Systematic assessment of temporal reasoning methods for use in autonomous agents

Erik Sandewall
Department of Computer and Information Science
Linköping University
Linköping, Sweden
E-mail `ejs@ida.liu.se`

December 20, 1998

Abstract

Non-monotonic logics for reasoning about time and action have traditionally been verified through their informal plausibility and their application to a small number of test examples. I present a systematic method using an *underlying semantics*, where a preferential non-monotonic logic is correct if the maximally preferred models equals the set of histories that can arise in a "scenario simulation" of the underlying semantics. Several old and new logics have been analyzed using the systematic method.

The systematic method has first been applied to the relatively straightforward case of scenarios with simple inertia, where non-deterministic actions and actions with duration different from one timestep are the only complications. In this extended abstract I first make a brief review of the approach to simple inertia, and then describe how the systematic approach can be extended to obtain a precise account of broader types of frame problems, such as problems involving surprises and ramifications.

1 The systematic approach

There has been much research in recent years on methods for reasoning about actions and change, and on finding solutions for the so-called "frame problems". The non-monotonic logics which have been proposed for this purpose have traditionally been verified through their informal plausibility and their application to a small number of test examples. The limitations of this approach have become more apparent in recent years, and at the same time some results of more systematic nature have been presented in particular by Lin and Shoham[?], Lifschitz[?], and Reiter[?], all of which have used a situation-calculus framework. In this extended abstract I shall describe another approach which is based on an explicit treatment of time in a linear time domain, and which is able to accomodate a broader range of reasoning problems. The complete results are available in a review version of a forthcoming book[?].

By an *assessment* of a nonmonotonic logic, I mean a statement that for a certain class of reasoning problems, the logic is guaranteed to always give the correct set of conclusions: the set of intended conclusions is equal to the set of actual conclusions obtained through the logic. A *systematic approach* to non-monotonic logic is one where it possible to obtain such assessments with formal proofs, based on precise definitions. A systematic approach requires in particular that one makes precise definitions of the class of reasoning problems in question, and of the set of intended conclusions.

I have developed such a systematic approach in model theoretic terms, where therefore the correctness criterion is that the set of intended models shall equal the set of actually obtained models in the logic. A central part of the approach is to use an *ontological taxonomy*, which provides the instrument for characterizing classes of systems or "worlds" which the logic is used for reasoning about. For a given logic, the assessment will specify a class within that taxonomy for which the logic always obtains the correct results. This makes it easy to compare the range of applicability of different logics.

The nonmonotonic logics that are to be assessed and compared in this fashion, can often be characterized in terms of a *preference relation* or a *selection function* on models. A selection function maps any set S of interpretations to a subset of S, and is used to map from the set of classical models for the given premises to a smaller set of selected models. In one important special case, the selection function obtains those members of S which are minimal according to a certain preference relation, so the maximally preferred models are the selected ones. More complex selection functions are needed to account for broader classes of reasoning problems. In particular one may need to divide the premises into several *partitions*, and to apply different selection functions to the classical model sets for the different partitions.

The main steps in the development of the approach are as follows.

1. Semiformal definition of an *overview taxonomy*, using a set of *specialities* i.e. features of reasoning problems. The following are some of the ontological specialities, i.e. those which characterize the world that is being reasoned about:

   C: Concurrent actions are allowed.

   L: Delayed effects are allowed, i.e. effects of an action that take place after the action has been completed.

   T: Timebound objects are allowed, i.e. objects that are created and terminated at some points in time.

   The overview taxonomy also specifies the epistemological assumptions, i.e. what assumptions are made about the completeness of the information in the given premises. Here I shall only consider the case that all actions are known, and that all potential effects of actions are also known although nondeterministic actions are allowed. This case will be represented as $\mathcal{K}$.

   The simplest category in the overview taxonomy is $\mathcal{K}$-**IA** representing the $\mathcal{K}$ assumption, integer time, sequential actions, an inertia assumption for all fluents i.e. the value of a fluent does not change except directly due to an action, but non-deterministic actions and actions with a duration in time (not just a single time-step) are allowed. Broader categories are formed e.g. as $\mathcal{K}$-**IAC** if the requirement for sequential actions is dropped, or $\mathcal{K}$-**IAL** if delayed effects are allowed.

