How (Not) To Minimize Events

Michael Thielscher*
Department of Computer Science
Dresden University of Technology
01062 Dresden (Germany)

Abstract

When drawing conclusions about narratives, minimizing—to a reasonable extent—the occurrence of events is crucial. We argue that unguided minimization is insufficient in case events are causally connected, for it easily fails to distinguish unmotivated event occurrences from those that have a cause. Two solutions are offered, the first of which has the advantage of being straightforwardly realized but on the other hand has a restricted range of applicability. Our second solution overcomes these restrictions but requires two uncommon and novel features. First, event occurrences are identified as *fluents*, which allows to adapt a recent causality-oriented solution to the Ramification Problem so that if an event is caused by another event then the former is obtained as *indirect effect* of what caused the latter. Second, volitional actions and natural events which have no cause inside the reasoning context, are furnished with a special cause, namely, the reaching of the time-point at which they take place. We present both a high-level narrative description language and an axiomatization based on a novel Fluent Calculus in which is realized this solution to the event minimization problem.

1 INTRODUCTION

Commonsense reasoning about narratives requires, in one way or the other, to minimize the occurrence of events. For otherwise none of the intended and intuitive conclusions can be drawn. Suppose, for example, we are told that adding hot water to a mug containing a tea bag produces tea, and furthermore that a robot first places a tea bag into the empty mug and, then, pours hot water into it. The expected conclusion would be that the robot has tea afterwards. This, however, does not follow until we explicitly exclude the occurrence of a variety of events, such as the robot emptying the mug in between the two actions, someone stealing the tea bag out of the mug, or a falling tile striking the mug while water is being poured etc. Any of these events falsifies the intended conclusion, and their non-occurrence does not follow *per se* from the narrative. Hence additional measures need to be taken to conclude, by default, that none of these nor any other intervening events materialize.

As long as events are mutually independent, minimizing them wrt. a given formalized narrative in a reasonable fashion is straightforward. Once we have to consider causal dependencies among events, however, finding a good general minimization strategy becomes a non-trivial challenge. In particular, simple global minimization is insufficient in that it may fail to produce the intended and intuitively expected conclusions. Let us illustrate this point with two simple examples, where we borrow basic notions and notations from the Event Calculus variant of [Shanahan, 1996]. The occurrence of an event e at time t is represented by the atomic expression $\mathit{Happens}(e,t)$, and minimization is achieved by circumscribing predicate *Happens* in a set of formulas which axiomatize a narrative.[1]

Example 1 Suppose a robot walking towards a wall will collide with it provided the robot does not stop beforehand. Ignoring precise values for distance and speed etc., let the formula

$$\begin{array}{l} \mathit{Happens}(\mathtt{walk},t) \wedge \neg \mathit{Happens}(\mathtt{stop},t+1) \\ \qquad \supset \mathit{Happens}(\mathtt{bump},t+2) \end{array} \tag{1}$$

encode this knowledge. Consider, now, a narrative which consists of the sole fact that the robot starts walking towards the wall at time $t=5$,

* On leave from Darmstadt University of Technology.

[1] It should be stressed, however, that the problems we elaborate in the following are of general nature and so do also emerge in other than this specific approach.

that is, $Happens(\mathtt{walk}, 5)$. What would one expect to happen? Since nothing indicates that the robot stops walking at time $t = 6$, the natural conclusion would be that $\neg Happens(\mathtt{stop}, 6)$, hence $Happens(\mathtt{bump}, 7)$. Minimizing *Happens* in the formula $N = Happens(\mathtt{walk}, 5) \wedge (1)$, however, does not suffice for this entailment. Rather it results in two kinds of models, one with the intended course of events and one where the opposite is true, viz. $Happens(\mathtt{stop}, 6)$ and $\neg Happens(\mathtt{bump}, 7)$. ■

Example 2 Suppose that if the left hand side of a table is lifted but not the right hand side, then soup spills out of a bowl sitting on the table; the same happens if the right hand side only is lifted:

$$Happens(\mathtt{lleft}, t) \equiv \neg Happens(\mathtt{lright}, t) \supset Happens(\mathtt{spills}, t) \tag{2}$$

Suppose we are told that the left hand side of the table is lifted at time $t = 2$, i.e., $Happens(\mathtt{lleft}, 2)$. Then we should expect soup spilling out rather than a magical simultaneous lifting of the right hand side. As before, however, circumscribing *Happens* yields two indistinguishable kinds of models and so does not allow this conclusion. ■

In both our two scenarios, unguided minimization does not yield the reasonably expected conclusions because it is too weak a minimization strategy.[2] In the following section, we show how a more elaborated minimization strategy is obtained by formally introducing the general distinction between *actions* (which are volitional, i.e., involve a free-will decision) and *natural events*. This strategy resembles the widely used *frame/non-frame* categorization [Lifschitz, 1990] in the context of the Ramification Problem [Ginsberg and Smith, 1988a], which is employed to distinguish caused from unmotivated indirect effects. The attraction of this categorizing events as either natural or volitional actions lies in its being straightforwardly realizable. By employing this minimization strategy a number of problems are resolved, including our introductory ones. Yet it is still crucial that the domain being modeled meets certain assumptions regarding mutual independence of events, namely, that natural events do not interfere and that actions are necessarily causally independent.

In Section 3, we assess these assumptions by presenting a scenario which involves only natural events and yet where some but not all events may be caused by others. Furthermore, we argue that even from the commonsense point of view actions are not necessarily causeless, namely, in case they are reflexive (hence not volitional), or in case of normative rules, stating that an action ought to be performed under certain circumstances. A universally applicable event minimization strategy therefore requires the incorporation of a suitable notion of causality by which it is generally possible to tell apart caused from unmotivated event occurrences.

Actually this general problem shows a striking similarity to the necessity of distinguishing caused from unmotivated indirect effects as part of the broader Ramification Problem. Research in this context has recently produced several successful approaches which appeal to causality (e.g., [Elkan, 1992; Geffner and Pearl, 1992; Lin, 1995; McCain and Turner, 1995; Thielscher, 1997]). In Section 4, we show how these results can be exploited to successfully address the problem considered in the present paper. The conceptually crucial step towards this end is to identify event occurrences with fluents. This allows us to interpret formulas like (2) as so-called state constraints, which then give rise to indirect effects. The problem of deriving the right event occurrences thus becomes part of the Ramification Problem, and so we can adapt, for instance, our causality-based approach of [Thielscher, 1997] to provide a solution. We present a high-level narrative description language in which is realized this solution to the event minimization problem. In Section 5, we furthermore illustrate how this solution can be axiomatized by a novel use of Fluent Calculus techniques. We conclude in Section 6.

2 ACTIONS VERSUS NATURAL EVENTS

The reason for global minimization failing to produce the intended conclusions in the introductory examples becomes apparent when we analyze the specifications (1) and (2) from a purely logical perspective. The constraint in (2), for instance, logically entails this formula:

$$Happens(\mathtt{lleft}, t) \wedge \neg Happens(\mathtt{spills}, t) \supset Happens(\mathtt{lright}, t)$$

This explains the existence of the unintended model, i.e., where $\neg Happens(\mathtt{spills}, 2)$ is assumed and $Happens(\mathtt{lright}, 2)$ follows. Similarly, formula (1) is equivalent to the implication,

$$Happens(\mathtt{walk}, t) \wedge \neg Happens(\mathtt{bump}, t + 2) \supset Happens(\mathtt{stop}, t + 1)$$

sanctioning the unintended model obtained in Example 1.

