Using Histories to Model Observations in Theories of Action

Javier A. Pinto*
Departamento de Ciencia de la Computación, Escuela de Ingeniería, Pontificia Universidad Católica de Chile, Casilla 306, Santiago, CHILE
jpinto@ing.puc.cl

Abstract. In this article we present an approach to integrate observations in theories of action written in the Situation Calculus, a state–based, branching temporal language. In our approach, observations are added as dynamic constraints on *valid histories*. We describe two applications of these dynamic constraints, namely the representation of observations of action occurrences as well as observations of truth values of fluents.

An important novelty of our proposal is that we do not need to make unnecessary assumptions regarding the occurrence or non-occurrence of events. Furthermore, by virtue of exploiting the branching nature of time in the Situation Calculus, we can reason, within the logic, about all possible ways in which the world can evolve; i.e., without having to appeal to meta-theoretic constructs.

1 Introduction

The Situation Calculus is a logical language designed to represent theories of action and change. One of its most salient features is that time is viewed as a branching structure. Indeed, in the Situation Calculus, there is a distinguished initial situation, S_0, and there are infinitely many other situations that result from the different actions that might be performed. The structure of situations forms a tree rooted at S_0. Any sequence of actions defines a branch that starts in the initial situation. A situation is considered to be legal, if all the actions that lead to it starting in S_0 obey precondition constraints that might be specified.

The theories of action written in the Situation Calculus include specifications of knowledge about the initial situation, plus the specification of *causal laws* that may be written as *effect axioms.* These effect axioms describe how change arises as actions are performed in the world. In order to deal with the *frame problem*, we use Reiter's monotonic approach [10]. Thus, we make completion assumptions, which, together with the effect axioms, lead to the *successor state axioms*. A successor state axiom

*This work has been partly funded by Fondecyt under grants 195-0885 and 197-1304

establishes, for each fluent[1] in the domain, necessary and sufficient conditions for the fluent to be true after an action is performed.

There has been a great deal of work to extend the language of the Situation Calculus in the context of Reiter's solution to the frame problem [13, 6, 9, 12]. In the same spirit, this article seeks to extend the language of the Situation Calculus in order to add the ability to incorporate observations regarding action occurrences and the status of fluents into the logical theories. This problem has been addressed by Shanahan and Miller [4], and by Pinto and Reiter [8]. Both approaches select one possible development, or branch, in the Situation Calculus as the actual development. Any observation is regarded as a constraint on this single actual development. A related approach is used by Baral, Gelfond, and Provetti [1] in the context of the $\mathcal{A}$ language. An important difference is that the Situation Calculus based approaches are axiomatic and rely on standard Tarskian semantics, whereas the $\mathcal{A}$ language relies on a special definition of an entailment relation in order to model reasoning about change [2].

In this paper we present a Situation Calculus in which observations are specified as constraints on the *legal* world developments or *partial histories*. For instance, assume that we state that some action A occurs at a certain time point T. Under these circumstances, we consider that any development that does not include the occurrence of A at time T is illegal. Our approach to representing observations, regarding action occurrences or fluent values, allows us to accept those observations without over-committing to some notion of preferred development. For instance, if the only observation available is that action A occurs, we do not make the assumption that A is the only action that occurs. Rather, we assume that the information is inherently incomplete and that all possible developments that include A are possible. In fact, we show that our axiomatization does not preclude one from considering action occurrences that are not entailed by the theory. This feature allows one to interpret statements about action occurrences as incomplete information about the development of the world. In certain applications, it might be necessary to make assumptions regarding the actions that occur in the world. For instance, one might want to introduce notions of *likely* or *preferred developments*. How this can be done is matter for future research.

The structure of the paper is as follows. In section 2, we present the theoretical framework on which the rest of the paper is based. We also present the basic constructs that allow us to represent observations. In section 3, we introduce the notion of situation legality. Also in section 3, we show how observations can be represented as constraints on the legality of situations. Finally, in section 4 we present our conclusions and discuss our ongoing research.

2 Theoretical Framework

The basic language and axiomatization are based on extensions to the Situation Calculus that have been proposed in [6, 12]. Of particular interest to us is the definition of a basic language over which different kinds of observations can be specified. As mentioned in the introduction, observations are taken as incomplete information regarding

[1] A fluent is a dynamic property of the world.

the evolution of the world.