2. For each category of reasoning problems in the overview taxonomy, one defines a corresponding *underlying semantics* in a formal and precise fashion. I use an underlying semantics which captures the basic A.I. intuitions, similar to the "agent model" of Genesereth and Nilsson [**?**]. The primary purpose of the underlying semantics is to define what is the set of intended models for a given set of premises. Indirectly, the underlying semantics also serves as a precise definition of the problem family at hand. Also it is the basis for the subsequent steps of analysis.

3. Next, a more fine-grained taxonomy is defined within the overview category at hand, by introducing subordinate specialities. For example, within the $\mathcal{K}$-**IA** category we define the more restricted classes $\mathcal{K}$-**IsA** where all actions must be performed over a single time-step, $\mathcal{K}$-**IAd** where all actions must be deterministic (the state of the world when the action starts determines completely the state of the world when the action ends), and a number of other classes. These sub-specialities can also be freely combined, e.g. as $\mathcal{K}$-**IsAd**.

4. Naturally there also needs to be a certain number of syntactic definitions in order to specify how formulae are to be written and organized. Besides

the usual kind of syntax definitions that are needed in any logic, some of the non-monotonic logics require that one separates the given premises into several distinct sets called *partitions* – for example one set of "action statements" and another set of "observations" – and there may by special constraints on the syntax of formulae in each of those premise partitions. The syntax in this extended sense must be related to the underlying semantics as well as to the conventional (Tarskian) semantics for the logic.

5. At this point all the conceptual tools are in order, and the assessment can be performed. The result of the assessment must be a statement that for a given class of reasoning problems which is defined in the fine-grained taxonomy and for a given non-monotonic logic, the set of intended models as defined by the underlying semantics equals the set of selected models as obtained by the logic. For example if the selection function in the non-monotonic logic is based on a preference relation $\ll$, then it is the set of $\ll$-minimal models that must equal the set of intended models. Naturally a formal proof of the assessment statement is also required.

This methodology de-dramatizes the issue of correctness for an applied nonmonotonic logic. If the range of applicability for logic A is a subset of the range for the logic B, then there may still be good reasons for using logic A, namely if one's application satisfies the restrictions that A imposes and if A can be implemented more efficiently than B. This is particularly important since logics (or more precisely, non-monotonic entailment criteria) that suffice for a broad range of common-sense reasoning problems tend to become quite complicated.

This systematic methodology was first applied to the case of simple inertia, where non-deterministic actions and actions with duration different from one timestep are the only complications. The results for simple inertia have been presented in the book manuscript[**?**] and in a separate paper [**?**]. In this extended abstract I will describe how the results for simple inertia can be extended to obtain a precise account of broader types of frame problems, such as problems involving surprises and ramifications. However I must begin with a brief review of the approach to simple inertia since the new results build on it.

In order to keep this paper short and to focus on the general approach and the main results, I omit a number of details and fine points in the formal definitions. Please refer to the book manuscript if a clarification is needed.

2 Simple inertia

It is natural to first address the simple category of $\mathcal{K}$-**IA**, and to analyze it in detail. This family of reasoning problems is broader than the ones which have been considered by Shoham, Lifschitz, and Reiter in particular by allowing the combination of actions with extended duration in time and non-deterministic actions such as e.g. flipping a coin. For non-deterministic actions the assumption of complete knowledge is that there are several possible outcomes of the action, even for a given state of the world when the actions starts, but the *set of possible outcomes* is known. More generally, to also account for actions with extended duration, the *set of possible trajectories* is assumed as known. Each trajectory specifies the duration of the action, the set of fluents which are changed in the course of the action, and the value of each of those fluents in each time-step within the trajectory. In other words each trajectory is a finite sequence of partial states of the world. Corresponding to the frame assumption of inertia, all fluents which are not included in the trajectory are assumed to remain constant for the duration of the action.