[2] It is worth mentioning that the more sophisticated so-called *chronological* minimization [Shoham, 1988] would solve our first but not the second example, because the two conflicting events, $Happens(\mathtt{lright}, 2)$ vs. $Happens(\mathtt{spills}, 2)$, are in no chronological order. Likewise, in the approach of [Stein and Morgenstern, 1994] an event can only *motivate*, as it is called, another event if the latter occurs at a later time-point, i.e., not concurrently. So this notion of motivation, too, does not help us detecting the unintended model in Example 2.

These observations indicate that both the two specifications (1) and (2) just contain insufficient information to rule out the respective anomalous model. In fact, the reason for us rejecting some of the minimal models is that we employ additional domain knowledge to the effect that some events, like the spilling of the soup, may be the natural consequence of certain circumstances while other events, like the lifting of the table, require a volitional act. In other words, the occurrence of the latter depends on a deliberate, free-will decision of some agent. Whenever a conflict needs to be resolved, we seem to prefer the occurrence of a natural event rather than postulating the performance of an action which the narrative does not necessitate. This argument also applies to Example 1: The robot bumping at the wall is a natural consequence of its walking towards it, whereas the robot stopping beforehand involves the explicit decision to act.

The distinction between actions on the one hand and natural events on the other, provides the basis for a refined strategy for event minimization in case causal dependencies are to be taken into account. If a domain description includes knowledge as to the category to which each event belongs, then minimization can exploit this knowledge in order to rule out any unmotivated action. What is especially appealing is that this solution to our two introductory example problems can be straightforwardly realized, e.g. in the approach of [Shanahan, 1996] as follows: We first introduce two predicates $A\_Happens(e,t)$ and $E\_Happens(e,t)$. The former is to be used whenever e is an action, the latter whenever e is a natural event. Let N be a set of formulas formalizing the course of events in a narrative, then by using *priority circumscription* [Lifschitz, 1987], written $CIRC[N; A\_Happens > E\_Happens]$, $A\_Happens$ can be minimized with higher priority than $E\_Happens$.[3] This circumscription policy reflects our intention to prefer minimization of action performances over the occurrence of natural events.

Example 1 (continued) In accordance with the above proposal, we rewrite (1) as follows:

$$A\_Happens(\mathtt{walk},t) \wedge \neg A\_Happens(\mathtt{stop},t+1) \supset E\_Happens(\mathtt{bump},t+2)$$

which reflects the fact that both walking and stopping are actions while the robot's bumping at the wall is a natural event. Suppose again that the robot walks towards the wall at time $t = 5$, which is now represented by the fact $A\_Happens(\mathtt{walk},5)$. Then our refined circumscription policy yields the intended conclusion that $\neg A\_Happens(\mathtt{stop},6)$ and, hence, $E\_Happens(\mathtt{bump},7)$. ■

[3] In order that the various foundational axioms which involve *Happens* need not be rewritten, the formula $Happens(e,t) \equiv A\_Happens(e,t) \vee E\_Happens(e,t)$ should be added after circumscribing, in N, the new predicates.

Example 2 (continued) We rewrite (2) as follows:

$$A\_Happens(\mathtt{lleft},t) \equiv \neg A\_Happens(\mathtt{lright},t) \supset E\_Happens(\mathtt{spills},t)$$

Suppose again that the left hand side is lifted at time $t = 2$, now represented by $A\_Happens(\mathtt{lleft},2)$. As above, our refined circumscription policy yields the intended conclusion that $\neg A\_Happens(\mathtt{lright},2)$ and, hence, $E\_Happens(\mathtt{spills},2)$. ■

3 INTERACTING EVENTS, REFLEXIVE ACTIONS, AND NORMATIVE RULES

Categorizing the underlying events as either actions or natural events provides a minimization strategy sophisticated enough for both our two example scenarios discussed in the introduction. As regards the general range of applicability, a closer examination reveals two fundamental assumptions which an application domain needs to satisfy in order for this strategy to guarantee the intended results. First, there must never be a priority between two natural events unless one of them, but not the other, is triggered by an action that is known to take place. Second, actions themselves need always be independent both of each other and of natural events. That is to say, an action is never caused by the performance of another action nor by the occurrence of a natural event or any other circumstances. As long as a domain complies with these assumptions, the distinction between volitional actions and natural events provides a suitable minimization policy. We again stress that the value of this strategy lies in its being amenable to straightforward integration into existing approaches.

On the other hand, the aforementioned basic assumptions are not of universal nature. Consider, as an example for interfering natural events, the three events $\mathtt{tile\_falls}$ (a tile falls from the roof), $\mathtt{tile\_hits\_plant}$ (the tile breaks a rare plant in the yard), and $\mathtt{tree\_falls}$ (a tree falls down). Suppose the falling tile hits the plant unless, curiously enough, the tree falls down at the very same time, crossing the trajectory of the tile:

$$E\_Happens(\mathtt{tile\_falls},t) \wedge \neg E\_Happens(\mathtt{tree\_falls},t) \supset E\_Happens(\mathtt{tile\_hits\_plant},t)$$

What should be concluded e.g. from the sole fact that $E\_Happens(\mathtt{tile\_falls},2)$? Obviously, the tile hitting the plant is much more likely in this case than the tree falling at the very time. Since, however, all three involved events are natural, categorization-based minimization does not arrive at this conclusion. Rather it produces an additional model where $\neg E\_Happens(\mathtt{tile\_hits\_plant},2)$ and, consequently, $E\_Happens(\mathtt{tree\_falls},2)$ hold. This illustrates that natural events may interfere without the explicit performance of any action.

Regarding the second requirement for the applicability of the categorization-based approach, consider, e.g., actions which are *reflexive*. These may well be the natural consequence of events. If, for example, we touch the working hot plate of a stove, then this causes us to involuntarily withdraw our hand. Nonetheless this withdrawing our hand cannot in general be considered a natural event because it may of course be a volitional act in other situations. The assumption that actions are always independent both of other actions and of events in general, does also not apply in case one intends to reason with *normative rules*, stating that an action ought to be performed in specific situations. The design of a robot's behavior might be given as set of such rules. Then again categorization-based minimization can no longer be guaranteed to distinguish action occurrences that are to be expected and those which a narrative does not necessitate.

4 EVENT MINIMIZATION AND CAUSALITY

Due to its formal elegance, categorization-based event minimization can be easily realized in existing frameworks. We have seen how this strategy solves the problem of unintended models whenever the distinction between actions and natural events is guaranteed to always help telling apart events whose occurrence admits a causal explanation. The spilling of the soup in Example 2, for instance, is a natural event *caused* by the lifting of the table on the left hand side, while the latter does not have the 'capability' of causing the volitional action of lifting up the right hand side of the table. On the other hand, in the preceding section we have met examples where our principle of categorization turned out too weak to identify event occurrences which are caused. Much like we have just argued, we can say that the tile's falling from the roof may cause the event that the plant breaks but cannot cause the simultaneous falling of the tree. And yet the two latter events both are natural, hence indistinguishable by the categorization principle. It therefore seems that a universal solution to the problem of unintended models resulting from event minimization needs to directly appeal to the notion of causality.

Fortunately, there is no need to start from scratch to this end. A ready approach is furnished by an existing solution to a related aspect of a different problem. Namely, the situation we arrived at shows a striking similarity to the problem of deriving undesired indirect effects in the context of the Ramification Problem [Ginsberg and Smith, 1988b]. Generally, indirect effects of actions, or of events, are not explicitly represented in some effect specification but follow from general laws, so-called *state constraints*, which describe certain state-independent relations among fluents.[4] A well-known challenge in this context is to select only the intended ones among all the potential indirect effects a state constraint suggests from a purely syntactic perspective. A categorization principle has been proposed as a solution to this problem, too, (see [Lifschitz, 1990]) and is widely used (e.g., [del Val and Shoham, 1993; Brewka and Hertzberg, 1993; Kartha and Lifschitz, 1994; Sandewall, 1995])—but it just as well turned out applicable to a certain extent only [Thielscher, 1997]. Recent research results on the Ramification Problem show how this limitations can be overcome by directly appealing to causality (e.g., [Elkan, 1992; Geffner and Pearl, 1992; Lin, 1995; McCain and Turner, 1995; Thielscher, 1997]). In the following, we exploit these results for the development of a causality-based solution to the problem of deriving the wrong event occurrences.

