T\urcorner$ to our $\ulcorner f$ at $T\urcorner$. In both cases we have actual occurrences at moments of time (or actual situations as we call them). If time is branching as in your second example in the paper where T corresponds to the sequence of actions then I do not fully understand the meaning of, say, $\ulcorner A$ occurs-at $S0\urcorner$. If it is still a statement of actual occurrence then I think that $\ulcorner A1$ occurs-at $S0\urcorner$ and $\ulcorner A2$ occurs-at $S0\urcorner$ should cause inconsistency. (In the case of linear time we just have concurrent actions).

The meaning of holds-at also seems to change. Instead of actual observations it becomes hypothetical. If I am right then I think this property of the language should be somewhat stressed. If not then some explanation will help.

The goal of $\mathcal{L}$ (as well as of the work by Pinto and Reiter) was to combine situation calculus ontology with actual history of the dynamical system. Since we have both we can combine reasoning about actual occurrences of actions and observations about values of fluents at particular moments of time with hypothetical reasoning of situation calculus useful for planning, counterfactual reasoning, etc.

Can you (and do you want to) use $\mathcal{E}$ for the same purpose?

My other questions are about your logic program. I do not fully understand your definition of initiation point. Do I understand correctly that it should be changed? If so, what happens with the correctness of logic program?

It may be useful to use some semantics of logic program instead of using SLDNF directly. SLDNF can give some results which are correct w.r.t. your specification even though the program is semantically meaningless (Say, its Clark's completion is too weak or inconsistent, or it does not have stable model, etc.) If you prove that the program is semantically correct one will be able to use this result directly even if your program is run on, say, XDB or SLG (which checks for some loops) and not under Prolog.

Finally, more comments on LP4 will help. I find comments like "$Resolve(A, B, C, D)$ is true iff [some English description]" extremely useful. Similarly for disjunctive_form, partition, etc.

A4. **The Authors** (5.11)

Hello Michael, thanks for your question (several questions in fact!). Here are replies to each of your points in turn.

You wrote:

> *I am trying to understand the relationship between your language $\mathcal{E}$ and language $\mathcal{L}$ by Baral, Provetti and myself.*

This is indeed an interesting question, and one that we tried to address to some extent in our first (JLP) paper on $\mathcal{E}$ (see Section 3, last three paragraphs).

You wrote:

> *My feeling is that the meaning really depends on what you call "the structure of time". If time is linear then your* happens-at *corresponds exactly to our* occurs-at *and your* $\ulcorner F$ holds-at $T\urcorner$ *to our* $\ulcorner f$ at $T\urcorner$.

Yes, that seems correct.

You wrote:

> *If time is branching as in your second example in the paper where T corresponds to the sequence of actions then I do not fully understand the meaning of, say,* $\ulcorner A$ occurs-at $S0\urcorner$. *If it is still a statement of actual occurrence then I think that* $\ulcorner A1$ occurs-at $S0\urcorner$ *and* $\ulcorner A2$ occurs-at $S0\urcorner$ *should cause inconsistency.*

Yes, the meaning of statements such as $\ulcorner A$ happens-at $S0\urcorner$ would indeed be hard to dissect if put in this type of domain description, hence we've avoided doing so in our examples.

Our intuition about Situation Calculus terms such as $S0$ and $Result(A, S0)$ is that they refer to (hypothetical) periods of time *between* (hypothetical) action occurrences. In other words, for all actions A, $S0$ is the period of time immediately before the (hypothetical) occurrence of A, and $Result(A, S0)$ is the period of time immediately afterwards.

Now, in order to simulate Situation-Calculus-like hypothetical reasoning in $\mathcal{E}$, we need to refer to the exact points at which actions (hypothetically) occur. Hence we include extra points in our structure of hypothetical time, such as $Start(Result(A, S0))$ (written $Start([A])$ in our syntax), and require that

$$S0 < Start([A]) < [A]$$

We then write $\ulcorner A$ happens-at $Start([A])\urcorner$ to assert that there is indeed a hypothetical occurrence of A just before the hypothetical time-point $[A]$ (i.e. $Result(A, S0)$. Once we've included the complete set of assertions such as this in the domain description, we can use the same general principles of initiation, termination and persistence (encapsulated in our Definitions 9 and 13 of a model) to reason about what holds in this branching structure of (hypothetical) time.