The underlying semantics for $\mathcal{K}$-**IA** captures these concepts, and is defined as follows. If r is a state of the world and A is an action, then $\mathtt{Infl}(A, r)$ is a set of features (names of fluents), and $\mathtt{Trajs}(A, r)$ is a set of trajectories each of which is a finite sequence of partial states where exactly the features in $\mathtt{Infl}(A, r)$ are assigned values. This means that the set of influenced fluents must be kept constant throughout a trajectory, and must be the same for all trajectories for a given A and r.

A reasoning problem is assumed to be stated in terms of (1) a set of *action laws* characterizing the effects of actions, (2) a set of *action statements* specifying which actions are performed and restrictions on their timing, and (3) a set of *observations* specifying the values of fluents at one or more points in time. For a given set of action laws, one can construct the corresponding functions `Infl` and `Trajs` with the structure which has just been defined, and consider them as the operative definition of the "world". The full definitions of the underlying semantics are such that the actions laws will uniquely define the world $\langle \mathtt{Infl}, \mathtt{Trajs} \rangle$.

The set of intended models for a given set of premises is then defined as follows. Consider a non-deterministic game between an "ego" and the world, where the two players take turns to perform moves, and a history in the world is constructed as the game proceeds. The ego is essentially Newell's "knowledge level", and the "world" is understood as the combination of the agent's physical environment and its own physical body or vehicle.

At the start of the game time has the value zero, and the world is in an arbitrary state. At each step in the game, one has constructed a history over a finite interval of time $[0, n]$ where n represents "now". Each move by the ego consists of issuing an internal command to perform an action A at time n. The only effect of that invocation in the history is to add the information that action A started at time n and is still in process.

When the world has the move, it implements the action that the ego initiated, as follows. Let r be the state of the world at time n i.e. the present ending state of the history, and suppose the ego has invoked action A starting at time n. The world selects an arbitrary member of $\mathtt{Trajs}(A, r)$. Suppose that member has length k. The world will then extend the present history to length $[0, n + k]$ by adding k new states which are obtained from the chosen trajectory by allowing all other features (except those included in $\mathtt{Infl}(A, r)$) to remain constant. Then it is the ego's turn to move again. When the ego has run out of moves, the constructed finite history is extrapolated to infinity without any changes of state.

For any given reasoning problem, one uses its action laws to construct $\langle \mathtt{Infl}, \mathtt{Trajs} \rangle$, and then constructs the set of all histories (games) for all possible initial states, all possible ego behaviors and all possible choices by the world when it selects a member of $\mathtt{Trajs}(A, r)$. Among those histories, one selects those which (1) satisfy all the action statements, (2) satisfy all the observations, and (3) do not contain any actions besides those included among the action statements. The resulting set of histories is by definition the intended ones, and sets the standard for what models the non-monotonic logic is supposed to prefer or select.

With these definitions it has been possible to assess the range of applicability of a number of previously proposed entailment methods, such as different variants of chronological minimization, uses of filtering and occlusion, causal minimization, explanation closure, etc. These assessments have been presented in [**?**, **?**].

Notice that under this definition of intended models, it is quite possible to have inconsistent reasoning problems, where the set of intended models is empty. For example in the stolen car scenario, the assumptions are that if the car is left overnight in the garage then it is still there in the morning, and that the car is left in the garage over two successive nights. Anyway after two nights it is gone. This is an inconsistent reasoning problem in $\mathcal{K}$-**IA** i.e. under the underlying semantics that has just been defined. Any non-monotonic logic that obtains a non-empty set of preferred models for this example is therefore incorrect for $\mathcal{K}$-**IA**. If one intends that the set of models in this example shall be non-empty, then one needs to use another and broader category in the taxonomy of reasoning problems. In particular the speciality **S** for "surprises" allows for additional changes of fluent values, besides those specified by $\langle \mathtt{Infl}, \mathtt{Trajs} \rangle$, which usually do not occur but which may need to be assumed in order to avoid obtaining an empty model set.