The Situation Calculus is a sorted first order language extended with some second order features. We distinguish the sorts $\mathcal{A}$, $\mathcal{C}$, $\mathcal{S}$, $\mathcal{F}$, and $\mathcal{T}$ for primitive actions, concurrent actions, situations, fluents, and time. We also use a sort $\mathcal{D}$ for other objects. The sort $\mathcal{T}$ ranges over the real numbers. We use the convention that all variable symbols start with lower case letters, whereas constant symbols start with upper case letters. Also, we use the Greek letter φ as a second order predicate variable, which is only used in the specification of the induction axiom given in the next subsection. The sort of the variables and constants should be obvious from the context. In formulas, we assume that all free variables are universally quantified. The sort $\mathcal{C}$ corresponds to sets of simple actions. We take the standard interpretation for sets and don't axiomatize them.

Function and Predicate Symbols

- $start : \mathcal{S} \to \mathcal{T}$. If s is a situation term, then $start(s)$ denotes the time at which that situation starts.
- $do : \mathcal{C} \times \mathcal{S} \to \mathcal{S}$. If c denotes a concurrent action and s denotes a situation, then $do(c,s)$ denotes the situation that results from performing c in s.
- $\in \subseteq \mathcal{A} \times \mathcal{C}$. This is the standard membership operator for sets. Thus, if a denotes a primitive action and c denotes a *concurrent* action, then $a \in c$ would be true if a was part of the concurrent action c.
- $holds \subseteq \mathcal{F} \times \mathcal{S}$. If f denotes some fluent and s denotes a situation. Then $holds(f,s)$ is intended to mean that fluent f is true in situation s.
- $not : \mathcal{F} \to \mathcal{F}$. If f denotes some dynamic property or fluent, $not(f)$ denotes its negation.
- $hasOccurred \subseteq (\mathcal{C} \cup \mathcal{A}) \times \mathcal{T} \times \mathcal{S}$. $hasOccurred(a,t,s)$ would be true if action a occurred at time t on the path that leads from S_0 to s. The first argument of *hasOccurred* can be either a concurrent action or a primitive action.
- $\prec \subseteq \mathcal{S} \times \mathcal{S}$. If s_1 and s_2 denote situations, then $s_1 \prec s_2$ is true if some sequence of concurrent actions leads from s_1 to s_2. We use the abbreviation $s_1 \preceq s_2 \equiv s_1 \prec s_2 \vee s_1 = s_2$.
- $Poss \subseteq (\mathcal{A} \cup \mathcal{C}) \times \mathcal{S}$. If c is a concurrent action and s a situation, then $Poss(c,s)$ is true if the concurrent action c is possible in s (i.e., if the action preconditions are satisfied in the situation denoted by s).
- $< \subseteq \mathcal{T} \times \mathcal{T}$. Standard comparison operator for reals (the abbreviation $\leq$ is also used).
- $legal_{poss}, legal \subseteq \mathcal{S}$. These are the legality predicates that we need to model observations (observations of action occurrences and observations of fluent values). As will be discussed later, a situation is *legal* if the set of situations that lead to it is faithful to the observations.
- $proper \subseteq \mathcal{S}$. Is an auxiliary predicate used to define *legal* situations.
- $Sit \subseteq \mathcal{T} \times \mathcal{S} \times \mathcal{S}$. If t is denotes a time point, and s and s_h denote situations, then the literal $Sit(t,s,s_h)$ should be true if t falls in the situation s in the path that leads from S_0 to s_h.

- $occurs_{\mathcal{T}} \subseteq \mathcal{A} \times \mathcal{T}$. Is used to represent observations of action occurrences. Thus, if a and t denote an action and a time point, then $occurs_{\mathcal{T}}(a,t)$ represents the observation that a occurs at time t.
- $holds_{\mathcal{T}} \subseteq \mathcal{F} \times \mathcal{T}$. Is used to represent observations of fluent values. Thus, if f and t denote a fluent and a time point, then $holds_{\mathcal{T}}(f,t)$ represents the observation that f holds at time t.