In order that this can be done, we first have to resolve an apparent fundamental difference between *events* on the one hand, and (indirect) effects—updates of value assignments to *fluents*—on the other hand. A small but conceptually crucial step achieves this: Event occurrences are identified as fluents. That is to say, each state of the world is also characterized by the events which currently happen, if any. By this we adapt the *actions-as-fluents* paradigm, which has been propagated, for instance, in [Lin and Shoham, 1992; Große, 1994; Thielscher, 1995]. To emphasize this shift, we will write event fluents using the symbol **happens** instead of *Happens*. Relations among events, like the one formalized in equation (2), can now be considered state constraints, which supposedly hold in all states, e.g.[5]

$$\mathtt{happens(lleft)} \equiv \neg\mathtt{happens(lright)} \supset \mathtt{happens(spills)}$$

Becoming state constraints, relations among event occurrences may give rise to indirect effects. If, for instance, **happens(lleft)** becomes true, then this additionally causes the indirect effect that **happens(spills)** according to the aforementioned constraint. Of course we need means to avoid the alternative conclusion that this constraint instead triggers the indirect effect **happens(lright)**. In this way the key problem of the present paper has become part of the Ramification Problem, and so is amenable to the existing causality-based techniques developed in that context. In particular, we can, and will, adopt our

[4] A fluent is an atomic property (of some object) which may change in the course of time.

[5] Relations among events which refer to different timepoints, like the one formalized in equation (1), cannot be reformulated solely by means of state constraints but by combining those with effect propositions (see below). Event occurrences triggered by such relations are then obtained as a combination of indirect and so-called delayed effects.

theory of causal relationships [Thielscher, 1997] and combine it with the events-as-fluents paradigm. The resulting theory is first presented as a formal, high-level narrative description language in the spirit of $\mathcal{A}$ [Gelfond and Lifschitz, 1993] or $\mathcal{E}$ [Kakas and Miller, 1997] etc. Thereafter, in Section 5, we show how this theory can be axiomatized on the basis of a novel variant of the Fluent Calculus.

Fluents are the only basic entity of domain descriptions in our language; events, and hence actions, are just a particular kind of fluent. As opposed to most 'ordinary' ones, event fluents have, however, a special characteristics: They are not subject to the commonsense law of persistence, which states that usually the value of a fluent 'tends to persist,' i.e., is stable unless an actual event effects a change. Event fluents are different in that they 'tend to disappear.' In this way event occurrences are always forced to have a cause. Following standard terminology, we call *momentary* all fluents of the latter kind, as proposed in [Lifschitz and Rabinov, 1989]. The other fluents we call *static*. Each momentary fluent has a designated default value, which it takes on unless, for an instant, something causes a different value.[6]

A *state* is composed of assignments of values to fluents. The atomic statement that a certain fluent f is of value v, written $f \hat{=} v$, is called a *fluent expression*. If f is a binary fluent with the domain $\{\mathtt{false}, \mathtt{true}\}$, then we abbreviate $f \hat{=} \mathtt{false}$ by $\neg f$ and $f \hat{=} \mathtt{true}$ simply by f. One more notation will be useful for later purpose: If S is a set of fluent assignments, then by $S|_{f \hat{=} v}$ we denote the set which is identical to S except for fluent f possessing value v.

Fluent expressions can be considered the underlying atoms for constructing *fluent formulas* using the standard logical connectives. The notion of fluent formulas being *true* in a state S is then based on defining a fluent expression $f \hat{=} v$ to be true if and only if $f \hat{=} v \in S$. *State constraints* are fluent formulas which have to be satisfied in all states that are possible in a domain.

Domain descriptions include propositions which specify the direct and indirect effects of events. The former are given by *effect propositions*, which indicate circumstances under which certain fluent changes are effected when moving on from one state to the next. An example is the effect proposition

$$\begin{array}{l} \mathtt{happens(spills)} \wedge \mathtt{stain} \hat{=} \mathtt{small} \\ \underline{\text{effects}}\ \mathtt{stain} \hat{=} \mathtt{large} \end{array}$$

meaning that if soup spills out with the tablecloth already being lightly stained, then the situation gets worse during the next state transition. Our language supports the specification of non-deterministic effects, as in

$$\begin{array}{l} \mathtt{happens(spills)} \wedge \mathtt{stain} \hat{=} \mathtt{none} \\ \underline{\text{effects}}\ \mathtt{stain} \hat{=} \mathtt{small} \,|\, \mathtt{stain} \hat{=} \mathtt{large} \end{array}$$

meaning that the spilling soup will produce either a small or large stain on a clean tablecloth.

Indirect effects of events are described by *causal relationships* [Thielscher, 1997], such as

$$\begin{array}{l} \mathtt{happens(lleft)}\ \underline{\text{causes}}\ \mathtt{happens(spills)} \\ \qquad\qquad\qquad\qquad \underline{\text{if}}\ \neg\mathtt{happens(lright)} \end{array}$$

which indicates that if $\mathtt{happens(lleft)}$ occurs as direct or indirect effect of a state transition, then this triggers the additional, indirect effect $\mathtt{happens(spills)}$, provided $\neg\mathtt{happens(lright)}$ holds. Notice a crucial difference between the specifications of direct and indirect effects: Cause and effect in effect propositions refer to two consecutive states, and to one and the same state in causal relationships.

Definition 1 A *domain description* consists of

1. a set $\mathcal{F} = \mathcal{F}_s \cup \mathcal{F}_m$ of *static* and *momentary fluents*, each of which is associated with a non-empty *domain*;
2. a set of *effect propositions* of the form

   $$C\ \underline{\text{effects}}\ E_1 \,|\, \ldots \,|\, E_n$$

   where the *condition* C is a fluent formula and the (alternative) *effects* E_i all are finite (possibly empty) sequences of fluent expressions $(n > 0)$;[7]
3. a set of *causal relationships* of the form

   $$\varepsilon\ \underline{\text{causes}}\ \varrho\ \underline{\text{if}}\ \Phi$$

   where ε, ϱ (the *effect* and the *ramification*, respectively) are fluent expressions and the *context* Φ is a fluent formula;
4. a set $\mathcal{C}$ of fluent formulas, called the *state constraints*.

A *state* is a set of assignments $f \hat{=} v$ such that to each $f \in \mathcal{F}$ is assigned a value of its domain. A state S *satisfies* $\mathcal{C}$ iff each constraint is true in S. ∎

Intuitively, an effect proposition C $\underline{\text{effects}}$ $E_1 | \ldots | E_n$ means that if C holds in some state, then exactly one of $E_1, \ldots, E_n$ materializes in the next state. Thus disjunction is interpreted as exclusive; inclusive disjunction can of course be obtained by simply moving from, say, two alternatives $e_1 \,|\, e_2$ to three, viz. $e_1 \,|\, e_2 \,|\, e_1, e_2$. The intuitive meaning of a causal relationship ε $\underline{\text{causes}}$ ϱ $\underline{\text{if}}$ Φ is that if ε occurs as (direct

[6] In what follows, we assume for the sake of uniformity that all momentary fluents be binary with the value domain $\{\mathtt{false}, \mathtt{true}\}$ and $\mathtt{false}$ being the default. This is just for the sake of clarity of presentation.