The formal analysis which leads up to assessment results for $\mathcal{K}$-**IA** is based on the use of a *full trajectory normal form*, FTNF, for action laws. Consider for example a shooting world with three propositional fluents, namely a ("the turkey is alive"), l ("the gun is loaded"), and r ("it is raining"). Also there is one action *Fire* which has the effect of unloading the gun, if it was loaded, and of killing the turkey iff the gun was loaded. The rain does not affect these outcomes, and the rain itself is not affected. Furthermore for the sake of the example, assume that in the case where the turkey is alive and the gun is loaded the action takes two time units, where the gun becomes unloaded during the first time-step, and the turkey dies in the second time-step. In all other cases it takes one time-step.

Under these assumptions the set $\mathtt{Trajs}(\mathit{Fire}, r)$ will have exactly one member for each choice of r. Table **??** specifies, for each choice of r, the value of $\mathtt{Infl}(\mathit{Fire}, r)$ and the single member of $\mathtt{Trajs}(\mathit{Fire}, r)$. The FTNF of the action

r	$\mathtt{Infl}(Fire, r)$	$\mathtt{Trajs}(Fire, r)$
$\{a:\mathtt{T}, l:\mathtt{T}, r:\mathtt{T}\}$	$\{a, l\}$	$\langle\{a:\mathtt{T}, l:\mathtt{F}\}, \{a:\mathtt{F}, l:\mathtt{F}\}\rangle$
$\{a:\mathtt{T}, l:\mathtt{T}, r:\mathtt{F}\}$	$\{a, l\}$	$\langle\{a:\mathtt{T}, l:\mathtt{F}\}, \{a:\mathtt{F}, l:\mathtt{F}\}\rangle$
$\{a:\mathtt{T}, l:\mathtt{F}, r:\mathtt{T}\}$	?	$\langle ? \rangle$
$\{a:\mathtt{T}, l:\mathtt{F}, r:\mathtt{F}\}$	?	$\langle ? \rangle$
$\{a:\mathtt{F}, l:\mathtt{T}, r:\mathtt{T}\}$	$\{l\}$	$\langle\{l:\mathtt{F}\}\rangle$
$\{a:\mathtt{F}, l:\mathtt{T}, r:\mathtt{F}\}$	$\{l\}$	$\langle\{l:\mathtt{F}\}\rangle$
$\{a:\mathtt{F}, l:\mathtt{F}, r:\mathtt{T}\}$	?	$\langle ? \rangle$
$\{a:\mathtt{F}, l:\mathtt{F}, r:\mathtt{F}\}$	?	$\langle ? \rangle$

Table 1: `The world description` $\langle\mathtt{Infl}, \mathtt{Trajs}\rangle$

law for *Fire* is

$$
\begin{aligned}
&[s,t]\,Fire \Rrightarrow \\
&\quad(([s]\,a \wedge l \wedge r) \Rightarrow ([s,t]\,a := \mathtt{F} \wedge l := \mathtt{F}) \wedge t = s+2 \wedge [s+1]\,a \wedge \neg l) \wedge \\
&\quad(([s]\,a \wedge l \wedge \neg r) \Rightarrow ([s,t]\,a := \mathtt{F} \wedge l := \mathtt{F}) \wedge t = s+2 \wedge [s+1]\,a \wedge \neg l) \wedge \\
&\quad(([s]\,a \wedge \neg l \wedge r) \Rightarrow t = s+1) \wedge \\
&\quad(([s]\,a \wedge \neg l \wedge \neg r) \Rightarrow t = s+1) \wedge \\
&\quad(([s]\,\neg a \wedge l \wedge r) \Rightarrow ([s,t]\,l := \mathtt{F}) \wedge t = s+1) \wedge \\
&\quad(([s]\,\neg a \wedge l \wedge \neg r) \Rightarrow ([s,t]\,l := \mathtt{F}) \wedge t = s+1) \wedge \\
&\quad(([s]\,\neg a \wedge \neg l \wedge r) \Rightarrow t = s+1) \wedge \\
&\quad(([s]\,\neg a \wedge \neg l \wedge \neg r) \Rightarrow t = s+1)
\end{aligned}
$$

The construct $[s,t]\,f := x$ designates that the fluent f changes its value some time(s) during the time interval $[s,t]$ and finally obtains the value x. The operator $\Rrightarrow$ may be read as implication for the present purpose.