In this paper, we consider a Situation Calculus in which actions are considered to be instantaneous. As discussed elsewhere (e.g., [5]), this restriction does not limit the representational power of the language. Also, we ignore precondition interactions and concurrent effects or cancellations. These problems are addressed in [5, 6].

The issues discussed in this paper are orthogonal to those mentioned above. Therefore, any approach to deal with precondition interactions and concurrent effects or cancellation of effects, should integrate well with the theory presented in this paper.

2.1 Basic Situation Calculus Axiomatization

In the Situation Calculus that we present here, there is an initial situation, denoted by the special constant symbol S_0. Given an arbitrary situation s, a new situation is obtained by performing some concurrent action c in s. Following Reiter [10], we use an induction axiom in order to force the set of situations to form a tree rooted at S_0. Thus, an arbitrary situation s identifies a unique sequence of concurrent actions that leads from S_0 to s. Each situation s in the tree has a unique starting time [8]; this time corresponds to the time of the last action in the sequence leading to s. On the other hand, there is no unique ending time for a situation. In fact, in [8], Pinto and Reiter define an *end* predicate that takes an action and a situation as arguments. In that approach, the end of a situation s, with respect to an action c, corresponds to the start time of the situation $do(c,s)$. Therefore, each situation has one ending time associated to each possible concurrent action. These multiple ending times arise from the fact that, in the Situation Calculus, the future branches.

The basic axioms for situations are:

$$(\forall \varphi).[\varphi(S_0) \wedge (\forall s,c)\ (\varphi(s) \supset \varphi(do(c,s)))] \supset (\forall s)\ \varphi(s), \tag{1}$$
$$do(c_1,s_1) = do(c_2,s_2) \supset c_1 = c_2, \tag{2}$$
$$s_1 \prec do(c,s_2) \equiv s_1 \preceq s_2, \tag{3}$$
$$s_1 \prec s_2 \supset \neg s_2 \prec s_1, \tag{4}$$
$$Poss(c,s) \equiv (\forall a)[a \in c \supset Poss(a,s)], \tag{5}$$
$$Poss(c,s) \supset start(s) < start(do(c,s)). \tag{6}$$

Axiom (1) states that the only situations that exist are S_0, the initial situation, and all the situations that are reachable from S_0 by performing sequences of actions starting in S_0. Axiom (2) ensures that the structure of situations forms a tree. Axioms (3) and (4) define the relation $\preceq$. The literal $S_1 \prec S_2$ is true if there is a path from S_1 to S_2 in the Situation Calculus tree; i.e., S_2 is reached by performing a sequence of actions starting from S_1. Axiom (5) states that a concurrent action is possible iff all its constituent actions are individually possible. We assume that the predicate *Poss* for simple actions is completely specified by domain axioms.

As mentioned before, the function *start* denotes the starting time of a situation. Axiom (6) ensures that the situations along a path are chronologically ordered. This ordering is only enforced for situations that are obtained when performing actions that are possible.

In the Situation Calculus we use in this paper, we treat fluents at the object level. An alternative, that we used in [7], is to regard fluents as predicates in the language. The *reification* of fluents is due to the fact that we want to be able to observe them. Furthermore, we need to be able to specify how the branches of the Situation Calculus tree are constrained given the observation that some arbitrary fluent is observed to hold. The reification is done by using the predicate *holds* and a function symbol *not*, along with the following axioms:

$$holds(not(f), s) \equiv \neg holds(f, s), \tag{7}$$

$$not(not(f)) = f. \tag{8}$$

Also, we require *unique names axioms* for fluent and action terms[2]. We omit these axioms (the interested reader may consult [6] for a similar use of these unique names axioms).