[7] If some E_i is empty, then one possibility for what happens if C holds is—nothing.

or indirect) effect and context Φ holds, then the indirect effect ϱ is additionally obtained. For causal relationships whose context is a logical tautology we use the short-hand form ε causes ϱ.[8]

Example 2 (continued) Consider the two static fluents $\mathcal{F}_s = \{\texttt{clock}, \texttt{stain}\}$ with the natural numbers and $\{\texttt{none}, \texttt{small}, \texttt{large}\}$ as the respective domain (where the latter indicates how badly the tablecloth is stained, if at all). Consider further the three momentary (hence binary) fluents $\mathcal{F}_m = \{\texttt{happens}(e) : e \in \{\texttt{lleft}, \texttt{lright}, \texttt{spills}\}\}$. Then the following components together constitute a domain description.

- The state constraint,

$$\begin{aligned} &\texttt{happens(lleft)} \equiv \neg\texttt{happens(lright)} \\ &\qquad \supset \texttt{happens(spills)} \end{aligned} \tag{3}$$

- The corresponding causal relationships,

$$\begin{aligned} \texttt{happens(lleft)}\ &\underline{\text{causes}}\ \texttt{happens(spills)} \\ &\qquad \underline{\text{if}}\ \neg\texttt{happens(lright)} \\ \texttt{happens(lright)}\ &\underline{\text{causes}}\ \texttt{happens(spills)} \\ &\qquad \underline{\text{if}}\ \neg\texttt{happens(lleft)} \end{aligned} \tag{4}$$

- The effect propositions,

$$\begin{aligned} &\texttt{happens(spills)} \wedge \texttt{stain}\hat{=}\texttt{none} \\ &\quad \underline{\text{effects}}\ \texttt{stain}\hat{=}\texttt{small} \mid \texttt{stain}\hat{=}\texttt{large} \\ &\texttt{happens(spills)} \wedge \neg\texttt{stain}\hat{=}\texttt{none} \\ &\quad \underline{\text{effects}}\ \texttt{stain}\hat{=}\texttt{large} \\ &\texttt{clock}\hat{=}t\ \underline{\text{effects}}\ \texttt{clock}\hat{=}t+1 \\ &\texttt{clock}\hat{=}1\ \underline{\text{effects}}\ \texttt{happens(lleft)} \end{aligned} \tag{5}$$

  The last but one effect proposition is a representative of all instances where t is a natural number. The very last proposition formalizes the initiation of a `lleft` action at time 2.

■

The effect proposition in (5) with condition $\texttt{clock}\hat{=}1$ illustrates how causal chains of events are initiated: Volitional actions or natural events which common sense prefers to consider causeless[9] are caused just by reaching the time-point at which they occur. Obviously, the fluent `clock` is crucial for this purpose, and so we assume it to be contained in any domain description. The time structure underlying our approach is left-bound, linear, and discrete; a challenge for future research is the generalization to continuous time and change.

Next we define the notion of successor states, which are obtained according to underlying effect laws and causal relationships. Since we allow for nondeterminism, a state may admit several possible successors. We begin by defining so-called *preliminary* successor states, in which all direct effects have been accounted for but which require further investigation to accommodate possible indirect effects.

Definition 2 Consider a domain description with static and momentary fluents $\mathcal{F}_s$ and $\mathcal{F}_m$, respectively; effect propositions $\mathcal{E}$; and state constraints $\mathcal{C}$. Let S be a state satisfying $\mathcal{C}$, then any state S' which obtains as follows is called a *preliminary successor of* S: For each C effects $E_1|\ldots|E_n \in \mathcal{E}$ such that C holds in S, select one E_i $(1 \leq i \leq n)$. Let E be the entire set of assignments thus obtained. Then S' consists of

1. all assignments in E,
2. all assignments $f\hat{=}\texttt{false}$ for $f \in \mathcal{F}_m$ such that E contains no assignment for f,
3. all assignments $f\hat{=}v \in S$ for $f \in \mathcal{F}_s$ such that E contains no assignment for f.

■

Put in words, the preliminary successors are obtained by first making a selection among the alternative outcomes of all applicable effect propositions and, then, by realizing all these effects, by setting all unaffected momentary fluents to their default value, and by letting all unaffected static fluents persist.[10]

Example 2 (continued) Consider this state:

$$\begin{aligned} S(1) = \{\, &\neg\texttt{happens(lleft)}, \neg\texttt{happens(lright)}, \\ &\texttt{happens(spills)}, \texttt{clock}\hat{=}1, \texttt{stain}\hat{=}\texttt{none}\,\} \end{aligned}$$

Of the effect propositions in (5) three are applicable,

[8] Causal relationships are typically rooted in state constraints but provide additional causal information. In [Thielscher, 1997] we have shown that it is not necessary to draw up causal relationships by hand. Rather these can be fully automatically extracted from a given set of state constraints plus suitable knowledge as to which fluents have the potential to causally affect what other fluents. Later on we will discuss this point in greater detail.

[9] While the universal causal law stipulates that everything has a cause, common sense always considers only a fraction of the whole universe and, hence, only a fraction of the entire history of events. Anything whose cause lies outside this fraction is considered causeless.

[10] It is noteworthy that in case of contradictory effect propositions the resulting set of assignments may be inconsistent in that it contains some $f\hat{=}v_1$ together with $f\hat{=}v_2$ such that $v_1 \neq v_2$. Then this set is not a state, hence does not constitute a preliminary successor according to the definition. States therefore may admit no preliminary and also no successor states at all.

namely,

$$\begin{array}{l} \texttt{happens(spills)} \wedge \texttt{stain}\hat{=}\texttt{none} \\ \quad \underline{\text{effects}}\ \texttt{stain}\hat{=}\texttt{small} \,|\, \texttt{stain}\hat{=}\texttt{large} \\ \texttt{clock}\hat{=}1\ \underline{\text{effects}}\ \texttt{clock}\hat{=}2 \\ \texttt{clock}\hat{=}1\ \underline{\text{effects}}\ \texttt{happens(lleft)} \end{array}$$

The topmost proposition being indeterminate, we obtain two preliminary successor states, viz.

$$\begin{array}{l} \{\, \texttt{happens(lleft)},\ \neg\texttt{happens(lright)}, \\ \quad \neg\texttt{happens(spills)},\ \texttt{clock}\hat{=}2,\ \texttt{stain}\hat{=}\texttt{small} \,\} \\ \{\, \texttt{happens(lleft)},\ \neg\texttt{happens(lright)}, \\ \quad \neg\texttt{happens(spills)},\ \texttt{clock}\hat{=}2,\ \texttt{stain}\hat{=}\texttt{large} \,\} \end{array}$$

Notice how the momentary fluent happens(spills) is automatically set to its default value, false. Notice further that neither of the two preliminary successors satisfies the state constraint (3), for we have not yet considered the possibility of indirect effects. ■

In order to account for indirect effects, preliminary successors are taken as starting points for 'causal propagation,' that is the successive application of causal relationships until overall satisfactory successor states obtain [Thielscher, 1997]. Formally, causal relationships operate on pairs (S, E), where S denotes an intermediate state, in which some but not yet all indirect effects have been realized, and where E contains all direct and indirect effects computed so far:[11]

Definition 3 Consider a pair (S, E) consisting of a state S and a set of fluent expressions E. A causal relationship ε $\underline{\text{causes}}$ ϱ $\underline{\text{if}}$ Φ is *applicable* to (S, E) iff $\Phi \wedge \neg\varrho$ is true in S and $\varepsilon \in E$. Its *application* yields the pair $(S|_\varrho, E|_\varrho)$. ■

That is to say, a causal relationship is applicable if the associated condition Φ holds, the particular indirect effect ϱ is currently false, and its cause ε is among the current effects. If $\mathcal{R}$ is a set of causal relationships, then by $(S, E) \stackrel{*}{\rightsquigarrow}_{\mathcal{R}} (S', E')$ we indicate that there is a (possibly empty) sequence of elements of $\mathcal{R}$ so that the successive application to (S, E) results in (S', E'). It is easy to verify that if S is a state and E is consistent (i.e., contains no double assignments to fluents), then $(S, E) \stackrel{*}{\rightsquigarrow}_{\mathcal{R}} (S', E')$ implies that S' is a state and E' is consistent, too.

