*

INTERNATIONAL COMPUTER SCIENCE INSTITUTE

1947 Center St. • Suite 600 • Berkeley, California 94704-1198 • (510) 643-9153 • FAX (510) 643-7684

Qualification and Causality

Michael Thielscher

TR-96-026

July 1996

Abstract

In formal theories for reasoning about actions, the qualification problem denotes the problem to account for the many conditions which, albeit being unlikely to occur, may prevent the successful execution of an action. While a solution to this problem must involve the ability to assume away by default these *abnormal* disqualifications of actions, the common straightforward approach of globally minimizing them is inadequate as it lacks an appropriate notion of causality. This is shown by a simple counter-example closely related to the well-known Yale Shooting scenario. To overcome this difficulty, we propose to incorporate causality by treating the fact that an action is qualified as ordinary fluent, i.e., a proposition which may change its truth value in the course of time by potentially being (indirectly) affected by the execution of actions. Abnormal disqualifications then are *initially* assumed away, unless there is evidence to the contrary. Our formal account of the qualification problem includes the proliferation of explanations for surprising disqualifications and also accommodates so-called miraculous disqualifications, which go beyond the agent's explanation capacity. In the second part, we develop a fluent calculus-based encoding of domains that require a proper treatment of abnormal disqualifications. In particular, default rules are employed to account for the intrinsic nonmonotonicity of the qualification problem. The resulting action calculus is proved correct wrt. our formal characterization of the qualification problem.

A short version will be presented at the *Fifth International Conference on Principles of Knowledge Representation and Reasoning (KR'96)*, Camebridge, MA, Nov. 5–8, 1996.

[*] On leave from FG Intellektik, TH Darmstadt

1 Introduction

A fundamental requirement for autonomous intelligent agents is the ability to reason about causality, which enables the agent to understand the world to an extent sufficient for acting intelligently on the basis of his or her knowledge as to the effects of actions. The first formal approach in AI research to model this ability has been suggested in [McCarthy, 1959], where agents are proposed to infer, on the basis of general causal knowledge and by means of deduction, the impact of the execution of an action sequence in a particular situation.

It was again McCarthy who two decades later pointed out a key problem in this context which occurs whenever agents need to reason about actions in other than artificial environments, where complete knowledge of all relevant facts cannot be assumed: The *qualification problem* [McCarthy, 1977] arises from the fact that generally the successful execution of actions depends on many more conditions than we are usually aware of. The reason for this unawareness is that most conditions are so likely to be satisfied that they are assumed away in case there is no evidence to the contrary.

A standard example to illustrate this is when we intend to start our car's engine, then we usually do not make sure that no potato in the tail pipe prevents us from doing so, despite the fact that a clogged tail pipe necessarily renders this action impossible.[1] While this *prima facie* ignorance is rational as it is generally impossible to verify all preconditions,[2] these cannot be completely disregarded in a formal causal model. Yet a proposition like "there is no potato in the tail pipe" should not be treated as a *strict* precondition in the formal specification of the action "start the engine," for otherwise the reasoning agent always has to verify this condition before assuming that the action can be successfully executed. Moreover, it is often difficult if not impossible to even think of all conceivable disqualifications in advance [McCarthy, 1977].

Assuming away so-called *abnormal* disqualifications by default naturally implies that if further knowledge hints at such unexpected disqualifications, then we have to withdraw the previous conclusion that the action in question is qualified. Thus the entire process is intrinsically nonmonotonic. As a consequence, McCarthy's proposal was to employ circumscription with the aim of minimizing abnormal disqualifications [McCarthy, 1977; McCarthy, 1980; McCarthy, 1986]. Little has been achieved since then towards formally integrating this concept into a specific action formalism, or towards an assessment of its range of applicability. In fact, a surprisingly simple example illustrates that the straightforward global minimization of abnormal disqualifications is inadequate. The example shows some similarities to the problem—first illustrated with the Yale Shooting example [Hanks and McDermott, 1987]—which occurs when neglecting causality in tackling the frame problem.

Imagine the following scenario (c.f. Figure 1): We can put a potato into the tail pipe whenever no abnormal disqualification prevents us from doing so (e.g., the potato surprisingly turns out to be too heavy); likewise we can start the engine except in case of an abnormal disqualification (like a potato in the tail pipe). Now, what would we predict as to the outcome of first trying to place a potato in the tail pipe and, then, trying to start the engine? Clearly, since nothing hints at an abnormal disqualification of the former action, we should expect this one to be successful. Then its effect (viz. a potato in the tail pipe) implies that the second action will be (abnormally) disqualified.

[1] According to [Ginsberg and Smith, 1988b], this example is also due to McCarthy.

[2] Aside from the fact that besides a clear tail pipe there are lots of other disqualifying, albeit unlikely, obstacles, how can we ensure that after checking the tail pipe it does not become clogged during us walking to the front door and taking a seat, prior to trying to start the engine?

Figure 1: In general, we would consider it abnormal if we fail to start our car. But would we still do so if we deliberately insert a potato into the tail pipe beforehand?

But what happens if abnormal disqualifications are globally minimized in this scenario? One minimal model is obviously obtained by considering the *put-potato* action qualified and the *start-engine* action unqualified, as expected. However, if instead the first action, *put-potato*, is assumed unqualified, then this avoids assuming a disqualification of the second, *start-engine*. For if the former is not qualified it fails to produce what otherwise causes the disqualification of the latter. Hence, in so doing we can construct a second minimal model for our scenario—which is clearly unintended.

The reason for the existence of the second, counter-intuitive model is that global minimization does not allow to distinguish disqualifications which can be explained from the standpoint of causality. Successfully introducing a potato into the tail pipe produces an effect which *causes* the fact that the second action, starting the engine, is unqualified. That is to say, while an abnormal disqualification of *put-potato* comes out of the blue in the unintended minimal model, an abnormal disqualification of *start-engine*, as claimed in the first minimal model, is easily explicable. One even tends to not call the latter abnormal since being unable to start the engine after having clogged the tail pipe is, after all, what one would normally expect. The reader might notice the similarities to the Yale Shooting problem: A gun that becomes magically unloaded while waiting deserves being called abnormal, whereas causality explains the death of the turkey if being shot at with a loaded gun.[3]

The only existing alternative to global minimization of abnormalities as an approach to the qualification problem is based on *chronological ignorance* [Shoham, 1987; Shoham, 1988]. The basic idea there is to assume away by default abnormal, disqualifying circumstances, and simultaneously to prefer minimization of abnormalities at earlier timepoints. While this method treats our example scenario correctly, it is inherently incapable of handling non-deterministic actions, or non-deterministic information in general, as has already been argued elsewhere. A detailed account of this approach is given in the concluding discussion, Section 5.

Given the inadequacy of global minimization and the limited expressiveness of chronological ignorance, we propose a formal account of the qualification problem which incorporates a suitable concept of causality. We accomplish this by assuming, by default, the world *starts* normal, as

[3] The Yale Shooting problem goes as follows (c.f. [Hanks and McDermott, 1987]): Suppose we call abnormal any change of a proposition's truth value during the execution of an action (as suggested in [McCarthy, 1986]). Given that shooting at a turkey with a loaded gun causes the former to drop dead, we would expect exactly this to happen when we start with the gun loaded, wait for a short period, and then shoot. Yet globally minimizing abnormalities in this example produces a second model where the gun becomes unloaded during the first action, waiting, and the turkey survives. While this magical change of the gun's status is abnormal, the turkey surviving the shot is normal in the above sense (as opposed to the change of its life status in the intended model)—hence, this second model minimizes abnormality as well, though it is obviously counter-intuitive.

opposed to assuming the world *is* normal at every instant.[4] More specifically, it is assumed, as far as possible, that each action be qualified initially.[5] Formally, the proposition that an action is abnormally disqualified is taken as a fluent, i.e., a proposition that may change its truth value in the course of time. By virtue of being fluent, this proposition may be indirectly affected by the execution of an action and otherwise is subject to the general law of persistence. This helps to distinguish action disqualifications which are (indirectly) caused by actions that have been observed. In particular, it solves our key problem: We can safely assume that both actions, viz. putting a potato into the tail pipe as well as starting the engine, are qualified at the beginning. Yet, the successful execution of the former affects this assumption regarding the latter—starting the engine becomes unqualified. Notice that this involves no abnormality at all. In contrast, the model in which the *put-potato* action is disqualified in the first place requires to assume an abnormal disqualification from the start, which is why the counter-intuitive model now vanishes when minimizing abnormalities.

Our approach requires to accommodate indirect effects of actions, which is commonly referred to as the *ramification problem* [Ginsberg and Smith, 1988a]. Not being part of the respective action specification, indirect effects are consequences of general laws describing dependencies among components of the world description. In particular, we will employ so-called domain constraints relating fluents of the form $disq(a)$—stating that action a is abnormally disqualified—with known unlikely impediments of performing a, such as in

$$disq(start) \;\equiv\; tail\text{-}pipe\text{-}clogged \lor tank\text{-}empty \lor low\text{-}battery \lor engine\text{-}problem \qquad (1)$$

E.g., whenever some action (as a direct or indirect effect) causes fluent *tail-pipe-clogged* to become true, then this indirectly causes $disq(start)$ to become true as well. For an appropriate treatment of indirect effects we will adopt the approach to the ramification problem proposed in [Thielscher, 1997], which incorporates a suitable notion of causality.

Aside from providing means to assume away abnormal disqualifications by default while properly taking into account possible causes for these disqualifications, the successful treatment of the qualification problem should include the proliferation of possible explanations in case an action has been—unexpectedly—observed unqualified. For example, suppose that, to our own surprise, we encounter difficulties with starting the engine, then we naturally seek a suitable explanation for this among the conceivable alternatives. Suppose further we have already cleaned the tail pipe, checked the tank, and have successfully switched on the radio (confirming the good status of the battery), then it is reasonable, on the basis of (1), to assume a problem with the engine as the cause for the surprising disqualification.

It may of course happen, though, that we are still unable to perform an action even if we have explicitly excluded, to the best of our knowledge, any imaginable preventing cause. However surprising this might be, it just shows us that we have only partial knowledge of the world, that is, the collection of conceivable explanations (like in (1)) turns out to be incomplete. We call *miraculous* a disqualification which is inexplicable in this sense. Thus, a disqualification is to be considered miraculous whenever it cannot be explained even if abnormal circumstances are granted. Consequently, miraculous disqualifications are to be minimized with higher priority than abnormal disqualifications which admit an explanation. Another characteristics of miraculous disqualifications is that they may occur or vanish even if, from our perspective, the situation has not changed. Again this is due to our lack of omniscience.

[4] Here "starts" refers to the initial situation in the scenario under consideration.

[5] Throughout the paper, by "(dis-)qualified" we mean "physically (im-)possible." The refinement that actions may be unqualified *as to producing a certain effect* will be discussed at the end, in Section 5.

To summarize, consider the general task of drawing reasonable conclusions given a domain specification consisting of causal knowledge as to the effects of actions plus some specific observations made during the execution of actions.[6] This might include the observation that in particular situations particular actions cannot be performed. A suitable treatment of the qualification problem should then involve the following. First, it should allow to jump to the default conclusion that an action is qualified once all of its strict preconditions have been verified and in case there is no evidence as to an abnormal disqualification. This should of course be achieved without getting caught in the 'causality trap' illustrated by our version of the Potato In Tail Pipe scenario. Second, if, on the other hand, an action unexpectedly turns out to be disqualified, then it should be possible to explain this, namely, by choosing among the conceivable, albeit unlikely, impediments which are known to render impossible the action in question. Third, the formalism should not capitulate if faced with miraculous, i.e., inexplicable, disqualifications. The formal account of the qualification problem presented in this paper satisfies all of these requirements.

In the second part, we develop, on the basis of the *fluent calculus* [Hölldobler and Schneeberger, 1990; Hölldobler and Thielscher, 1995], an action calculus which includes a proper treatment of abnormal disqualifications. Our encoding builds on the fluent calculus-based solution to the ramification problem developed in [Thielscher, 1997]. Since the qualification problem requires some sort of nonmonotonic feature, we employ *default rules* in the sense of [Reiter, 1980] to formalize the initial normality assumptions as well as the assumption that miraculous disqualifications do not occur. In view of the required priority of the latter, we use the concept of *Prioritized Default Logic* [Brewka, 1994; Rintanen, 1995]. While nonmonotonicity is inherent in the qualification problem, the basic fluent calculus augmented by the aforementioned solution to the ramification problem itself is monotone. This appears to be a decisive advantage when integrating a solution to the qualification problem, for one does not have to worry about possible unintended interferences among different forms of nonmonotonicity which are employed to tackle different problems within a single formalism. The resulting action calculus is proved correct wrt. our formal characterization of the qualification problem.

The paper is organized as follows. In the next section, 2, we introduce a basic action theory including domain constraints and non-deterministic actions. In addition, we recapitulate the aforementioned solution to the ramification problem, which is based on the notion of so-called causal relationships and their application to account for indirect effects of actions. (Section 2.2). As a side gain, this enables us to accommodate *implicit* strict preconditions of actions, which are not part of an action specification but derive from certain domain constraints (see Section 2.3). This is sometimes considered part of the qualification problem, e.g. in [Ginsberg and Smith, 1988b; Lin and Reiter, 1994]. In Section 3, our action theory is extended by a formal account of the qualification problem in the way informally described above. In particular, we formalize the notion of a model for a given domain specification and introduce a suitable preference relation among these models, by which abnormal and miraculous disqualifications are minimized. In Section 4, we then turn to the fluent calculus, extend it by means to successfully handle abnormal disqualifications of actions, and prove the adequacy of this extension with regard to the theory developed in Section 3. Finally, our results are summarized, reviewed and, in particular, compared to related work in Section 5.

[6] The term "reasonable conclusions" appeals to what common sense suggests as to how the given observations are to be interpreted.

2 A Theory of Actions with Ramifications

We first introduce, in Section 2.1, a suitably simple action theory including non-determinism and domain constraints. In Section 2.2, we then recall the causality-based solution to the ramification problem proposed in [Thielscher, 1997]. The latter involves the notion of additional, implicit strict preconditions of actions, which will be illustrated in Section 2.3.

2.1 A Basic Theory of Actions

The basic entities of action scenarios are *states*, each of which is a snapshot of the underlying dynamic system, i.e., the part of the world being modeled, at a particular instant. Formally, we describe a state by assigning truth values to a fixed set of propositional constants.[7]

Definition 1 Let $\mathcal{F}$ be a finite set of symbols called *fluent names*. A *fluent literal* is either a fluent name $f \in \mathcal{F}$ or its negation, denoted by $\overline{f}$. A set of fluent literals is *inconsistent* iff it contains some $f \in \mathcal{F}$ along with $\overline{f}$. A *state* is a maximal consistent set of fluent literals. ■

Notice that formally any combination of truth values denotes a state, which, however, might be considered impossible due to specific dependencies among some fluents (see below). Throughout the paper we assume the following notational conventions: If ℓ is a fluent literal, then $|\ell|$ denotes its affirmative component, that is, $|f| = |\overline{f}| = f$ where $f \in \mathcal{F}$. This notation extends to sets of fluent literals S as follows: $|S| = \{|\ell| : \ell \in S\}$. E.g., for each state S we have $|S| = \mathcal{F}$. Furthermore, if ℓ is a negative fluent literal then $\overline{\ell}$ should be interpreted as $|\ell|$. In other words, $\overline{\overline{f}} = f$. Finally, if S is a set of fluent literals then by $\overline{S}$ we denote the set $\{\overline{\ell} : \ell \in S\}$. E.g., $\overline{\mathcal{F}}$ contains all negative fluent literals given a set $\mathcal{F}$ of fluent names.

The elements of an underlying set of fluent names can be considered atoms for constructing (propositional) formulas to allow for statements about states. Truth and falsity, respectively, of these formulas wrt. a particular state S are based on defining a literal ℓ to be true if and only if $\ell \in S$.

Definition 2 Let $\mathcal{F}$ be a set of fluent names. The set of *fluent formulas* is inductively defined as follows: Each fluent literal in $\mathcal{F} \cup \overline{\mathcal{F}}$ and $\top$ (*tautology*) and $\bot$ (*contradiction*) are fluent formulas, and if F and G are fluent formulas then so are $F \wedge G$, $F \vee G$, $F \supset G$, and $F \equiv G$.[8]

Let S be a state and F a fluent formula, then the notion of F being *true* (resp. *false*) in S is inductively defined as follows:

1. $\top$ is true and $\bot$ is false in S;
2. a fluent literal ℓ is true in S iff $\ell \in S$;
3. $F \wedge G$ is true in S iff F and G are true in S;
4. $F \vee G$ is true in S iff F or G is true in S (or both);
5. $F \supset G$ is true in S iff F is false in S or G is true in S (or both);
6. $F \equiv G$ is true in S iff F and G are true in S, or else F and G are false in S.

[7] A more expressive language, involving non-propositional fluents, will be used in the second part of the paper.

[8] As negation can be expressed through negative literals, we omit the standard connective "$\neg$". This is just for the sake of readability as it avoids too many different forms of negation.

■

Fluent formulas provide means to distinguish states that cannot occur due to specific dependencies among particular fluents. Formulas which have to be satisfied in all states that are possible in a domain are also called *domain constraints*.

Example 1 To model a basic version of the Potato In Tail Pipe scenario, we use the fluent names $\mathcal{F} = \{pot, clog, runs, heavy\}$ to state whether, respectively, there is a potato in the tail pipe, the tail pipe is clogged, the engine is running, and the potato is too heavy. The fluent formula

$$pot \supset clog \tag{2}$$

expresses the fact that the tail pipe is clogged whenever it houses a potato. Taken as domain constraint, this formula is true in the state $\{\overline{pot}, \overline{clog}, \overline{runs}, heavy\}$, say, but false in $\{pot, \overline{clog}, \overline{runs}, \overline{heavy}\}$. ■

The second basic entity in frameworks to reason about dynamic environments are *actions*, whose execution causes state transitions. Since stress shall lie on the qualification problem rather than on sophisticated methods of specifying the direct effects of actions, we employ a suitably simple, Strips-style [Fikes and Nilsson, 1971; Lifschitz, 1986] notion of action specification. Each *action law* consists of

- A *condition* C, which is a set of fluent literals all of which must be contained in the state at hand in order to apply the action law.
- A (direct) *effect* E, which is a set of fluent literals, too, all of which hold in the resulting state after having applied the action law.

It is assumed that $|C| = |E|$, that is, condition and effect refer to the very same set of fluent names. This is just for the sake of simplicity, for it enables us to obtain the state resulting from the direct effect by simply removing set C from the state at hand and adding set E to it. This assumption does not impose a restriction of expressiveness since we allow several laws for a single action, and since any (unrestricted) action law can be replaced by an equivalent set of action laws which obey the assumption.

Definition 3 Let $\mathcal{F}$ be a set of fluent names, and let $\mathcal{A}$ be a finite set of symbols, called *action names*, such that $\mathcal{F} \cap \mathcal{A} = \{\}$. An *action law* is a triple $\langle C, a, E \rangle$ where C, called *condition*, and E, called *effect*, are consistent sets of fluent literals such that $|C| = |E|$; and $a \in \mathcal{A}$.

If S is a state, then an action law $\alpha = \langle C, a, E \rangle$ is *applicable* in S iff $C \subseteq S$. The *application* of α to S yields the state $(S \setminus C) \cup E$. ■

Obviously, S being a state, C and E being consistent, and $|C| = |E|$ guarantee $(S \setminus C) \cup E$ to be a state again—not necessarily, however, one which satisfies the underlying domain constraints.