Like the Situation Calculus and the Language $\mathcal{A}$, with time structures such as this everything is intended to be in hypothetical mode, so that, as you suggest, $\ulcorner F$ holds-at $[A1, A2]\urcorner$ should be read as "F is true in the hypothetical situation $[A1, A2]$".

It is straightforward to extend this approach to partially deal with hypothetical reasoning about concurrent actions, by adapting Chitta Baral's and your ideas. Our structure of time would include sequences of sets of action symbols, e.g. $[C1, C2]$, and, for example, h-propositions of the form $\ulcorner A$ happens-at $Start([C1, C2])\urcorner$ for each A in $C2$.

You wrote:

The goal of $\mathcal{L}$ (as well as of the work by Pinto and Reiter) was to combine situation calculus ontology with actual history of the dynamical system.

Yes. A plug for Miller and Shanahan (JLC 1994) is irresistible here! That work had the same aim (as you point out in your papers), and there is perhaps more similarity between $\mathcal{L}$ and [Miller and Shanahan] than with [Pinto and Reiter]. [Miller and Shanahan] also has the advantage that it deals with concurrent, divisible and overlapping actions.

You wrote:

Since we have both [situation calculus ontology and an actual history] we can combine reasoning about actual occurrences of actions and observations about values of fluents at particular moments of time with hypothetical reasoning of situation calculus useful for planning, counterfactual reasoning, etc.

Can you (and do you want to) use $\mathcal{E}$ for the same purpose?

We haven't thought about this a great deal, although it seems possible that hypothetical and "actual" reasoning (for want of a better term) could be combined in $\mathcal{E}$ by an appropriately rich structure of time. (A simple solution might be to index hypothetical time-points such as $[A1, A2]$ with the actual time-point - typically a natural or real number - from which they were being hypothetically projected, and extend the ordering between all time-points appropriately.)

But (at the risk of re-opening an old and seemingly unstoppable debate), at least for planning our first choice would be to use abduction with a linear time structure rather than deduction with a hypothetical branching time structure. Again, there are some remarks about this in the original (JLP) paper on the Language $\mathcal{E}$.

You wrote:

My other questions are about your logic program. I do not fully understand your definition of initiation point. Do I understand correctly that it should be changed? If so, what happens with the correctness of logic program?

The definition doesn't need to be changed. The reply to Michael Thielscher simply fills in the details that we did not include in the paper. So the proof of correctness of the logic programs is unchanged. Also note that the notions of initiation and termination points are implemented in the logic programs using Proposition 2.

You wrote:

> *It may be useful to use some semantics of logic program instead of using SLDNF directly. SLDNF can give some results which are correct w.r.t. your specification even though the program is semantically meaningless (Say, its Clark's completion is too weak or inconsistent, or it does not have stable model, etc.) If you prove that the program is semantically correct one will be able to use this result directly even if your program is run on, say, XDB or SLG (which checks for some loops) and not under Prolog.*

We agree, and we are in fact working on these lines, as we say in the paper towards the end of Section 5. The point is that the present approach gives us a baseline translation that would be accepted by any semantics of logic programs, at least for those cases (as you say) where the corresponding logic program has a meaning under any semantics. Of course, there is also the debate as to whether every logic program should have a meaning, but this is probably not the place to discuss this issue.

You wrote:

> *Finally, more comments on LP4 will help. I find comments like "Resolve(A, B, C, D) is true iff [some English description]" extremely useful. Similarly for disjunctive_form, partition, etc.*

Yes, sorry. *Resolve* is just a simple implementation of a propositional resolution based prover for positive or negative literals. *Resolve*$(l1, c, l, t)$ means that we can show that l holds by resolution starting from the clause corresponding to *Whenever*$(l1, c)$ applied at the time instant t. (The details are really not that important, and in fact *Resolve* can be replaced by any sound propositional theorem prover). It first transforms the "implication" of the r-proposition into normal disjunctive form, using the predicate *DisjunctiveForm*, then the *Partition* predicate picks out the literal l that we are interested in proving, and finally we try to show through the predicate *NothingHoldsIn* that the rest of the disjunction is false, by showing that for each of its literals its negation holds. So, as we say above, it is just a simple and naive implementation of resolution.

Rob and Tony.