Now suppose given a set of fluent expressions S as the result of having accounted for all direct effects E via the given effect propositions. This preliminary successor S may violate the state constraints. Additional, indirect effects are then accommodated by (non-deterministically) selecting and (serially) applying causal relationships until a state satisfying all state constraints obtains.

[11] For a clarification of the crucial role of the second component, E, as well as for further details we suggest to consult [Thielscher, 1997].

Definition 4 Consider a domain description with causal relationships $\mathcal{R}$ and state constraints $\mathcal{C}$. Furthermore, let S be a state satisfying $\mathcal{C}$. A state T is a *possible successor state* of S iff there exists a preliminary successor S' obtained through direct effects E and such that

1. $(S', E) \stackrel{*}{\rightsquigarrow}_{\mathcal{R}} (T, E')$ for some E', and
2. T satisfies $\mathcal{C}$. ■

Example 2 (continued) Starting off from the two preliminary successors of our state $S(1)$ from above, on the basis of the causal relationships and state constraints (4) and (3), respectively, we obtain two possible successor states, viz.

$$\begin{array}{l} \{\, \texttt{happens(lleft)},\ \neg\texttt{happens(lright)}, \\ \quad \texttt{happens(spills)},\ \texttt{clock}\hat{=}2,\ \texttt{stain}\hat{=}\texttt{small} \,\} \\ \{\, \texttt{happens(lleft)},\ \neg\texttt{happens(lright)}, \\ \quad \texttt{happens(spills)},\ \texttt{clock}\hat{=}2,\ \texttt{stain}\hat{=}\texttt{large} \,\} \end{array}$$

Both these two successor states are obtained by application of the causal relationship

$$\begin{array}{l} \texttt{happens(lleft)}\ \underline{\text{causes}}\ \texttt{happens(spills)} \\ \qquad\qquad\qquad\qquad \underline{\text{if}}\ \neg\texttt{happens(lright)} \end{array}$$

It applies as $\neg\texttt{happens(lright)} \wedge \neg\texttt{happens(spills)}$ holds in the respective preliminary successor state, and on account of happens(lleft) being among the direct effects. ■

Obtaining the intended result by applying causal relationships to accommodate indirect effects depends, to state the obvious, on a suitable set of these relationships. This set should be complete in that it covers all indirect effects that reasonably follow from the state constraints, and, in particular, it should be sound in that it does not sanction indirect effects which do not follow from the standpoint of causality. Our two causal relationships of equation (4), for instance, constitute such a suitable set. Notice, however, that from a purely syntactical point of view state constraint (3) suggests additional, unintended causal relationships, such as

$$\begin{array}{l} \texttt{happens(lleft)}\ \underline{\text{causes}}\ \texttt{happens(lright)} \\ \qquad\qquad\qquad\qquad \underline{\text{if}}\ \neg\texttt{happens(spills)} \end{array}$$

Precisely this is the motivation for employing causal relationships, which convey more information than the mere state constraints. In [Thielscher, 1997] we have argued that causal relationships need not be drawn up all by hand but can be automatically generated on the basis of additional domain knowledge as to potential causal influence of some fluents upon others. In its simplest form, this knowledge is formally provided by a binary relation $\mathcal{I}$ on fluents, called *influence information*. If $(f_1, f_2) \in \mathcal{I}$, then this is intended to denote that a change of f_1's value potentially causally affects the value of f_2. In our running example, the suitable influence information is to let $\mathcal{I}$ consist of the two elements $(\texttt{happens(lleft)}, \texttt{happens(spills)})$ and $(\texttt{happens(lright)}, \texttt{happens(spills)})$. That is to say, the events `lleft` and `lright` may causally affect the event `spills` but not vice versa, nor do they mutually interfere. If applied to the state constraint of equation (3), the two causal relationships (4) are obtained. For further details we refer the reader to [Thielscher, 1997].[12]

The formal definition of successor states completes the crucial part of our narrative description language. What remains to be done is to introduce *observation* statements and, then, to give a precise notion of models, which are *histories*, and of entailment. Reflecting the intention to never consider event occurrences without a cause, we require that each momentary fluent takes on its default value at the initiation of a history. In so doing we employ the general principle of initial minimization (see, e.g., [Shanahan, 1995b; Thielscher, 1996]). It is further assumed that the clock always shows the right time.

Definition 5 A *formal narrative* is a domain description augmented by a set of *observations*, which are expressions of the form $[t]\,F$ where t is a timepoint and F a fluent formula. A *history* is an infinite sequence of states $S(0), S(1), S(2), \ldots$ Such a history is a *model* of a formal narrative iff

1. each momentary fluent is false in $S(0)$,
2. each $S(t+1)$ is a successor state of $S(t)$ $(t \geq 0)$,
3. $\texttt{clock}\hat{=}t \in S(t)$ for each $t \geq 0$, and
4. for each observation $[t]\,F$, fluent formula F is true in state $S(t)$.

An observation is *entailed* iff it holds in all models.

Example 2 (continued) Let us add to our example domain description the effect proposition

$$\texttt{clock}\hat{=}0 \ \underline{\text{effects}} \ \texttt{happens(spills)} \tag{6}$$

Consider the narrative consisting of the resulting domain description plus the observation that initially the tablecloth is not badly stained, i.e.,

$$[0]\,\neg\texttt{stain}\hat{=}\texttt{large} \tag{7}$$

Then any model must have initial state

$$\begin{aligned} S(0) = \{\, & \neg\texttt{happens(lleft)}, \neg\texttt{happens(lright)}, \\ & \neg\texttt{happens(spills)}, \texttt{clock}\hat{=}0, \texttt{stain}\hat{=}v \} \end{aligned} \tag{8}$$

where v is either `none` or `small`. Following (6), a `spills` event occurs at time $t = 1$. According to the topmost two effect propositions in (5), this event in turn produces a small or a large stain in case $\texttt{stain}\hat{=}\texttt{none} \in S(0)$, and a large stain in case $\texttt{stain}\hat{=}\texttt{small} \in S(0)$. Hence the narrative entails

$$[2]\,\texttt{stain}\hat{=}\texttt{small} \vee \texttt{stain}\hat{=}\texttt{large} \tag{9}$$

Now, according to the bottommost effect proposition in (5), a `lleft` event occurs at the same time, which, as we have seen, has the indirect effect of yet another `spills` event. Therefore, knowing that (9) holds in all models, we see that our narrative also entails

$$[3]\,\texttt{stain}\hat{=}\texttt{large}$$

■

As an important feature our notion of entailment supports explanatory reasoning, i.e., reasoning backwards in time, as far as incomplete knowledge of static fluents is concerned. For instance, if the observation $[2]\,\texttt{stain}\hat{=}\texttt{small}$ were added to the example narrative from above, then it is easy to see that the observation $[0]\,\texttt{stain}\hat{=}\texttt{none}$ would be entailed.