Example 1 (continued) We define the action names *start* (starting the engine) and *put-p* (putting a potato into the tail pipe), which are accompanied by these action laws:

$$\begin{gathered} \langle\, \{\overline{runs}\}, start, \{runs\}\,\rangle \\ \langle\, \{\overline{pot}\}, put\text{-}p, \{pot\}\,\rangle \end{gathered} \tag{3}$$

In words, starting the engine is possible if it is not running and causes it to do so; similarly, a potato may be added to the tail pipe. The second law, for instance, is applicable in the state $S = \{\overline{pot}, \overline{clog}, \overline{runs}, \overline{heavy}\}$ since $\{\overline{pot}\} \subseteq S$. Its application yields

$$(S \setminus \{\overline{pot}\}) \cup \{pot\} \;=\; \{pot, \overline{clog}, \overline{runs}, \overline{heavy}\}$$

Notice that while the produced set of fluent literals constitutes a state in the sense of Definition 1, it does not satisfy our underlying domain constraint, $pot \supset clog$. ■

Our example illustrates that a state obtained through the application of an action law may violate the underlying domain constraints since only direct effects have been specified: Putting a potato into the tail pipe has the *indirect* effect that the latter becomes clogged. The problem of accommodating additional, indirect effects is commonly referred to as the *ramification problem* [Ginsberg and Smith, 1988a]. In the following section, we recall a solution to this problem which is based on so-called causal relationships and their posterior application to the result of the application of an action law.

Prior to this, observe that according to Definition 3 it is possible to construct a set of action laws which, given a state, contains more than one applicable law for a single action name. This can be used to formalize actions with non-deterministic effects.

Example 2 Suppose we park our car in a neighborhood that is known for its suffering from a tail pipe marauder.[9] We therefore must expect that after waiting for a certain amount of time, a potato may have randomly been introduced into our car's tail pipe. This is formally captured by giving a non-deterministic specification of an action with the name $wait$. Let $\mathcal{F} = \{pot, clog, runs\}$ and $\mathcal{A} = \{wait, start\}$. Performing a $wait$ action either has no effect at all, or else it causes pot become true provided there is not already a potato in the tail pipe. Accordingly, we employ the following two action laws:

$$\langle\, \{\}, wait, \{\}\,\rangle \quad \text{and} \quad \langle\, \{\overline{pot}\}, wait, \{pot\}\,\rangle \tag{4}$$

Both of them are applicable, for instance, in the state $\{\overline{pot}, \overline{clog}, \overline{runs}\}$, which suggests two possible outcomes, viz. $\{\overline{pot}, \overline{clog}, \overline{runs}\}$ and $\{pot, \overline{clog}, \overline{runs}\}$. ■

2.2 The Ramification Problem

The ramification problem arises as soon as it does not suffice to compute the direct effects of actions only, for the resulting collection of fluent literals may violate underlying domain constraints, which in turn give rise to additional, indirect effects. In [Thielscher, 1997], it has been proposed to regard the resulting collection of fluent literals, obtained after having applied an action law as described in Definition 3, merely as an intermediate state, which requires additional computation accounting for possible indirect effects.[10] More specifically, a single indirect effect is obtained according to a directed *causal* relation between two particular fluents.

Definition 4 Let $\mathcal{F}$ be a set of fluent names. A *causal relationship* is an expression of the form ε `causes` ϱ `if` Φ where Φ is a fluent formula and ε and ϱ are fluent literals. ■

[9] This example has been suggested by Erik Sandewall (personal communication).

[10] Related approaches to the ramification problem have been developed in, e.g., [Elkan, 1992; Geffner, 1992; Brewka and Hertzberg, 1993; Lin, 1995; McCain and Turner, 1995]. See [Thielscher, 1997] for a detailed comparison.

The intended reading is the following: Under condition Φ , the (previously obtained, direct or indirect) effect ε triggers the indirect effect ϱ . E.g., the causal relationship *pot* `causes` *clog* `if` $\top$ will be used below to state that the effect *pot* always gives rise to the additional effect *clog* .

Causal relationships operate on pairs (S, E) , where S denotes the current state and E contains all direct and indirect effects computed so far. The reason for employing and manipulating the second component, E , is that identical intermediate states S can be reached by different effects E , each of which may require a different, sometimes opposite treatment (see [Thielscher, 1997] for details).

Definition 5 Let (S, E) be a pair consisting of a state S and a set of fluent literals E , then a causal relationship ε `causes` ϱ `if` Φ is *applicable* to (S, E) iff $\Phi \wedge \overline{\varrho}$ is true in S and $\varepsilon \in E$. Its application yields the pair (S', E') where $S' = (S \setminus \{\overline{\varrho}\}) \cup \{\varrho\}$ and $E' = (E \setminus \{\overline{\varrho}\}) \cup \{\varrho\}$.

Let $\mathcal{R}$ be a set of causal relationships, then by $(S, E) \leadsto_{\mathcal{R}} (S', E')$ we denote the existence of an element in $\mathcal{R}$ whose application to (S, E) yields (S', E') . ■

In words, 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. Notice that if S is a state and E is consistent, then $(S, E) \leadsto_{\mathcal{R}} (S', E')$ implies that S' is a state and E' is consistent, too. In what follows, we say that a sequence of causal relationships $r_1, \ldots, r_n$ is *applicable* to a pair (S_0, E_0) iff there exist pairs $(S_1, E_1), \ldots, (S_n, E_n)$ such that for each $1 \leq i \leq n$, r_i is applicable to (S_{i-1}, E_{i-1}) and yields (S_i, E_i) . We adopt a standard notation in writing $(S, E) \overset{*}{\leadsto}_{\mathcal{R}} (S', E')$ to indicate the existence of a (possibly empty) sequence of causal relationships in $\mathcal{R}$ which is applicable to (S, E) and yields (S', E') .

Example 1 (continued) The following two causal relationships state, respectively, that the effect *pot* always gives rise to the indirect effect *clog* , and that the effect $\overline{clog}$ (as a result of clearing the tail pipe, say) always gives rise to the indirect effect $\overline{pot}$:

$$\begin{array}{llll} pot & \texttt{causes} & clog & \texttt{if} \ \top \\ \overline{clog} & \texttt{causes} & \overline{pot} & \texttt{if} \ \top \end{array} \tag{5}$$

Recall the state $S = \{\overline{pot}, \overline{clog}, \overline{runs}, \overline{heavy}\}$ and action *put-p* . The application of the corresponding action law in (3) yields the state $S_{new} = \{pot, \overline{clog}, \overline{runs}, \overline{heavy}\}$ along with the effect $E = \{pot\}$. Given the pair (S_{new}, E) , the first causal relationship in (5) is applicable on account of both $\top \wedge \overline{clog}$ being true in S_{new} and $pot \in E$. The application yields $((S_{new} \setminus \{\overline{clog}\}) \cup \{clog\}, (E \setminus \{\overline{clog}\}) \cup \{clog\})$, i.e.,

$$(\ \{pot, clog, \overline{runs}, \overline{heavy}\}\ ,\ \{pot, clog\}\ ) \tag{6}$$

■

Now, suppose given a suitable underlying set of causal relationships and a set of fluent literals S as the result of having computed the direct effects of an action via Definition 3. State S may violate the domain constraints. We then compute additional, indirect effects by (non-deterministically) selecting and (serially) applying causal relationships. If this eventually results in a state satisfying the domain constraints, then this state is considered a *successor* state.

Definition 6 Let $\mathcal{F}$ and $\mathcal{A}$ be sets of fluent and action names, respectively, $\mathcal{L}$ a set of action laws, $\mathcal{D}$ a set of domain constraints, and $\mathcal{R}$ a set of causal relationships. Furthermore, let S be a state satisfying $\mathcal{D}$ and $a \in \mathcal{A}$. A state S' is a *successor state* of S and a iff there exists an applicable (wrt. S) action law $\langle C, a, E \rangle \in \mathcal{L}$ such that

1. $((S \setminus C) \cup E, E) \overset{*}{\rightsquigarrow}_{\mathcal{R}} (S', E')$ for some E', and
2. S' satisfies $\mathcal{D}$.

■

E.g., recall the state-effect pair (6). By virtue of being consistent wrt. our domain constraint, $pot \supset clog$, its first component constitutes a successor state of $\{\overline{pot}, \overline{clog}, \overline{runs}, \overline{heavy}\}$ and $put\text{-}p$. The analogue holds for Example 2: There are two successor states of $\{\overline{pot}, \overline{clog}, \overline{runs}\}$ and $wait$, viz. $\{\overline{pot}, \overline{clog}, \overline{runs}\}$ and $\{pot, clog, \overline{runs}\}$.

Based on the above definition, a set of causal laws along with a set of domain constraints and a set of causal relationships determine a *causal model* Σ which maps any pair of an action name and a state to a set of states as follows: $\Sigma(a, S) := \{S' : S' \text{ successor of } S \text{ and } a\}$.

While the order in which causal relationships are applied might be crucial insofar as a different ordering may allow for a different set of causal relationships be applied, we have order independence in case a unique set of relationships is used to obtain a successor state (see [Thielscher, 1997] for details). Yet it is important to realize that neither uniqueness nor the existence of a successor state is guaranteed in general; that is, $\Sigma(a, S)$ may contain several elements or may be empty. The former characterizes actions with non-deterministic behavior even though these actions might be deterministic as regards their direct effects. If no successor exists although an applicable action law can be found, then this indicates that the action under consideration has *implicit* preconditions which are not met. The latter is the subject of the following section, in which we also raise the crucial issue of how to obtain an adequate set of causal relationships on the basis of given domain constraints.

2.3 Implicit Qualifications

Obtaining the intended result by applying causal relationships in order to compute indirect effects of actions relies, to state the obvious, on a suitable collection of these relationships. The two elements in (5), for instance, serve this purpose for our Potato In Tail Pipe domain. While clearly there is a close correspondence between these causal relationships and the underlying domain constraint, not every causal relationship suggested from a pure syntactical point of view by a domain constraint is desirable.[11] In [Thielscher, 1997], we have argued that since the mere domain constraints do not provide sufficient information to exclude unintended causal relationships, additional domain knowledge is required as to possible causal influences between fluents. Called *influence information*, this knowledge is formalized by a binary relation $\mathcal{I}$ on the underlying set of fluent names. Whenever $(f_1, f_2) \in \mathcal{I}$, then this is intended to denote that a change of f_1's truth value might possibly influence the truth value of f_2. On this basis, an adequate set of causal relationships can be automatically extracted from a given set of domain constraints as follows:

Definition 7 Let $\mathcal{F}$ be a set of fluent names, $\mathcal{D}$ a set of domain constraints, and $\mathcal{I} \subseteq \mathcal{F} \times \mathcal{F}$ some influence information. These determine a set of causal relationships $\mathcal{R}$ according to this procedure:

1. Let $\mathcal{R} := \{\}$.

[11] The classical example illustrating this is a domain constraint relating the positions of two switches and the state of a light bulb in an electric circuit. Although being syntactically suggested, toggling one of the two switches must not cause the other one to jump its position in order to preserve the status of the light bulb; see [Lifschitz, 1990].

2. Let $D_1 \wedge \ldots \wedge D_n$ ($n \geq 0$) be the conjunctive normal form (CNF) of $\bigwedge \mathcal{D}$. For each $D_i = \ell_1 \vee \ldots \vee \ell_{m_i}$ ($i = 1, \ldots, n$) do the following:

3. For each $j = 1, \ldots, m_i$ do the following:

4. For each $k = 1, \ldots, m_i$, $k \neq j$ such that $(|\ell_j|, |\ell_k|) \in \mathcal{I}$, add this causal relationship to $\mathcal{R}$:
$$\overline{\ell_j} \ \mathtt{causes} \ \ell_k \ \mathtt{if} \bigwedge_{\substack{l = 1, \ldots, m_i \\ l \neq j, l \neq k}} \overline{\ell_l} \tag{7}$$

■

The reader may verify that given $\mathcal{I} = \{(pot, clog), (clog, pot)\}$, the application of this procedure to the domain constraint $pot \supset clog$ results in the two causal relationships (5).

The following extension of Example 1 shows how domain constraints in conjunction with suitable influence information sometimes give rise to implicit strict preconditions rather than indirect effects.

Example 3 The set of fluent names employed in Example 1 is augmented by key , which is intended to be true if a key is in the ignition lock. Then the additional domain constraint $runs \supset key$ expresses the fact that the engine running requires a key. While a change of the truth value of key may also influence the truth value of $runs$ (namely, removing the key causes the engine to stop), a change of $runs$ cannot possibly influence key (that is, the key cannot magically appear nor disappear by changing the status of the engine). We thus employ the influence information $\mathcal{I} = \{(key, runs)\}$. The application of Definition 7 then yields causal relationships as follows:

- The CNF of $runs \supset key$ is $\overline{runs} \vee key$.

- In case $j = 1, k = 2$ we have $(runs, key) \notin \mathcal{I}$; therefore, no causal relationship is generated.

- In case $j = 2, k = 1$ we have $(key, runs) \in \mathcal{I}$; therefore, the following causal relationship is generated:
$$\overline{key} \ \mathtt{causes} \ \overline{runs} \ \mathtt{if} \ \top \tag{8}$$

Now, consider the state $\{\overline{pot}, \overline{clog}, \overline{runs}, \overline{heavy}, \overline{key}\}$ and action $start$. Following Definition 3, the first law in (3) is applicable and produces the state $S' = \{\overline{pot}, \overline{clog}, runs, \overline{heavy}, \overline{key}\}$ along with the effect $E = \{runs\}$. Clearly, S' violates the domain constraint $runs \supset key$, which is why the former does not constitute a successor state. Moreover, the only causal relationship obtained above, (8), is not applicable to (S', E) , nor are the ones listed in (5). Hence, according to Definition 6 there is no successor state of executing $start$ in the state above. ■

An analysis of this situation reveals key to be an additional, implicit qualification for starting the engine: Whenever key is false, the domain constraint $runs \supset key$ prevents a state obtained by the application of $start$ from being consistent—and no causal relationship exists that may 'correct' this. In general, the non-existence of a successor state despite an applicable action law can be found, hints at additional, implicit preconditions for the action at hand. Notice, however, that these preconditions still are strict and as such not part of the qualification problem dealing with the necessity of assuming away abnormal qualifications.

3 Abnormal Disqualifications

We now take the action theory introduced in the preceding section as the basis for our formal account of the qualification problem. As indicated in the introduction, the general objective is to appropriately interpret a given formal scenario description and to draw reasonable conclusions about it. Any such description involves general action laws in conjunction with causal relationships, plus specific observations as to both the values of certain fluents and, especially, the non-executability of certain actions in particular situations. The term "reasonable conclusions" appeals to what common sense suggests as to how the given observations are to be interpreted.

Definition 8 Let $\mathcal{F}$ and $\mathcal{A}$ be sets of fluent and action names, respectively. An *observation* is an expression of one of the following forms:

$$F \texttt{ after } [a_1, \ldots, a_n] \tag{9}$$

$$a \texttt{ disqualified after } [a_1, \ldots, a_n] \tag{10}$$

where F is a fluent formula and $a, a_1, \ldots, a_n$ are action names ($n \geq 0$). ■

Intuitively, observation (9) indicates that if the sequence of actions $[a_1, \ldots, a_n]$ were performed in the initial state, then F would hold in the resulting state. Likewise, (10) indicates that after performing the sequence of actions $[a_1, \ldots, a_n]$, action a would be unqualified. For instance, given the fluent and action names underlying Example 1, these are possible observations:

$$\overline{pot} \wedge \overline{runs} \texttt{ after } [\,]$$

$$start \texttt{ disqualified after } [put\text{-}p]$$

Definition 9 A *domain description* (or *domain*, for short) consists of sets $\mathcal{F}$ and $\mathcal{A}$ of fluent and action names; sets $\mathcal{L}$, $\mathcal{D}$, and $\mathcal{R}$ of action laws, domain constraints, and causal relationships, respectively; and a set $\mathcal{O}$ of observations. ■

In the remainder of this section, we develop formal notions of interpretations and models for domain descriptions, and we introduce a suitable preference relation among models in view of assuming away, by default, abnormal disqualifications. This model preference criterion induces a (nonmonotonic) entailment relation. Together these concepts constitute our proposal as how to formalize the qualification problem.

3.1 Persistence of Action Qualifications

The unintended model which occurs in the Put Potato In Tail Pipe example when globally minimizing abnormal disqualifications illustrates the necessity of distinguishing disqualifications that admit a *causal* explanation. We have already argued that this can be accomplished by considering the fact that an action is or is not abnormally disqualified as potentially being affected by the execution of other actions and otherwise being subject to the general law of persistence. In other words, the proposition that an action is or is not abnormally disqualified is taken as a fluent. According to the general assumption that the world is 'normal' unless there is information to the contrary, this fluent is assumed *initially* false by default. Restricting the assumption of normality to the initial state enables us to consider it normal, as intended, when an action occurs whose effects suggest an action disqualification which, under general circumstances, would be abnormal.

Let, for each action name a, $disq(a)$ be a fluent name.[12] The intended meaning is that if $\overline{disq(a)}$ holds in some state, then action a is not disqualified for some abnormal reason—which shall imply that a be qualified if and only if all strict preconditions are satisfied.[13] Recall, for instance, Example 3. Whenever $\overline{disq(start)} \in S$ holds in some state S, then the action *start* shall be qualified iff $\overline{runs}, key \in S$, for these two fluent literals are the explicit and implicit strict preconditions of starting the engine.

Abnormal disqualifications indicate abnormal circumstances. These may be described by fluents which, too, are to be assumed false by default. Example fluents of this kind might be *clog* and *pot*, as one normally assumes that the tail pipe is not clogged, let alone the possibility of its housing a potato. Fluents describing abnormal circumstances can be combined in domain constraints to describe the conditions for a particular action being abnormally disqualified. In particular, it is often desirable to equate a fluent $disq(a)$ with the disjunction consisting of all (to the best of the agent's knowledge) the causes for an abnormal disqualification of a, like in (1). This does not only allow to derive an action disqualification from the occurrence of one of its causes, it also supports the proliferation of explanations for abnormal disqualifications that have been observed (see Section 3.2.2, below).

To make all this precise, let $\mathcal{F}$ and $\mathcal{A}$ be sets of fluent and action names, respectively, of a domain description. From now on we always assume determined a certain subset $\mathcal{F}_{ab} \subseteq \mathcal{F}$ of fluents that will be considered initially false by default. It is assumed that $disq(a) \in \mathcal{F}_{ab}$ for each action $a \in \mathcal{A}$. A typical domain constraint involving these special fluents is of the form

$$disq(a) \;\equiv\; \bigvee_{i \in I_a} f_i \tag{11}$$

for some index set I_a such that each $f_i \in \mathcal{F}_{ab}$. That is, each of the 'abnormality' fluents f_i is a potential cause of an abnormal disqualification of action a.[14] These domain constraints may give rise to indirect effects, namely, a change of the truth value of an element in the disjunction may also affect the truth value of $disq(a)$. These indirect effects are formally obtained according to a suitable set of causal relationships. Suppose given that as for the influence information we have $(f_i, disq(a)) \in \mathcal{I}$ for each $i \in I_a$. According to Definition 7, then, a domain constraint of the form (11) determines the following causal relationships:

- The CNF of (11) is $[\overline{disq(a)} \vee \bigvee_{i \in I_a} f_i] \wedge \bigwedge_{i \in I_a} [disq(a) \vee \overline{f_i}]$.
- The first conjunct determines the causal relationship
$$\overline{f_i} \;\; \texttt{causes} \;\; \overline{disq(a)} \;\; \texttt{if} \;\; \bigwedge_{j \in I_a \setminus \{i\}} \overline{f_j} \tag{12}$$
for each $i \in I_a$.
- Each conjunct $disq(a) \vee \overline{f_i}$, $i \in I_a$, determines the causal relationship
$$f_i \;\; \texttt{causes} \;\; disq(a) \;\; \texttt{if} \;\; \top \tag{13}$$

[12] In order to fit our definition of a fluent, each instance $disq(a)$ should be considered a (unique) symbol.

[13] For the moment we neglect the possibility of miraculous disqualifications, which will be discussed later, in Section 3.3.

[14] Instead of explicitly providing the "only-if" part in (11), i.e., $disq(a) \supset \bigvee_{i \in I_a} f_i$, this could be implicitly obtained through circumscribing [McCarthy, 1980] the predicate *disq* in a given set of domain constraints; c.f. [Lifschitz, 1987], where this idea is applied to strict preconditions of actions.