It is seen that this formula is a direct translation of the information in the table into logical formulae. The similarity is an advantage in the formal proofs, but for practical purposes one would of course use a simpler formula which is logically equivalent to the full trajectory normal form, such as

$$
\begin{aligned}
&[s,t]\,Fire \Rrightarrow \\
&\quad(([s]\,a \wedge l) \Rightarrow ([s,t]\,a := \mathtt{F}) \wedge t = s+2 \wedge [s+1]\,a \wedge \neg l) \wedge \\
&\quad(([s]\,\neg a \vee \neg l) \Rightarrow t = s+1) \wedge \\
&\quad[s,t]\,l := \mathtt{F}
\end{aligned}
$$

Since most of the entailment methods are defined in terms of model sets, the transformation to a classically equivalent formula does not influence their sets of selected or preferred models. However the formal proofs of assessments for various entailment methods rely on the direct correspondence between the FTNF syntax and the underlying semantics.

3 The frame problems

3.1 Towards common-sense formulations of action laws

The primary formulation of the action laws which was used in the analysis for $\mathcal{K}$-**IA** – the full trajectory normal form – reflects the structure of the underlying semantics. Another possible choice of primary formulation would be to use the "most natural" expression of common-sense facts in logic. As the example showed, there is a fairly big difference between these two possible choices of primary formulations of action laws due to the precise structure of the underlying semantics.

Let me refer to these choices as the *semantics-motivated* and the *common-sense-motivated* forms for expressing action laws. From an A.I. point of view it is the common-sense-motivated form(s) that one really wants to capture. However that form is notoriously difficult to pinpoint, and the semantics-motivated form has the advantage of an exact definition which makes a precise analysis possible. My approach is therefore to start with the semantics-motivated form of the action laws, and to introduce and analyze various transformations and extensions of the semantics-motivated form which may bring it closer to the common-sense-motivated forms.

This approach differs from the standard approach in the A.I. study of common-sense reasoning, which has been to start from the common-sense-motivated form of the action law, and to inquire how the inference machinery of the logic would have to be chosen in order to obtain the intended conclusions from those formulae. The problem with the standard approach is that it does not provide a reliable basis for the continued formal analysis, and useful general results have therefore not been obtained. Therefore I start instead from the underlying trajectory semantics, and ask how the logic (syntax and entailment conditions) *can be modified* in the direction of more natural or common-sense-like formulations of the action laws and of the other components of the chronicle.

A small step in this direction was already taken above with the observation that one can replace each action law in FTNF by a simplified formula which is more compact but logically equivalent (in the sense of classical logic, i.e. having the same model set). However such local rewriting will only obtain a moderate improvement of the naturalness in the action laws.

A major desideratum for more natural action laws is that one should be able to "factor out" some information from the laws i.e. to make separate statements of some of the relevant information. The action law can then be restricted to expressing the "typical" or "primary" effects of the action, whereas statements about "unusual" or "indirect" effects are written in other formulae. It seems plausible that this resembles how we tend to describe the effects of actions in common-sense terms. From an operational point of view there are several advantages with such a reformulation: the action laws will be more easily legible, and circumstances which are shared between several action laws can be stated just once rather than repeatedly inside each law. I shall use the terms *main action law* and *supplementary laws* for the two types of statements.

The main action law will be assumed to have the same syntactical form as was exemplified for plain action laws above. In particular it defines a certain number of features which are affected by the action, and (constraints on) their values during and at the end of the action. On this assumption the supplementary laws serve to characterize three phenomena:

- *Surprises:* Unusual changes in additional features besides those stated in the main law.
- *Ramification:* Normal changes in additional features, for example due to cause-effect chains.
- *Qualification:* Unusual exceptions from the main law, whereby some of the features change in another fashion (including the case of no change at all) than what is specified in the main law.