2.2 Situation Legality

In the previous sub-section, we presented the basic axiomatization that defines the universe of situations $\mathcal{S}$ in terms of the initial situation S_0 and the set of concurrent actions $\mathcal{C}$. Notice that this universe of situations is defined without taking into consideration whether or not the situations were actually possible. Our objective is to specify which situations in the Situation Calculus tree can be regarded as viable or *legal* given knowledge regarding action preconditions, laws of physics, occurrence statements, etc. In [11], Reiter defines a first notion of legality which basically says that a situation s is legal if it can be reached by performing actions in S_0 and all the actions leading to s are performed in situations in which their preconditions are met. This motivates the following:

Definition 2.1 *A situation s is considered legal with respect to the action preconditions, denoted $legal_{poss}$, if all the actions that lead from S_0 to s have been performed in situations in which they are possible. Thus, $legal_{poss}$ is defined as:*

$$legal_{poss}(s) \equiv [s = S_0 \vee (\forall c, s')\ do(c, s') \preceq s \supset Poss(c, s').] \tag{9}$$

As illustrated by the legality predicate $legal_{poss}$, we can constrain the tree of situations by refining the notion of legality. Legal situations form a subtree of the original tree structure of situations. Given an observation, many branches in the Situation Calculus tree can be pruned. This is the approach taken in section 3, where all the branches that do not comply with the given observations are regarded as not *legal*.

[2]These unique names axioms exclude nesting of the *not* fluent function symbol.

2.3 Histories or Paths

The notion of a history, within the Situation Calculus, corresponds to the sequence of actions that leads to a situation starting from S_0. Also, we refer to this as a *partial path* within the Situation Calculus tree[3]. Any situation s identifies a unique partial path that starts at S_0 and ends with s. In fact, any situation in the domain $\mathcal{S}$ is fully identified with a sequence of actions. For instance, the sequence $c_1, c_2, \ldots, c_n$ uniquely identifies the situation S_n, where:

$$S_n = do(c_n, do(\ldots, do(c_1, S_0)\ldots)),$$

which can also be abbreviated as $do([c_1, \ldots, c_n], S_0)$. Thus, all the actions $c_1, \ldots, c_n$ can be said to belong to or have occurred before S_n, and they correspond to the history of situation $do([c_1, \ldots, c_n], S_0)$. It turns out that one essential feature of the Situation Calculus, not available in linear time languages, is that there are terms denoting situations, which also denote histories. Thus, a situation s corresponds to a single history or linear time description of one possible evolution of the world starting from S_0. A situation s' is said to appear or belong to a history s iff $s' \prec s$. Thus, S_0 belongs to all histories (except to the empty history, which corresponds to S_0). Notice that the notion of history comes for free in the Situation Calculus, and no added machinery is necessary to introduce them in the language.

The branching nature of time in the Situation Calculus makes it difficult to impose a unique ending time on situations (they do have a unique start time, since time in the Situation Calculus is left linear). On the other hand, in a given history, any situation has a unique starting and a unique ending time. If s and $do(c,s)$ belong to some history s', then $start(s)$ and $start(do(c,s))$ correspond to the starting and ending times of s along the history s'.

In this paper, we exploit the notion of histories to define occurrences. We do so by defining the notion of a concurrent action c occurring within a history, partial path or situation:

Definition 2.2 *We say that a concurrent action c has occurred at time t along the history of a situation s with the predicate hasOccurred as follows:*

$$hasOccurred(c,t,s) \equiv (\exists s')\ do(c,s') \preceq s \wedge start(do(c,s')) = t. \tag{10}$$

This can be extended for primitive actions as:

$$hasOccurred(a,t,s) \equiv (\exists c)\ a \in c \wedge hasOccurred(c,t,s). \tag{11}$$

Finally, we introduce the following:

Definition 2.3 *We say that $Sit(t,s,s_h)$ is true whenever s corresponds to the situation in the history s_h which includes time t (thus t is between the start of s and the end of s along the history s_h). Formally,*

$$Sit(t,s,s_h) \equiv (\exists c)\ do(c,s) \preceq s_h \wedge start(s) < t \wedge t \leq start(do(c,s)). \tag{12}$$

[3] A *path* in the Situation Calculus tree corresponds to an infinite sequence of situations that start in S_0, a partial path is a prefix of infinitely many paths.

Using these definitions we have:

Observation 2.1 *From axioms (1) to (12) it follows that:*

$$Sit(start(do(c,s)),s,do(c,s)),$$
$$Sit(t,s_1,s_h) \wedge Sit(t,s_2,s_h) \supset s_1 = s_2.$$

As we show in the next section, the predicates *hasOccurred* and *Sit* allow us to express *dynamic state constraints* to represent observations within the language of the situation calculus. One key idea, is that in the Situation Calculus, situation objects denote histories. Therefore, one can quantify over histories by quantifying over situations.