In words, if a disqualifying cause disappears and none of the alternative causes holds then $\mathit{disq}(a)$ becomes false, (12), whereas $\mathit{disq}(a)$ becomes true through the appearance of any disqualifying cause, (13).

Example 1 (continued) Let the set $\mathcal{F}_{ab}$ consist of the fluents pot, clog, heavy, along with $\mathit{disq}(\mathit{start})$ and $\mathit{disq}(\mathit{put\text{-}p})$. Suppose further that the set of domain constraints includes

$$\begin{array}{rcl} \mathit{disq}(\mathit{start}) & \equiv & \mathit{clog} \\ \mathit{disq}(\mathit{put\text{-}p}) & \equiv & \mathit{heavy} \end{array} \tag{14}$$

aside from $\mathit{pot} \supset \mathit{clog}$. Given $(\mathit{clog}, \mathit{disq}(\mathit{start})), (\mathit{heavy}, \mathit{disq}(\mathit{put\text{-}p})) \in \mathcal{I}$, the additional domain constraints determine these four causal relationships:

$$\begin{array}{llll} \overline{\mathit{clog}} \;\texttt{causes}\; \overline{\mathit{disq}(\mathit{start})} \;\texttt{if}\; \top & & \overline{\mathit{heavy}} \;\texttt{causes}\; \overline{\mathit{disq}(\mathit{put\text{-}p})} \;\texttt{if}\; \top \\ \mathit{clog} \;\texttt{causes}\; \mathit{disq}(\mathit{start}) \;\texttt{if}\; \top & & \mathit{heavy} \;\texttt{causes}\; \mathit{disq}(\mathit{put\text{-}p}) \;\texttt{if}\; \top \end{array} \tag{15}$$

in conjunction with the ones shown in (5). Suppose, now, we perform action $\mathit{put\text{-}p}$ in the state $S = \{\overline{\mathit{pot}}, \overline{\mathit{clog}}, \overline{\mathit{runs}}, \overline{\mathit{heavy}}, \overline{\mathit{disq}(\mathit{start})}, \overline{\mathit{disq}(\mathit{put\text{-}p})}\}$. The application of the corresponding action law in (3) yields the state-effect pair

$$(\; \{\mathit{pot}, \overline{\mathit{clog}}, \overline{\mathit{runs}}, \overline{\mathit{heavy}}, \overline{\mathit{disq}(\mathit{start})}, \overline{\mathit{disq}(\mathit{put\text{-}p})}\} \;,\; \{\mathit{pot}\} \;)$$

The first component does not satisfy $\mathit{pot} \supset \mathit{clog}$, but we can apply the first causal relationship in (5), viz. $\mathit{pot} \;\texttt{causes}\; \mathit{clog} \;\texttt{if}\; \top$, yielding

$$(\; \{\mathit{pot}, \mathit{clog}, \overline{\mathit{runs}}, \overline{\mathit{heavy}}, \overline{\mathit{disq}(\mathit{start})}, \overline{\mathit{disq}(\mathit{put\text{-}p})}\} \;,\; \{\mathit{pot}, \mathit{clog}\} \;)$$

While now the aforementioned domain constraint is satisfied, the first fluent formula in (14) is no longer so, which is why we further apply the appropriate causal relationship in (15), namely, $\mathit{clog} \;\texttt{causes}\; \mathit{disq}(\mathit{start}) \;\texttt{if}\; \top$, which results in the pair

$$(\; \{\mathit{pot}, \mathit{clog}, \overline{\mathit{runs}}, \overline{\mathit{heavy}}, \mathit{disq}(\mathit{start}), \overline{\mathit{disq}(\mathit{put\text{-}p})}\} \;,\; \{\mathit{pot}, \mathit{clog}, \mathit{disq}(\mathit{start})\} \;) \tag{16}$$

Its first component satisfies all domain constraints and, thus, constitutes a successor state. Notice that action start is declared abnormally disqualified in the resulting state. This disqualification occurs as an indirect effect of having performed $\mathit{put\text{-}p}$. On the other hand, executing this action did not affect the fluent $\mathit{disq}(\mathit{put\text{-}p})$, which thus remains false according to the law of persistence. ■

3.2 Assuming Qualification by Default

The intention of distinguishing a set of 'abnormality' fluents $\mathcal{F}_{ab}$ is to prefer among all suitable interpretations of domain descriptions those in which they are initially false. This would enable us to assume away abnormal circumstances whenever that is reasonable. Prior to discussing preference, however, we need to formalize the notions of interpretation and model in general. Clearly, they both ought to respect the causal model Σ underlying the domain in question. Each interpretation (and model) contains a partial function Res which maps finite action sequences to states with the intended meaning that $\mathit{Res}([a_1, \ldots, a_n])$ would be the result of executing the action sequence $[a_1, \ldots, a_n]$ in the initial state (which itself is determined by $\mathit{Res}([\,])$). If $\mathit{Res}([a_1, \ldots, a_n])$ is undefined, then this indicates that some action a_i in this sequence is unqualified in the corresponding state $\mathit{Res}([a_1, \ldots, a_{i-1}])$. All this is made precise in the following definition:

Definition 10 Let Σ be the causal model determined by a domain description with fluent and action names $\mathcal{F}$ and $\mathcal{A}$, respectively, and domain constraints $\mathcal{D}$. A pair (Res, Σ) is an *interpretation* for this domain iff Res is a partial mapping from finite sequences of action names to states such that the following holds:

1. $Res([\,])$ is defined and satisfies $\mathcal{D}$.

2. For any finite sequence $[a_1, \ldots, a_{n-1}, a_n]$ of action names ($n > 0$), $Res([a_1, \ldots, a_{n-1}, a_n])$ is defined iff

   (a) $Res([a_1, \ldots, a_{n-1}])$ is defined;
   (b) $\overline{disq(a_n)}$ holds in $Res([a_1, \ldots, a_{n-1}])$; and
   (c) $\Sigma(a_n, Res([a_1, \ldots, a_{n-1}])) \neq \{\}$

   If it is defined, then $Res([a_1, \ldots, a_{n-1}, a_n]) \in \Sigma(a_n, Res([a_1, \ldots, a_{n-1}]))$.

■

If $Res([a_1, \ldots, a_n])$ is defined, we also say that the action sequence $[a_1, \ldots, a_n]$ is *qualified*. Then Definition 10 states that $[a_1, \ldots, a_{n-1}, a_n]$ is qualified if so is $[a_1, \ldots, a_{n-1}]$, if all (explicit or implicit) preconditions of a_n are met—which implies the existence of a successor state of a_n and $Res([a_1, \ldots, a_{n-1}])$—, and if the state $Res([a_1, \ldots, a_{n-1}])$ does not imply an abnormal disqualification of a_n—which is indicated by fluent $disq(a_n)$ being false in this state. If $[a_1, \ldots, a_{n-1}, a_n]$ is qualified, then $Res([a_1, \ldots, a_{n-1}, a_n])$ must be a successor state of $Res([a_1, \ldots, a_{n-1}])$ and a_n. Notice that all function values of Res which are defined necessarily satisfy the underlying domain constraints since $Res([\,])$ does, as required in clause 1.

Based on the given a set of observations, an interpretation for a domain is considered a model iff all the observations hold in that interpretation.

Definition 11 Let Σ be the causal model of a domain description with fluent and action names $\mathcal{F}$ and $\mathcal{A}$, respectively, and observations $\mathcal{O}$. An interpretation (Res, Σ) is a *model of* $\mathcal{O}$ iff each observation in $\mathcal{O}$ holds in (Res, Σ), where

1. F `after` $[a_1, \ldots, a_n]$ is said to *hold* in (Res, Σ) iff $Res([a_1, \ldots, a_n])$ is defined and F is true in $Res([a_1, \ldots, a_n])$;

2. a `disqualified after` $[a_1, \ldots, a_n]$ is said to *hold* in (Res, Σ) iff $Res([a_1, \ldots, a_n])$ is defined but $Res([a_1, \ldots, a_n, a])$ is not.

■

In words, an observation of type (9) holds if the respective action sequence is qualified and the assigned state satisfies the given formula. An observation of type (10) holds if again the respective action sequence is qualified while in the resulting state the additional action, a, cannot be performed—either because an abnormal disqualification occurs (c.f. Definition 10, clause 2(b)) or some strict precondition is not satisfied (c.f. Definition 10, clause 2(c)).

Example 1 (continued) Let Σ be the causal model determined by the action laws (3), the domain constraints (2) and (14), and the causal relationships (5) and (15). Suppose given the observation

$$\overline{runs} \ \texttt{after} \ [\,] \tag{17}$$

and consider, say, these four initial states:[15]

$$\begin{aligned}
Res_1([\,]) &= \{\overline{pot}, \overline{clog}, \overline{runs}, heavy, \overline{disq(start)}, \overline{disq(put\text{-}p)}\} \\
Res_2([\,]) &= \{\overline{pot}, \overline{clog}, runs, \overline{heavy}, \overline{disq(start)}, \overline{disq(put\text{-}p)}\} \\
Res_3([\,]) &= \{\overline{pot}, \overline{clog}, \overline{runs}, \overline{heavy}, \overline{disq(start)}, \overline{disq(put\text{-}p)}\} \\
Res_4([\,]) &= \{\overline{pot}, \overline{clog}, \overline{runs}, heavy, \overline{disq(start)}, disq(put\text{-}p)\}
\end{aligned} \qquad (18)$$

While (Res_1, Σ) is not an interpretation since $Res_1([\,])$ violates the second domain constraint in (14), and (Res_2, Σ) is not a model since it violates the given observation (17), both (Res_3, Σ) and (Res_4, Σ) are models. Notice, however, that no 'abnormality' fluent is true in $Res_3([\,])$, as opposed to $Res_4([\,])$. Since $disq(start) \in \Sigma(put\text{-}p, Res_3([\,]))$ (c.f. (16)), the model (Res_3, Σ) entails that the engine cannot be ignited after putting a potato into the tail pipe. In contrast, the model (Res_4, Σ) is the formal counterpart of the counter-intuitive conclusion where the action *put-p* is assumed to be abnormally disqualified in the first place. ■

While an interpretation must satisfy the given observations in order to constitute a model, this criterion alone does not suffice to assume away abnormal disqualifications. Obviously, the addition of observations can only decrease the set of models, never produce new ones. Consequently, if one defines an entailment relation stating that an observation is entailed by a set of observations if the former holds in all models of the latter, then this relation is *monotone*.[16] In view of the qualification problem, however, this kind of monotonicity needs to be dropped because additional observations, such as detecting a potato in the tail pipe, may force us to withdraw previous (default) conclusions, like the conclusion that we are able to start the engine. This is formally achieved by introducing a preference relation among the set of models, with the intention to select those which initially minimize truth of fluents in $\mathcal{F}_{ab}$ to the largest possible extent. When talking about entailment, attention is then restricted to models which are preferred in this sense. The following definition constitutes the core of our formal characterization of the qualification problem:

Definition 12 Let $\mathcal{F} \supseteq \mathcal{F}_{ab}$ be the set of fluent names and $\mathcal{O}$ the set of observations of a domain description with causal model Σ. An interpretation $M' = (Res', \Sigma)$ is *less abnormal* than an interpretation $M = (Res, \Sigma)$, written $M' \prec M$, iff $Res'([\,]) \cap \mathcal{F}_{ab} \subsetneqq Res([\,]) \cap \mathcal{F}_{ab}$.

A model M of $\mathcal{O}$ is *preferred* iff there is no model M' of $\mathcal{O}$ such that $M' \prec M$. An observation o is *entailed*, written $\mathcal{O} \mathrel{|\!\!\sim}_{\Sigma} o$, iff o holds in each preferred model of $\mathcal{O}$. ■

In words, the less fluents in $\mathcal{F}_{ab}$ occur affirmatively in the initial state in a model the better. Obviously, the induced entailment relation, $\mathrel{|\!\!\sim}_{\Sigma}$, is nonmonotonic as the addition of observations may change the set of preferred models entirely. In the sequel, we illustrate how this formal account of the qualification problem satisfies all the requirements which we demanded in the introduction.

3.2.1 How to assume away abnormal disqualifications

The fundamental issue of the qualification problem is to assume away abnormal disqualifications by default. This, however, should only concern those disqualifications which do not admit a

[15] Notice that if all actions in a domain are deterministic (that is, each $\Sigma(a, S)$ is singleton or empty), then an interpretation is uniquely characterized by its initial state, $Res([\,])$. We assume that the following four functions Res_k are defined accordingly, i.e., wrt. the underlying causal model in the example.

[16] This phenomenon is called *restricted monotonicity* in [Lifschitz, 1993]; see also Section 5.

causal explanation. Our key example, in particular, is now treated in the expected way. Namely, any potential abnormal disqualification preventing us from putting a potato into the tail pipe is assumed away, for there is no evidence to the contrary. Likewise, any abnormal disqualification preventing us from starting the engine is assumed away as regards the initial state, whereas an abnormal disqualification of this very action after the insertion of a potato follows from the causal model without the necessity of granting abnormal circumstances.

Example 1 (continued) Recall from (18) the two models $M_3 = (Res_3, \Sigma)$ and $M_4 = (Res_4, \Sigma)$ of the observation (17). Following Definition 12, we have $M_3 \prec M_4$ due to $Res_3([\,]) \cap \mathcal{F}_{ab} = \{\}$ and $Res_4([\,]) \cap \mathcal{F}_{ab} = \{heavy, disq(put\text{-}p)\}$. Since each 'abnormality' fluent is false in the initial state in M_3, the latter obviously constitutes the unique preferred model. Whatever holds in M_3 is thus entailed by the domain. In particular, from $\overline{pot}, \overline{disq(put\text{-}p)} \in Res_3([\,])$ and from $\overline{runs}, \overline{disq(start)} \in Res_3([\,])$, we conclude that both $[put\text{-}p]$ and $[start]$ are qualified, according to the underlying causal model Σ. But, as we have seen in (16), we also know that $disq(start) \in Res_3([put\text{-}p])$. This implies that $[put\text{-}p, start]$ is not qualified in M_3, which in turn sanctions the entailment of

$$start \;\; \texttt{disqualified after} \;\; [put\text{-}p]$$

This constitutes the intended solution to our key example: The first action, *put-p*, is qualified by default and, as a consequence, action *start* is unqualified afterwards. ■

3.2.2 How to explain observed abnormal disqualifications

Aside from assuming away abnormal disqualifications by default, one naturally seeks conceivable explanations in case a disqualifications has been—unexpectedly—observed without an apparent cause. Each preferred model that contains an abnormal disqualification also includes, provided the underlying domain constraints support this, a particular explanation. For otherwise the domain constraints would be violated in the state in which the disqualification occurs. The set of conceivable explanations thus is determined by the set of preferred models, as the following example illustrates.

Example 4 We extend the set of fluent names given in Example 1 by *tank-empty*, *low-battery*, and *engine-problem*, each of which shall belong to the subset $\mathcal{F}_{ab}$. These fluent names are combined in this domain constraint:

$$disq(start) \;\equiv\; clog \vee tank\text{-}empty \vee low\text{-}battery \vee engine\text{-}problem \tag{19}$$

which shall replace the first formula in (14). Now suppose we are in a state where the engine is not running and where we also know that the tail pipe is not clogged nor is the tank empty, but nonetheless we encounter difficulties with starting the engine. The corresponding observations, i.e.,

$$\begin{array}{l} \overline{runs} \;\; \texttt{after} \;\; [\,] \\ \overline{clog} \wedge \overline{tank\text{-}empty} \;\; \texttt{after} \;\; [\,] \\ start \;\; \texttt{disqualified after} \;\; [\,] \end{array}$$

admit two preferred models (Res, Σ), each of which satisfies $disq(start) \in Res([\,])$ since $[start]$ is unqualified according to the third observation although the only strict precondition of *start*, viz. $\overline{runs}$, is initially true according to the first observation. Given $disq(start) \in Res([\,])$, the above domain constraint, (19), requires an additional 'abnormality' fluent be initially true in any model. The second observation excludes both clog and $\mathit{tank\text{-}empty}$. Hence, each preferred model satisfies either $\mathit{low\text{-}battery} \in \mathit{Res}([\,])$ or $\mathit{engine\text{-}problem} \in \mathit{Res}([\,])$. This in turn sanctions the entailment of the observation

$$\mathit{low\text{-}battery} \vee \mathit{engine\text{-}problem} \;\; \texttt{after} \;\; [\,] \tag{20}$$

In other words, both problems with the battery and problems with the engine itself are the conceivable explanations of the observed abnormal disqualification of start . ■

3.2.3 How to assume away unnecessary abnormal explanations

Unless an abnormal disqualification follows from the standpoint of causality, as in Example 1, conceivable explanations themselves describe some unusual circumstances, such as the two explanations found in (20). As such, we expect these explanations, too, be assumed away to the largest possible extent. That is to say, if an observed abnormal disqualification can be explained by some abnormal circumstances which have to be assumed anyway, then it is reasonable to consider this the appropriate explanation and to exclude other unlikely causes. Our model preference criterion affords this, which is illustrated with the following example.

Example 5 Let us extend further the previous Example 4 by $\mathit{radio\text{-}on}, \mathit{radio\text{-}problem} \in \mathcal{F}_{ab}$ and the action name $\mathit{turn\text{-}on\text{-}radio}$ along with the action law

$$\langle\, \{\overline{\mathit{radio\text{-}on}}\}, \mathit{turn\text{-}on\text{-}radio}, \{\mathit{radio\text{-}on}\}\,\rangle$$

and the domain constraint

$$\mathit{disq}(\mathit{turn\text{-}on\text{-}radio}) \;\equiv\; \mathit{low\text{-}battery} \vee \mathit{radio\text{-}problem} \tag{21}$$

Suppose again the engine is not running and further that the radio is silent but we know it is intact. Nonetheless we encounter that we are not able to start the engine nor to turn on the radio. Given the formal observations

$$\begin{aligned}
&\overline{\mathit{runs}} \wedge \overline{\mathit{radio\text{-}problem}} \;\; \texttt{after} \;\; [\,] \\
&\mathit{start} \;\; \texttt{disqualified after} \;\; [\,] \\
&\mathit{turn\text{-}on\text{-}radio} \;\; \texttt{disqualified after} \;\; [\,]
\end{aligned}$$

each model must include both $\mathit{disq}(\mathit{start})$ and $\mathit{disq}(\mathit{turn\text{-}on\text{-}radio})$ since the respective strict preconditions are satisfied and yet both $[\mathit{start}]$ and $[\mathit{turn\text{-}on\text{-}radio}]$ are not qualified. From (21) and the first observation it then follows that each model satisfies $\mathit{low\text{-}battery} \in \mathit{Res}([\,])$. According to domain constraint (19), $\mathit{low\text{-}battery}$ also accounts for the observed abnormal disqualification of start . Hence, the domain admits a single preferred model, which entails, among others,

$$\overline{\mathit{clog}} \;\; \texttt{after} \;\; [\,]$$

That is to say, since the failure of trying to turn on the radio suggests a battery problem, which also explains the failure of trying to start the engine, we arrive at the reasonable (default) conclusion that the tail pipe is not clogged. ■

3.2.4 How to deal with non-deterministic information

The failure of the *chronological ignorance* approach to the qualification problem [Shoham, 1987; Shoham, 1988] in case of non-deterministic actions demonstrates a crucial difficulty with combining both abnormal disqualifications and non-determinism. As will be shown in more detail later, in Section 5, the problem occurs whenever non-deterministic information provides sufficient evidence for an abnormal disqualification without, by virtue of being non-deterministic, necessitating it. Any formalism by which abnormal circumstances are negated whenever they do not provably hold, ignores uncertain evidence and, in so doing, supports unsound conclusions. As the following example illustrates, our formal characterization of the qualification problem does not interfere with non-deterministic information and treats the latter in the appropriate, namely, the cautious way.