Clearly there is a certain arbitrariness as to the separation between main law and supplementary laws, and these three categories are only well defined for a given choice of main laws.

I distinguish, therefore, between the basic issue of inertia whose analysis was outlined in the previous section, and the three modularization-oriented issues of surprises, ramification, and qualification. These are essentially the same as the traditionally recognized aspects of the "frame problem". My claim here is that *the traditional frame problem can be analyzed rigorously in terms of simple inertia* (the $\mathcal{K}$-**IA** class of reasoning problems, as defined in the previous section) *plus the decomposition of action laws into main action laws and supplementary laws.*

The division that is used here differs somewhat from the usual one in the literature, since surprises are usually considered as an aspect of the "inertia problem" or of the "ramification problem", and have not been identified as a separate issue in its own right. Apart from that, the distinction between the problems of inertia, ramification, and qualification is the generally accepted one.

3.2 The systematic perspective

In order to analyze the distinction between main action laws and supplementary laws, it is not sufficient to rewrite each action law to a classically equivalent formula. Two other and stronger methods will be used:

- *Premise transformation:* Transformations on the whole chronicle, i.e. on the sets of premises including transfer of information between the premise partitions, in ways which do not change the set of selected models.

- *Extensions of the semantics:* Modifications of the underlying semantics, and corresponding addition of more premise partitions as well as definition of entailment criteria that use the additional partitions.

The shift towards common-sense-motivated forms for action laws is performed in the first case by rewriting the specific set of premises to an equivalent form, and in the second case by changing the logic itself.

Surprises and qualifications involve notions of genuine preferences and defaults: in a choice between a model that involves a surprise and one that does not, one should prefer the latter. They can therefore not be accomodated in the trajectory semantics for $\mathcal{K}$-**IA**, so an extension of the semantics as well as additional specialities are needed. I will identify the phenomenon of surprises with the speciality code **S**, and analyze qualifications in terms of a concept of normalcy which has the speciality code **N**.

Ramifications offer a more complex picture. One possible viewpoint is to exclude ramifications from the underlying semantics. The trajectory-semantics definition of the world is retained like for $\mathcal{K}$-**IA**, where it is assumed that the trajectories for a given action and starting state characterize *all* changes of feature values. The ramification problem is then only an issue about how a given world description, in trajectory semantics, can be correctly expressed using a combination of main action laws and supplementary laws. In such a *semantics-preserving* approach to ramification, premise transformations are the natural instrument.

The other possible viewpoint is that the world description of the form $\langle$`Infl`, `Trajs`$\rangle$ that is used for $\mathcal{K}$-**IA** shall only correspond to the main action laws, and that some other device is needed in the underlying semantics corresponding to the supplementary laws. This means in particular that the definition of the "game" between ego and world becomes more complex. The world's moves can no longer be described just in terms of selecting and imposing one trajectory; there must also be some *update policy* about how to modify the finite history with secondary changes corresponding to the supplementary laws. This *semantics-extending* view of ramification opens a number of possibilities, and several additional specialities are required to account for ramification in this view.

3.3 Recycling vs. embedding of entailment criteria

Since an extension of the underlying semantics will be needed for surprises, qualification, and some approaches to ramification, we shall repeatedly encounter situations where a number of entailment criteria have been defined and analyzed for a family $\mathcal{Z}$ of reasoning problems, and similar criteria are needed for a larger family $\mathcal{Z}' \supseteq \mathcal{Z}$. For example, we may wish to use our entailment criteria for $\mathcal{K}$-**IA** as a basis for criteria for $\mathcal{K}$-**IAS**. There are two possible ways of proceeding:

- To check whether some of the known methods for $\mathcal{Z}$ will apply also to the extended family, or at least to some part of it which can be defined using a sub-speciality. For example, maybe some method for $\mathcal{K}$-**IA** will also work for a family $\mathcal{K}$-**IAS**x for some sub-speciality x under the speciality **S**. The existing criterion is *recycled*[1] to apply for the additional speciality.