What cannot be plainly derived from a narrative are events which are not caused by what is known to happen. For instance, if the observation $[2]\,\neg\texttt{happens(spills)}$ were added to our example narrative, then the latter would admit no models at all because the observation cannot be explained without granting a new causeless event. Additional means are needed to abduce a suitable explanation, e.g., that a `lright` event occurs at time $t = 2$ in addition to the `lleft` event.

This property of our high-level language and semantics carries over to the axiomatization to be presented in the next section. If it concerns values of static fluents, an explanation for observed facts will be deductively derivable. Abduction will be required to explain observations by uncaused event happenings.

5 A FLUENT CALCULUS AXIOMATIZATION

Having presented a high-level language for describing and reasoning about narratives, we will now illustrate a way of axiomatizing narratives described in this language so that our specific notion of entailment becomes entailment in classical logic and, hence, the reasoning can be carried out by fully automated deduction. Just like our narrative description language does it, the resulting axiomatization successfully copes with the problem of causally connected events. We restrict ourselves to deterministic domains, i.e., where all states admit a unique successor state. Due to lack of space, we illustrate the axiomatization

[12] The method described in [Thielscher, 1997] is originally defined only for state constraints over two-valued fluents, but the generalization is straightforward. An unsatisfactory property is that this method may yield different sets of causal relationships for semantically equivalent state constraints. Following a suggestion by Javier Pinto, independence of syntax is achieved by processing the prime implicants of a set of constraints.

merely by example. General correctness wrt. the notion of entailment in our high-level language is proved in [Thielscher, 1998a].

Our axiomatization is based on a novel use of Fluent Calculus, which was introduced in [Hölldobler and Schneeberger, 1990] and so christened in [Bornscheuer and Thielscher, 1997]. While historically the Fluent Calculus arose from approaches to the Frame Problem using non-classical, linear logics, in [Thielscher, 1998b] we argue that it can alternatively be viewed as a development of the Situation Calculus in order to cope with both the representational and the inferential aspect of the Frame Problem, without leaving classical logic. The key to this new Fluent Calculus is to reformulate successor state axioms [Reiter, 1991] applying the principle of reification, which means to use terms instead of atoms as the formal denotation of statements. To be more specific, reification in the Fluent Calculus means not only to denote single fluent-value assignments $f \hat{=} v$ as terms (which we will write as $\langle f, v\rangle$), but also conjunctions of them. Required to this end is a binary function, denoted by the symbol "$\circ$" and written in infix notation, by which conjunction is reified. The great advantage of so doing is that term variables can occur in state descriptions to indicate incomplete knowledge of the state at hand, as in, e.g., the specification

$$\exists z, v\, [\, S_0 = \langle \mathtt{clock}, 0\rangle \circ \langle \mathtt{stain}, v\rangle \circ z \;\wedge\; v \neq \mathtt{large}\,]$$

which says that of state S_0 it is merely known that $\mathtt{clock} \hat{=} 0$ holds and that $\mathtt{stain} \hat{=} \mathtt{large}$ is false.

Central to the Fluent Calculus variant of [Thielscher, 1998b] is a function $State(s)$ which assigns to each situation a state term. This allows to rewrite successor state axioms to so-called *state update axioms*. These are of the form $\Delta[s] \supset \Gamma[State(Do(a,s)), State(s)]$, where $\Delta[s]$ describes the conditions on situation s under which the state associated with s is updated according to Γ to become the state associated with situation $Do(a,s)$.

A most interesting feature of this new Fluent Calculus is that by a slight modification it can be adapted from its branching time structure, which is typical for the Situation Calculus, to linear time, which brings it closer to the Event Calculus. The basic idea is to let the function $State$ range over time-points instead of situations. For instance, the following is an assertion about the initial state by which is axiomatized observation (7) of our example narrative at the end of the preceding section:

$$\exists z, v\, [\, State(0) = \langle \mathtt{stain}, v\rangle \circ z \;\wedge\; v \neq \mathtt{large}\,]$$

The binary function which reifies the logical conjunction needs to inherit from the latter an important property. In logical conjunctions the order is irrelevant in which the elements are given. Formally, order ignorance is ensured by stipulating the laws of associativity and commutativity, that is,

$$\begin{aligned} \forall x, y, z.\ (x \circ y) \circ z &= x \circ (y \circ z) \\ \forall x, y.\qquad x \circ y &= y \circ x \end{aligned}$$

It is convenient to also reify the empty conjunction, a logical tautology, by a constant denoted $\emptyset$ and which satisfies

$$\forall x.\ x \circ \emptyset = x$$

The three equational axioms, jointly abbreviated AC1, in conjunction with the standard axioms of equality entail the equivalence of two state terms whenever they are built up from an identical collection of reified fluents.[13]

A new feature required for our solution to the event minimization problem is the notion of momentary fluents. These are declared using a unary predicate $Momentary(f)$ in conjunction with a binary predicate $DefaultValue(f,v)$ determining the default value v of fluent f. A third predicate, $InDomain(f,v)$, is used to specify domains for the fluents. For our running example, adequate definitions of these three predicates are the following:[14]

$$\begin{aligned} Momentary(f) &\equiv f = \mathtt{happens(e)} \\ DefaultValue(f,v) &\equiv f = \mathtt{happens(e)} \wedge v = \mathtt{false} \\ InDomain(f,v)& \\ \equiv\ & f = \mathtt{happens(e)} \wedge (v = \mathtt{true} \vee v = \mathtt{false}) \\ &\vee f = \mathtt{clock} \wedge \exists t.\ v = t \\ &\vee f = \mathtt{stain} \wedge \\ &\quad (v = \mathtt{none} \vee v = \mathtt{small} \vee v = \mathtt{large}) \end{aligned}$$

The above being domain-dependent axioms, we now introduce two foundational axioms which define the space of possible states. First, in a state to each fluent must be assigned a value of its domain:

$$\begin{aligned} &\exists v.\ InDomain(f,v) \\ &\quad \supset \exists z, v'\, [\, State(t) = \langle f, v'\rangle \circ z \;\wedge\; InDomain(f,v')\,] \end{aligned}$$

Second, no two values shall be assigned to the same fluent:

$$State(t) \neq \langle f, v\rangle \circ \langle f, v'\rangle \circ z$$

In order to increase readability of statements about states, we introduce a predicate $Holds(f,v,s)$ as an

[13] The reader may wonder why function $\circ$ is not expected to be idempotent, i.e., $\forall x.\ x \circ x = x$, which is yet another property of logical conjunction. The (subtle) reason for this is given below.

[14] In what follows, variables will be denoted by (possibly primed) lower-case letters. The particular variable t shall range over the natural numbers, including 0. Free variables in formulas are assumed universally quantified. For the sake of readability, we will furthermore use a variable $\mathtt{e}$ which can be replaced by either of our three events **lleft**, **lright**, or **spills**.

abbreviation, meaning that fluent f has value v in state s:

$$Holds(f, v, s) \equiv \exists z.\, s = \langle f, v \rangle \circ z$$

A simple use of this macro is to ascertain that time and the value of the fluent `clock` coincide:

$$Holds(\mathtt{clock}, t, State(t))$$

Effect propositions are axiomatized by implications of the form $\Delta[s] \supset Effects(f, v, v', s)$, where $\Delta[s]$ is a specification of the conditions that must be satisfied in state s in order that fluent f changes its value from v to v'. For example, the three effect propositions

$$\begin{array}{rcl} \mathtt{happens(spills)} & \underline{\text{effects}} & \mathtt{stain}\hat{=}\mathtt{large} \\ \mathtt{clock}\hat{=}t & \underline{\text{effects}} & \mathtt{clock}\hat{=}t+1 \\ \mathtt{clock}\hat{=}1 & \underline{\text{effects}} & \mathtt{happens(lleft)} \end{array}$$

are axiomatized as follows:

$$\begin{gathered} Holds(\mathtt{happens(spills)}, \mathtt{true}, s) \wedge Holds(\mathtt{stain}, v, s) \\ \supset Effects(\mathtt{stain}, v, \mathtt{large}, s) \end{gathered}$$

$$Holds(\mathtt{clock}, t, s) \supset Effects(\mathtt{clock}, t, t+1, s)$$

$$\begin{gathered} Holds(\mathtt{clock}, 1, s) \wedge Holds(\mathtt{happens(lleft)}, v, s) \\ \supset Effects(\mathtt{happens(lleft)}, v, \mathtt{true}, s) \end{gathered}$$