Example 2 (continued) Suppose given the observation

$$\overline{runs}\ \texttt{after}\ [\,]$$

Then the set of preferred models for the Tail Pipe Marauder domain divides into two classes. Since it is consistent with the observation to consider initially false all members of $\mathcal{F}_{ab}$, any preferred model (Res, Σ) must satisfy

$$Res([\,]) = \{\overline{pot}, \overline{clog}, \overline{runs}, \overline{disq(wait)}, \overline{disq(start)}\}$$

The action $wait$ being non-deterministic (c.f. (4)), we know that $Res([wait]) = Res([\,])$ or $Res([wait]) = \{pot, clog, \overline{runs}, \overline{disq(wait)}, disq(start)\}$ holds in preferred models. Therefore, nothing definite follows about the status of the tail pipe, hence of the qualification of $start$, after performing $[wait]$. Consequently, the observation $runs$ after $[wait, start]$, say, is not entailed, as intended. ■

3.3 Miraculous Disqualifications

Thus far our theory supports generating explanations for surprising disqualifications by selecting among the conceivable reasons for this abnormality. Yet whenever the domain description renders invalid each of these explanations, then this goes beyond the capacity of the theory. Suppose given, as an example, the two observations

$$\begin{array}{l} start\ \texttt{disqualified after}\ [\,] \\ runs\ \texttt{after}\ [wait, start] \end{array} \tag{22}$$

where $wait$ is assumed to have no effects at all on the underlying fluents. No however (*a priori*) 'unlikely' model exists which simultaneously satisfies both of the observations. The reason is that any abnormality explaining the first disqualification necessarily transfers to the state after waiting, which contradicts the following success of performing $start$. Nonetheless, such situations, where the available explanations are insufficient to account for surprising disqualifications, are well conceivable and just prove our lack of omniscience.

We therefore need to extend our formalism to allow for observed yet inexplicable, in the above sense, action disqualifications. To this end, the formal notions of interpretation and model are enhanced by a component accommodating these so-called *miraculous* disqualifications. As we have seen, a miraculous disqualification may appear or disappear even though the truth values of the fluents suggest identical states. This is why any such disqualification is to be associated with the sequence of actions after whose execution it occurs, rather than with the respective state. Formally, the new component, denoted by Υ, consists of non-empty action sequences indicating the following: If $[a_1, \ldots, a_{n-1}, a_n] \in \Upsilon$ ($n > 0$), then action a_n is disqualified in the state resulting from performing $[a_1, \ldots, a_n]$ even if all strict preconditions of a_n and also $\overline{\mathit{disq}(a_n)}$ hold in that state. The following extends Definitions 10 and 11 accordingly.

Definition 13 Let Σ be the causal model determined by a domain description with fluent and action names $\mathcal{F}$ and $\mathcal{A}$, respectively, and domain constraints $\mathcal{D}$. A triple $(\mathit{Res}, \Sigma, \Upsilon)$ is an *interpretation* for this domain iff Υ is a set of non-empty, finite sequences of action names and Res is a partial mapping from finite sequences of action names to states such that the following holds:

1. $\mathit{Res}([\,])$ is defined and satisfies $\mathcal{D}$.
2. For any finite sequence $[a_1, \ldots, a_{n-1}, a_n]$ of action names ($n > 0$), $\mathit{Res}([a_1, \ldots, a_{n-1}, a_n])$ is defined iff
   (a) $\mathit{Res}([a_1, \ldots, a_{n-1}])$ is defined;
   (b) $\overline{\mathit{disq}(a_n)}$ holds in $\mathit{Res}([a_1, \ldots, a_{n-1}])$;
   (c) $\Sigma(a_n, \mathit{Res}([a_1, \ldots, a_{n-1}])) \neq \{\}$; and
   (d) $[a_1, \ldots, a_{n-1}, a_n] \notin \Upsilon$.

   If it is defined, then $\mathit{Res}([a_1, \ldots, a_{n-1}, a_n]) \in \Sigma(a_n, \mathit{Res}([a_1, \ldots, a_{n-1}]))$.

An interpretation $(\mathit{Res}, \Sigma, \Upsilon)$ for a domain with observations $\mathcal{O}$ is a *model of* $\mathcal{O}$ iff each observation in $\mathcal{O}$ holds in $(\mathit{Res}, \Sigma, \Upsilon)$, where

1. F `after` $[a_1, \ldots, a_n]$ is said to *hold* in $(\mathit{Res}, \Sigma, \Upsilon)$ iff $\mathit{Res}([a_1, \ldots, a_n])$ is defined and F is true in $\mathit{Res}([a_1, \ldots, a_n])$;
2. a `disqualified after` $[a_1, \ldots, a_n]$ is said to *hold* in $(\mathit{Res}, \Sigma, \Upsilon)$ iff $\mathit{Res}([a_1, \ldots, a_n])$ is defined but $\mathit{Res}([a_1, \ldots, a_n, a])$ is not.

■

The additional clause, 2(d), states that a sequence of actions can only be qualified if it is not miraculously disqualified.

Example 6 The domain discussed in Example 1 is extended by the action name *wait* in conjunction with the action law $\langle \{\}, \mathit{wait}, \{\} \rangle$. Furthermore, suppose given the aforementioned observations (22). While no model $(\mathit{Res}, \Sigma, \Upsilon)$ with $\Upsilon = \{\}$ exists for this domain, as argued above, both these observations hold in the interpretation $(\mathit{Res}, \Sigma, \Upsilon)$ where

$$\begin{aligned} \mathit{Res}([\,]) &= \{\overline{\mathit{pot}}, \overline{\mathit{clog}}, \overline{\mathit{runs}}, \overline{\mathit{heavy}}, \overline{\mathit{disq}(\mathit{start})}, \overline{\mathit{disq}(\mathit{put\text{-}p})}\} \\ \Upsilon &= \{\, [\mathit{start}] \,\} \end{aligned} \tag{23}$$

This interpretation thus constitutes a model. ■

Clearly, miraculous disqualifications, too, are to be minimized to the largest possible extent. Moreover, miraculous disqualifications are meant as means to account for abnormal disqualifications which do not admit an explanation even by granting abnormal circumstances. As such, miraculous disqualifications need to be minimized with higher priority. As opposed to explicable disqualifications, miraculous ones can well be minimized globally, that is, without worrying about causality—would they admit a causal explanation they would not be miraculous. We thus arrive at the following extension of our preference criterion:

Definition 14 Let $\mathcal{F} \supseteq \mathcal{F}_{ab}$ be the set of fluent names and $\mathcal{O}$ the set of observations of a domain description with causal model Σ. An interpretation $M' = (\mathit{Res}', \Sigma, \Upsilon')$ is *less abnormal* than an interpretation $M = (\mathit{Res}, \Sigma, \Upsilon)$, written $M' \prec M$, iff

1. either $\Upsilon' \subsetneqq \Upsilon$,
2. or $\Upsilon' = \Upsilon$ and $\mathit{Res}'([\,]) \cap \mathcal{F}_{ab} \subsetneqq \mathit{Res}([\,]) \cap \mathcal{F}_{ab}$.

The notions of preferred model and entailment of Definition 12 modify accordingly. ■

Example 6 (continued) We have seen that the domain considered above does not admit a model without miraculous disqualifications. It follows that the model $M = (\mathit{Res}, \Sigma, \Upsilon)$ which satisfies (23) is preferred, for it declares a single action sequence miraculously disqualified and negates each 'abnormality' fluent in the initial state. As a matter of fact, M is the only preferred model since any model $(\mathit{Res}', \Sigma, \Upsilon')$ must satisfy $[\mathit{start}] \in \Upsilon'$ and also $\overline{\mathit{runs}} \in \mathit{Res}'([\,])$ (the latter is due to $[\mathit{wait}, \mathit{start}]$ being qualified according to (22)). ■

This completes our formal characterization of the qualification problem. Let us summarize: Each domain is supposed to contain a distinguished set of fluents $\mathcal{F}_{ab}$, each of which describes abnormal circumstances and thus is to be assumed false by default. This assumption, however, needs to be restricted to the initial state, so that these fluents are subject to the general law of persistence but are also potentially (directly or indirectly) affected by the execution of actions. Among these 'abnormality' fluents are propositions, denoted $\mathit{disq}(a)$, which state that an action a is abnormally disqualified. Domain constraints relating these fluents with possible causes of an abnormal disqualification support the proliferation of explanations in case an abnormal disqualification—surprisingly—occurs. In addition, miraculous disqualifications accommodate situations in which a suitable explanation cannot be provided. The default assumption of 'normality' is formally represented by a model preference criterion (Definition 14), which induces a nonmonotonic entailment relation among observations.

4 A Fluent Calculus Solution to the Qualification Problem

Following our proposal for a formal account of the qualification problem, the second part of the paper is devoted to the development of an action calculus which is capable of handling abnormal action disqualifications. Our encoding employs the representation technique underlying the *fluent calculus* [Hölldobler and Schneeberger, 1990; Hölldobler and Thielscher, 1995]. We begin by repeating the fluent calculus-based formalization of causal relationships developed in [Thielscher, 1997] (Section 4.1). In Section 4.2, this calculus is extended by a suitable encoding of observations. Finally and as a solution to the qualification problem, in Section 4.3 we embed the entire formalism in a default theory, where default rules are used to express the various assumptions of normality. As the main result of this second part of the paper, the action calculus is proved correct wrt. the formal characterization of the qualification problem developed in the first part.

4.1 Fluent Calculus and Ramification

The atomic elements of state descriptions have been restricted, for the sake of simplicity, to propositional constants throughout the first part of the paper. For our calculus, we introduce a richer notion of fluents. A fluent is now an n-place predicate with arguments chosen from a given set of objects (or *entities*) [Sandewall, 1994; Kartha and Lifschitz, 1994]. This involves both a generalized concept of action laws and fluent formulas including quantifications.

Definition 15 Let $\mathcal{E}$ be a finite set of symbols called *entities*. Let $\mathcal{F}$ denote a set of fluent names, each of which is associated with a natural number called *arity*. A *fluent* is an expression $f(e_1, \ldots, e_n)$ where $f \in \mathcal{F}$ is of arity n and $e_1, \ldots, e_n \in \mathcal{E}$. A *fluent literal* is a fluent or its negation, denoted by $\overline{f(e_1, \ldots, e_n)}$.

Let $\mathcal{V}$ be a denumerable set of *variables*. An expression $f(t_1, \ldots, t_n)$ and its negation $\overline{f(t_1, \ldots, t_n)}$ are called *fluent expressions* iff $f \in \mathcal{F}$ is of arity n and $t_i \in \mathcal{E} \cup \mathcal{V}$ ($1 \leq i \leq n$). ■

As before, a *state* is a maximal consistent set of fluent literals. For the sake of simplicity, from now on we assume given an arbitrary but fixed set $\mathcal{E}$ of entities, a set $\mathcal{F}$ of fluent names with subset $\mathcal{F}_{ab}$, and a set $\mathcal{V}$ of variables, respectively. It is assumed, also for the sake of clarity, that if $f_{ab} \in \mathcal{F}_{ab} \subseteq \mathcal{F}$, then any instance $f_{ab}(e_1, \ldots, e_n)$ expresses abnormal circumstances, thus is subject to minimization.

As opposed to the situation calculus [McCarthy and Hayes, 1969; Reiter, 1991], the fluent calculus employs structured state terms, each of which consists in a collection of all fluent literals that are true in the state being represented. To this end, fluent literals are reified [Quine, 1960], i.e., formally represented as terms. These terms are connected via a special binary function, which is illustratively denoted by $\circ$ and written in infix notation. For instance, suppose $S = \{\mathit{in\text{-}pipe}(po), \overline{\mathit{heavy}(po)}, \mathit{clog}\}$ is a state, then a term representation of S is

$$(\mathit{in\text{-}pipe}(po) \circ \overline{\mathit{heavy}(po)}) \circ \mathit{clog} \tag{24}$$

where the bar denoting negative fluent expressions is formally a unary function. It has first been argued in [Hölldobler and Schneeberger, 1990] that this representation technique avoids extra axioms (e.g., frame axioms [McCarthy and Hayes, 1969; Green, 1969]) to encode the general law of persistence: The effects of actions are modeled by manipulating terms like (24) through removal and addition of sub-terms. Then all sub-terms which are not affected by these operations remain in the state term, hence continue to be true.

Intuitively, the position at which a fluent literal occurs in a state term should be irrelevant. That is, (24) and the term $\overline{\mathit{heavy}(po)} \circ (\mathit{clog} \circ \mathit{in\text{-}pipe}(po))$, say, represent identical states. This intuition is modeled by requiring the following formal properties for the connection function $\circ$:

$$\begin{array}{rrcll}
\forall x, y, z. & (x \circ y) \circ z & = & x \circ (y \circ z) & \text{(associativity)} \\
\forall x, y. & x \circ y & = & y \circ x & \text{(commutativity)} \\
\forall x. & x \circ \emptyset & = & x & \text{(unit element)}
\end{array}$$

where the special constant $\emptyset$ denotes a unit element for $\circ$. This constant represents the empty collection of fluent literals. The above axioms constitute an *equational theory*, which we abbreviate by AC1. Given the law of associativity, from now on we omit parentheses on the level of $\circ$. Notice that the axioms AC1 formalize essential properties of the datastructure "set." For formal reasons, we introduce a mapping τ from sets of fluent expressions $A = \{\ell_1, \ldots, \ell_n\}$ to the term representation $\tau_A = \ell_1 \circ \cdots \circ \ell_n$ (including $\tau_{\{\}} = \emptyset$).

In order that the inequality of two state terms follows whenever they consist in different collections of fluent literals, an extension of the standard *unique name assumption* is needed, namely, the concept of *unification completeness* known from logic programming (see, e.g., [Jaffar *et al.*, 1984; Shepherdson, 1992; Thielscher, 1996]): Let E be an equational theory, that is, a set of universally quantified equations. Two terms s and t are said to be *E-equal*, written $s =_E t$, iff $s = t$ is entailed by E plus the standard axioms of equality (see (25), below). A substitution σ is called an *E-unifier* of s and t iff $s\sigma =_E t\sigma$. A set $cU_E(s,t)$ of E-unifiers of s and t is called *complete* if it contains, for each E-unifier of s and t, a more or equally general substitution.[17] A consistent set of formulas E^* is then called *unification complete* wrt. E iff E^* contains the following:

1. The axioms in E.

2. The standard equality axioms, viz.

$$\begin{array}{lll}
x = x & \text{(reflexivity)} & \\
x = y \;\supset\; y = x & \text{(symmetry)} & \\
x = y \wedge y = z \;\supset\; x = z & \text{(transitivity)} & (25) \\
x_i = y \;\supset\; f(x_1,\ldots,x_i,\ldots,x_n) = f(x_1,\ldots,y,\ldots,x_n) & \text{(substitutivity I)} & \\
x_i = y \;\supset\; [P(x_1,\ldots,x_i,\ldots,x_n) \equiv P(x_1,\ldots,y,\ldots,x_n) & \text{(substitutivity II)} &
\end{array}$$

   for each n-place function symbol f and predicate P, and for each $1 \leq i \leq n$. All variables are universally quantified.

3. Equational formulas, i.e., formulas with "$=$" as the only predicate, such that for any two terms s and t with variables $\tilde{x}$ the following holds:

   (a) If s and t are not E-unifiable, then $E^* \models \neg\exists\tilde{x}.\, s \doteq t$.

   (b) If s and t are E-unifiable, then for each complete set of unifiers $cU_E(s,t)$ we have

$$E^* \models \forall\tilde{x}\left[\, s = t \;\supset \bigvee_{\sigma\in cU_E(s,t)} \exists\tilde{y}.\,\sigma_= \,\right] \tag{26}$$

   where $\tilde{y}$ denotes the variables which occur in $\sigma_=$ but not in $\tilde{x}$.[18]

As shown in [Hölldobler and Thielscher, 1995], a unification complete theory for our axioms AC1 can be obtained by computing, for each two terms s, t, some complete set $cU_{\text{AC1}}(s,t)$ of AC1-unifiers (see, e.g., [Stickel, 1981; Büttner, 1986]) and taking the corresponding equational formula which is to the right of the entailment symbol in (26). In what follows, this theory will be called *extended unique name assumption*, abbreviated *EUNA*. As an example, consider the terms $\overline{heavy(x)} \circ z$ and $\mathit{in\text{-}pipe}(po) \circ \overline{heavy(po)} \circ \mathit{clog}$. The singleton $\{\{x \mapsto po, z \mapsto \mathit{in\text{-}pipe}(po) \circ \mathit{clog}\}\}$ is a complete set of AC1-unifiers of these terms. According to (26), *EUNA* thus entails

$$\forall x, z\, [\, \overline{heavy(x)} \circ z \;=\; \mathit{in\text{-}pipe}(po) \circ \overline{heavy(po)} \circ \mathit{clog} \;\supset\; x = po \wedge z = \mathit{in\text{-}pipe}(po) \circ \mathit{clog}\,]$$

[17] That is, whenever $s\sigma =_E t\sigma$ then there exists some $\sigma' \in cU_E(s,t)$ such that $(\sigma' \leq_E \sigma)|_{\mathcal{V}ar(s)\cup\mathcal{V}ar(t)}$. Here, $\mathcal{V}ar(t)$ denotes the set of variables occurring in term t, and $(\sigma' \leq_E \sigma)|_V$ means the existence of a substitution θ such that $(\sigma'\theta =_E \sigma)|_V$. The latter holds iff for each variable $x \in V$, the two terms $(x\sigma')\theta$ and $x\sigma$ are E-equal.

[18] By $\sigma_=$ we denote the equational formula $x_1 = t_1 \wedge \ldots \wedge x_n = t_n$ constructed from the substitution $\sigma = \{x_1 \mapsto t_1, \ldots, x_n \mapsto t_n\}$.

The following crucial properties of *EUNA* show how, respectively, the subset relation and the set difference and union operations can be modeled on the term level. This will later be exploited when manipulating state terms according to action laws and causal relationships.

Proposition 16 [Thielscher, 1997] *Let A, B be two sets of fluent literals.*

1. *IF $A \subseteq B$ then $EUNA \models \exists z.\, \tau_A \circ z = \tau_B$, else $EUNA \models \forall z.\, \tau_A \circ z \neq \tau_B$.*
2. *If $A \subseteq B$ then $EUNA \models \forall z \, [\, \tau_A \circ z = \tau_B \;\equiv\; z = \tau_{A \setminus B} \,]$.*
3. *If $A \cap B = \{\}$ then $EUNA \models \forall z \, [\, z = \tau_A \circ \tau_B \;\equiv\; z = \tau_{A \cup B} \,]$.*

The fluent calculus is based on a many-sorted logic language, here consisting of five sorts, namely, fluent literals, collections (of fluent literals), actions, sequences of actions, and entities.[19] Collections are composed of fluent literals, the constant $\emptyset$, and our connection function $\circ$. Variables of the sort "fluent literal" are indicated by ℓ, variables of the sort "action" by a, variables of the sort "sequence of actions" by a^*, and variables of the sort "entity" by x, sometimes with subscripts. All other variables are of the sort "collection." Free variables are implicitly assumed to be universally quantified.

The following two foundational axioms determine the constitutional properties of state terms:

$$Holds(\ell, s) \;\equiv\; \exists z.\, \ell \circ z = s \tag{27}$$

$$State(s) \;\equiv\; \forall \ell \, [\, Holds(\ell, s) \equiv \neg Holds(\overline{\ell}, s) \,] \wedge \forall \ell, z.\, s \neq \ell \circ \ell \circ z \tag{28}$$

In words, $Holds(\ell, s)$ is true if ℓ occurs in s ; and s represents a state if it contains each fluent literal or its negation but not both, and if no fluent literal occurs twice (or more) in s. This formalization has been proved adequate in the following sense:

Proposition 17 [Thielscher, 1997] *Let s be a collection of fluent literals, then $EUNA, (27), (28) \models State(s)$ iff there exists some state S such that $EUNA \models s = \tau_S$, else $EUNA, (27), (28) \models \neg State(s)$.*

Based on the extended notion of a fluent, fluent formulas may now quantify over entities.