- To define a more complex entailment criterion, for example by introducing an additional premise category or an additional preference relation. The expression for the selected models according to the new criterion contains the model-selection expression for the old criterion as a sub-expression. The new criterion is formed as an *embedding* of the existing one.

As a rule of thumb, the use of recycling ought to be preferred since it avoids the need for introducing more complexity in entailment criteria. However it is not always possible to proceed by recycling, so embedding may sometimes be the only possibility besides inventing an entirely new entailment criterion. If and when entailment methods are chosen on the basis of representative examples, one must be particularly careful not to choose a recycled criterion merely because it works for a few examples in the extended family.

4 Brief overview of assessments for surprises, ramification, and qualification

4.1 Surprises

It is clear that surprises should be minimized non-chronologically, and that it is the cardinality or the combined weight of the surprises that is to be minimized rather than the set of surprises. The only exception is that if one can assume that there is at most one surprise, then of course minimization of cardinality and of set extension is equivalent.

Since the minimization is to be done non-chronologically, then the recycling of approaches for $\mathcal{K}$-**IA** to apply for $\mathcal{K}$-**IAS** will only work for non-chronological minimization methods. However such methods have an extremely limited range of applicability. In practice it is therefore only of interest to use embedding approaches, where one makes a separate minimization of the surprises included in the models, thereby increasing by one the number of selection functions that occur in the entailment criterion.

4.2 Ramification

The "ramification problem" has been defined from the point of view of common-sense-motivated formulations for action laws, and concerns how one shall take

[1] The term "extended" would also have been appropriate, but is already used for other purposes in non-monotonic logics.

care of all the normal, indirect effects of an action, where the *direct* effects of the action are by definition those that are expressed using the main action law. I will recognize three types of phenomena within this general definition:

- Changes which occur after the termination of the action, as a result of cause-effect chains with delays. This case will be counted to the **L** speciality for delayed effects, and is not being considered in this paper.

- Changes in features of objects which are not included as arguments of the action. This phenomenon will be referred to as *dependencies* or *structural ramification*. The full definition of the semantics for $\mathcal{K}$-**IA** makes an explicit assumption that there are no dependencies, and leaves the case with dependencies to the broader family $\mathcal{K}$-**IAD** of reasoning problems. Dependencies will not be further discussed here.

- Changes in additional features of objects which are included as arguments of the action, including changes in features without arguments. This will be considered as the speciality of *local ramification*, named by the speciality code **U**, and is the topic of the present subsection.

The concrete treatment of ramification in the A.I. literature is restricted to the case of local ramification. The distinction between structural and local ramification is quite important in the underlying semantics, since the proofs for assessment of currently used entailment criteria depend critically on the assumption that there are no dependencies.

One possible way of motivating ramification and of formulating supplementary laws is in terms of *constraints* i.e. propositions which are supposed to hold at all times. Constraints may have the form

$$\Box\kappa$$

where $\Box$ represents "always" and κ is a fluent formula with certain syntactic restrictions, expressing a condition that must hold momentarily at each timepoint and which does not refer to previous timepoints or to current change. Constraints of that form will be said to be *synchronic*. In the existing literature, most authors identify ramification with the use of synchronic constraints.

4.2.1 Semantics-preserving approaches

accomodate constraints on the syntactic level, but without changing the underlying semantics or the entailment criteria. Then the constraints do not define any additional ontological speciality. Let us consider this possibility in the basic case of the $\mathcal{K}$-**IA** ontological family, with trajectory semantics. The underlying semantics defines the world in terms of a pair $\langle \mathtt{Infl}, \mathtt{Trajs} \rangle$. If the underlying semantics is unchanged, the trajectories in $\mathtt{Trajs}(A, r)$ still represent *all* changes of feature values when an action A is performed from the starting state r. Even within this rigid framework, there are two reasonable uses for constraints. *Active constraints* represent an approach to ramification; *passive constraints* actually have nothing to do with ramification and are instead a kind of derived invariants.