3 Formalizing Observations

3.1 Situation Legality

Observations – i.e. observations of action occurrences or observation of the status of fluents – are considered to be primitive statements that constrain the paths in the Situation Calculus tree that can be regarded as legal. Legality is introduced in the language with the predicate $legal \subseteq \mathcal{S}$, which in turn is defined using the notion of a *proper history* which is introduced with the predicate $proper \subseteq \mathcal{S}$. A situation s is considered to be a proper history if the sequence of actions that lead to s is consistent with all the observations that might have been introduced. *proper* is a predicate defined in terms of occurrences and the $legal_{poss}$ legality predicate. However, for presentation purposes, the specification of *proper* will be given as a set of implications or necessary conditions. In the end, the definition of *proper* is taken to be the completion of these necessary conditions.

The first necessary condition for a situation to represent a proper partial path is:

$$proper(s) \supset legal_{poss}(s). \tag{13}$$

All other constraints on *proper* will be introduced in the context of the occurrence predicates that are introduced later.

The predicate *legal* should be taken as a refinement of the legality notion introduced with the $legal_{poss}$ predicate. In fact, if a situation is *legal*, then it is also $legal_{poss}$. We say that a situation is *legal* if it obeys all constraints, related to action preconditions and *occurrence* sentences. The definition of *legal* is done in terms of the *proper* predicate. The definition of *legal* is:

Definition 3.1 *A situation or history s is considered to be legal with respect to the action preconditions and observation statements if every time after its start there is a proper future history that contains s. This is formalized as:*

$$legal(s) \equiv ((\forall t)\ start(s) < t \supset (\exists s')\ s \prec s' \wedge t < start(s') \wedge proper(s')) \tag{14}$$

Notice that according to (14) for a situation to be legal, at any time t after its start, some action must always be possible some time t' after t. This precludes one from writing theories in which infinitely many actions arise within a history. Thus, axiomatizing Zeno's paradox in this context would yield a contradiction. It should be noted here that the notion of a history or partial path, which is formally equivalent to a situation, is essential for the representation of observations. In fact, the basic language introduced in the previous section might be considered as the foundational layer that allows one to introduce observations. The set of axioms (1)-(14) constitute the background axioms that will be used to express dynamic constraints and to represent observations.

3.2 Observing Values of Fluents

As mentioned in the introduction, in a Situation Calculus theory we normally specify observations regarding the values of fluents in the initial situation, S_0. However, oftentimes we have knowledge regarding time points other than the initial one. This is particularly true when we try to formalize narratives that might include direct observations. As a simple example, consider the narrative:

> Yolanda leaves home for work 7:30 in the morning while her husband Raoul is at home. She arrives at her office and calls home at 8 a.m.; her son, Art, tells her that dad, Raoul, is not there.

This simple narrative will allow us to exemplify our approach to handling observations. We use the constants $Yolanda$, $Raoul$, and Art, to denote three individuals. We also use location constants $Home$ and $Office$. We assume that all domain terms are conveniently sorted.

We will have two types of actions. First, the action *leaves* by which an individual leaves a certain location; the action has two arguments, so that $leaves(Yolanda,Home)$ denotes the action of Yolanda leaving her home. The other action is *arrives* which also has two arguments, so that $arrives(Yolanda,Office)$ denotes the action of Yolanda arriving at her office.

Finally, in this domain we have the fluent *at* which takes two arguments, an *individual* and a *location*. Thus the fluent term $at(Yolanda,Home)$ would hold in those situations in which Yolanda is at home. We also have the fluent *commuting* with one argument, such that $commuting(Yolanda)$ would hold in situations in which $Yolanda$ is traveling.

The axiomatization of this example in the language of the Situation Calculus includes the following action precondition axioms:

$$Poss(leaves(ind,loc),s) \supset holds(at(ind,loc),s), \tag{15}$$

$$Poss(arrives(ind,loc),s) \supset holds(commuting(ind),s), \tag{16}$$

There are gross simplifications regarding axiom (16), since the conditions for arrival are much more complex. However, a more precise axiomatization of this domain would not contribute to the issue that concerns ourselves in this article.