Definition 18 The set of *fluent formulas* is inductively defined as follows: Each fluent expression and $\top$ and $\bot$ are fluent formulas, and if F and G are fluent formulas then so are $F \wedge G$, $F \vee G$, $F \supset G$, $F \equiv G$, $\exists x.\, F$, and $\forall x.\, F$ (where $x \in \mathcal{V}$).

A *closed* formula is a fluent formula without free variables, that is, where each occurring variable is bound by some quantifier. Let S be a state and F a closed fluent formula, then the notion of F being *true* (resp. *false*) in S is inductively defined as follows:

1. $\top$ is true and $\bot$ is false in S ;
2. a fluent literal ℓ is true in S iff $\ell \in S$;
3. $F \wedge G$ is true in S iff F and G are true in S ;
4. $F \vee G$ is true in S iff F or G is true in S (or both);
5. $F \supset G$ is true in S iff F is false in S or G is true in S (or both);

[19] Under the extended notational expressiveness, actions are composed of action names and entities, such as $put(po)$; see Definition 20 below for a precise definition.

6. $F \equiv G$ is true in S iff F and G are true in S, or else F and G are false in S;

7. $\exists x. F$ is true in S iff there exists some $e \in \mathcal{E}$ such that $F\{x \mapsto e\}$ is true in S;

8. $\forall x. F$ is true in S iff for each $e \in \mathcal{E}$, $F\{x \mapsto e\}$ is true in S.

Here, $F\{x \mapsto o\}$ denotes the fluent formula resulting from replacing in F all free occurrences of x by o. ■

The encoding of fluent formulas in the fluent calculus is straightforward. To state that a fluent formula is true in a state represented by some term s, each fluent literal ℓ occurring in this formula is replaced by the expression $\mathit{Holds}(\ell, s)$; for an example see (34), below. For notational convenience, we will write $\mathit{Holds}(F, s)$ to denote this encoding of a fluent formula F. Given the definition of *Holds*, (27), and the extended unique name assumption, this encoding is correct:

Proposition 19 [Thielscher, 1997] *Let F be a fluent formula and S a state, then $\mathit{EUNA}, (27) \models \mathit{Holds}(F, \tau_S)$ iff F is true in S, else $\mathit{EUNA}, (27) \models \neg \mathit{Holds}(F, \tau_S)$.*

In particular, we call *possible* a term that satisfies a given set of domain constraints $\mathcal{D}$:

$$\mathit{Possible}(s) \equiv \bigwedge_{D \in \mathcal{D}} \mathit{Holds}(D, s) \tag{29}$$

We proceed by introducing an extended notion of action laws. An action law may now contain variables, in which case it is considered representative for all of its ground instances. In what follows, the expression $\widetilde{x}$ (resp. $\widetilde{e}$) denotes a finite sequence of variables chosen from the given set $\mathcal{V}$ (resp. entities chosen from $\mathcal{E}$) of arbitrary but fixed length. If $\widetilde{x}$ is a sequence of the variables that occur free in some expression ξ, then this is written $\xi[\widetilde{x}]$. Let $\widetilde{x} = x_1, \ldots, x_n$, then a *ground instance* of some expression $\xi[\widetilde{x}]$ is obtained by applying a substitution $\theta = \{x_1 \mapsto e_1, \ldots, x_n \mapsto e_n\}$ to ξ, where $e_1, \ldots, e_n \in \mathcal{E}$. Let $\widetilde{e} = e_1, \ldots, e_n$, then $\xi[\widetilde{x}]\theta$ is also denoted by $\xi[\widetilde{e}]$.

Definition 20 Let $\mathcal{A}$ be a set of action names, each of which is associated with a natural number called *arity*. If $a \in \mathcal{A}$ of arity $n \geq 0$ and $\widetilde{e} = e_1, \ldots, e_n$ is a sequence of entities, then the expression $a(\widetilde{e})$ is an *action*.

An *action law* is a triple $\langle C[\widetilde{x}], a(\widetilde{x}), E[\widetilde{x}] \rangle$ where $C[\widetilde{x}]$ and $E[\widetilde{x}]$ are sets of fluent expressions and $a \in \mathcal{A}$ is of arity equal to the length of $\widetilde{x}$. It is assumed that $|C[\widetilde{e}]| = |E[\widetilde{e}]|$ for any sequence $\widetilde{e}$ of entities.

If S is a state, then a ground instance $\alpha[\widetilde{e}]$ of an action law $\alpha[\widetilde{x}] = \langle C[\widetilde{x}], a(\widetilde{x}), E[\widetilde{x}] \rangle$ is *applicable* in S iff $C[\widetilde{e}] \subseteq S$. The *application* of $\alpha[\widetilde{e}]$ to S yields $(S \setminus C[\widetilde{e}]) \cup E[\widetilde{e}]$. ■

A set of action laws $\mathcal{L} = \{\langle C_1[\widetilde{x}_1], a_1(\widetilde{x}_1), E_1[\widetilde{x}_1] \rangle, \ldots, \langle C_n[\widetilde{x}_n], a_n(\widetilde{x}_n), E_n[\widetilde{x}_n] \rangle\}$ is encoded by the following formula:

$$\mathit{Action}(c, a, e) \equiv \bigvee_{i=1}^{n} \exists \widetilde{x}_i \left[c = \tau_{C_i[\widetilde{x}_i]} \wedge a = a_i(\widetilde{x}_i) \wedge e = \tau_{E_i[\widetilde{x}_i]} \right] \tag{30}$$

Similar to the case of action laws, we may exploit the extended notational expressiveness to formulate causal relationships with variables in their components. These relationships are then considered representatives for all of their ground instances.

Definition 21 A *causal relationship* is an expression of the form ε `causes` ϱ `if` Φ where Φ is a fluent formula and ε and ϱ are (possibly negated) fluent expressions.

Let (S, E) be a pair consisting of a state S and a set of fluent literals E. Furthermore, let $r = \varepsilon$ `causes` ϱ `if` Φ be a causal relationship, and let $\tilde{x}$ denote a sequence of all free variables occurring in ε, ϱ, or Φ. Then an instance $r[\tilde{e}]$ is *applicable* to (S, E) iff $S \models \Phi[\tilde{e}] \wedge \overline{\varrho[\tilde{e}]}$ and $\varepsilon[\tilde{e}] \in E$. Its application yields the pair (S', E') where $S' = (S \setminus \{\overline{\varrho[\tilde{e}]}\}) \cup \{\varrho[\tilde{e}]\}$ and $E' = (E \setminus \{\overline{\varrho[\tilde{e}]}\}) \cup \{\varrho[\tilde{e}]\}$.

Let $\mathcal{A}$ be a set of action names, $\mathcal{L}$ a set of action laws, $\mathcal{D}$ a set of domain constraints, and $\mathcal{R}$ a set of causal relationships. Furthermore, let S be a state satisfying $\mathcal{D}$, $a \in \mathcal{A}$ of arity m, and $\tilde{e}$ a sequence of entities of length m. A state S' is a *successor state* of S and $a(\tilde{e})$ iff there exists an applicable (wrt. S) instance $\alpha[\tilde{e}]$ of an action law $\alpha[\tilde{x}] = \langle C[\tilde{x}], a(\tilde{x}), E[\tilde{x}] \rangle \in \mathcal{L}$ such that

1. $((S \setminus C[\tilde{e}]) \cup E[\tilde{e}], E[\tilde{e}]) \overset{*}{\leadsto}_{\mathcal{R}} (S', E')$ for some E', and

2. S' satisfies $\mathcal{D}$.

■

The encoding of a set of causal relationships $\mathcal{R} = \{r_1[\tilde{x}_1] = \varepsilon_1$ `causes` ϱ_1 `if` $\Phi_1, \ldots, r_n[\tilde{x}_n] = \varepsilon_n$ `causes` ϱ_n `if` $\Phi_n\}$ in the fluent calculus follows this definition. We introduce a predicate $\mathit{Causes}(s, e, s', e')$ which is intended to be true iff there is an instance of a causal relationship in $\mathcal{R}$ which is applicable to (S, E) and whose application yields (S', E')—where s, e, s', e' are term representations of S, E, S', E':

$$\mathit{Causes}(s, e, s', e') \;\equiv\; \bigvee_{i=1}^{n} \exists \tilde{x}_i \left\{ \begin{array}{c} \mathit{Holds}(\Phi_i \wedge \overline{\varrho_i}, s) \;\wedge\; \exists z\, (\, \overline{\varrho_i} \circ z = s \;\wedge\; s' = z \circ \varrho_i \,) \\ \wedge \\ \exists v.\ \varepsilon_i \circ v = e \\ \wedge \\ \left[\begin{array}{c} \forall w.\ \overline{\varrho_i} \circ w \neq e \;\wedge\; e' = e \circ \varrho_i \\ \vee \\ \exists w\, (\, \overline{\varrho_i} \circ w = e \;\wedge\; e' = w \circ \varrho_i \,) \end{array} \right] \end{array} \right\} \tag{31}$$

The first row in the right hand side of the formula encodes the two conditions $\Phi_i \wedge \overline{\varrho_i}$ be true in S and $S' = (S \setminus \{\overline{\varrho_i}\}) \cup \{\varrho_i\}$. The second row represents the condition $\varepsilon_i \in E$. Finally, to model that $E' = (E \setminus \{\overline{\varrho_i}\}) \cup \{\varrho_i\}$, two cases need to be distinguished: If $\overline{\varrho_i} \notin E$, then we just have to add ϱ_i to the corresponding term e (third row). If, on the other hand, $\overline{\varrho_i} \in E$, then we have to additionally remove the sub-term $\overline{\varrho_i}$ from e (fourth row).

According to Definition 21, a successor state is obtained from the respective intermediate state by repeatedly applying causal relationships until a state results that does not violate the domain constraints. In order to formalize this in our fluent calculus, we define a predicate $\mathit{Ramify}(s, e, s')$. It is intended 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, s' are term representations of S, E, S'. This essentially requires to construct the transitive closure of the Causes relation. As this cannot be expressed in first-order logic, we use the standard way of encoding transitive closure using a second-order formula:

$$
\mathit{Ramify}(s,e,s') \;\equiv\; \forall\Pi \left\{ \begin{array}{c} \forall s_1, e_1.\ \Pi(s_1,e_1,s_1,e_1) \\ \wedge \\ \left[\begin{array}{l} \forall s_1,e_1,s_2,e_2,s_3,e_3 \\ \quad (\,\Pi(s_1,e_1,s_2,e_2) \wedge \mathit{Causes}(s_2,e_2,s_3,e_3) \\ \qquad\qquad \supset\ \Pi(s_1,e_1,s_3,e_3)\,) \end{array} \right] \\ \supset \\ \exists e'.\ \Pi(s,e,s',e') \end{array} \right\} \tag{32}
$$

$$
\wedge\ \mathit{Possible}(s')
$$

That is, $\mathit{Ramify}(s,e,s')$ is true if there is some e' such that (s,e,s',e') is in the transitive closure of Causes , and if s' satisfies the domain constraints according to formula (29).

Finally, an instance $\mathit{Successor}(s,a,s')$ shall be true if s' represents a successor state of action a and the state represented by s :

$$
\mathit{Successor}(s,a,s') \;\equiv\; \exists c,e,z\,[\,\mathit{Action}(c,a,e) \wedge c \circ z = s \wedge \mathit{Ramify}(z \circ e, e, s')\,] \tag{33}
$$

Notice that the first equation, $c \circ z = s$, ensures that the condition of the action law at hand be contained in the state represented by s (c.f. clause 1 of Proposition 16). This equation also guarantees that z contains all fluent literals in s but not in c (c.f. clause 2 of Proposition 16). Thus, $z \circ e$ represents the state resulting from the application of an action law (c.f. clause 3 of Proposition 16); hence, the pair $(z \circ e, e)$ constitutes the starting point of the ramification process (32).

To summarize, let $\mathcal{FC}_{\mathit{ramif}}$ denote the union of EUNA with the definitions of *Holds* (27), *Possible* (29), *Action* (30), *Causes* (31), *Ramify* (32), and *Successor* (33), based on given sets of domain constraints, action laws, and causal relationships. The following correctness result is known:

Theorem 22 [Thielscher, 1997] *Let $\mathcal{FC}_{\mathit{ramif}}$ be the encoding of sets of, respectively, domain constraints, action laws, and causal relationships. Furthermore, let S, S' be two states, a an action name of arity m , and $\widetilde{e}$ a sequence of entities of length m . Then*

$$
\mathcal{FC}_{\mathit{ramif}} \;\models\; \mathit{Successor}(s, a(\widetilde{e}), s')
$$

iff there is a successor state S' of S and $a(\widetilde{e})$ such that $\mathit{EUNA} \models s' = \tau_{S'}$, else

$$
\mathcal{FC}_{\mathit{ramif}} \;\models\; \neg\mathit{Successor}(s, a(\widetilde{e}), s')
$$

Example 7 We take the singleton set of entities $\mathcal{E} = \{\mathit{po}\}$ (whose element represents a particular potato) along with the unary fluents *in-pipe* and *heavy* plus the nullary fluent *clog* , each of which shall belong to $\mathcal{F}_{ab}$. We also define the unary action name *put* to describe the insertion of an object into the tail pipe. The various fluents are related through these domain constraints:

$$
\mathit{Possible}(s) \;\equiv\; \left\{ \begin{array}{c} \exists x.\ \mathit{Holds}(\mathit{in\text{-}pipe}(x),s) \supset \mathit{Holds}(\mathit{clog},s) \\ \wedge \\ \forall x\,[\,\mathit{Holds}(\mathit{disq}(\mathit{put}(x)),s) \equiv \mathit{Holds}(\mathit{heavy}(x),s)\,] \end{array} \right\} \tag{34}
$$

That is to say, the tail pipe is clogged whenever it houses an object, and putting an object into the tail pipe is abnormally disqualified if the former is too heavy. The domain constraints give rise to the following four causal relationships:[20]

$$\begin{array}{llll}
\mathit{in\text{-}pipe}(x) \ \texttt{causes} \ \mathit{clog} \ \texttt{if} \ \top & & \mathit{heavy}(x) \ \texttt{causes} \ \mathit{disq}(\mathit{put}(x)) \ \texttt{if} \ \top \\
\overline{\mathit{in\text{-}pipe}(x)} \ \texttt{causes} \ \overline{\mathit{clog}} \ \texttt{if} \ \forall y.\, \overline{\mathit{in\text{-}pipe}(y)} & & \overline{\mathit{heavy}(x)} \ \texttt{causes} \ \overline{\mathit{disq}(\mathit{put}(x))} \ \texttt{if} \ \top
\end{array}$$

which in turn determine the encoding of the *Causes* predicate:

$$\mathit{Causes}(s,e,s',e') \;\equiv\; \exists x \left\{ \begin{array}{c}
\mathit{Holds}(\overline{\mathit{clog}}, s) \ \wedge \ \exists z \, (\, \overline{\mathit{clog}} \circ z = s \ \wedge \ s' = z \circ \mathit{clog} \,) \\
\wedge \\
\exists v.\ \mathit{in\text{-}pipe}(x) \circ v = e \\
\wedge \\
\left[\begin{array}{c}
\forall w.\ \overline{\mathit{clog}} \circ w \neq e \ \wedge \ e' = e \circ \mathit{clog} \\
\vee \\
\exists w \, (\, \overline{\mathit{clog}} \circ w = e \ \wedge \ e' = w \circ \mathit{clog} \,)
\end{array} \right]
\end{array} \right\}$$

$$\vee \ \ldots$$

Finally, our only action is specified by

$$\mathit{Action}(c,a,e) \;\equiv\; \exists x \, [\, c = \overline{\mathit{in\text{-}pipe}(x)} \ \wedge \ a = \mathit{put}(x) \ \wedge \ e = \mathit{in\text{-}pipe}(x) \,]$$

Let $\mathcal{FC}_7$ denote the entire fluent calculus-based formalization of this domain, then we have, for instance,

$$\begin{array}{lll}
\mathcal{FC}_7 & \models & \mathit{Successor}(\, \overline{\mathit{in\text{-}pipe}(\mathit{po})} \circ \overline{\mathit{heavy}(\mathit{po})} \circ \overline{\mathit{clog}} \circ \overline{\mathit{disq}(\mathit{put}(\mathit{po}))} \, , \\
& & \quad \mathit{put}(\mathit{po}) \, , \\
& & \quad \mathit{in\text{-}pipe}(\mathit{po}) \circ \overline{\mathit{heavy}(\mathit{po})} \circ \mathit{clog} \circ \overline{\mathit{disq}(\mathit{put}(\mathit{po}))} \,)
\end{array}$$

according to Theorem 22. ■

4.2 Observations and Models

The fluent calculus encoding we arrived at in the previous section provides a formal account of successor states, including a solution to the ramification problem. Next, and prior to addressing the qualification problem, we formalize the application of whole action sequences to an (unspecified) initial state. This provides means for encoding observations. Our objective is to extend $\mathcal{FC}_{ramif}$ in such a way that there is a one-to-one correspondence between the (standard, i.e., 'classical') models of the resulting set of formulas and the models of a set of observations in the sense of Definition 13.

At the core of this extension are three new predicates named, respectively, *Qualified*, *Result*, and *Miracle*. Their intuitive meaning is the following. If an instance $\mathit{Qualified}(a^*)$ is true, then this indicates that the action sequence a^* is qualified. If an instance $\mathit{Result}(a^*, s)$ is true, then this indicates that s represents the state which would result from performing the action sequence a^* in the initial state. Finally, if an instance $\mathit{Miracle}(a^*)$ is true, then this indicates that the action sequence a^* is miraculously disqualified. As a notational convention, if a^* is a (possibly empty) action sequence and a an action, then $[a^*|a]$ denotes the action sequence

[20] See [Thielscher, 1997] for how Definition 7 can be extended to cope with domain constraints that contain quantifications.

which consists in a^* followed by a. The crucial properties of the new predicates are then determined by the following axioms:

$$\mathit{Qualified}([\,]) \wedge \neg\mathit{Miracle}([\,]) \tag{35}$$

$$\mathit{Qualified}([a^*|a]) \equiv \mathit{Qualified}(a^*) \wedge \neg\mathit{Miracle}([a^*|a]) \wedge \exists s, s' \left[\begin{array}{l} \mathit{Result}(a^*, s) \wedge \\ \mathit{Holds}(\overline{\mathit{disq}(a)}, s) \wedge \\ \mathit{Successor}(s, a, s') \end{array} \right] \tag{36}$$

$$\mathit{Result}([a^*|a], s') \supset \forall s\, [\, \mathit{Result}(a^*, s) \supset \mathit{Successor}(s, a, s')\,] \tag{37}$$

The reading of the topmost conjunction, (35), is obvious. Formula (36) states, in words, that an action sequence $[a^*|a]$ is qualified if so is a^*, if $[a^*|a]$ is not miraculously disqualified, if the result s of performing a^* does not entail an abnormal disqualification as regards a, and if there exists a successor s' of s and a. The implication in (37) ensures that s' can only be the result of performing a sequence $[a^*|a]$ if s' is a possible successor when executing a in the state resulting from performing a^*. Notice, however, that (36) and (37) do not entail the existence of a resulting state whenever the corresponding action sequence is qualified, nor do they entail uniqueness of resulting states. We therefore need to add the following:

$$\exists s.\, \mathit{Result}(a^*, s) \equiv \mathit{Qualified}(a^*) \tag{38}$$

$$\mathit{Result}(a^*, s) \wedge \mathit{Result}(a^*, s') \supset s = s' \tag{39}$$

Finally, the term intended to represent the initial state should qualify as such, namely, both in being a proper state term and in satisfying the domain constraints, that is,

$$\mathit{Result}([\,], s) \supset \mathit{State}(s) \wedge \mathit{Possible}(s) \tag{40}$$

The reader may notice that the formulas (35)–(40) are domain-independent axioms.