In the context of ramification, the purpose of constraint formulae which are used for active constraints is as an economy measure when writing out action laws. Suppose one is given a $\mathcal{K}$-**IA** world where the discrete state domain $\mathcal{R}_I$ contains only a subset of all the feature assignments that are allowed by the logic. For example, if a represents "it is dead" and d represents "it is alive", then a feature assignment where both a and d are assigned the value $\mathtt{T}$ is not included in the set $\mathcal{R}_I$ – it is not considered to ever occur. The constraint $\Box\,\kappa$ characterizes this state of affairs precisely if $\mathcal{R}_I$ equals the model set for κ.

In such a case, all trajectories in $\mathtt{Trajs}(A, r)$ will be consistent with the constraints. The full trajectory normal form for the world has been assumed to enumerate all changes explicitly, but one can also consider an equivalent set of premises consisting of maximally simple action laws, which are intended to be complemented by the separate statement of the constraints. In each action law the consequent will just consist of the appropriate occlusion statements[2] for all f in $\mathtt{Infl}(A, r)$, combined with the minimal amount of other information about the action's consequences. The additional constraints on the values of the occluded (= possibly changing) fluents is provided by the supplementary laws which are constraint formulae.

4.2.2 Semantics-extending approaches

may deal with constraints in a manner similar to surprises. The basic logic for inertia, i.e. for the ontological family $\mathcal{K}$-**IA**, defines in a very restrictive way what changes are possible, but if these definitions are applied without exception then one obtains a contradiction. In the case of surprises it was the observations that generated a contradiction and a need for escape; in the case of ramification it is instead the constraints that create this need.

Using again the "dead xor alive" constraint

$\Box\,(a \Leftrightarrow \neg d)$

for an example, suppose one has also the observation

$[10]\, a \wedge \neg d,$

saying that a is true and d is false at time 10, as well as the action statement

$[10, 12]\, Kill,$

saying that the "Kill" action takes place over the interval from time 10 to time 12, and the action law

$[s, t]\, Kill \Rrightarrow [s, t]\, a := \mathtt{F}$

which does not imply any possibility of change in d. Under the assumptions of $\mathcal{K}$-**IA** one will obtain an empty intended-model set, and an additional device in

[2]Statements about the possibility of change, whose only purpose is to override the default of inertia.

the underlying semantics is required in order to save the situation.

The presently most popular choice of that additional device is to use "minimization of change" or more precisely the minimization of the set of changes. However a precise analysis shows that this update policy has serious flaws which seem impossible to repair except by very drastic changes to the underlying logic.

4.3 Normality

The speciality of normality concerns what is otherwise known as the qualification problem, i.e. the cases where some of the changes that are stated in the main action law do not take place or take place differently than stated there. The current A.I. literature only contains one concrete reasoning example of qualification, namely the "potato in the tailpipe" example, and surprisingly few proposals for how to deal with the problem. Also it is quite easy to find counterexamples to those approaches, such as the "tailpipe marauder" example.

At this point I can not report any positive results about assessed entailment methods for the speciality of normality. However the underlying semantics and the other aspects of the systematic methodology has at least made it possible to define the qualification problem in more precise terms than before.

5 Conclusion

For many years, the research on common sense reasoning has been troubled by the choice between two evils. A lot of the reported research has been formal, precise, but apparently of little relevance to actual common-sense reasoning problems. A lot of other research has addressed actual problems, but using a very ad hoc methodology in particular by relying too much on representative examples of common-sense reasoning.

With the systematic methodology which is now emerging for this research, it becomes possible at last to make precise and formally proven statements about whole classes of problems and about the general applicability of logic variants, and not merely about single instances of reasoning problems. As one would expect in a formal approach to any problem area, the systematic methodology starts by analyzing a relatively simple case in depth, namely the $\mathcal{K}$-**IA** class of reasoning problems, and then builds on it in order to analyze successively broader and more useful classes of problems. In this way we can gradually address the plethora of logical complexities which have been assembled under the term "the frame problem", but with a firm and reliable basis at each step in the analysis.