Since we are using Reiter's monotonic approach to the frame problem, we proceed to specify, for each fluent function, with the sole exception of the fluent function *not*, a *successor state axiom*. In the example, we only have fluent function *at*. Thus, we write:

$$\begin{aligned}
Poss(a,s) \supset holds(at(ind,loc),do(a,s)) \equiv \\
& holds(at(ind,loc),s) \wedge a \neq leaves(ind,loc) \vee \\
& a = arrives(ind,loc).
\end{aligned}$$

Finally, we specify the values of fluents in the initial situation.

$$\begin{aligned}
& holds(at(Yolanda,Home),S_0), \\
& holds(at(Raoul,Home),S_0), \\
& start(S_0) < 730.
\end{aligned}$$

As mentioned before, in a standard theory of change in the Situation Calculus we are able to specify the dynamics of the world. The theory includes successor state and precondition axioms. Also, we include a specification of the initial situation (which does not need to be complete). Such a theory allows us to wonder about the outcome of all possible sequences of actions that might be started at S_0. For instance, in the example, we can ask what the world looks like in the situation:

$$S_2 = do(arrives(Yolanda,Office),do(leaves(Yolanda,Home),S_0)).$$

or in the situation:

$$\begin{aligned}
S_4 = do(&arrives(Yolanda,Office), \\
& do(leaves(Raoul,Home),do(leaves(Yolanda,Home),S_0))).
\end{aligned}$$

Are these situations legal? Since we have not yet introduced observations, we can take *proper* to be the completion of (13), namely:

$$proper(s) \equiv legal_{poss}(s).$$

Therefore, the situations S_2 and S_4 are both legal, since the actions obey the precondition axioms (15) and (16). In order to incorporate the *observation* that Raoul is not home at eight o'clock, we add the literal:

$$\neg holds_{\mathcal{T}}(at(Raoul,Home),800). \tag{17}$$

By modeling the acquisition of information in this manner, we are glossing over several important open issues. For instance, in order to properly model the communication between Art and Yolanda, we should introduce communication actions. Also, communication among individuals should have no effect on the environment. Instead, communication actions should change the epistemic state of the individuals [13]. These issues are left for further research.

Now, how should we interpret assertions like (17). Our proposal is to *prune* the subtree of legal situations. This is done by providing further axioms about the legality of situations. Thus, with respect to the example, we want an axiomatization in which a situation is considered legal if and only if it has some possible future in which Raoul is not at home at eight o'clock. Furthermore, a situation cannot be legal if it starts after eight o'clock and is part of a history in which Raoul is at home at eight o'clock. This notion of legality is achieved by using axiom (14) and by providing further constraint on the definition of *proper*. Thus, we have (13) and the following:

$$proper(s) \supset [holds_{\mathcal{T}}(f,t) \wedge t < start(s) \wedge Sit(t,s',s) \supset holds(f,s')]. \qquad (18)$$

Therefore, in our particular example, any situation that represents a proper history, and that starts after eight o'clock, must be part of a history in which $at(Raoul, Home)$ does not hold at eight o'clock.

If we take *proper* as the completion of (13) and (18), we obtain:

$$\begin{aligned} proper(s) \equiv\ & legal_{poss}(s) \wedge \\ & [holds_{\mathcal{T}}(f,t) \wedge t < start(s) \wedge Sit(t,s',s) \supset holds(f,s')]. \end{aligned} \qquad (19)$$

From where it follows that the situation:

$$S_2 = do(arrives(Yolanda, Office), do(leaves(Yolanda, Home), S_0)),$$

where

$$start(do(arrives(Yolanda, Office), do(leaves(Yolanda, Home), S_0))) = 800,$$

cannot be legal. In fact, the successor state axiom for the fluent *at*, along with the specification for the initial state S_0, allow us to infer that in situation S_2, the fluent $at(Raoul, Home)$ holds, independently of the time assignment for the situations in the history of S_2. Thus, S_2 violates the observation (17).

On the other hand, the situation:

$$do(arrives(Yolanda, Office), do(leaves(Raoul, Home), do(leaves(Yolanda, Home), S_0))).$$

is legal, as long as it has a starting time earlier than 800.