Example 7 (continued) Let us consider the set of formulas $\mathcal{FC}_7 \cup \{(35)\text{–}(40)\}$, and suppose that in addition we are given the fact $\neg\mathit{Qualified}([\mathit{put}(\mathit{po})])$. According to (36) and (35), the latter entails

$$\mathit{Miracle}([\mathit{put}(\mathit{po})]) \vee \forall s, s' \left[\begin{array}{l} \neg\mathit{Result}([\,], s) \vee \\ \neg\mathit{Holds}(\overline{\mathit{disq}(\mathit{put}(\mathit{po}))}, s) \vee \\ \neg\mathit{Successor}(s, \mathit{put}(\mathit{po}), s') \end{array} \right]$$

From (35), (38), and (40) we know that $\exists s\, [\, \mathit{Result}([\,], s) \wedge \mathit{State}(s)\,]$ is always true, and from Theorem 22 we further conclude that $\mathcal{FC}_7 \models \mathit{State}(s) \supset \forall s'.\, \neg\mathit{Successor}(s, \mathit{put}(\mathit{po}), s')$ iff $\mathcal{FC}_7 \models \mathit{Holds}(\mathit{in\text{-}pipe}(\mathit{po}), s)$ since $\overline{\mathit{in\text{-}pipe}(\mathit{po})}$ is the only strict precondition of $\mathit{put}(\mathit{po})$. Hence, the above implies

$$\mathit{Miracle}([\mathit{put}(\mathit{po})]) \vee \forall s\, [\, \mathit{Result}([\,], s) \supset \mathit{Holds}(\mathit{disq}(\mathit{put}(\mathit{po})), s) \vee \mathit{Holds}(\mathit{in\text{-}pipe}(\mathit{po}), s)\,] \tag{41}$$

That is to say, $\mathit{put}(\mathit{po})$ being disqualified in the initial state either implies a miraculous disqualification, an abnormal disqualification, or that there already is a potato in the tail pipe. ■

In what follows, we prove that by adding the above formulas we achieve what we have promised. Suppose given a domain with causal model Σ, and let $\mathcal{FC}_{ramif}$ denote its encoding as described in Section 4.1. If ι is a model of the formulas $\mathcal{FC}_{ramif} \cup \{(35)\text{–}(40)\}$ and

(Res, Σ, Υ) an interpretation for the domain at hand, then we say that ι and (Res, Σ, Υ) *correspond* iff for all action sequences a^*, states S, and collections of fluent literals s such that $EUNA \models s = \tau_S$, we find that[21]

$$\begin{aligned} Res(a^*) = S \quad &\text{iff} \quad [Result(a^*, s)]^\iota \text{ is true} \\ a^* \in \Upsilon \quad &\text{iff} \quad [Miracle(a^*)]^\iota \text{ is true} \end{aligned} \tag{42}$$

Then we can prove the following:

Theorem 23 *Let $\mathcal{FC}_{ramif}$ be the encoding of a domain description with causal model Σ, then for each model ι of $\mathcal{FC}_{ramif} \cup \{(35)–(40)\}$ there exists a corresponding interpretation (Res, Σ, Υ) and vice versa.*

Proof: See appendix.

This result shows the adequacy of the domain-independent axioms (35)–(40) as regards the formal notions both of action sequences being qualified and of states resulting from performing action sequences. Next, we concentrate on the domain-specific expressions which are based on these concepts, namely, the observations. Their formalization in our calculus is straightforward. An observation of the form F `after` $[a_1, \ldots, a_n]$ is encoded by

$$\exists s\ [\ Result([a_1, \ldots, a_n], s) \wedge Holds(F, s)\ ] \tag{43}$$

That is to say, the action sequence $[a_1, \ldots, a_n]$ must admit a resulting state in which, moreover, fluent formula F holds. An observation of the form a `unqualified after` $[a_1, \ldots, a_n]$ is encoded by

$$Qualified([a_1, \ldots, a_n]) \wedge \neg Qualified([a_1, \ldots, a_n, a]) \tag{44}$$

That is to say, the action sequence $[a_1, \ldots, a_n]$ must be qualified while $[a_1, \ldots, a_n, a]$ must not so. The addition of these formulas $\mathcal{FC}_{ramif} \cup \{(35)–(40)\}$ automatically restricts the set of classical models to those which correspond to interpretations in which the respective observations hold.

Example 7 (continued) Suppose given the two observations

$$\begin{aligned} &put(po)\ \texttt{disqualified after}\ [\,] \\ &\overline{clog}\ \texttt{after}\ [\,] \end{aligned} \tag{45}$$

We have already seen that the encoding of the first one, viz. $Qualified([\,]) \wedge \neg Qualified([put(po)])$, entails the disjunction (41). The encoding of the second observation, viz.

$$\exists s\,[\,Result([\,], s) \wedge Holds(\overline{clog}, s)\,]$$

implies $\forall s\,[\,Result([\,], s) \supset \neg\exists x.\, Holds(in\text{-}pipe(x), s)\,]$ according to (40) and the underlying domain constraints, (34). Given this, (41) can be strengthened to

$$Miracle([put(po)]) \vee \forall s\,[\,Result([\,], s) \supset Holds(disq(put(po)), s)\,] \tag{46}$$

Correspondingly, every model (Res, Σ, Υ) of the observations (45) satisfies $[put(po)] \in \Upsilon$ or $disq(put(po)) \in Res([\,])$, or both. ■

[21] Below, by "$[P(t_1, \ldots, t_n)]^\iota$ is true" we mean that the n-tuple $(t_1^\iota, \ldots, t_n^\iota)$ is member of the relation which ι assigns to predicate P, where t_i^ι ($1 \leq i \leq n$) denotes the element of ι's universe to which ι maps term t_i.

Let, given a domain description, $W_{\mathcal{FC}}$ denote its fluent calculus encoding consisting of the formulas $\mathcal{FC}_{ramif}$, the axioms (36)–(40), and the formulas representing the underlying observations as in (43) and (44). This encoding is correct according to Definition 13.

Theorem 24 *Let $W_{\mathcal{FC}}$ be the encoding of a domain description with causal model Σ and observations $\mathcal{O}$, then for each model ι of $W_{\mathcal{FC}}$ there exists a corresponding model (Res, Σ, Υ) of $\mathcal{O}$ and vice versa.*

> **Proof:** Let ι be a model of $\mathcal{FC}_{ramif} \cup \{(36)\text{–}(40)\}$ and (Res, Σ, Υ) a corresponding interpretation. Given Theorem 23, it suffices to show that ι is a model of (43) and (44) iff the respective observations hold in (Res, Σ, Υ).
>
> 1. By definition, an observation F `after` $[a_1, \ldots, a_n]$ holds in (Res, Σ, Υ) iff $Res([a_1, \ldots, a_n])$ is defined and F is true in that state. This in turn is equivalent to ι being model of (43) according to (38), Proposition 19, and the fact that ι and (Res, Σ, Υ) correspond.
> 2. By definition, an observation a `unqualified after` $[a_1, \ldots, a_n]$ holds in (Res, Σ, Υ) iff $Res([a_1, \ldots, a_n])$ is defined but $Res([a_1, \ldots, a_n, a])$ is not. This in turn is equivalent to ι being model of (44) given that ι and (Res, Σ, Υ) correspond.
>
> ■

4.3 Fluent Calculus and Qualification

We have now reached the stage where we concern ourselves with the qualification problem in the fluent calculus. Since nonmonotonicity is an intrinsic feature of the qualification problem, the encoding we arrived at will be embedded in a nonmonotonic framework. Notice that the fluent calculus provides a *monotonic* solution both to the frame problem as well as to the ramification problem. For the set $W_{\mathcal{FC}}$ is composed only of axioms in classical logic, and these formulas are interpreted in the standard way. This property of the fluent calculus is of advantage when additionally coping with the qualification problem because it avoids possible unintended interferences among different forms of nonmonotonicity employed to tackle different problems within a single formalism.

The nonmonotonic extension of the fluent calculus we propose in the sequel is based on the machinery of *Default Logic* [Reiter, 1980]. More precisely, the formulas $W_{\mathcal{FC}}$ are taken as the foundational axioms of the default theory to be constructed, and so-called default rules are used to express the necessary defeasible assumptions of normality. Namely, by default an 'abnormality' fluent $f_{ab}(e_1, \ldots, e_n)$ is false in the initial state and a sequence of actions is not miraculously disqualified. Since the latter default assumption needs to be preferred in case of conflicts, a special variant of Default Logic is required which is capable of dealing with priorities among default rules. In what follows, we first recall the basics of standard Default Logic; thereafter, we give a brief but sufficiently detailed introduction to the aforementioned variant, namely, *Prioritized Default Logic* [Brewka, 1994]; and finally, we use these concepts to solve the qualification problem in the fluent calculus.

Default Logic

The fundamental idea in Default Logic is to extend classical logic, which is taken to encode precise knowledge, by expressions that formalize somehow vague, defeasible knowledge. Called *default rules* (or *defaults*, for short), these expressions allow for stating that some property α 'normally' implies some property ω. The reference to normality is made precise by specifying circumstances $\neg\beta$ which must *provably* hold in order that the conclusion from α to ω shall not be valid. The formal syntax of a default is

$$\frac{\alpha \,:\, \beta}{\omega}$$

where α (the *prerequisite*), β (the *justification*), and ω (the *consequence*) all are formulas in classical logic. For our purpose, it suffices to only consider defaults which are of the form $\frac{\alpha\,:\,\omega}{\omega}$, i.e., where justification and consequence coincide. Any such default may informally be regarded as a deduction rule stating that "If α has already been deduced and ω is consistent with (i.e., does not contradict) what has been deduced so far, then conclude ω." E.g., the default

$$\frac{:\, \forall s\, [\, \mathit{Result}([\,], s) \supset \neg \mathit{Holds}(\mathit{disq}(\mathit{put}(\mathit{po})), s)\,]}{\forall s\, [\, \mathit{Result}([\,], s) \supset \neg \mathit{Holds}(\mathit{disq}(\mathit{put}(\mathit{po})), s)\,]} \tag{47}$$

states that if it is consistent to assume that there be no abnormal disqualification of the action $\mathit{put}(\mathit{po})$ in the initial state, then this very conclusion is to be drawn. In case one or more components of a default contain free variables, the default is considered representative for all of its ground instances. The default (50) below, for example, in which the variable a^* occurs free, stands for all the assumptions that a particular action sequence normally is not miraculously disqualified, e.g.,

$$\frac{:\, \neg \mathit{Miracle}([\mathit{put}(\mathit{po})])}{\neg \mathit{Miracle}([\mathit{put}(\mathit{po})])} \tag{48}$$

A *default theory* $\Delta = (D, W)$ consists of a set of defaults D and a set of closed formulas W, the latter of which is called *world-* (or *background*) *knowledge*. Reasoning in default theories is based on the formation of so-called extensions. The idea is to start with the background knowledge, W, and to successively apply defaults chosen from D, that is, to add their consequences provided their justification is consistent with what is finally obtained as extension. Once there are no more applicable defaults left, the deductive closure of the resulting set of formulas constitutes an extension. An extension may be regarded as one conceivable view on the state of affairs. The formal definition is as follows:

Definition 25 [Reiter, 1980] Let $\Delta = (D, W)$ be a default theory, and let E be a set of closed formulas. We define[22]

1. $\Gamma_0 := W$
2. $\Gamma_i := \mathit{Th}(\Gamma_{i-1}) \cup \{\omega : \frac{\alpha\,:\,\omega}{\omega} \in D \text{ and } \alpha \in \Gamma_{i-1} \text{ and } \neg\omega \notin E\}$, for $i = 1, 2, \ldots$

Then E is an *extension* of Δ iff $E = \bigcup_{i=0}^{\infty} \Gamma_i$. ■

A default theory may admit multiple extensions, each of which is obtained by applying different subsets of the underlying defaults. Then a closed formula is said to be *skeptically entailed* in a default theory iff it is contained in all extensions of the latter. Suppose, as an example, $W = \{\mathit{Miracle}([\mathit{put}(\mathit{po})]) \vee \exists s\, [\, \mathit{Result}([\,], s) \wedge \mathit{Holds}(\mathit{disq}(\mathit{put}(\mathit{po})), s)\,]\}$ (c.f. (46)) and $D = \{(47), (48)\}$, then the default theory (D, W) determines two extensions, one of which includes $\mathit{Miracle}([\mathit{put}(\mathit{po})]) \wedge \forall s\, [\, \mathit{Result}([\,], s) \supset \neg \mathit{Holds}(\mathit{disq}(\mathit{put}(\mathit{po})), s)\,]$ while the other one

[22] Below, $\mathit{Th}(\Psi)$ denotes the deductive closure of the set of formulas Ψ, that is, $\mathit{Th}(\Psi) := \{\psi : \Psi \models \psi\}$.

includes $\neg\mathit{Miracle}([\mathit{put}(\mathit{po})]) \wedge \exists s\,[\,\mathit{Result}([\,],s) \wedge \mathit{Holds}(\mathit{disq}(\mathit{put}(\mathit{po})),s)\,]$. The reason is that after adding to W the consequence of (47), say, the resulting set of formulas is inconsistent with $\neg\mathit{Miracle}([\mathit{put}(\mathit{po})])$, which blocks the application of (48). Notice that there is no formal preference between the two extensions. Consequently, $\neg\mathit{Miracle}([\mathit{put}(\mathit{po})])$ is not skeptically entailed. This, however, would be desirable in this example since miraculous disqualifications ought to be primarily minimized. In order to accomplish this, we employ a suitable extension of classical Default Logic.

Prioritized Default Logic

When constructing extensions of a default theory according to Definition 25, all defaults are applied with the same priority. Yet an adequate solution to the qualification problem requires that, in case of conflicts, minimizing miraculous disqualifications is to be preferred over minimizing 'abnormality' fluents. A recent variant of Default Logic, namely, *Prioritized Default Logic* [Brewka, 1994], serves this purpose by supporting the specification of (possibly partial) preference orderings among defaults. This ordering is exploited to select among the extensions of a default theory those in which the most preferred defaults have been applied. In what follows, we adopt a formalization of Prioritized Default Logic proposed in a subsequent paper.

Definition 26 [Rintanen, 1995] A *prioritized default theory* is a triple $(D,W,<)$ where D and W are as in classical Default Logic and $<$ is a partial ordering on D.

If E is a closed set of formulas, then a default $\frac{\alpha\,:\,\omega}{\omega}$ is said to be *applied in* E iff $\alpha,\omega \in E$. Let $\Delta = (D,W,<)$ be a prioritized default theory, then an extension E of the (standard) default theory (D,W) is a *prioritized extension* of Δ iff there is a strict total ordering $\ll$ extending $<$ such that the following holds for all extensions E' of (D,W) and all defaults $\delta' \in D$: If δ' is applied in $E' \setminus E$, then there is some $\delta \ll \delta'$ which is applied in $E \setminus E'$. ■

In words, a standard extension E is prioritized if we can find a total ordering respecting $<$ such that the following is true: Whenever some default δ' is not applied in E but in some other standard extension E', then there is also a default δ which is applied in E but not in E' and which has higher priority than δ' according to the total ordering. Recall, for instance, our example default theory (D,W) discussed right after Definition 25. If we define $(48) < (47)$, then the prioritized default theory $(D,W,<)$ declares only one of the two extensions of (D,W) as prioritized, viz. the one that includes $\neg\mathit{Miracle}([\mathit{put}(\mathit{po})])$. The reason is that, as regards the alternative extension, there is no 'compensation' for applying default (47) but not default (48). Hence, if we restrict attention to prioritized extension, then $\neg\mathit{Miracle}([\mathit{put}(\mathit{po})])$ is skeptically entailed, as intended.

We are now prepared for embedding the basic fluent calculus into a suitable nonmonotonic framework in view of successfully coping with the qualification problem. For a given domain description, we construct a prioritized default theory in which the formulas $W_{\mathcal{FC}}$ constitute the background knowledge. The necessary default assumptions of normality are formalized as default rules. A particular preference ordering among these defaults allows to prefer minimization of miraculously disqualified action sequences whenever this conflicts with minimizing abnormal circumstances in the initial state.

For each 'abnormality' fluent $f_{ab} \in \mathcal{F}_{ab}$, to begin with, the default rule

$$\frac{:\ \forall s\,[\,\mathit{Result}([\,],s) \supset \neg\mathit{Holds}(f_{ab}[\widetilde{x}],s)\,]}{\forall s\,[\,\mathit{Result}([\,],s) \supset \neg\mathit{Holds}(f_{ab}[\widetilde{x}],s)\,]} \tag{49}$$

is used to express the default assumptions that a ground instance $f_{ab}[\widetilde{e}]$ be false in the initial state. That is to say, as long as it is consistent to assume that $f_{ab}[\widetilde{e}]$ does not hold in the initial state, do it. The various assumptions of normality as regards miraculous disqualifications are formalized by the default rule

$$\frac{:\ \neg \mathit{Miracle}(a^*)}{\neg \mathit{Miracle}(a^*)} \tag{50}$$

That is to say, as long as it is consistent to assume that a particular sequence of actions is not miraculously disqualified, do it. Since miraculous disqualifications are to be minimized with higher priority than initial truth of 'abnormality' fluents, we define a partial ordering $<_{\mathcal{FC}}$ as follows: For any action sequence a^* and any ground instance of an element of $\mathcal{F}_{ab}$, we have $(50) <_{\mathcal{FC}} (49)$.

This completes our action calculus for domain descriptions involving potential abnormal disqualifications of actions. The prioritized default theory $\Delta_{\mathcal{FC}} = (D_{\mathcal{FC}}, W_{\mathcal{FC}}, <_{\mathcal{FC}})$ constitutes a solution to the qualification problem in the fluent calculus.

Example 7 (continued) Let $\Delta_{\mathcal{FC}_7} = (D_{\mathcal{FC}_7}, W_{\mathcal{FC}_7}, <_{\mathcal{FC}})$ be the fluent calculus encoding of the domain considered in this example, including the observations (45). In particular, $D_{\mathcal{FC}_7}$ contains instances of the default (49) for each of *in-pipe*(po), *heavy*(po), *clog*, and *disq*$(\mathit{put}(po))$. As we have seen, $W_{\mathcal{FC}_7}$ entails the disjunction (46). This implies that the defaults

$$\frac{:\ \neg \mathit{Miracle}([\mathit{put}(po)])}{\neg \mathit{Miracle}([\mathit{put}(po)])} \quad \text{and} \quad \frac{:\ \forall s\, [\, \mathit{Result}([\,], s) \supset \neg \mathit{Holds}(\mathit{disq}(\mathit{put}(po)), s)\,]}{\forall s\, [\, \mathit{Result}([\,], s) \supset \neg \mathit{Holds}(\mathit{disq}(\mathit{put}(po)), s)\,]}$$

are mutually exclusive, for either consequence in conjunction with (46) implies the negation of the other rule's justification (given $(35), (39) \in W_{\mathcal{FC}_7}$). According to the priority ordering $<_{\mathcal{FC}}$, the first of these two defaults has to be preferred. Now, $W_{\mathcal{FC}_7}$ and $\forall s\, [\, \mathit{Result}([\,], s) \supset \mathit{Holds}(\mathit{disq}(\mathit{put}(po), s))\,]$ imply $\forall s\, [\, \mathit{Result}([\,], s) \supset \mathit{Holds}(\mathit{heavy}(po, s))\,]$ according to the underlying domain constraints, (34). Thus the instance $f_{ab}[\widetilde{e}] = \mathit{heavy}(po)$ of default (49), too, is not applicable. The justifications of all other defaults in $D_{\mathcal{FC}_7}$, however, are consistent with $W_{\mathcal{FC}_7} \cup \{\neg \mathit{Miracle}([\mathit{put}(po)])\}$, so that the default theory $\Delta_{\mathcal{FC}_7}$ admits a unique prioritized extension E which includes $\neg \mathit{Miracle}(a^*)$ for any action sequence a^*. Moreover, the initial state is completely determined in E since there is no fluent that does not belong to $\mathcal{F}_{ab}$. Hence, E also includes the formula

$$\forall s\, [\, \mathit{Result}([\,], s) \supset s = \overline{\mathit{in\text{-}pipe}(po)} \circ \mathit{heavy}(po) \circ \overline{\mathit{clog}} \circ \mathit{disq}(\mathit{put}(po))\,]$$

Notice that the domain description admits a unique preferred model $(\mathit{Res}, \Sigma, \Upsilon)$ which satisfies $\Upsilon = \{\}$ and $\mathit{Res}([\,]) = \{\overline{\mathit{in\text{-}pipe}(po)}, \mathit{heavy}(po), \overline{\mathit{clog}}, \mathit{disq}(\mathit{put}(po))\}$. ■

The last observation suggests a close correspondence between the models of a domain and the prioritized extensions of the domain's encoding in the fluent calculus. As the main result of this second part, we prove that this holds in general, which shows that our embedding of the fluent calculus into a (prioritized) default theory solves the qualification problem. Let $\Delta_{\mathcal{FC}}$ be the encoding of a domain description, then a prioritized extension E of $\Delta_{\mathcal{FC}}$ and an interpretation $(\mathit{Res}, \Sigma, \Upsilon)$ of this domain are said to *correspond* iff for all instances $f_{ab}[\widetilde{e}]$ wrt. $\mathcal{F}_{ab}$ and all action sequences a^*, we find that

$$\begin{array}{rcl} a^* \notin \Upsilon & \text{iff} & \neg \mathit{Miracle}(a^*) \in E \\ \overline{f_{ab}[\widetilde{e}]} \in \mathit{Res}([\,]) & \text{iff} & \forall s\, [\, \mathit{Result}([\,], s) \supset \neg \mathit{Holds}(f_{ab}[\widetilde{e}], s)\,] \in E \end{array} \tag{51}$$

This notion is used to state the correctness of our action calculus wrt. the formal characterization of the qualification problem developed in the first part of the paper.