In addition, a foundational axiom ensures that each momentary fluent changes to its default value if currently it enjoys another value and if no effect proposition implies its getting a non-default value:

$$\begin{gathered} Momentary(f) \wedge DefaultValue(f, v) \wedge Holds(f, v', s) \\ \wedge\ v \neq v' \wedge \neg(Effects(f, v', v'', s) \wedge v \neq v'') \\ \supset Effects(f, v', v, s) \end{gathered}$$

The effect propositions are assumed to constitute a complete description of what changes when moving from one state to the next. In order to reflect this assumption, we circumscribe the predicate *Effects*, and so solve the representational aspect of the Frame Problem. The crucial next step is to solve the inferential aspect, too.

Taken together, the single effects can be viewed as a sound and complete set of constraints on two terms τ and ι in which are conjoined, via "$\circ$", the assignments that terminate to hold and the assignments that are initiated, respectively. These terms are then employed for updating the current state so as to arrive at the successor. Updating means that all unaffected fluent-value pairs remain untouched in the new state. Thus the inferential Frame Problem gets solved. The definition for τ and ι is as follows:

$$Change(t, \tau, \iota) \equiv \left[\begin{array}{l} Effects(f, v, v', State(t)) \\ \quad \equiv \exists z, z' \left[\begin{array}{lcl} \tau & = & \langle f, v \rangle \circ z\ \wedge \\ \iota & = & \langle f, v' \rangle \circ z' \end{array} \right] \end{array} \right]$$

If only direct effects were to be considered, then the concept of successor states could now be modeled by the elegant schematic implication $State(t) = \tau \circ z \supset State(t{+}1) = z \circ \iota$. Incidentally, this scheme is the reason for not stipulating that $\circ$ be idempotent. For if it were, then, given that $State(t)$ includes the fluent-value assignments τ, the equation $State(t) = \tau \circ z$ would be satisfied if z is substituted by $State(t)$. Hence equating $State(t+1)$ with $z \circ \iota$ would not guarantee that all assignments in τ terminate.

In addition to the direct effects, ramifications need to be accounted for. The definition of how to obtain successor states therefore employs the predicate $Ramify(s, e, s')$ as introduced in [Thielscher, 1997], which is meant to be true if the successive application of causal relationships to (S, E) eventually results in a pair whose first component, S', satisfies the domain constraints—where s, e, and s' are reifications of S, E, and S', respectively. With this predicate, to be defined below, the following foundational axiom models the definition of successor states according to Definition 4:

$$\begin{gathered} Change(t, \tau, \iota) \wedge State(t) = \tau \circ z \wedge Ramify(z \circ \iota, \iota, s) \\ \supset State(t+1) = s \end{gathered}$$

The definition of *Ramify* can be directly adopted from [Thielscher, 1997] as being the transitive closure of a predicate $Causes(s, e, s', e')$, which in turn is meant to be true iff there is an instance of a causal relationship which is applicable to (S, E) and whose application yields (S', E')—where s, e, s', e' are reifications of S, E, S', E'. The causal relationships in our example domain, c.f. equation (4), are thus suitably axiomatized as follows:

$$\begin{array}{lll} \multicolumn{3}{l}{Causes(s, e, s', e')} \\ \equiv & & \exists x.\ e = \langle \mathtt{happens(lleft)}, \mathtt{true} \rangle \circ x \\ & & \wedge\ Holds(\mathtt{happens(lright)}, \mathtt{false}, s) \\ & & \wedge\ s = \langle \mathtt{happens(spills)}, \mathtt{false} \rangle \circ y \\ & & \wedge\ s' = y \circ \langle \mathtt{happens(spills)}, \mathtt{true} \rangle \\ & & \wedge\ e' = e \circ \langle \mathtt{happens(spills)}, \mathtt{true} \rangle \\ & \vee & \exists x.\ e = \langle \mathtt{happens(lright)}, \mathtt{true} \rangle \circ x \\ & & \wedge\ Holds(\mathtt{happens(lleft)}, \mathtt{false}, s) \\ & & \wedge\ s = \langle \mathtt{happens(spills)}, \mathtt{false} \rangle \circ y \\ & & \wedge\ s' = y \circ \langle \mathtt{happens(spills)}, \mathtt{true} \rangle \\ & & \wedge\ e' = e \circ \langle \mathtt{happens(spills)}, \mathtt{true} \rangle \end{array}$$

Transitive closure cannot be expressed in first-order logic, which is why predicate *Ramify* is defined using the standard way of encoding transitive closure by a second-order formula:

$$\begin{array}{l} Ramify(s, e, s') \equiv \\ \quad Possible(s') \wedge \forall \Pi. \\ \left\{ \begin{array}{c} \forall s_1, e_1.\ \Pi(s_1, e_1, s_1, e_1) \\ \wedge \\ \left[\begin{array}{l} \forall s_1, e_1, s_2, e_2, s_3, e_3 \\ \quad [\Pi(s_1, e_1, s_2, e_2) \wedge Causes(s_2, e_2, s_3, e_3) \\ \qquad \supset \Pi(s_1, e_1, s_3, e_3)] \end{array} \right] \\ \supset \\ \Pi(s, e, s', e') \end{array} \right\} \end{array}$$

That is, $Ramify(s, e, s')$ is true iff s' satisfies the state constraints and there is some e' such that (s, e, s', e') belongs to the transitive closure of *Causes*.

What remains to be axiomatized are the state constraints of a domain description, which in our example looks as follows:

$$Possible(s) \equiv \left\{ \begin{array}{l} \left[\begin{array}{l} Holds(\texttt{happens}(\texttt{lleft}), \texttt{true}, s) \\ \equiv Holds(\texttt{happens}(\texttt{lright}), \texttt{false}, s) \end{array} \right] \\ \quad \supset Holds(\texttt{happens}(\texttt{spills}), \texttt{true}, s) \end{array} \right\}$$

This completes our axiomatization. We refer to [Thielscher, 1998a] for a proof of general correctness—wrt. our narrative description language—of axiomatizations based on the ideas illustrated here. Let us just mention the following specific result:

Theorem 6 *Let Σ be the conjunction of all Fluent Calculus formulas above. Then $CIRC[\Sigma; Effects]$ entails*

$$Holds(\texttt{stain}, \texttt{large}, State(2))$$

6 CONCLUSION

Coming from the observation that straightforwardly minimizing events in narratives is insufficient in case of causal dependencies among events, we have proposed two refined minimization strategies. The first of which exploits the distinction between two categories of events, namely, volitional actions vs. natural events. The range of applicability of this approach has been discussed, and we have then developed a general solution which helps telling apart caused event occurrences by directly appealing to the notion of causality.

Our first, categorization-based minimization strategy is particularly appealing because it can be easily integrated into existing approaches, which we have illustrated with the Event Calculus axiomatization of [Shanahan, 1996]. For our second, more general strategy we have identified event occurrences with fluents, which then has allowed us to apply an existing causality-based solution to the Ramification Problem. We have presented a high-level narrative description language and a novel Fluent Calculus axiomatization in which is realized this solution to the event minimization problem.