In general, we have:

Theorem 3.1 *From axioms (1)-(12), (14) and (19), it follows that:*

$$holds_{\mathcal{T}}(f,t) \wedge legal(s) \wedge t < start(s) \wedge Sit(t,s',s) \supset holds(f,s') \qquad (20)$$

Thus, if fluent f is observed to hold at time t, then all legal histories that include time t, must make f true at time t.

3.3 Observing Action Occurrences

In the narrative of the previous subsection, we have references to different actions that occur. For instance, Yolanda's leaving home for work and her arrival to the office. The axiomatization that we presented does not, in any sensible manner, establish that these actions must occur. Indeed, if there were other ways in which Yolanda could have shown up at work, they would act as alternative explanations.

We need some way of explicitly stating that certain actions occur. In the example, we want to say that at some time before eight o'clock, Yolanda leaves home, and that she arrives at eight o'clock. For simplicity, we don't refer to the phone call as an explicit event. In order to specify the occurrences, we use the predicate *occurs* which takes an action and a time as arguments. Therefore, we may write:

$$occurs(leaves(Yolanda, Home), 730),$$
$$occurs(arrives(Yolanda, Office), 800).$$

The occurring statements should also constrain the legal situations. Thus, if we state that the event $arrives(Yolanda, Office)$ occurs at eight o'clock, then we do not want to have any legal situation that starts at eight o'clock or after and that does not have a history in which the occurrence is present. To formalize this constraint, we add a new axiom for $proper$. Thus, along (13) and (18), we now have:

$$proper(s) \supset [occurs(a,t) \wedge t < start(s) \supset hasOccurred(a,t,s)]. \tag{21}$$

So far, we have introduced three axioms that establish necessary conditions for a situation to be $proper$. If we consider that these necessary conditions are also sufficient conditions, then we may replace (13), (18) and (21), with the axiom:

$$\begin{aligned} proper(s) \equiv\ & legal_{poss}(s) \wedge \\ & [holds_{\mathcal{T}}(f,t) \wedge t < start(s) \wedge Sit(t,s',s) \supset holds(f,s')] \wedge \\ & [occurs(a,t) \wedge t < start(s) \supset hasOccurred(a,t,s)]. \end{aligned} \tag{22}$$

The following theorem establishes essential conditions to deal with narratives. In fact, as discussed in [7], a simple narrative can be described in terms of these occurrences (which are called *non-preventable*) and this theorem ensures that all legal situation include the events mentioned in the narrative:

Theorem 3.2 *From axioms (1)-(12), (14) and (22) it follows that:*

$$occurs(a,t) \wedge legal(s) \wedge t < start(s) \supset hasOccurred(a,t,s).$$

Thus, in our example, for a situation to be considered a proper history, it has to fulfill the following conditions:

1. The sequence of actions leading to the situation have to obey the action preconditions. Thus, if a *leaves* action is performed, then it has to have been performed in a situation satisfying axiom (15).
2. The history of the situation must comply with all fluent value observations made before the situation's starting time.
3. All action occurrences that are observed before the starting time of the situation must belong to the history of the situation.

The occurrence constraints introduced with the *occurs* predicate have several features that are fairly restrictive. Indeed, in order for a model to satisfy a literal $occurs(A,T)$

every legal history must have the occurrence of A at the same time. However, we might want to allow for the action to occur at different time points in different histories. In order to express this, we need to introduce another type of occurrence. For instance, we might write $occurs'(A,T)$ to mean that action A must occur in every valid history before time T. This frees us from having to give A a fixed time of occurrence. In the same vein, there are many other kinds of dynamic constraints that can be expressed as occurrence of different kinds. In [7], we explore these issues and apply this approach to several problems. For instance, we show how to model some forms of *triggers* using these dynamic constraints.

4 Concluding Remarks

In this article we have presented a novel approach to representing observations of fluent values and action occurrences in the Situation Calculus as dynamic constraints on valid histories. This approach allows one to reason about hypothetical developments which obey all constraints imposed by observation literals (i.e., $holds_{\mathcal{T}}$ and *occurs* literals). One interesting feature of the approach presented is that the constraints allow one to *prune* the tree of situations, keeping *all* the histories that are consistent with the observations. This is a feature that must be present in any language used to represent incomplete knowledge about the evolution of dynamic domains. One important departure of our contribution from other related research [1, 4, 8], is that we consider observations to be constraints on all possible developments, as opposed to constraints on one preferred *actual* development.