Theorem 27 *Let $\Delta_{\mathcal{FC}}$ be the prioritized default theory encoding a domain description with observations $\mathcal{O}$, then for each prioritized extension of $\Delta_{\mathcal{FC}}$ there exists a corresponding preferred model of $\mathcal{O}$ and vice versa.*

Proof: See appendix.

An immediate consequence of this one-to-one correspondence is that, as far as observations are concerned, the notion of skeptical entailment in our action calculus and the notion of entailment suggested by our formal account of the qualification problem coincide.

Corollary 28 *Let $\Delta_{\mathcal{FC}}$ be the prioritized default theory encoding a domain description, then an observation is entailed by the domain iff the corresponding formula (i.e., (43) or (44)) is skeptically entailed in $\Delta_{\mathcal{FC}}$.*

This result completes the second part of the paper. It shows that our nonmonotonic extension of the basic fluent calculus successfully deals with potential abnormal disqualifications of actions. The resulting action calculus thus combines in a single framework solutions to three of the most recognized problems in reasoning about actions, the frame, the ramification, and the qualification problem.

5 Discussion

We have proposed a formal characterization of the qualification problem from the perspective that requiring global minimization of abnormal disqualifications is obviously inadequate. We have argued that the property of an action to be abnormally disqualified should be formalized as a fluent. These fluents are to be assumed false in the initial state unless there is evidence to the contrary. In the course of time, by virtue of being fluents, these propositions are potentially indirectly affected by the execution of actions. This accounts for the fact that unusual disqualifications may be *caused*, in which case their occurrence is not to be considered abnormal. The fact that action disqualifications might be indirect effects of actions necessitates a suitable solution to the ramification problem. Our theory moreover accommodates miraculous disqualifications, which need to be assumed whenever an action disqualification cannot be explained even if abnormal circumstances are granted. Consequently, miraculous disqualifications are minimized with higher priority.

Assuming away by default abnormal or miraculous disqualifications is an inherently nonmonotonic process. In [Lifschitz, 1993], a property called *restricted monotonicity* has been claimed generally desirable in theories of actions. A formalism possesses this property if additional observations can only increase the set of observations that are entailed by a domain description. This, however, is no longer appropriate when being confronted with the qualification problem. As a consequence, the entailment relation $\mid\!\sim_{\Sigma}$ induced by our model preference criterion is nonmonotonic.

Using a suitably simple action language, the focus in this paper has been on the qualification problem. The underlying principles of our theory, however, are sufficiently fundamental and general to not depend on this specific language. Thus these principles could equally well be employed in other, more elaborated formal theories of actions like, e.g., [Gelfond and Lifschitz, 1993;

Sandewall, 1994; Thielscher, 1995], to tackle the qualification problem. Likewise, existing action calculi may be enhanced on this basis in order that they become capable of dealing with abnormal action disqualifications. As an example formalism, in the second part of the paper we have embedded the fluent calculus in an appropriate nonmonotonic theory. The adequacy of the resulting framework has been established by relating it to our formal characterization of the qualification problem. This adds another item to the list of ontological aspects which the fluent calculus—besides being closely related, in its basic form, to the *Linear Connection Method* [Bibel, 1986] and reasoning about actions based on *Linear Logic* [Girard, 1987; Masseron *et al.*, 1993]—is capable of dealing with, such as non-deterministic and concurrent actions [Bornscheuer and Thielscher, 1997], indirect effects of actions [Thielscher, 1997], or continuous change [Herrmann and Thielscher, 1996].

Besides the proposal pursued in this paper, the only existing alternative to global minimization as a solution to the qualification problem is the concept of *chronological ignorance* [Shoham, 1987; Shoham, 1988]. Roughly speaking, the crucial idea there is to assume away, by default, abnormal circumstances, and simultaneously to prefer minimization of abnormalities at earlier timepoints.[23] Formally, the approach employs a certain kind of modal logic as a means to express the distinction between provable facts and propositions which might or might not be true. For instance, our introductory example one would formulate in the framework of [Shoham, 1987; Shoham, 1988] by these two action descriptions:

$$\Box\,\textsc{True}\,(t, \mathit{put\text{-}p}) \;\wedge\; \Diamond\,\textsc{True}\,(t, \overline{\mathit{heavy}}) \;\supset\; \Box\,\textsc{True}\,(t+1, \mathit{pot}) \tag{52}$$

$$\Box\,\textsc{True}\,(t, \mathit{start}) \;\wedge\; \Diamond\,\textsc{True}\,(t, \overline{\mathit{pot}}) \;\supset\; \Box\,\textsc{True}\,(t+1, \mathit{runs}) \tag{53}$$

where $\Box\,\textsc{True}\,(t, \ell)$ should be read as "at time t fluent literal ℓ provably holds" and $\Diamond\,\textsc{True}\,(t, \ell)$ as "at time t fluent literal ℓ may or may not hold." Thus the first of the two implications states that if it is known that the action *put-p* occurs at time t and it is possible that $\overline{\mathit{heavy}}$ holds at that time, then *pot* provably holds at time $t+1$. Likewise, if it is known that the action *start* occurs at time t and it is possible that $\overline{\mathit{pot}}$ holds at that time, then *runs* provably holds at time $t+1$. Observe how abnormal disqualifications, like $\overline{\mathit{heavy}}$ and $\overline{\mathit{pot}}$, are assumed away whenever the contrary does not provably hold. Now suppose given $\Box\,\textsc{True}\,(1, \mathit{put\text{-}p}) \wedge \Box\,\textsc{True}\,(2, \mathit{start})$. Then chronological ignorance tells us that $\Diamond\,\textsc{True}\,(1, \overline{\mathit{heavy}})$ holds since nothing is known about $\textsc{True}\,(1, \overline{\mathit{heavy}})$ itself. Hence, (52) implies $\Box\,\textsc{True}\,(2, \mathit{pot})$, which in turn gives us $\neg\Diamond\,\textsc{True}\,(2, \overline{\mathit{pot}})$.[24] Thus the antecedent of (53) is false and, consequently, the second action, *start*, cannot be successfully executed, as intended. Notice that this being the unique conclusion relies on the chronological order in which minimization is performed. Otherwise, it could equally well be concluded that $\Diamond\,\textsc{True}\,(2, \overline{\mathit{pot}})$ holds, for, in the first place, nothing is known about $\textsc{True}\,(2, \overline{\mathit{pot}})$ itself. This in turn entails $\neg\Diamond\,\textsc{True}\,(1, \overline{\mathit{heavy}})$, i.e., $\Box\,\textsc{True}\,(1, \mathit{heavy})$, since the implication (52) is logically equivalent to

$$\Box\,\textsc{True}\,(t, \mathit{put\text{-}p}) \;\wedge\; \Diamond\,\textsc{True}\,(t+1, \overline{\mathit{pot}}) \;\supset\; \neg\Diamond\,\textsc{True}\,(t, \overline{\mathit{heavy}})$$

This alternative conclusion corresponds to what we have called the counter-intuitive model but, as indicated, it is not supported by chronological ignorance.

The interesting, albeit informal, reason for chronological ignorance coming to the desired conclusion in this and similar cases is a certain respect of causality hidden in this method: By

[23] This explains the naming: Potential abnormal disqualifications are *ignored* whenever possible, and this is done in *chronological* order. Assuming away obstacles whenever their occurrence cannot be proved, Shoham also calls the *ostrich* principle, or: *what-you-don't-know-won't-hurt-you*.

[24] As usual in modal logic, $\neg\Box\,\textsc{True}\,(t, \ell)$ is equivalent to $\Diamond\,\textsc{True}\,(t, \overline{\ell})$.

minimizing chronologically, one tends to minimize causes rather than effects—which is the right thing to do—simply because in general causes precede effects. On the other hand, it has already been shown elsewhere (e.g., [Kautz, 1986; Sandewall, 1993; Stein and Morgenstern, 1994]) that the applicability of chronological minimization is intrinsically restricted to domains which do not include non-deterministic information. This is best illustrated with the Tail Pipe Marauder scenario of Example 2. The following formula expresses the fact that if at some time t there is no potato in the tail pipe, then at time $t+1$ this may or may not have changed:

$$\Box\,\textsc{True}\,(t, \overline{pot}) \;\supset\; \Box\,\textsc{True}\,(t+1, \overline{pot}) \,\vee\, \Box\,\textsc{True}\,(t+1, pot) \tag{54}$$

Consider this in conjunction with (53), and suppose given $\Box\,\textsc{True}\,(1, \overline{pot}) \wedge \Box\,\textsc{True}\,(2, start)$. Then from (54) nothing definite can be concluded about $\textsc{True}\,(2, pot)$, which is why chronological ignorance tells us that $\Diamond\,\textsc{True}\,(2, \overline{pot})$ holds, hence $\Box\,\textsc{True}\,(3, runs)$. Thus, chronological ignorance sanctions the conclusion that *start* is qualified at timepoint 2, despite the possibility that the tail pipe marauder has struck by then. The reason for this undesired conclusion is that the *what-you-don't-know-won't-hurt-you* principle is not suited for non-deterministic information. While the qualification problem means to assume away abnormal circumstances whenever they do not provably holds, this is in general too optimistic if the execution of a non-deterministic action renders quite possible such circumstances. Our characterization of the qualification problem accounts for this as the minimization procedure applied to abnormal or miraculous disqualifications does not interfere with the results of non-deterministic actions. It thus seems justified to say that our approach introduces a *smart ostrich* principle, or: *what-you-can't-expect-won't-hurt-you*—clearly, an abnormal disqualification of *start* after carelessly parking the car in the dangerous neighborhood is to be expected from one of the possible effects of waiting, which is why this potential disqualification ought not to be assumed away.

Our characterization of the qualification problem shares with *Motivated Action Theory* [Stein and Morgenstern, 1994; Amsterdam, 1991] the insight that an appropriate notion of causality is necessary when assuming away abnormalities. In the latter framework, occurrences of actions and events are assumed away by default while considering the possibility that they are caused (or, in other words, *motivated*, hence the name). This minimizing unmotivated events and our minimizing non-caused abnormal disqualifications are somehow complementary while based on similar principles. Of course, the formal realizations are quite different. An unsatisfactory property of Motivated Action Theory is that the preference criterion, that is, *motivation*, depends on the syntactical structure of the formulas representing causal knowledge. As a consequence, logical equivalent formalizations may induce different preference criteria, of which only one is the desired. Moreover, the formal concept of motivation becomes rather complicated in case of disjunctive (i.e., non-deterministic) information, which entails difficulties with assessing its range of applicability.

Throughout the paper, we have taken action disqualifications as rendering the execution of the respective action physically impossible. A desirable refinement is to consider actions be disqualified *as to producing a certain effect* (c.f. [Gelfond *et al.*, 1991], e.g.). This is accomplished with a simple, straightforward extension of our theory. In addition to the fluents $disq(a)$, we introduce fluents of the form $disq(a, \ell)$, whose intended reading is "action a fails to produce effect ℓ." These fluents, too, belong to the set $\mathcal{F}_{ab}$ and may be related to other 'abnormality' fluents by means of domain constraints, like in

$$disq(shoot, \overline{alive}) \;\equiv\; bad\text{-}sight \vee bad\text{-}shooter \vee bad\text{-}gun$$

Suppose, then, $\langle C, a, E\rangle$ is the action law to be applied to some state S. The effect which a

actually manages to produce if performed in S is formally given by $E' := E \backslash \{\ell : disq(a,\ell) \in S\}$. Let $C' := C \setminus \{\ell, \overline{\ell} : \ell \in E \setminus E'\}$, which guarantees $|C'| = |E'|$, then $(S \setminus C') \cup E'$ is taken as the intermediate state which is subject to the following ramification process. The notion of a successor state modifies accordingly while all further concepts, viz. interpretations, models, and the preference criterion, remain unaltered.

Finally, it needs to be mentioned that we gave emphasis only to the representational aspect of the qualification problem, as opposed to the computational aspect. That the latter is of equal importance has been pointed out in, e.g., [Elkan, 1995]. Our analysis has revealed some hitherto unnoticed problems with the representational aspect and, to state the obvious, the computational aspect cannot be pursued without an appropriate representation of the problem. Named the computational part of the qualification problem, the challenge is to find a computational model that enables the reasoning agent to assume that an action be qualified without even *thinking* of all possible disqualifying causes—unless some piece of knowledge hints at their presence. In principle, the special fluents $disq(a)$ employed in our theory serve this purpose: By assuming $\overline{disq(a)}$, one jumps to the conclusion that a be qualified provided all strict preconditions are met. Still, on the other hand, in order that this assumption be justified, its consistency as regards the underlying domain constraints must be guaranteed. In a standard reasoning system, this in turn involves consideration (and exclusion) of all the potential disqualifying abnormal circumstances. A solution to the computational part of the qualification problem thus requires a different computational model, presumably based on some parallel architecture, by which all related domain constraints are ignored unless they are explicitly 'activated' by some piece of information. Although this aspect was not among the topics of this paper, the foundations have been laid.

Acknowledgments. The author is grateful to Wolfgang Bibel and Christoph Herrmann for helpful comments and suggestions, and to Erik Sandewall for enlightening remarks on the subject of the paper.

Appendix. Proofs of Theorems 23 and 27

Theorem 23 *Let $\mathcal{FC}_{ramif}$ be the encoding of a domain description with causal model Σ, then for each model ι of $\mathcal{FC}_{ramif} \cup \{(35)\text{–}(40)\}$ there exists a corresponding interpretation (Res, Σ, Υ) and vice versa.*

Proof:
" $\Rightarrow$ ":
Let ι be a model of $\mathcal{FC}_{ramif} \cup \{(35)\text{–}(40)\}$. We define a set of action sequences Υ and a partial mapping Res from finite action sequences to states as follows: $Res(a^*)$ is defined whenever $[Qualified(a^*)]^\iota$ is true; and in case it is defined, let s be a collection of fluent literals such that $[Result(a^*, s)]^\iota$ is true, then $Res(a^*) := S$ where $EUNA \models s = \tau_S$. Furthermore, $a^* \in \Upsilon$ iff $[Miracle(a^*)]^\iota$ is true. By induction on n, we show that, for any action sequence $a^* = [a_1, \ldots, a_n]$, $Res(a^*)$ along with Υ satisfy the conditions of Definition 13, which proves (Res, Σ, Υ) constitute an interpretation that, by construction, corresponds to ι.

In the base case, $n = 0$, following Definition 13 we have to show that $Res([\,])$ is defined and satisfies the domain constraints, and that $[\,] \notin \Upsilon$. According to (35), $[Qualified([\,])]^\iota$ is true; hence, $Res([\,])$ is defined. Given that $[Qualified([\,])]^\iota$ is true, (38) and (39) imply that there is a unique (modulo AC1) term s such that $[Result([\,], s)]^\iota$ is true. In conjunction with

Proposition 17, formula (40) guarantees that s represents a state which, moreover, satisfies the domain constraints following (29) and Proposition 19. Finally, $[\mathit{Miracle}([\,])]^\iota$ is false according to (35); hence, $[\,] \not\in \Upsilon$.

For the induction step let $n > 0$ and suppose the claim holds for the action sequence $[a_1, \ldots, a_{n-1}]$. According to Definition 13, we have to show both that $\mathit{Res}([a^*|a])$ is defined iff clauses 2(a)–2(d) hold and that, in case it is defined, it denotes a successor state of $\mathit{Res}(a^*)$ and a_n . From (36) and the induction hypothesis for a^* we conclude that $[\mathit{Qualified}([a^*|a])]^\iota$ is true iff $\mathit{Res}(a^*)$ is defined, both $[\mathit{Miracle}(a^*)]^\iota$ and $[\mathit{Holds}(\mathit{disq}(a), \tau_{\mathit{Res}(a^*)})]^\iota$ are false, and there exists a term s' such that $[\mathit{Successor}(\tau_{\mathit{Res}(a^*)}, a_n, s')]^\iota$ is true. In turn, these four conditions are equivalent to clause 2(a), clause 2(d) (according to the construction of Υ), clause 2(b) (according to Proposition 19), and clause 2(c) (according to Theorem 22). Moreover, (38) and (39) imply that if $\mathit{Res}([a^*|a_n])$ is defined then there is a unique (modulo AC1) term s such that $[\mathit{Result}([a^*|a_n], s)]^\iota$ is true. From (37), the induction hypothesis for a^* , and Theorem 22, it follows that s represents a successor state of $\mathit{Res}(a^*)$ and a_n .

" $\Leftarrow$ ":
Let $(\mathit{Res}, \Sigma, \Upsilon)$ be an interpretation of the domain description, and let ι be an interpretation of $\mathcal{FC}_{\mathit{ramif}} \cup \{(36)–(40)\}$ which satisfies the following:

1. ι is a model of $\mathcal{FC}_{\mathit{ramif}}$;
2. for any action sequence a^* and collection of fluent literals s ,
   (a) $[\mathit{Qualified}(a^*)]^\iota$ is true iff $\mathit{Res}(a^*)$ is defined;
   (b) $[\mathit{Result}(a^*, s)]^\iota$ is true iff $\mathit{Res}(a^*)$ is defined and $\mathit{EUNA} \models s = \tau_{\mathit{Res}(a^*)}$; and
   (c) $[\mathit{Miracle}(a^*)]^\iota$ is true iff $a^* \in \Upsilon$.

We have to show that ι is a model of $\mathcal{FC}_{\mathit{ramif}} \cup \{(36)–(40)\}$, in which case it corresponds to $(\mathit{Res}, \Sigma, \Upsilon)$ by construction. Given that ι is a model of $\mathcal{FC}_{\mathit{ramif}}$, it suffices to show that it is also a model of (36)–(40). This in turn can be proved by induction on the length of the argument a^* of *Qualified*, *Result*, and *Miracle*. This induction proof is entirely analogous to the above. ■

Theorem 27 *Let $\Delta_{\mathcal{FC}}$ be the prioritized default theory encoding a domain description with observations $\mathcal{O}$, then for each prioritized extension of $\Delta_{\mathcal{FC}}$ there exists a corresponding preferred model of $\mathcal{O}$ and vice versa.*

The proof of this proposition requires some preparation. In what follows, for notational convenience we use the abbreviation $\mathit{Initially}(\overline{\ell}) \equiv \forall s\, [\, \mathit{Result}([\,], s) \supset \neg \mathit{Holds}(\ell, s)\,]$. Let $\Delta_{\mathcal{FC}}$ be the encoding of a domain description with 'abnormality' fluents $\mathcal{F}_{ab}$, and let F be a set of formulas which consists in the following:

1. $W_{\mathcal{FC}}$;
2. either $\neg \mathit{Miracle}(a^*)$ or $\mathit{Miracle}(a^*)$, for each action sequence a^* ; and
3. either $\mathit{Initially}(\overline{f_{ab}})$ or $\neg \mathit{Initially}(\overline{f_{ab}})$, for each ground instance f_{ab} of an element in $\mathcal{F}_{ab}$.