Regarding related work, we first note that arguments in favor of the theory of causal relationships as a solution to the Ramification Problem, and a through comparison to other approaches, can be found in the article [Thielscher, 1997]. In particular, there we have argued that excluding uncaused indirect effects, and not minimizing change, is the real issue of dealing with ramifications. This is especially crucial in case of so-called stabilizing state constraints [Thielscher, 1998c]. As for comparisons with existing event minimization strategies, it has already been mentioned that *chronological minimization* [Shoham, 1988] is not applicable to specifications like (2), i.e., which involve concurrent events. This argument applies to *Motivated Action Theory* [Stein and Morgenstern, 1994], too.

Our novel Fluent Calculus appears to be related in several interesting respects to the Event Calculus, in particular to the variant of [Shanahan, 1995a], where an explicit notion of state is used. A detailed comparison of the two is an important aspect of ongoing research.

Finally, we should stress that reasonably minimizing events is of great importance not only for reasoning about narratives, where it is a mere convention that events do not happen unless they follow from what has been said. It is also essential for setting up plans: Suppose a planning goal be to produce a stain in the tablecloth. A reasonable plan would be, say, to lift up the left hand side of the table on which the bowl of soup is located. Yet this plan can be concluded successful only under the assumption that no intervening agent lifts the right hand side simultaneously. Generally, it is difficult if not impossible to devise plans that are perfectly reliable in reality. Assuming away disturbing events for which there is no indication that they will occur, is therefore the only way to come up with plans which at least by default can be concluded to achieve a goal. Of course things may not turn out as expected when a plan is being executed. Agents therefore need to constantly update, e.g. by abduction, their knowledge about the actual course of events.

References

[Bornscheuer and Thielscher, 1997] S.-E. Bornscheuer and M. Thielscher. Explicit and implicit indeterminism: Reasoning about uncertain and contradictory specifications of dynamic systems. *J. of Logic Programming*, 31(1–3):119–155, 1997.

[Brewka and Hertzberg, 1993] G. Brewka and J. Hertzberg. How to do things with worlds: On formalizing actions and plans. *J. of Logic and Computation*, 3(5):517–532, 1993.

[del Val and Shoham, 1993] A. del Val and Y. Shoham. Deriving properties of belief update from theories of action (II). In R. Bajcsy, editor, *Proceedings of the IJCAI*, pages 732–737, Chambéry, France, August 1993. Morgan Kaufmann.

[Elkan, 1992] C. Elkan. Reasoning about action in first-order logic. In *Proceedings of the Conference of the Canadian Society for Computational Studies of Intelligence (CSCSI)*, pages 221–227, Vancouver, Canada, May 1992. Morgan Kaufmann.

[Geffner and Pearl, 1992] H. Geffner and J. Pearl. Conditional entailment: bridging two approaches to default reasoning. *Artificial Intelligence*, 23:209–244, 1992.

[Gelfond and Lifschitz, 1993] M. Gelfond and V. Lifschitz. Representing action and change by logic programs. *J. of Logic Programming*, 17:301–321, 1993.

[Ginsberg and Smith, 1988a] M. L. Ginsberg and D. E. Smith. Reasoning about action I: A possible worlds approach. *Artificial Intelligence*, 35:165–195, 1988.

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

[Große, 1994] G. Große. Propositional State-Event Logic. In C. MacNish, D. Peirce, and L. M. Peireira, editors, *Proceedings of the European Workshop on Logics in AI (JELIA)*, volume 838, pages 316–331. Springer, September 1994.

[Hölldobler and Schneeberger, 1990] S. Hölldobler and J. Schneeberger. A new deductive approach to planning. *New Generation Computing*, 8:225–244, 1990.

[Kakas and Miller, 1997] A. Kakas and R. Miller. A simple declarative language for describing narratives with actions. *J. of Logic Programming*, 31(1–3):157–200, 1997.

[Kartha and Lifschitz, 1994] G. N. Kartha and V. Lifschitz. Actions with indirect effects. In J. Doyle, E. Sandewall, and P. Torasso, editors, *Proceedings of the KR*, pages 341–350, Bonn, Germany, May 1994. Morgan Kaufmann.

[Lifschitz and Rabinov, 1989] V. Lifschitz and A. Rabinov. Things that change by themselves. In *Proceedings of the IJCAI*, pages 864–867, Detroit, MI, 1989.

[Lifschitz, 1987] V. Lifschitz. Pointwise circumscription. In M. L. Ginsberg, editor, *Readings in Nonmonotonic Reasoning*, chapter 3.3, pages 179–193. Morgan Kaufmann, 1987.

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

[Lin and Shoham, 1992] F. Lin and Y. Shoham. Concurrent actions in the situation calculus. In *Proceedings of the AAAI*, pages 590–595, San Jose, CA, 1992. MIT Press.

[Lin, 1995] F. Lin. Embracing causality in specifying the indirect effects of actions. In C. S. Mellish, editor, *Proceedings of the IJCAI*, pages 1985–1991, Montreal, Canada, August 1995. Morgan Kaufmann.

[McCain and Turner, 1995] N. McCain and H. Turner. A causal theory of ramifications and qalifications. In C. S. Mellish, editor, *Proceedings of the IJCAI*, pages 1978–1984, Montreal, Canada, August 1995. Morgan Kaufmann.

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

[Sandewall, 1995] E. Sandewall. Reasoning about actions and change with ramification. In *Computer Science Today*, volume 1000 of *LNCS*. Springer, 1995.

[Shanahan, 1995a] M. Shanahan. A circumscriptive calculus of events. *Artificial Intelligence*, 77:249–284, 1995.

[Shanahan, 1995b] M. Shanahan. Default reasoning about spatial occupancy. *Artificial Intelligence*, 74:147–163, 1995.

[Shanahan, 1996] M. Shanahan. Robotics and the common sense informatic situation. In W. Wahlster, editor, *Proceedings of the ECAI*, pages 684–688, Budapest, Hungary, August 1996. John Wiley.

[Shoham, 1988] Y. Shoham. Chronological ignorance: Experiments in nonmonotonic temporal reasoning. *Artificial Intelligence*, 36:279–331, 1988.

[Stein and Morgenstern, 1994] L. A. Stein and L. Morgenstern. Motivated action theory: A formal theory of causal reasoning. *Artificial Intelligence*, 71:1–42, 1994.

[Thielscher, 1995] M. Thielscher. The logic of dynamic systems. In C. S. Mellish, editor, *Proceedings of the IJCAI*, pages 1956–1962, Montreal, Canada, August 1995. Morgan Kaufmann.

[Thielscher, 1996] M. Thielscher. Causality and the qualification problem. In L. C. Aiello, J. Doyle, and S. Shapiro, editors, *Proceedings of the KR*, pages 51–62, Cambridge, MA, November 1996. Morgan Kaufmann.

[Thielscher, 1997] M. Thielscher. Ramification and causality. *Artificial Intelligence*, 89(1–2):317–364, 1997.

[Thielscher, 1998a] M. Thielscher. Event Minimization with the Fluent Calculus. Technical Report TUD/FI98-03, Department of Computer Science, Dresden University of Technology, March 1998. Electronically available at `www.intellektik.informatik.tu-darmstadt.de/~mit`.

[Thielscher, 1998b] M. Thielscher. From Situation Calculus to Fluent Calculus: A New Motivation. Manuscript, 1998. Electronically available at `www.intellektik.informatik.tu-darmstadt.de/~mit`.

[Thielscher, 1998c] M. Thielscher. Steady versus stabilizing state constraints. In *Proceedings of the Symposium on Logical Formalizations of Commonsense Reasoning*, pages 239–248, London, UK, January 1998.