There are several important extensions to this work that we would like to pursue. For instance, we would like to consider occurrence statements involving complex actions (actions that involve conditionals, loops, etc.). Also, we are interested in investigating our approach to specifying theories of action and the approach based on the GOLOG language being developed at the Cognitive Robotics group in the University of Toronto [3].

Another interesting issue is the study of the computational aspects of reasoning with our theory. It is likely that tasks like *plan verification* will be difficult. In particular, non-preventable occurrences may introduce difficulties to prove legality of situations. The reason is that one might be forced to think about the future of a situation in order to determine its legality. This difficulty does not always arise, and we would like to characterize classes of theories for which this reasoning is easy.

We are interested in studying planning in the presence of exogenous events, probably formalized using occurrence statements. For instance, we would like to be able to come up with a plan for an agent to go to the airport, knowing that the bus to the airport leaves for the airport every hour on the hour. However, we should not be able to prove that the plan is successful, unless we assume that the bus does not run into trouble. To do this, we can have likelihood predicates and plan for likely situations.

References

1. C. Baral, M. Gelfond, and A. Provetti. Representing Actions: Laws, Observations and Hypotheses. *Journal of Logic Programming*, 31:201–244, 1997.
2. Michael Gelfond and Vladimir Lifschitz. Representing Action and Change by Logic Programs. *The Journal of Logic Programming*, 17:301–322, 1993.
3. Hector J. Levesque, Raymond Reiter, Yves Lespérance, Fangzhen Lin, and Richard B. Scherl. GOLOG: A Logic Programming Language for Dynamic Domains. *The Journal of Logic Programming*, 31:59–84, 1997.
4. Rob Miller and Murray Shanahan. Narratives in the Situation Calculus. *The Journal of Logic and Computation*, 4(5):513–530, 1994.
5. Javier Pinto. Concurrent Events: Synergy and Cancellation of Effects. In *European Conference on Artificial Intelligence, Workshop on Logic and Change*, pages 105–110, 1994. URL = ftp://lyrcc.ing.puc.cl/pub/jpinto/ecai.ps.gz.
6. Javier Pinto. *Temporal Reasoning in the Situation Calculus.* PhD thesis, Department of Computer Science, University of Toronto, Toronto, Ontario, Canada, February 1994. URL = ftp://ftp.cs.toronto.edu/˜cogrob/jpThesis.ps.Z.
7. Javier Pinto. Occurrences and Narratives as Constraints in the Branching Structure of the Situation Calculus. *Journal of Logic and Computation*, 1998. To appear.
8. Javier Pinto and Raymond Reiter. Reasoning about Time in the Situation Calculus. *Annals of Mathematics and Artificial Intelligence*, 14(2-4):251–268, September 1995.
9. Javier A. Pinto. On the Existence and Formalization of Natural Events. In *Proceedings of the IJCAI Workshop on Nonmonotonic Reasoning, Action and Change.*, August 1995. URL = ftp://lyrcc.ing.puc.cl/pub/jpinto/ijcwrk.ps.gz.
10. Raymond Reiter. *The Frame Problem in the Situation Calculus: A Simple Solution (sometimes) and a completeness result for goal regression*, pages 359–380. Artificial Intelligence and Mathematical Theory of Computation: Papers in Honor of John McCarthy. Academic Press, San Diego, CA, 1991.
11. Raymond Reiter. Proving Properties of States in the Situation Calculus. *Artificial Intelligence*, 64(2):337–351, December 1993.
12. Raymond Reiter. Natural Actions, Concurrency and Continuous Time in the Situation Calculus. In *Principles of Knowledge Representation and Reasoning: Proceedings of the Fifth International Conference(KR'96)*, Cambridge, Massachussetts, U.S.A., November 1996. Morgan Kaufmann.
13. Richard Scherl and Hector Levesque. The Frame Problem and Knowledge Producing Actions. In *Proceedings AAAI-93*, pages 689–695, Washington, D.C., July 1993. AAAI.