Then we call $E = \mathit{Th}(F)$ a *potential extension* of $\Delta_{\mathcal{FC}}$. Notice that potential extensions may be inconsistent, e.g., if $W_{\mathcal{FC}}$ entails a miraculous disqualification of, say, $[\mathit{put}(\mathit{po})]$ but $\neg \mathit{Miracle}([\mathit{put}(\mathit{po})]) \in F$.

Given a potential extension $E = \mathit{Th}(F)$, we call *induced by* E any total ordering $\ll$ that extends $<_{\mathcal{FC}}$ such that

1. $(50)_{a^*} \ll (50)_{b^*}$ whenever $\neg \mathit{Miracle}(a^*) \in F$ and $\mathit{Miracle}(b^*) \in F$; and
2. $(49)_{f_{ab}} \ll (49)_{f'_{ab}}$ whenever $\mathit{Initially}(\overline{f_{ab}}) \in F$ and $\neg \mathit{Initially}(\overline{f'_{ab}}) \in F$.

Induced orderings will be used below to verify the conditions of Definition 26 for potential extensions which are claimed to constitute prioritized extensions. It is easy to verify that the standard extensions of a theory $(D_{\mathcal{FC}}, W_{\mathcal{FC}})$ are always potential extensions.

Lemma 28 *Let $\Delta_{\mathcal{FC}} = (D_{\mathcal{FC}}, W_{\mathcal{FC}}, <_{\mathcal{FC}})$ be the encoding of a domain description, then each (standard) extension of $(D_{\mathcal{FC}}, W_{\mathcal{FC}})$ is a potential extension.*

Proof: Let E be an extension of $(D_{\mathcal{FC}}, W_{\mathcal{FC}})$, and let

1. $\Gamma_0 = W_{\mathcal{FC}}$;
2. $\Gamma_1 = \mathit{Th}(\Gamma_0) \cup \{\omega : \frac{:\omega}{\omega} \in D_{\mathcal{FC}} \text{ and } \neg\omega \notin E\}$; and
3. $\Gamma_2 = \mathit{Th}(\Gamma_1)$.

Since all possibly applicable defaults in $D_{\mathcal{FC}}$ have been applied to compute Γ_1 and since E is extension, we know $\Gamma_2 = E$ according to Definition 25. By construction, Γ_2, hence E, is subset of some potential extension. Thus it remains to show the following:

1. Let a^* be any action sequence. From $\frac{:\neg \mathit{Miracle}(a^*)}{\neg \mathit{Miracle}(a^*)} \in D_{\mathcal{FC}}$ and the construction of Γ_1, we know that either $\neg \mathit{Miracle}(a^*) \in \Gamma_1 \subseteq E$ or else $\mathit{Miracle}(a^*) \in E$.
2. Let f_{ab} be any ground instance of a member of $\mathcal{F}_{ab}$. From $\frac{:\mathit{Initially}(\overline{f_{ab}})}{\mathit{Initially}(\overline{f_{ab}})} \in D_{\mathcal{FC}}$ and the construction of Γ_1, we know that either $\mathit{Initially}(\overline{f_{ab}}) \in \Gamma_1 \subseteq E$ or else $\neg \mathit{Initially}(\overline{f_{ab}}) \in E$.

■

Let $\Delta_{\mathcal{FC}}$ be the encoding of a domain description. The notion of correspondence introduced in Section 4.3 is extended to potential extension in the obvious way, that is, a potential extension $E = \mathit{Th}(F)$ and an interpretation $(\mathit{Res}, \Sigma, \Upsilon)$ correspond iff the conditions in (51) hold for all ground instances $f_{ab}[\widetilde{e}]$ wrt. $\mathcal{F}_{ab}$ and all action sequences a^*. Notice that each interpretation has a unique corresponding potential extension, whereas there might be multiple interpretations corresponding to a single potential extension. Notice further that whenever E is consistent then there exists a corresponding interpretation which is model of the underlying observations $\mathcal{O}$. This is granted by Theorem 24, for if E is consistent then it admits a (classical) model ι.

We are now prepared to prove Theorem 27.

Proof:

"$\Leftarrow$":

Let $M = (\mathit{Res}, \Sigma, \Upsilon)$ be a preferred model of $\mathcal{O}$, and let $E = \mathit{Th}(F)$ be the potential extension corresponding to M. First, we prove that E is a standard extension of $(D_{\mathcal{FC}}, W_{\mathcal{FC}})$. Let

1. $\Gamma_0 = W_{\mathcal{FC}}$;
2. $\Gamma_1 = \mathit{Th}(\Gamma_0) \cup \{\omega : \frac{:\omega}{\omega} \in D_{\mathcal{FC}} \text{ and } \neg\omega \notin E\}$; and

3. $\Gamma_2 = Th(\Gamma_1)$.

Then we have to verify that $\Gamma_2 = E$ (c.f. the proof of Lemma 28). Clearly, $\Gamma_2 \subseteq E$, since for any $\frac{:\omega}{\omega} \in D_{\mathcal{FC}}$ such that $\omega \in \Gamma_1$, we have $\neg\omega \notin E$, which in turn implies $\omega \in E$ as E is a potential extension. Moreover, the assumption $\Gamma_2 \subsetneqq E$ leads to a contradiction: Given $\Gamma_2 \subsetneqq E$, this indicates the existence of some $\frac{:\omega}{\omega} \in D_{\mathcal{FC}}$ (where $\omega = \neg \mathit{Miracle}(a^*)$ or $\omega = \mathit{Initially}(\overline{f_{ab}})$ for some action sequence a^* or some ground instance f_{ab} wrt. $\mathcal{F}_{ab}$) such that $\neg\omega \in E$ but $\neg\omega \notin \Gamma_2$. Let Ω be the set of all these ω, i.e., $\Omega := \{\neg\omega \in E : \neg\omega \notin \Gamma_2\}$. Then $E' := (E \setminus \Omega) \cup \{\omega : \neg\omega \in \Omega\}$ is an extension of $(D_{\mathcal{FC}}, W_{\mathcal{FC}})$. From Lemma 28, we know that E' is potential extension. Let M' be an interpretation corresponding to E' such that M' is a model of $\mathcal{O}$. From the construction of E' and from (51), it follows that M' contains strictly less abnormality assumptions than M, given that Ω is non-empty. Thus, $M' \prec M$, which contradicts M being preferred model.

It remains to be shown that E is prioritized according to Definition 26. Let $\ll$ be any total ordering induced by E. Furthermore, let E' be any extension of $(D_{\mathcal{FC}}, W_{\mathcal{FC}})$, and let M' be an interpretation corresponding to E' such that M' is a model of $\mathcal{O}$. Suppose $\frac{:\omega'}{\omega'} \in D_{\mathcal{FC}}$ is a default which is applied in $E' \setminus E$. We consider two cases:

1. If $\omega' = \neg \mathit{Miracle}(a^*)$ for some action sequence a^*, then $a^* \in \Upsilon'$ but $a^* \notin \Upsilon$. M being preferred model, we know that $M' \not\prec M$. Thus there also exists some $b^* \in \Upsilon$ such that $b^* \notin \Upsilon'$. Hence, $\neg \mathit{Miracle}(b^*) \in E$, $\neg \mathit{Miracle}(b^*) \notin E'$, and $\frac{:\neg \mathit{Miracle}(b^*)}{\neg \mathit{Miracle}(b^*)} \ll \frac{:\neg \mathit{Miracle}(a^*)}{\neg \mathit{Miracle}(a^*)}$ (since $\ll$ is induced by E), that is, there exists a default which is preferred (wrt. $\ll$) to $\frac{:\omega'}{\omega'}$ and which has been applied in $E \setminus E'$.

2. If $\omega' = \mathit{Initially}(\overline{f_{ab}})$ for some ground instance f_{ab} of some fluent name in $\mathcal{F}_{ab}$, then $\overline{f_{ab}} \in \mathit{Res}'([\,])$ but $\overline{f_{ab}} \notin \mathit{Res}([\,])$. Again, M being preferred model, we know that $M' \not\prec M$. Thus either there exists some action sequence a^* such that $a^* \in \Upsilon$ and $a^* \notin \Upsilon'$, or there exists some $f'_{ab} \in \mathcal{F}_{ab}$ such that $\overline{f'_{ab}} \in \mathit{Res}([\,])$ and $\overline{f'_{ab}} \notin \mathit{Res}'([\,])$. As above, this implies the existence of some default $\frac{:\omega}{\omega} \ll \frac{:\omega'}{\omega'}$ which has been applied in $E \setminus E'$.

"$\Rightarrow$": Let E be prioritized extension of $\Delta_{\mathcal{FC}}$. From Lemma 28 we know that E is potential extension. Let M be an interpretation corresponding to E such that M is a model of $\mathcal{O}$. We prove by contradiction that M is preferred. Suppose there exists a preferred model M' of $\mathcal{O}$ such that $M' \prec M$. This implies the existence of a corresponding prioritized extension E' of $\Delta_{\mathcal{FC}}$ according to the first half ("$\Leftarrow$") of this proof. Given $M' \prec M$, we distinguish two cases:

1. Suppose $\Upsilon' \subsetneqq \Upsilon$. This implies the existence of some $\frac{:\neg \mathit{Miracle}(a^*)}{\neg \mathit{Miracle}(a^*)} \in D_{\mathcal{FC}}$ which is applied in $E' \setminus E$. It also implies there is no $\frac{:\neg \mathit{Miracle}(b^*)}{\neg \mathit{Miracle}(b^*)} \in D_{\mathcal{FC}}$ which is applied in $E \setminus E'$. Since each total ordering in the sense of Definition 26 must respect $<_{\mathcal{FC}}$, this contradicts E being prioritized extension.

2. Suppose $\Upsilon' = \Upsilon$ and $\mathit{Res}'([\,]) \cap \mathcal{F}_{ab} \subsetneqq \mathit{Res}([\,]) \cap \mathcal{F}_{ab}$. This implies the existence of some $\frac{:\mathit{Initially}(\overline{f'_{ab}})}{\mathit{Initially}(\overline{f'_{ab}})} \in D_{\mathcal{FC}}$ which is applied in $E' \setminus E$. It also implies there is no $\frac{:\mathit{Initially}(\overline{f_{ab}})}{\mathit{Initially}(\overline{f_{ab}})} \in D_{\mathcal{FC}}$ which is applied in $E \setminus E'$. Moreover, there cannot be some $\frac{:\neg \mathit{Miracle}(a^*)}{\neg \mathit{Miracle}(a^*)} \in D_{\mathcal{FC}}$ which has been applied in $E \setminus E'$ due to $\Upsilon' = \Upsilon$. Altogether, this contradicts E being prioritized extension.

References

[Amsterdam, 1991] J. B. Amsterdam. Temporal reasoning and narrative conventions. In J. F. Allen, R. Fikes, and E. Sandewall, editors, *Proceedings of the International Conference on Principles of Knowledge Representation and Reasoning (KR)*, pages 15–21, Cambridge, MA, 1991.

[Bibel, 1986] Wolfgang Bibel. A deductive solution for plan generation. *New Generation Computing*, 4:115–132, 1986.

[Bornscheuer and Thielscher, 1997] Sven-Erik Bornscheuer and Michael Thielscher. Explicit and implicit indeterminism: Reasoning about uncertain and contradictory specifications of dynamic systems. *Journal of Logic Programming*, 1997. (To appear).

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

[Brewka, 1994] Gerhard Brewka. Adding priorities and specificity to default logic. In C. MacNish, D. Pearce, and L. M. Pereira, editors, *Proceedings of the European Workshop on Logics in AI (JELIA)*, volume 838 of *LNAI*, pages 50–65. Springer, September 1994.

[Büttner, 1986] Wolfram Büttner. Unification in datastructure multisets. *Journal of Automated Reasoning*, 2:75–88, 1986.

[Elkan, 1992] Charles 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.

[Elkan, 1995] Charles Elkan. On solving the qualification problem. In C. Boutilier and M. Goldszmidt, editors, *Extending Theories of Actions: Formal Theory and Practical Applications*, volume SS–95–07 of *AAAI Spring Symposia*, Stanford University, March 1995. AAAI Press.

[Fikes and Nilsson, 1971] Richard E. Fikes and Nils J. Nilsson. STRIPS: A new approach to the application of theorem proving to problem solving. *Artificial Intelligence Journal*, 2:189–208, 1971.

[Geffner, 1992] Hector Geffner. *Default Reasoning: Causal and Conditional Theories.* MIT Press, 1992.

[Gelfond and Lifschitz, 1993] Michael Gelfond and Vladimir Lifschitz. Representing action and change by logic programs. *Journal of Logic Programming*, 17:301–321, 1993.

[Gelfond *et al.*, 1991] Michael Gelfond, Vladimir Lifschitz, and Arkady Rabinov. What are the limitations of the situation calculus? In S. Boyer, editor, *Automated Reasoning, Essays in Honor of Woody Bledsoe*, pages 167–181. Kluwer Academic, 1991.

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

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

[Girard, 1987] Jean-Yves Girard. Linear Logic. *Journal of Theoretical Computer Science*, 50(1):1–102, 1987.

[Green, 1969] Cordell Green. Application of theorem proving to problem solving. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*, pages 219–239, Los Altos, CA, 1969. Morgan Kaufmann.

[Hanks and McDermott, 1987] Steve Hanks and Drew McDermott. Nonmonotonic logic and temporal projection. *Artificial Intelligence Journal*, 33(3):379–412, 1987.

[Herrmann and Thielscher, 1996] Christoph S. Herrmann and Michael Thielscher. Reasoning about continuous processes. In B. Clancey and D. Weld, editors, *Proceedings of the AAAI National Conference on Artificial Intelligence*, pages 639–644, Portland, OR, August 1996. MIT Press.

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

[Hölldobler and Thielscher, 1995] Steffen Hölldobler and Michael Thielscher. Computing change and specificity with equational logic programs. *Annals of Mathematics and Artificial Intelligence*, 14(1):99–133, 1995.

[Jaffar *et al.*, 1984] Joxan Jaffar, Jean-Louis Lassez, and Michael J. Maher. A theory of complete logic programs with equality. *Journal of Logic Programming*, 1(3):211–223, 1984.

[Kartha and Lifschitz, 1994] G. Neelakantan Kartha and Vladimir Lifschitz. Actions with indirect effects. In J. Doyle, E. Sandewall, and P. Torasso, editors, *Proceedings of the International Conference on Principles of Knowledge Representation and Reasoning (KR)*, pages 341–350, Bonn, Germany, May 1994. Morgan Kaufmann.

[Kautz, 1986] Henry Kautz. The logic of persistence. In *Proceedings of the AAAI National Conference on Artificial Intelligence*, pages 401–405, Philadelphia, PA, August 1986.

[Lifschitz, 1986] Vladimir Lifschitz. On the semantics of STRIPS. In M. P. Georgeff and A. L. Lansky, editors, *Proceedings of the Workshop on Reasoning about Actions & Plans*. Morgan Kaufmann, 1986.

[Lifschitz, 1987] Vladimir Lifschitz. Formal theories of action (preliminary report). In J. McDermott, editor, *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*, pages 966–972, Milan, Italy, August 1987. Morgan Kaufmann.

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

[Lifschitz, 1993] Vladimir Lifschitz. Restricted monotonicity. In *Proceedings of the AAAI National Conference on Artificial Intelligence*, pages 432–437, Washington, DC, July 1993.

[Lin and Reiter, 1994] Fangzhen Lin and Ray Reiter. State constraints revisited. *Journal of Logic and Computation*, 4(5):655–678, 1994.

[Lin, 1995] Fangzhen Lin. Embracing causality in specifying the indirect effects of actions. In C. S. Mellish, editor, *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*, pages 1985–1991, Montreal, Canada, August 1995. Morgan Kaufmann.

[Masseron *et al.*, 1993] M. Masseron, Christophe Tollu, and Jacqueline Vauzielles. Generating plans in linear logic I. Actions as proofs. *Journal of Theoretical Computer Science*, 113:349–370, 1993.

[McCain and Turner, 1995] Norman McCain and Hudson Turner. A causal theory of ramifications and qalifications. In C. S. Mellish, editor, *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*, pages 1978–1984, Montreal, Canada, August 1995. Morgan Kaufmann.

[McCarthy and Hayes, 1969] John McCarthy and Patrick J. Hayes. Some philosophical problems from the standpoint of artificial intelligence. *Machine Intelligence*, 4:463–502, 1969.

[McCarthy, 1959] John McCarthy. Programs with Common Sense. In *Proceedings of the Teddington Conference on the Mechanization of Thought Processes*, London, 1959. (Reprinted in: J. McCarthy, *Formalizing Common Sense*, Ablex, Norwood, New Jersey, 1990).

[McCarthy, 1977] John McCarthy. Epistemological problems of artificial intelligence. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*, pages 1038–1044, Cambridge, MA, 1977. MIT Press.

[McCarthy, 1980] John McCarthy. Circumscription—a form of non-monotonic reasoning. *Artificial Intelligence Journal*, 13:27–39, 1980.

[McCarthy, 1986] John McCarthy. Applications of circumscription to formalizing common-sense knowledge. *Artificial Intelligence Journal*, 28:89–116, 1986.

[Quine, 1960] Willard V. Quine. *Word and Object*. MIT Press, 1960.

[Reiter, 1980] Ray Reiter. A logic for default reasoning. *Artificial Intelligence Journal*, 13:81–132, 1980.

[Reiter, 1991] Ray 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.

[Rintanen, 1995] Jussi Rintanen. On specificity in default logic. In C. S. Mellish, editor, *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*, pages 1474–1479, Montreal, Canada, August 1995. Morgan Kaufmann.

[Sandewall, 1993] Erik Sandewall. Systematic assessment of temporal reasoning methods for use in autonomous systems. In B. Fronhöfer, editor, *Workshop on Reasoning about Action & Change at IJCAI*, pages 21–36, Chambéry, August 1993.

[Sandewall, 1994] Erik Sandewall. *Features and Fluents. The Representation of Knowledge about Dynamical Systems.* Oxford University Press, 1994.

[Shepherdson, 1992] John C. Shepherdson. SLDNF-resolution with equality. *Journal of Automated Reasoning*, 8:297–306, 1992.

[Shoham, 1987] Yoav Shoham. *Reasoning about Change.* MIT Press, 1987.

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

[Stein and Morgenstern, 1994] Lynn Andrea Stein and Leora Morgenstern. Motivated action theory: a formal theory of causal reaosning. *Artificial Intelligence Journal*, 71:1–42, 1994.

[Stickel, 1981] Mark E. Stickel. A unification algorithm for associative commutative functions. *Journal of the ACM*, 28(3):207–274, 1981.

[Thielscher, 1995] Michael Thielscher. The logic of dynamic systems. In C. S. Mellish, editor, *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*, pages 1956–1962, Montreal, Canada, August 1995. Morgan Kaufmann.

[Thielscher, 1996] Michael Thielscher. On the completeness of SLDENF-resolution. *Journal of Automated Reasoning*, 1996. (To appear).

[Thielscher, 1997] Michael Thielscher. Ramification and causality. *Artificial Intelligence Journal*, 1997. (To appear. A preliminary version is available as Technical Report TR-96-003, ICSI, Berkeley, CA).